

**PROGRAMA DE APOIO A REALIZAÇÃO DE PROPOSTAS  
COMERCIAIS**

Eurico André Mendes Da Silva Clemente

Projeto submetido como requisito parcial para obtenção do grau de Mestre em  
Gestão empresarial

Orientador:

Prof. Doutor Carlos Fontão de Carvalho, Prof. Auxiliar, ISCTE Business School,  
Departamento de contabilidade

Coorientador(a):

Dr. Eduardo Eusébio, Diretor do Instituto Superior de Engenharia de Lisboa

Fevereiro 2019

## **Agradecimentos**

Em primeiro lugar, quero agradecer aos professores que aceitarem o desafio de orientação deste projeto e por toda a disponibilidade demonstrada em ajudar-me sempre que precisei, bem como responder a todas as minhas dúvidas, mas acima de tudo pelo espírito de excelência que me inculcaram. A vossa disponibilidade, dedicação e espírito crítico não serão esquecidos.

Em segundo lugar, quero atribuir parte do sucesso deste projeto ao pessoal do secretariado e restantes departamentos de apoio (muitas vezes esquecidos), pela sua paciência nas dúvidas protocolares da realização deste documento.

Por último, mas não menos importante, quero agradecer à escola de negócios AESE pela disponibilidade de contacto a CEO's de empresas de destaque no tecido empresarial português, permitindo compreender, quais as suas necessidades de apoio (criação de programa) para reforçar/colmatar a anomalia identificada.

## Resumo

Em qualquer empresa, a área comercial assume um papel preponderante, uma vez que é através desta que as empresas comunicam com o mercado e com os seus clientes ou potenciais clientes, angariando, assim, negócio para a empresa. Existem setores de atividade, como é o caso do setor da construção, em que os respetivos departamentos comerciais assumem um peso significativo na estrutura, verificando-se, muitas, vezes situações de *overworking* (Hedley, G. 2017). Este projeto tem como objetivo responder às necessidades sentidas por estas empresas, durante a realização das suas propostas comerciais, bem como abranger os seus estímulos exteriores não controláveis pelas mesmas (atualizações, revisões, entre outros), no registo de inovar para o sucesso Caird (2002).

Para a realização deste projeto, foram estudados os parâmetros, fórmulas e variáveis matemáticas, de forma a implementar expressões de cálculo específicas, para a realização deste programa de apoio à realização de propostas comerciais (utilizando os ensinamentos da IBS), sendo as expressões de cálculo (matemáticas) referidas o culminar da investigação desenvolvida.

Na execução do programa referido foram ainda abordadas linguagem de programação, tais como *Java*, *SQL* e *HTML*, cuja conceção foi baseada nas definições identificadas por Abreu (2017), Damas (2017) e Eckel (2006). Estando em anexo (CD), os ficheiros referidos. Nota para o facto deste programa ser executado de forma a ser suportado numa grande variedade de plataformas (androide, Windows, ios, outros)

Este programa permite, também, aos comerciais (*sales force*), que muitas vezes não se encontram presentes (fisicamente) na empresa, devido a realizarem um contacto próximo com os clientes, o acesso a informação presente na sede (*company headquarters*), em tempo real, representando, assim, um grande avanço tecnológico.

## Palavras-chave

Programa de apoio;  
Mudança Tecnológica;  
Expressões de cálculo;  
Inovação.

## **Abstract**

In any company, the commercial area assumes a predominant role, since it is through this, that companies communicate with the market and with their customers or potential customers, thus raising business to the company. There are sectors of activity, such as the construction sector, where the respective commercial departments take on a significant weight in the structure, often being overworked situations (Hedley, G. 2017).

This project aims to respond to the needs felt by these companies and their commercial DP as well as to cover external forces (can not be controlled by them, TM).

In order to carry out this project, the needs of mathematical formulas were studied in order to implement specific calculations (support program - commercial proposals), using the IBS lessons.

In this program, I also used programming languages such as Java, SQL and HTML, based on their design definitions - Abreu (2017), Damas (2017) and Eckel (2006). Note to the fact that this program runs on a wide variety of platforms (android, windows, ios, others).

This program also allows sales force and they own representatives, access to information of them headquarters in real time, without physically presence, representing a great technological advance for small companies.

## **Keywords**

assistance program;  
Technological change;  
Mathematical calculations;  
Innovation.



# ÍNDICE

<b>Capítulo 1. Introdução .....</b>	<b>11</b>
1.1 - Enquadramento.....	12
1.2 - Objetivo do Estudo.....	13
1.3 - Motivação.....	14
1.4 - Organização do Texto .....	14
1.5- Notação.....	15
<b>Capítulo 2. Revisão da literatura .....</b>	<b>17</b>
2 – Revisão da literatura e softwares .....	18
2.1 – Estrutura .....	18
2.1.1 – Atualização métodos de trabalho.....	19
2.2 – Comparação com concorrência do mercado.....	20
2.3 – Sustentabilidade.....	21
2.4 – Legislação e normas .....	21
2.5 – Variáveis e cálculos .....	22
2.5.1 – Elementos a otimizar .....	24
2.5.1.1 – Fator tempo.....	24
2.5.1.2 – Fator monetário .....	25
2.6 – Cronograma .....	25
<b>Capítulo 3. Metodologia e funcionamento.....</b>	<b>27</b>
3 – Metodologia e funcionamento .....	28
3.1 – Processamento informação .....	29
3.1.1 – <i>Input</i> .....	29
3.1.2 – Cálculos e <i>Output</i> .....	30
3.2 – Conceção da simulação <i>back-end</i> .....	30
3.2.1 – Base para identificação da proposta .....	34
3.2.2 – Bases de cálculo para orçamentação .....	35
3.2.3 – Bases Auxiliares .....	36
3.2.4 – Bases Contentores .....	37
3.2.5 – Base para identificação dos elementos referentes ao mapa de quantidades .....	38
3.2.6 – Base para resumo da orçamentação.....	39
3.2.7 – Restantes elementos de cálculo .....	40

3.3 – Conceção da simulação <i>front-end</i> .....	41
3.3.1 – Criação de propostas ( <i>front-end</i> ) .....	42
3.3.1.1 – Base para identificação da proposta .....	43
3.3.1.2 – Base de cálculo para orçamentação .....	44
3.3.1.3 – Base Auxiliar .....	45
3.3.1.4 – Base Contentores .....	46
3.3.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades .....	47
<b>Capítulo 4. Modus operandi e user interface .....</b>	<b>49</b>
4 – <i>Modus operandi</i> e <i>user interface</i> .....	50
4.1 – Introdução ao UI ( <i>user interface</i> ) .....	51
4.1.1 – <i>Management</i> .....	52
4.1.2 – <i>Materials</i> .....	55
4.1.3 – <i>Proposal</i> .....	58
4.2 – Tutorial .....	60
4.2.1 – <i>Login</i> ou <i>logout</i> .....	62
4.2.1 – <i>Management</i> .....	62
4.3 – Caso de estudo .....	65
<b>Capítulo 5. Conclusões e melhoramentos .....</b>	<b>67</b>
5.1 - Principais conclusões .....	68
5.2 - Possíveis Melhoramentos .....	69
<b>Capítulo 6. Referências bibliográficas .....</b>	<b>71</b>
Referências bibliográficas .....	72

# ÍNDICE

Referências

Nota: As figuras, tabelas e expressões são apresentadas com referência ao capítulo a que pertencem e são numeradas de ordem sequencial, dentro do respectivo capítulo. Todas as tabelas, figuras e expressões com referência à letra “a” referenciam-se os anexos (apêndice 2, anexo 2.1).

## Tabelas

Tabela 2.2.1 - Cronograma ( Data concussão etapas - previstas).....	26
Tabela 3.1 - Softwares (Comparação com alternativas do mercado) .....	28
Tabela 3.2 - Conceção da simulação (explicativa e abreviada).....	31
Tabela 3.3 - Folha de cálculo (Excel).....	33
Tabela 4.1 - Tabela Tutorial .....	61

## Figuras

Figura 1.1 - Constituição empresarial genérica (departamento comercial).....	12
Figura 2.1 - Contexto da utilização .....	18
Figura 2.2 - Estrutura do simulador.....	19
Figura 2.3 - Organização estrutural das variáveis ( <i>Input/Output</i> ) .....	23
Figura 2.4 - Exemplo linhas de código utilizadas (janela) .....	23
Figura 3.1 - Diagrama interligação <i>Input/Output</i> .....	29
Figura 3.2 - Implementação conceito (informático).....	31
Figura 3.3 - Exemplo linhas de código utilizadas (Base 1) .....	35
Figura 3.4 - Exemplo linhas de código utilizadas (Base 2) .....	36
Figura 3.5 - Exemplo linhas de código utilizadas (Base 3) .....	37
Figura 3.6 - Exemplo linhas de código utilizadas (Base 4) .....	38
Figura 3.7 - Exemplo linhas de código utilizadas (Base 5) .....	39
Figura 3.8 - Exemplo linhas de código utilizadas (Base 6) .....	40
Figura 3.9 - Exemplo linhas de código utilizadas (Base 7) .....	41
Figura 3.10 - Analogia com sistema mecânico do quotidiano.....	42
Figura 3.11 - Exemplo de um preenchimento da base referente à identificação da proposta ( <i>front-end</i> ).....	44

Figura 3.12 - Exemplo de um preenchimento da base referente a orçamentação da proposta ( <i>front-end</i> ).....	45
Figura 3.13 - Exemplo de um preenchimento da base auxiliar ( <i>front-end</i> ).....	46
Figura 3.14 - Exemplo de um preenchimento da base contentores ( <i>front-end</i> ).....	47
Figura 3.15 - Exemplo de um preenchimento da base materiais ( <i>front-end</i> ) .....	48
Figura 4.1 - Identificação da localização bases UI ( <i>Proposal details</i> ).....	50
Figura 4.2 - Componente da repartição UI.....	51
Figura 4.3 - Composição vertente UI (aplicada) .....	52
Figura 4.4 - Janela <i>New Client</i> .....	53
Figura 4.5 - Janela <i>Client Details</i> .....	54
Figura 4.6 - Janela <i>Client Results</i> .....	55
Figura 4.7 - Janela <i>New Material</i> .....	56
Figura 4.8 - Janela <i>Material Details</i> .....	57
Figura 4.9 - Janela <i>Material Results</i> .....	57
Figura 4.10 - Janela <i>New Proposal</i> .....	58
Figura 4.11 - Janela <i>Proposal Details</i> .....	59
Figura 4.12 - Janela <i>Proposal Results</i> .....	60
Figura 4.13 - Janela programa destaque canto superior direito ( <i>user</i> ).....	62
Figura 4.14 - Janela programa destaque botão <i>Management</i> .....	63
Figura 4.15 - Janela programa destaque botão <i>Management (units)</i> .....	64
Figura 4.16 - Janela programa destaque botão <i>create new units</i> .....	65

## Apêndices

<b>Apêndice 1 - Identificação das variáveis.....</b>	<b>75</b>
1.1 – <i>Input</i> .....	77
1.2 - Cálculos e <i>Output</i> .....	83
<b>Apêndice 2 – Formulas de cálculo (índice) .....</b>	<b>93</b>
2.1 – <i>Input</i> .....	99
2.1.1 - Bases para identificação da proposta ( <i>Input</i> ).....	101
2.1.2 - Bases de cálculo para orçamentação ( <i>Input</i> ).....	105
2.1.3 - Bases Auxiliares ( <i>Input</i> ).....	109
2.1.4 - Bases Contentores ( <i>Input</i> ).....	113
2.1.5 - Bases para identificação dos elementos referentes ao mapa de quantidades ( <i>Material</i> ).....	117
2.2 - Cálculos e <i>Output</i> .....	131

2.2.1 - Bases Auxiliares.....	133
2.2.2 - Bases Contentores.....	143
2.2.3 - Bases para identificação dos elementos referentes ao mapa de quantidades ( <i>Material</i> ).....	149
2.2.4 - Bases para resumo da orçamentação .....	187
2.2.5 - Restantes elementos de cálculo .....	197
<b>Apêndice 3 - Programa linguagem .....</b>	<b>203</b>
3.1 – <i>Back-end</i> .....	205
3.1.1 - Bases para identificação da proposta ( <i>Proposal</i> ) .....	207
3.1.2 - Bases de cálculo para orçamentação ( <i>BudgetBase</i> ) .....	211
3.1.3 - Bases Auxiliares ( <i>AuxiliarBase</i> ) .....	217
3.1.4 - Bases Contentores ( <i>Containers</i> ) .....	219
3.1.5 - Bases para identificação dos elementos (mapa quantidades).....	221
3.2 – front Criação proposta (Ex) .....	225
3.3 – front Criação material (Ex).....	255
<b>Apêndice 4 – Teste de assertividade .....</b>	<b>259</b>
4.1 – Teste de valores (resultados manuais com as formulas automáticas) .....	271
4.2 - Programa de teste aos <i>inputs</i> .....	281
4.3 - Programa de teste aos valores .....	311

## Lista de Abreviaturas

### SIGLAS:

<b>CD</b>	- <i>Compact disc</i>
<b>CEO</b>	- <i>Chief Executive Officer</i>
<b>CCO</b>	- <i>Chief Commercial Officer</i>
<b>COO</b>	- <i>Chief Operation Officer</i>
<b>CSR</b>	- <i>Corporate social responsibility</i>
<b>DPO</b>	- <i>Encarregado de Proteção de Dados</i>
<b>EU</b>	- <i>European union</i>
<b>HTML</b>	- <i>Hyper text markup language</i>
<b>IBS</b>	- <i>ISCTE Business School</i>
<b>RGPD</b>	- <i>Regulamento Geral de Proteção de Dados</i>
<b>SQL</b>	- <i>Structured Query Language</i>
<b>UI</b>	- <i>User interface</i>
<b>VRP</b>	- <i>vehicle routing problem</i>
<b>SI</b>	- <i>Système international d'unités (sistema internacional unidades métricas)</i>
<b>PMEs</b>	- <i>Pequenas e Médias Empresas</i>



---

## CAPÍTULO

# 1

---

### Introdução

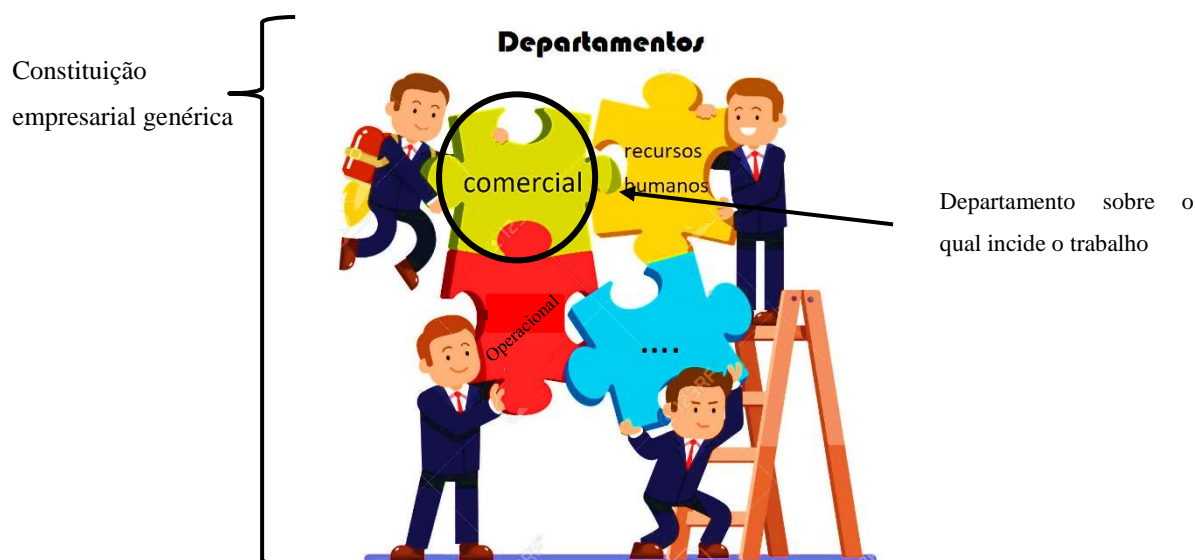
*Neste capítulo é feita uma introdução sucinta sobre o fundamento do presente projeto, focado na elaboração de propostas comerciais e suas anomalias, apresentando, também, uma solução para as colmatar (desenvolvimento programa), assim como a notação utilizada ao longo do trabalho.*



## 1.1 - Enquadramento

Este projeto tem como objetivo a obtenção do Grau-Mestre em gestão de empresas. Com esse objetivo, foi abordado um tema atual de muita importância no seio da comunidade empresarial. Com esse objetivo em mente, foram entrevistados CEO's de empresas portuguesas, tais como Leonor Freitas (Empresa Casa Ermelinda Freitas) e José Luís Simões (Empresa Luis Simoes), por meio de entrevistas ocorridas no âmbito da escola de negócios AESE e seu respetivo programa *summer school*, onde foi permitido a estudantes o contacto direto com vários representantes de empresas nacionais.

Estas entrevistas tiveram como principal objetivo averiguar, quais as necessidades sentidas pelos mesmos nas empresas respetivas (projeto com implementação prática).



**Figura 1.1 - Constituição empresarial genérica (departamento comercial)**

Fonte: Google imagens (alterada, para efeitos de explicação incidência trabalho, não aplicável o código de cores)

A vertente comercial é um dos pilares fundamentais de qualquer empresa, contribuindo, desta forma, para o desenvolvimento da atividade. O foco principal comercial é responder às necessidades (em tempo real) dos seus clientes. Em empresas de fornecimento e mão-de-obra (grande parte das empresas portuguesas), isto significa a elaboração de propostas comerciais (conforme referido na informação/nota Banco de Portugal sobre o Endividamento do setor não financeiro, disponível no site do mesmo).

Desta forma, podemos definir as propostas comerciais como um dos fatores de desenvolvimento das empresas, pois permitem o ampliar da capacidade de negócio, obedecendo sempre às regras da concorrência, às normas/regulamentos e leis aplicáveis.

## 1.2 - Objetivo do Estudo

“Mundo” comercial não é um “objeto” estático. Ao longo dos séculos, o comércio tem acompanhado a evolução tecnológica e criado novas técnicas para uma negociação, afeto às mudanças tecnológicas e comportamentais, como, por exemplo, o *e-commerce*.

Estudo efetuado por *Dionísio et al.* (2012), defende a condição referida (necessária adaptação constante à realidade).

Contudo, na vertente empresarial do comércio, o avanço tecnológico implica em muitos casos a necessidade de contratação de bens materiais ou laborais (como bastidores informáticos ou programas específicos). Isto implica uma transação monetária (Henriques, A. 2016).

O presente trabalho tem por objetivo a elaboração de um programa de apoio às empresas, que permita a realização de estudos sobre as propostas comerciais e perspectivas sobre as suas infraestruturas, nos mais variados contextos, tendo sempre atenção, ser elaborado de forma a que os utilizadores disponham de uma plataforma de fácil interação, bem como de livre utilização (sem custo).

Neste trabalho pretende-se refletir também sobre a orçamentação e otimização de todo o processo conducente à mesma, em tudo o que se refere à determinação de valores de custo de materiais, mão-de-obra, fórmulas e revisão de preços. O programa produzirá um documento com o resumo detalhado do estudo para utilização junto de empresas ou clientes, bem como para o respetivo utilizador.

Este programa será elaborado, através de ferramentas computacionais e matemáticas. Em suma, esta dissertação visa evidenciar a importância das propostas comerciais de uma empresa e a sua aplicação em contexto exemplificativo, que facilite e otimize o processo de orçamentação.

## 1.3 - Motivação

A motivação para a realização deste estudo deve-se ao facto de, na minha experiência profissional, ter efetuado uma passagem numa empresa de orçamentação, a empresa “H.seabra”, na qual havia um constante esforço para acompanhar e realizar todas as propostas comerciais. Apesar de não ter sido identificado nenhum estudo sobre isso, pela experiência adquirida nesta empresa é possível afirmar que apenas um terço das propostas obtinha resposta. Tal se devia às dificuldades de realização de um orçamento preciso e elaborado em tempo útil. Conforme referido anteriormente, em entrevistas realizadas com CEO’s de empresas portuguesas (Casa Ermelinda Freitas, Luís Simões, entre outras) foi possível averiguar o elevado nível de importância destas propostas para as empresas respetivas.

Em alguns ramos empresariais, como a construção, os montantes envolvidos nas propostas são bastante elevados (casa dos milhares ou milhões de euros), conforme indicado na nota de informação estatística sobre a análise do setor da construção do *Banco de Portugal (2016)* e respetivas notas. Por este motivo, podemos afirmar que este projeto representa uma solução empresarial, permitindo aos diretores comerciais uma mais fácil gestão do seu departamento.

Desta forma, pretende-se com o projeto apresentado, fornecer uma nova base/pilar para a realização de propostas.

## 1.4 - Organização do Texto

A organização deste documento encontra-se repartida em seis capítulos, de forma a separar os vários tópicos abordados, permitindo uma mais fácil leitura do mesmo, sendo que na parte final deste documento estarão expressos os apêndices respetivos.

O texto está organizado em capítulos, sendo atribuído um tópico a cada um; o primeiro corresponde à Introdução.

O segundo capítulo está dividido em cinco partes distintas, permitindo ao leitor uma subdivisão da revisão de literatura, na qual são apresentados os pontos focais do *software* (programa informático), bem como a estrutura dos seus cálculos dedicados.

No terceiro capítulo estão presentes a Metodologia, os parâmetros do *back-end* e *front-end*, bem como a identificação/explicação das variáveis *Inputs* e *Outputs*.

Já no quarto capítulo é apresentado o modo de utilização e um caso de estudo, demonstrando, desta forma ao leitor o *modus operandi* de todo o sistema e fazendo a análise de informação referente os módulos do *UI* (*user interface*).

Por sua vez, no quinto capítulo estão expressas as conclusões, assim como as possibilidades para desenvolvimentos futuros do projeto identificado e suas limitações.

Por fim, é apresentada a bibliografia utilizada, seguida dos respetivos apêndices (utilizando os padrões IBS).

## 1.5 - Notação

As figuras, tabelas e expressões são apresentadas com referência ao capítulo a que pertencem e são numeradas de ordem sequencial no respetivo capítulo. A identificação de expressões é apresentada entre parênteses curvos “( )”, e a identificação de referências bibliográficas é apresentada segundo as regras deferidas pela IBS (bibliografia consultada).

Expressões em língua estrangeira, siglas e referências são apresentadas em itálico.

Os restantes elementos foram executados, tendo por base o referido, durante o seminário de investigação.

Devido ao elevado grau de complexidade foram ainda criados códigos de cor:

Elementos pertencentes a base n.º 1 – verde claro (- -)

Elementos pertencentes a base n.º 2 – castanho claro (- -)

Elementos pertencentes a base n.º 3 – amarela (- -)

Elementos pertencentes a base n.º 4 – azul claro (- -)

Elementos pertencentes a base n.º 5 – vermelho (- -)

Elementos pertencentes a base n.º 6 – violeta (- -)

Elementos pertencentes a base n.º 7 – castanho escuro (- -)

Sendo as restantes cores, como, exemplo, o verde escuro, para elementos identificáveis, sem ligação ao código de cores.

---

## CAPÍTULO

# 2

---

### **Revisão da literatura**

*No 2º capítulo está a revisão da literatura, sendo abordados/descritos softwares dedicados a este tipo de solução, bem como o tema da otimização das respostas aos pedidos de cotação.*

## 2 – Revisão da literatura e software

No presente capítulo será efetuada a revisão de literatura relativamente ao tema proposto, sendo que esta está dividida em subtópicos, de modo a facilitar a navegação no próprio documento, bem como a identificação dos vários assuntos dentro do mesmo tema.

Neste capítulo estão incluídos os elementos relativos à componente de carácter geral (revisão da literatura, sendo abordados/descritos softwares dedicados a este tipo de solução, bem como o tema da otimização das respostas aos pedidos de cotação)

### 2.1 – Estrutura

O programa aqui apresentado representa um avanço tecnológico, em comparação com os processos atuais utilizados. Segundo os estudos de *Cristia et al. (2017)*, a capacidade para a adaptação às novas tecnologias surge “naturalmente” ao ser humano.

O método para construção de programas deste tipo é algo padronizado na comunidade, permitindo, futuramente, a realização de atualizações por terceiros (pessoas não envolvidas no desenvolvimento direto do programa), com alguma facilidade (programa aberto).



**Figura 2.1 - Contexto da utilização**

Fonte: Próprio (Eurico Clemente)

Desta forma, o programa referido segue os padrões expostos em anexo, o que permite uma adaptação interna a qualquer empresa (não utilizando linguagem proprietária).

Este tipo de arquitetura teve por base o defendido por *Giffin, Levy, Stefan, Terei, Mazières, Mitchell e Russo* (2017).

Foi também tida em atenção a relação entre entidades como ponto de foco. Devido a isso foram consultadas várias plataformas (físicas e digitais) de forma a responder com uma configuração mais clara e convenientemente possível às necessidades dos utilizadores alvos, ponderando o referido por *Ebert* (2014) e *Brindley* (2004).

### 2.1.1 – Atualização métodos de trabalho

Conforme defendo anteriormente nesta tese, o facto deste programa ser executado no século atual, permite ao mesmo representar uma mudança nas tecnologias utilizadas, culminando numa ferramenta *web* e permitindo, desta forma, uma fácil filtragem dos documentos e uma mais rápida utilização.

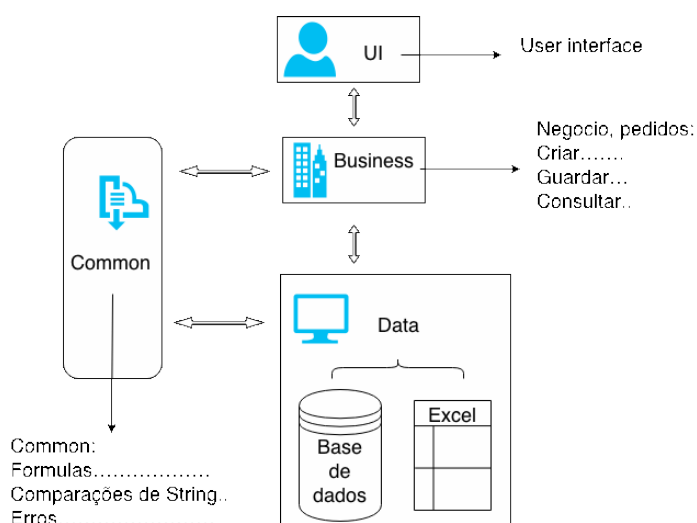


Figura 2.2 - Estrutura do simulador

Fonte: Próprio (Eurico Clemente)



Desta forma, o *UI (user interface)* é criado para permitir ao normal utilizador (utilizador sem conhecimentos de métodos de programação) uma interação *User Friendly*.

Nota ainda para o facto de poderem ser adicionados utilizadores novos, permitindo ao representante da empresa (COO ou CCO), uma interação próxima com os seus colaboradores, aumentando, desta forma, a comunicação interna, conforme defendido por *Smidts, Pruyn e Van Riel* (2001).

## **2.2 – Comparação com concorrência do mercado**

No mercado abordado, existem alguns *players* (empresas do sector da programação), que oferecem soluções distintas da referida (abordagens diferentes), tais como o *proposalsASS*, utilizado por empresas como a *PLM*, ou mesmo o *OneStream XF* na sua vertente *oneXF*. Contudo estas soluções não são as ideais e representam um investimento muito avultado, que grande parte das empresas portuguesas não suportam (o referido é verificado a título de exemplo no software *Prisma4*).

De forma a ultrapassar estas barreiras, o programa referido é totalmente livre de utilização e permite responder a várias necessidades, não abordadas nos restantes programas, como, por exemplo, fórmulas de cálculo para fornecimento de material em países estrangeiros (alteração de moeda de referência), culminado numa atmosfera de custo/benefício superior a estes e permitindo desta forma às empresas de pequena dimensão responder de modo adequado aos pedidos dos seus clientes.

Assim, podemos definir este programa, como o ponto de viragem, para a grande parte das empresas no ramo, fazendo um acompanhamento e trazendo uma novidade para o mercado, não esquecendo os modelos (Software e configuração) apontados por *Maria, et al.* (2017), *Gonçalves* (2010) e *Alyrio* (2011), de forma a definir os pontos relevantes.

## 2.3 – Sustentabilidade

Este programa constitui uma contribuição positiva para o mercado das *PME's* (pequenas e médias empresas) uma vez que estas são os principais favorecidos, devido ao carácter *free* deste programa.

Segundo o *Banco de Portugal* (2017), nos anos que precederam a recente crise económica e financeira, observou-se uma rápida acumulação de dívida por parte destas sociedades, a qual está agora em fase de recuperação.

Assim, podemos definir o carácter social deste projeto, que, para além do seu grau de desenvolvimento (gestão e engenharia), representa o ideal defendido por *Lins, Servaes e Tamayo* (2017) e pela *CSR* (*Corporate social responsibility*) da *EU* (*European union*), contribuindo para uma sociedade melhor.

A melhoria alcançada pelas empresas onde é utilizado, representa um fator de orgulho pessoal, bem como dos restantes elementos envolvidos, tais como orientador e co-orientador, contribuindo para a construção de um futuro "empresarial sustentável".

## 2.4 – Legislação e normas

Tendo em conta que o "*total target*" abrangido pelo produto é variado, desde pessoas singulares até entidades coletivas, temos de frisar, que, para realização deste projeto, foram efetuadas revisões de literatura, relativamente à legislação aplicada, bem com às normas PT e EN, dando lugar de destaque ao *RGPD* (Regulamento Geral de Proteção de Dados).

Sendo que o novo regulamento reveste-se de alguma complexidade, com novos princípios e conceitos, dando direitos para os titulares de dados, que significam novos deveres para as empresas que com eles lidam. A avaliação de impacto, a privacidade na conceção de novos produtos ou serviços com dados, a privacidade por defeito, as notificações das violações de segurança e a figura do *DPO* (Encarregado de Proteção de Dados), são alguns dos exemplos.

Uma vez que este projeto é universitário e todos os dados estarão no servidor “principal” da entidade administradora do sistema, fica a seu cargo toda a gestão deste subtópico.

## 2.5 – Variáveis e cálculos

A relação entre a economia e a sociedade é visível, estando presente já há vários séculos na nossa sociedade, conforme defendido por *Akamatsu* (1962), e utilizando formas de se desenvolver, por via de campos muitas vezes relacionados com a matemática.

Tendo por base os ensinamentos da IBS, foi possível verificar, quais os cálculos necessários para o projeto, bem como proceder à sua execução. Esta tarefa exaustiva teve de ser realizada em parceria com o orientador e coorientador, devido ao seu grau de complexidade. Destaque, também para a ajuda dos restantes professores, durante a realização do projeto referido.

Para um bom funcionamento do nosso sistema a nível de gestão matemática (programa), temos de definir quais as variáveis a realizar *Input* e *Ouput*, bem como elaborar os cálculos (gestão empresarial) aplicados. Os cálculos referidos são a *master key* (ponto chave) de todo o projeto, representando o *back-end* do programa (Figura 2.1). Este fio condutor liga a gestão à engenharia, numa única plataforma/sistema.

Sendo que todos os cálculos estão identificados no anexo respetivo (apêndice 2, anexo 2.1), as variáveis de *input* estão identificadas como a letra “i” e as de *ouput* com a letra “o”, culminando na totalidade com vinte e sete variáveis de *input* e cento e quarenta de *Ouput* (apêndice 1, anexo 1.1 e 1.2).

Para o referido, foram utilizadas 15.924 linhas de código, as quais se convertem, na prática, em fórmulas matemáticas elaboradas na consulta dos documentos de Puccini (2009), *Gonçalo* (2015) entre outros. Estes ensinamentos levaram à elaboração da organização estrutural das variáveis, conforme abaixo exposto.

Nota ainda, para o facto da estrutura estar repartida em cinco pontos, permitindo a divisão dos cálculos e, desta forma, executar os exemplos de teste e manutenção. Estes pontos estão definidos como “bases”, sendo que as variáveis dos vários pontos se posam inter-relacionar.

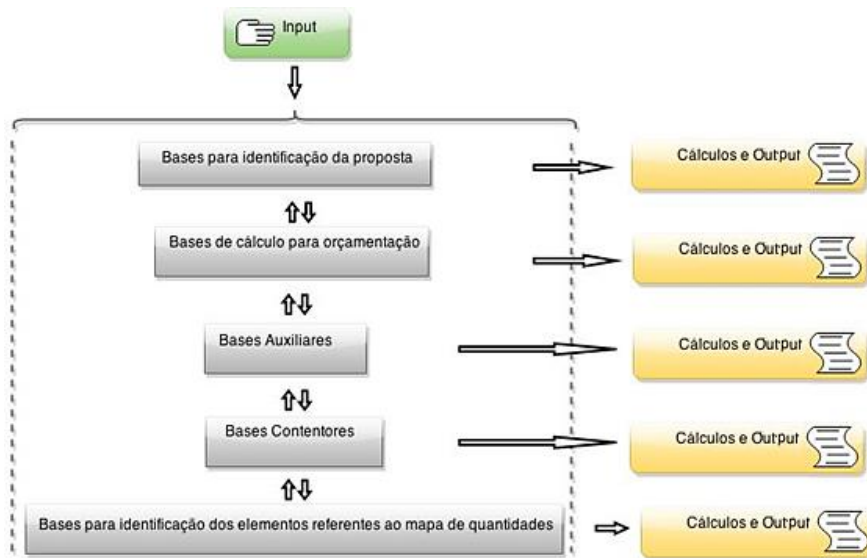


Figura 2.3 - Organização estrutural das variáveis (Input/Output)

Fonte: Próprio (Eurico Clemente)

Importa ainda salientar que está em anexo um exemplo de cálculo (apêndice 4, anexo 4.1), de forma a permitir ao leitor examinar toda a gestão das fórmulas utilizadas, que são passíveis de alterar pelo administrador do sistema. É de realçar que estes cálculos são o pilar de todo o projeto, resultando na sua boa implementação.

```

package com.factus.observatorio.dbo;

import java.util.*;

/**
 * Created by Eurico on 13/12/2017.
 * Last changed: 01/01/2018.
 * ----- 3.2.1.4 - Bases para identificação dos elementos referentes ao mapa de quantidades -----
 *
 * @see:
 * Atenção aos inputs, pois se for um orçamento para o próprio país as variáveis 10,112,113,115,116,117,118,
 * e 119 devem ser automaticamente 133, usando o 131 e 132;
 */
public class Material {

    private long materialId;

    private EnumType elementId; /* 131 - Tipo de elemento*/
    private EnumSubType elementSubType; /* 132 - Subtipo de elemento*/
    private long materialCode; /* 133 - Contém o material number code */ private String
    private EnumType unitType; /* 135 - Contém o material unit type */
    private double quantity; /* 136 - Contém o material quantity */
    private double price; /* 137 - Unit value (material) */
    private double priceInvoicing; /* 138 - Unit value of invoicing */
    private double priceTransport; /* 139 - Value for the transportation (equipment) */
}

```

Figura 2.4 - Exemplo linhas de código utilizadas (janela)

Fonte: Próprio (Eurico Clemente)

A programação representa apenas o meio utilizado para permitir uma interação fácil para o utilizador, contudo os cálculos matemáticos e a respetiva gestão são os elementos a destacar.

## 2.5.1 – Elementos a otimizar

Neste capítulo são evidenciados dois tipos de elementos a otimizar com a realização deste programa, o fator tempo e o fator monetário, permitindo que as empresas aumentem a sua oportunidade de criar valor. Estes elementos são otimizados, devido aos cálculos efetuados pelo próprio programa (variáveis *Input* e *Ouput*) referidos anteriormente.

Segundo *Uchoa, et al. (2016)*, utilizando a abordagem correta é possível solucionar elementos de otimização complexos, tais como *VRP (vehicle routing problem)*. Seguindo os ensinamentos de que a abordagem correta pode solucionar problemas complexos, utilizámos a nossa solução para responder aos seguintes elementos:

### 2.5.1.1 – Fator tempo

Este fator representa a constante temporal e possui três dimensões distintas, que serão abordadas separadamente:

Oportunidade (*Time Windows*): No sentido em que cada pedido de cotação é referente a uma única proposta, sendo esta uma oportunidade de negócio. Pela minha experiência profissional, posso afirmar que eram respondidos, em média, apenas um terço das propostas de cotação solicitadas (não foram identificados estudos para o referido). Neste sentido, pode-se assinalar a oportunidade como um dos elementos a otimizar.

Capacidade de espera: Esta capacidade representa o tempo para elaboração da respetiva propostas, uma vez que cada pedido vem com um tempo máximo de resposta (quer positiva ou negativa). Em relação a este elemento, pode-se indicar como sendo a variável relacionada com o período compreendido entre a receção do pedido de cotação e a sua resposta.

Sequência: No sentido em que todas as propostas devem ser acompanhadas e executadas e, em caso de necessidade, elaboradas, posteriormente, as suas revisões/atualizações.

## 2.5.1.2 – Fator monetário

O fator monetário está diretamente relacionado com o valor monetário. Este valor é expresso numericamente e está associado a uma mercadoria/serviço. O conceito de preço é global, mas a sua unidade tem variantes em vários países/continentes, pelo que é possível adicionar ou eliminar novas variantes.

Erros de cálculo: Para executar uma proposta, muitas vezes são feitos inúmeros cálculos. Com este programa, todas as fórmulas são calculadas automaticamente, não existindo interação humana, sendo por isso de esperar uma diminuição dos erros envolvidos

Gestão de materiais: Representa um dos campos apresentados no programa (*Material*), onde são permitidas alterações por parte do utilizador ou administrador, de forma a criar, padronizar e alterar o respetivo elemento de forma a responder as suas necessidades, diminuindo, desta forma, o tempo gasto (aumento indireto do fator monetário).

## 2.6 – Cronograma

O projeto referido teve como data prevista de conclusão setembro 2018. Devido ao seu elevado grau de complexidade, foi planeado, de forma detalhada, e foi executado escrupulosamente, para não colocar em risco o projeto (conclusão em setembro, conforme exposto).

Para este objetivo, o projeto encontra-se dividido num conjunto de bases, em que cada uma representa um tópico diferente. As bases encontram-se numeradas e representadas pelo código de cores respetivo (conforme referido anteriormente), estando as tarefas destinadas a serem realizadas no seu nome técnico.

**Tabela 2.2.1 - Cronograma (Data concussão etapas - previstas)**

Tarefas	2017 (ou anterior)			2018								
	Outubro (ou anterior)	Novembro	Dezembro	Janeiro	Fevereiro	Març	Abril	Mai	Junho	Julho	Agost	Setembr
<i>Submissão de formulário (online)</i>	-											
<i>Revisão de literatura</i>		-	-									
<i>Recolha de informação</i>	-	-	-									
<i>Teste de variáis</i>			-									
<i>Teste de fórmulas matemáticas</i>				-								
<i>Base nº 1</i>	-	-	-									
<i>Base nº 2</i>	-	-	-									
<i>Base nº 3</i>	-	-	-									
<i>Base nº 4</i>	-	-	-									
<i>Base nº 5</i>	-	-	-									
<i>Base nº 6</i>	-	-	-									
<i>Base nº 7</i>	-	-	-	-								
<i>Teste de aplicação sql</i>				-								
<i>Teste de aplicação java</i>				-	-	-						
<i>Teste de aplicação html</i>				-	-	-						
<i>Teste final programa</i>					-	-						
<i>Aplicação prática e instalação máquina</i>					-	-	-					
<i>Revisão da tese e análise com orientador</i>							-	-	-			
<i>Conclusão da tese</i>									-	-	-	
<i>Submissão da tese de mestrado</i>												-

Fonte: Próprio (Eurico Clemente)

Destaque ainda para o facto da execução deste projeto nas suas várias vertentes (matemática e informática) ter sido executado em blocos, permitindo efetuar os testes necessários para a validação de resultados.

Todas as etapas, bem como a sua própria cadeia de sucessão, foram escolhidas de forma determinada e sempre tendo em atenção a aplicação do resultado final (programa), que cumpre o defendido nos capítulos 1.1 e 1.2 (programa prático e de aplicação gratuita, permitindo as empresas disporem de uma ferramenta de apoio ao seu departamento comercial, na respetiva criação ou apoio de propostas).

---

## CAPÍTULO

# 3

---

### **Metodologia e funcionamento**

*Neste capítulo está apresentada a metodologia, o back-end e o front-end do programa referido. Nota para o facto de ser utilizada informação de carácter técnico (para a leitura deste capítulo é aconselhada a leitura de livros sobre a elaboração e instalação de programas matemáticos)*



### 3 – Metodologia e funcionamento

O desenvolvimento do presente projeto de mestrado tem como objetivo (em parceria com o referido anteriormente) a criação de uma plataforma de apoio às empresas (departamentos comerciais), permitindo uma análise das propostas, bem como facilitar a elaboração destas. Para isto foram analisadas alternativas existentes no mercado, tendo sido abordados os aspetos positivos e negativos da concorrência e da própria análise crítica (interna), culminando na realização da plataforma referida (Northcut, et al. 2009).

**Tabela 3.1 - Softwares (Comparação com alternativas do mercado)**

<i>Softwares</i>		
<b>Classificações a comparar</b>	<i>Mercado/alternativo (*)</i>	<i>Objetivo/realizado</i>
<b>User Friendly</b>	Não	Sim (possibilita que o software seja utilizado por um leque variado de pessoas sem as mesmas terem instrução para tal tarefa)
<b>Aspeto gráfico</b>	Fraco	Bom
<b>Funcionalidade</b>	Bom	Médio
<b>Conteúdos</b>	Abrangente (constituído por diversos módulos: Obras, Materiais, Pessoal, Equipamentos, Orçamentos e Autos de Medição)	Médio (Constituído por 3 módulos, com o objetivo de facilitar as operações intra-módulos)
<b>Tempo de processamento</b>	Médio	Baixo
<b>Variedade de plataformas (androide, Windows, ios, outros)</b>	Alto	Alto
<b>Banco de dados</b>	Alto	Alto
<b>Licença de Software</b>	Pagamento	Free

(\*) – Foram analisadas plataformas referidas no capítulo 2.2.

Tendo como ideal o referido, foram criadas todas as condições para os critérios apresentados serem realizados, permitindo criar uma plataforma, que responde ao referido acima.

## 3.1 – Processamento informação

Uma vez que este projeto tem como pilar um conjunto de fórmulas matemáticas, temos de verificar, quais os parâmetros de entrada e saída (variáveis), necessários para a correta realização dos cálculos pretendidos.

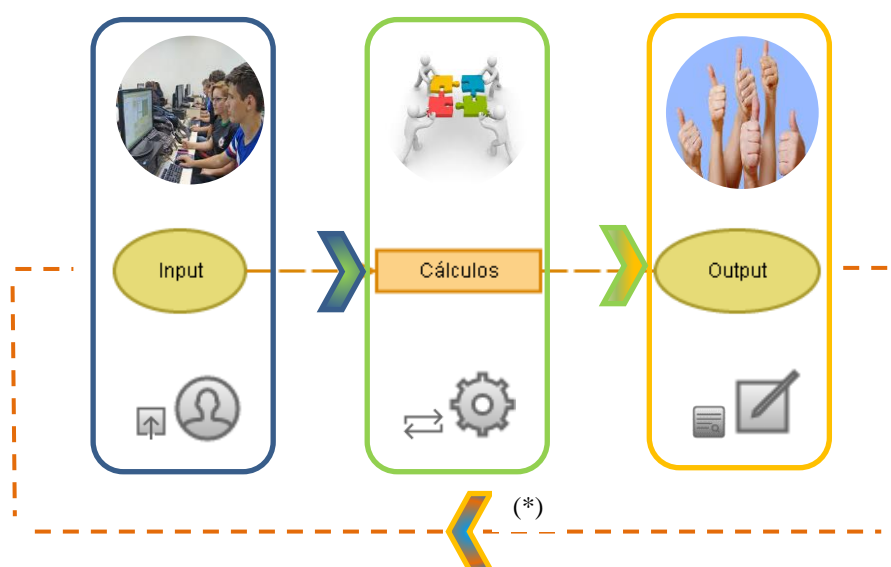


Figura 3.1 - Diagrama interligação *Input/Output*

Fonte: Próprio (Eurico Clemente)

(\*) – Em alguns casos, o *Output* (proveniente de uma fórmula de cálculo) é utilizado como *Input* noutra fórmula de cálculo

### 3.1.1 – Input

*Input* é o termo utilizado internacionalmente (termo de referência), que denomina qualquer tipo de entrada ou mudanças, que são inseridos num sistema, como, por exemplo, os nomes, as datas, os preços ou outros, permitindo ativar ou modificar um processo. É um conceito abstrato, usado na exploração de modelagem/sistemas, e está habitualmente associado a outros termos (em português), como, por exemplo, o campo de entrada, variável de entrada e parâmetro de entrada.

Deste modo, é possível afirmar que o *Input* define-se como o conjunto de variáveis que são atribuídas pelo utilizador. Estas mesmas variáveis estão presentes na elaboração do relatório e respetivos cálculos (apêndice 1, anexo 1.1).

### **3.1.2 – Cálculos e *Output***

Os cálculos têm um papel crucial, uma vez que são estes que permitem relacionar as entradas (*inputs*) com as saídas (*outputs*). Estes cálculos representam a conversão dos dois tipos de variáveis, manobrando matematicamente todo o processo, para que o utilizador tenha na sua posse uma ferramenta simplificada.

Por seu lado, o *Output* é o termo que denomina qualquer tipo de saída, referente a um sistema, como, por exemplo, as denominações, as distâncias, os cálculos técnicos ou outros. É um conceito abstrato usado na exploração de modelagem/sistemas, estando, habitualmente, associado a outros termos (em português), tais como, o campo de saída, a variável de saída e o parâmetro de saída.

Neste sentido, o *output* define-se como o conjunto de informações alusivas "à saída" de um sistema (organismo, mecanismo ou digital). Estas mesmas variáveis estão presentes na elaboração do relatório e respetivos cálculos (apêndice 1 anexo 1.2).

## **3.2 – Conceção da simulação *back-end***

Conforme defendido no capítulo 2.1, referente à estrutura do programa, o mesmo desagrega-se em duas componentes o *front-end* e *back-end* (esta divisão é clássica para qualquer elemento, entre quem trabalha com desenvolvimento *web*).

**Tabela 3.2 - Conceção da simulação (explicativa e abreviada)**

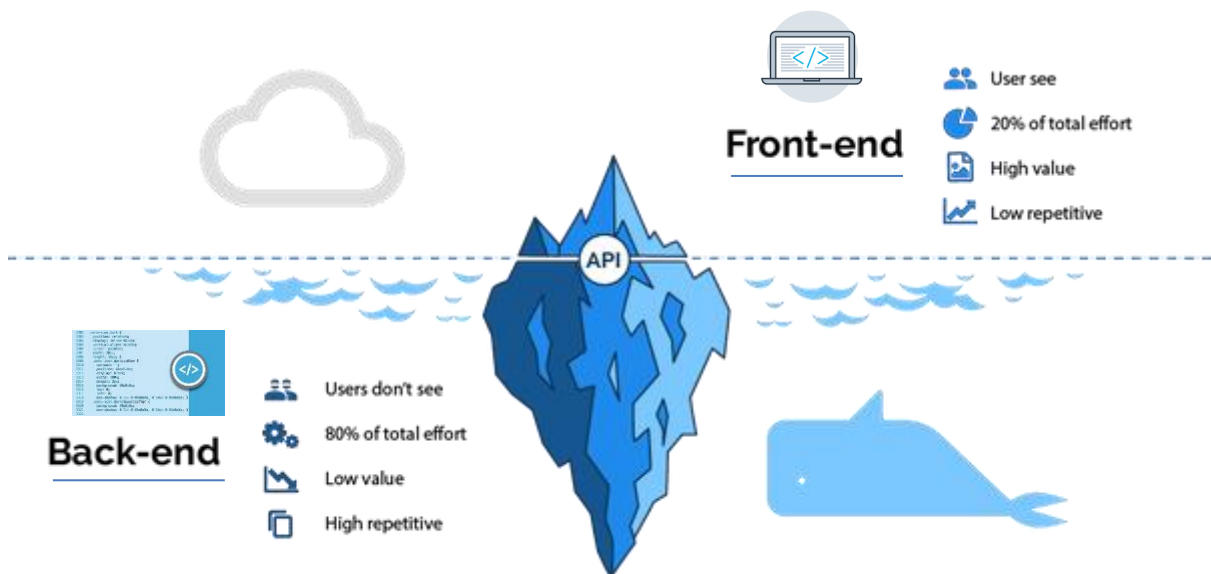
Elementos identificáveis (Conceção da simulação)	Repartição (organizacional texto)	Exemplos (*)
<i>front-end</i>	Componente/campos	<i>Management</i>
<i>back-end</i>	Bases	Bases Contentores

(\*) Exemplos utilizados, pertencentes à execução do programa

(\*) explicação sucinta

O *front-end* representa aquilo que o utilizador observa e interage, ou seja, a interface gráfica. Como o nome sugere, a componente *back-end* trabalha na parte de "trás" dessa interface. Ela é a responsável, em termos gerais, pela implementação da “regra do programa” (fórmulas matemáticas entre outros), vista do lado do servidor.

No caso de um icebergue (exemplo prático), podemos comparar o conceito na imagem abaixo.



**Figura 3.2 - Implementação conceito (informático)**

Fonte: Google imagens (alterada, para efeitos de explicação do *back-end*)

Para ser executado corretamente o *back-end* do projeto (figura 2.1), foram, inicialmente, projetadas e testadas todas as componentes matemáticas, utilizando a ferramenta de cálculo

distribuída pelo *Microsoft Office* (Excel). No decorrer da conceção destas simulações foram elaboradas/identificadas as diversas bases (conjunto de elementos específicos, referentes às variáveis *Input* e *Ouput*), bem como outras necessárias à leitura da informação, sendo que estas estão identificadas em anexo respetivo (apêndice 2, anexo 2.1 e 2.2).

Desta forma, foram criadas divisões, com o objetivo de facilitar a criação e leitura das propostas, culminando na sua totalidade com sete bases/tópicos.

É de assinalar que há cinco bases para a criação das propostas, sendo que as divisões referidas (bases) encontram-se abaixo identificadas (bases para criação da proposta, compreendidas entre 1 e 5), tendo em atenção que esta separação (criação propostas) em cinco bases distintas permite repartir a informação inserida pelo utilizador, facilitando a compreensão dos *Inputs*.

Tendo como suporte a ideia acima exposta de repartir a informação (com o objetivo de facilitar o utilizador), foram também criadas mais duas bases (bases para leitura da proposta, compreendidas entre 6 e 7), tornando mais fácil a sua compreensão dos *Ouput* (\*).

Foram identificados os assuntos/tópicos específicos para cada base (criação de proposta):

- 1 - Bases para identificação da proposta;
- 2 - Bases de cálculo para orçamentação;
- 3 - Bases Auxiliares;
- 4 - Bases Contentores;
- 5 - Bases para identificação dos elementos referentes ao mapa de quantidades.

Foram também identificadas as bases de leitura das propostas:

- 6 - Bases para resumo da orçamentação;
- 7 - Restantes elementos de cálculo (elementos necessários a correta realização dos cálculos).

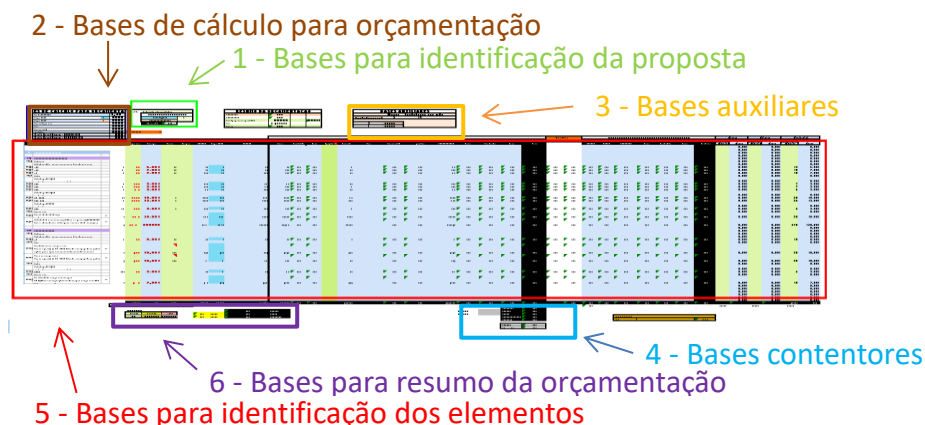
\*nota nº1 para o facto da leitura das propostas na base 6, também incluir as bases de criação (compreendidas entre 1-5);

\*nota nº2 as cores referidas acima (número da base) acompanham a mesma, durante todo o documento, facilitando, deste modo, a orientação espacial do leitor.

Importa frisar o facto de a base “restantes elementos de cálculo” (leitura da proposta) apenas se encontra presente no programa na sua vertente *back-end*.

Todos os cálculos internos das bases e inter-relação entre as mesmas, foram, exaustivamente, testados, durante as várias fases de desenvolvimento do programa, conforme o anexo respectivo (apêndice 4).

**Tabela 3.3 - Folha de cálculo (Excel)**



A tabela acima exposta representa o primeiro passo do desenvolvimento do programa referido, pois esta ferramenta (Excel) representa a coluna vertebral, sobre a qual foram desenvolvidas as fórmulas matemáticas e testadas as variáveis respectivas. A folha de cálculo referida encontra-se no CD que foi entregue. Todas as fórmulas de cálculo concebidas estão identificadas no anexo respectivo (apêndice 2).

Nota importante para o leitor, é a possibilidade de adaptar todas as bases, por meio do administrador (pessoa encarregada por manter e operar a sua rede), conforme referido no capítulo 2.5.

Sendo que se define como administrador (físico) a pessoa ou conjunto de pessoas membros do departamento de Tecnologia da Informação (TI), encarregados de instalar, suportar, manter servidores ou outros sistemas e planear para responder a indisponibilidades de serviços e outros problemas relacionados.

Outras tarefas incluem também programação de *scripts* ou programas pequenos, o gerenciamento de componentes para projetos relacionados a sistemas, supervisionamento ou treino de técnicos de operação e de apoio.

Por meio dos administradores, é possível enquadrar o programa apresentado no decorrer desta dissertação num futuro imprevisível (futuro próxima ou futuros distantes), permitindo à empresa abordar mentalidades estratégicas para períodos temporais ampliados.

Relembramos o leitor que o utilizador deste programa (utilizador e não administrador) apenas tem acesso ao *UI*. Estando o código informático a correr de forma automática, sem intervenção direta deste.

### **3.2.1 – Base para identificação da proposta**

Nesta base estão alocadas as variáveis referentes à identificação da proposta no seu carácter geral, tais como, o nome/referência ou mesmo a identificação do cliente destinatário, não esquecendo que todos os elementos (“*print*”) estão em língua inglesa (programa) para facilitar a importação e exportação de dados (apêndice 1, anexo 1.1 e anexo 1.2).

As informações pormenorizadas sobre os *inputs* desta base estão expressas em anexo (apêndice 2, anexo 2.1.1), sendo a sua leitura aconselhada, para entender os procedimentos indigitados da base número um (Bases para identificação da proposta).

Para criar toda esta plataforma, recorreu-se, conforme referido no capítulo 2.5, a várias linguagens de programação. Tendo em conta este leque variado de linguagens e com o objetivo de não subcarregar os anexos, foi colocada apenas uma das referidas linguagens em anexo (apêndice 3, anexo 3.1.1).

O *modus operandi* desta base está exposto no capítulo 4.2 deste projeto de tese, bem como um exemplo de *UI*.

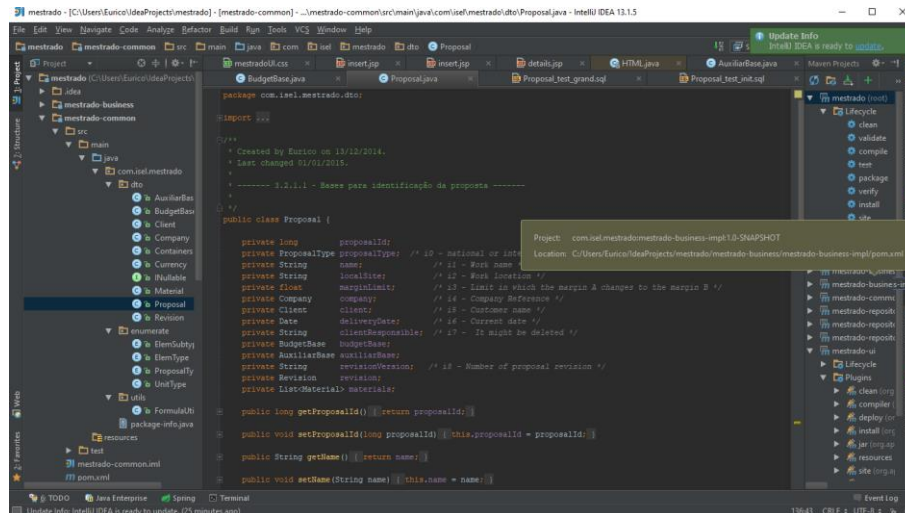


Figura 3.3 - Exemplo linhas de código utilizadas (Base 1)

Fonte: Próprio (Eurico Clemente)

### 3.2.2 – Bases de cálculo para orçamentação

Na base agora referida, estão expostos os elementos principais e de carácter comum, aplicável à proposta alvo, tais como margens e impostos, não esquecendo uma vez mais que todos os elementos estão em língua inglesa (“*print*” programa), para facilitar a importação e exportação de dados.

Neste ponto serão definidos pelo utilizador, quais as imposições e valores a reter na proposta elaborada, não esquecendo que todos os elementos são passíveis de posterior alteração, por meio da revisão de proposta.

A informação referente aos *inputs* da presente base está expressa em anexo (apêndice 2, anexo 2.1.2), sendo a sua leitura aconselhada. Esta base em parceria com a anterior representa uma das cinco bases elaborada para facilitar o apoio à criação de propostas.

Nota para o facto de poderem ser efetuadas alterações dos parâmetros de entrada, através da edição, tendo sempre em mente que o utilizador pode utilizar a unidade internacional de referência, como orientação (*modus operandi* exposto no capítulo 4.2).



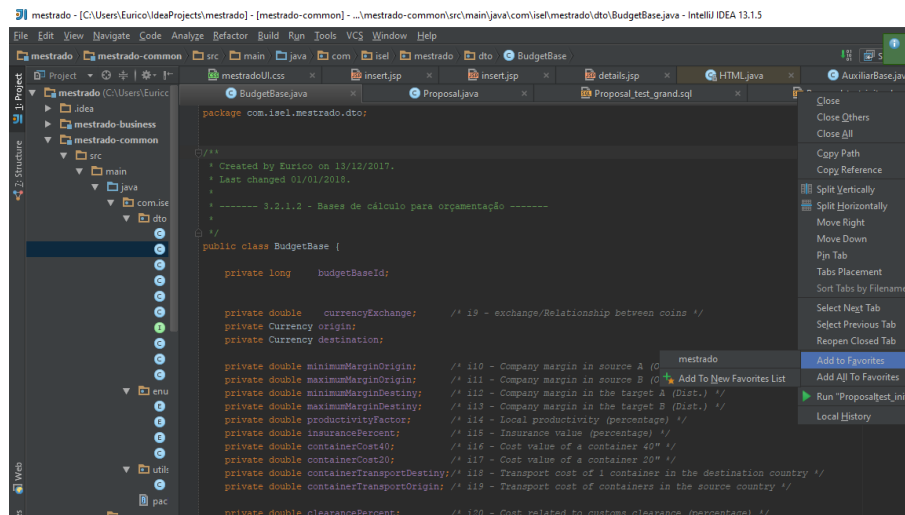


Figura 3.4 - Exemplo linhas de código utilizadas (Base 2)

Fonte: Próprio (Eurico Clemente)

### 3.2.3 – Bases Auxiliares

As bases auxiliares têm como objetivo a gestão/edição da mão-de-obra e tempos da proposta, estando em sua posse as variáveis como o número de dias úteis/trabalháveis por mês. Uma vez mais devido à exportação de dados, todos os elementos alocados à base referida estão expressos em língua inglesa (programa).

Isto representa um aspeto de destaque relativamente aos restantes programas do mercado, pois aborda a componente do fornecimento de mão-de-obra (pessoas qualificadas, para a realização do proposto em orçamento), permitindo, desta forma, realizar os orçamentos de fornecimento material, fornecimento de material e mão-de-obra ou mesmo orçamentos para o fornecimento apenas de mão-de-obra (realizando, desta forma, os três tipos de orçamentos possíveis).

As informações sobre os *inputs* e *outputs* desta base estão expressos no anexo (apêndice 2, anexo 2.1.3 e anexo 2.2.1, respetivamente). Informa-se também o leitor para a necessidade da leitura nos anexos relativos a linguagem de programa utilizada (apêndice 3, anexo 3.1.3) e *modus operandi* da referida base, que se encontra exposto no capítulo 4.2 deste projeto de tese.

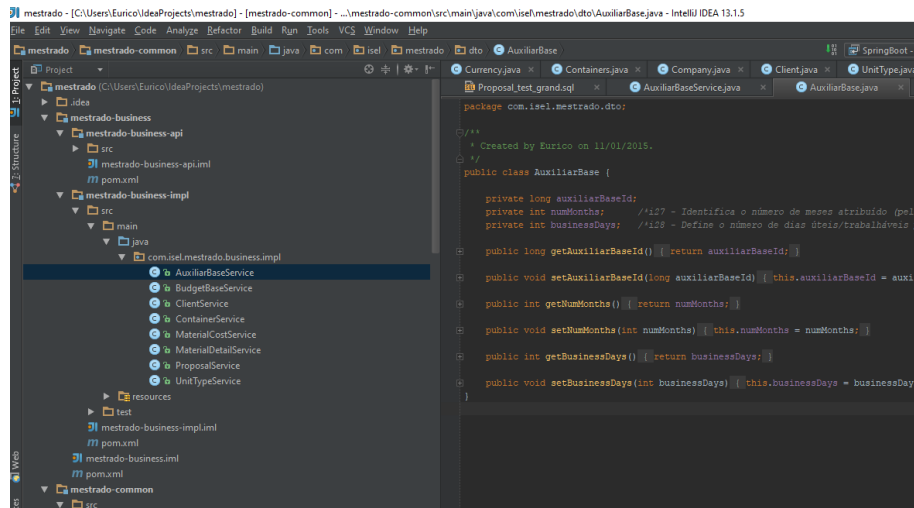


Figura 3.5 - Exemplo linhas de código utilizadas (Base 3)

Fonte: Próprio (Eurico Clemente)

### 3.2.4 – Bases Contentores

A base contentores, como o próprio nome designa, são o conjunto de variáveis alusivas aos contentores de transporte marítimo (caso aplicável). De notar que as variáveis referidas nesta base não são de preenchimento obrigatório, ficando à escolha do utilizador, a sua aplicação.

Esta base tem bastante importância em orçamentos internacionais (país de destino, diferente do país de origem), permitindo até a mais pequena empresa candidatar-se a proposta originárias de outros países (ampliação das fronteiras territoriais pelas empresas utilizadoras do programa).

As variáveis contidas na base agora exposta estão expressas em língua inglesa (em parceria com as restantes bases), sendo aconselhada a leitura do anexo de identificação das variáveis *input* e *output* (apêndice 1, anexo 1.1 e anexo 1.2, respetivamente).

A aplicação do código da base contentores encontra-se em anexo (apêndice 3, anexo 3.1.4) e o seu *modus operandi* no capítulo 4.2 deste projeto de tese, bem como exemplos de UI.

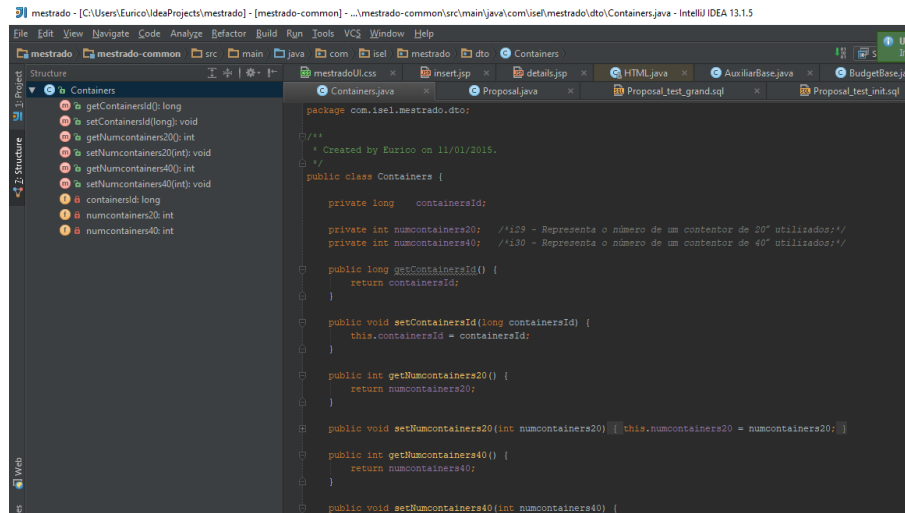


Figura 3.6 - Exemplo linhas de código utilizadas (Base 4)

Fonte: Próprio (Eurico Clemente)

### 3.2.5 – Base para identificação dos elementos referentes ao mapa de quantidades

Esta base (identificação dos elementos referentes ao mapa de quantidades) é por definição o conjunto das bases com a maior extensão, devido às variadas opções de orçamento possíveis (*input* adereçáveis pelo utilizador), estando identificadas as referidas variáveis nos anexos respetivos (apêndice 2, anexo 2.1.5 e 2.2.3, *input* e *output*, respetivamente).

Para compreensão desta base é aconselhada a leitura também do apêndice 1.

A interligação entre bases (matematicamente) acontece na presente base, sendo que todas as variáveis de interligação funcionam de forma automática.

Importante ainda frisar que nesta base são expostos todos os elementos referentes aos materiais.

Uma vez que as variáveis contidas na base estão expostas em língua inglesa, em caso de dúvida deverá ser efetuada a leitura do anexo de identificação das variáveis *input* e *output* (apêndice 1, anexo 1.1 e anexo 1.2).

Informa-se também o leitor que o exemplo de código da base 5 encontra-se em anexo (apêndice 3, anexo 3.1.5) e por sua vez o *modus operandi* desta base está exposto no capítulo (4.2).

```

package com.isel.mesclau.dto;

import ...

/**
 * Created by Eurico on 19/12/2014.
 * Last changed 01/01/2015.
 *
 * ----- 3.2.1.4 - Bases para identificação dos elementos referentes ao mapa de quantidades -----
 *
 * NOTA:
 * Atenção aos inputs, pois se for um orçamento para o próprio país as variáveis 119,112,113,115,116,117,1
 * Atenção preenchimento deve ser automatico 133, Usando o 131 e 132;
 */
public class Material {

    private long        materialId;

    private ElementType elemType;           // 131 - Tipo de elemento//
    private ElemSubType elemSubType;       // 132 - Subtipo de elemento//
    private long        numericCode;       // 133 - Contains the material numeric code //
    private String      description;       // 134 - Contains the material description //
    private UnitType    unitType;          // 135 - Contains the material unit type //
    private double      quantity;          // 136 - Contains the material quantity //
    private double      price;             // 137 - Unit value (material) //
    private double      priceAccessories;  // 138 - Unit value of accessories //
    private double      priceTransport;     // 139 - Value for the transportation (equipment) //
    private double      priceLocalDestination; // 140 - Transport price (international only) //
    private int         timeAssembly;      // 141 - Assembly time in minutes equipment type 1 //
    private int         timeAssembly2;     // 142 - Assembly time in minutes equipment type 2 //
}

```

Figura 3.7 - Exemplo linhas de código utilizadas (Base 5)

Fonte: Próprio (Eurico Clemente)

### 3.2.6 – Base para resumo da orçamentação

A base para resumo da orçamentação, como o próprio nome indica, serve para ajudar o leitor a fazer uma fácil leitura de toda a sua orçamentação, fazendo, assim, uma melhor gestão/economia do tempo.

É possível verificar as variáveis de *output*, referentes a esta base no apêndice 2, anexo 2.2.4. Remete-se ainda o leitor deste projeto de tese para a leitura do código informático aplicado, não esquecendo que este projeto tem duas alas (pilares) bastante fortes e complementares a destacar: a matemática e, também, a informática.

Esta base destina-se a permitir que o utilizador possa verificar possíveis pontos de melhoria no orçamento efetuado, bem como ter forma de analisar o mesmo.

No capítulo 4 está exposto o *modus operandi* desta base, tendo em mente que foi executado para ser o mais simples possível, permitindo até às pessoas sem formação na matéria (orçamentação) executar um orçamento preciso.

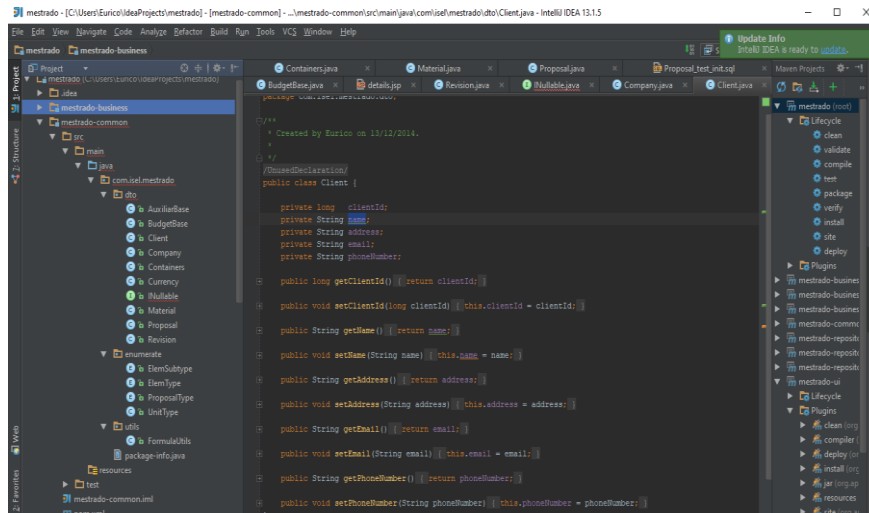


Figura 3.8 - Exemplo linhas de código utilizadas (Base 6)

Fonte: Próprio (Eurico Clemente)

### 3.2.7 – Restantes elementos de cálculo

Neste capítulo/tópico é apresentado o conjunto de cálculos necessários para a boa execução do programa que não faz parte das restantes bases, ou seja, os elementos externos aos cálculos, mas de caráter indispensável ao mesmo.

Todos os elementos referidos na presente base encontram-se em anexo (apêndice 2, anexo 2.2.5) e têm de ser tido em conta na composição do programa. Em parceria com os outros elementos referidos nas bases anteriores, também estes devem ser alvo de leitura pormenorizada.

Informa-se também o leitor deste projeto que será nesta base (base 7), que terão lugar os vários tipos de opção referente aos valores totais das pró-formas possíveis, bem como a criação de subcategorias e outros elementos, como moedas referência ou mesmo unidades de medida SI (unidades de medida reconhecidas pelo sistema internacional).

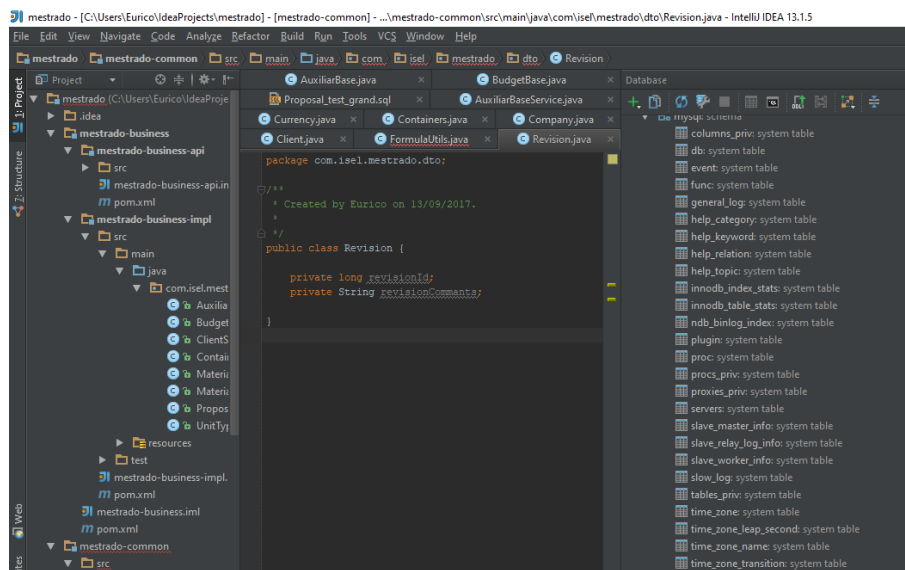


Figura 3.9 - Exemplo linhas de código utilizadas (Base 7)

Fonte: Próprio (Eurico Clemente)

O código presente neste capítulo é referente à componente *back-end* e interrelaciona as componentes informática e matemática.

### 3.3 – Conceção da simulação *front-end*

Em conjunto com o capítulo 3.2, também o capítulo 3.3 (*front-end*) está repartido nas mesmas subpartes, referentes à criação de uma proposta por parte do utilizador e sobre a leitura destas. Nota para o facto de o administrador (programa) poder criar barreiras na criação ou leitura, sendo que estas barreiras são efetuadas, através do endereço de *login* (fazendo seleção da informação disponibilizada aos utilizadores endereçados, permissões à utilização), conforme solicitação pelos CEO’s referidos no capítulo 1.3.

Existe informação que devido ao seu carácter confidencial não deverá ser do conhecimento público da totalidade dos utilizadores. Desta forma, a entidade administradora poderá fazer a regulação da informação escalonada por sectores (grupos de utilizadores) ou pessoas individuais (utilizador individual).

Informamos o leitor deste documento que a dissertação exibida não se limita a ficar pelos conceitos teóricos, colocando na prática todas as apreciações defendidas e respondendo de forma firme ao seu propósito (pelo que é importante a leitura dos anexos apresentados).

Para a compreensão desta conceção, é solicitado ao leitor a verificação dos anexos e a leitura dos documentos apresentados nas referências bibliográficas relativas aos módulos de informática e matemática.

### 3.3.1 – Criação de propostas (*front-end*)

Todos os programas deste tipo (interação do utilizador com uma base de dados “editável”), tem a composição definida nos dois pontos abordados, *back-end* e *front-end*, aludindo o presente capítulo à segunda componente.

As linguagens utilizadas para ambos os conteúdos são diferentes, devido à sua aplicação, estando identificados em anexo alguns exemplos de utilização.

Para uma compreensão deste tópico (sistema aplicável informaticamente), é apresentado uma analogia com um sistema mecânico do quotidiano, facilitando deste modo a compreensão deste tema (pessoas não acostumadas com o tópico apresentado). Para tal, remetemos o leitor para a visualização da imagem a baixo identificada (veículo automóvel).



**Figura 3.10 - Analogia com sistema mecânico do quotidiano**

Fonte: Google imagens

Neste sistema podemos esclarecer o leitor, quanto à importância das duas componentes e identificar o *front-end* como os vidros ou chapa (não aplicável), enquanto o *back-end* neste caso será o motor ou molas do mesmo (não aplicável).

Com o referido (analogia), é possível demonstrar ao leitor que ambos são relevantes e necessários para uma utilização eficaz do sistema (diferenciados, mas afetos ao mesmo sistema).

### **3.3.1.1 – Base para identificação da proposta**

A base para identificação da proposta ou *information* (denominação inglesa presente no programa) tem referido no capítulo 3.2.1 o seu *back-end*, contudo, conforme mencionado no capítulo 2.1, esta contém, também, uma componente de *front-end*, que será agora apresentada.

Adequado este capítulo à respetiva limitação de páginas (imposto pela entidade avaliadora deste projeto IBS) é efetuado um pequeno resumo do exposto em título neste capítulo, estando uma parte do código em anexo respetivo (apêndice 3, anexo 3.1.1), para o qual deverá ser efetuado uma leitura, para compreensão do contexto deste sistema.

Para compreensão deste conceito (*front-end*) por parte de um profissional externo à área (informática), podemos afirmar de forma sumária, que o mesmo se define como responsável por "dar vida" à interface: trabalha com a parte da aplicação, que interage, diretamente, com o utilizador. Destaque para a importância relativa como o sistema se unifica nas duas variantes (*back-end* e *front-end*), permitindo a qualquer utilizador da plataforma usufruir desta, sem ter quaisquer conhecimentos/conceitos do tópico.

Informa-se ainda o leitor, para a visualização da janela indicada a baixo (componente limitada pela cor verde claro), que foi elaborada, tendo como objetivo transmitir um ambiente harmonioso (inserção das variáveis de *input*, acompanhadas de nome e informação extra), facilitando sempre a interação de qualquer utilizador (com formação ou não na área) na criação de uma proposta.



**Figura 3.11 - Exemplo de um preenchimento da base referente a identificação da proposta (*front-end*)**

Fonte: Programa próprio (Eurico Clemente)

A componente de *front-end* referente ao conjunto base para identificação da proposta encontra-se acima exposta, limitada pela fronteira de cor verde claro (- -).

A conceção da simulação e *back-end* estão identificados em anexo respetivo (anexo 3.1.1).

### 3.3.1.2 – Base de cálculo para orçamentação

O presente capítulo é alusivo à vertente de *front-end* referente à base para orçamentação da proposta ou *Budget Base* (denominação inglesa presente no programa). Esta base tem como particularidade dispor de métodos de seleção fazendo com que o utilizador possa optar pelo modo como pretende preencher a proposta.

Nota importante para o facto desta base seguir em série a base anterior (base para identificação da proposta), conforme indicado na figura 2.3, permitindo ao utilizador não perder o encadeamento de raciocínio, necessário para a boa elaboração de uma proposta (minimizando desta forma a necessidade de concentração, ou seja, reduzindo a probabilidade da ocorrência de um possível erro).

The screenshot shows a web browser window with the URL `localhost:8000/proposals/new`. The page title is 'Business Manager'. The navigation menu includes 'Proposals', 'Materials', and 'Management'. The user is logged in as 'User'. The main content area is titled 'Budget Base' and contains the following fields:

Field Name	Value	Unit
Currency Origin	Euro [€]	
Currency Destination	British Pound [£]	
Currency Exchange	i9	
Min. Margin Origin (%)	i10	%
Max. Margin Origin (%)	i11	%
Min. Margin Destination (%)	i13	%
Max. Margin Destination (%)	i14	%
Container Cost (40")	i16	€
Container Cost (20")	i15	€
Internal Transportation Destination	i18	€
Tax Declaration (%)	i20	%
Tax Rights (%)	i21	%
Tax Retention (%)	i22	%
Tax Stamp (%)	i23	%
Man Cost Type 1	i24	€
Man Cost Type 2	i25	€
Man Cost Type 3	i26	€
Productivity Factor	i14	
Insurance (%)	i15	%
Other Costs (%)	i44	%

Below the 'Budget Base' section, the 'Auxiliar Base' section is partially visible and highlighted in yellow.

**Figura 3.12 - Exemplo de um preenchimento da base referente a orçamentação da proposta (*front-end*)**

Fonte: Programa próprio (Eurico Clemente)

A zona limitada pela cor castanha (- -) representa o *front-end*, referente ao conjunto base de cálculo para orçamentação.

Conceção da simulação e *back-end* identificados no anexo 3.1.2.

### 3.3.1.3 – Base Auxiliar

O capítulo agora apresentado, e na sequência dos dois anteriores, é alusivo à vertente de *front-end*, sendo, agora, abordada a base auxiliar, com a denominação no programa de *Auxiliar Base* (denominação inglesa). Esta base tem na sua constituição a conceção da simulação e *back-end* apresentados no anexo 3.1.3.

A zona amarela (- -) delimita a zona alusiva à base referida, dentro da lista vertical de criação de propostas. É de frisar que esta base é alusiva à componente temporal (como, por exemplo, o tempo corrente dos trabalhos solicitado em proposta), sendo esta informação, muito importante, principalmente em propostas de mão-de-obra internacionais.

Para verificação do respetivo valor por parte dos utilizadores aparece a variável *duration* (*output*), no formato de cálculo automático (permitindo ao utilizador uma verificação em tempo real da vertente temporal da orçamentação).

The screenshot shows a web browser window titled 'Business Manager' with the URL 'localhost:8000/proposals/new'. The page has a navigation bar with 'Proposals', 'Materials', and 'Management' menus. The main content area is divided into sections: 'Productivity Factor' (0.9), 'Insurance (%)' (0.03), and 'Other Costs (%)' (0.50). Below these is the 'Auxiliar Base' section, which is highlighted with a yellow dashed border. It contains three input fields: 'Months' (4), 'Working Days per month' (20), and 'Duration [Days]' (80). Underneath is the 'Container Base' section with 'Container 20" Quantity' (1) and 'Container 40" Quantity' (0). At the bottom, there are 'Materials' tabs labeled 'f10', 'f20', and 'f30'.

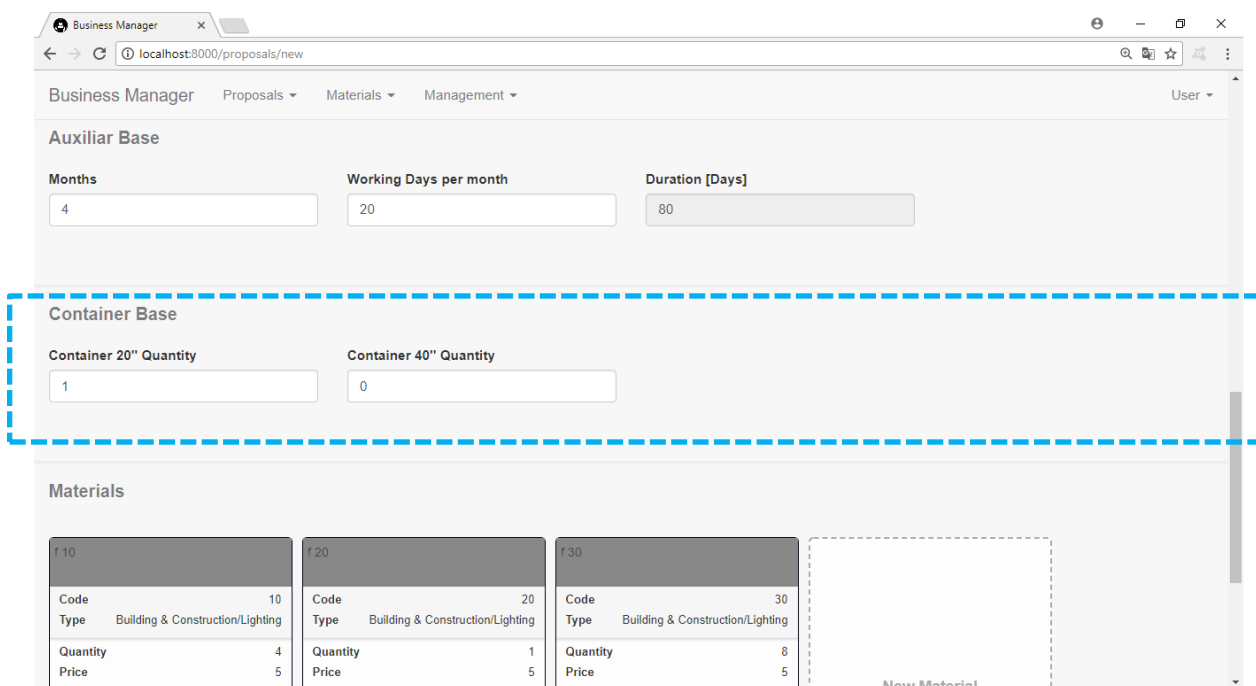
Figura 3.13 - Exemplo de um preenchimento da base auxiliar (*front-end*)

Fonte: Programa próprio (Eurico Clemente)

### 3.3.1.4 – Base Contentores

O atual capítulo é referente à vertente de *front-end*, pertencente à base contentores (transporte marítimo) da proposta ou *Container base* (denominação inglesa presente no programa). Esta base, tem como singularidade, o dispor de dois métodos de transporte marítimos (padrão correspondente ao transporte normalizado internacionalmente).

Todo o código das várias bases foi elaborado, tendo atenção a fácil interação do utilizador e respetiva manipulação de variáveis de *input* e não esquecendo, que grande parte dos utilizadores do programa referido não necessita de formação na área (normal utilizador), estando o administrador (pessoa com formação da área) a efetuar a gestão de contas (*login*) e, as adaptações pontuais no código fonte (programa aberto).



**Figura 3.14 - Exemplo de um preenchimento da base contentores (*front-end*)**

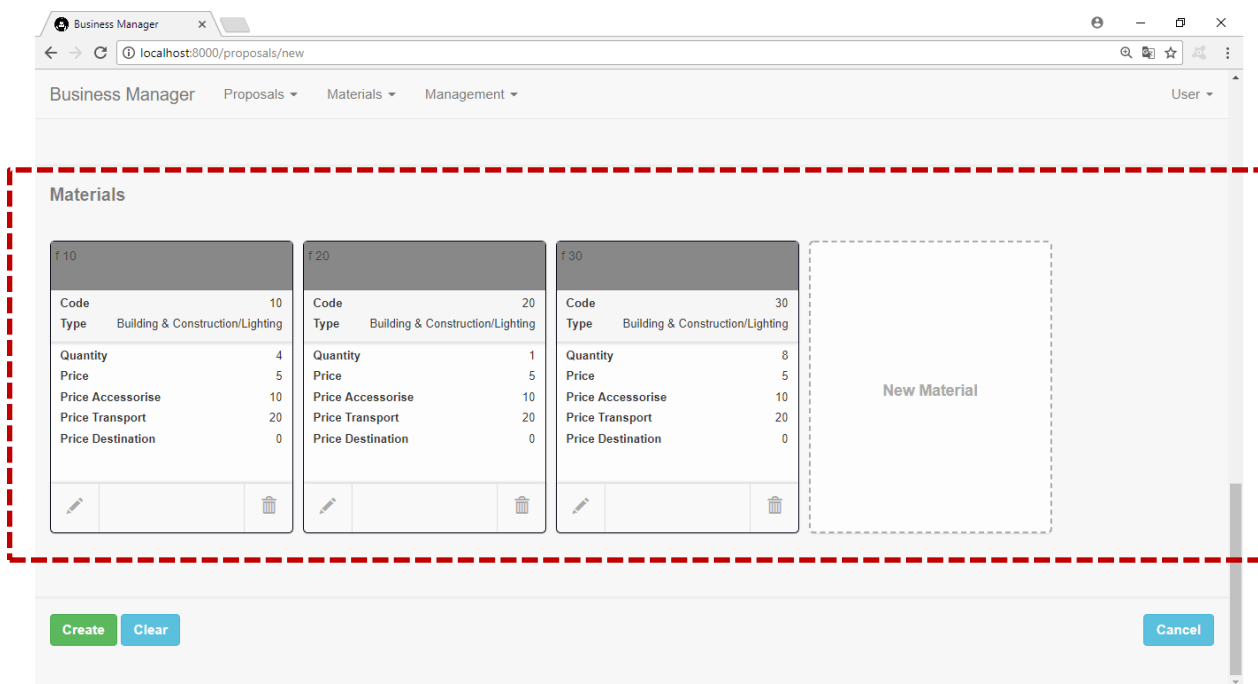
Fonte: Programa próprio (Eurico Clemente)

A zona limitada pela cor azul claro (- -) representa o *front-end*, referente ao conjunto base de cálculo para orçamentação, com os *inputs* identificados no anexo 2.1.4.

### **3.3.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades**

Neste capítulo é apresentada a composição *front-end*, referente à “base para identificação dos elementos referentes ao mapa de quantidades” da proposta ou *Materials* (denominação inglesa presente no programa),

Conforme o próprio nome sugere, nesta base é possível adicionar, remover ou adaptar as matérias incorporados na proposta. Também é possível criar matérias de forma geral (múltiplas propostas) e endereçá-las (pelo meio de apontadores, código informático) à base de dados respetiva.



**Figura 3.15 - Exemplo de um preenchimento da base materiais (*front-end*)**

Fonte: Programa próprio (Eurico Clemente)

A zona limitada pela cor vermelho (- -) representa o *front-end*, referente ao conjunto base para identificação dos elementos referentes ao mapa de quantidades, cujos *inputs* estão identificados no anexo 2.1.5.

---

## CAPÍTULO

# 4

---

### **Modus operandi e user interface**

*No quarto capítulo é apresentado o modo de utilização e um caso de estudo, demonstrando, desta forma, ao leitor o modus operandi de todo o sistema e fazendo, também, a análise de informação referente os módulos do UI (user interface).*

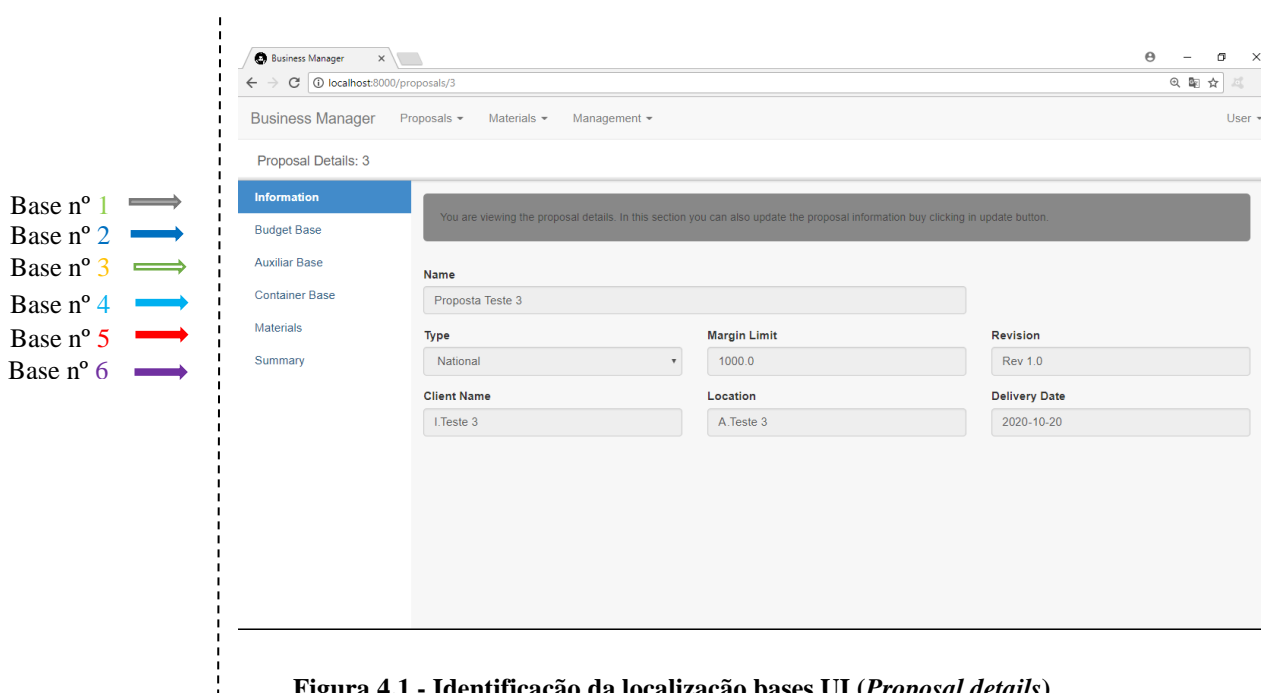
## 4 – *Modus operandi e user interface*

O desenvolvimento do projeto de mestrado teve um UI focado no utilizador comum, sendo que este foi criado, tendo em atenção as possíveis aplicações praticas defendidas, pelo que foi objetivo do estudo a referida constituição visual (Abreu, 2017).

A constituição do agrupamento de código (linguagens de desenvolvimento) dividem esta componente do programa em dois sub-constituintes, primeiro a criação e, posteriormente, a sua respetiva leitura (base de dados SQL).

Sugere-se ao leitor que seja efetuada a leitura do apêndice 3, anexo3.2, para uma melhor compreensão do agora apresentado.

As leituras das propostas em conformidade com a criação, também se encontram divididas por bases (identificadas anteriormente), contudo pode também ser consultada a proposta em formato resumo (forma facilitadora e de leitura rápida). Destaque para as bases de maior peso monetário no orçamento.



**Figura 4.1 - Identificação da localização bases UI (*Proposal details*)**

Fonte: Programa próprio (Eurico Clemente)

## 4.1 – Introdução ao UI (*user interface*)

Para elaboração do UI, foram testadas várias hipóteses, entre as quais, visualizações do *layout* gráfico da interface. Após a elaboração dos possíveis UI's, foi selecionada a que melhor respondia ao pretendido.

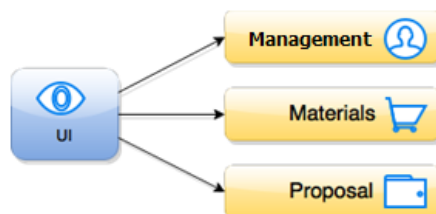


Figura 4.2 - Componente da repartição UI

Fonte: próprio (Eurico Clemente)

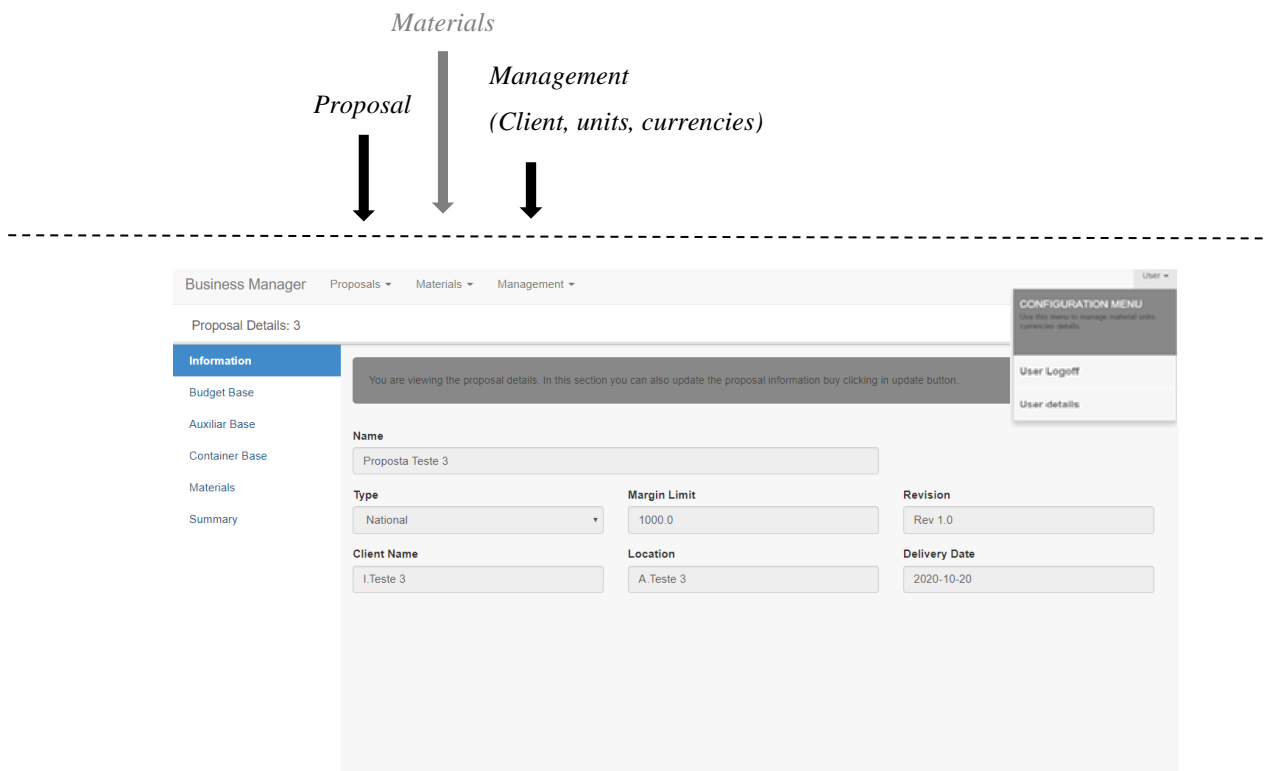
O programa na sua vertente *UI* está repartido em três partes. Esta divisão surge de forma a criar um fácil manuseamento, por parte do utilizador do programa.

A primeira parte é referente aos clientes, unidades (*SI*) e moedas referência e está mencionada como *Management*; a segunda, é alusiva aos materiais e é denominada como *Materials*. A terceira parte apresenta as propostas e tem como nome *Proposal*.

*Management* é um componente físico (ficha unitária, sem possível variação) com um grau de complexidade elevado, ao qual foram adicionados outros campos físico fixos, como o tópico *Client* (opções de endereço/definições optativas da proposta).

Nota para o facto de a componente *Client* encontrar-se dentro do campo *Management*, juntamente com o *units* e *currency*. Todos estes elementos não têm relação com o valor monetário no orçamento.





**Figura 4.3 - Composição vertente UI (aplicada)**

Fonte: Programa próprio (Eurico Clemente)

Neste capítulo estarão expostos excertos na componente visível (utilizador).

### 4.1.1 – *Management*

Este campo *Management*, contém, juntamente, os elementos *client*, *units* e *currencies*, conforme referido anteriormente, sendo que neste capítulo é dado destaque à partição *Client*, pessoa singular ou coletiva, destinatária da proposta, em linguagem portuguesa Cliente.

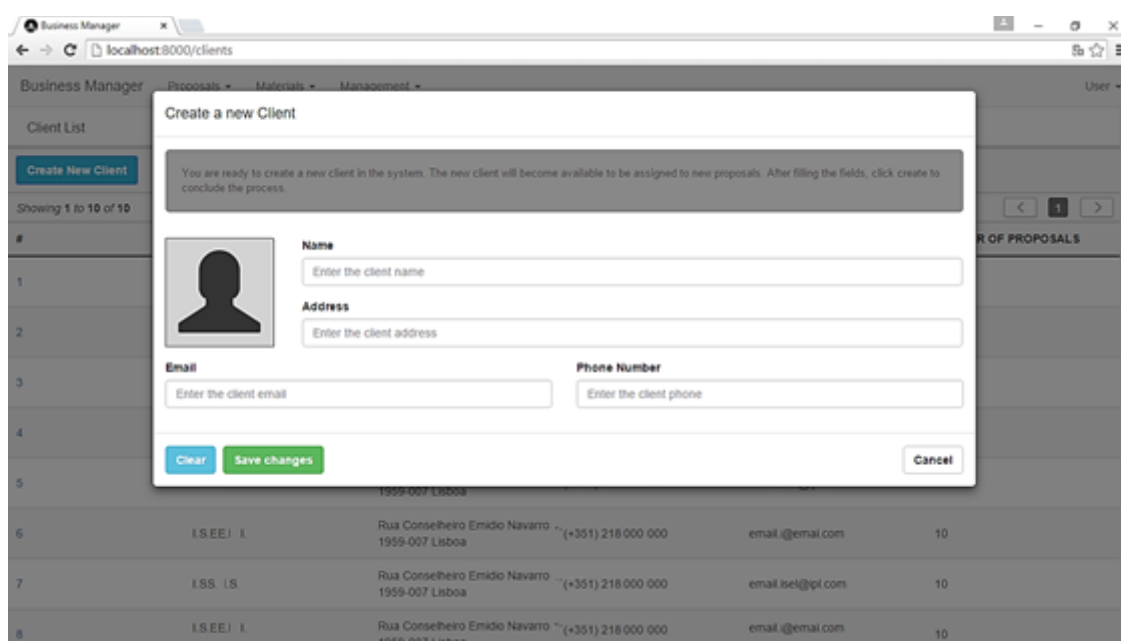
Outra partição do campo *Management* é o elemento *units*, que representa as unidades de medição padrão, utilizadas nos países abrangidos pela proposta. Existem várias unidades de referência (medição), dependendo da origem do país, como, por exemplo, o sistema métrico, que é um sistema de medição internacional decimalizado, utilizado no sistema Internacional (SI). Contudo existem outras unidades referência como o sistema imperial (milhas ou jardas).

Neste elemento é possível editar estas unidades (alterar, adicionar ou remover sistemas de medição).

A partição *currencies* representa as unidades monetárias possíveis na proposta. Em paralelo com o elemento *units*, também este é passível de alterar, adicionar ou remover novos sistemas monetários, adaptando a proposta para qualquer mercado/pais.

Conforme referido, o campo de destaque é a partição *Client* e importa referir que o campo *client* também está, por sua vez, repartido em três divisórias/sub-campos: criação, detalhes e consulta.

O primeiro sub-campo criação define-se como a divisória da componente *Client*, que se encontra referenciada no programa como *new client*. Nesta é preciso atribuir um nome, email, localização e contacto telefónico do mesmo, conforme exposto na figura 4.4.



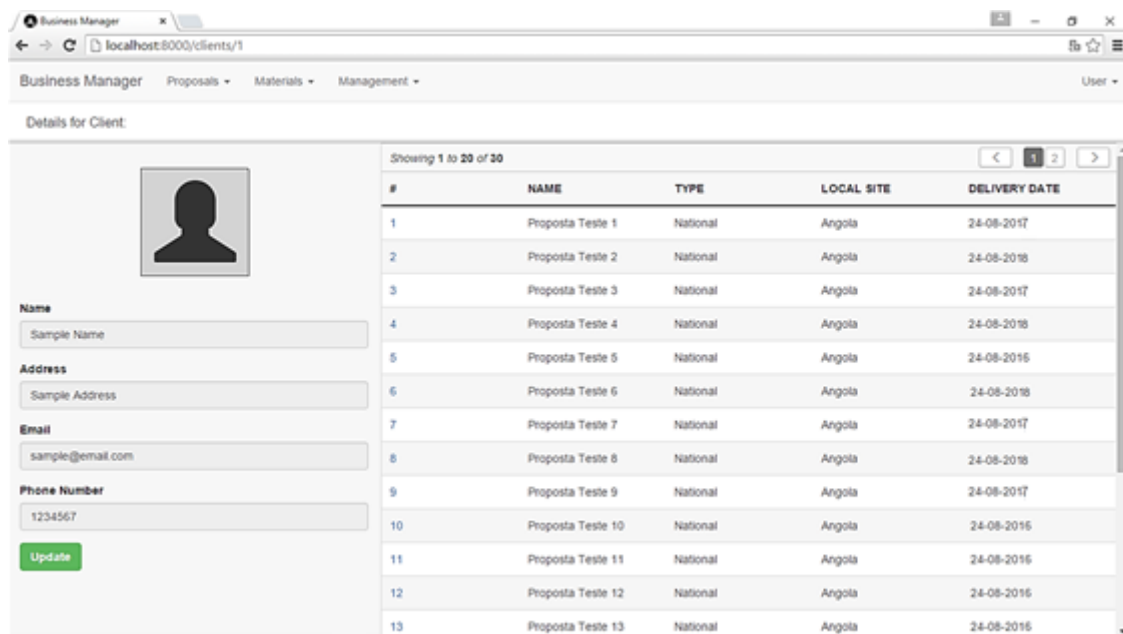
The screenshot shows a web browser window with the URL `localhost:8000/clients`. The application is titled "Business Manager" and has a navigation menu with "Proposals", "Materials", and "Management". The main content area is titled "Client List" and shows a table with columns for ID, Name, Address, Email, and Phone Number. A "Create New Client" button is visible on the left. A modal dialog box titled "Create a new Client" is open in the center. The dialog contains a message: "You are ready to create a new client in the system. The new client will become available to be assigned to new proposals. After filling the fields, click create to conclude the process." Below the message are four input fields: "Name" (with placeholder "Enter the client name"), "Address" (with placeholder "Enter the client address"), "Email" (with placeholder "Enter the client email"), and "Phone Number" (with placeholder "Enter the client phone"). At the bottom of the dialog are three buttons: "Clear" (blue), "Save changes" (green), and "Cancel" (white).

**Figura 4.4 - Janela *New Client***

Fonte: Programa próprio (Eurico Clemente)

O segundo componente é referente aos detalhes (*Client Details*), onde são expressos os elementos mais importantes referentes ao cliente (figura 4.5). Aqui, como o próprio nome

indica, é possível verificar os detalhes de cada cliente (bem como alterar informações pré-efetuadas, através do botão alterar)



**Figura 4.5 - Janela *Client Details***

Fonte: Programa próprio (Eurico Clemente)

Por fim, o terceiro sub-campo consiste na Consulta (*Client Results*). Esta divisória, conforme o próprio nome indica, consiste na consulta de clientes, dentro da respetiva base de dados (figura 4.6). Nota para o facto de a Consulta comunicar com os Detalhes, criando, desta forma, uma otimização de recursos, bem como de tempo (fácil utilização).

A atualização da base de dados é efetuada, antes da respetiva consulta, permitindo ao utilizador efetuar a mesma em toda a base.

Esta pode ser efetuada através da busca geral ou através de texto (filtragem), utilizando botão



#	NAME	LOCAL SITE	DELIVERY DATE	REVISION		
1	Proposta Teste 1	Angola	24-08-2018	Rev 0.1		
2	Proposta Teste 2	Angola	24-08-2018	Rev 0.1		
3	Proposta Teste 3	Angola	24-08-2018	Rev 0.1		
4	Proposta Teste 4	Angola	24-08-2018	Rev 0.1		
5	Proposta Teste 5	Angola	24-08-2018	Rev 0.1		
6	Proposta Teste 6	I.S.E. I.S.	National	Angola	24-08-2018	Rev 0.1
7	Proposta Teste 7	I.S.E.LL	National	Angola	24-08-2018	Rev 0.1
8	Proposta Teste 8	I.S.E. I.S.	National	Angola	24-08-2018	Rev 0.1
9	Proposta Teste 9	I.S.E.LL	National	Angola	24-08-2018	Rev 0.1
10	Proposta Teste 10	I.S.E. I.S.	National	Angola	24-08-2018	Rev 0.1
11	Proposta Teste 11	I.S.E.LL	National	Angola	24-08-2018	Rev 0.1
12	Proposta Teste 12	I.S.E. I.S.	National	Angola	24-08-2018	Rev 0.1
13	Proposta Teste 13	I.S.E.LL	National	Angola	24-08-2018	Rev 0.1

**Figura 4.6 - Janela *Client Results***

Fonte: Programa próprio (Eurico Clemente)


## 4.1.2 – *Materials*

Os materiais são o pilar base da orçamentação, pois toda esta se baseia na relação compra e venda (necessidade de vender/fornecer ou mesmo de comprar). Deste modo, os materiais são um dos três elementos a ser referidos no *UI*.

Seguindo a mesma cadeia de pensamento do componente anterior (*Client*), bem como de toda a elaboração do simulador, explicada anteriormente, também este tópico (*Materials*) será repartido em três divisórias (criação, detalhes e consulta).

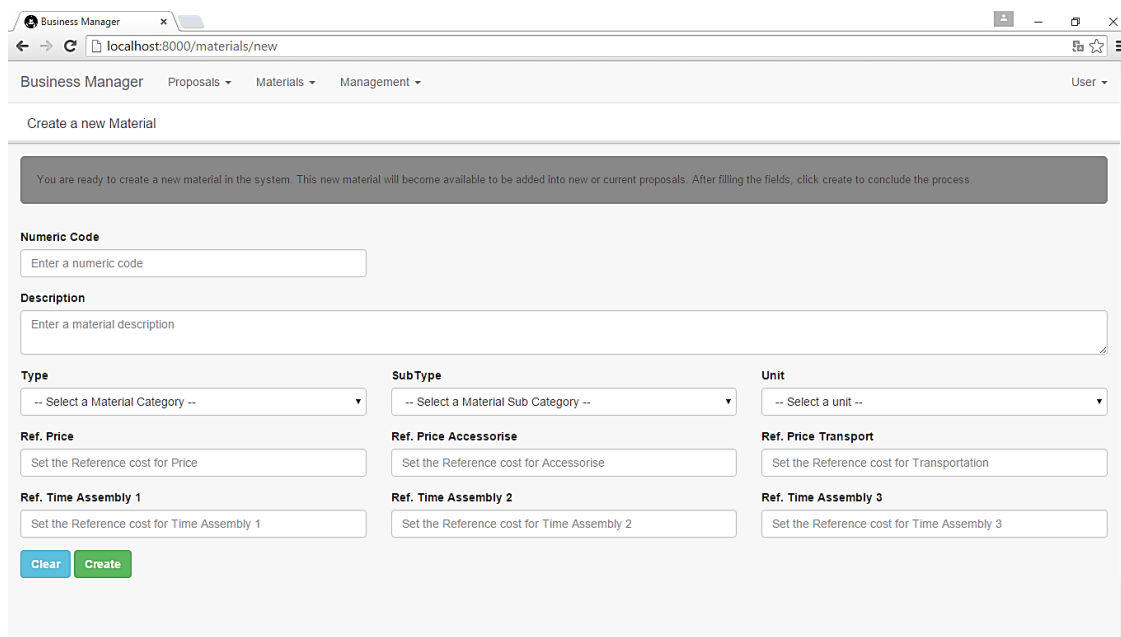
No campo Criação desta divisória, pode-se criar materiais específicos para uma proposta, bem como arquivá-los na base de dados, para posterior utilização em propostas futuras (interligação com a base de dados).

Seguindo o exposto na criação anterior (*New Client*), também aqui (*New Material*) é possível efetuar limpeza ou criar novos elementos, utilizando os botões azul e verde (cores diferentes das escalas, códigos de cores):

O botão “clear”  serve para a eventualidade de um erro na inserção dos valores pretendidos, efetuando a respetiva anulação.

O botão “create”  serve para guardar o novo *Material* na base de dados

\*nota as cores referidas acompanham o *UI*, para a elaboração da janela, de novos elementos, facilitando, deste modo, a orientação espacial do leitor, contudo o texto é adaptado em cada um dos campos.



The screenshot shows a web browser window titled 'Business Manager' with the URL 'localhost:8000/materials/new'. The page header includes 'Business Manager', 'Proposals', 'Materials', and 'Management'. The main heading is 'Create a new Material'. A message box states: 'You are ready to create a new material in the system. This new material will become available to be added into new or current proposals. After filling the fields, click create to conclude the process.' The form contains the following fields:

- Numeric Code:** Enter a numeric code
- Description:** Enter a material description
- Type:** -- Select a Material Category --
- Sub Type:** -- Select a Material Sub Category --
- Unit:** -- Select a unit --
- Ref. Price:** Set the Reference cost for Price
- Ref. Price Accessorise:** Set the Reference cost for Accessorise
- Ref. Price Transport:** Set the Reference cost for Transportation
- Ref. Time Assembly 1:** Set the Reference cost for Time Assembly 1
- Ref. Time Assembly 2:** Set the Reference cost for Time Assembly 2
- Ref. Time Assembly 3:** Set the Reference cost for Time Assembly 3

At the bottom of the form are two buttons: 'Clear' (blue) and 'Create' (green).

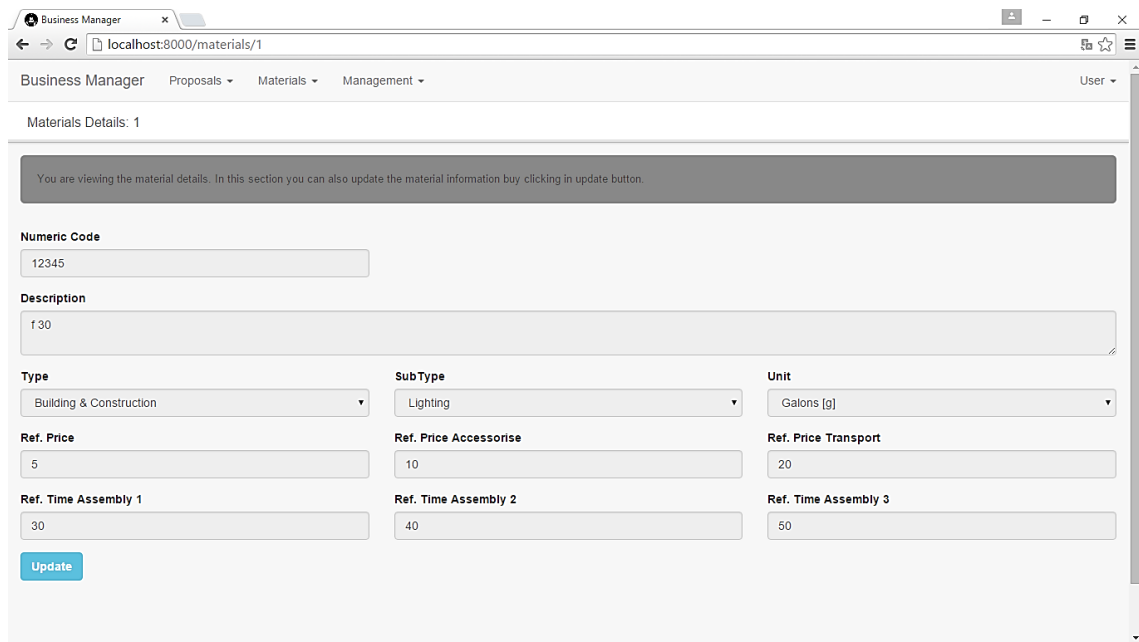
**Figura 4.7 - Janela *New Material***

Fonte: Programa próprio (Eurico Clemente)

Por outro lado, o campo Detalhes ou *Material Details* permite, para além dos restantes (*Clients* e *Proposal*), uma análise entre preços nas várias propostas, como, por exemplo, o preço máximo ou mínimo.

Com isto podemos definir, que nesta divisória estão expressos os elementos mais importantes referentes aos materiais no seu carácter geral (figura 4.8).

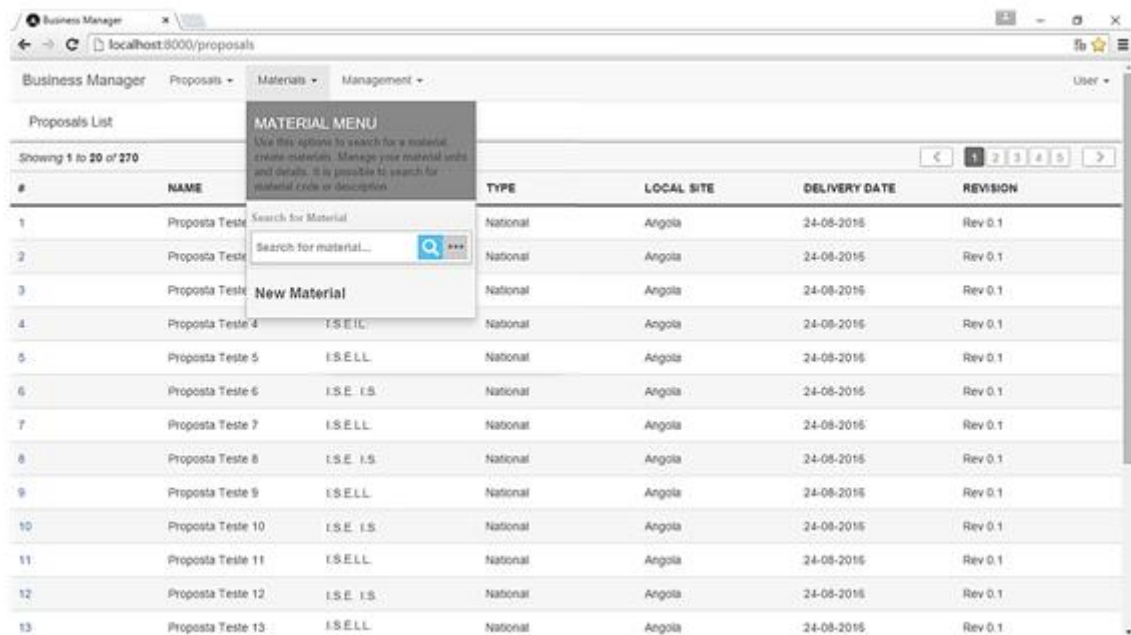
Todas as variáveis encontram-se identificadas, sendo aconselhada a leitura do apêndice 1.



**Figura 4.8 - Janela *Material Details***

Fonte: Programa próprio (Eurico Clemente)

O terceiro campo do *Material* consiste na Consulta (*Material Results*). Esta divisória exibe resultados idênticos aos referidos no *Client Results*, mas referente aos *Materials*, sendo que a consulta permite filtrar materiais pelo nome atribuído, conforme figura 4.9.



**Figura 4.9 - Janela *Material Results***

Fonte: Programa próprio (Eurico Clemente)

### 4.1.3 – Proposal

O último dos elementos referido no *UI* é a *Proposal*. Este campo representa a proposta na sua vertente comercial, pois apresenta o elemento de ligação entre os três campos (deverá ser preenchido no fim, para um mais fácil manuseamento dos dados).

Este campo também se encontra repartido em três divisórias, sendo que na primeira divisória se podem criar propostas (nacionais ou internacionais), bem como introduzir informação adicional.

Conforme exposto no início deste capítulo, este campo deverá ser preenchido no fim, pois permite localizar/endereçar clientes, unidades, matérias, entre outros, da base de dados, preenchendo a sua composição, através de apontadores informáticos e matemáticos (facilitando a interação do utilizador menos experiente na realização de propostas).

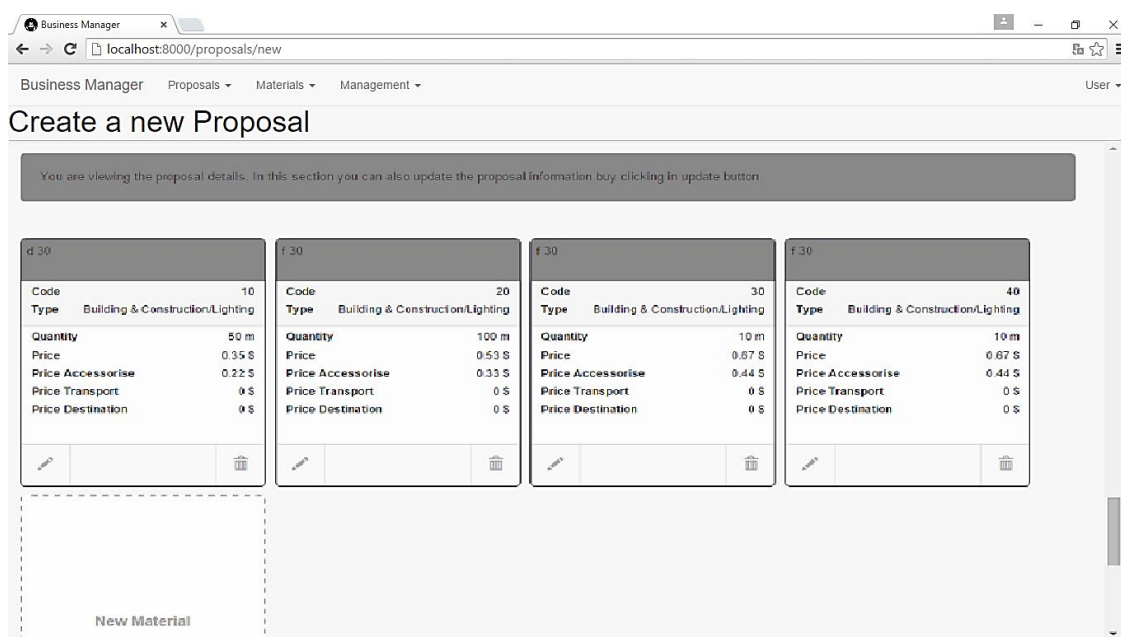


Figura 4.10 - Janela *New Proposal*

Fonte: Programa próprio (Eurico Clemente)

Informa-se o leitor que na segunda divisória da *Proposal ou Details* da *Proposal* é possível fazer uma revisão/inspeção de toda a orçamentação, referente à proposta identificada, permitindo a leitura de duas formas, uma “vista” geral (através da verificação visual dos valores referentes aos cálculos e *outputs* da base 6) ou uma “vista” mais minuciosa (através da verificação visual dos valores referentes aos cálculos e *outputs* compreendidos entre as bases 1 e 5).

O controlo de acesso é efetuado pelo administrador do programa, permitindo que várias pessoas trabalhem em conjunto.

As duas formas de visualização (geral e minuciosa) podem ser restringidas ao utilizador, através do endereço *Login* (entrada no programa) na conta administrador, permitindo desta forma um controlo da área comercial da empresa.

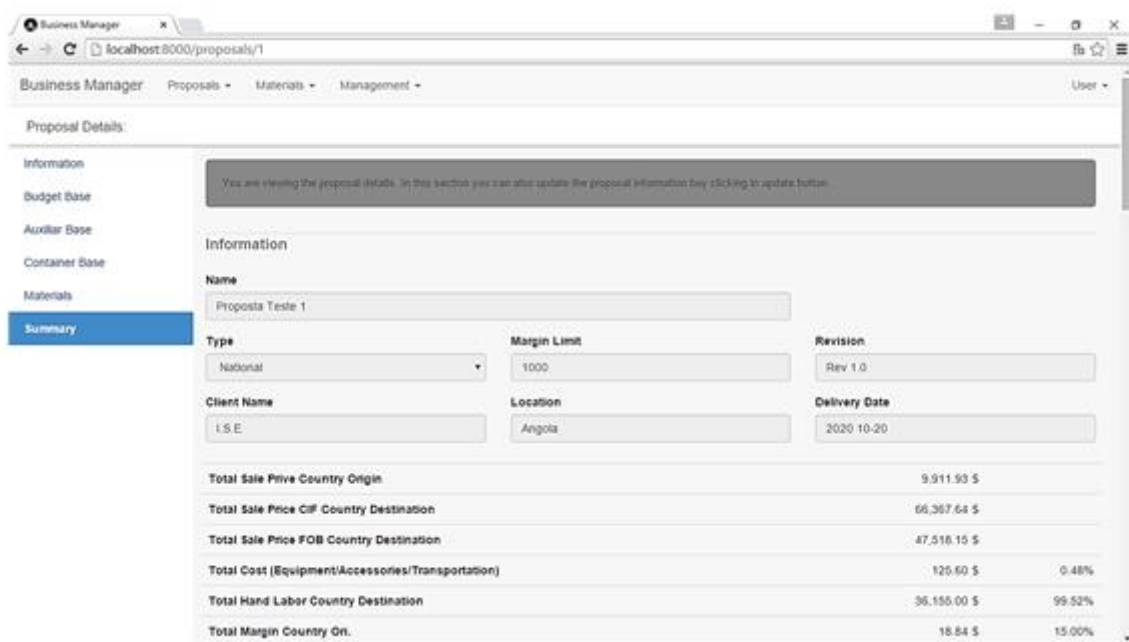


Figura 4.11 - Janela *Proposal Details*

Fonte: Programa próprio (Eurico Clemente)



Por fim, a consulta, conforme exposto anteriormente (representa a capacidade para pesquisar um determinado elemento na base de dados), permite identificar uma proposta específica no leque de propostas realizadas.

#			TYPE	LOCAL SITE	DELIVERY DATE	REVISION
1			National	Angola	24-08-2015	Rev 0.1
2			National	Angola	24-08-2015	Rev 0.1
3			National	Angola	24-08-2015	Rev 0.1
4	Proposta Teste 4	I.S.E.L.L.	National	Angola	24-08-2015	Rev 0.1
5	Proposta Teste 5	I.S.E.L.L.	National	Angola	24-08-2015	Rev 0.1
6	Proposta Teste 6	I.S.E.L.L.	National	Angola	24-08-2015	Rev 0.1
7	Proposta Teste 7	I.S.E.L.L.	National	Angola	24-08-2015	Rev 0.1
8	Proposta Teste 8	I.S.E.L.L.	National	Angola	24-08-2015	Rev 0.1
9	Proposta Teste 9	I.S.E.L.L.	National	Angola	24-08-2015	Rev 0.1
10	Proposta Teste 10	I.S.E.L.L.	National	Angola	24-08-2015	Rev 0.1
11	Proposta Teste 11	I.S.E.L.L.	National	Angola	24-08-2015	Rev 0.1
12	Proposta Teste 12	I.S.E.L.L.	National	Angola	24-08-2015	Rev 0.1
13	Proposta Teste 13	I.S.E.L.L.	National	Angola	24-08-2015	Rev 0.1

**Figura 4.12 - Janela *Proposal Results***

Fonte: Programa próprio (Eurico Clemente)

No caso das *Proposal*, o programa executa uma ação idêntica aos referidos no *Clients e Materials*.

Importante salientar que as propostas devem ser criadas pela mesma ordem de execução acima referida (*Client, Material e Proposal*).

## 4.2 – Tutorial

Devido à limitação de extração (número de páginas) apresentado pela organização responsável pela avaliação deste projeto (IBS), são apresentados links para visualização do modo de utilização idealizado/projetado e para cada uma das bases, juntamente com os restantes elementos introduzidos.

Para mais informação visualizar o TM.

Tabela 4.1 - Tabela Tutorial

Tipo utilização	Links	Plataforma/ repositorio
Login no programa:	<a href="https://www.youtube.com/watch?v=sCbUEeOp4QA&amp;feature=youtu.be">https://www.youtube.com/watch?v=sCbUEeOp4QA&amp;feature=youtu.be</a>	You 
Tutorial geral:	<a href="https://www.youtube.com/watch?v=_DvqOT_hnIM&amp;feature=youtu.be">https://www.youtube.com/watch?v=_DvqOT_hnIM&amp;feature=youtu.be</a>	You 
Proposal - bases (figura 4.1):	<a href="https://www.youtube.com/watch?v=MX18iCrGpLE">https://www.youtube.com/watch?v=MX18iCrGpLE</a>	You 
Para outros exemplos de utilização:	<a href="#">Utilizar canal TM (app ou meios de acesso oficial)</a>	
online	Necessária ligação rede para visualização	

Fonte: próprio (Eurico Clemente)

Tutorial efetuado, utilizando o gravador *screen* da Microsoft (captação vídeo), por sua vez a leitura do texto exposto no Tutorial foi efetuada através do narrador de texto Google.

Os programas referidos são: Microsoft encoder 4 (gravador de *screen*), disponível no site oficial da Microsoft; Voz MT (conversão de texto em voz Google), disponível na *chrome web store*.

Informamos o leitor para a existência do canal chat (assistência e perguntas).

São disponibilizadas todas as ferramentas para uma correta utilização.

Lembramos o leitor, para a necessidade de acesso/ligação rede, para a visualização dos tutoriais, bem como canal chat

## 4.2.1 – Login ou logout

Inicialmente, deverá ser efetuado o *login* no sistema/programa, através do endereço e *password* fornecidos pelo administrador deste; para tal, deverá ser utilizado o botão localizado fisicamente no canto superior direito (*user*). Após clicar no referido botão é possível efetuar *login* ou *logout*.

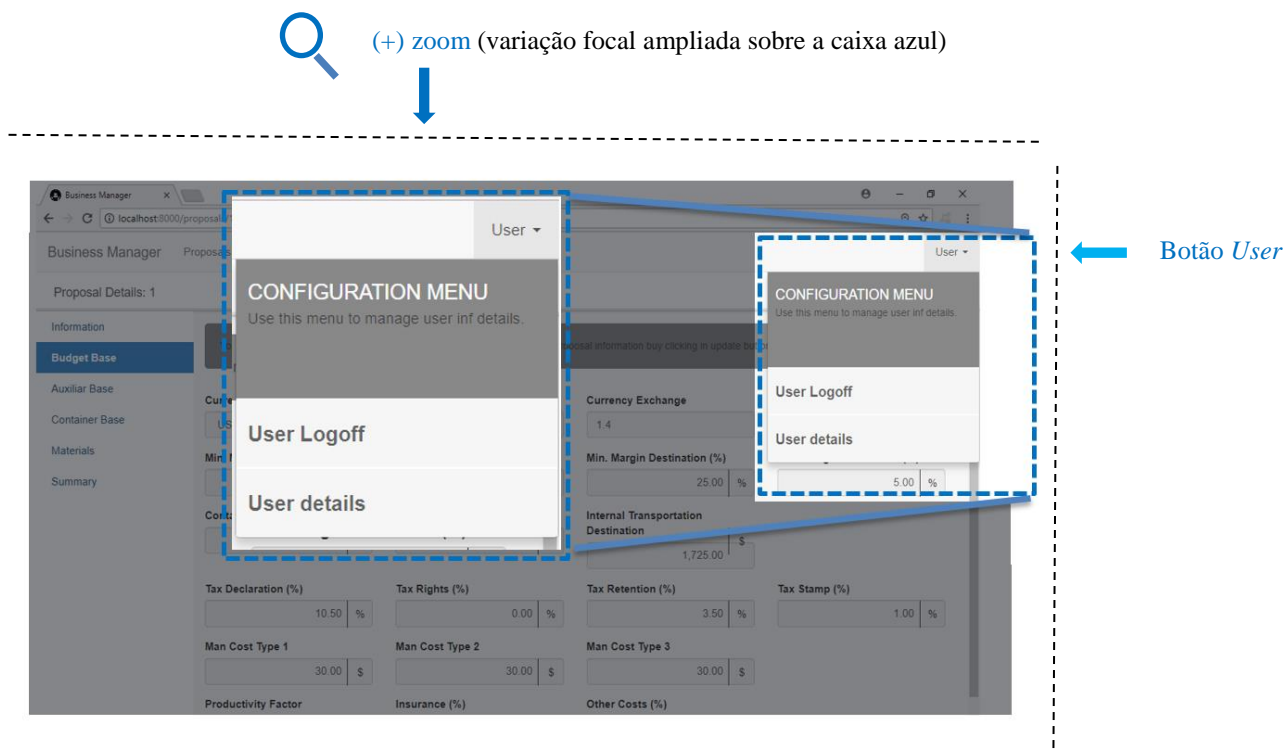


Figura 4.13 - Janela programa destaque canto superior direito (*user*)

Fonte: Programa próprio (Eurico Clemente)

Nota para a importância do controlo das permissões pelo meio do *login*, conforme exposto anteriormente.

## 4.2.1 – Management

O campo *Management* (*Client, units e currency*) é o primeiro campo a contar da direita.

Conforme escrito anteriormente, na proposta é possível endereçar elementos, como por exemplo, as opções *Management* e *Material*, pelo que estes elementos deverão ser editados antecipadamente à realização da mesma.

Os elementos apresentados em *Management* não influenciam o preço da proposta, sendo utilizado este mesmo campo (botões identificados na figura 4.14),

Neste botão (*Management*) é possível alterar, remover ou inserir elementos nas três opções, conforme sequência a baixo exposta.

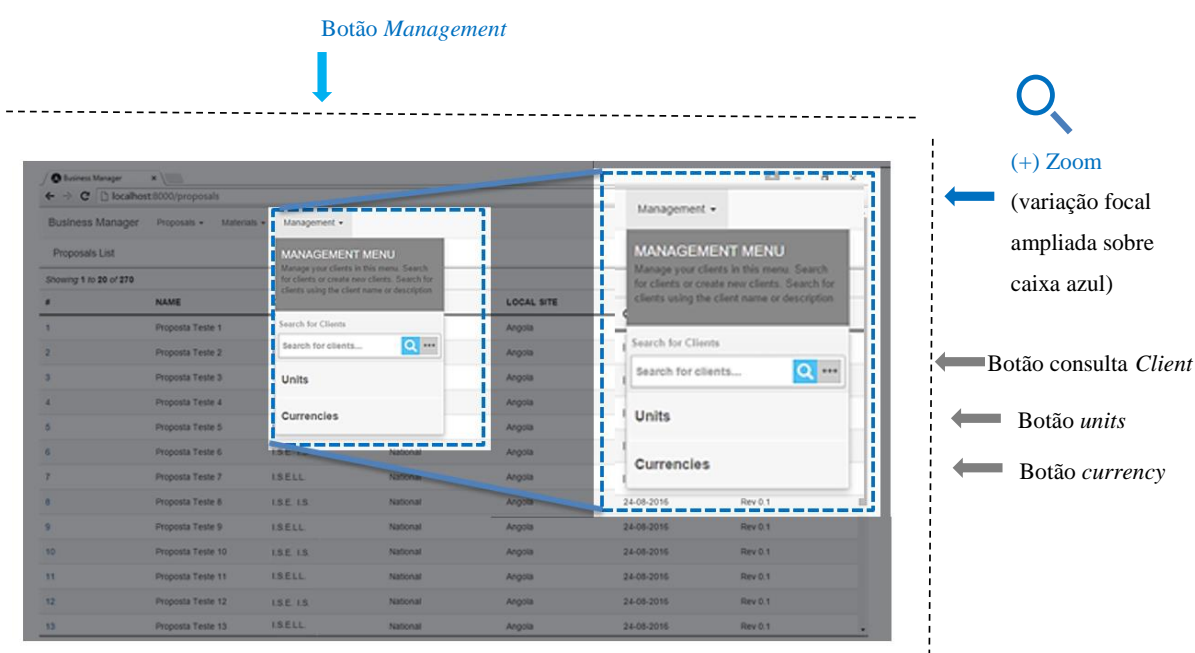


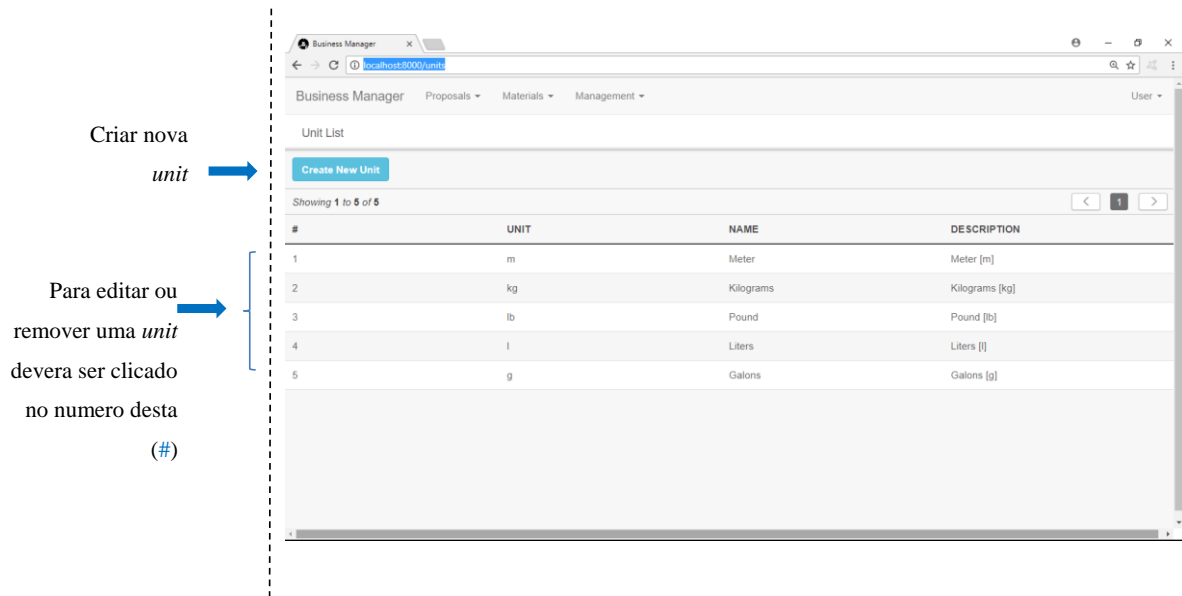
Figura 4.14 - Janela programa destaque botão *Management*

Fonte: Programa próprio (Eurico Clemente)

Conforme referido anteriormente, a categoria *Management* está dividida em três subtipos, sendo que o subtipo *client* encontra-se referido no capítulo 4.1.1 (referido previamente) e ficando em falta apenas os subtipos *units* e *currency*, que serão abordados neste capítulo.

No que toca a estes dois elementos, foi efetuada uma constituição, segundo os padrões mencionados, estando ambos os elementos em concordância com os restantes.

A seguir a clicar no botão *units* aparece o menu identificado na figura 4.15, que permite efetuar todos os mesmos procedimentos, anteriormente referidos nos vários elementos abordados.



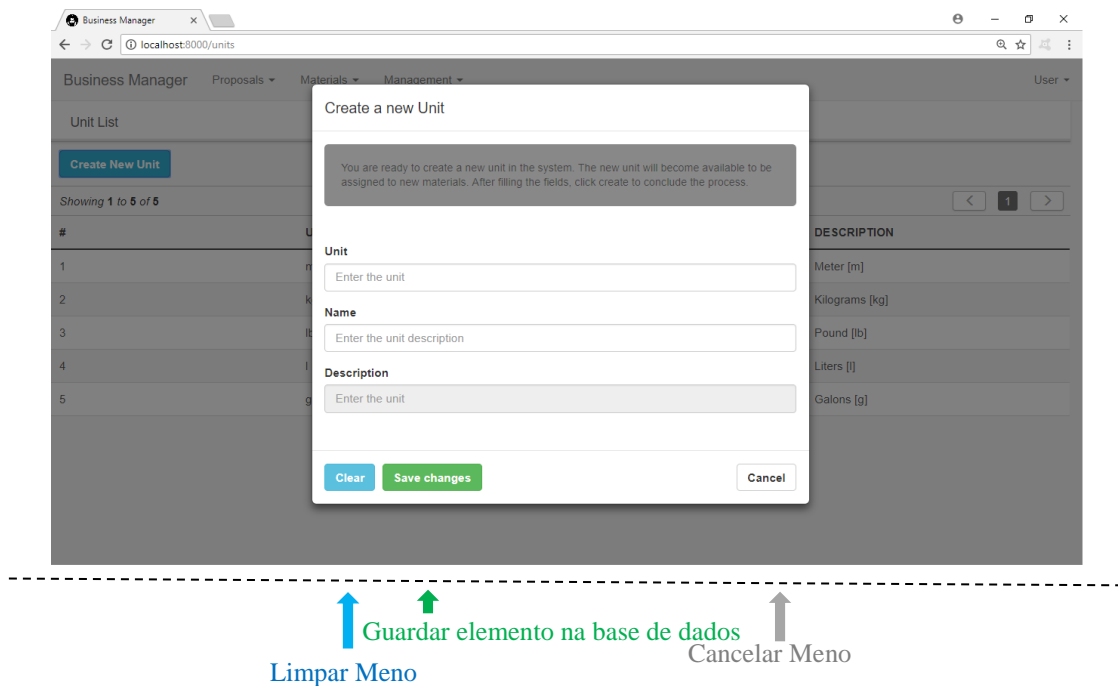
**Figura 4.15 - Janela programa destaque botão *Management (units)***

Fonte: Programa próprio (Eurico Clemente)

Neste campo conforme é possível visualizar na imagem acima exposta, podemos efetuar a criação através do botão (azul, cor diferente das referenciadas na escala/código de cores), colocado no canto esquerdo da janela.

Por outro lado, para editar ou remover, deverá ser primeiro identificado, qual o elemento (*unit*), através da seleção do mesmo, clicando no seu numero (#), ordenado por ordem de criação (do mais antigo para o mais recente).

Seguidamente, mostram-se como criar uma unidade de medição nova ao clicar no botão *create new units* (botão azul), que permite criar unidades de referência para os elementos endereçáveis.



**Figura 4.16 - Janela programa destaque botão *create new units***

Fonte: Programa próprio (Eurico Clemente)

A organização e modo de funcionamento é equivalente ao apresentado nos restantes campos, como por exemplo *New Material*.

Em parceria com este *menu (units)*, também o *menu currency* tem o mesmo modelo base na sua constituição, logo todos os procedimentos são realizados da mesma forma em ambos os *menus*.

### 4.3 – Caso de estudo

Para testar o programa, é necessário analisar cada uma das componentes constituintes do mesmo, pelo que foram efetuadas verificações individuais das componentes informáticas (*front-end e back-end*).

A componente pilar (matemática) foi verificada exaustivamente, não esquecendo todos os elementos envolvidos.

Este teste das componentes matemáticas (variáveis e fórmulas) permitiu a verificação, mais ampla do caso de estudo.

O presente caso de estudo agrega também uma verificação de várias propostas (verificação global do funcionamento completo do sistema), de forma a comprovar o bom funcionamento deste programa, conforme apresentado no Apêndice 4.

---

## CAPÍTULO

# 5

---

### **Conclusões e Melhoramentos**

*Apresenta-se neste capítulo um resumo das ilações mais significativas resultantes da elaboração desta dissertação.*

*Neste capítulo estão, ainda, indicados possíveis desenvolvimentos futuros do trabalho efetuado.*



## 5.1 - Principais conclusões

Em parceria com o inicialmente afirmado no capítulo 1, o grande número de orçamentos “solicitados”, por parte dos requerentes (clientes), não permite fisicamente às PMEs reponderem da mesma forma (valores e proporções) do que as parceiras de maior dimensão. Desta forma, a competição entre os dois tipos de empresas cria uma assimetria fissional dos possíveis negócios (como é verificado em varias estatísticas disponíveis, como, por exemplo, na acinGov, plataforma online para compras públicas).

Mesmo nas empresas de maior dimensão, algumas vezes os orçamentos/propostas comerciais são realizados de forma ineficiente, devido à ocorrência de erros, omissões ou outros elementos que causem alteração do valor proposto.

No decorrer desta dissertação foram abordadas as principais razões e as causas para tal comportamento. Para evitar estas situações e para melhorar todo este processo, foi elaborado um simulador, que visa colmatar todos os problemas identificados (programa disponível gratuitamente).

Este simulador pretende substituir e padronizar a forma como as empresas constroem as suas propostas comerciais, conforme referido no capítulo 1.2, cumprindo sempre a legislação e normas aplicáveis.

Em algumas empresas, as atuais formas para execução dos orçamentos baseiam-se na utilização de ferramentas rudimentares como *Word*, *Excel* ou mesmo em programas proprietários, como, por exemplo, o caso do *Business services proposal*, usando o template *Word* do *Office 365*. Estas plataformas de cálculo não são exatas e incluem um vasto leque de componentes, que prolongam o tempo necessário para a execução das mesmas.

A dissertação foi construída, tendo em mente uma ferramenta de destaque, com aplicação prática no contexto referido (empresas).

Com o objetivo de demonstrar as fórmulas de cálculo, uma vez que estas foram realizadas/concebidas de forma a poderem responder aos vários tipos de utilizadores e às suas exigências, uma razoável parte dos elementos constituintes (informáticos e matemáticos)

encontram-se em anexo. É de salientar ainda o facto destas fórmulas matemáticas serem a base/pilar para a realização deste documento.

Nesta dissertação foram utilizadas linguagens de programação, como, por exemplo, a linguagem *Java*.

Foi também realizado um cálculo exaustivo e minucioso de forma a construir uma plataforma (programa) de cálculo estável e notoriamente *User Friendly*.

Realizou-se, ainda, uma pesquisa intensiva em vários campos, como leis (por exemplo a proteção de dados), orçamentação, importação/exportação de bens e noutras normas em vigor.

## **5.2 - Possíveis Melhoramentos**

Sendo uma ferramenta informática, este tipo de programa necessita de uma constante atualização/manutenção (efetuada por o administrador ou gestor de cliente, em conformidade com as restantes aplicações desta configuração), pois estas evoluem em paralelo com as necessidades dos utilizadores.

Estas atualizações/manutenções podem ser maximizadas com um melhor aproveitamento de espaço (memória informática) e acompanhamento temporal no decorrer da utilização deste (programa/simulador e componente matemática).

Desenvolvimentos futuros e importantes são, ainda, a melhoria de alguns componentes do programa desenvolvidos, como a componente linguística, bem como a adição do modo *offline*.

Existe, também, a possibilidade de, no futuro, se proceder à aplicação deste programa a outras áreas ou campos de estudo, como a contabilidade interna das empresas ou acompanhamento dos recursos geridos (por exemplo à frota automóvel).

Outra das hipóteses será a inclusão neste programa nos sistemas formais de avaliação de desempenho e sistemas de recompensa, bem como noutros tipos de contributos possíveis para melhorar a gestão empresarial.



---

## **REFERÊNCIAS BIBLIOGRÁFICAS**

---

*No capítulo agora descrito são apresentadas as várias referências bibliográficas, consultadas no decorrer deste projeto.*

## Referências bibliográficas

- Abreu, L. 2017. *HTML atualizada e aumentada* (4th). Lisboa: FCA- Editora informática lda.
- Akamatsu, K. 1962. A historical pattern of economic growth in developing countries. *Journal of the Economic Geographical Society of Korea*, 1: 3-25.
- Alyrio, R. 2011. *Negociação e processo decisório*. Fluminense: Fundação Cecierj.
- Banco de Portugal. 2016. Nota de informação estatística. *Análise do setor da construção*, 1-7. Lisboa: Banco de Portugal
- Banco de Portugal. 2017. Uma análise de restrições de financiamento às pme's portuguesas. *Artigos de Estabilidade Financeira*, 4-19. Lisboa: Banco de Portugal
- Brindley, C. 2004. *Supply chain risk*. Manchester: Manchester Metropolitan University.
- Caird, S. 2002. How important is the innovator for the commercial success of innovative products in SMEs?. *Technovation*, 14: 71-83
- Chenevier, G., Clozel, L., & Amer J. 2013. The theory MAT. *The Journal of the American Mathematical Society*, 22: 427-461.
- Christof, E. 2014. Software management. *IEEE Journals*, 31: 21-24.
- Coughlan, P., & Coughlan, D. 2002. Action research for operations management. *International Journal of Operations & Production Management*, 22: 220-240.
- Cristia, J., Ibarrarán, P., Cueto, S., Santiago, A., & Severín, E. 2017. Technology and child development: Evidence from the one laptop per child program. *American Economic Journal: Applied Economics*, 9: 295-320.
- Damas, L. 2017. *SQL ti tecnologia da informação* (14th). Lisboa: FCA- Editora informática lda.
- Dionísio, P., Pereira, H. G., & Cardoso M; Comportamentos de Procura de Informação e Compra Online, Confederação do Comércio e Serviços de Portugal, <http://www.ccp.pt/CCP/pt-PT/63/1044/Psq.aspx>, documento publicado 2012.
- Ebert, C. 2013. *Global software engineering: Lessons from industry*. Paper presented at the IEEE 8th International Conference on Global Software Engineering Workshops, Bari.
- Eckel, B. 2006. *Thinking in java* (4th). New Jersey: Prentice Hall.
- Gonçalo, J. 2015. *Excel avançado - Módulo 2. Apontamentos da unidade curricular de 2º Semestre*. Lisboa: Instituto Superior de Economia e Gestão.

Gonçalves, R. 2010. *Colaboração universidade-empresas*. Instituto Superior de Ciências do Trabalho e da Empresa - Departamento de Ciências e Tecnologias da Informação.

Giffin, D., Levy, A., Stefan, D., Terei, D., Mazières, D., Mitchell, J., & Russo, A. 2017. Protecting data privacy in untrusted web applications. *Journal of Computer Security*, 25: 427-461.

Hedley, G. 2017. *Don't Overlook the Overworked*. The Business Management Magazine for Contractors. Birmingham: Cahaba Media Group publication

Henriques, A. 2016. *Empreendedorismo tecnológico: modelo de negócios de package point*. published master dissertation, ISCAL, Lisboa

Lins, V. K., Servaes, H., & Tamayo, A. 2017. Social capital, trust, and firm performance: The value of corporate social responsibility during the financial crisis. *The Journal of the American Finance Association*, 72: 1785-1824.

Maria, T., & Francis, T;. Model-based software management, Wiley Telecom eBooks, documento publicado 2017.

Northcut, K. M et al. 2009. **Proposal writing from three perspectives: technical communication, engineering and science**, IEEE International Professional Communication Conference.

Project management institute (Vários). 2012. *A guide to the project management body of knowledge: PMBOK guide* (5º ed). Pennsylvania: Project Management Institute.

Puccini, L, A. 2009. *Matemática financeira objetiva e aplicada* (8th). São Paulo: Saraiva.

Smidts, A., Pruyn A. T., & Van Riel, B. M. 2001. The impact of employee communication and perceived external prestige on organizational identification. *Academy of Management Journal*. 44: 1051-1062.

Uchoa, E., Pecin, D., Contardo, C., & Desaulniers, G. 2016. New enhancements for the exact solution of the vehicle routing problem with time windows. *Inform Journal on Computing*. 29: 377-580.

Ma, Z; Liu, Z. BIM – base inteligente acquisition of construction information for cost



---

**APÊNDICE**

**1**

---

**Identificação das variáveis**





---

## APÊNDICE

# 1.1

---

### **Input**

*As variáveis referidas na tabela *Input*, representam os elementos de informação (entrada) para a execução dos cálculos, podendo algumas ter o valor nulo caso não aplicadas.*

## Apêndice nº1 - Identificação das variáveis

### 1.1 - Input

#### Input

Variáveis patenteadas com a letra “i” (*input* programa) + numeração respetiva

<b>Variável</b>	<b>Interpretação</b>	<b>Denominação sumaria</b> <i>(program) *</i>
<b>i1</b>	Definição do nome da obra, conforme expresso na memória descritiva e justificativa (MDJ)	Name
<b>i2</b>	Identificação da localização da obra, conforme o concelho ou província destinatária	Location
<b>i3</b>	Descrição do limite na qual a margem A se altera para a margem B (Nota: margem A e B serão referidas posteriormente nos elementos compreendidos entre i10 e i14)	Margin Limit
<b>i4</b>	Identificação da referência interna da empresa concorrente à obra	Rev.
<b>i5</b>	Identificação do cliente destinatário da proposta	Cliente Name
<b>i6</b>	Definição da data para a qual a proposta é entregue ao destinatário (solicitante)	Delivery Date
<b>i7</b>	Indicação do nome da pessoa responsável pela receção da proposta (lado do cliente)	Cliente Responsible
<b>i8</b>	Indicação do número de revisão da proposta	revisionVersion
<b>i9</b>	Identificação do câmbio/relação entre as moedas referência de dois tipos distintos, particularmente, neste exemplo, será utilizada a relação do câmbio entre Angola e Portugal \$/€	Currency Exchange

## Apêndice nº1 - Identificação das variáveis

### 1.1 - Input

<b>i10</b>	Identificação da margem de venda em Portugal (margem definida para elementos com valor inferior ao referido em i3 )	Min. Margin Origin
<b>i11</b>	Identificação da margem de venda em Portugal (margem definida para elementos com valor superior ao referido em i3)	Max. Margin Origin
<b>i12</b>	Representação da margem de venda no país de destino (margem definida para elementos com valor inferior ao referido em i3)	Min. Margin Destiny
<b>i13</b>	Identificação da margem de venda no país de destino (margem definida para elementos com valor superior ao referido em i3)	Max. Margin Destiny
<b>i14</b>	Definição por meio de percentagem da produtividade local	Productivity Factor
<b>i15</b>	Definição do valor do seguro (em percentagem)	Insurance
<b>i16</b>	Representação do valor de custo de um contentor de 40´ (40 pés)	Container Cost 40
<b>i17</b>	Representação do valor de custo de um contentor de 20´ (20 pés)	Container Cost 20
<b>i18</b>	Identificação do custo com transporte de cada contentor no país de destino	Container Transport Destiny
<b>i19</b>	Identificação do custo com transporte dos contentores em Portugal	Container Transport Origin
<b>i20</b>	Definição dos custos relacionados com o desalfandegamento (em percentagem)	Tax declaration
<b>i21</b>	Identificação dos custos relacionados com os direitos (em percentagem)	Tax right
<b>i22</b>	Identificação dos custos relacionados com a retenção IRT (em percentagem)	Tax retention
<b>i23</b>	Identificação dos custos relacionados com o imposto selo (em percentagem)	Tax stamp

## Apêndice nº1 - Identificação das variáveis

### 1.1 - Input

<b>i24</b>	Definição de um dos três valores possíveis para o custo de mão-de-obra à hora	Man cost type 1
<b>i25</b>	Definição de um dos três valores possíveis para o custo de mão-de-obra à hora	Man cost type 2
<b>i26</b>	Definição de um dos três valores possíveis para o custo de mão-de-obra à hora	Man cost type 3
<b>i27</b>	Definição do número de meses atribuído (pelo cliente) para a execução da obra	Months
<b>i28</b>	Definição do número de dias úteis/trabalháveis por mês	Working Days per month
<b>i29</b>	Representação do número de um contentor de 20" utilizados	Containers 20` Quantity
<b>i30</b>	Representação do número de um contentor de 40" utilizados	Containers 40` Quantity
<b>i31</b>	Representação o tipo do material	Type
<b>i32</b>	Representação o subtipo do material	Sub type
<b>i33</b>	Identificação do código numérico endereço (ID) a um elemento específico	Numeric Code
<b>i34</b>	Descrição ortográfica de um elemento específico	Materials
<b>i35</b>	Identificação do tipo de unidade, em que os valores estão expressos (tabela 3.11)	Unit Type
<b>i36</b>	Identificação da quantidade de elementos referidos	Quantity
<b>i37</b>	Definição do valor unitário do equipamento	Price
<b>i38</b>	Definição do valor unitário dos acessórios (referentes ao equipamento identificado na variável anterior)	Price Accessorise
<b>i39</b>	Definição do valor referente ao transporte do equipamento	Price Transport
<b>i40</b>	Identificação do valor monetário referente aos equipamento locais	Price Local Destination

## Apêndice nº1 - Identificação das variáveis

### 1.1 - Input

<b>i41</b>	Identificação do tempo de montagem do equipamento em minutos (tendo como referência o custo de Hora/Homem -1)	Man time Assembly1
<b>i42</b>	Identificação do tempo de montagem do equipamento em minutos (tendo como referência o custo de Hora/Homem -2)	Man time Assembly2
<b>i43</b>	Identificação do tempo de montagem do equipamento em minutos (tendo como referência o custo de Hora/Homem -3)	Man Time Assembly3
<b>i44</b>	Identificação do valor gasto relacionado com outros custos, como elementos exteriores ao fornecimento/realização dos trabalhos (em percentagem) (somente nas propostas com países de destino e de origem diferentes)	Other Costs
<b>i45</b>	Definição das condições de venda	Sales terms

\*Denominação program, podendo ser identificado por o nome *Front-end* ou *back-end*.



---

## APÊNDICE

# 1.2

---

### **Cálculos e Output**

*Apenas estão identificadas as variáveis com valor relevante para as fórmulas matemáticas, sendo que foram utilizadas mais variáveis do que as expostas devido endereçamento (linguagem programação).*



## Apêndice nº1 - Identificação das variáveis

### 1.2 - Cálculos e Output

#### Cálculos e Output

Variáveis patenteadas com a letra “o” (*output* programa) + numeração respetiva

<b>Variável</b>	<b>Interpretação</b>	<b>Denominação sumaria (<i>program</i>) *</b>
<b>o1</b>	Número total de dias atribuído (pelo cliente) para a execução da obra	Number Total Days
<b>o2</b>	Número de horas total relacionada com o custo hora/homem -1	Number Hours Total Cost Man1
<b>o3</b>	Número de dias totais relacionada com o custo hora/homem -1	Number Total Days Type1
<b>o4</b>	Número de homens necessário, para realizar os trabalhos relacionados com o custo hora/homem -1	Number Of Men Type1
<b>o5</b>	Número de equipas compostas por um oficial mais um ajudante (custo hora/homem -1);	Number Of Teams Type1
<b>o6</b>	Número de horas total relacionada com o custo hora/homem -2	Number Hours Total Cost Man2
<b>o7</b>	Número de dias total relacionada com o custo hora/homem -2	Number Total Days Type2
<b>o8</b>	Número de homens necessários, para realizar os trabalhos relacionados com o custo hora/homem -2	Number Of Men Type2
<b>o9</b>	Número de equipas compostas por um oficial mais um ajudante (custo hora/homem -2)	Number Of Teams Type2
<b>o10</b>	Número de horas total relacionada com o custo hora/homem -3	Number Hours Total Cost Man3
<b>o11</b>	Número de dias total relacionada com o custo hora/homem -3	Number Total Days Type3

## Apêndice nº1 - Identificação das variáveis

### 1.2 - Cálculos e Output

<b>o12</b>	Número de homens necessários, para realizar os trabalhos relacionados com o custo hora/homem -3	Number Of Men Type3
<b>o13</b>	Número de equipas compostas por um oficial mais um ajudante (custo hora/homem -3)	Number Of Teams Type3
<b>o14</b>	Número total de horas gasto na execução da obra	Total Number Hours In Ex
<b>o15</b>	Número total de dias gasto na realização da empreitada	Total Number Days In Ex
<b>o16</b>	Número total de homens necessários, para realizar os trabalhos	Total Number Men In Ex
<b>o17</b>	Número de equipas totais compostas por um oficial mais um ajudante	Total Number Teams Composed
<b>o18</b>	Somatório dos custos finais dos equipamentos no país de destino	Total Cost Free On Boards
<b>o19</b>	Custo derivado ao seguro dos contentores	Cost Insurance Containers
<b>o20</b>	custo derivado ao frete marítimo	Cost Derived Sea Freight
<b>o21</b>	Custo relacionado com o transporte interno no país de destino	Cost Internal Transport Destination
<b>o22</b>	Somatório dos elementos referenciados como bases contentores	Sum Elements Referenced Containers
<b>o23</b>	Valor do custo relacionado com o frete marítimo (em percentagem)	Cost Maritime Freight Percentage
<b>o24</b>	Valor referente ao custo relacionado com o transporte interno no país de destino (em percentagem)	Cost Internal Transport Destination Percentage
<b>o25</b>	Valor auxiliar 1, necessário para o bom funcionamento do programa (guarda temporariamente valores de outras variáveis)	Aux25
<b>o26</b>	Valor auxiliar 2, necessário para o bom funcionamento do programa (guarda temporariamente valores de outras variáveis)	Aux26

## Apêndice nº1 - Identificação das variáveis

### 1.2 - Cálculos e Output

<b>o27</b>	Preço da montagem, referente a um equipamento específico (unitário)	Mount Price Equipment
<b>o28</b>	Preço de custo, referente a um equipamento específico (unitário)	Cost Prices Equipment
<b>o29</b>	Margem no equipamento referente ao país de envio (unitária)	Margin Equipment Source
<b>o30</b>	Preço de venda, referente a um equipamento específico no país de envio (unitário)	Sale Price Equipment Source
<b>o31</b>	Custo do material físico, no país de destino (unitário)	Cost Material Destination Country
<b>o32</b>	Custo relacionado com a mão-de-obra, no país de destino (unitário)	Cost Hand Labor Destination Country
<b>o33</b>	Custo referente ao equipamento, no país de destino (unitário)	Cost Price Equipment Destination Country
<b>o34</b>	Valor de custo destinado ao desalfandegamento (unitário)	Value For Customs Clearance
<b>o35</b>	Valor destinado aos direitos (unitário)	Value For Rights
<b>o36</b>	Valor destinado a retenção IRT (unitário)	Value For IRT Retention
<b>o37</b>	Valor destinado ao imposto selo (unitário)	Value For Stamp Dut
<b>o38</b>	Custo físico total, referente a um equipamento específico no país de destino (unitário)	Cost Total Price Destination Country
<b>o39</b>	Custo alusivo a mão-de-obra total, referente a um equipamento específico no país de destino (unitário)	Cost Total Price Hand Labor Destination Country
<b>o40</b>	Custo total, referente a um equipamento específico no país de destino (unitário)	Total Cost Price Destination Country
<b>o41</b>	Margem no equipamento físico referente ao país de destino (unitária)	Margin Equipment Destination Country
<b>o42</b>	Margem relacionada com a mão-de-obra, referente ao país de destino (unitário)	Margin Hand Labor Destination Country

## Apêndice nº1 - Identificação das variáveis

### 1.2 - Cálculos e Output

<b>o43</b>	Margem total no equipamento referente ao país de destino (unitária)	Margin Total Destination Country
<b>o44</b>	Preço de venda dos materiais, no país de destino (unitário)	Total Sale Price Equipment Destination Country
<b>o45</b>	Preço de venda referente a mão-de-obra, no país de destino (unitário)	Total Sale Price Hand Labor Destination Country
<b>o46</b>	Preço de venda total do equipamento, no país de destino (unitário)	Total Sale Destination Country
<b>o47</b>	Preço de venda total dos vários equipamentos, referenciados como iguais e quantificados na mesma célula, no país de destino (parcial)	Item Sale Price
<b>o48</b>	Valor gasto relacionado com outros custos, como elementos exteriores ao fornecimento/realização dos trabalhos (unitário)	Amount Spent Other Costs
<b>o49</b>	Valor gasto relacionado com outros custos, como elementos exteriores ao fornecimento/realização dos trabalhos (parcial)	Item Spent Other Cost
<b>o50</b>	Valor destinado as despesas com transporte (seguro)	Item Spent Transport Insurance
<b>o51</b>	Valor destinado as despesas com transporte (frete)	Item Spent Transport Sea Freight
<b>o52</b>	Valor destinado as despesas com transporte interno, no país de destino	Item Spent Internal Transport Destination Country
<b>o53</b>	Preço de venda dos materiais e do transporte, no país de destino (unitário)	Doublesale Price Materials And Transport Destination Country
<b>o54</b>	Preço referente a mão-de-obra e transporte, no país de destino (unitário)	Hand Labor And Shipping Price Destination Country
<b>o55</b>	Preço de venda total e do transporte, no país de destino (unitário)	Total Sale Price Plus Transport Destination Country Unitary

## Apêndice nº1 - Identificação das variáveis

### 1.2 - Cálculos e Output

<b>o56</b>	Preço de venda total e do transporte, no país de destino (parcial)	Total Sale Price Plus Transport Destination Country
<b>o57</b>	Preço de custo relativo à mão-de-obra, do tipo 1	Cost Price Labor1
<b>o58</b>	Preço de custo relativo à mão-de-obra, do tipo 2	Cost Price Labor2
<b>o59</b>	Preço de custo relativo à mão-de-obra, do tipo 3	Cost Price Labor3
<b>o60</b>	Somatório dos preços dos equipamentos descritos no mapa de quantidade	Sum Prices Equipment
<b>o61</b>	Somatório dos preços das montagens dos equipamentos descritos no mapa de quantidade	Sum Prices Hand Labor
<b>o62</b>	Somatório dos preços dos acessórios, necessários para o bom funcionamento dos equipamentos referidos	Sum Prices Equipment Accessorise
<b>o63</b>	Valor destinado aos transportes	Sum Prices Equipment Transportation
<b>o64</b>	Somatório dos vários preços de custo, referente aos equipamentos	Sum Prices Cost Equipment
<b>o65</b>	Somatório das margens, referente ao país de envio	Sum Prices Margin Equipment Source
<b>o66</b>	Somatório dos preços de venda, referente aos equipamentos no país de envio	Total Selling Prices Source
<b>o67</b>	Somatório alusivo ao custo dos materiais no país de destino	Sum Cost Material Destination Country
<b>o68</b>	Somatório do custo da mão-de-obra no país de destino	Sum Hand Labor Destination Country
<b>o69</b>	Somatório do custo dos equipamentos no país de destino	Sum Cost Price Equipment Destination Country
<b>o70</b>	Somatório do custo relacionado com os equipamentos locais, no país de destino	Sum Prices Equipment Destination

## Apêndice nº1 - Identificação das variáveis

### 1.2 - Cálculos e Output

<b>o71</b>	Somatório dos custos derivados ao desalfandegamento	Sum Value For Customs Clearance
<b>o72</b>	Somatório dos custos derivados aos direitos	Sum Value For Rights
<b>o73</b>	Somatório dos custos derivados à retenção IRT	Sum Value For IRT Retention
<b>o74</b>	Somatório dos custos oriundos do imposto de selo	Sum Value For Stamp Duty
<b>o75</b>	Somatório dos custos dos vários equipamentos referidos, referente ao país de destino	Sum Cost Total Price Destination Country
<b>o76</b>	Somatório dos custos, referentes à mão-de-obra dos vários equipamentos referidos, no país de destino	Sum Cost Total Price Hand Labor Destination Country
<b>o77</b>	somatório dos preços de custo, no país de destino	Sum Total Cost Price Destination Country
<b>o78</b>	Somatório das margens alusivas aos materiais no país de destino	Sum Margin Equipment Destination Country
<b>o79</b>	somatório das margens alusivas à mão-de-obra no país de destino	Sum Margin Hand Labor Destination Country
<b>o80</b>	somatório das margens, no país de destino	Sum Margin Total Destination Country
<b>o81</b>	somatório dos preços de venda dos materiais	Sum Total Sale Price Equipment Destination Country
<b>o82</b>	somatório dos preços de venda referente à mão-de-obra	sumTotalSalePriceHandLabor DestinationCountry
<b>o83</b>	somatório dos preços de venda dos trabalhos a realizar, referente ao país de destino	Sum Total Sale Destination Country
<b>o84</b>	somatório dos preços de venda totais dos vários equipamentos, no país de destino	Total Sales Prices Equipment Destination
<b>o85</b>	somatório dos valores gastos relacionados com outros custos, como por exemplo,	Sum Amount Spent Other Costs

## Apêndice nº1 - Identificação das variáveis

### 1.2 - Cálculos e Output

	elementos exteriores ao fornecimento/realização dos trabalhos (unitário)	
<b>o86</b>	somatório dos valores gastos relacionados com outros custos, como por exemplo, elementos exteriores ao fornecimento/realização dos trabalhos (parcial)	Sum Item Spent Other Costs
<b>o87</b>	somatório dos valores destinado às despesas com transporte (seguro)	Sum Item Spent Transport Insurance
<b>o88</b>	somatório dos valores destinado às despesas com transporte (frete)	Sum Item Spent Transport Sea Freight
<b>o89</b>	somatório dos valores destinado às despesas com transporte interno, no país de destino	Sum Item Spent Internal Transport Destination Country
<b>o90</b>	somatório dos preços de venda dos materiais e do transporte, no país de destino (unitários)	Sum Doublesale Price Materials And Transport Destination Count
<b>o91</b>	somatório dos preços referentes à mão-de-obra e transporte, no país de destino (unitários)	Sum Hand Labor And Shipping Price Destination Country
<b>o92</b>	somatório dos preços de venda totais e do transporte, no país de destino (unitários)	Sum Total Sale Price Plus Transport Destination Country Unit
<b>o93</b>	somatório dos preços de venda total e do transporte, no país de destino (parciais)	Sum Item Spent Other Costs
<b>o94</b>	somatório temporal (em horas) relativo à mão-de-obra, do tipo 1	Total Assembly Time Type1
<b>o95</b>	somatório dos preços de custo relativo à mão-de-obra, do tipo 1	Sum Cost Price Labor1
<b>o96</b>	somatório temporal (em horas) relativo à mão-de-obra, do tipo 2	Total Assembly Time Type2

## Apêndice nº1 - Identificação das variáveis

### 1.2 - Cálculos e Output

<b>o97</b>	somatório dos preços de custo relativo à mão-de-obra, do tipo 2	Sum Cost Price Labor2
<b>o98</b>	somatório temporal (em horas) relativo à mão-de-obra, do tipo 3	Total Assembly TimeType3
<b>o99</b>	somatório dos preços de custo relativo à mão-de-obra, do tipo 3	Sum Cost Price Labor3
<b>o100</b>	Preço de venda total da empresa no país de origem	Total Sale Price Country Origin
<b>o101</b>	Preço de venda total CIF, da empresa no país de destino	Total SalePrice CIF Destination Count
<b>o102</b>	Preço de venda total FOB, da empresa no país de destino	Total SalePrice FOB Destination Country
<b>o103</b>	Custo dos vários equipamentos no país de origem	Total Cost Equipment Country Origin
<b>o104</b>	Custo de montagem (mão-de-obra) dos vários equipamentos no país de destino	Total Hand Labor Destination Country
<b>o105</b>	Margem da empresa no país de origem	Total Margin Country Origin
<b>o106</b>	Margem da empresa no país de destino	Total Margin Destination Country
<b>o107</b>	Valor destinado à retenção IRT	Total Value IRT Retention
<b>o108</b>	Valor destinado a retenção do imposto de selo	Total Value Stamp Duty Retention
<b>o109</b>	Valor destinado ao diretos	Total Value Of The Rights
<b>o110</b>	Preço dos contentores, referente ao país de destino	Total Price Container Destination Country
<b>o111</b>	Valor destinado ao transporte interno (contentores) no país de destino	T Value Internal Transport Container Destination Country
<b>o112</b>	Valor destinado ao desalfandegamento	Total Value For Customs Clearance
<b>o113</b>	Valor destinado ao seguro transposto no país de destino	Total Price Insurance Containers Destination Country



## Apêndice nº1 - Identificação das variáveis

### 1.2 - Cálculos e Output

<b>o114</b>	Custo dos vários equipamentos no país de origem (em percentagem)	Total Cost Equipment Country Origin Percentage
<b>o115</b>	Custo de montagem (mão-de-obra) dos vários equipamentos no país de destino (em percentagem)	Total Cost Hand Labor Country Origin Percentage
<b>o116</b>	Margem da empresa no país de origem (em percentagem)	T Margin Country Origin Percentage
<b>o117</b>	Margem da empresa no país de destino (em percentagem)	Total Margin Destination Country Percentage
<b>o118</b>	Valor total referente ao frete marítimo	Full Amount Of Ocean Freight
<b>o119</b>	Valor final dos trabalhos referidos no mapa de quantidades, no país de destino	Final Value On The Destination Country
<b>o120</b>	Preço parcial dos equipamentos (sem mão-de-obra), projetados no país de origem	Partial Price Equipment Origin Country
<b>o121</b>	Valor do seguro, projetado no país de origem	Insurance Value Origin Country
<b>o122</b>	Valor do frete, projetado no país de origem	Valueofocanfreight Origin Country
<b>o123</b>	Valor total, referente ao pró-forma número 5	Total Amount Pro Forma Number5
<b>o124</b>	Preço unitário da montagem, referente ao país de origem	Unit Price Assembly
<b>o125</b>	Peço parcial montagem, referente ao país de origem	Partial Price Assembly
<b>o126</b>	somatório dos preços parcial da montagem, referente ao país de origem	Sum Partial Price Assembly
<b>o127</b>	Valor total, referente ao pró-forma número 7	Total Amount ProForma Number7

\*Denominação program, podendo ser identificado por o nome *Fornt-end* ou *back-end*.

---

## APÊNDICE

# 2

---

### Fórmulas de cálculo

*Informação sobre formulas, bem com a sua posição e importância nas respectivas bases,*

## Apêndice nº2 - Formulas de calculo

Posição de variáveis/formulas no cálculo ( Bases)

# ÍNDICE

Anexos

Nota: As figuras, tabelas e expressões são numeradas de ordem sequencial. Todas as tabelas, figuras e expressões com referência à letra “a” referenciam-se os anexos.

### Tabelas

#### Input

Tabela a2.1 - Bases para identificação da proposta (variáveis input).....	102
Tabela a2.1 - Bases para identificação da proposta (exemplo) .....	103
Tabela a2.3 - Bases de cálculo para orçamentação (variáveis) .....	106
Tabela a2.4 - Bases de cálculo para orçamentação (exemplo).....	108
Tabela a2.5 - Bases auxiliares (variáveis) .....	110
Tabela a2.6 - Bases auxiliares (exemplo) .....	111
Tabela a2.7- Características técnicas/referências para contentores 20”.....	114
Tabela a2.8 - Características referências para contentores 40” .....	115
Tabela a2.9 - Bases contentores (variáveis) .....	115
Tabela a2.10 - Bases contentores (exemplo) .....	116
Tabela a2.11 - Valores possíveis para as variáveis i31 e i32.....	118
Tabela a2.12 - Extensão da folha de cálculo.....	119
Tabela a2.13 - Divisórias da folha de cálculo.....	119
Tabela a2.14 - Bases para identificação dos elementos (variáveis) .....	120
Tabela a2.15 - Valores possíveis para as variáveis i35.....	124
Tabela a2.16 - Bases para identificação dos elementos (exemplo) .....	125

#### Output

Tabela a2.5 - Bases auxiliares (variáveis) .....	135
Tabela a2.6 - Bases auxiliares (exemplo) .....	141
Tabela a2.9 - Bases contentores (variáveis) .....	144
Tabela a2.10 - Bases contentores (exemplo) .....	147
Tabela a2.14 - Bases para identificação dos elementos (variáveis) .....	150
Tabela a2.16 - Bases para identificação dos elementos (exemplo) .....	181
Tabela a2.17 - Resumo da orçamentação (variáveis).....	188
Tabela a2.18 - Resumo da orçamentação (exemplo).....	195

## Apêndice nº2 - Formulas de calculo

Posição de variáveis/formulas no cálculo ( Bases)

### Figuras

Figura a1 - sistema fechado.....	134
----------------------------------	-----

### Equações

Equação 1 – Cálculo o1.....	135
Equação 2 – Cálculo o2.....	135
Equação 3 – Cálculo o3.....	136
Equação 4 – Cálculo o4.....	136
Equação 5 – Cálculo o5.....	136
Equação 6 – Cálculo o6.....	137
Equação 7 – Cálculo o7.....	137
Equação 8 – Cálculo o8.....	137
Equação 9 – Cálculo o9.....	138
Equação 10 – Cálculo o10.....	138
Equação 11 – Cálculo o11.....	138
Equação 12 – Cálculo o12.....	139
Equação 13 – Cálculo o13.....	139
Equação 14 – Cálculo o14.....	139
Equação 15 – Cálculo o15.....	140
Equação 16 – Cálculo o16.....	140
Equação 17 – Cálculo o17.....	140
Equação 18 – Cálculo o18.....	144
Equação 19 – Cálculo o19.....	145
Equação 20 – Cálculo o20.....	145
Equação 21 – Cálculo o21.....	146
Equação 22 – Cálculo o22.....	146
Equação 23 – Cálculo o23.....	147
Equação 24 – Cálculo o24.....	147
Equação 27 – Cálculo o27.....	155
Equação 28 – Cálculo o28.....	155
Equação 29 – Cálculo o29.....	156
Equação 30 – Cálculo o30.....	156
Equação 31 – Cálculo o31.....	157
Equação 32 – Cálculo o32.....	157

## Apêndice nº2 - Formulas de calculo

### Posição de variáveis/formulas no cálculo ( Bases)

Equação 33 – Cálculo o33.....	157
Equação 34 – Cálculo o34.....	158
Equação 35 – Cálculo o35.....	158
Equação 36 – Cálculo o36.....	158
Equação 37 – Cálculo o37.....	159
Equação 38 – Cálculo o38.....	159
Equação 39 – Cálculo o39.....	160
Equação 40 – Cálculo o40.....	160
Equação 41 – Cálculo o41.....	160
Equação 42 – Cálculo o42.....	161
Equação 43 – Cálculo o43.....	162
Equação 44 – Cálculo o44.....	162
Equação 45 – Cálculo o45.....	163
Equação 46 – Cálculo o46.....	163
Equação 47 – Cálculo o47.....	163
Equação 48 – Cálculo o48.....	164
Equação 49 – Cálculo o49.....	164
Equação 50 – Cálculo o50.....	165
Equação 51 – Cálculo o51.....	165
Equação 52 – Cálculo o52.....	166
Equação 53 – Cálculo o53.....	166
Equação 54 – Cálculo o54.....	167
Equação 55 – Cálculo o55.....	167
Equação 56 – Cálculo o56.....	167
Equação 57 – Cálculo o57.....	168
Equação 58 – Cálculo o58.....	168
Equação 59 – Cálculo o59.....	168
Equação 60 – Cálculo o60.....	169
Equação 61 – Cálculo o61.....	169
Equação 62 – Cálculo o62.....	169
Equação 63 – Cálculo o63.....	170
Equação 64 – Cálculo o64.....	170
Equação 65 – Cálculo o65.....	170
Equação 66 – Cálculo o66.....	171
Equação 67 – Cálculo o67.....	171
Equação 68 – Cálculo o68.....	171

## Apêndice nº2 - Formulas de calculo

### Posição de variáveis/formulas no cálculo ( Bases)

Equação 69 – Cálculo o69.....	172
Equação 70 – Cálculo o70.....	172
Equação 71 – Cálculo o71.....	172
Equação 72 – Cálculo o72.....	173
Equação 73 – Cálculo o73.....	173
Equação 74 – Cálculo o74.....	173
Equação 75 – Cálculo o75.....	174
Equação 76 – Cálculo o76.....	174
Equação 77 – Cálculo o77.....	174
Equação 78 – Cálculo o78.....	175
Equação 79 – Cálculo o79.....	175
Equação 80 – Cálculo o80.....	175
Equação 81 – Cálculo o81.....	176
Equação 82 – Cálculo o82.....	176
Equação 83 – Cálculo o83.....	176
Equação 84 – Cálculo o84.....	177
Equação 85 – Cálculo o85.....	177
Equação 86 – Cálculo o86.....	177
Equação 87 – Cálculo o87.....	178
Equação 88 – Cálculo o88.....	178
Equação 89 – Cálculo o89.....	178
Equação 90 – Cálculo o90.....	179
Equação 91 – Cálculo o91.....	179
Equação 92 – Cálculo o92.....	179
Equação 93 – Cálculo o93.....	180
Equação 94 – Cálculo o94.....	180
Equação 95 – Cálculo o95.....	180
Equação 96 – Cálculo o96.....	181
Equação 97 – Cálculo o97.....	181
Equação 98 – Cálculo o98.....	181
Equação 99 – Cálculo o99.....	182
Equação 100 – Cálculo o100.....	188
Equação 101 – Cálculo o101.....	189
Equação 102 – Cálculo o102.....	189
Equação 103 – Cálculo o103.....	190
Equação 104 – Cálculo o104.....	190

## Apêndice nº2 - Formulas de calculo

### Posição de variáveis/formulas no cálculo ( Bases)

Equação 105 – Cálculo o105.....	190
Equação 106 – Cálculo o106.....	191
Equação 107 – Cálculo o107.....	191
Equação 108 – Cálculo o108.....	191
Equação 109 – Cálculo o109.....	192
Equação 110 – Cálculo o110.....	192
Equação 111 – Cálculo o111.....	192
Equação 112 – Cálculo o112.....	193
Equação 113 – Cálculo o113.....	193
Equação 114 – Cálculo o114.....	193
Equação 115 – Cálculo o115.....	194
Equação 116 – Cálculo o116.....	194
Equação 117 – Cálculo o117.....	194
Equação 118 – Cálculo o118.....	198
Equação 119 – Cálculo o119.....	198
Equação 120 – Cálculo o120.....	199
Equação 121 – Cálculo o121.....	199
Equação 122 – Cálculo o122.....	199
Equação 123 – Cálculo o123.....	200
Equação 124 – Cálculo o124.....	200
Equação 125 – Cálculo o125.....	200
Equação 126 – Cálculo o126.....	201
Equação 127 – Cálculo o127.....	201

Nota: A25 e A26 não se encontram indicadas, pois estas foram usadas como variáveis auxiliares (permitindo guardar valores de transição, uma vez que estes são necessários para o bom funcionamento)

---

## APÊNDICE

# 2.1

---

### **Input**

*As variáveis referidas como Input, podem ser consultadas neste anexo.*





---

## APÊNDICE

# 2.1.1

---

### **Base para identificação da proposta**

*Anexo (Input) Base para identificação da proposta*

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

### 2.1.1 – Base para identificação da proposta (Input)

Todas as propostas são identificadas, neste ponto onde estarão apresentados os vários *Inputs* possíveis, relacionados com a base para identificação da proposta (tabela a2.1).

Tabela a2.1 - Base para identificação da proposta (variáveis input)

BASES PARA IDENTIFICAÇÃO DA PROPOSTA			
Nome Obra	i1		
Localização	i2		
	APLICAÇÃO DAS MARGENS		
	MARGEM "A" ATÉ		i3
	MARGEM "B" A PARTIR DE		
Ref <sup>a</sup>	i4	Revisão	i8
Ciente	i5		
DATA	i6		
A/C ENG <sup>o</sup> .	i7		

Na base para identificação da proposta (tabela a2.1) tem-se:

- i1 – Nome da obra, conforme expresso na memória descritiva e justificativa (MDJ);
- i2 - Localização da obra, conforme o concelho ou província destinatária;
- i3 - Limite na qual a margem A se altera para a margem B (Nota: margem A e B serão referidas posteriormente nos elementos compreendidos entre i10 e i14);
- i4 - Referência interna da empresa concorrente à obra;
- i5 - Identificação do cliente destinatário da proposta;
- i6 - Data para a qual a proposta é entregue ao destinatário (solicitante);
- i7 - Nome da pessoa responsável pela receção da proposta (lado do cliente);
- i8 - Número de revisão da proposta.

A tabela a2.2 demonstra um exemplo do preenchimento da tabela base para identificação da proposta, seguindo as regras acima referidas

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.1.1 – Base para identificação da proposta (Input)

Tabela a2.1 - Base para identificação da proposta (exemplo)

<b>Bases para identificação da proposta</b>			
Nome Obra	Luanda – Remodelação da loja		
Localização	Angola		
	<b>APLICAÇÃO DAS MARGENS</b>		
	MARGEM "A" ATÉ	50 000,00 €	
	MARGEM "B" A PARTIR DE		
Refª	Prop. 20180301	Revisão	R_00
Cliente	Construbild		
DATA	7 de março de 2018		
A/C ENGº:	Eurico Clemente		



---

**APÊNDICE**

**2.1.2**

---

**Base de  
cálculo para orçamentação**

*Anexo (Input) Base de cálculo para orçamentação*

## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.1.2 – Base de cálculo para orçamentação (Input)

Conjuntamente com as tabelas “base para identificação da proposta” (tabela a2.1 e a2.2) identificadas no anexo anterior, também as tabelas agora expostas (a2.3 e a2.4) compreendem outro conjunto de *Inputs* possíveis para a realização das propostas.

Nestas tabelas estão expressos os valores que vão influenciar o cálculo da orçamentação de forma dominante.

Referenciar ainda para o facto de que todos os valores apresentados deverão ser positivos, ou seja, superior a zero.

Tabela a2.1 - Base de cálculo para orçamentação (variáveis)

BASES DE CÁLCULO PARA ORÇAMENTAÇÃO		
Câmbio USD/EURO		i9
Margem EMP.(Orig.)	A	i10
Margem EMP.(Orig.)	B	i11
Margem TA(Dist.)	A	i12
Margem TA(Dist.)	B	i13
Factor Produtividade		i14
Seguro		i15
Custo FCL 40*		i16
Custo FCL 20*		i17
Trasportes Internos (Dist.)		i18
Trasportes Internos (Orig.)		i19
Desalfandegamento		i20
Direitos		i21
Retenção IRT		i22
Imposto Selo		i23
Custo Hora/Homem - 1		i24
Custo Hora/Homem - 2		i25
<b>Custo Hora/Homem - 3</b>		<b>i26</b>
<b>OUTROS CUSTOS</b>		<b>i44</b>

Na tabela da base de cálculo para orçamentação (tabela a2.3) tem-se:

i9 - Identificação do câmbio/relação entre as moedas referência de dois tipos distintos, particularmente, neste exemplo, será utilizada a relação do câmbio entre Angola e Portugal \$/€;

## **Apêndice nº2 - Formulas de cálculo**

### Posição de variáveis no cálculo (Bases)

#### 2.1.2 – Base de cálculo para orçamentação (Input)

- i10 - Margem de venda em Portugal (margem definida para elementos com valor inferior ao referido em i3);
- i11 - Margem de venda em Portugal (margem definida para elementos com valor superior ao referido em i3);
- i12 - Margem de venda no país de destino (margem definida para elementos com valor inferior ao referido em i3);
- i13 - Margem de venda no país de destino (margem definida para elementos com valor superior ao referido em i3);
- i14 - Definição por meio de percentagem da produtividade local;
- i15 - Valor do seguro (em percentagem);
- i16 - Valor de custo de um contentor de 40” (40 pés);
- i17 - Valor de custo de um contentor de 20” (20 pés);
- i18 - Custo com transporte de cada contentor no país de destino;
- i19 - Custo com transporte dos contentores em Portugal;
- i20 - Custo relacionados com o desalfandegamento (em percentagem);
- i21 - Custo relacionados com os direitos (em percentagem);
- i22 - Custo relacionados com a retenção IRT (em percentagem);
- i23 - Custo relacionados com o imposto selo (em percentagem);
- i24/ i25/ i26 - Definição de um dos três valores possíveis para o custo de mão-de-obra à hora (com referência Hora/Homem- 1, Hora/Homem- 2 ou Hora/Homem- 3 );
- i44 - Valor gasto relacionado com outros custos, como elementos exteriores ao fornecimento/realização dos trabalhos (em percentagem) (somente nas propostas com países de destino e de origem diferentes).

A tabela que se segue (a2.4) exemplifica o preenchimento tipo para a base de cálculo referidas neste anexo:



## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

### 2.1.2 – Base de cálculo para orçamentação (Input)

Tabela a2.2 - Base de cálculo para orçamentação (exemplo)

BASES DE CÁLCULO PARA ORÇAMENTAÇÃO		
Câmbio USD/EURO		1,40
Margem EMP.(Orig.)	A	15,00%
Margem EMP.(Orig.)	B	15,00%
Margem TA(Dist.)	A	25,00%
Margem TA(Dist.)	B	25,00%
Factor Produtividade		1,00
Seguro		1,50%
Custo FCL 40"		4 000,00 €
Custo FCL 20"		2 650,00
Trasportes Internos (Dist.)		\$ 1 725,00
Trasportes Internos (Orig.)		0,00
Desalfandegamento		10,50%
Direitos		0,00%
Retenção IRT		3,50%
Imposto Selo		1,00%
Custo Hora/Homem - 1		30,00 €
Custo Hora/Homem - 2		30,00 €
<b>Custo Hora/Homem - 3</b>		<b>30,00 €</b>
<b>OUTROS CUSTOS</b>		<b>0,00%</b>

---

**APÊNDICE**

**2.1.3**

---

**Base  
Auxiliar**

*Anexo (Input) Base Auxiliar*

## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.1.3 - Base Auxiliar (Input)

A base agora referida (auxiliar) consiste em elementos possíveis de adicionar a elaboração dos cálculos de orçamentação (cálculos suplementares), tendo como ponto de foco os tempos e equipas necessários para a conclusão do trabalho mencionado (tabela a2.5).

Tabela a2.1 - Base auxiliar (variáveis)

BASES AUXILIARES				
	Meses	as trabalháveis por mês	Duração/Dias	
Prazo de Execução Obra	i27	i28	o1	
	Horas	Dias	Homens	Equipa (1 Of+1 Aj)
AEROLICA	o2	o3	o4	o5
HIDRAULICA	o6	o7	o8	o9
ELECTRICIDADE	o10	o11	o12	o13
Total Obra	o14			
Dias	o15			
Homens	o16			
Equipa (1 Of+1 Aj)	o17			

Nesta tabela (tabela a2.5) tem-se:

i27 - Número de meses atribuído (pelo cliente) para a execução da obra;

i28 - Número de dias úteis/trabalháveis por mês.

Para mais informação sobre as variáveis apresentadas com a letra “i” (*input* programa) ver apêndice nº1 anexo 1.1 e variáveis com a letra “o” (*output* programa) ver anexo 1.2.

Seguindo os padrões estabelecidos (início do anexo com informação sobre a tabela de variáveis aplicadas seguida da tabela exemplo aplicado), segue-se a tabela a2.6 (apresenta um exemplo do preenchimento) base auxiliar:

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

### 2.1.3 - Base Auxiliar (Input)

Tabela a2.2 - Base auxiliar (exemplo)

BASES AUXILIARES				
	Meses	Dias trabalháveis por mês	Duração/Dias	
Prazo de Execução Obra	3	20	60	
	Horas	Dias	Homens	Equipa (1 Of+1 Aj)
AEROLICA	833,33	104,1666667	2	1
HIDRAULICA	25,00	3,125	0	0
ELECTRICIDADE	2,50	0,3125	0	0
Total Obra	860,83			
Dias	108			
Homens	2			
Equipa (1 Of+1 Aj)	1			



---

## APÊNDICE

# 2.1.4

---

### Base Contentores

*Anexo (Input) Base Contentores*

## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.1.4 – Base Contentores (Input)

A base referida (contentores) representam os *Inputs* relacionados com os contentores para transportes marítimos (tabelas a2.9 e tabela a2.10). Estas tabelas permitem a elaboração de uma proposta comercial internacional (país de origem diferente do país de destino).

Estes contentores são fabricados tendo em atenção o *standard* internacional, seguidamente são apresentadas algumas características destes.

**Tabela a2.7- Características técnicas/referências para contentores 20”**

	Dimensões Externas	Dimensões Internas	Abertura da Porta	Pesos
Comprimento	6058 mm	5906 mm	---	---
Largura	2438 mm	2340 mm	2346 mm	---
Altura	2591 mm	2388 mm	2282 mm	---
Cubagem	---	---	---	33,2 m <sup>3</sup>
Peso Máximo	---	---	---	2.4000 kg
Tara	---	---	---	2080 kg
Carga	---	---	---	21920 kg

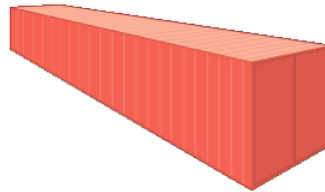
## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.1.4 – Base Contentores (Input)

**Tabela a1.8 - Características referências para contentores 40”**

**Standard 40 pés**  
É o mais utilizado entre todos devido à sua versatilidade, atendendo as necessidades de cargas secas, granéis e também as cargas úmidas e/ou líquidas.



	Dimensões Externas	Dimensões Internas	Abertura da Porta	Pesos
Comprimento	12.192 mm	12.044 mm	----	----
Largura	2438 mm	2.342 mm	2.337 mm	----
Altura	2591 mm	2.380 mm	2.280 mm	----
Cubagem	----	----	----	67,6 m3
Peso Máximo	----	----	----	30.480 kg
Tara	----	----	----	3.550 kg
Carga	----	----	----	26.930 kg

Seguidamente são apresentadas as bases de contentores (variáveis e exemplo), seguindo o esquema de explicação utilizado nas bases anteriores (primeira tabela representa as variáveis em fonte e a segunda tabela um exemplo de preenchimento).

**Tabela a2.9 - Bases contentores (variáveis)**

<b>Bases Contentores</b>			
Quantidades contentores		VALOR FOB	o18
i29	Q. FCL 20"	SEGURO	o19
i30	Q. FCL 40"	FRETE	o20
		TRANS. INTERNO	o21
		VALOR CIF	o22
		% FRETE	o23
		% T. Interno Dist.	o24



## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.1.4 – Base Contentores (Input)

Na tabela das bases contentores (tabela a2.9) tem-se:

i29 - Número de um contentor de 20” utilizados;

i30 - Número de um contentor de 40” utilizados.

**Tabela a2.10 - Bases contentores (exemplo)**

BASES CONTENTORES			
Quantidades contentores			
2,00	Q. FCL 20"	VALOR FOB	\$ 47 518,15
		SEGURO	\$ 3,49
1,00	Q. FCL 40"	FRETE	\$ 14 973,00
		TRANS. INTERNO	\$ 5 175,00
		VALOR CIF	\$ 67 669,64
		% FRETE	31,51%
		% T. Interno Dist.	10,89%

---

## APÊNDICE

# 2.1.5

---

### **Base para identificação dos elementos referentes ao mapa de quantidades**

*Anexo (Input) Base para identificação dos elementos referentes ao mapa de quantidades*

## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades (Material) (Input)

Uma das variáveis mais importantes para a identificação dos elementos é o seu tipo e subtipo (componente física). A utilização destes dois parâmetros surgem como uma ajuda à criação do mapa de quantidades, pois desta forma, é possível agrupar os elementos pelo seu tipo de utilização ou características, elaborando deste modo uma “árvore” de elementos, conforme defendido no *Manual do Formando* (2005).

Os valores para estas variáveis estão identificados na tabela a2.11.

**Tabela a2.11 - Valores possíveis para as variáveis i31 e i32**

Tipo (i31)	Subtipo (i32)
	Network
	Power
	Supply
	UPS
Electrical	Generator
Telecom	Productive
Security	Source
Waters	Specific
Sewage	Lighting
Building	Cables
Air	Sound
Mechanics	Panels
Other	Pipe
	Duct
	Fire
	CCTV

Após a identificação dos elementos é necessário listá-los. Para tal será criado um *Array* de elementos.

Devido ao tamanho da folha de cálculo, referente ao *Array* expresso nas tabelas a2.12 e a2.13, a mesma está repartida em várias divisórias, sendo que cada uma das linhas horizontais, dentro das bases para identificação dos elementos representa um *Array* distinto.

## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades (Material) (Input)

Tabela a2.12 - Extensão da folha de cálculo

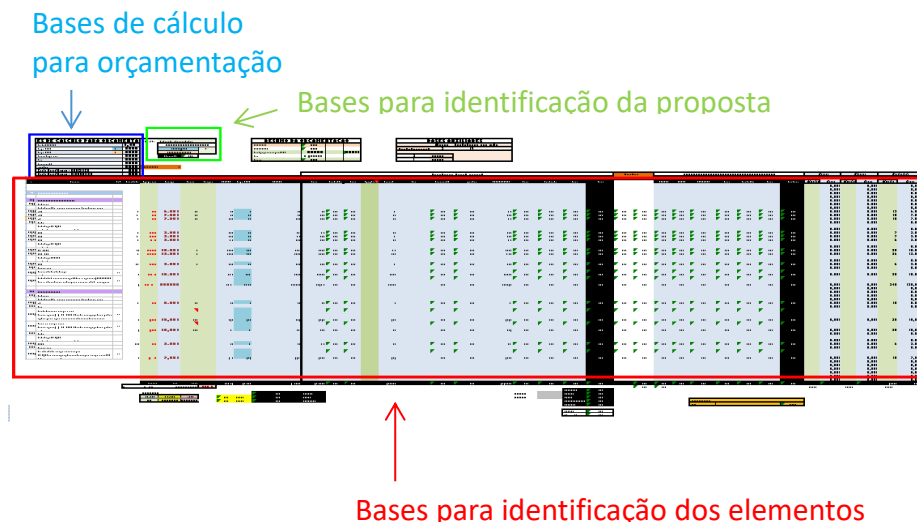
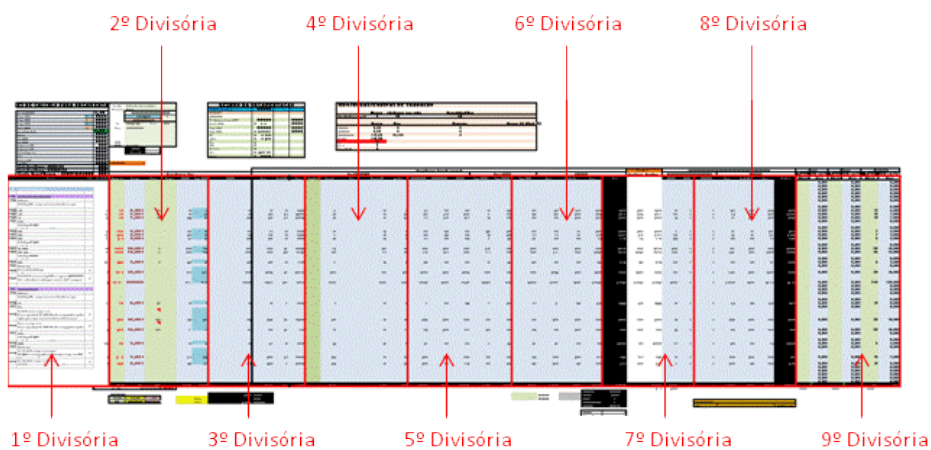


Tabela a2.13 - Divisórias da folha de cálculo



Conforme já referido anteriormente as bases para identificação dos elementos encontram-se repartidas em várias divisórias distintas. Estas divisórias apresentam-se

## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades (Material) (Input)

nas tabelas seguintes (tabela a2.14 a tabela a2.15). Nota ainda para o facto do ficheiro de cálculo encontra-se no *CD* entregue (formato Excel).

**Tabela a2.14 - Base para identificação dos elementos (variáveis)**

**Tabela a2.14.1 – 1ª Divisória**

Item	Descrição	Unid	Quantidades
i33	i34	i35	i36
i33	i34	i35	i36
i33	i34	i35	i36
i33	i34	i35	i36
i33	i34	i35	i36
i33	i34	i35	i36
i33	i34	i35	i36
i33	i34	i35	i36

**Tabela a2.14.2 – 2ª Divisória**

Valores Unitários em Euros (€)					
Equipamento	Montagem	Acessórios	Transportes	PC EMP.(Orig.)	Margem EMP.(Orig.)
<b>i37</b>	<b>o27</b>	i38	i39	o28	o29
<b>i37</b>	<b>o27</b>	i38	i39	o28	o29
<b>i37</b>	<b>o27</b>	i38	i39	o28	o29
<b>i37</b>	<b>o27</b>	i38	i39	o28	o29
<b>i37</b>	<b>o27</b>	i38	i39	o28	o29
<b>i37</b>	<b>o27</b>	i38	i39	o28	o29
<b>i37</b>	<b>o27</b>	i38	i39	o28	o29
<b>i37</b>	<b>o27</b>	i38	i39	o28	o29
o60	o61	o62	o63	o64	o65

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Input)

Tabela a2.14.3 – 3ª Divisória

		Valores Unitários em Dolares Americanos (\$)		
Valores Unitários em Euros (€)		Custos EMP. Dist. (USD)		
PV EMP.(Orig.)		Materiais	Mão de Obra	Unitário
o30		o31	o32	o33
o30		o31	o32	o33
o30		o31	o32	o33
o30		o31	o32	o33
o30		o31	o32	o33
o30		o31	o32	o33
o30		o31	o32	o33
o30		o31	o32	o33
o30		o31	o32	o33
o66		o67	o68	o69

Tabela a2.14.4 – 4ª Divisória

Valores Unitários em Dolares Americanos (\$)				
Custos EMP. Dist. (USD)				
Equip Locais	Desalfand.	Direitos	Retenção IRT	Imposto Selo
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
o70	o71	o72	o73	o74

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Input)

Tabela a2.14.5 – 5ª Divisória

Valores Unitários em Dolares Americanos (\$)			
Custos EMP. Dist. (USD)			Lucro EMP. Dist.
PC MATERIAIS (Dist.)	PC MÃO OBRA (Dist.)	PC EMP.(Dist.)	Margem EMP.(Dist.) Mat.
o38	o39	o40	o41
o38	o39	o40	o41
o38	o39	o40	o41
o38	o39	o40	o41
o38	o39	o40	o41
o38	o39	o40	o41
o38	o39	o40	o41
o38	o39	o40	o41
o75	o76	o77	o78

Tabela a2.14.6 – 6ª Divisória

Valores Unitários em Dolares Americanos (\$)			
Lucro EMP. Dist.		PV EMP. Dist.	
Margem EMP.(Dist.) MO	Margem EMP.(Dist.)	Materiais	Mão de obra
o42	o43	o44	o45
o42	o43	o44	o45
o42	o43	o44	o45
o42	o43	o44	o45
o42	o43	o44	o45
o42	o43	o44	o45
o42	o43	o44	o45
o42	o43	o44	o45
o79	o80	o81	o82

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Input)

Tabela a2.14.7 – 7ª Divisória

Valores Unitários em Dolares American		Outros Custos		PREÇO VENDA EMP. Dist. C	
PV EMP. Dist.		Valor Unitário	Valor Final	DESPESAS COM TRANSPOR	
Unitário	Parcial			SEGURO	FRETE
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o83	o84	o85	o86	o87	o88

Tabela 3.14.8 – 8ª Divisória

OM DESPESAS DE TRANSPORTE (\$)				
RTE	PV EMP. Dist.			
T. INTERNOS	Materiais	Mão de Obra	Unitário	Total Parcial
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o89	o90	o91	o92	o93



## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades (Material) (Input)

**Tabela a2.14.9 – 9ª Divisória**

<i>Custo Hora/Homem -1</i>		<i>Custo Hora/Homem -2</i>		<i>Custo Hora/Homem -3</i>	
Tempos	Custo MO Unitário	Tempos	Custo MO Unitário	Tempos	Custo MO Unitário
<i>Minutos</i>	<i>Euros</i>	<i>Minutos</i>	<i>Euros</i>	<i>Minutos</i>	<i>Euros</i>
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
o94	o95	o96	o97	o98	o99

Nas tabelas anteriores (tabelas a2.14.1 a a2.14.9) tem-se:

i33 - Código numérico endereço (ID) a um elemento específico;

i34 - Descrição ortográfica de um elemento específico;

i35 - Identificação do tipo de unidade, padrão expressos (tabela a2.15) podendo ser adicionadas outras unidades através do *Meno unit* , ver capítulo da tese 4.2.1.

**Tabela a2.15 - Valores possíveis para as variáveis i35**

<b>i35</b>	<b>Identificação</b>
Uh.	Unidade
Vg.	Conjunto
m	Metro
O	Outros

i36 - Identificação da quantidade de elementos referidos;

i37 - Valor unitário do equipamento;

i38 - Valor unitário dos acessórios (referentes ao equipamento identificado na variável anterior);

i39 - Valor referente ao transporte do equipamento;

## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades (Material) (Input)

i40 - Valor monetário referente aos equipamento locais;

i41 - Tempo de montagem do equipamento em minutos (tendo como referência o custo de Hora/Homem -1);

i42 - Tempo de montagem do equipamento em minutos (tendo como referência o custo de Hora/Homem -2);

i43 - Tempo de montagem do equipamento em minutos (tendo como referência o custo de Hora/Homem -3).

i45 - Definição das condições de venda.

Nota: O facto destas variáveis pertencerem a um *Array* significa que as variáveis referidas serão multiplicadas em quantidade pelo número de elementos. Querendo com isto dizer que para cada elemento haverá um i31, i32, i33,...até ao i43.

Esta multiplicação encontra-se especialmente visível na programação, uma vez que nesta estão identificadas as variáveis como lista (conjunto de elementos).

Nas tabelas a2.16.1 a a2.16.9 são apresentados os exemplos de preenchimento das bases para identificação dos elementos referentes às divisórias em causa.

**Tabela a2.1 - Base para identificação dos elementos (exemplo)**

**Tabela a2.16.1 – 1ª Divisória**

Item	Descrição	Unid	Quantidades
	Building		
	Lighting		
1	d 25	m	50
2	d 32	m	100
3	d 40	m	10

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Input)

Tabela a2.16.2 – 2ª Divisória

Valores Unitários em Euros (€)					
Equipamento	Montagem	Acessórios	Transportes	PC EMP.(Orig.)	Margem EMP.(Orig.)
0,35	500,00 €	0,22		0,57	0,09
0,53	7,50 €	0,33		0,86	0,13
0,67	7,50 €	0,44		1,11	0,17
77,20	25 825,00	48,40	0,00	125,60	18,84

Tabela a2.16.3 – 3ª Divisória

Valores Unitários em Euros (€)	Valores Unitários em Dolares Americanos (\$)		
	Custos EMP. Dist. (USD)		
PV EMP.(Orig.)	Materiais	Mão de Obra	Unitário
0,66	0,92	700,00	700,9177
0,99	1,38	10,50	11,8846
1,28	1,79	10,50	12,2871
144,44	202,22	36 155,00	36 357,22

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Input)

Tabela a2.16.4 – 4ª Divisória

Valores Unitários em Dolares Americanos (\$)				
Custos EMP. Dist. (USD)				
Equip Locais	Desalfand.	Direitos	Retenção IRT	Imposto Selo
	0,10	0,00	24,53	7,01
	0,15	0,00	0,42	0,12
	0,19	0,00	0,43	0,12
0,00	21,23	0,00	1 272,50	363,57

Tabela a2.16.5 – 5ª Divisória

Valores Unitários em Dolares Americanos (\$)			
Custos EMP. Dist. (USD)			Lucro EMP. Dist.
PC MATERIAIS (Dist.)	PC MÃO OBRA (Dist.)	PC EMP.(Dist.)	Margem EMP.(Dist.) Mat.
1,06	731,50	732,56	0,26
1,59	10,97	12,56	0,40
2,06	10,97	13,03	0,51
232,55	37 781,98	38 014,52	58,14

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Input)

Tabela a2.16.6 – 6ª Divisória

Valores Unitários em Dolares Americanos (\$)			
Lucro EMP. Dist.		PV EMP. Dist.	
Margem EMP.(Dist.) MO	Margem EMP.(Dist.)	Materiais	Mão de obra
182,88	183,14	1,32	914,38
2,74	3,14	1,99	13,72
2,74	3,26	2,57	13,72
9 445,49	9 503,63	290,69	47 227,47

Tabela a2.16.7 – 7ª Divisória

Valores Unitários em Dolares Americanos		Outros Custos		PREÇO VENDA EMP. Dist. C	
PV EMP. Dist.		Valor Unitário	Valor Final	DESPESAS COM TRANSPORTE	
Unitário	Parcial			SEGURO	FRETE
915,69	45 784,71	915,69	45 784,71	0,79	\$14 426,79
15,71	1 570,60	15,71	1 570,60	2,39	\$494,90
16,28	162,85	16,28	162,85	0,31	\$51,31
47 518,15	47 518,15		47 518,15	\$3,49	\$14 973,00

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.1.5 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Input)

Tabela a2.16.8 – 8ª Divisória

PREÇO VENDA EMP. Dist. COM DESPESAS DE TRANSPORTE (\$)					
PV EMP. Dist.					Total Parcial
T. INTERNOS	Materiais	Mão de Obra	Unitário		
\$4 986,22	389,60	914,38	1 303,97		65 198,51
\$171,05	8,67	13,72	22,39		2 238,93
\$17,73	9,50	13,72	23,22		232,20
\$5 175,00	20 442,17	47 227,47	67 669,64		\$67 669,64

Tabela a2.16.9 – 9ª Divisória

Custo Hora/Homem -1		Custo Hora/Homem -2		Custo Hora/Homem -3	
Tempos	Custo MO Unitário	Tempos	Custo MO Unitário	Tempos	Custo MO Unitário
Minutos	Euros	Minutos	Euros	Minutos	Euros
1000	500,00 €	15	7,50 €	15	7,50 €
833,33	25 000,00	25,00	750,00	2,50	75,00

Nótula para o facto de este anexo servir apenas de orientação para o leitor, não substituindo o texto/programa original identificado no Projeto de mestrado e caso surgimento de dúvidas sobre as fórmulas de cálculo verificar o ficheiro Excel em CD respetivo.



---

## APÊNDICE

# 2.2

---

### Cálculos e Output

*Neste anexo estão descritas as tabelas matemáticas utilizadas nas várias bases cominado nas várias expressões de cálculo (Extensão de cálculo).*





---

**APÊNDICE**

**2.2.1**

---

**Base  
Auxiliar**

*Anexo (Output) Base Auxiliar*

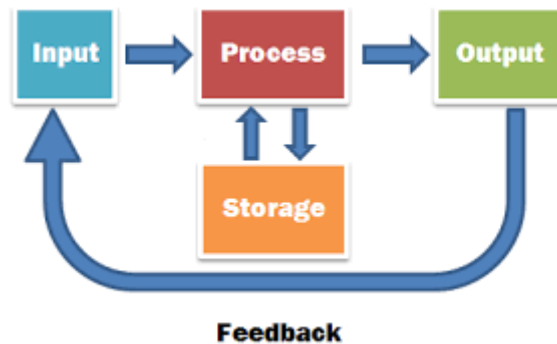
## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.2.1 - Base Auxiliar (Output)

A base referida (auxiliar) tem na sua fisionomia elementos de *input* e de *output*, sendo que as variáveis de *input* se encontram no anexo 2.1.3. A relação entre os dois tipos de variáveis também se aplica na sua componente inre-bases.

Tendo o anexo referido, como ponto de foco a componente a saída do sistema (figura 3.1), serão também abordadas e demonstrada a relação bilateral entre os *inputs* e os *outputs*. Para melhor compreender esta relação verifiquemos a imagem abaixo exposta.



Fonte: Google ®

Figura a1 - sistema fechado

Tendo em mente que a bases auxiliar, como já referido anteriormente consistem na tabela onde estão expressos os valores relacionados com a mão-de-obra (tabela a2.5).

## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.2.1 - Base Auxiliar (Output)

Tabela a2.1 - Base auxiliar (variáveis)

BASES AUXILIARES				
	Meses	as trabalhaveis por mês	Duração/Dias	
Prazo de Execução Obra	i27	i28	o1	
	Horas	Dias	Homens	Equipa (1 Of+1 Aj)
AEROLICA	o2	o3	o4	o5
HIDRAULICA	o6	o7	o8	o9
ELECTRICIDADE	o10	o11	o12	o13
Total Obra	o14			
Dias	o15			
Homens	o16			
Equipa (1 Of+1 Aj)	o17			

Temos como elementos a referir nos vários cálculos elaborados ao longo desta tabela, os elementos abaixo discriminados:

**o1** - Número total de dias atribuído (pelo cliente) para a execução da obra;

$$o1 = i27 \times i28 \text{ [dias]}$$

#### Equação 1 – Cálculo o1

i27 - Definição do número de meses atribuído (pelo cliente) para a execução da obra;

i28 - Definição do número de dias úteis/trabalháveis por mês.

**o2** - Número de horas total relacionada com o custo hora/homem -1;

$$o2 = o94 \times i14 \text{ [horas]}$$

#### Equação 2 - Cálculo o2

i14 - Definição por meio de percentagem da produtividade local;

o94 - Somatório temporal (em horas) relativo à mão-de-obra, do tipo 1;

## **Apêndice nº2 - Formulas de cálculo**

### Posição de variáveis no cálculo (Bases)

#### 2.2.1 - Base Auxiliar (Output)

**o3** - Número de dias totais relacionada com o custo hora/homem -1;

$$o3 = o2 \div 8 \text{ [dias]}$$

#### **Equação 3 - Cálculo o3**

o2 - Número de horas total relacionada com o custo hora/homem -1;

**o4** - Número de homens necessário, para realizar os trabalhos relacionados com o custo hora/homem -1;

$$o4 = o3 \div o1 \text{ [número de homens]}$$

#### **Equação 4 - Cálculo o4**

o1 - Número total de dias atribuído (pelo cliente) para a execução da obra;

o3 - Número de dias totais relacionada com o custo hora/homem -1;

**o5** - Número de equipas compostas por um oficial mais um ajudante (custo hora/homem -1);

$$o5 = o4 \div 2 \text{ [número de equipas]}$$

#### **Equação 5 - Cálculo o5**

o4 - Número de homens necessário, para realizar os trabalhos relacionados com o custo hora/homem -1;

## **Apêndice nº2 - Formulas de cálculo**

### Posição de variáveis no cálculo (Bases)

#### 2.2.1 - Base Auxiliar (Output)

**o6** - Número de horas total relacionada com o custo hora/homem -2;

$$o6 = o96 \times i14 \text{ [horas]}$$

#### **Equação 6 - Cálculo o6**

i14 - Definição por meio de percentagem da produtividade local;

o96 - Somatório temporal (em horas) relativo à mão-de-obra, do tipo 2;

**o7** - Número de dias total relacionada com o custo hora/homem -2;

$$o7 = o6 \div 8 \text{ [dias]}$$

#### **Equação 7 - Cálculo o7**

o6 - Número de horas total relacionada com o custo hora/homem -2;

**o8** - Número de homens necessários, para realizar os trabalhos relacionados com o custo hora/homem -2;

$$o8 = o7 \div o1 \text{ [número de homens]}$$

#### **Equação 8 - Cálculo o8**

o1 - Número total de dias atribuído (pelo cliente) para a execução da obra;

o7 - Número de dias total relacionada com o custo hora/homem -2;

## **Apêndice nº2 - Formulas de cálculo**

### Posição de variáveis no cálculo (Bases)

#### 2.2.1 - Base Auxiliar (Output)

**o9** - Número de equipas compostas por um oficial mais um ajudante (custo hora/homem -2);

$$o9 = o8 \div 2 \text{ [número de equipas]}$$

#### **Equação 9 - Cálculo o9**

**o8** - Número de homens necessários, para realizar os trabalhos relacionados com o custo hora/homem -2;

**o10** - Número de horas total relacionada com o custo hora/homem -3;

$$o10 = o98 \times i14 \text{ [horas]}$$

#### **Equação 10 - Cálculo o10**

i14 - Definição por meio de percentagem da produtividade local;

o98 - Somatório temporal (em horas) relativo à mão-de-obra, do tipo 3;

**o11** - Número de dias total relacionada com o custo hora/homem -3;

$$o11 = o10 \div 8 \text{ [dias]}$$

#### **Equação 11 - Cálculo o11**

o10 - Número de horas total relacionada com o custo hora/homem -3;

## **Apêndice nº2 - Formulas de cálculo**

### Posição de variáveis no cálculo (Bases)

#### 2.2.1 - Base Auxiliar (Output)

**o12** - Número de homens necessários, para realizar os trabalhos relacionados com o custo hora/homem -3;

$$o12 = o11 \div o1 \text{ [número de homens]}$$

#### **Equação 12 - Cálculo o12**

o1 - Número total de dias atribuído (pelo cliente) para a execução da obra;

o11 - Número de dias total relacionada com o custo hora/homem -3;

**o13** - Número de equipas compostas por um oficial mais um ajudante (custo hora/homem -3);

$$o13 = o12 \div 2 \text{ [número de equipas]}$$

#### **Equação 13 - Cálculo o13**

o12 - Número de homens necessários, para realizar os trabalhos relacionados com o custo hora/homem -3;

**o14** - Número total de horas gasto na execução da obra;

$$o14 = o2 + o6 + o10 \text{ [horas]}$$

#### **Equação 14 - Cálculo o14**

o2 - Número de horas total relacionada com o custo hora/homem -1;

o6 - Número de horas total relacionada com o custo hora/homem -2;

o10 - Número de horas total relacionada com o custo hora/homem -3;



## **Apêndice nº2 - Formulas de cálculo**

### Posição de variáveis no cálculo (Bases)

#### 2.2.1 - Base Auxiliar (Output)

**o15** - Número total de dias gasto na realização da empreitada;

$$o15 = o14 \div 8 \text{ [dias]}$$

**Equação 15 - Cálculo o15**

o14 - Número total de horas gasto na execução da obra;

**o16** - Número total de homens necessários, para realizar os trabalhos;

$$o16 = o15 \div o1 \text{ [número de homens]}$$

**Equação 16 - Cálculo o16**

o1 - Número total de dias atribuído (pelo cliente) para a execução da obra;

o15 - Número total de dias gasto na realização da empreitada;

**o17** - Número de equipas totais compostas por um oficial mais um ajudante;

$$o17 = o16 \div 2 \text{ [número de equipas]}$$

**Equação 17 - Cálculo o17**

o16 - Número total de homens necessários, para realizar os trabalhos;

Seguindo os padrões estabelecidos (início tabela de variáveis aplicadas seguida da tabela exemplo aplicado), segue-se a seguir a tabela a2.6 que apresenta um exemplo do preenchimento da base auxiliar:

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

### 2.2.1 - Base Auxiliar (Output)

Tabela a2.2 - Base auxiliar (exemplo)

BASES AUXILIARES				
	Meses	Dias trabalháveis por mês	Duração/Dias	
Prazo de Execução Obra	3	20	60	
	Horas	Dias	Homens	Equipa (1 Of+1 Aj)
AEROLICA	833,33	104,1666667	2	1
HIDRAULICA	25,00	3,125	0	0
ELECTRICIDADE	2,50	0,3125	0	0
Total Obra	860,83			
Dias	108			
Homens	2			
Equipa (1 Of+1 Aj)	1			



---

## APÊNDICE

# 2.2.2

---

### Base Contentores

*Anexo (Output) Base Contentores*

## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.2.2 – Base Contentores (Output)

A base contentores, conforme já referido anteriormente é a tabela onde estão expressos os valores relacionados com transporte e aluguer dos contentores marítimos. Neste anexo estão informações alusivas aos *Outputs*, sendo que os *Inputs* desta base foram identificados no apêndice 2 anexo 2.1.4.

Utilizado o modelo de explicação defendido, são apresentadas as bases de contentores (variáveis e exemplo).

**Tabela a2.9 - Base contentores (variáveis)**

<b>Bases Contentores</b>			
Quantidades contentores		VALOR FOB	o18
i29	Q. FCL 20"	SEGURO	o19
i30	Q. FCL 40"	FRETE	o20
		TRANS. INTERNO	o21
		VALOR CIF	o22
		% FRETE	o23
		% T. Interno Dist.	o24

Nesta tabela tem-se:

**o18** - Somatório dos custos finais dos equipamentos no país de destino;

$$o18 = o84 \text{ [€]}$$

**Equação 1 - Cálculo o18**

**o84** - Somatório dos preços de venda totais dos vários equipamentos, no país de destino;

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.2.2 – Base Contentores (Output)

**o19** - Custo derivado ao seguro dos contentores;

$o19 = [o66 \times i15 \times (1 + (i10 \text{ ou } i11))] \times i9$ ; Dependendo da relação entre o o66 e o i3 [€]

### Equação 2 - Cálculo o19

Se  $o66 < i3$ , valor a assumir na fórmula i10 (margem menor)

Se  $o66 > i3$ , valor a assumir na fórmula i11 (margem maior)

i3 - Descrição do limite na qual a margem A se altera para a margem B;

i9 - Identificação do câmbio/relação entre as moedas referência de dois tipos distintos;

i10 - Identificação da margem de venda em Portugal (margem definida para elementos com valor inferior ao referido em i3);

i11 - Identificação da margem de venda em Portugal (margem definida para elementos com valor superior ao referido em i3);

i15 - Definição do valor do seguro (em percentagem);

o66 - Somatório dos preços de venda, referente aos equipamentos no país de envio;

**o20** - custo derivado ao frete marítimo;

$o20 = [i29 \times (i17 \times i9) + i30(i16 \times i9)] + [i29 \times (i17 \times i9) + i30(i16 \times i9)] \times (i10 \text{ ou } i11)$ ; Dependendo da relação entre o i3 e  $(i29 \times i17 + i30 \times i16)$  [€]

### Equação 3 - Cálculo o20

Se  $i29 \times i17 + i30 \times i16 < i3$ , valor a assumir na fórmula i10 (margem menor)

Se  $i29 \times i17 + i30 \times i16 > i3$ , valor a assumir na fórmula i11 (margem maior)

i3 - Descrição do limite na qual a margem A se altera para a margem B;

i9 - Identificação do câmbio/relação entre as moedas referência de dois tipos distintos;

i10 - Identificação da margem de venda em Portugal (margem definida para elementos com valor inferior ao referido em i3);

i11 - Identificação da margem de venda em Portugal (margem definida para elementos com valor superior ao referido em i3);

## **Apêndice nº2 - Formulas de cálculo**

### Posição de variáveis no cálculo (Bases)

#### 2.2.2 – Base Contentores (Output)

i16 - Representação do valor de custo de um contentor de 40” (40 pés);

i17 - Representação do valor de custo de um contentor de 20” (20 pés);

i29 - Representação do número de um contentor de 20” utilizados;

i30 - Representação do número de um contentor de 40” utilizados.

**o21** - Custo relacionado com o transporte interno no país de destino;

$$o21 = (i29 + i30) \times i18 \text{ [€]}$$

#### **Equação 4 - Cálculo o21**

i18 - Identificação do custo com transporte de cada contentor no país de destino;

i29 - Representação do número de um contentor de 20” utilizados;

i30 - Representação do número de um contentor de 40” utilizados.

**o22** - Somatório dos elementos referenciados como bases contentores;

$$o22 = o18 + o19 + o20 + o21 \text{ [€]}$$

#### **Equação 5 - Cálculo o22**

o18 - Somatório dos custos finais dos equipamentos no país de destino;

o19 - Custo derivado ao seguro dos contentores;

o20 - custo derivado ao frete marítimo;

o21 - Custo relacionado com o transporte interno no país de destino;

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

### 2.2.2 – Base Contentores (Output)

**o23** - Valor do custo relacionado com o frete marítimo (em percentagem);

$$o23 = o20 \div o84 \text{ [€]}$$

#### Equação 6 - Cálculo o23

o20 - custo derivado ao frete marítimo;

o84 - Somatório dos preços de venda totais dos vários equipamentos, no país de destino;

**o24** - Valor referente ao custo relacionado com o transporte interno no país de destino (em percentagem);

$$o24 = o21 \div o84 \text{ [€]}$$

#### Equação 7 - Cálculo o24

o21- Custo relacionado com o transporte interno no país de destino;

o84 - Somatório dos preços de venda totais dos vários equipamentos, no país de destino;

Apresente-se um exemplo de preenchimento da tabela

**Tabela a2.10 - Bases contentores (exemplo)**

BASES CONTENTORES			
Quantidades contentores		VALOR FOB	\$ 47 518,15
2,00	Q. FCL 20"	SEGURO	\$ 3,49
1,00	Q. FCL 40"	FRETE	\$ 14 973,00
		TRANS. INTERNO	\$ 5 175,00
		VALOR CIF	\$ 67 669,64
		% FRETE	31,51%
		% T. Interno Dist.	10,89%





---

## APÊNDICE

# 2.2.3

---

### **Base para identificação dos elementos referentes ao mapa de quantidades**

*Anexo (Output) Base para identificação dos elementos referentes ao mapa de quantidades*

## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades (Material) (Output)

Conforme o próprio nome sugere na base para identificação dos elementos, é apresentada uma descrição dos vários elementos, bem como os cálculos por detrás dos mesmos.

Análogo ao já referido anteriormente a base para identificação dos elementos encontram-se repartidas em várias divisórias distintas. Estas divisórias apresentam-se nas tabelas seguintes (tabela a2.14 a tabela a2.15, variáveis e exemplos respetivamente).

**Tabela a2.14 - Bases para identificação dos elementos (variáveis)**

**Tabela a2.14.1 – 1ª Divisória**

Item	Descrição	Unid	Quantidades
i33	i34	i35	i36
i33	i34	i35	i36
i33	i34	i35	i36
i33	i34	i35	i36
i33	i34	i35	i36
i33	i34	i35	i36
i33	i34	i35	i36
i33	i34	i35	i36

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

Tabela a2.14.2 – 2ª Divisória

Valores Unitários em Euros (€)					
Equipamento	Montagem	Acessórios	Transportes	PC EMP.(Orig.)	Margem EMP.(Orig.)
i37	o27	i38	i39	o28	o29
i37	o27	i38	i39	o28	o29
i37	o27	i38	i39	o28	o29
i37	o27	i38	i39	o28	o29
i37	o27	i38	i39	o28	o29
i37	o27	i38	i39	o28	o29
i37	o27	i38	i39	o28	o29
i37	o27	i38	i39	o28	o29
o60	o61	o62	o63	o64	o65

Tabela a2.14.3 – 3ª Divisória

Valores Unitários em Dolares Americanos (\$)			
Valores Unitários em Euros (€)	Custos EMP. Dist. (USD)		
PV EMP.(Orig.)	Materiais	Mão de Obra	Unitário
o30	o31	o32	o33
o30	o31	o32	o33
o30	o31	o32	o33
o30	o31	o32	o33
o30	o31	o32	o33
o30	o31	o32	o33
o30	o31	o32	o33
o30	o31	o32	o33
o30	o31	o32	o33
o66	o67	o68	o69

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

Tabela a2.14.4 – 4ª Divisória

Valores Unitários em Dolares Americanos (\$)				
Custos EMP. Dist. (USD)				
Equip Locais	Desalfand.	Direitos	Retenção IRT	Imposto Selo
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
i40	o34	o35	o36	o37
o70	o71	o72	o73	o74

Tabela a2.14.5 – 5ª Divisória

Valores Unitários em Dolares Americanos (\$)			
Custos EMP. Dist. (USD)			Lucro EMP. Dist.
PC MATERIAIS (Dist.)	PC MÃO OBRA (Dist.)	PC EMP.(Dist.)	Margem EMP.(Dist.) Mat.
o38	o39	o40	o41
o38	o39	o40	o41
o38	o39	o40	o41
o38	o39	o40	o41
o38	o39	o40	o41
o38	o39	o40	o41
o38	o39	o40	o41
o38	o39	o40	o41
o38	o39	o40	o41
o75	o76	o77	o78

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

Tabela a2.14.6 – 6ª Divisória

Valores Unitários em Dolares Americanos (\$)			
Lucro EMP. Dist.		PV EMP. Dist.	
Margem EMP.(Dist.) MO	Margem EMP.(Dist.)	Materiais	Mão de obra
o42	o43	o44	o45
o42	o43	o44	o45
o42	o43	o44	o45
o42	o43	o44	o45
o42	o43	o44	o45
o42	o43	o44	o45
o42	o43	o44	o45
o42	o43	o44	o45
o79	o80	o81	o82

Tabela a2.14.7 – 7ª Divisória

Valores Unitários em Dolares Americanos		Outros Custos		PREÇO VENDA EMP. Dist. C	
PV EMP. Dist.		Valor Unitário	Valor Final	DESPESAS COM TRANSPORTE	
Unitário	Parcial			SEGURO	FRETE
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o46	o47	o48	o49	o50	o51
o83	o84	o85	o86	o87	o88

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

Tabela 3.14.8 – 8ª Divisória

OM DESPESAS DE TRANSPORTE (\$)				
RTE	PV EMP. Dist.			
T. INTERNOS	Materiais	Mão de Obra	Unitário	Total Parcial
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o52	o53	o54	o55	o56
o89	o90	o91	o92	o93

Tabela a2.14.9 – 9ª Divisória

Custo Hora/Homem -1		Custo Hora/Homem -2		Custo Hora/Homem -3	
Tempos	Custo MO Unitário	Tempos	Custo MO Unitário	Tempos	Custo MO Unitário
Minutos	Euros	Minutos	Euros	Minutos	Euros
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
i41	o57	i42	o58	i43	o59
o94	o95	o96	o97	o98	o99

## **Apêndice nº2 - Formulas de cálculo**

### Posição de variáveis no cálculo (Bases)

#### 2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades (Material) (Output)

Na base referida temo de destacar:

**o27** - Preço da montagem, referente a um equipamento específico (unitário);

$$o27 = o57 + o58 + o59 \text{ [€]}$$

#### **Equação 1 - Cálculo o27**

o57 - Preço de custo relativo à mão-de-obra, do tipo 1;

o58 - Preço de custo relativo à mão-de-obra, do tipo 2;

o59 - Preço de custo relativo à mão-de-obra, do tipo 3;

**o28** - Preço de custo, referente a um equipamento específico (unitário);

$$A28 = i37 + i38 + i39 \text{ [€]}$$

#### **Equação 2 - Cálculo o28**

i37 - Definição do valor unitário do equipamento;

i38 - Definição do valor unitário dos acessórios (referentes ao equipamento identificado na variável anterior);

i39 - Definição do valor referente ao transporte do equipamento;



## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o29** - Margem no equipamento referente ao país de envio (unitária);

$$o29 = o28 \times (i10 \text{ ou } i11) \quad ; \text{ Dependendo da relação entre } o37 \text{ e } o3 \text{ [€]}$$

### **Equação 3 - Cálculo o29**

i3 - Descrição do limite na qual a margem A se altera para a margem B;

i10 - Identificação da margem de venda em Portugal (margem definida para elementos com valor inferior ao referido em i3);

i11 - Identificação da margem de venda em Portugal (margem definida para elementos com valor superior ao referido em i3);

o28 - Preço de custo, referente a um equipamento específico (unitário);

i37 - Definição do valor unitário do equipamento;

**o30** - Preço de venda, referente a um equipamento específico no país de envio (unitário);

$$o30 = o28 + o29 \text{ [€]}$$

### **Equação 4 - Cálculo o30**

o28 - Preço de custo, referente a um equipamento específico (unitário);

o29 - Margem no equipamento referente ao país de envio (unitária);

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o31** - Custo do material físico, no país de destino (unitário);

$$o31 = o30 \times i9 \text{ [€]}$$

**Equação 5 - Cálculo o31**

i9 - Identificação do câmbio/relação entre as moedas referência de dois tipos distintos,

o30 - Preço de venda, referente a um equipamento específico no país de envio  
(unitário);

**o32** - Custo relacionado com a mão-de-obra, no país de destino (unitário);

$$o32 = o27 \times i14 \times i9 \text{ [€]}$$

**Equação 6 - Cálculo o32**

i9 - Identificação do câmbio/relação entre as moedas referência de dois tipos distintos,

i14 - Definição por meio de percentagem da produtividade local;

o27 - Preço da montagem, referente a um equipamento específico (unitário);

**o33** - Custo referente ao equipamento, no país de destino (unitário);

$$o33 = o31 + o32 \text{ [€]}$$

**Equação 7 - Cálculo o33**

o31 - Custo do material físico, no país de destino (unitário);

o32 - Custo relacionado com a mão-de-obra, no país de destino (unitário);

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o34** - Valor de custo destinado ao desalfandegamento (unitário);

$$o34 = o31 \times i20 \text{ [€]}$$

**Equação 8 - Cálculo o34**

i20 - Definição dos custo relacionados com o desalfandegamento (em percentagem);

o31 - Custo do material físico, no país de destino (unitário);

**o35** - Valor destinado aos direitos (unitário);

$$o35 = o31 \times i21 \text{ [€]}$$

**Equação 9 - Cálculo o35**

i21 - Identificação dos custo relacionados com os direitos (em percentagem);

o31 - Custo do material físico, no país de destino (unitário);

**o36** - Valor destinado a retenção IRT (unitário);

$$o36 = o33 \times i22 \text{ [€]}$$

**Equação 10 - Cálculo o36**

i22 - Identificação dos custo relacionados com a retenção IRT (em percentagem);

o33 - Custo referente ao equipamento, no país de destino (unitário);

## **Apêndice nº2 - Formulas de cálculo**

### Posição de variáveis no cálculo (Bases)

#### 2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades (Material) (Output)

**o37** - Valor destinado ao imposto selo (unitário);

$$o37 = o33 \times i23 \text{ [€]}$$

#### **Equação 11 - Cálculo o37**

i23 - Identificação dos custo relacionados com o imposto selo (em percentagem);

o33 - Custo referente ao equipamento, no país de destino (unitário);

**o38** - Custo físico total, referente a um equipamento específico no país de destino (unitário);

$$o38 = (o31 + i40 + o34 + o35) + (o31 \times i22) + (o31 \times i23) \text{ [€]}$$

#### **Equação 12 - Cálculo o38**

i22 - Identificação dos custo relacionados com a retenção IRT (em percentagem);

i23 - Identificação dos custo relacionados com o imposto selo (em percentagem);

o31 - Custo do material físico, no país de destino (unitário);

o34 - Valor de custo destinado ao desalfandegamento (unitário);

o35 - Valor destinado aos direitos (unitário);

i40 - Identificação do valor monetário referente aos equipamento locais;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o39** - Custo alusivo a mão-de-obra total, referente a um equipamento específico no país de destino (unitário);

$$o39 = o32 + (o32 \times i22) + (o32 \times i23) \text{ [€]}$$

**Equação 13 - Cálculo o39**

i22 - Identificação dos custo relacionados com a retenção IRT (em percentagem);

i23 - Identificação dos custo relacionados com o imposto selo (em percentagem);

o32 - Custo relacionado com a mão-de-obra, no país de destino (unitário);

**o40** - Custo total, referente a um equipamento específico no país de destino (unitário);

$$o40 = o38 + o39 \text{ [€]}$$

**Equação 14 - Cálculo o40**

o38 - Custo físico total, referente a um equipamento específico no país de destino  
(unitário);

o39 - Custo alusivo a mão-de-obra total, referente a um equipamento específico no país de destino (unitário);

**o41** - Margem no equipamento físico referente ao país de destino (unitária);

$$o41 = o38 \times (i12 \text{ ou } i13) \text{ ; Dependendo da relação entre o } i37 \text{ e o } i3 \text{ [€]}$$

**Equação 15 - Cálculo o41**

i3 - Descrição do limite na qual a margem A se altera para a margem B;

## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades (Material) (Output)

i12 - Representação da margem de venda no país de destino (margem definida para elementos com valor inferior ao referido em i3);

i13 - Identificação da margem de venda no país de destino (margem definida para elementos com valor superior ao referido em i3);

i37 - Definição do valor unitário do equipamento;

o38 - Custo físico total, referente a um equipamento específico no país de destino (unitário);

**o42** - Margem relacionada com a mão-de-obra, referente ao país de destino (unitário);

$o42 = o39 \times (i12 \text{ ou } i13)$  ; Dependendo da relação entre o i37 e o i3 [€]

#### Equação 16 - Cálculo o42

i3 - Descrição do limite na qual a margem A se altera para a margem B;

i12 - Representação da margem de venda no país de destino (margem definida para elementos com valor inferior ao referido em i3);

i13 - Identificação da margem de venda no país de destino (margem definida para elementos com valor superior ao referido em i3);

i37 - Definição do valor unitário do equipamento;

o39 - Custo alusivo a mão-de-obra total, referente a um equipamento específico no país de destino (unitário);

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o43** - Margem total no equipamento referente ao país de destino (unitária);

$$Ao43 = o40 \times (i12 \text{ ou } i13); \text{ Dependendo da relação entre o } i37 \text{ e o } i3 \text{ [€]}$$

### **Equação 17 - Cálculo o43**

i3 - Descrição do limite na qual a margem A se altera para a margem B;

i12 - Representação da margem de venda no país de destino (margem definida para elementos com valor inferior ao referido em i3);

i13 - Identificação da margem de venda no país de destino (margem definida para elementos com valor superior ao referido em i3);

i37 - Definição do valor unitário do equipamento;

o40 - Custo total, referente a um equipamento específico no país de destino (unitário);

**o44** - Preço de venda dos materiais, no país de destino (unitário);

$$o44 = o38 + A41 \text{ [€]}$$

### **Equação 18 - Cálculo Ao44**

o38 - Custo físico total, referente a um equipamento específico no país de destino (unitário);

o41 - Margem no equipamento físico referente ao país de destino (unitária);

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o45** - Preço de venda referente a mão-de-obra, no país de destino (unitário);

$$o45 = o39 + o42 \text{ [€]}$$

**Equação 19 - Cálculo o45**

o39 - Custo alusivo a mão-de-obra total, referente a um equipamento específico no país de destino (unitário);

o42 - Margem relacionada com a mão-de-obra, referente ao país de destino (unitário);

**o46** - Preço de venda total do equipamento, no país de destino (unitário);

$$o46 = o40 + o43 \text{ [€]}$$

**Equação 20 - Cálculo o46**

o40 - Custo total, referente a um equipamento específico no país de destino (unitário);

o43 - Margem total no equipamento referente ao país de destino (unitária);

**o47** - Preço de venda total dos vários equipamentos, referenciados como iguais e quantificados na mesma célula, no país de destino (parcial);

$$o47 = o46 \times i36 \text{ [€]}$$

**Equação 21 - Cálculo o47**

i36 - Identificação da quantidade de elementos referidos;

o46 - Preço de venda total do equipamento, no país de destino (unitário);



## **Apêndice nº2 - Formulas de cálculo**

### Posição de variáveis no cálculo (Bases)

#### 2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades (Material) (Output)

**o48** - Valor gasto relacionado com outros custos, como elementos exteriores ao fornecimento/realização dos trabalhos (unitário);

$$o48 = o46 \times i44 \text{ [€]}$$

#### **Equação 22 - Cálculo o48**

A46 - Preço de venda total do equipamento, no país de destino (unitário);

i44 - Identificação do valor gasto relacionado com outros custos, como elementos exteriores ao fornecimento/realização dos trabalhos (em percentagem);

**o49** - Valor gasto relacionado com outros custos, como elementos exteriores ao fornecimento/realização dos trabalhos (parcial);

$$o49 = o48 \times i36 \text{ [€]}$$

#### **Equação 23 - Cálculo o49**

i36 - Identificação da quantidade de elementos referidos;

o48 - Valor gasto relacionado com outros custos, como elementos exteriores ao fornecimento/realização dos trabalhos (unitário);

## Apêndice nº2 - Formulas de cálculo

### Posição de variáveis no cálculo (Bases)

#### 2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades (Material) (Output)

**o50** - Valor destinado as despesas com transporte (seguro);

$o50 = i36 \times o30 \times i15 \times [1 + (i10 \text{ ou } i11)] \times i9$ ; Dependendo da relação entre o  $o66 \times i15$  e o  $i3$  [€]

#### Equação 24 - Cálculo o50

i3 - Descrição do limite na qual a margem A se altera para a margem B;

i9 - Identificação do câmbio/relação entre as moedas referência de dois tipos distintos;

i10 - Identificação da margem de venda em Portugal (margem definida para elementos com valor inferior ao referido em i3);

i11 - Identificação da margem de venda em Portugal (margem definida para elementos com valor superior ao referido em i3);

i15 - Definição do valor do seguro (em percentagem);

o30 - Preço de venda, referente a um equipamento específico no país de envio (unitário);

i36 - Identificação da quantidade de elementos referidos;

o66 - Somatório dos preços de venda, referente aos equipamentos no país de envio;

**o51** - Valor destinado as despesas com transporte (frete);

$o51 = o47 \times (o20 \div o84)$  [€]

#### Equação 25 - Cálculo o51

o20 - custo derivado ao frete marítimo;

o47 - Preço de venda total dos vários equipamentos, referenciados como iguais e quantificados na mesma célula, no país de destino (parcial);

o84 - Somatório dos preços de venda totais dos vários equipamentos, no país de destino;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o52** - Valor destinado as despesas com transporte interno, no país de destino;

$$o52 = o47 \times (o21 \div o84) [\text{€}]$$

**Equação 26 - Cálculo o52**

o21 - Custo relacionado com o transporte interno no país de destino;

o47 - Preço de venda total dos vários equipamentos, referenciados como iguais e  
quantificados na mesma célula, no país de destino (parcial);

o84 - Somatório dos preços de venda totais dos vários equipamentos, no país de destino;

**o53** - Preço de venda dos materiais e do transporte, no país de destino (unitário);

$$o53 = o44 + (o50 + o51 + o52) \div i36 [\text{€}]$$

**Equação 27 - Cálculo o53**

i36 - Identificação da quantidade de elementos referidos;

o44 - Preço de venda dos materiais, no país de destino (unitário);

o50 - Valor destinado as despesas com transporte (seguro);

o51 - Valor destinado as despesas com transporte (frete);

o52 - Valor destinado as despesas com transporte interno, no país de destino;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o54** - Preço referente a mão-de-obra, no país de destino (unitário);

$$Ao54 = o45 \text{ [€]}$$

**Equação 28 - Cálculo o54**

o45 - Preço de venda referente a mão-de-obra, no país de destino (unitário);

**o55** - Preço de venda total e do transporte, no país de destino (unitário);

$$o55 = o53 + o54 \text{ [€]}$$

**Equação 29 - Cálculo o55**

o53 - Preço de venda dos materiais e do transporte, no país de destino (unitário);

o54 - Preço referente a mão-de-obra, no país de destino (unitário);

**o56** - Preço de venda total e do transporte, no país de destino (parcial);

$$o56 = o47 + o50 + o51 + o52 \text{ [€]}$$

**Equação 30 - Cálculo o56**

o47 - Preço de venda total dos vários equipamentos, referenciados como iguais e quantificados na mesma célula, no país de destino (parcial);

o50 - Valor destinado as despesas com transporte (seguro);

o51 - Valor destinado as despesas com transporte (frete);

o52 - Valor destinado as despesas com transporte interno, no país de destino;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o57** - Preço de custo relativo à mão-de-obra, do tipo 1;

$$o57 = i41 \times (i24 \div 60) [\text{€}]$$

**Equação 31 - Cálculo o57**

i24 - Definição de um dos três valores possíveis para o custo de mão-de-obra à hora;

i41 - Identificação do tempo de montagem do equipamento em minutos (tendo como referência o custo de Hora/Homem -1);

**o58** - Preço de custo relativo à mão-de-obra, do tipo 2;

$$o58 = i42 \times (i25 \div 60) [\text{€}]$$

**Equação 32 - Cálculo o58**

i25 - Definição de um dos três valores possíveis para o custo de mão-de-obra à hora;

i42 - Identificação do tempo de montagem do equipamento em minutos (tendo como referência o custo de Hora/Homem -2);

**o59** - Preço de custo relativo à mão-de-obra, do tipo 3;

$$o59 = i43 \times (i26 \div 60) [\text{€}]$$

**Equação 33 - Cálculo o59**

i26 - Definição de um dos três valores possíveis para o custo de mão-de-obra à hora;

i43 - Identificação do tempo de montagem do equipamento em minutos (tendo como referência o custo de Hora/Homem -3).

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o60** - Somatório dos preços dos equipamentos descritos no mapa de quantidade;

$$o60 = \sum(i36 \times i37) [\text{€}]$$

**Equação 34 - Cálculo o60**

i36 - Identificação da quantidade de elementos referidos;

i37 - Definição do valor unitário do equipamento;

**o61** - Somatório dos preços das montagens dos equipamentos descritos no mapa de quantidade;

$$Ao61 = \sum(i36 \times o27) [\text{€}]$$

**Equação 35 - Cálculo o61**

i36 - Identificação da quantidade de elementos referidos;

o27 - Preço da montagem, referente a um equipamento específico (unitário);

**o62** - Somatório dos preços dos acessórios, necessários para o bom funcionamento dos equipamentos referidos;

$$o62 = \sum(i36 \times i38) [\text{€}]$$

**Equação 36 - Cálculo o62**

i36 - Identificação da quantidade de elementos referidos;

i38 - Definição do valor unitário dos acessórios (referentes ao equipamento identificado na variável anterior);

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o63** - Valor destinado aos transportes;

$$o63 = \sum(i36 \times i39) [\text{€}]$$

**Equação 37 - Cálculo o63**

i36 - Identificação da quantidade de elementos referidos;

i39 - Definição do valor referente ao transporte do equipamento;

**o64** - Somatório dos vários preços de custo, referente aos equipamentos;

$$o64 = \sum(i36 \times o28) [\text{€}]$$

**Equação 38 - Cálculo o64**

i36 - Identificação da quantidade de elementos referidos;

o28 - Preço de custo, referente a um equipamento específico (unitário);

**o65** - Somatório das margens, referente ao país de envio;

$$o65 = \sum(i36 \times o29) [\text{€}]$$

**Equação 39 - Cálculo o65**

i36 - Identificação da quantidade de elementos referidos;

o29 - Margem no equipamento referente ao país de envio (unitária);

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o66** - Somatório dos preços de venda, referente aos equipamentos no país de envio;

$$o66 = \sum(i36 \times o30) [€]$$

**Equação 40 - Cálculo o66**

i36 - Identificação da quantidade de elementos referidos;

o30 - Preço de venda, referente a um equipamento específico no país de envio  
(unitário);

**o67** - Somatório alusivo ao custo dos materiais no país de destino;

$$o67 = \sum(i36 \times o31) [€]$$

**Equação 41 - Cálculo o67**

i36 - Identificação da quantidade de elementos referidos;

o31 - Custo do material físico, no país de destino (unitário);

**o68** - Somatório do custo da mão-de-obra no país de destino;

$$o68 = \sum(i36 \times o32) [€]$$

**Equação 42 - Cálculo o68**

i36 - Identificação da quantidade de elementos referidos;

o32 - Custo relacionado com a mão-de-obra, no país de destino (unitário);



## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o69** - Somatório do custo dos equipamentos no país de destino;

$$o69 = \sum(i36 \times o33) \text{ [€]}$$

**Equação 43 - Cálculo o69**

i36 - Identificação da quantidade de elementos referidos;

o33 - Custo referente ao equipamento, no país de destino (unitário);

**o70** - Somatório do custo relacionado com os equipamentos locais, no país de destino;

$$o70 = \sum(i36 \times i40) \text{ [€]}$$

**Equação 44 - Cálculo o70**

i36 - Identificação da quantidade de elementos referidos;

i40 - Identificação do valor monetário referente aos equipamento locais;

**o71** - Somatório dos custos derivados ao desalfandegamento;

$$o71 = \sum(i36 \times o34) \text{ [€]}$$

**Equação 45 - Cálculo o71**

i36 - Identificação da quantidade de elementos referidos;

o34 - Valor de custo destinado ao desalfandegamento (unitário);

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o72** - Somatório dos custos derivados aos direitos;

$$o72 = \sum(i36 \times o35) [€]$$

**Equação 46 - Cálculo o72**

i36 - Identificação da quantidade de elementos referidos;

o35 - Valor destinado aos direitos (unitário);

**o73** - Somatório dos custos derivados à retenção IRT;

$$o73 = \sum(i36 \times o36) [€]$$

**Equação 47 - Cálculo o73**

i36 - Identificação da quantidade de elementos referidos;

o36 - Valor destinado a retenção IRT (unitário);

**o74** - Somatório dos custos oriundos do imposto de selo;

$$o74 = \sum(i36 \times o37) [€]$$

**Equação 48 - Cálculo o74**

i36 - Identificação da quantidade de elementos referidos;

o37 - Valor destinado ao imposto selo (unitário);

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o75** - Somatório dos custos dos vários equipamentos referidos, referente ao país de destino;

$$o75 = \sum(i36 \times o38) [\text{€}]$$

**Equação 49 - Cálculo o75**

i36 - Identificação da quantidade de elementos referidos;

o38 - Custo físico total, referente a um equipamento específico no país de destino  
(unitário);

**o76** - Somatório dos custos, referentes à mão-de-obra dos vários equipamentos referidos, no país de destino;

$$o76 = \sum(i36 \times o39) [\text{€}]$$

**Equação 50 - Cálculo o76**

i36 - Identificação da quantidade de elementos referidos;

o39 - Custo alusivo a mão-de-obra total, referente a um equipamento específico no país de destino (unitário);

**o77** - Somatório dos preços de custo, no país de destino,

$$o77 = \sum(i36 \times o40) [\text{€}]$$

**Equação 51 - Cálculo o77**

i36 - Identificação da quantidade de elementos referidos;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

o40 - Custo total, referente a um equipamento específico no país de destino (unitário);

**o78** - Somatório das margens alusivas aos materiais no país de destino;

$$o78 = \sum(i36 \times o41) [\text{€}]$$

**Equação 52 - Cálculo o78**

i36 - Identificação da quantidade de elementos referidos;

o41 - Margem no equipamento físico referente ao país de destino (unitária);

**o79** - Somatório das margens alusivas à mão-de-obra no país de destino;

$$o79 = \sum(i36 \times o42) [\text{€}]$$

**Equação 53 - Cálculo o79**

i36 - Identificação da quantidade de elementos referidos;

o42 - Margem relacionada com a mão-de-obra, referente ao país de destino (unitário);

**o80** - Somatório das margens, no país de destino;

$$o80 = \sum(i36 \times o43) [\text{€}]$$

**Equação 54 - Cálculo o80**

i36 - Identificação da quantidade de elementos referidos;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

o43 - Margem total no equipamento referente ao país de destino (unitária);

**o81** - Somatório dos preços de venda dos materiais;

$$o81 = \sum(i36 \times o44) [\text{€}]$$

**Equação 55 - Cálculo o81**

i36 - Identificação da quantidade de elementos referidos;

o44 - Preço de venda dos materiais, no país de destino (unitário);

**o82** - Somatório dos preços de venda referente à mão-de-obra;

$$o82 = \sum(i36 \times o45) [\text{€}]$$

**Equação 56 - Cálculo o82**

i36 - Identificação da quantidade de elementos referidos;

o45 - Preço de venda referente a mão-de-obra, no país de destino (unitário);

**o83** - Somatório dos preços de venda dos trabalhos a realizar, referente ao país de destino;

$$o83 = \sum(i36 \times o46) [\text{€}]$$

**Equação 57 - Cálculo o83**

i36 - Identificação da quantidade de elementos referidos;

o46 - Preço de venda total do equipamento, no país de destino (unitário);

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o84** - Somatório dos preços de venda totais dos vários equipamentos, no país de destino;

$$o84 = \sum o47 \text{ [€]}$$

**Equação 58 - Cálculo o84**

o47 - Preço de venda total dos vários equipamentos, referenciados como iguais e quantificados na mesma célula, no país de destino (parcial);

**o85** - Somatório dos valores gastos relacionados com outros custos, como por exemplo, elementos exteriores ao fornecimento/realização dos trabalhos (unitário);

$$o85 = \sum o48 \text{ [€]}$$

**Equação 59 - Cálculo o85**

o48 - Valor gasto relacionado com outros custos, como elementos exteriores ao fornecimento/realização dos trabalhos (unitário);

**o86** - Somatório dos valores gastos relacionados com outros custos, como por exemplo, elementos exteriores ao fornecimento/realização dos trabalhos (parcial);

$$o86 = \sum o49 \text{ [€]}$$

**Equação 60 - Cálculo o86**

o49 - Valor gasto relacionado com outros custos, como elementos exteriores ao fornecimento/realização dos trabalhos (parcial);

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o87** - Somatório dos valores destinado às despesas com transporte (seguro);

$$o87 = \sum o50 \text{ [€]}$$

**Equação 61 - Cálculo o87**

o50 - Valor destinado as despesas com transporte (seguro);

**o88** - Somatório dos valores destinado às despesas com transporte (frete);

$$o88 = \sum o51 \text{ [€]}$$

**Equação 62 - Cálculo o88**

o51 - Valor destinado as despesas com transporte (frete);

**o89** - Somatório dos valores destinado às despesas com transporte interno, no país de destino;

$$o89 = \sum o52 \text{ [€]}$$

**Equação 63 - Cálculo o89**

o52 - Valor destinado as despesas com transporte interno, no país de destino;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o90** - Somatório dos preços de venda dos materiais e do transporte, no país de destino (unitários);

$$o90 = \sum(i36 \times o53) [\text{€}]$$

**Equação 64 - Cálculo o90**

i36 - Identificação da quantidade de elementos referidos;

o53 - Preço de venda dos materiais e do transporte, no país de destino (unitário);

**o91** - Somatório dos preços referentes à mão-de-obra e transporte, no país de destino (unitários);

$$o91 = \sum(i36 \times o54) [\text{€}]$$

**Equação 65 - Cálculo o91**

i36 - Identificação da quantidade de elementos referidos;

o54 - Preço referente a mão-de-obra e transporte, no país de destino (unitário);

**o92** - Somatório dos preços de venda totais e do transporte, no país de destino (unitários);

$$o92 = \sum(i36 \times o55) [\text{€}]$$

**Equação 66 - Cálculo o92**

i36 - Identificação da quantidade de elementos referidos;

o55 - Preço de venda total e do transporte, no país de destino (unitário);



## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o93** - Somatório dos preços de venda total e do transporte, no país de destino (parciais);

$$o93 = \sum o56 \text{ [€]}$$

**Equação 67 - Cálculo o93**

o56 - Preço de venda total e do transporte, no país de destino (parcial);

**o94** - Somatório temporal (em horas) relativo à mão-de-obra, do tipo 1;

$$o94 = [\sum(i36 \times i41)] \div 60 \text{ [horas]}$$

**Equação 68 - Cálculo o94**

i36 - Identificação da quantidade de elementos referidos;

i41 - Identificação do tempo de montagem do equipamento em minutos (tendo como referência o custo de Hora/Homem -1);

**o95** - Somatório dos preços de custo relativo à mão-de-obra, do tipo 1;

$$o95 = \sum(i36 \times o57) \text{ [€]}$$

**Equação 69 - Cálculo o95**

i36 - Identificação da quantidade de elementos referidos;

o57 - Preço de custo relativo à mão-de-obra, do tipo 1;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o96** - Somatório temporal (em horas) relativo à mão-de-obra, do tipo 2;

$$o96 = [\sum(i36 \times i42)] \div 60 \text{ [horas]}$$

**Equação 70 - Cálculo o96**

i36 - Identificação da quantidade de elementos referidos;

i42 - Identificação do tempo de montagem do equipamento em minutos (tendo como referência o custo de Hora/Homem -2);

**o97** - Somatório dos preços de custo relativo à mão-de-obra, do tipo 2;

$$o97 = \sum(i36 \times o58) \text{ [€]}$$

**Equação 71 - Cálculo o97**

i36 - Identificação da quantidade de elementos referidos;

o58 - Preço de custo relativo à mão-de-obra, do tipo 2;

**o98** - Somatório temporal (em horas) relativo à mão-de-obra, do tipo 3;

$$o98 = [\sum(i36 \times i43)] \div 60 \text{ [horas]}$$

**Equação 72 - Cálculo o98**

i36 - Identificação da quantidade de elementos referidos;

i43 - Identificação do tempo de montagem do equipamento em minutos (tendo como referência o custo de Hora/Homem -3).

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

**o99** - Somatório dos preços de custo relativo à mão-de-obra, do tipo 3;

$$o99 = \sum(i36 \times o59) [\text{€}]$$

**Equação 73 - Cálculo o99**

i36 - Identificação da quantidade de elementos referidos;

o59 - Preço de custo relativo à mão-de-obra, do tipo 3;

Nas tabelas a2.16.1 a a2.16.9 são apresentados os exemplos de preenchimento da base para identificação dos elementos referentes às divisórias em causa.

**Tabela a2.1 - Base para identificação dos elementos (exemplo)**

**Tabela a2.16.1 – 1ª Divisória**

Item	Descrição	Unid	Quantidades
	Building		
	Lighting		
1	d 25	m	50
2	d 32	m	100
3	d 40	m	10

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

Tabela a2.16.2 – 2ª Divisória

Valores Unitários em Euros (€)					
Equipamento	Montagem	Acessórios	Transportes	PC EMP.(Orig.)	Margem EMP.(Orig.)
0,35	500,00 €	0,22		0,57	0,09
0,53	7,50 €	0,33		0,86	0,13
0,67	7,50 €	0,44		1,11	0,17
77,20	25 825,00	48,40	0,00	125,60	18,84

Tabela a2.16.3 – 3ª Divisória

Valores Unitários em Euros (€)		Valores Unitários em Dolares Americanos (\$)		
PV EMP.(Orig.)		Custos EMP. Dist. (USD)		
		Materiais	Mão de Obra	Unitário
0,66		0,92	700,00	700,9177
0,99		1,38	10,50	11,8846
1,28		1,79	10,50	12,2871
144,44		202,22	36 155,00	36 357,22

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

Tabela a2.16.4 – 4ª Divisória

Valores Unitários em Dolares Americanos (\$)				
Custos EMP. Dist. (USD)				
Equip Locais	Desalfand.	Direitos	Retenção IRT	Imposto Selo
	0,10	0,00	24,53	7,01
	0,15	0,00	0,42	0,12
	0,19	0,00	0,43	0,12
0,00	21,23	0,00	1 272,50	363,57

Tabela a2.16.5 – 5ª Divisória

Valores Unitários em Dolares Americanos (\$)			
Custos EMP. Dist. (USD)			Lucro EMP. Dist.
PC MATERIAIS (Dist.)	PC MÃO OBRA (Dist.)	PC EMP.(Dist.)	Margem EMP.(Dist.) Mat.
1,06	731,50	732,56	0,26
1,59	10,97	12,56	0,40
2,06	10,97	13,03	0,51
232,55	37 781,98	38 014,52	58,14

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

Tabela a2.16.6 – 6ª Divisória

Valores Unitários em Dolares Americanos (\$)			
Lucro EMP. Dist.		PV EMP. Dist.	
Margem EMP.(Dist.) MO	Margem EMP.(Dist.)	Materiais	Mão de obra
182,88	183,14	1,32	914,38
2,74	3,14	1,99	13,72
2,74	3,26	2,57	13,72
9 445,49	9 503,63	290,69	47 227,47

Tabela a2.16.7 – 7ª Divisória

Valores Unitários em Dolares Americanos		Outros Custos		PREÇO VENDA EMP. Dist. C	
PV EMP. Dist.		Valor Unitário	Valor Final	DESPESAS COM TRANSPORTE	
Unitário	Parcial			SEGURO	FRETE
915,69	45 784,71	915,69	45 784,71	0,79	\$14 426,79
15,71	1 570,60	15,71	1 570,60	2,39	\$494,90
16,28	162,85	16,28	162,85	0,31	\$51,31
47 518,15	47 518,15		47 518,15	\$3,49	\$14 973,00

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

2.2.3 – Base para identificação dos elementos referentes ao mapa de quantidades  
(Material) (Output)

Tabela a2.16.8 – 8ª Divisória

PREÇO VENDA EMP. Dist. COM DESPESAS DE TRANSPORTE (\$)					
PV EMP. Dist.					Total Parcial
T. INTERNOS	Materiais	Mão de Obra	Unitário		
\$4 986,22	389,60	914,38	1 303,97		65 198,51
\$171,05	8,67	13,72	22,39		2 238,93
\$17,73	9,50	13,72	23,22		232,20
\$5 175,00	20 442,17	47 227,47	67 669,64		\$67 669,64

Tabela a2.16.9 – 9ª Divisória

Custo Hora/Homem -1		Custo Hora/Homem -2		Custo Hora/Homem -3	
Tempos	Custo MO Unitário	Tempos	Custo MO Unitário	Tempos	Custo MO Unitário
Minutos	Euros	Minutos	Euros	Minutos	Euros
1000	500,00 €	15	7,50 €	15	7,50 €
833,33	25 000,00	25,00	750,00	2,50	75,00

---

## APÊNDICE

# 2.2.4

---

### **Base para resumo da orçamentação**

*Anexo (Output) Base para resumo da orçamentação*



## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

### 2.2.4 - Base para resumo da orçamentação (Output)

A tabela a2.17 permite ao utilizador do simulador retirar conclusões da sua própria orçamentação. Para isso, a tabela apresenta as variáveis mais importantes de forma a possibilitar ao utilizador uma fácil leitura de todo o processo.

Nota para o facto de ser necessário guardar os valores do array (programação executada por meio de apontadores).

Tabela a2.17 - Resumo da orçamentação (variáveis)

RESUMO DA ORÇAMENTAÇÃO			
PV EMP.(Orig.) (Inc. Frete e Seguro)	o100		
PV CIF EMP.(Dist.)	o101		
PV FOB EMP.(Dist.)	o102		
Custo( Equip., acc. e transportes) EMP.(Orig.)	o103		o114
Custo Mont. EMP.(Dist.)	o104		o115
Margem EMP.(Orig.)	o105		o116
Margem EMP.(Dist.)	o106		o117
IRT	o107		
Imp. Selo	o108		
Direitos	o109		
FCL	o110		
Frete Interno	o111		
Desalf.	o112		
Seguros	o113		

As variáveis da tabela a2.17 têm o seguinte significado:

**o100** - Preço de venda total da empresa no país de origem;

$o100 = o66 + [o66 \times i15 \times (1 + (i10 \text{ ou } i11))] + (o20 \div i9)$  ; Dependendo da relação entre  $o66 \times i15$  e  $o13$  [€]

#### Equação 1 - Cálculo o100

i3 - Descrição do limite na qual a margem A se altera para a margem B;

i9 - Identificação do câmbio/relação entre as moedas referência de dois tipos distintos;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

### 2.2.4 - Base para resumo da orçamentação (Output)

i10 - Identificação da margem de venda em Portugal (margem definida para elementos com valor inferior ao referido em i3);

i11 - Identificação da margem de venda em Portugal (margem definida para elementos com valor superior ao referido em i3);

i15 - Definição do valor do seguro (em percentagem);

o20 - custo derivado ao frete marítimo;

o66 - Somatório dos preços de venda, referente aos equipamentos no país de envio;

**o101** - Preço de venda total CIF, da empresa no país de destino;

$$o101 = o22 \text{ [€]}$$

**Equação 2 - Cálculo o101**

o22 - Somatório dos elementos referenciados como bases contentores;

**o102** - Preço de venda total FOB, da empresa no país de destino;

$$o102 = o18 \text{ [€]}$$

**Equação 3 - Cálculo o102**

o18 - Somatório dos custos finais dos equipamentos no país de destino;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.4 - Base para resumo da orçamentação (Output)

**o103** - Custo dos vários equipamentos no país de origem;

$$o103 = o64 \text{ [€]}$$

**Equação 4 - Cálculo o103**

o64 - Somatório dos vários preços de custo, referente aos equipamentos;

**o104** - Custo de montagem (mão-de-obra) dos vários equipamentos no país de destino;

$$o104 = o61 \times i9 \times i14 \text{ [€]}$$

**Equação 5 - Cálculo o104**

i9 - Identificação do câmbio/relação entre as moedas referência de dois tipos distintos;

i14 - Definição por meio de percentagem da produtividade local;

o61 - Somatório dos preços das montagens dos equipamentos descritos no mapa de quantidade;

**A105** - Margem da empresa no país de origem;

$$A105 = o65 \text{ [€]}$$

**Equação 6 - Cálculo o105**

o65 - Somatório das margens, referente ao país de envio;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.4 - Base para resumo da orçamentação (Output)

**o106** - Margem da empresa no país de destino;

$$o106 = o80 \text{ [€]}$$

**Equação 7 - Cálculo o106**

o80 - Somatório das margens, no país de destino;

**o107** - Valor destinado à retenção IRT;

$$o107 = o73 \text{ [€]}$$

**Equação 8 - Cálculo o107**

o73 - Somatório dos custos derivados à retenção IRT;

**o108** - Valor destinado a retenção do imposto de selo;

$$o108 = o74 \text{ [€]}$$

**Equação 9 - Cálculo o108**

o74 - Somatório dos custos oriundos do imposto de selo;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.4 - Base para resumo da orçamentação (Output)

**o109** - Valor destinado aos diretos;

$$o109 = o72 \text{ [€]}$$

**Equação 10 - Cálculo o109**

o72 - Somatório dos custos derivados aos diretos;

**o110** - Preço dos contentores, referente ao país de destino;

$$o110 = (i29 \times i17 + i30 \times i16) \times i9 \text{ [€]}$$

**Equação 11 - Cálculo o110**

i9 - Identificação do câmbio/relação entre as moedas referência de dois tipos distintos;

i16 - Representação do valor de custo de um contentor de 40" (40 pés);

i17 - Representação do valor de custo de um contentor de 20" (20 pés);

i29 - Representação do número de um contentor de 20" utilizados;

i30 - Representação do número de um contentor de 40" utilizados.

**o111** - Valor destinado ao transporte interno (contentores) no país de destino;

$$o111 = (i29 + i30) \times i18 \text{ [€]}$$

**Equação 12 - Cálculo o111**

i18 - Identificação do custo com transporte de cada contentor no país de destino;

i29 - Representação do número de um contentor de 20" utilizados;

i30 - Representação do número de um contentor de 40" utilizados.

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.4 - Base para resumo da orçamentação (Output)

**o112** - Valor destinado ao desalfandegamento;

$$o112 = o71 \text{ [€]}$$

**Equação 13 - Cálculo o112**

o71 - Somatório dos custos derivados ao desalfandegamento;

**o113** - Valor destinado ao seguro transposto no país de destino;

$$o113 = o19 \text{ [€]}$$

**Equação 14 - Cálculo o113**

o19 - Custo derivado ao seguro dos contentores;

**o114** - Custo dos vários equipamentos no país de origem (em percentagem);

$$o114 = o103 \div (o103 + o104 \div i9) \text{ [%]}$$

**Equação 15 - Cálculo o114**

i9 - Identificação do câmbio/relação entre as moedas referência de dois tipos distintos;

o103 - Custo dos vários equipamentos no país de origem;

o104 - Custo de montagem (mão-de-obra) dos vários equipamentos no país de destino;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

### 2.2.4 - Base para resumo da orçamentação (Output)

**o115** - Custo de montagem (mão-de-obra) dos vários equipamentos no país de destino (em percentagem);

$$o115 = o104 \div (o104 + o103 \times i9) [\%]$$

#### **Equação 16 - Cálculo o115**

i9 - Identificação do câmbio/relação entre as moedas referência de dois tipos distintos;

o103 - Custo dos vários equipamentos no país de origem;

o104 - Custo de montagem (mão-de-obra) dos vários equipamentos no país de destino;

**A116** - Margem da empresa no país de origem (em percentagem);

$$o116 = o105 \div o103 [\%]$$

#### **Equação 17 - Cálculo o116**

o103 - Custo dos vários equipamentos no país de origem;

o105 - Margem da empresa no país de origem;

**o117** - Margem da empresa no país de destino (em percentagem);

$$o117 = o106 \div o77 [\%]$$

#### **Equação 18 - Cálculo o117**

o106 - Margem da empresa no país de destino;

o77 - Somatório dos preços de custo, no país de destino,

## Apêndice nº2 - Formulas de cálculo

Posição de variáveis no cálculo (Bases)

### 2.2.4 - Base para resumo da orçamentação (Output)

apresenta-se um exemplo de preenchimento da tabela resumo da orçamentação:

**Tabela a2.18 - Resumo da orçamentação (exemplo)**

<b>RESUMO DA ORÇAMENTAÇÃO</b>			
PV EMP.(Orig.) (Inc. Frete e Seguro)	<b>10 841,93 €</b>		
PV CIF EMP.(Dist.)	\$ 67 669,64		
PV FOB EMP.(Dist.)	\$ 47 518,15		
Custo( Equip., acc, e transportes) EMP.(Orig.)	<b>125,60 €</b>		<b>0,48%</b>
Custo Mont. EMP.(Dist.)	\$ 36 155,00		<b>99,52%</b>
Margem EMP.(Orig.)	<b>18,84 €</b>		<b>15,00%</b>
Margem EMP.(Dist.)	\$ 9 503,63		<b>25,00%</b>
IRT	\$ 1 272,50		
Imp. Selo	\$ 363,57		
Direitos	\$ -		
FCL	\$ 13 020,00		
Frete Interno	\$ 5 175,00		
Desalf.	\$ 21,23		
Seguros	\$ 3,49		

Nota: A sigla CIF significa que o frete e o seguro são pagos pelo fornecedor que é o responsável pela entrega até o local de destino. No caso da FOB o cliente é que paga o frete e o seguro da mercadoria em questão.





---

## APÊNDICE

# 2.2.5

---

### **Restantes elementos de cálculo**

*Anexo (Output) Restantes elementos de cálculo*

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

### 2.2.5 - Restantes elementos de cálculo (Output)

Devido aos vários tipos de exigências/normas aplicadas pelos clientes e de forma a responder a todos estes, este simulador terá vários tipos de documentos (pró-formas) disponibilizados para entrega ao cliente.

Nos pró-formas disponíveis existem as seguintes variáveis de referência:

**o118** - Valor total referente ao frete marítimo;

$$o118 = o20 + o21 \text{ [€]}$$

**Equação 1 - Cálculo o118**

o20 - custo derivado ao frete marítimo;

o21 - Custo relacionado com o transporte interno no país de destino;

**o119** - Valor final dos trabalhos referidos no mapa de quantidades, no país de destino;

$$o119 = o118 + o19 + o86 \text{ [€]}$$

**Equação 2 - Cálculo o119**

o19 - Custo derivado ao seguro dos contentores;

o86 - Somatório dos valores gastos relacionados com outros custos, como por exemplo, elementos exteriores ao fornecimento/realização dos trabalhos (parcial);

o118 - Valor total referente ao frete marítimo;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

### 2.2.5 - Restantes elementos de cálculo (Output)

**o120** - Preço parcial dos equipamentos (sem mão-de-obra), projetados no país de origem;

$$o120 = i36 \times o30 \text{ [€]}$$

**Equação 3 - Cálculo o120**

o30 - Preço de venda, referente a um equipamento específico no país de envio  
(unitário);

i36 - Identificação da quantidade de elementos referidos;

**o121** - Valor do seguro, projetado no país de origem;

$$o121 = o19 \div i9 \text{ [€]}$$

**Equação 4 - Cálculo o121**

i9 - Identificação do câmbio/relação entre as moedas referência de dois tipos distintos;

o19 - Custo derivado ao seguro dos contentores;

**o122** - Valor do frete, projetado no país de origem;

$$o122 = o20 \div i9 \text{ [€]}$$

**Equação 5 - Cálculo o122**

i9 - Identificação do câmbio/relação entre as moedas referência de dois tipos distintos;

o20 - custo derivado ao frete marítimo;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

2.2.5 - Restantes elementos de cálculo (Output)

**o123** - Valor total, referente ao pró-forma número 5;

$$o123 = o122 + o121 + o120 \text{ [€]}$$

**Equação 6 - Cálculo o123**

o120 - Preço parcial dos equipamentos (sem mão-de-obra), projetados no país de origem;

o121 - Valor do seguro, projetado no país de origem;

o122 - Valor do frete, projetado no país de origem;

**o124** - Preço unitário da montagem, referente ao país de origem;

$$o124 = o27 \times i14 \text{ [€]}$$

**Equação 7 - Cálculo o124**

i14 - Definição por meio de percentagem da produtividade local;

o27 - Preço da montagem, referente a um equipamento específico (unitário);

**o125** - Preço parcial montagem, referente ao país de origem;

$$o125 = o124 \times i36 \text{ [€]}$$

**Equação 8 - Cálculo o125**

i36 - Identificação da quantidade de elementos referidos;

o124 - Preço unitário da montagem, referente ao país de origem;

## **Apêndice nº2 - Formulas de cálculo**

Posição de variáveis no cálculo (Bases)

### 2.2.5 - Restantes elementos de cálculo (Output)

**o126** - Somatório dos preços parcial da montagem, referente ao país de origem;

$$o126 = \sum o125 \text{ [€]}$$

**Equação 9 - Cálculo o126**

o125 - Peça parcial montagem, referente ao país de origem;

**o127** - Valor total, referente ao pró-forma número 7;

$$o127 = o66 + o126 \text{ [€]}$$

**Equação 10 - Cálculo o127**

o66 - Somatório dos preços de venda, referente aos equipamentos no país de envio;

o126 - Somatório dos preços parcial da montagem, referente ao país de origem;



---

**APÊNDICE**

**3**

---

**Programa linguagem**

*Informação sobre Programa linguagem,*





---

## APÊNDICE

# 3.1

---

### Back-end

*Anexo (Programa linguagem) Back-end*



---

**APÊNDICE**

**3.1.1**

---

**Base para identificação da proposta (*Proposal*)**

*Anexo (Programa linguagem) Base para identificação da propost*

## Apêndice nº3 - Programa linguagem

### 3.1 – Back-end

#### 3.1.1 - Base para identificação da proposta (*Proposal*)

```
package com.ibs.mestrado.dto;

/**
 * Created by Eurico on 13/12/2017.
 *
 * ----- 3.2.1.1 - Bases para identificação da proposta -----
 */
public class Proposal {

    private long    proposalId;
    private ProposalType proposalType; /* i0 - national or
international budget */
    private String  name;              /* i1 - Work name */
    private String  localSite;        /* i2 - Work location */
    private float   marginLimit;      /* i3 - Limit in which the
margin A changes to the margin B */
    private Company company;          /* i4 - Company Reference
*/
    private Client  client;           /* i5 - Customer name */
    private Date    deliveryDate;     /* i6 - Current date */
    private String  clientResponsible; /* i7 - It might be
deleted */

    private String  revisionVersion; /* i8 - Number of proposal
revision */
    private Revision revision;

    public long getProposalId() {
        return proposalId;
    }

    public void setProposalId(long proposalId) {
        this.proposalId = proposalId;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getLocalSite() {
        return localSite;
    }

    public void setLocalSite(String localSite) {
        this.localSite = localSite;
    }

    public float getMarginLimit() {
        return marginLimit;
    }
}
```

## Apêndice nº3 - Programa linguagem

### 3.1 – Back-end

#### 3.1.1 - Base para identificação da proposta (*Proposal*)

```
}

public void setMarginLimit(float marginLimit) {
    this.marginLimit = marginLimit;
}

public Company getCompany() {
    return company;
}

public void setCompany(Company company) {
    this.company = company;
}

public Client getClient() {
    return client;
}

public void setClient(Client client) {
    this.client = client;
}

public Date getDeliveryDate() {
    return deliveryDate;
}

public void setDeliveryDate(Date deliveryDate) {
    this.deliveryDate = deliveryDate;
}

public String getClientResponsible() {
    return clientResponsible;
}

public void setClientResponsible(String clientResponsible) {
    this.clientResponsible = clientResponsible;
}

public String getRevisionVersion() {
    return revisionVersion;
}

public void setRevisionVersion(String revisionVersion) {
    this.revisionVersion = revisionVersion;
}

public Revision getRevision() {
    return revision;
}

public void setRevision(Revision revision) {
    this.revision = revision;
}

public ProposalType getProposalType() {
    return proposalType;
}
```

## Apêndice nº3 - Programa linguagem

### 3.1 – Back-end

#### 3.1.1 - Base para identificação da proposta (*Proposal*)

```
}  
  
public void setProposalType(ProposalType proposalType) {  
    this.proposalType = proposalType;  
}  
}
```

---

**APÊNDICE**

**3.1.2**

---

**Base de cálculo para orçamentação (*BudgetBase*)**

*Anexo (Programa linguagem) Base de cálculo para orçamentação*



## Apêndice nº3 - Programa linguagem

### 3.1 – Back-end

#### 3.1.2 - Base de cálculo para orçamentação (*BudgetBase*)

```
package com.ibs.mestrado.dto;

/**
 * Created by Eurico on 13/12/2017.
 *
 * ----- 3.2.1.2 - Bases de cálculo para orçamentação -----
 *
 */
public class BudgetBase {

    private long        budgetBaseId;

    private double      currencyExchange;        /* i9 -
exchange/Relationship between coins */
    private Currency    origin;
    private Currency    destination;

    private double      minimumMarginOrigin;     /* i10 - Company
margin in source A (Org.) */
    private double      maximumMarginOrigin;     /* i11 - Company
margin in source B (Org.) */
    private double      minimumMarginDestiny;     /* i12 - Company
margin in the target A (Dist.) */
    private double      maximumMarginDestiny;     /* i13 - Company
margin in the target B (Dist.) */
    private double      productivityFactor;       /* i14 - Local
productivity (percentage) */
    private double      insurancePercent;        /* i15 - Insurance
value (percentage) */
    private double      containerCost40;         /* i16 - Cost value of
a container 40" */
    private double      containerCost20;         /* i17 - Cost value of
a container 20" */
    private double      containerTransportDestiny; /* i18 - Transport
cost of 1 container in the destination country */
    private double      containerTransportOrigin; /* i19 - Transport
cost of containers in the source country */

    private double      clearancePercent;        /* i20 - Cost related
to customs clearance (percentage) */
    private double      rightCostPercent;        /* i21 - Right Cost
(percentage) */
    private double      rightCostIRTPercent;     /* i22 - Costs
relating to importation taxes */
    private double      sealPercent;             /* i23 - Cost related
to the stamp tax (percentage) */
    private double      assemblyPerHour1;        /* i24 - Cost of hand
labor time type 1 */
    private double      assemblyPerHour2;        /* i25 - Cost of hand
labor time type 2 */
    private double      assemblyPerHour3;        /* i26 - Cost of hand
labor time type 3 */
}
```

## Apêndice nº3 - Programa linguagem

### 3.1 – Back-end

#### 3.1.2 - Base de cálculo para orçamentação (*BudgetBase*)

```
private double otherCostsPercent;          /* i44 - Amount spent
related to other cost (percentage) (only on proposals with the
country of destination and origin is different */
private String salesterms;                 /* i45 - sales terms */

public String getSalesterms() {
    return salesterms;
}

public void setSalesterms(String salesterms) {
    this.salesterms = salesterms;
}

public long getBudgetBaseId() {
    return budgetBaseId;
}

public void setBudgetBaseId(long budgetBaseId) {
    this.budgetBaseId = budgetBaseId;
}

public double getCurrencyExchange() {
    return currencyExchange;
}

public void setCurrencyExchange(double currencyExchange) {
    this.currencyExchange = currencyExchange;
}

public Currency getOrigin() {
    return origin;
}

public void setOrigin(Currency origin) {
    this.origin = origin;
}

public Currency getDestination() {
    return destination;
}

public void setDestination(Currency destination) {
    this.destination = destination;
}

public double getMinimumMarginOrigin() {
    return minimumMarginOrigin;
}

public void setMinimumMarginOrigin(double minimumMarginOrigin)
{
    this.minimumMarginOrigin = minimumMarginOrigin;
}

public double getMaximumMarginOrigin() {
    return maximumMarginOrigin;
}
```

## Apêndice nº3 - Programa linguagem

### 3.1 – Back-end

#### 3.1.2 - Base de cálculo para orçamentação (*BudgetBase*)

```
    }

    public void setMaximumMarginOrigin(double maximumMarginOrigin)
    {
        this.maximumMarginOrigin = maximumMarginOrigin;
    }

    public double getMinimumMarginDestiny() {
        return minimumMarginDestiny;
    }

    public void setMinimumMarginDestiny(double
minimumMarginDestiny) {
        this.minimumMarginDestiny = minimumMarginDestiny;
    }

    public double getMaximumMarginDestiny() {
        return maximumMarginDestiny;
    }

    public void setMaximumMarginDestiny(double
maximumMarginDestiny) {
        this.maximumMarginDestiny = maximumMarginDestiny;
    }

    public double getProductivityFactor() {
        return productivityFactor;
    }

    public void setProductivityFactor(double productivityFactor) {
        this.productivityFactor = productivityFactor;
    }

    public double getInsurancePercent() {
        return insurancePercent;
    }

    public void setInsurancePercent(double insurancePercent) {
        this.insurancePercent = insurancePercent;
    }

    public double getContainerCost40() {
        return containerCost40;
    }

    public void setContainerCost40(double containerCost40) {
        this.containerCost40 = containerCost40;
    }

    public double getContainerCost20() {
        return containerCost20;
    }

    public void setContainerCost20(double containerCost20) {
        this.containerCost20 = containerCost20;
    }
}
```

## Apêndice nº3 - Programa linguagem

### 3.1 – Back-end

#### 3.1.2 - Base de cálculo para orçamentação (*BudgetBase*)

```
public double getContainerTransportDestiny() {
    return containerTransportDestiny;
}

public void setContainerTransportDestiny(double
containerTransportDestiny) {
    this.containerTransportDestiny = containerTransportDestiny;
}

public double getContainerTransportOrigin() {
    return containerTransportOrigin;
}

public void setContainerTransportOrigin(double
containerTransportOrigin) {
    this.containerTransportOrigin = containerTransportOrigin;
}

public double getClearancePercent() {
    return clearancePercent;
}

public void setClearancePercent(double clearancePercent) {
    this.clearancePercent = clearancePercent;
}

public double getRightCostPercent() {
    return rightCostPercent;
}

public void setRightCostPercent(double rightCostPercent) {
    this.rightCostPercent = rightCostPercent;
}

public double getRightCostIRTPercent() {
    return rightCostIRTPercent;
}

public void setRightCostIRTPercent(double rightCostIRTPercent)
{
    this.rightCostIRTPercent = rightCostIRTPercent;
}

public double getSealPercent() {
    return sealPercent;
}

public void setSealPercent(double sealPercent) {
    this.sealPercent = sealPercent;
}

public double getAssemblyPerHour1() {
    return assemblyPerHour1;
}
}
```

## Apêndice nº3 - Programa linguagem

### 3.1 – Back-end

#### 3.1.2 - Base de cálculo para orçamentação (*BudgetBase*)

```
public void setAssemblyPerHour1(double assemblyPerHour1) {
    this.assemblyPerHour1 = assemblyPerHour1;
}

public double getAssemblyPerHour2() {
    return assemblyPerHour2;
}

public void setAssemblyPerHour2(double assemblyPerHour2) {
    this.assemblyPerHour2 = assemblyPerHour2;
}

public double getAssemblyPerHour3() {
    return assemblyPerHour3;
}

public void setAssemblyPerHour3(double assemblyPerHour3) {
    this.assemblyPerHour3 = assemblyPerHour3;
}

public double getOtherCostsPercent() {
    return otherCostsPercent;
}

public void setOtherCostsPercent(double otherCostsPercent) {
    this.otherCostsPercent = otherCostsPercent;
}
}
```

---

APÊNDICE

# 3.1.3

---

**Base Auxiliares (*AuxiliarBase*)**

*Anexo (Programa linguagem) Bases Auxiliares*

## Apêndice nº3 - Programa linguagem

### 3.1 – Back-end

#### 3.1.3 - Bases Auxiliares (*AuxiliarBase*)

```
package com.ibs.mestrado.dto;

/**
 * Created by Eurico on 11/01/2018.
 */
public class AuxiliarBase {

    private long auxiliarBaseId;
    private int numMonths;      /*i27 - Identifica o número de
meses atribuído (pelo cliente) para execução da obra;*/
    private int businessDays;  /*i28 - Define o número de dias
úteis/trabalháveis por mês;*/

    public long getAuxiliarBaseId() {
        return auxiliarBaseId;
    }

    public void setAuxiliarBaseId(long auxiliarBaseId) {
        this.auxiliarBaseId = auxiliarBaseId;
    }

    public int getNumMonths() {
        return numMonths;
    }

    public void setNumMonths(int numMonths) {
        this.numMonths = numMonths;
    }

    public int getBusinessDays() {
        return businessDays;
    }

    public void setBusinessDays(int businessDays) {
        this.businessDays = businessDays;
    }
}
```

---

APÊNDICE

# 3.1.4

---

**Base Contentores (*Containers*)**

*Back-end Base Contentores*



## Apêndice nº3 - Programa linguagem

### 3.1 – Back-end

#### 3.1.4 - Bases Contentores (*Containers*)

```
package com.ibs.mestrado.dto;

/**
 * Created by Eurico on 11/01/2018.
 */
public class Containers {

    private long    containersId;

    private int numcontainers20; /*i29 - Representa o número de
um contentor de 20" utilizados;*/
    private int numcontainers40; /*i30 - Representa o número de
um contentor de 40" utilizados;*/

    public long getContainersId() {
        return containersId;
    }

    public void setContainersId(long containersId) {
        this.containersId = containersId;
    }

    public int getNumcontainers20() {
        return numcontainers20;
    }

    public void setNumcontainers20(int numcontainers20) {
        this.numcontainers20 = numcontainers20;
    }

    public int getNumcontainers40() {
        return numcontainers40;
    }

    public void setNumcontainers40(int numcontainers40) {
        this.numcontainers40 = numcontainers40;
    }
}
```

---

**APÊNDICE**

**3.1.5**

---

**Base para identificação dos elementos referentes ao mapa  
de quantidades (*Material*)**

*Back-end Base para identificação dos elementos referentes ao mapa de quantidades*

## Apêndice nº3 - Programa linguagem

### 3.1 – Back-end

#### 3.15 - Bases para identificação dos elementos referentes ao mapa de quantidades (Material)

```
package com.ibs.mestrado.dto;

/**
 * Created by Eurico on 13/12/2017.
 *
 * ----- 3.2.1.4 - Bases para identificação dos elementos
referentes ao mapa de quantidades -----
 *
 */
public class Material {

    private long        materialId;

    private ElemType    elemType;          /* i31 - Tipo de
elemento*/
    private ElemSubtype elemSubtype;      /* i32 - Subtipo de
elemento*/
    private long        numericCode;      /* i33 - Contains the
material numeric code */
    private String      description;      /* i34 - Contains the
material description */
    private UnitType    unitType;        /* i35 - Contains the
material unit type */
    private double      quantity;        /* i36 - Contains the
material quantity */
    private double      price;           /* i37 - Unit value
(material) */
    private double      priceAccessories; /* i38 - Unit value
of accessories */
    private double      priceTransport;   /* i39 - Value for
the transportation (equipment) */
    private double      priceLocalDestination; /* i40 - Transport
price (international only) */
    private int         timeAssembly1;    /* i41 - Assembly
time in minutes equipment type 1 */
    private int         timeAssembly2;    /* i42 - Assembly
time in minutes equipment type 2 */
    private int         timeAssembly3;    /* i43 - Assembly
time in minutes equipment type 3 */

    public long getMaterialId() {
        return materialId;
    }

    public ElemType getElemType() {
        return elemType;
    }

    public void setElemType(ElemType elemType) {
        this.elemType = elemType;
    }
}
```

## Apêndice nº3 - Programa linguagem

### 3.1 – Back-end

#### 3.15 - Bases para identificação dos elementos referentes ao mapa de quantidades

*(Material)*

```
public ElemSubtype getElemSubtype() {
    return elemSubtype;
}

public void setElemSubtype(ElemSubtype elemSubtype) {
    this.elemSubtype = elemSubtype;
}

public void setMaterialId(long materialId) {
    this.materialId = materialId;
}

public long getNumericCode() {
    return numericCode;
}

public void setNumericCode(long numericCode) {
    this.numericCode = numericCode;
}

public String getDescription() {
    return description;
}

public void setDescription(String description) {
    this.description = description;
}

public UnitType getUnitType() {
    return unitType;
}

public void setUnitType(UnitType unitType) {
    this.unitType = unitType;
}

public double getQuantity() {
    return quantity;
}

public void setQuantity(double quantity) {
    this.quantity = quantity;
}

public double getPrice() {
    return price;
}

public void setPrice(double price) {
    this.price = price;
}

public double getPriceAccessorisese() {
```

## Apêndice nº3 - Programa linguagem

### 3.1 – Back-end

#### 3.15 - Bases para identificação dos elementos referentes ao mapa de quantidades

*(Material)*

```
        return priceAccessorise;
    }

    public void setPriceAccessorise(double priceAccessorise) {
        this.priceAccessorise = priceAccessorise;
    }

    public double getPriceTransport() {
        return priceTransport;
    }

    public void setPriceTransport(double priceTransport) {
        this.priceTransport = priceTransport;
    }

    public double getPriceLocalDestination() {
        return priceLocalDestination;
    }

    public void setPriceLocalDestination(double
priceLocalDestination) {
        this.priceLocalDestination = priceLocalDestination;
    }

    public int getTimeAssembly1() {
        return timeAssembly1;
    }

    public void setTimeAssembly1(int timeAssembly1) {
        this.timeAssembly1 = timeAssembly1;
    }

    public int getTimeAssembly2() {
        return timeAssembly2;
    }

    public void setTimeAssembly2(int timeAssembly2) {
        this.timeAssembly2 = timeAssembly2;
    }

    public int getTimeAssembly3() {
        return timeAssembly3;
    }

    public void setTimeAssembly3(int timeAssembly3) {
        this.timeAssembly3 = timeAssembly3;
    }
}
```

---

## APÊNDICE

# 3.2

---

### **Criação proposta (Ex)**

*Front - Criação proposta (Exemplo)*

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
<html lang="en" ng-app="app" ng-controller="AppController"
class="ng-scope"><head><style type="text/css">.ng-
animate.item:not(.left):not(.right){-webkit-transition:0s ease-in-
out left;transition:0s ease-in-out left}</style><style
type="text/css">@charset "UTF-8";[ng\:cloak],[ng-cloak],[data-ng-
cloak],[x-ng-cloak],.ng-cloak,.x-ng-cloak,.ng-hide:not(.ng-hide-
animate){display:none !important;}ng\:form{display:block;}ng-
animate-shim{visibility:hidden;}ng-
anchor{position:absolute;}</style>
  <base href="/">
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Business Manager</title>
  <meta name="description" content="">
  <meta name="viewport" content="width=device-width, initial-
scale=1 user-scalable=no">
  <link rel="stylesheet" ng-href="/css/main.css"
href="/css/main.css">
  <style>
    [ng\:cloak],[ng-cloak],.ng-cloak { display: none
!important; }
  </style>
</head>
<body>
  <nav class="navbar navbar-default navbar-fixed-top ng-
isolate-scope">
  <div class="container-fluid">
    <!-- Brand and toggle get grouped for better mobile display
-->
    <div class="navbar-header">
      <button type="button" class="navbar-toggle collapsed"
data-toggle="collapse" data-target="#bs-example-navbar-collapse-1"
aria-expanded="false">
        <span class="sr-only">Toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
</button>
  <a class="navbar-brand" href="#">Business Manager</a>
</div>

<!-- Collect the nav links, forms, and other content for
toggling -->
  <div class="collapse navbar-collapse" id="bs-example-
navbar-collapse-1">
    <ul class="nav navbar-nav navbar-left">
      <!-- ngRepeat: item in navMenu --><li ng-
repeat="item in navMenu" ng-class="{dropdown: item.subitem !==
undefined}" ng-controller="NavMenuController" nav-
menu="navBar.left" class="ng-scope dropdown">
        <!-- ng-class="{dropdown: item.subitem !== undefined, open:
item.open}" -->
        <!--class="dropdown-toggle" data-toggle="dropdown"
role="button" aria-haspopup="true" aria-expanded="{{item.open}}"-->
        <a href="#" data-toggle="dropdown" class="ng-binding" aria-
expanded="false">
          Proposals <span class="caret"></span>
        </a>
        <!-- ngIf: item.subitem !== undefined --><ul class="dropdown-
menu ng-scope" ng-if="item.subitem !== undefined" ng-
init="subNavMenu = item.subitem">
          <li>
            <div class="menu-area-title">
              <span class="menu-area-name ng-binding">Proposal
Menu</span>
              <span class="menu-area-description ng-binding">Use
this options to search for proposals or create a new one. It is
possible to search by client, description or proposal code.</span>
            </div>
          </li>
          <!-- ngIf: subNavMenu.search !== undefined --><li ng-
if="subNavMenu.search !== undefined" ng-
init="search=subNavMenu.search" class="ng-scope">
            <div class="search-field">
```



## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
<div class="separator ng-binding">Search for
Proposal</div>
  <form id="proposals-search" ng-submit="close();
submitSeach(search.actionPath);" novalidate="novalidate" class="ng-
pristine ng-valid">
    <input type="text" name="searchFor"
placeholder="Search for proposals..." autocomplete="off">
    <div class="search-controls">
      <button type="submit" id="submit"
class="search-button glyphicon glyphicon-search" aria-
hidden="true"></button>
      <button type="button" class="search-advance
glyphicon glyphicon-option-horizontal" aria-hidden="true"></button>
    </div>
  </form>
</div>
</li><!-- end ngIf: subNavMenu.search !== undefined -->
<!-- ngRepeat: subitem in subNavMenu.items --><li ng-
repeat="subitem in subNavMenu.items" class="ng-scope">
  <div class="menu-item">
    <a href="/proposals/new" id="proposals-create">
      <span class="ng-binding">New Proposal</span>
    </a>
  </div>
</li><!-- end ngRepeat: subitem in subNavMenu.items -->
</ul><!-- end ngIf: item.subitem !== undefined -->
</li><!-- end ngRepeat: item in navMenu --><li ng-repeat="item in
navMenu" ng-class="{dropdown: item.subitem !== undefined}" ng-
controller="NavMenuController" nav-menu="navBar.left" class="ng-
scope dropdown">
  <!-- ng-class="{dropdown: item.subitem !== undefined, open:
item.open}" -->
  <!--class="dropdown-toggle" data-toggle="dropdown"
role="button" aria-haspopup="true" aria-expanded="{{item.open}}"-->
  <a href="#" data-toggle="dropdown" class="ng-binding">
    Materials <span class="caret"></span>
  </a>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
<!-- ngIf: item.subitem !== undefined --><ul class="dropdown-  
menu ng-scope" ng-if="item.subitem !== undefined" ng-  
init="subNavMenu = item.subitem">  
  <li>  
    <div class="menu-area-title">  
      <span class="menu-area-name ng-binding">Material  
Menu</span>  
      <span class="menu-area-description ng-binding">Use  
this options to search for a material, create materials. Manage  
your material units and details. It is possible to search for  
material code or description</span>  
    </div>  
  </li>  
  <!-- ngIf: subNavMenu.search !== undefined --><li ng-  
if="subNavMenu.search !== undefined" ng-  
init="search=subNavMenu.search" class="ng-scope">  
    <div class="search-field">  
      <div class="separator ng-binding">Search for  
Material</div>  
      <form id="materials-search" ng-submit="close();  
submitSeach(search.actionPath);" novalidate="novalidate" class="ng-  
pristine ng-valid">  
        <input type="text" name="searchFor"  
placeholder="Search for material..." autocomplete="off">  
        <div class="search-controls">  
          <button type="submit" id="submit"  
class="search-button glyphicon glyphicon-search" aria-  
hidden="true"></button>  
          <button type="button" class="search-advance  
glyphicon glyphicon-option-horizontal" aria-hidden="true"></button>  
        </div>  
      </form>  
    </div>  
  </li><!-- end ngIf: subNavMenu.search !== undefined -->  
  <!-- ngRepeat: subitem in subNavMenu.items --><li ng-  
repeat="subitem in subNavMenu.items" class="ng-scope">  
    <div class="menu-item">
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
<a href="/materials/new" id="material-create">
  <span class="ng-binding">New Material</span>
</a>
</div>
</li><!-- end ngRepeat: subitem in subNavMenu.items -->
</ul><!-- end ngIf: item.subitem !== undefined -->
</li><!-- end ngRepeat: item in navMenu --><li ng-repeat="item in
navMenu" ng-class="{dropdown: item.subitem !== undefined}" ng-
controller="NavMenuController" nav-menu="navBar.left" class="ng-
scope dropdown">
  <!-- ng-class="{dropdown: item.subitem !== undefined, open:
item.open}" -->
  <!--class="dropdown-toggle" data-toggle="dropdown"
role="button" aria-haspopup="true" aria-expanded="{{item.open}}"-->
  <a href="#" data-toggle="dropdown" class="ng-binding">
    Management <span class="caret"></span>
  </a>
  <!-- ngIf: item.subitem !== undefined --><ul class="dropdown-
menu ng-scope" ng-if="item.subitem !== undefined" ng-
init="subNavMenu = item.subitem">
    <li>
      <div class="menu-area-title">
        <span class="menu-area-name ng-binding">Management
Menu</span>
        <span class="menu-area-description ng-
binding">Manage your clients in this menu. Search for clients or
create new clients. Search for clients using the client name or
description</span>
      </div>
    </li>
    <!-- ngIf: subNavMenu.search !== undefined --><li ng-
if="subNavMenu.search !== undefined" ng-
init="search=subNavMenu.search" class="ng-scope">
      <div class="search-field">
        <div class="separator ng-binding">Search for
Clients</div>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
<form id="client-search" ng-submit="close();
submitSeach(search.actionPath);" novalidate="novalidate" class="ng-
pristine ng-valid">
    <input type="text" name="searchFor"
placeholder="Search for clients..." autocomplete="off">
    <div class="search-controls">
        <button type="submit" id="submit"
class="search-button glyphicon glyphicon-search" aria-
hidden="true"></button>
        <button type="button" class="search-advance
glyphicon glyphicon-option-horizontal" aria-hidden="true"></button>
    </div>
</form>
</div>
</li><!-- end ngIf: subNavMenu.search !== undefined -->
<!-- ngRepeat: subitem in subNavMenu.items --><li ng-
repeat="subitem in subNavMenu.items" class="ng-scope">
    <div class="menu-item">
        <a href="/units" id="units-config">
            <span class="ng-binding">Units</span>
        </a>
    </div>
</li><!-- end ngRepeat: subitem in subNavMenu.items --><li
ng-repeat="subitem in subNavMenu.items" class="ng-scope">
    <div class="menu-item">
        <a href="/currencies" id="currency-config">
            <span class="ng-binding">Currencies</span>
        </a>
    </div>
</li><!-- end ngRepeat: subitem in subNavMenu.items -->
</ul><!-- end ngIf: item.subitem !== undefined -->
</li><!-- end ngRepeat: item in navMenu -->
</ul>
<ul class="nav navbar-nav navbar-right">
    <!-- ngRepeat: item in navMenu --><li ng-
repeat="item in navMenu" ng-class="{dropdown: item.subitem !==
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
undefined}" ng-controller="NavMenuController" nav-
menu="navBar.right" class="ng-scope dropdown">
  <!-- ng-class="{dropdown: item.subitem !== undefined, open:
item.open}" -->
  <!--class="dropdown-toggle" data-toggle="dropdown"
role="button" aria-haspopup="true" aria-expanded="{{item.open}}"-->
  <a href="#" data-toggle="dropdown" class="ng-binding" aria-
expanded="false">
    User <span class="caret"></span>
  </a>
  <!-- ngIf: item.subitem !== undefined --><ul class="dropdown-
menu ng-scope" ng-if="item.subitem !== undefined" ng-
init="subNavMenu = item.subitem">
    <li>
      <div class="menu-area-title">
        <span class="menu-area-name ng-
binding">Configuration Menu</span>
        <span class="menu-area-description ng-binding">Use
this menu to manage material units, currencies details.</span>
      </div>
    </li>
    <!-- ngIf: subNavMenu.search !== undefined -->
    <!-- ngRepeat: subitem in subNavMenu.items --><li ng-
repeat="subitem in subNavMenu.items" class="ng-scope">
      <div class="menu-item">
        <a href="/users/logoff" id="user-logoff">
          <span class="ng-binding">User Logoff</span>
        </a>
      </div>
    </li><!-- end ngRepeat: subitem in subNavMenu.items --><li
ng-repeat="subitem in subNavMenu.items" class="ng-scope">
      <div class="menu-item">
        <a href="/users" id="user-details">
          <span class="ng-binding">User details</span>
        </a>
      </div>
    </li><!-- end ngRepeat: subitem in subNavMenu.items -->
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
</ul><!-- end ngIf: item.subitem !== undefined -->
</li><!-- end ngRepeat: item in navMenu -->
    </ul>
  </div>
</div>
</nav>
  <!-- ngView: undefined --><ng-view class="ng-scope"><div
class="container-fluid container-wrapper ng-scope">
  <div class="container-header">
    <h1>Create a new Proposal</h1>
  </div>
  <div class="container-body row">
    <div class="container-description">
      You are creating a new proposal. In this section you
need to insert all proposal information
      by filling all the input values.
    </div>
    <form ng-submit="createProposal();" class="ng-pristine ng-
valid">
      <div class="panel-body summary-container">
        <div class="summary-title">
          <b>Information</b>
        </div>
        <div>
          <div class="row">
            <div class="col-md-8">
              <div class="form-group">
                <label class="control-label"
for="name">Name</label>
                <input class="form-control ng-
pristine ng-untouched ng-valid" id="name" ng-
model="proposal.info.proposalName">
              </div>
            </div>
          </div>
          <div class="row">
            <div class="col-md-4">
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
<div class="form-group">
  <label class="control-label"
for="proposalType">Type</label>
  <select id="proposalType"
class="form-control ng-pristine ng-untouched ng-valid" ng-
options="type.id as type.description for type in
resources.proposalTypes" ng-
model="proposal.info.proposalType"><option value="?"
selected="selected"></option><option value="string:NAT"
label="National">National</option><option value="string:INT"
label="International">International</option></select>
</div>
</div>
<div class="col-md-4">
  <div class="form-group">
    <label class="control-label"
for="marginLimit">Margin Limit</label>
    <input id="marginLimit"
class="form-control ng-pristine ng-untouched ng-valid" ng-
model="proposal.info.marginLimit">
  </div>
</div>
<div class="col-md-4">
  <div class="form-group">
    <label class="control-label"
for="revisionVersion">Revision</label>
    <input class="form-control ng-
pristine ng-untouched ng-valid" id="revisionVersion"
readonly="true" ng-model="proposal.info.revisionVersion">
  </div>
</div>
</div>
<div class="row">
  <div class="col-md-4">
    <div class="form-group">
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
        <input type="hidden" ng-
model="proposal.client.cliendId" class="ng-pristine ng-untouched
ng-valid">
        <label class="control-label"
for="clientName">Client Name</label>
        <select id="clientName"
class="form-control ng-pristine ng-untouched ng-valid" ng-
options="type.clientId as type.name for type in clients" ng-
model="proposal.info.clientId"><option value="?"
selected="selected"></option><option value="string:1"
label="I.S.E.L">I.S.E.L</option><option value="string:2"
label="I.S.E.L 2">I.S.E.L 2</option><option value="string:3"
label="I.S.E.L 3">I.S.E.L 3</option></select>
        </div>
    </div>
    <div class="col-md-4">
        <div class="form-group">
            <label class="control-label"
for="localSite">Location</label>
            <input class="form-control ng-
pristine ng-untouched ng-valid" id="localSite" ng-
model="proposal.info.localSite">
            </div>
        </div>
        <div class="col-md-4">
            <div class="form-group">
                <label class="control-label"
for="deliveryDate">Delivery Date</label>
                <input class="form-control ng-
pristine ng-untouched ng-valid" id="deliveryDate" ng-
model="proposal.info.deliveryDate">
                </div>
            </div>
        </div>
    </div>
</div>
</div>
```



## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
<div class="panel-body summary-container">
  <div class="summary-title">
    <b>Budget Base</b>
  </div>
  <div>
    <div class="row">
      <div class="col-md-3">
        <div class="form-group">
          <input type="hidden" ng-
model="proposal.budgetBase.budgetBaseId" class="ng-pristine ng-
untouched ng-valid">
          <label class="control-label"
for="budgetBase.origin">Currency Origin</label>
          <select class="form-control ng-
pristine ng-untouched ng-valid" id="budgetBase.origin" ng-
model="proposal.budgetBase.currencyOrigin" ng-
change="updateCurrency()" ng-options="item.id as item.description
for item in resources.currencies" ng-readonly="viewMode" ng-
disabled="viewMode"><option value="?"
selected="selected"></option><option value="string:USD" label="US
Dolar [$]">US Dolar [$]</option><option value="string:GBP"
label="British Pound [£]">British Pound [£]</option><option
value="string:EUR" label="Euro [€]">Euro [€]</option></select>
        </div>
      </div>
      <div class="col-md-3">
        <div class="form-group">
          <label class="control-label"
for="budgetBase.destination">Currency Destination</label>
          <select class="form-control ng-
pristine ng-untouched ng-valid" id="budgetBase.destination" ng-
model="proposal.budgetBase.currencyDestination" ng-options="item.id
as item.description for item in resources.currencies" ng-
readonly="viewMode" ng-disabled="viewMode"><option value="?"
selected="selected"></option><option value="string:USD" label="US
Dolar [$]">US Dolar [$]</option><option value="string:GBP"
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
label="British Pound [£]">British Pound [£]</option><option
value="string:EUR" label="Euro [€]">Euro [€]</option></select>
</div>
</div>
<div class="col-md-3">
  <div class="form-group">
    <label class="control-label"
for="budgetBase.currencyExchange">Currency Exchange</label>
    <input class="form-control ng-
pristine ng-untouched ng-valid" id="budgetBase.currencyExchange"
ng-readonly="viewMode" ng-disabled="viewMode" ng-
model="proposal.budgetBase.currencyExchange">
    </div>
  </div>
</div>
<div class="row">
  <div class="col-md-3">
    <div class="form-group has-unit right-
value">
      <label class="control-label"
for="budgetBase.minimumMarginOrigin">Min. Margin Origin (%)</label>
      <input class="form-control ng-
pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.minimumMarginOrigin" app-number="" ng-
model="proposal.budgetBase.minimumMarginOrigin" ng-
readonly="viewMode" ng-disabled="viewMode">
      <span class="form-control-
unit">%</span>
    </div>
  </div>
</div>
<div class="col-md-3">
  <div class="form-group has-unit right-
value">
    <label class="control-label"
for="budgetBase.maximumMarginOrigin">Max. Margin Origin (%)</label>
    <input class="form-control ng-
pristine ng-untouched ng-valid ng-isolate-scope"
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
id="budgetBase.maximumMarginOrigin" app-number="" ng-
model="proposal.budgetBase.maximumMarginOrigin" ng-
readonly="viewMode" ng-disabled="viewMode">
    <span class="form-control-
unit">%</span>
    </div>
</div>
<div class="col-md-3">
    <div class="form-group has-unit right-
value">
        <label class="control-label"
for="budgetBase.minimumMarginDestiny">Min. Margin Destination
(%)</label>
        <input class="form-control ng-
pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.minimumMarginDestiny" align="right" app-number=""
ng-model="proposal.budgetBase.minimumMarginDestiny" ng-
readonly="viewMode" ng-disabled="viewMode">
        <span class="form-control-
unit">%</span>
    </div>
</div>
<div class="col-md-3">
    <div class="form-group has-unit right-
value">
        <label class="control-label"
for="budgetBase.maximumMarginDestiny">Max. Margin Destination
(%)</label>
        <input class="form-control ng-
pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.maximumMarginDestiny" app-number="" ng-
model="proposal.budgetBase.maximumMarginDestiny" ng-
readonly="viewMode" ng-disabled="viewMode">
        <span class="form-control-
unit">%</span>
    </div>
</div>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
</div>
<div class="row">
  <div class="col-md-3">
    <div class="form-group has-unit right-
value">
      <label class="control-label"
for="budgetBase.containerCost40">Container Cost (40'')</label>
      <input class="form-control value
ng-pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.containerCost40" ng-
model="proposal.budgetBase.containerCost40" app-number="" ng-
readonly="viewMode" ng-disabled="viewMode">
      <span class="form-control-unit ng-
binding"></span>
    </div>
  </div>
  <div class="col-md-3">
    <div class="form-group has-unit right-
value">
      <label class="control-label"
for="budgetBase.containerCost20">Container Cost (20'')</label>
      <input class="form-control value
ng-pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.containerCost20" app-number="" ng-
model="proposal.budgetBase.containerCost20" ng-readonly="viewMode"
ng-disabled="viewMode">
      <span class="form-control-unit ng-
binding"></span>
    </div>
  </div>
  <div class="col-md-3">
    <div class="form-group has-unit right-
value">
      <label class="control-label"
for="budgetBase.containerTransportDestiny">Internal Transportation
Destination</label>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
        <input class="form-control value
ng-pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.containerTransportDestiny" app-number="" ng-
model="proposal.budgetBase.containerTransportDestiny" ng-
readonly="viewMode" ng-disabled="viewMode">
        <span class="form-control-unit ng-
binding"></span>
    </div>
</div>
</div>
<div class="row">
    <div class="col-md-3">
        <div class="form-group has-unit right-
value">
            <label class="control-label"
for="budgetBase.clearancePercent">Tax Declaration (%)</label>
            <input class="form-control ng-
pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.clearancePercent" app-number="" ng-
model="proposal.budgetBase.clearancePercent" ng-readonly="viewMode"
ng-disabled="viewMode">
            <span class="form-control-
unit">%</span>
        </div>
    </div>
    <div class="col-md-3">
        <div class="form-group has-unit right-
value">
            <label class="control-label"
for="budgetBase.rightCostPercent">Tax Rights (%)</label>
            <input class="form-control ng-
pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.rightCostPercent" app-number="" ng-
model="proposal.budgetBase.rightCostPercent" ng-readonly="viewMode"
ng-disabled="viewMode">
            <span class="form-control-
unit">%</span>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
</div>
</div>
<div class="col-md-3">
  <div class="form-group has-unit right-
value">
    <label class="control-label"
for="budgetBase.rightCostIRTPercent">Tax Retention (%)</label>
    <input class="form-control ng-
pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.rightCostIRTPercent" app-number="" ng-
model="proposal.budgetBase.rightCostIRTPercent" ng-
readonly="viewMode" ng-disabled="viewMode">
    <span class="form-control-
unit">%</span>
  </div>
</div>
<div class="col-md-3">
  <div class="form-group has-unit right-
value">
    <label class="control-label"
for="budgetBase.sealPercent">Tax Stamp (%)</label>
    <input class="form-control ng-
pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.sealPercent" app-number="" ng-
model="proposal.budgetBase.sealPercent" ng-readonly="viewMode" ng-
disabled="viewMode">
    <span class="form-control-
unit">%</span>
  </div>
</div>
</div>
<div class="row">
  <div class="col-md-3">
    <div class="form-group has-unit right-
value">
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
        <label class="control-label"
for="budgetBase.assemblyPerHour1">Man Cost Type 1</label>
        <input class="form-control value
ng-pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.assemblyPerHour1" app-number="" ng-
model="proposal.budgetBase.assemblyPerHour1" ng-readonly="viewMode"
ng-disabled="viewMode">
        <span class="form-control-unit ng-
binding"></span>
    </div>
</div>
<div class="col-md-3">
    <div class="form-group has-unit right-
value">
        <label class="control-label"
for="budgetBase.assemblyPerHour2">Man Cost Type 2</label>
        <input class="form-control value
ng-pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.assemblyPerHour2" app-number="" ng-
model="proposal.budgetBase.assemblyPerHour2" ng-readonly="viewMode"
ng-disabled="viewMode">
        <span class="form-control-unit ng-
binding"></span>
    </div>
</div>
<div class="col-md-3">
    <div class="form-group has-unit right-
value">
        <label class="control-label"
for="budgetBase.assemblyPerHour3">Man Cost Type 3</label>
        <input class="form-control value
ng-pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.assemblyPerHour3" app-number="" ng-
model="proposal.budgetBase.assemblyPerHour3" ng-readonly="viewMode"
ng-disabled="viewMode">
        <span class="form-control-unit ng-
binding"></span>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
</div>
</div>
</div>
<div class="row">
  <div class="col-md-3">
    <div class="form-group">
      <label class="control-label"
for="budgetBase.productivityFactor">Productivity Factor</label>
      <input class="form-control ng-
pristine ng-untouched ng-valid" id="budgetBase.productivityFactor"
ng-model="proposal.budgetBase.productivityFactor" ng-
readonly="viewMode" ng-disabled="viewMode">
    </div>
  </div>
  <div class="col-md-3">
    <div class="form-group has-unit right-
value">
      <label class="control-label"
for="budgetBase.insurancePercent">Insurance (%)</label>
      <input class="form-control ng-
pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.insurancePercent" app-number="" ng-
model="proposal.budgetBase.insurancePercent" ng-readonly="viewMode"
ng-disabled="viewMode">
      <span class="form-control-
unit">%</span>
    </div>
  </div>
  <div class="col-md-3">
    <div class="form-group has-unit right-
value">
      <label class="control-label"
for="budgetBase.otherCostsPercent">Other Costs (%)</label>
      <input class="form-control ng-
pristine ng-untouched ng-valid ng-isolate-scope"
id="budgetBase.otherCostsPercent" app-number="" ng-
```



## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
model="proposal.budgetBase.otherCostsPercent" ng-
readonly="viewMode" ng-disabled="viewMode">
    <span class="form-control-
unit">%</span>
    </div>
</div>
</div>
</div>
</div>
</div>
<div class="panel-body summary-container">
    <div class="summary-title">
        <b>Auxiliar Base</b>
    </div>
    <div>
        <div class="row">
            <div class="col-md-3">
                <div class="form-group">
                    <label class="control-label"
for="auxiliarBase.numMonths">Months</label>
                    <input class="form-control ng-
pristine ng-untouched ng-valid" id="auxiliarBase.numMonths" ng-
model="proposal.auxiliarBase.numMonths">
                </div>
            </div>
            <div class="col-md-3">
                <div class="form-group">
                    <label class="control-label"
for="auxiliarBase.businessDays">Working Days per month</label>
                    <input class="form-control ng-
pristine ng-untouched ng-valid" id="auxiliarBase.businessDays" ng-
model="proposal.auxiliarBase.businessDays">
                </div>
            </div>
            <div class="col-md-3">
                <div class="form-group">
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```

            <label class="control-label"
for="auxiliarBase.duration">Duration [Days]</label>
            <span class="form-control ng-
binding" id="auxiliarBase.duration" readonly="">
                0
            </span>
        </div>
    </div>
</div>
</div>
</div>
</div>
</div>

<div class="panel-body summary-container">
    <div class="summary-title">
        <b>Container Base</b>
    </div>
    <div>
        <div class="row">
            <div class="col-md-3">
                <div class="form-group">
                    <label class="control-label"
for="containerBase.numcontainers20">Container 20'' Quantity</label>
                    <input class="form-control ng-
pristine ng-untouched ng-valid" id="containerBase.numcontainers20"
ng-model="proposal.containerBase.numcontainers20">
                </div>
            </div>
            <div class="col-md-3">
                <div class="form-group">
                    <label class="control-label"
for="containerBase.numcontainers40">Container 40'' Quantity</label>
                    <input class="form-control ng-
pristine ng-untouched ng-valid" id="containerBase.numcontainers40"
ng-model="proposal.containerBase.numcontainers40">
                </div>
            </div>
        </div>
    </div>
</div>
</div>
</div>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
</div>
</div>

<div class="panel-body summary-container">
  <div class="summary-title">
    <b>Materials</b>
  </div>
  <div>
    <div class="row">
      <div style="margin: 10px">
        <div class="cardex-container">
          <!-- ngRepeat: material in
proposal.materials -->
          <!-- New Cardex -->
          <!-- ng-show="!viewMode" -->
          <div class="cardex cardex-new" ng-
click="controlModal.setMaterial({}, 'create');" data-
target="#appProposalMaterialModal" data-toggle="modal">
            <span class="cardex-
message">New Material</span>
          </div>
        </div>
      </div>
    </div>
  </div>
  <div class="modal fade"
id="appProposalMaterialModal">
    <div class="modal-dialog" style="width: 1000px;">
      <div class="modal-content">
        <!-- ngIf: isDelete -->
        <!-- ngIf: isDetail || isUpdate -->
        <!-- ngIf: isCreate -->
        <div class="modal-body">
          <!-- ngIf: isCreate -->
          <div class="row">
            <div class="col-md-4">
              <div class="form-group">
                <label class="control-label"
for="material-modal.numericCode">Numeric Code</label>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
<input class="form-control ng-pristine
ng-untouched ng-valid" id="material-modal.numericCode" ng-
readonly="true" ng-disabled="true" ng-model="material.numericCode"
readonly="readonly" disabled="disabled">
</div>
</div>
<div class="col-md-8">
<div class="form-group">
<label class="control-label"
for="material-modal.description">Description</label>
<input class="form-control ng-pristine
ng-untouched ng-valid" id="material-modal.description" ng-
readonly="true" ng-disabled="true" ng-model="material.description"
readonly="readonly" disabled="disabled">
</div>
</div>
</div>
<div class="row">
<div class="col-md-4">
<div class="form-group">
<label class="control-label"
for="material-modal.elemType">Type</label>
<select id="material-modal.elemType"
class="form-control ng-pristine ng-untouched ng-valid" ng-
model="material.elemType" ng-options="type.id as type.description
for type in resources.materialTypes" ng-readonly="true" ng-
disabled="true" readonly="readonly" disabled="disabled"><option
value="?" selected="selected"></option><option
value="string:Electrical"
label="Electrical">Electrical</option><option
value="string:Telecom"
label="Telecommunication">Telecommunication</option><option
value="string:Security" label="Security">Security</option><option
value="string:Waters" label="Waters">Waters</option><option
value="string:Sewage" label="Sewage">Sewage</option><option
value="string:Building" label="Building &
Construction">Building & Construction</option><option
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
value="string:Air" label="Air">Air</option><option
value="string:Mechanics"
label="Mechanics">Mechanics</option><option value="string:Other"
label="Other Types">Other Types</option></select>
    </div>
</div>
<div class="col-md-4">
    <div class="form-group">
        <label class="control-label"
for="material-modal.elemSubType">Sub Type</label>
        <select id="material-modal.elemSubType"
class="form-control ng-pristine ng-untouched ng-valid" ng-
model="material.elemSubType" ng-options="type.id as
type.description for type in resources.materialSubTypes" ng-
readonly="true" ng-disabled="true" readonly="readonly"
disabled="disabled"><option value="?"
selected="selected"></option><option value="string:Network"
label="Network">Network</option><option value="string:Power"
label="Power">Power</option><option value="string:Supply"
label="Supply">Supply</option><option value="string:UPS"
label="UPS">UPS</option><option value="string:Generator"
label="Generator">Generator</option><option
value="string:Productive"
label="Productive">Productive</option><option value="string:Source"
label="Source">Source</option><option value="string:Specific"
label="Specific">Specific</option><option value="string:Lighting"
label="Lighting">Lighting</option><option value="string:Cables"
label="Cables">Cables</option><option value="string:Sound"
label="Sound">Sound</option><option value="string:Panels"
label="Panels">Panels</option><option value="string:Pipe"
label="Pipe">Pipe</option><option value="string:Duct"
label="Duct">Duct</option><option value="string:Fire"
label="Fire">Fire</option><option value="string:CCTV"
label="CCTV">CCTV</option><option value="string:Access"
label="Access">Access</option><option value="string:Entry"
label="Entry">Entry</option><option value="string:Intrusion"
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
label="Intrusion">Intrusion</option><option value="string:Other"
label="Other">Other</option></select>

    </div>
  </div>
  <div class="col-md-4">
    <div class="form-group has-unit right-
value">
      <label class="control-label"
for="material-modal.quantity">Quantity</label>
      <input class="form-control ng-pristine
ng-untouched ng-valid ng-isolate-scope" id="material-
modal.quantity" ng-readonly="isDelete || isDetail" ng-
model="material.quantity" app-number="">
      <span class="form-control-unit ng-
binding"></span>
    </div>
  </div>
</div>
<div class="row">
  <div class="col-md-3">
    <div class="form-group has-unit right-
value">
      <label class="control-label"
for="material-modal.price">Price</label>
      <input class="form-control ng-pristine
ng-untouched ng-valid ng-isolate-scope" id="material-modal.price"
ng-readonly="isDelete || isDetail" ng-model="material.price" app-
number="">
      <span class="form-control-unit ng-
binding"></span>
    </div>
  </div>
  <div class="col-md-3">
    <div class="form-group has-unit right-
value">
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
<label class="control-label"
for="material-modal.priceAccessorise">Price Accessorise</label>
  <input class="form-control ng-pristine
ng-untouched ng-valid ng-isolate-scope" id="material-
modal.priceAccessorise" ng-readonly="isDelete || isDetail" ng-
model="material.priceAccessorise" app-number="">
  <span class="form-control-unit ng-
binding"></span>
</div>
</div>
<div class="col-md-3">
  <div class="form-group has-unit right-
value">
    <label class="control-label"
for="material-modal.priceTransport">Price Transport</label>
    <input class="form-control ng-pristine
ng-untouched ng-valid ng-isolate-scope" id="material-
modal.priceTransport" ng-readonly="isDelete || isDetail" ng-
model="material.priceTransport" app-number="">
    <span class="form-control-unit ng-
binding"></span>
  </div>
</div>
<div class="col-md-3">
  <div class="form-group has-unit right-
value">
    <label class="control-label"
for="material-modal.priceLocalDestination">Price Local
Destination</label>
    <input class="form-control ng-pristine
ng-untouched ng-valid ng-isolate-scope" id="material-
modal.priceLocalDestination" ng-readonly="isDelete || isDetail" ng-
model="material.priceLocalDestination" app-number="">
    <span class="form-control-unit ng-
binding"></span>
  </div>
</div>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
</div>
<div class="row">
  <div class="col-md-3">
    <div class="form-group has-unit right-
value">
      <label class="control-label"
for="material-modal.timeAssembly1">Man Time Assembly 1</label>
      <input class="form-control ng-pristine
ng-untouched ng-valid ng-isolate-scope" id="material-
modal.timeAssembly1" ng-readonly="isDelete || isDetail" ng-
model="material.timeAssembly1" app-number="">
      <span class="form-control-unit ng-
binding"></span>
    </div>
  </div>
  <div class="col-md-3">
    <div class="form-group has-unit right-
value">
      <label class="control-label"
for="material-modal.timeAssembly2">Man Time Assembly 2</label>
      <input class="form-control ng-pristine
ng-untouched ng-valid ng-isolate-scope" id="material-
modal.timeAssembly2" ng-readonly="isDelete || isDetail" ng-
model="material.timeAssembly2" app-number="">
      <span class="form-control-unit ng-
binding"></span>
    </div>
  </div>
  <div class="col-md-3">
    <div class="form-group has-unit right-
value">
      <label class="control-label"
for="material-modal.timeAssembly3">Man Time Assembly 3</label>
      <input class="form-control ng-pristine
ng-untouched ng-valid ng-isolate-scope" id="material-
modal.timeAssembly3" ng-readonly="isDelete || isDetail" ng-
model="material.timeAssembly3" app-number="">
```



## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```

        <span class="form-control-unit ng-
binding"></span>
        </div>
    </div>
</div>
<div class="modal-footer">
    <div class="pull-left">
        <!-- ngIf: isDelete -->
        <!-- ngIf: isUpdate -->
        <!-- ngIf: isCreate -->
        <!-- ngIf: isDetail -->
    </div>
    <div class="pull-right">
        <button type="button" class="btn btn-default"
data-dismiss="modal">Cancel</button>
    </div>
</div>
</div>
</div>
</div>
        </div>
    </div>
</div>
<div class="panel-body summary-container">
    <div>
        <div class="row">
            <div class="col-md-12">
                <div class="pull-left">
                    <button type="submit" class="btn
btn-success">Create</button>
                    <button type="button" class="btn
btn-info" ng-click="clearModel();">Clear</button>
                </div>
                <div class="pull-right">

```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.2 – Criação proposta (Ex)

```
        <button type="button" class="btn
btn-info" ng-click="cancel();">Cancel</button>
        </div>
    </div>
</div>
</div>
</div>
</div>
</form>
</div>
</div></ng-view>
    <script type="text/javascript" src="/js/app.js"></script>

</body></html>
```



---

## APÊNDICE

# 3.3

---

### **Criação Material (Ex)**

*Front - Criação Material (Exemplo)*

## Apêndice nº3 - Programa linguagem

### Front

#### 3.3 – Criação Material (Ex)

```
<html lang="en" ng-app="app" ng-controller="AppController"
class="ng-scope"><head><style type="text/css">.ng-
animate.item:not(.left):not(.right){-webkit-transition:0s ease-in-
out left;transition:0s ease-in-out left}</style><style
type="text/css">@charset "UTF-8";[ng\:cloak],[ng-cloak],[data-ng-
cloak],[x-ng-cloak],.ng-cloak,.x-ng-cloak,.ng-hide:not(.ng-hide-
animate){display:none !important;}ng\:form{display:block;}ng-
animate-shim{visibility:hidden;}ng-
anchor{position:absolute;}</style>
  <base href="/">
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Business Manager</title>
  <meta name="description" content="">
  <meta name="viewport" content="width=device-width, initial-
scale=1 user-scalable=no">
  <link rel="stylesheet" ng-href="/css/main.css"
href="/css/main.css">
  <style>
    [ng\:cloak],[ng-cloak],.ng-cloak { display: none
!important; }
  </style>
</head>
<body>
  <nav class="navbar navbar-default navbar-fixed-top ng-
isolate-scope">
  <div class="container-fluid">
    <!-- Brand and toggle get grouped for better mobile display
-->
    <div class="navbar-header">
      <button type="button" class="navbar-toggle collapsed"
data-toggle="collapse" data-target="#bs-example-navbar-collapse-1"
aria-expanded="false">
        <span class="sr-only">Toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.3 – Criação Material (Ex)

```
</button>
<a class="navbar-brand" href="#">Business Manager</a>
</div>

<!-- Collect the nav links, forms, and other content for
toggling -->
<div class="collapse navbar-collapse" id="bs-example-
navbar-collapse-1">
  <ul class="nav navbar-nav navbar-left">
    <!-- ngRepeat: item in navMenu --><li ng-
repeat="item in navMenu" ng-class="{dropdown: item.subitem !==
undefined}" ng-controller="NavMenuController" nav-
menu="navBar.left" class="ng-scope dropdown">
      <!-- ng-class="{dropdown: item.subitem !== undefined, open:
item.open}" -->
      <!--class="dropdown-toggle" data-toggle="dropdown"
role="button" aria-haspopup="true" aria-expanded="{{item.open}}"-->
      <a href="#" data-toggle="dropdown" class="ng-binding" aria-
expanded="false">
        Proposals <span class="caret"></span>
      </a>
      <!-- ngIf: item.subitem !== undefined --><ul class="dropdown-
menu ng-scope" ng-if="item.subitem !== undefined" ng-
init="subNavMenu = item.subitem">
        <li>
          <div class="menu-area-title">
            <span class="menu-area-name ng-binding">Proposal
Menu</span>
            <span class="menu-area-description ng-binding">Use
this options to search for proposals or create a new one. It is
possible to search by client, description or proposal code.</span>
          </div>
        </li>
        <!-- ngIf: subNavMenu.search !== undefined --><li ng-
if="subNavMenu.search !== undefined" ng-
init="search=subNavMenu.search" class="ng-scope">
          <div class="search-field">
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.3 – Criação Material (Ex)

```
<div class="separator ng-binding">Search for
Proposal</div>
  <form id="proposals-search" ng-submit="close();
submitSeach(search.actionPath);" novalidate="novalidate" class="ng-
pristine ng-valid">
    <input type="text" name="searchFor"
placeholder="Search for proposals..." autocomplete="off">
    <div class="search-controls">
      <button type="submit" id="submit"
class="search-button glyphicon glyphicon-search" aria-
hidden="true"></button>
      <button type="button" class="search-advance
glyphicon glyphicon-option-horizontal" aria-hidden="true"></button>
    </div>
  </form>
</div>
</li><!-- end ngIf: subNavMenu.search !== undefined -->
<!-- ngRepeat: subitem in subNavMenu.items --><li ng-
repeat="subitem in subNavMenu.items" class="ng-scope">
  <div class="menu-item">
    <a href="/proposals/new" id="proposals-create">
      <span class="ng-binding">New Proposal</span>
    </a>
  </div>
</li><!-- end ngRepeat: subitem in subNavMenu.items -->
</ul><!-- end ngIf: item.subitem !== undefined -->
</li><!-- end ngRepeat: item in navMenu --><li ng-repeat="item in
navMenu" ng-class="{dropdown: item.subitem !== undefined}" ng-
controller="NavMenuController" nav-menu="navBar.left" class="ng-
scope dropdown">
  <!-- ng-class="{dropdown: item.subitem !== undefined, open:
item.open}" -->
  <!--class="dropdown-toggle" data-toggle="dropdown"
role="button" aria-haspopup="true" aria-expanded="{{item.open}}"-->
  <a href="#" data-toggle="dropdown" class="ng-binding" aria-
expanded="false">
    Materials <span class="caret"></span>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.3 – Criação Material (Ex)

```
</a>
<!-- ngIf: item.subitem !== undefined --><ul class="dropdown-
menu ng-scope" ng-if="item.subitem !== undefined" ng-
init="subNavMenu = item.subitem">
  <li>
    <div class="menu-area-title">
      <span class="menu-area-name ng-binding">Material
Menu</span>
      <span class="menu-area-description ng-binding">Use
this options to search for a material, create materials. Manage
your material units and details. It is possible to search for
material code or description</span>
    </div>
  </li>
  <!-- ngIf: subNavMenu.search !== undefined --><li ng-
if="subNavMenu.search !== undefined" ng-
init="search=subNavMenu.search" class="ng-scope">
    <div class="search-field">
      <div class="separator ng-binding">Search for
Material</div>
      <form id="materials-search" ng-submit="close();
submitSeach(search.actionPath);" novalidate="novalidate" class="ng-
pristine ng-valid">
        <input type="text" name="searchFor"
placeholder="Search for material..." autocomplete="off">
        <div class="search-controls">
          <button type="submit" id="submit"
class="search-button glyphicon glyphicon-search" aria-
hidden="true"></button>
          <button type="button" class="search-advance
glyphicon glyphicon-option-horizontal" aria-hidden="true"></button>
        </div>
      </form>
    </div>
  </li><!-- end ngIf: subNavMenu.search !== undefined -->
  <!-- ngRepeat: subitem in subNavMenu.items --><li ng-
repeat="subitem in subNavMenu.items" class="ng-scope">
```



## Apêndice nº3 - Programa linguagem

### Front

#### 3.3 – Criação Material (Ex)

```
<div class="menu-item">
  <a href="/materials/new" id="material-create">
    <span class="ng-binding">New Material</span>
  </a>
</div>
</li><!-- end ngRepeat: subitem in subNavMenu.items -->
</ul><!-- end ngIf: item.subitem !== undefined -->
</li><!-- end ngRepeat: item in navMenu --><li ng-repeat="item in
navMenu" ng-class="{dropdown: item.subitem !== undefined}" ng-
controller="NavMenuController" nav-menu="navBar.left" class="ng-
scope dropdown">
  <!-- ng-class="{dropdown: item.subitem !== undefined, open:
item.open}" -->
  <!--class="dropdown-toggle" data-toggle="dropdown"
role="button" aria-haspopup="true" aria-expanded="{{item.open}}"-->
  <a href="#" data-toggle="dropdown" class="ng-binding" aria-
expanded="false">
    Management <span class="caret"></span>
  </a>
  <!-- ngIf: item.subitem !== undefined --><ul class="dropdown-
menu ng-scope" ng-if="item.subitem !== undefined" ng-
init="subNavMenu = item.subitem">
    <li>
      <div class="menu-area-title">
        <span class="menu-area-name ng-binding">Management
Menu</span>
        <span class="menu-area-description ng-
binding">Manage your clients in this menu. Search for clients or
create new clients. Search for clients using the client name or
description</span>
      </div>
    </li>
    <!-- ngIf: subNavMenu.search !== undefined --><li ng-
if="subNavMenu.search !== undefined" ng-
init="search=subNavMenu.search" class="ng-scope">
      <div class="search-field">
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.3 – Criação Material (Ex)

```
<div class="separator ng-binding">Search for
Clients</div>
  <form id="client-search" ng-submit="close();
submitSeach(search.actionPath);" novalidate="novalidate" class="ng-
pristine ng-valid">
    <input type="text" name="searchFor"
placeholder="Search for clients..." autocomplete="off">
    <div class="search-controls">
      <button type="submit" id="submit"
class="search-button glyphicon glyphicon-search" aria-
hidden="true"></button>
      <button type="button" class="search-advance
glyphicon glyphicon-option-horizontal" aria-hidden="true"></button>
    </div>
  </form>
</div>
</li><!-- end ngIf: subNavMenu.search !== undefined -->
<!-- ngRepeat: subitem in subNavMenu.items --><li ng-
repeat="subitem in subNavMenu.items" class="ng-scope">
  <div class="menu-item">
    <a href="/units" id="units-config">
      <span class="ng-binding">Units</span>
    </a>
  </div>
</li><!-- end ngRepeat: subitem in subNavMenu.items --><li
ng-repeat="subitem in subNavMenu.items" class="ng-scope">
  <div class="menu-item">
    <a href="/currencies" id="currency-config">
      <span class="ng-binding">Currencies</span>
    </a>
  </div>
</li><!-- end ngRepeat: subitem in subNavMenu.items -->
</ul><!-- end ngIf: item.subitem !== undefined -->
</li><!-- end ngRepeat: item in navMenu -->
  </ul>
  <ul class="nav navbar-nav navbar-right">
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.3 – Criação Material (Ex)

```
<!-- ngRepeat: item in navMenu --><li ng-repeat="item in navMenu" ng-class="{dropdown: item.subitem !== undefined}" ng-controller="NavMenuController" nav-menu="navBar.right" class="ng-scope dropdown">
  <!-- ng-class="{dropdown: item.subitem !== undefined, open: item.open}" -->
  <!--class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true" aria-expanded="{{item.open}}"-->
  <a href="#" data-toggle="dropdown" class="ng-binding" aria-expanded="false">
    User <span class="caret"></span>
  </a>
  <!-- ngIf: item.subitem !== undefined --><ul class="dropdown-menu ng-scope" ng-if="item.subitem !== undefined" ng-init="subNavMenu = item.subitem">
    <li>
      <div class="menu-area-title">
        <span class="menu-area-name ng-binding">Configuration Menu</span>
        <span class="menu-area-description ng-binding">Use this menu to manage material units, currencies details.</span>
      </div>
    </li>
    <!-- ngIf: subNavMenu.search !== undefined -->
    <!-- ngRepeat: subitem in subNavMenu.items --><li ng-repeat="subitem in subNavMenu.items" class="ng-scope">
      <div class="menu-item">
        <a href="/users/logoff" id="user-logoff">
          <span class="ng-binding">User Logoff</span>
        </a>
      </div>
    </li><!-- end ngRepeat: subitem in subNavMenu.items --><li ng-repeat="subitem in subNavMenu.items" class="ng-scope">
      <div class="menu-item">
        <a href="/users" id="user-details">
          <span class="ng-binding">User details</span>
        </a>
      </div>
    </li>
  </ul>
</li>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.3 – Criação Material (Ex)

```
</div>
</li><!-- end ngRepeat: subitem in subNavMenu.items -->
</ul><!-- end ngIf: item.subitem !== undefined -->
</li><!-- end ngRepeat: item in navMenu -->
</ul>
</div>
</div>
</nav>
<!-- ngView: undefined --><ng-view class="ng-scope"><div
class="container-fluid container-wrapper ng-scope">
<div class="container-header">
<h1>Create a new Material</h1>
</div>
<div class="container-body">
<div class="container-description">
You are ready to create a new material in the system.
This new material will become available to be
added into new or current proposals. After filling the
fields, click create to conclude the process.
</div>
<div class="panel-body">
<form ng-submit="createMaterial(material);" class="ng-
pristine ng-valid">
<div class="row">
<div class="col-md-4">
<div class="form-group">
<label for="numericCode"
class="control-label">Numeric Code</label>
<input id="numericCode"
name="numericCode" class="form-control ng-pristine ng-untouched ng-
valid" type="text" placeholder="Enter a numeric code" ng-
model="material.numericCode">
</div>
</div>
</div>
<div class="row">
<div class="col-md-12">
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.3 – Criação Material (Ex)

```
<div class="form-group">
  <label for="description"
class="control-label">Description</label>
  <textarea id="description"
name="description" class="form-control ng-pristine ng-untouched ng-
valid" placeholder="Enter a material description" ng-
model="material.description"></textarea>
</div>
</div>
<div class="row">
  <div class="col-md-4">
    <div class="form-group">
      <label for="elemType" class="control-
label">Type</label>
      <select id="elemType" name="elemType"
class="form-control ng-pristine ng-untouched ng-valid" ng-
model="material.elemType" ng-options="type.id as type.description
for type in resources.materialTypes"><option value="" class=""
selected="selected">-- Select a Material Category --
</option><option value="string:Electrical"
label="Electrical">Electrical</option><option
value="string:Telecom"
label="Telecommunication">Telecommunication</option><option
value="string:Security" label="Security">Security</option><option
value="string:Waters" label="Waters">Waters</option><option
value="string:Sewage" label="Sewage">Sewage</option><option
value="string:Building" label="Building &
Construction">Building & Construction</option><option
value="string:Air" label="Air">Air</option><option
value="string:Mechanics"
label="Mechanics">Mechanics</option><option value="string:Other"
label="Other Types">Other Types</option></select>
    </div>
  </div>
  <div class="col-md-4">
    <div class="form-group">
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.3 – Criação Material (Ex)

```
        <label for="elemSubtype"
class="control-label">SubType</label>
        <select id="elemSubtype"
name="elemSubtype" class="form-control ng-pristine ng-untouched ng-
valid" ng-model="material.elemSubType" ng-options="type.id as
type.description for type in resources.materialSubTypes"><option
value="" class="" selected="selected">-- Select a Material Sub
Category --</option><option value="string:Network"
label="Network">Network</option><option value="string:Power"
label="Power">Power</option><option value="string:Supply"
label="Supply">Supply</option><option value="string:UPS"
label="UPS">UPS</option><option value="string:Generator"
label="Generator">Generator</option><option
value="string:Productive"
label="Productive">Productive</option><option value="string:Source"
label="Source">Source</option><option value="string:Specific"
label="Specific">Specific</option><option value="string:Lighting"
label="Lighting">Lighting</option><option value="string:Cables"
label="Cables">Cables</option><option value="string:Sound"
label="Sound">Sound</option><option value="string:Panels"
label="Panels">Panels</option><option value="string:Pipe"
label="Pipe">Pipe</option><option value="string:Duct"
label="Duct">Duct</option><option value="string:Fire"
label="Fire">Fire</option><option value="string:CCTV"
label="CCTV">CCTV</option><option value="string:Access"
label="Access">Access</option><option value="string:Entry"
label="Entry">Entry</option><option value="string:Intrusion"
label="Intrusion">Intrusion</option><option value="string:Other"
label="Other">Other</option></select>
        </div>
    </div>
    <div class="col-md-4">
        <div class="form-group">
            <label for="unitType.id"
class="control-label">Unit</label>
            <select id="unitType.id"
name="unitType.id" class="form-control ng-pristine ng-untouched ng-
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.3 – Criação Material (Ex)

```
valid" ng-model="material.units" ng-options="type.id as
type.description for type in resources.materialUnits"><option
value="" class="" selected="selected">-- Select a unit --
</option><option value="string:1" label="Meter [m]">Meter
[m]</option><option value="string:2" label="Kilograms
[kg]">Kilograms [kg]</option><option value="string:3" label="Pound
[lb]">Pound [lb]</option><option value="string:4" label="Liters
[l]">Liters [l]</option><option value="string:5" label="Galons
[g]">Galons [g]</option></select>
    </div>
  </div>
</div>
<div class="row">
  <div class="col-md-4">
    <div class="form-group">
      <label for="refPrice" class="control-
label">Ref. Price</label>
      <input id="refPrice" name="refPrice"
class="form-control ng-pristine ng-untouched ng-valid"
placeholder="Set the Reference cost for Price" ng-
model="material.refPrice">
    </div>
  </div>
  <div class="col-md-4">
    <div class="form-group">
      <label for="refPriceAccessorise"
class="control-label">Ref. Price Accessorise</label>
      <input id="refPriceAccessorise"
name="refPriceAccessorise" class="form-control ng-pristine ng-
untouched ng-valid" placeholder="Set the Reference cost for
Accessorise" ng-model="material.refPriceAccessorise">
    </div>
  </div>
  <div class="col-md-4">
    <div class="form-group">
      <label for="refPriceTransport"
class="control-label">Ref. Price Transport</label>
```

## Apêndice nº3 - Programa linguagem

### Front

#### 3.3 – Criação Material (Ex)

```
<input id="refPriceTransport"
name="refPriceTransport" class="form-control ng-pristine ng-
untouched ng-valid" placeholder="Set the Reference cost for
Transportation" ng-model="material.refPriceTransport">
</div>
</div>
</div>
<div class="row">
<div class="col-md-4">
<div class="form-group">
<label for="refTimeAssembly1"
class="control-label">Ref. Time Assembly 1</label>
<input id="refTimeAssembly1"
name="refTimeAssembly1" class="form-control ng-pristine ng-
untouched ng-valid" placeholder="Set the Reference cost for Time
Assembly 1" ng-model="material.refTimeAssembly1">
</div>
</div>
<div class="col-md-4">
<div class="form-group">
<label for="refTimeAssembly2"
class="control-label">Ref. Time Assembly 2</label>
<input id="refTimeAssembly2"
name="refTimeAssembly2" class="form-control ng-pristine ng-
untouched ng-valid" placeholder="Set the Reference cost for Time
Assembly 2" ng-model="material.refTimeAssembly2">
</div>
</div>
<div class="col-md-4">
<div class="form-group">
<label for="refTimeAssembly3"
class="control-label">Ref. Time Assembly 3</label>
<input id="refTimeAssembly3"
name="refTimeAssembly3" class="form-control ng-pristine ng-
untouched ng-valid" placeholder="Set the Reference cost for Time
Assembly 3" ng-model="material.refTimeAssembly3">
</div>
</div>
```



## Apêndice nº3 - Programa linguagem

### Front

#### 3.3 – Criação Material (Ex)

```
</div>
</div>
<div class="row">
  <div class="col-md-12">
    <div class="pull-left">
      <button type="button" class="btn btn-
info    " ng-click="clearModel()">Clear</button>
      <button type="submit" class="btn btn-
success ">Create</button>
    </div>
  </div>
</div>
</form>
</div>
</div>
</div></ng-view>
  <script type="text/javascript" src="/js/app.js"></script>

</body></html>
```

---

## APÊNDICE

# 4

---

### Teste de assertividade

*Teste efetuados para comprovar funcionamento correto - programa*



---

## APÊNDICE

# 4.1

---

### **Teste de valores (resultados manuais com as fórmulas automáticas)**

*Comparação resultados*

## Apêndice nº4 – Teste de assertividade

### 4.1 – Teste de valores (resultados manuais com as fórmulas automáticas)

#### Comparação resultados

Para a verificação do bom funcionamento, programa e respetivos resultados foram executados vários testes, pelo que foi exposto um exemplo.

Para a execução do respetivo teste teve-se se em contas os seguintes *Inputs*:

Variável introduzida	Valor introduzido
<b>i1</b>	Teste
<b>i2</b>	Ang
<b>i3</b>	50000
<b>i4</b>	Int_00x
<b>i5</b>	I.B.S
<b>i6</b>	07/03/2018
<b>i7</b>	Eurico Clemente
<b>i8</b>	Rev_01
<b>i9</b>	1,4
<b>i10</b>	15
<b>i11</b>	15
<b>i12</b>	25
<b>i13</b>	25
<b>i14</b>	1
<b>i15</b>	1,5
<b>i16</b>	4000
<b>i17</b>	2650
<b>i18</b>	1750
<b>i19</b>	0
<b>i20</b>	10,5
<b>i21</b>	0
<b>i22</b>	3,5

## Apêndice nº4 – Teste de assertividade

### 4.1 – Teste de valores (resultados manuais com as fórmulas automáticas)

<b>i23</b>	1
<b>i24</b>	30
<b>i25</b>	30
<b>i26</b>	30
<b>i27</b>	3
<b>i28</b>	20
<b>i29</b>	2
<b>i30</b>	1
<b>i31</b>	Electrical
<b>i32</b>	power
<b>i33.1</b>	1.1
<b>i34.1</b>	d 25
<b>i35.1</b>	M
<b>i36.1</b>	50
<b>i37.1</b>	0,35
<b>i38.1</b>	0,22
<b>i39.1</b>	0
<b>i40.1</b>	0
<b>i41.1</b>	1000
<b>i42.1</b>	0
<b>i43.1</b>	0
<b>i33.2</b>	1.2
<b>i34.2</b>	d 32
<b>i35.2</b>	m
<b>i36.2</b>	100
<b>i37.2</b>	0,53
<b>i38.2</b>	0,33
<b>i39.2</b>	0
<b>i40.2</b>	0
<b>i41.2</b>	0
<b>i42.2</b>	15

## Apêndice nº4 – Teste de assertividade

### 4.1 – Teste de valores (resultados manuais com as fórmulas automáticas)

<b>i43.2</b>	0
<b>i33.3</b>	1.3
<b>i34.3</b>	d 40
<b>i35.3</b>	m
<b>i36.3</b>	10
<b>i37.3</b>	0,67
<b>i38.3</b>	0,44
<b>i39.3</b>	0
<b>i40.3</b>	0
<b>i41.3</b>	0
<b>i42.3</b>	0
<b>i43.3</b>	15
<b>i44</b>	0

Nota: encontram-se presentes três elementos/materiais distintos (valores compreendidos entre i33 e i43) e identificados como .1, .2 e .3 em array.

Cálculos comparativos do valor das variáveis, entre ambos os tipos:

Fórmula de cálculo	Valores		
	Cálculo manual/teórico	Cálculo Programa	
<b>A1 = i27 × i28</b>	60	60	✓
<b>A2 = A94 × i14</b>	833,333	833,333	✓
<b>A3 = A2 ÷ 8</b>	104,167	104,167	✓
<b>A4 = A3 ÷ A1</b>	2	2	✓
<b>A5 = A4 ÷ 2</b>	1	1	✓
<b>A6 = A96 × i14</b>	25	25	✓
<b>A7 = A6 ÷ 8</b>	3,125	3,125	✓
<b>A8 = A7 ÷ A1</b>	1	1	✓
<b>A9 = A8 ÷ 2</b>	0	0	✓
<b>A10 = A98 × i14</b>	2,5	2,5	✓
<b>A11 = A10 ÷ 8</b>	0,313	0,313	✓

## Apêndice nº4 – Teste de assertividade

### 4.1 – Teste de valores (resultados manuais com as fórmulas automáticas)

<b>A12 = A11 ÷ A1</b>	1	1	✓
<b>A13 = A12 ÷ 2</b>	0	0	✓
<b>A14 = A2 + A6 + A10</b>	860,83	860,83	✓
<b>A15 = A14 ÷ 8</b>	108	108	✓
<b>A16 = A15 ÷ A1</b>	2	2	✓
<b>A17 = A16 ÷ 2</b>	1	1	✓
<b>A18 = A84</b>	47518,15	47518,15	✓
<b>A19 = [A66 × i15 × (1 + (i10 ou i11))] × i9</b>	3,49	3,49	✓
<b>A20 = [i29 × (i17 × i9) + i30(i16 × i9)] + [i29 × (i17 × i9) + i30(i16 × i9)] × (i10 ou i11)</b>	14973	14973	✓
<b>A21 = (i29 + i30) × i18</b>	5175	5175	✓
<b>A22 = A18 + A19 + A20 + A21</b>	67669,64	67669,64	✓
<b>A23 = A20 ÷ A84</b>	31,51	31,51	✓
<b>A24 = A21 ÷ A84</b>	10,89	10,89	✓
<b>A27 = A57 + A58 + A59</b>	7,5	7,5	✓
<b>A28 = i37 + i38 + i39</b>	1,11	1,11	✓
<b>A29 = A28 × (i10 ou i11)</b>	0,17	0,17	✓
<b>A30 = A28 + A29</b>	1,28	1,28	✓
<b>A31 = A30 × i9</b>	1,79	1,79	✓
<b>A32 = A27 × i14 × i9</b>	10,50	10,50	✓
<b>A33 = A31 + A32</b>	12,286	12,286	✓
<b>A34 = A31 × i20</b>	0,19	0,19	✓
<b>A35 = A31 × i21</b>	0	0	✓
<b>A36 = A33 × i22</b>	0,43	0,43	✓
<b>A37 = A33 × i23</b>	0,12	0,12	✓
<b>A38 = (A31 + i40 + A34 + A35) + (A31 × i22) + (A31 × i23)</b>	2,06	2,06	✓



## Apêndice nº4 – Teste de assertividade

### 4.1 – Teste de valores (resultados manuais com as fórmulas automáticas)

<b>A39 = A32 + (A32 × i22) + (A32 × i23)</b>	10,97	10,97	✓
<b>A40 = A38 + A39</b>	13,03	13,03	✓
<b>A41 = A38 × (i12 ou i13)</b>	0,51	0,51	✓
<b>A42 = A39 × (i12 ou i13)</b>	2,74	2,74	✓
<b>A43 = A40 × (i12 ou i13)</b>	3,26	3,26	✓
<b>A44 = A38 + A41</b>	2,57	2,57	✓
<b>A45 = A39 + A42</b>	13,72	13,72	✓
<b>A46 = A40 + A43</b>	16,28	16,28	✓
<b>A47 = A46 × i36</b>	162,85	162,85	✓
<b>A48 = A46 × i44</b>	16,28	16,28	✓
<b>A49 = A48 × i36</b>	162,85	162,85	✓
<b>A50 = i36 × A30 × i15 × [1 + (i10 ou i11)] × i9</b>	0,31	0,31	✓
<b>A51 = A47 × (A20 ÷ A84)</b>	51,31	51,31	✓
<b>A52 = A47 × (A21 ÷ A84)</b>	17,73	17,73	✓
<b>A53 = A44 + (A50 + A51 + A52) ÷ i36</b>	9,5	9,5	✓
<b>A54 = A45</b>	13,72	13,72	✓
<b>A55 = A53 + A54</b>	23,22	23,22	✓
<b>A56 = A47 + A50 + A51 + A52</b>	232,2	232,2	✓
<b>A57 = i41 × (i24 ÷ 60)</b>	0	0	✓
<b>A58 = i42 × (i25 ÷ 60)</b>	0	0	✓
<b>A59 = i43 × (i26 ÷ 60)</b>	7,5	7,5	✓
<b>A60 = <math>\sum</math> (i36 × i37)</b>	77,2	77,2	✓
<b>A61 = <math>\sum</math> (i36 × A27)</b>	25825	25825	✓
<b>A62 = <math>\sum</math> (i36 × i38)</b>	48,4	48,4	✓
<b>A63 = <math>\sum</math> (i36 × i39)</b>	0	0	✓

## Apêndice nº4 – Teste de assertividade

### 4.1 – Teste de valores (resultados manuais com as fórmulas automáticas)

$A64 = \sum (i36 \times A28)$	125,6	125,6	✓
$A65 = \sum (i36 \times A29)$	18,84	18,84	✓
$A66 = \sum (i36 \times A30)$	144,44	144,44	✓
$A67 = \sum (i36 \times A31)$	202,22	202,22	✓
$A68 = \sum (i36 \times A32)$	36155	36155	✓
$A69 = \sum (i36 \times A33)$	36357,22	36357,22	✓
$A70 = \sum (i36 \times i40)$	0	0	✓
$A71 = \sum (i36 \times A34)$	21,23	21,23	✓
$A72 = \sum (i36 \times A35)$	0	0	✓
$A73 = \sum (i36 \times A36)$	1272,5	1272,5	✓
$A74 = \sum (i36 \times A37)$	363,57	363,57	✓
$A75 = \sum (i36 \times A38)$	232,55	232,55	✓
$A76 = \sum (i36 \times A39)$	37781,98	37781,98	✓
$A77 = \sum (i36 \times A40)$	38014,52	38014,52	✓
$A78 = \sum (i36 \times A41)$	58,14	58,14	✓
$A79 = \sum (i36 \times A42)$	9445,49	9445,49	✓
$A80 = \sum (i36 \times A43)$	9503,63	9503,63	✓
$A81 = \sum (i36 \times A44)$	290,69	290,69	✓
$A82 = \sum (i36 \times A45)$	47227,47	47227,47	✓
$A83 = \sum (i36 \times A46)$	47518,15	47518,15	✓
$A84 = \sum A47$	47518,15	47518,15	✓

## Apêndice nº4 – Teste de assertividade

### 4.1 – Teste de valores (resultados manuais com as fórmulas automáticas)

$A85 = \sum A48$	947,68	947,68	✓
$A86 = \sum A49$	47518,15	47518,15	✓
$A87 = \sum A50$	3,49	3,49	✓
$A88 = \sum A51$	14973	14973	✓
$A89 = \sum A52$	5175	5175	✓
$A90 = \sum (i36 \times A53)$	20442,17	20442,17	✓
$A91 = \sum (i36 \times A54)$	47227,47	47227,47	✓
$A92 = \sum (i36 \times A55)$	67669,64	67669,64	✓
$A94 = \left[ \sum (i36 \times i41) \right] \div 60$	833,33	833,33	✓
$A95 = \sum (i36 \times A57)$	25000	25000	✓
$A96 = \left[ \sum (i36 \times i42) \right] \div 60$	25	25	✓
$A97 = \sum (i36 \times A58)$	750	750	✓
$A98 = \left[ \sum (i36 \times i43) \right] \div 60$	2,5	2,5	✓
$A99 = \sum (i36 \times A59)$	75	75	✓
$A100 = A66 + [A66 \times i15 \times (1 + (i10 \text{ ou } i11))] + (A20 \div i9)$	10841,93	10841,93	✓
$A101 = A22$	67669,64	67669,64	✓
$A102 = A18$	47518,15	47518,15	✓
$A103 = A64$	125,6	125,6	✓
$A104 = A61 \times i9 \times i14$	36155	36155	✓
$A105 = A65$	18,84	18,84	✓
$A106 = A80$	9503,63	9503,63	✓
$A107 = A73$	1272,5	1272,5	✓

## Apêndice nº4 – Teste de assertividade

### 4.1 – Teste de valores (resultados manuais com as fórmulas automáticas)

<b>A108 = A74</b>	363,57	363,57	✓
<b>A109 = A72</b>	0	0	✓
<b>A110 = (i29 × i17 + i30 × i16) × i9</b>	13020	13020	✓
<b>A111 = (i29 + i30) × i18</b>	5175	5175	✓
<b>A112 = A71</b>	21,23	21,23	✓
<b>A113 = A19</b>	3,49	3,49	✓
<b>A114 = A103 ÷ [A103 + A104 ÷ i9]</b>	0,48	0,48	✓
<b>A115 = A104 ÷ [A104 + A103 × i9]</b>	99,52	99,52	✓
<b>A116 = A105 ÷ A103</b>	15	15	✓
<b>A117 = A106 ÷ A77</b>	25	25	✓
<b>A118 = A20 + A21</b>	20148	20148	✓
<b>A119 = A118 + A19 + A86</b>	67669,64	67669,64	✓
<b>A120 = i36 × A30</b>	12,77	12,77	✓
<b>A121 = A19 ÷ i9</b>	2,49	2,49	✓
<b>A122 = A20 ÷ i9</b>	10695	10695	✓
<b>A123 = A122 + A121 + A120</b>	10841,93	10841,93	✓
<b>A124 = A27 × i14</b>	7,5	7,5	✓
<b>A125 = A124 × i36</b>	75	75	✓
<b>A126 = <math>\sum</math> A125</b>	25825	25825	✓
<b>A127 = A66 + A126</b>	25969,44	25969,44	✓

Nota: A25 e A26 não se encontram indicadas, pois estas foram usadas como variáveis auxiliares (permitindo guardar valores de transição, uma vez que estes são necessários para o bom funcionamento do programa/simulador supracitado).

Confirma-se o correto funcionamento do programa de cálculo desenvolvido.



---

## APÊNDICE

# 4.2

---

### **Programa de teste aos inputs**

*Programa de teste aos inputs, com valores forçados para verificação dos mesmos no modo teste*

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-- Apaga a base de dados!
DROP DATABASE mestrado;

-- Cria a base de dados chamada mestrado
CREATE DATABASE IF NOT EXISTS mestrado;

-- Client
-- Cria uma tabela chamada client na base de dados mestrado
-- com as diversas colunas.
CREATE TABLE IF NOT EXISTS mestrado.client (
  clientId      INT            UNSIGNED NOT NULL UNIQUE AUTO_INCREMENT,
  name          VARCHAR(200)    NOT NULL UNIQUE,
  address       VARCHAR(200)    NOT NULL,
  email        VARCHAR(100)    NOT NULL,
  phoneNumber   VARCHAR(100)    NOT NULL,
  PRIMARY KEY (clientId)
)AUTO_INCREMENT=1;

-- Currency
-- Cria uma tabela chamada currency na base de dados mestrado
-- com as diversas colunas.
CREATE TABLE IF NOT EXISTS mestrado.currency (
  currencyId INT            UNSIGNED NOT NULL UNIQUE AUTO_INCREMENT,
  name       VARCHAR(500),
  symbol     VARCHAR(500),

  PRIMARY KEY (currencyId)
);

-- Proposal
-- Cria uma tabela chamada proposal na base de dados mestrado
-- com as diversas colunas.
CREATE TABLE IF NOT EXISTS mestrado.Proposal (
  proposalId      INT            UNSIGNED NOT NULL UNIQUE
AUTO_INCREMENT,
  clientId        INT            UNSIGNED NOT NULL,
  name           VARCHAR(500),
  localSite     VARCHAR(500),
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
marginLimit          FLOAT DEFAULT 0,
clientResponsible   VARCHAR(500),
revisionVersion     VARCHAR(500),
proposalType        VARCHAR(20) DEFAULT 'NATIONAL',
deliveryDate        DATETIME,
PRIMARY KEY (proposalId),
FOREIGN KEY (clientId) REFERENCES mestrado.client (clientId)
);

-- AuxiliarBase
-- Cria uma tabela chamada auxiliarBase na base de dados mestrado
-- com as diversas colunas.
CREATE TABLE IF NOT EXISTS mestrado.AuxiliarBase (
    auxiliarBaseId INT          UNSIGNED NOT NULL UNIQUE AUTO_INCREMENT,
    proposalId      INT          UNSIGNED NOT NULL,
    numMonths       INT DEFAULT 0,
    businessDays    INT DEFAULT 0,

    PRIMARY KEY (auxiliarBaseId),
    FOREIGN KEY (proposalId) REFERENCES mestrado.Proposal (proposalId)
);

-- BudgetBase
-- Cria uma tabela chamada BudgetBase na base de dados mestrado
-- com as diversas colunas.
CREATE TABLE IF NOT EXISTS mestrado.BudgetBase (
    budgetBaseId          INT          UNSIGNED NOT NULL UNIQUE
AUTO_INCREMENT,
    proposalId            INT          UNSIGNED NOT NULL,
    currencyOriId         INT          UNSIGNED NOT NULL,
    currencyDstId         INT          UNSIGNED NOT NULL,
    currencyExchange      DOUBLE DEFAULT 0.0,
    minimumMarginOrigin   DOUBLE DEFAULT 0.0,
    maximumMarginOrigin   DOUBLE DEFAULT 0.0,
    minimumMarginDestiny  DOUBLE DEFAULT 0.0,
    maximumMarginDestiny  DOUBLE DEFAULT 0.0,
    productivityFactor     DOUBLE DEFAULT 0.0,
    insurancePercent       DOUBLE DEFAULT 0.0,
```



## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
containerCost40          DOUBLE DEFAULT 0.0,
containerCost20          DOUBLE DEFAULT 0.0,
containerTransportDestiny DOUBLE DEFAULT 0.0,
containerTransportOrigin DOUBLE DEFAULT 0.0,
clearancePercent        DOUBLE DEFAULT 0.0,
rightCostPercent        DOUBLE DEFAULT 0.0,
rightCostIRTPercent     DOUBLE DEFAULT 0.0,
sealPercent             DOUBLE DEFAULT 0.0,
assemblyPerHour1        DOUBLE DEFAULT 0.0,
assemblyPerHour2        DOUBLE DEFAULT 0.0,
assemblyPerHour3        DOUBLE DEFAULT 0.0,
otherCostsPercent       DOUBLE DEFAULT 0.0,
salesterms              VARCHAR(500),

-- private String salesterms;          /* Em falta */
PRIMARY KEY (budgetBaseId),
FOREIGN KEY (proposalId) REFERENCES mestrado.Proposal
(proposalId),
FOREIGN KEY (currencyOriId) REFERENCES mestrado.currency
(currencyId),
FOREIGN KEY (currencyDstId) REFERENCES mestrado.currency
(currencyId)
);

-- Company
-- Cria uma tabela chamada client na base de dados mestrado
-- com as diversas colunas.
CREATE TABLE IF NOT EXISTS mestrado.company (
  companyId INT UNSIGNED NOT NULL UNIQUE AUTO_INCREMENT,
  companyName VARCHAR(500),

  PRIMARY KEY (companyId)
);

-- Containers
-- Cria uma tabela chamada containers na base de dados mestrado
-- com as diversas colunas.
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
CREATE TABLE IF NOT EXISTS mestrado.containers (
  containersId INT UNSIGNED NOT NULL UNIQUE AUTO_INCREMENT,
  proposalId INT UNSIGNED NOT NULL,
  numcontainers20 INT DEFAULT 0.0,
  numcontainers40 INT DEFAULT 0.0,

  PRIMARY KEY (containersId),
  FOREIGN KEY (proposalId) REFERENCES mestrado.Proposal (proposalId)
);

-- Unit Type
CREATE TABLE IF NOT EXISTS mestrado.unit_type (
  unitTypeId INT UNSIGNED NOT NULL UNIQUE AUTO_INCREMENT,
  unit VARCHAR(10) NOT NULL,
  description VARCHAR(200),
  PRIMARY KEY (unitTypeId)
);

-- Material
-- Cria uma tabela chamada material na base de dados mestrado
-- com as diversas colunas.
--
-- Ao colocar o numericCode como unique garanto que so posso
-- ter apenas um material como o valor '2' (por exemplo).
CREATE TABLE IF NOT EXISTS mestrado.MATERIAL_DETAILS (
  numericCode INT UNSIGNED UNIQUE NOT NULL,
  description VARCHAR(500),
  elemType VARCHAR(100) NOT NULL,
  elemSubtype VARCHAR(100) NOT NULL,
  unitTypeId INT UNSIGNED NOT NULL,
  refPrice DOUBLE DEFAULT 0.0,
  refPriceAccessorisise DOUBLE DEFAULT 0.0,
  refPriceTransport DOUBLE DEFAULT 0.0,
  refTimeAssembly1 INT DEFAULT 0,
  refTimeAssembly2 INT DEFAULT 0,
  refTimeAssembly3 INT DEFAULT 0,
  PRIMARY KEY (numericCode),
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
FOREIGN KEY (unitTypeId) REFERENCES mestrado.unit_type
(unitTypeId)
);

CREATE TABLE IF NOT EXISTS mestrado.MATERIAL_COST (
  materialCostId INT UNSIGNED NOT NULL UNIQUE
  AUTO_INCREMENT,
  proposalId INT UNSIGNED NOT NULL,
  numericCode INT UNSIGNED NOT NULL,
  quantity DOUBLE DEFAULT 0.0,
  price DOUBLE DEFAULT 0.0,
  priceAccessorise DOUBLE DEFAULT 0.0,
  priceTransport DOUBLE DEFAULT 0.0,
  priceLocalDestination DOUBLE DEFAULT 0.0,
  timeAssembly1 INT DEFAULT 0,
  timeAssembly2 INT DEFAULT 0,
  timeAssembly3 INT DEFAULT 0,
  PRIMARY KEY (materialCostId)
);

ALTER TABLE mestrado.MATERIAL_COST ADD CONSTRAINT
mcost_material_details_fk FOREIGN KEY (numericCode) REFERENCES
mestrado.MATERIAL_DETAILS (numericCode);

ALTER TABLE mestrado.MATERIAL_COST ADD CONSTRAINT mcost_proposal_fk
FOREIGN KEY (proposalId) REFERENCES mestrado.Proposal (proposalId);

-- Revision
-- Cria uma tabela chamada revision na base de dados mestrado
-- com as diversas colunas.
CREATE TABLE IF NOT EXISTS mestrado.revision (
  revisionId INT UNSIGNED NOT NULL UNIQUE AUTO_INCREMENT,
  revisionCommants VARCHAR(500),

  PRIMARY KEY (revisionId)
);

USE mestrado;
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-----  
-----  
-- Proposal for client IBS  
-----  
-----  
  
-- Create Currency  
INSERT INTO currency (currencyId, name, symbol) VALUES (100, 'Euro',  
'€');  
INSERT INTO currency (currencyId, name, symbol) VALUES (101, 'Pound',  
'£');  
INSERT INTO currency (currencyId, name, symbol) VALUES (200,  
'Dollar', '$');  
  
-- Ceate the Unit table  
INSERT INTO UNIT_TYPE(unitTypeId, unit, description) VALUES (1, 'm',  
'meters');  
INSERT INTO UNIT_TYPE(unitTypeId, unit, description) VALUES (2, 'Un',  
'meters');  
  
-- Create List of materials  
INSERT INTO MATERIAL_DETAILS( numericCode, description, elemType,  
elemSubtype, unitTypeId ) VALUES (0000010, 'd 25', 'Building',  
'Lighting', 1);  
INSERT INTO MATERIAL_DETAILS( numericCode, description, elemType,  
elemSubtype, unitTypeId ) VALUES (0000020, 'd 32', 'Building',  
'Lighting', 1);  
INSERT INTO MATERIAL_DETAILS( numericCode, description, elemType,  
elemSubtype, unitTypeId ) VALUES (0000030, 'd 40', 'Building',  
'Lighting', 1);  
INSERT INTO MATERIAL_DETAILS( numericCode, description, elemType,  
elemSubtype, unitTypeId ) VALUES (0000001, 'f 30', 'Building',  
'Lighting', 1);  
INSERT INTO MATERIAL_DETAILS( numericCode, description, elemType,  
elemSubtype, unitTypeId ) VALUES (0000002, 'f 54', 'Building',  
'Lighting', 1);
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
INSERT INTO MATERIAL_DETAILS( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES (0000003, 'f 82', 'Building',
'Lighting', 1);

-- Create the client
INSERT INTO CLIENT
(name, address, email, phoneNumber) VALUES
('I.B.S', 'AV forças armadas, Lisboa', 'email.iBS@iscte.com',
'+(351) 000 000 000');

--
--
-- -- Create the client
-- INSERT INTO CLIENT
-- (name, address, email, phoneNumber) VALUES
-- ('I.B.S', 'Rua Conselheiro Emadio Navarro 1, 1959-007 Lisboa',
'email.iBS@iscte.com', '(+351) 218 317 000');
--
-- -- Create a proposal
-- INSERT INTO PROPOSAL
-- (clientId, name, localSite, marginLimit, clientResponsible,
revisionVersion)
-- SELECT clientId, 'Proposta Teste 1', 'Angola', 50000, 'Eurico
Clemente', null
-- FROM client
-- WHERE name = 'I.B.S';
--
-- -- Create Auxiliar base
-- INSERT INTO AUXILIARBASE
-- (proposalId, numMonths, businessDays)
-- SELECT p.proposalId, 3, 20
-- FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
-- WHERE c.name = 'I.B.S';
--
-- -- Create Budget Base
-- INSERT INTO BUDGETBASE
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
--      (proposalId,      currencyExchange,      minimumMarginOrigin,
maximumMarginOrigin,      minimumMarginDestiny,
maximumMarginDestiny,productivityFactor,
--      insurancePercent,      containerCost40,      containerCost20,
containerTransportDestiny,      containerTransportOrigin,
clearancePercent, rightCostPercent,
--      rightCostIRTPercent,      sealPercent,      assemblyPerHour1,
assemblyPerHour2, assemblyPerHour3,otherCostsPercent, currencyOriId
,currencyDstId)
-- SELECT p.proposalId, 1.4, 15, 15, 25, 25, 1, 1.5, 4000, 2650,
1725, 0.0, 10.5, 0.0, 3.5, 1, 30, 30, 30, 0.0, 100, 100
-- FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
-- WHERE c.name = 'I.B.S';
--
-- -- Ceate the Unit table
-- INSERT INTO UNIT_TYPE(unitTypeId, unit, description) VALUES (1,
'm', 'Meters');
-- INSERT INTO UNIT_TYPE(unitTypeId, unit, description) VALUES (2,
'Un', 'Units');
-- INSERT INTO UNIT_TYPE(unitTypeId, unit, description) VALUES (3,
'Kg', 'Kilograms');
-- INSERT INTO UNIT_TYPE(unitTypeId, unit, description) VALUES (4,
'l', 'Liters');
-- INSERT INTO UNIT_TYPE(unitTypeId, unit, description) VALUES (5,
'ga', 'Gallons');
--
-- -- Create List of materials
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54111,'Tubo do tipo VD em montagem
saliente, assente em braaadeiras com aperto mecanico: a25',
'Mechanics' , 'Duct', 1);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54112,'Tubo do tipo VD em montagem
saliente, assente em braaadeiras com aperto mecanico: a32',
'Mechanics' , 'Duct', 1);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54113,'Tubo do tipo VD em montagem
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
saliente, assente em braaadeiras com aperto mecanico: a40',
'Mechanics' , 'Duct', 1);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54121,'Cabo do tipo XV(0,6-1kV) com
bainha exterior preta, montagem entubada: 5G6', 'Electrical',
'Cables', 1);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54122,'Cabo do tipo XV(0,6-1kV) com
bainha exterior preta, montagem entubada: 5G4', 'Electrical',
'Cables', 1);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54123,'Cabo do tipo XV(0,6-1kV) com
bainha exterior preta, montagem entubada: 3G4', 'Electrical',
'Cables', 1);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54124,'Cabo do tipo XV(0,6-1kV) com
bainha exterior preta, montagem em esteira: 3x70+2G35', 'Electrical',
'Cables', 1);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54125,'Cabo do tipo XV(0,6-1kV) com
bainha exterior preta, montagem em esteira: 3x120+2G70',
'Electrical', 'Cables', 1);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54126,'Cabo do tipo NHXH E90 com
bainha exterior laranja, montagem em esteira: 4x1,5', 'Electrical',
'Cables', 1);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54131,'Equipamentos: Botoneira de
Corte de Energia com dupla sinalizaaao', 'Electrical', 'Access', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54132,'Equipamentos: Unidade de
Alimentaaao Ininterrupta (UPS), com uma potancia 15 kVA - TRI/TRI,
50 Hz, com uma bateria de acumuladores para uma autonomia de 30
minutos a plena carga.', 'Electrical', 'UPS', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54211,'Tubo do tipo VD em montagem
saliente, assente em braaadeiras com aperto mecanico: a20',
'Electrical', 'Duct', 1);
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54221,'Caixas: Caixa de derivacao -
montagem em esteira Material termoplastico 100x100x55mm, IP65-IK09,
devidamente equipada com ligadores rapidos, tampa com aperto
mecanico, incluindo acessario de fixacao a esteira', 'Electrical',
'Specific', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54222,'Caixas: Caixa terminal -
montagem saliente Material termoplastico 100x100x55mm, IP65-IK09,
devidamente equipada com ligadores rapidos, tampa com aperto
mecanico', 'Electrical', 'Specific', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54231,'Cabo do tipo XV(0,6-1kV) com
bainha exterior preta, montagem entubada: 3G2,5', 'Electrical',
'Cables', 1);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54241,'Equipamentos: Tomada schuko
- montagem saliente estanque 230V/16A, com alvaolos protegidos,
incluindo caixa para montagem saliente, IP55-IK07, centro na cor
vermelha', 'Electrical', 'Source', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54242,'Equipamentos: Tomada schuko
- montagem em calha tacnica 230V/16A, com alvaolos protegidos, centro
na cor vermelha', 'Electrical', 'Source', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54311,'Tubo do tipo VD em montagem
saliente, assente em braaadeiras com aperto mecanico: a20',
'Electrical','Duct', 1);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54312,'Tubo do tipo VD em montagem
saliente, assente em braaadeiras com aperto mecanico: a25',
'Electrical','Duct', 1);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54321,'Caixas: Caixa de derivacao -
montagem saliente Material termoplastico 100x100x55mm, IP65-IK09,
devidamente equipada com ligadores rapidos, tampa com aperto
mecanico', 'Electrical', 'Other', 2);
```



## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54322,'Caixas: Caixa de derivacao -
montagem em esteira Material termoplastico 100x100x55mm, IP65-IK09,
devidamente equipada com ligadores rapidos, tampa com aperto
mecanico, incluindo acessario de fixacao a esteira', 'Electrical',
'Other', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(54323,'Caixas: Caixa terminal -
montagem saliente Material termoplastico 100x100x55mm, IP65-IK09,
devidamente equipada com ligadores rapidos, tampa com aperto
mecanico', 'Electrical', 'Other', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56111,'SEGURANaA Sinalatica: 5 -
Extintor de pa ABC 240x85mm em montagem colada a parede', 'Security',
'Panels', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56112,'SEGURANaA Sinalatica: 6 -
Extintor de CO2 240x85mm em montagem colada a parede', 'Security',
'Panels', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56113,'SEGURANaA Sinalatica: 28 -
Saada 400x200mm em montagem colada a parede', 'Security', 'Panels',
2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56114,'SEGURANaA Sinalatica: 28 -
Saada 400x200mm em montagem suspensa', 'Security', 'Panels', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56115,'SEGURANaA Sinalatica: 29 -
Saada a direita 400x200mm em montagem suspensa', 'Security',
'Panels', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56116,'SEGURANaA Sinalatica: 30 -
Saada a esquerda 400x200mm em montagem suspensa', 'Security',
'Panels', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56117,'SEGURANaA Sinalatica: 29 -
Saada a direita 400x200mm em montagem colada a parede', 'Security',
'Panels', 2);
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56118,'SEGURANaA Sinalatica: 35 -
Extintor Do tipo panoramico, 300x300mm em montagem colada a parede',
'Security', 'Panels', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56119,'SEGURANaA Sinalatica: 36 -
Boca de incendio Do tipo panoramico, 300x300mm em montagem colada a
parede', 'Security', 'Panels', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(561110,'SEGURANaA Sinalatica: 37 -
Botao de alarme de incendio Reposicionamento de placa existente do
tipo panoramico com 300x300mm em montagem colada a parede',
'Security', 'Panels', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(561111,'SEGURANaA Sinalatica: PE -
Planta de emergencia 400x300mm em montagem colada a parede',
'Security', 'Panels', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56121,'Meios de Extinaao: Extintor
portatil de pa ABC Com a capacidade de 6kg, cor vermelha, incluindo
acessorio de fixaaao para montagem justaposta a parede, com topo a
1,10m', 'Security','Fire', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56122,'Meios de Extinaao: Extintor
portatil de CO2 com a capacidade de 5kg, cor vermelha, incluindo
suporte para a sua montagem, fixo a parede, com topo a 1,10m',
'Security', 'Fire', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56231,'DETEEaaO DE INCaNDIO: Central
de deteaaao de incendios Endereaavel, 1 loop analagico, fonte de
alimentaaao e respetivas baterias (18ah/autonomia de 72h) e
transmissor de alarmes', 'Security', 'Fire', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56232,'DETEEaaO DE INCaNDIO: Detetor
atico Endereaavel, incluindo base com isolador, para montagem
justaposta ao teto', 'Security', 'Fire', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56233,'DETEEaaO DE INCaNDIO: Botao
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
de alarme Endereavel, para montagem saliente na parede', 'Security',
'Fire', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56234,'DETEaaO DE INCaNDIO: Sirene
Endereavel, incluindo base com isolador, para montagem justaposta
ao teto', 'Security', 'Fire', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56235,'DETEaaO DE INCaNDIO: Madulo
de comando Endereavel, 1 saada, para montagem justaposta a parede',
'Security', 'Fire', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56236,'DETEaaO DE INCaNDIO:
Bloqueador de porta', 'Security', 'Fire', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56237,'DETEaaO DE INCaNDIO: Fonte
de alimentaaaO para bloqueador de porta', 'Security', 'Fire', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56611,'CONTROLO DE ACESSOS: Sistema
de gestao de parque de estacionamento totalmente instalado e pronto
a funcionar, incluindo todos os acessorios', 'Security', 'Access',
2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56731,'VIDEOVIGILaNCIA: Gravador de
video para 16 canais analogicos, a instalar no bastidor de
telecomunicaaaes existente', 'Security', 'CCTV', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56732,'VIDEOVIGILaNCIA: Monitor TFT
19', 'Security', 'CCTV', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56733,'VIDEOVIGILaNCIA: C1 - Camara
de Videovigilancia de Interior, do tipo analagica', 'Security',
'CCTV', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56734,'VIDEOVIGILaNCIA: C2 - Camara
de Videovigilancia de Interior, do tipo mini-dome analagica',
'Security', 'CCTV', 2);
-- INSERT INTO MATERIAL_DETAILS ( numericCode, description, elemType,
elemSubtype, unitTypeId ) VALUES(56735,'VIDEOVIGILaNCIA: C3 - Camara
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

de Videovigilancia de Exterior, do tipo analagica, incluindo caixa para exterior', 'Security', 'CCTV', 2);

--

-- -- Create List of Material Costs for proposal

-- INSERT INTO MATERIAL\_COST

-- (proposalId, numericCode, quantity, price, priceAccessorise, priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2, timeAssembly3)

-- SELECT p.proposalId,54111, 50, 0.35, 0.22,  
0 , 0 , 0 , 0 , 12

-- FROM PROPOSAL p ;

--

-- INSERT INTO MATERIAL\_COST

-- (proposalId, numericCode, quantity, price, priceAccessorise, priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2, timeAssembly3)

-- SELECT p.proposalId,54112, 100, 0.53, 0.33,  
0 , 0 , 0 , 0 , 15

-- FROM PROPOSAL p ;

--

-- INSERT INTO MATERIAL\_COST

-- (proposalId, numericCode, quantity, price, priceAccessorise, priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2, timeAssembly3)

-- SELECT p.proposalId,54113, 10, 0.67, 0.44,  
0, 0 , 0 , 0 , 15

-- FROM PROPOSAL p ;

--

-- INSERT INTO MATERIAL\_COST

-- (proposalId, numericCode, quantity, price, priceAccessorise, priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2, timeAssembly3)

-- SELECT p.proposalId,54121, 10, 2.52, 0 ,  
0 , 0 , 0 , 0 , 7

-- FROM PROPOSAL p ;

--

-- INSERT INTO MATERIAL\_COST

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-- (proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54122,      100,  1.75,      0 ,
0 ,      0 ,      0 ,      0 ,      7
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54123,      50,  1.09,      0 ,
0 ,      0 ,      0 ,      0 ,      6
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54124,      25, 23.69,      0.05,
0 ,      0 ,      0 ,      0 ,      20
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54125,      60, 43.33,      0.05,
0 ,      0 ,      0 ,      0 ,      26
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54126,      120,  1.14,      0.02,
0 ,      0 ,      0 ,      0 ,      6
-- FROM PROPOSAL p ;
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54131,          2, 86.03,          0 ,
0 ,          0 ,          0 ,          0 ,          20
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54132,          1, 4605,          0 ,
0 ,          0 ,          0 ,          0 ,          240
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54211,          80, 0.24,          0.19,
0 ,          0 ,          0 ,          0 ,          10
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54221,          4, 1.65,          6.53,
0 ,          0 ,          0 ,          0 ,          20
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-- SELECT p.proposalId,54222,          1,  1.65,          2.35,
0 ,          0 ,          0 ,          0 ,          20
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54231,          225,  0.65,          0 ,
0 ,          0 ,          0 ,          0 ,          6
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54241,          2,  10.09,          0 ,
0 ,          0 ,          0 ,          0 ,          15
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54242,          4,  7.18,          0 ,
0 ,          0 ,          0 ,          0 ,          10
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54311,          160,  0.24,          0.19,
0 ,          0 ,          0 ,          0 ,          10
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54312,          10,  0.35,          0.22,
0 ,          0 ,          0 ,          0 ,          12
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54321,          11,  1.65,          2.35,
0 ,          0 ,          0 ,          0 ,          20
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54322,          8,  1.65,          6.53,
0 ,          0 ,          0 ,          0 ,          20
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,54323,          2,  1.65,          2.35,
0 ,          0 ,          0 ,          0 ,          20
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56111,          7,  2.73,          0 ,
0 ,          0 ,          0 ,          0 ,          10
-- FROM PROPOSAL p ;
```



## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56112,          1,  2.73,          0 ,
0 ,          0 ,          0 ,          0 ,          10
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56113,          5,  7.78,          0 ,
0 ,          0 ,          0 ,          0 ,          10
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56114,          1,  7.78,          0 ,
0 ,          0 ,          0 ,          0 ,          10
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56115,          9,  9.19,          0 ,
0 ,          0 ,          0 ,          0 ,          10
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-- SELECT p.proposalId,56116,          11,  9.19,          0 ,
0 ,          0 ,          0 ,          0 ,          10
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56117,          1,  9.19,          0 ,
0 ,          0 ,          0 ,          0 ,          10
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56118,          8, 18.14,          0 ,
0 ,          0 ,          0 ,          0 ,          10
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56119,          3, 18.14,          0 ,
0 ,          0 ,          0 ,          0 ,          10
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,561110,          5, 18.14,          0 ,
0 ,          0 ,          0 ,          0 ,          10
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-- (proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,561111,      2, 54.15,      0 ,
0 ,      0 ,      0 ,      0 ,      20
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56121,      7, 24.8,      0 ,
10,      0 ,      0 ,      0 ,      15
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56122,      1, 68.62,      0 ,
15,      0 ,      0 ,      0 ,      15
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56231,      1,770.86,      0 ,
0 ,      0 ,      0 ,      240,      0
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56232,      33, 27.97,      0 ,
0 ,      0 ,      0 ,      15,      0
-- FROM PROPOSAL p ;
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56233,          5, 41.28,          0 ,
0 ,          0 ,          0 ,          15,          0
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56234,          4, 73.97,          0 ,
0 ,          0 ,          0 ,          15,          0
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56235,          3, 31.37,          0 ,
0 ,          0 ,          0 ,          20,          0
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56236,          3, 43.73,          0 ,
0 ,          0 ,          0 ,          20,          0
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-- SELECT p.proposalId,56237,          2, 45.17,          0 ,
0 ,          0 ,          0 ,          30,          0
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56611,          1, 25711.06,          0 ,
5000,          0 ,          0 ,          0 ,          960
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56731,          2, 1732.08,          0 ,
0 ,          0 ,          120,          0 ,          0
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56732,          1, 200,          0 ,
0 ,          0 ,          30,          0 ,          0
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56733,          12, 294.56,          0 ,
0 ,          500,          60,          0 ,          0
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56734,          1, 276.48,          0 ,
0 ,          650,          60,          0 ,          0
-- FROM PROPOSAL p ;
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId,56735,          4, 333.76,          0 ,
0 ,          120,          90,          0 ,          0
-- FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
-- WHERE c.name = 'I.B.S';
--
-- INSERT INTO MATERIAL_COST
-- (proposalId, numericCode, quantity, price, priceAccessorise,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
-- SELECT p.proposalId, 0000030,    10,  0.67,          0.44,
0.0,          0.0,          0.0,          0.0,          15
-- FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
-- WHERE c.name = 'I.B.S';
--
-- INSERT INTO CONTAINERS
-- (proposalId, numcontainers20, numcontainers40)
-- SELECT p.proposalId, 2,  1
-- FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
-- WHERE c.name = 'I.B.S';
--
-- COMMIT;

-----
-----
-- Proposal for client IBS - Proposal 1
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-----  
-----  
  
-- Create a proposal  
INSERT INTO PROPOSAL  
(clientId, name, localSite, marginLimit, clientResponsible,  
revisionVersion)  
SELECT clientId, 'Proposta Teste 1', 'Angola', 50000, 'Eurico  
Clemente', null  
FROM client  
WHERE name = 'I.B.S';  
  
-- Create Auxiliar base  
INSERT INTO AUXILIARBASE  
(proposalId, numMonths, businessDays)  
SELECT max(p.proposalId), 3, 20  
FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId  
WHERE c.name = 'I.B.S';  
  
-- Create Budget Base  
INSERT INTO BUDGETBASE  
(proposalId, currencyExchange, minimumMarginOrigin,  
maximumMarginOrigin, minimumMarginDestiny,  
maximumMarginDestiny, productivityFactor,  
insurancePercent, containerCost40, containerCost20,  
containerTransportDestiny, clearancePercent, rightCostPercent,  
rightCostIRTPercent, sealPercent, assemblyPerHour1,  
assemblyPerHour2, assemblyPerHour3, otherCostsPercent, currencyOriId  
, currencyDstId)  
SELECT max(p.proposalId), 1.4, 15, 15, 25, 25, 1, 1.5, 4000, 2650,  
1725, 10.5, 0.0, 3.5, 1, 30, 30, 30, 0.0, 200, 101  
FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId  
WHERE c.name = 'I.B.S';  
  
-- Create List of Material Costs for proposal  
INSERT INTO MATERIAL_COST
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
(proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
```

```
SELECT max(p.proposalId), 0000010, 50, 0.35, 0.22,
0.0, 0.0, 1000, 0, 0
FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
WHERE c.name = 'I.B.S';
```

```
INSERT INTO MATERIAL_COST
```

```
(proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
```

```
SELECT max(p.proposalId), 0000020, 100, 0.53, 0.33,
0.0, 0.0, 0.0, 15, 0
FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
WHERE c.name = 'I.B.S';
```

```
INSERT INTO MATERIAL_COST
```

```
(proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
```

```
SELECT max(p.proposalId), 0000030, 10, 0.67, 0.44,
0.0, 0.0, 0.0, 0.0, 15
FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
WHERE c.name = 'I.B.S';
```

```
INSERT INTO CONTAINERS
```

```
(proposalId, numcontainers20, numcontainers40)
```

```
SELECT max(p.proposalId), 2, 1
```

```
FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
```

```
WHERE c.name = 'I.B.S';
```

```
COMMIT;
```

```
-----  
-----
```



## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
-- Proposal for client IBS - Proposal 2
-----
-----

-- Create a proposal
INSERT INTO PROPOSAL
(clientId, name, localSite, marginLimit, clientResponsible,
revisionVersion)
  SELECT clientId, 'Proposta Teste 2', 'Angola', 1000, 'Eurico
Clemente', null
  FROM client
  WHERE name = 'I.B.S';

-- Create Auxiliar base
INSERT INTO AUXILIARBASE
(proposalId, numMonths, businessDays)
  SELECT max(p.proposalId), 1, 18
  FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
  WHERE c.name = 'I.B.S';

-- Create Budget Base
INSERT INTO BUDGETBASE
(proposalId, currencyExchange, minimumMarginOrigin,
maximumMarginOrigin, minimumMarginDestiny,
maximumMarginDestiny,productivityFactor,
insurancePercent, containerCost40, containerCost20,
containerTransportDestiny, clearancePercent, rightCostPercent,
rightCostIRTPercent, sealPercent, assemblyPerHour1,
assemblyPerHour2, assemblyPerHour3,otherCostsPercent, currencyOriId
,currencyDstId)
  SELECT max(p.proposalId), 0.90, 15, 20, 25, 35, 1.1, 1.4, 3800,
2500, 1250, 10.5, 3, 3.5, 1.1, 25, 30, 40, 5, 100, 101
  FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
  WHERE c.name = 'I.B.S';

-- Create List of Material Costs for proposal
INSERT INTO MATERIAL_COST
```

## Apêndice nº4 - Teste de assertibilidade

### 4.2 - Programa de teste aos inputs

```
(proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
```

```
SELECT max(p.proposalId), 0000001, 1, 1100, 0.22,
0.0, 1000, 60, 0, 0
FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
WHERE c.name = 'I.B.S';
```

```
INSERT INTO MATERIAL_COST
```

```
(proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
```

```
SELECT max(p.proposalId), 0000002, 50, 1.81, 0.33,
0.0, 0.0, 0.0, 1440, 2
FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
WHERE c.name = 'I.B.S';
```

```
INSERT INTO MATERIAL_COST
```

```
(proposalId, numericCode, quantity, price, priceAccessoris,
priceTransport, priceLocalDestination, timeAssembly1, timeAssembly2,
timeAssembly3)
```

```
SELECT max(p.proposalId), 0000003, 10, 43, 0.44,
20, 0.0, 0.0, 0.0, 850
FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
WHERE c.name = 'I.B.S';
```

```
INSERT INTO CONTAINERS
```

```
(proposalId, numcontainers20, numcontainers40)
```

```
SELECT max(p.proposalId), 2, 1
FROM CLIENT c inner join PROPOSAL p ON c.clientId = p.clientId
WHERE c.name = 'I.B.S';
```

```
COMMIT;
```



---

## APÊNDICE

# 4.3

---

### Programa de teste aos valores

*Output, Programa de teste aos valores, com uma margem de erro máximo apenas 0.01.  
Programa correu sem erros, valor esperado igual ao calculado*

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
package com.ibs.mestrado.utils;

import com.ibs.mestrado.dto.*;
import com.ibs.mestrado.enumerate.ElemSubtype;
import com.ibs.mestrado.enumerate.ElemType;
import com.ibs.mestrado.enumerate.ProposalType;
import com.ibs.mestrado.enumerate.UnitType;
import org.junit.Assert;
import org.junit.Before;
import org.junit.Test;
import java.util.*;

public class FormulaUtilsTest {

    AuxiliarBase base;
    List<Material> materialList;
    BudgetBase budgetBase;
    Containers containers;
    Proposal proposal;
    UnitType unitType;

    @Before
    public void setUpTest(){

        Material materialAux;

        /*Define the Auxiliar base*/
        base = new AuxiliarBase();
        base.setBusinessDays(20);          /*i28 - Define o número
de dias úteis/trabalháveis por mês;*/
        base.setNumMonths(3);            /*i27 - Identifica o
número de meses atribuído (pelo cliente) para execução da obra;*/

        /*Define Budged Base */
        budgetBase = new BudgetBase();
        budgetBase.setCurrencyExchange(1.4);          /* i9 -
exchange/Relationship between coins */
        budgetBase.setMinimumMarginOrigin(15);        /* i10 -
Company margin in source A (Org.) */
        budgetBase.setMaximumMarginOrigin(15);        /* i11 -
Company margin in source B (Org.)*/
        budgetBase.setMinimumMarginDestiny(25);        /* i12 -
Company margin in the target A (Dist.) */
        budgetBase.setMaximumMarginDestiny(25);        /* i13 -
Company margin in the target B (Dist.) */
        budgetBase.setProductivityFactor(1);          /* i14 - Local
productivity (percentage) */
        budgetBase.setInsurancePercent(1.5);          /* i15 -
Insurance value (percentage) */
        budgetBase.setContainerCost40(4000);          /* i16 - Cost
value of a container 40" */
        budgetBase.setContainerCost20(2650);          /* i17 - Cost
value of a container 20" */
        budgetBase.setContainerTransportDestiny(1725); /* i18 -
Transport cost of 1 container in the destination country */
    }
}
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
        budgetBase.setContainerTransportOrigin(0);          /* i19 -
Transport cost of containers in the source country */
        budgetBase.setClearancePercent(10.5);              /* i20 -
Cost related to customs clearance (percentage) */
        budgetBase.setRightCostPercent(0);                /* i21 -
Right Cost (percentage) */
        budgetBase.setRightCostIRTPercent(3.5);           /* i22 -
Costs relating to importation taxes */
        budgetBase.setSealPercent(1);                     /* i23 -
Cost related to the stamp tax (percentage) */
        budgetBase.setAssemblyPerHour1(30);               /* i24 -
Cost of hand labor time type 1 */
        budgetBase.setAssemblyPerHour2(30);               /* i25 -
Cost of hand labor time type 2 */
        budgetBase.setAssemblyPerHour3(30);               /* i26 -
Cost of hand labor time type 3 */
        budgetBase.setOtherCostsPercent(0);                /* i44 -
Amount spent related to other cost (percentage) (only on proposals
with the country of destination and origin is different */

        /*Define the Containers*/
        containers = new Containers();
        containers.setNumcontainers20(2); /*i29 - Representa o
número de um contentor de 20" utilizados;*/
        containers.setNumcontainers40(1); /*i30 - Representa o
número de um contentor de 40" utilizados;*/

        unitType = new UnitType();
        unitType.setDescription("Meters");
        unitType.setId('m');
        unitType.setUnity("m");

        /*Material List*/
        materialList = new ArrayList<>(); /*Build the list where
the materials are*/

        materialAux = new Material(); /*Material d25*/

        materialAux.setElemType(ElemType.Air);             /* i31 -
Tipo de elemento*/
        materialAux.setElemSubtype(ElemSubtype.Cables);   /*
i32 - Subtipo de elemento*/
        materialAux.setNumericCode(1);                     /* i33 -
Contains the material numeric code */
        materialAux.setDescription("d 25");               /* i34 -
Contains the material description */
        materialAux.setUnitType(unitType);                 /* i35 -
Contains the material unit type */
        materialAux.setQuantity(50);                       /* i36 -
Contains the material quantity */
        materialAux.setPrice(0.35);                        /* i37 - Unit
value (materiel) */
        materialAux.setPriceAccessorise(0.22);            /* i38 - Unit
value of accessories */
        materialAux.setPriceTransport(0);                  /* i39 -
Value for the transportation (equipment) */
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
        materialAux.setPriceLocalDestination(0);          /* i40 -
Transport price (international only) */
        materialAux.setTimeAssembly1(1000);              /* i41 -
Assembly time in minutes equipment type 1 */
        materialAux.setTimeAssembly2(0);                 /* i42 -
Assembly time in minutes equipment type 2 */
        materialAux.setTimeAssembly3(0);                 /* i43 -
Assembly time in minutes equipment type 3 */

//TODO: Add Other properties/material nº1
materialList.add(materialAux);

        materialAux = new Material();                    /*Material d32*/

        materialAux.setElemType(ElemType.Air);          /* i31 -
Tipo de elemento*/
        materialAux.setElemSubtype(ElemSubtype.Cables); /*
i32 - Subtipo de elemento*/
        materialAux.setUnitType(unitType);              /* i35 -
Contains the material unit type */
        materialAux.setNumericCode(2);                  /* i33 -
Contains the material numeric code */
        materialAux.setDescription("d 32");             /* i34 -
Contains the material description */
        materialAux.setQuantity(100);                   /* i36 -
Contains the material quantity */
        materialAux.setPrice(0.53);                     /* i37 - Unit
value (materiel) */
        materialAux.setPriceAccessorise(0.33);          /* i38 - Unit
value of accessories */
        materialAux.setPriceTransport(0);               /* i39 -
Value for the transportation (equipment) */
        materialAux.setPriceLocalDestination(0);        /* i40 -
Transport price (international only) */
        materialAux.setTimeAssembly1(0);                 /* i41 -
Assembly time in minutes equipment type 1 */
        materialAux.setTimeAssembly2(15);                /* i42 -
Assembly time in minutes equipment type 2 */
        materialAux.setTimeAssembly3(0);                 /* i43 -
Assembly time in minutes equipment type 3 */

//TODO: Add Other properties/material nº2
materialList.add(materialAux);

        materialAux = new Material();                    /*Material d40*/

        materialAux.setElemType(ElemType.Air);          /* i31 -
Tipo de elemento*/
        materialAux.setElemSubtype(ElemSubtype.Cables); /*
i32 - Subtipo de elemento*/
        materialAux.setUnitType(unitType);              /* i35 -
Contains the material unit type */
        materialAux.setNumericCode(3);                  /* i33 -
Contains the material numeric code */
        materialAux.setDescription("d 40");             /* i34 -
Contains the material description */
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
        materialAux.setQuantity(10);                /* i36 -
Contains the material quantity */
        materialAux.setPrice(0.67);                /* i37 - Unit
value (materiel) */
        materialAux.setPriceAccessories(0.44);     /* i38 - Unit
value of accessories */
        materialAux.setPriceTransport(0);          /* i39 -
Value for the transportation (equipment) */
        materialAux.setPriceLocalDestination(0);   /* i40 -
Transport price (international only) */
        materialAux.setTimeAssembly1(0);           /* i41 -
Assembly time in minutes equipment type 1 */
        materialAux.setTimeAssembly2(0);           /* i42 -
Assembly time in minutes equipment type 2 */
        materialAux.setTimeAssembly3(15);          /* i43 -
Assembly time in minutes equipment type 3 */
//TODO: Add Other properties/material nº3
        materialList.add(materialAux);

        Client client;
        client = new Client();

        Company company;
        company = new Company();
        company.setCompanyId(123);
        company.setCompanyName("example");

        /*Define Proposal */
        proposal = new Proposal();
        proposal.setProposalType(ProposalType.INTERNATIONAL);
/* i0 - national or international budget */
        proposal.setName("teste1");                /* i1 -
Work name */
        proposal.setLocalSite("Angola");           /* i2 -
Work location */
        proposal.setMarginLimit(50000);            /* i3 -
Limit in which the margin A changes to the margin B */
        proposal.setCompany(company);              /* i4 -
Company Reference */
        proposal.setClient(client);                /* i5 -
Customer name */
        proposal.setDeliveryDate(new Date());      /* i6 -
Current date */
        proposal.setClientResponsible("Eurico Clemente"); /* i7 -
It might be deleted */
        proposal.setRevisionVersion("0.1 Rev");
/* i8 - Number of proposal revision */

    }

    @Test /*A1 */
    public void testNumberTotalDays() throws Exception {
        int expected = 60;
        int num;

        num = FormulaUtils.numberTotalDays( base );
        Assert.assertEquals(expected, num );
    }
}
```



## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
}

@Test /*A2 ;*/
public void testNumberHoursTotalCostMan1() throws Exception {
    double expected = 833.33;
    double num;

    num = FormulaUtils.numberHoursTotalCostMan1( materialList,
budgetBase);
    Assert.assertEquals(expected, num, 0.01 );
}

@Test /*A3 ;*/
public void testNumberTotalDaysType1() throws Exception {
    double expected = 104.1666667;
    double num;

    num = FormulaUtils.numberTotalDaysType1( materialList,
budgetBase);
    Assert.assertEquals(expected, num, 0.01 );
}

@Test /*A4 Arredondar para cima ;*/
public void testNumberOfMenType1() throws Exception {
    int expected = 2;
    int num;

    num = FormulaUtils.numberOfMenType1( materialList,
budgetBase, base);
    Assert.assertEquals(expected, num );
}

@Test /*A5 ;*/
public void testNumberOfTeamsType1() throws Exception {
    int expected = 1;
    int num;

    num = FormulaUtils.numberOfTeamsType1( materialList,
budgetBase, base);
    Assert.assertEquals(expected, num );
}

@Test /*A6 ;*/
public void testNumberHoursTotalCostMan2() throws Exception {
    double expected = 25;
    double num;

    num = FormulaUtils.numberHoursTotalCostMan2( materialList,
budgetBase);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A7 ;*/
public void testNumberTotalDaysType2() throws Exception {
    double expected = 3.125;
    double num;
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
        num = FormulaUtils.numberTotalDaysType2( materialList,
budgetBase);
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A8 Arredondar para cima ;*/
    public void testNumberOfMenType2() throws Exception {
        int expected = 1;
        int num;

        num = FormulaUtils.numberOfMenType2( materialList,
budgetBase, base);
        Assert.assertEquals(expected, num );
    }

    @Test /*A9 ;*/
    public void testNumberOfTeamsType2() throws Exception {
        int expected = 0;
        int num;

        num = FormulaUtils.numberOfTeamsType2( materialList,
budgetBase, base);
        Assert.assertEquals(expected, num );
    }

    @Test /*A10 ;*/
    public void testNumberHoursTotalCostMan3() throws Exception {
        double expected = 2.5;
        double num;

        num = FormulaUtils.numberHoursTotalCostMan3( materialList,
budgetBase);
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A11 ;*/
    public void testNumberTotalDaysType3() throws Exception {
        double expected = 0.3125;
        double num;

        num = FormulaUtils.numberTotalDaysType3( materialList,
budgetBase);
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A12 Arredondar para cima ;*/
    public void testNumberOfMenType3() throws Exception {
        int expected = 1;
        int num;

        num = FormulaUtils.numberOfMenType3( materialList,
budgetBase, base);
        Assert.assertEquals(expected, num );
    }

    @Test /*A13 ;*/
    public void testNumberOfTeamsType3() throws Exception {
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
        int expected = 0;
        int num;

        num = FormulaUtils.numberOfTeamsType3( materialList,
budgetBase, base);
        Assert.assertEquals(expected, num );
    }

    @Test /*A14 ;*/
    public void testTotalNumberHoursInEx() throws Exception {
        double expected = 860.83;
        double num;

        num = FormulaUtils.totalNumberHoursInEx( materialList,
budgetBase);
        Assert.assertEquals(expected, num, 0.01 );
    }

    @Test /*A15 ;*/
    public void testTotalNumberDaysInEx() throws Exception {
        int expected = 108;
        int num;

        num = FormulaUtils.totalNumberDaysInEx( materialList,
budgetBase);
        Assert.assertEquals(expected, num );
    }

    @Test /*A16 ;*/
    public void testTotalNumberMenInEx() throws Exception {
        int expected = 2;
        int num;

        num = FormulaUtils.totalNumberMenInEx( materialList,
budgetBase, base);
        Assert.assertEquals(expected, num );
    }

    @Test /*A17 Arredondar para cima ;*/
    public void testTotalNumberTeamsComposed() throws Exception {
        int expected = 1;
        int num;

        num = FormulaUtils.totalNumberTeamsComposed( materialList,
budgetBase, base);
        Assert.assertEquals(expected, num );
    }

    @Test /*A18 ;*/
    public void testTotalCostsEquipmentDestination() throws
Exception {
        double expected = 47518.15;
        double num;

        num = FormulaUtils.totalCostFreeOnBoards(materialList,
budgetBase, proposal);
        Assert.assertEquals(expected, num, 0.01);
    }
}
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
}

@Test /*A19 ;*/
public void testCostInsuranceContainers() throws Exception {
    double expected = 3.49 ;
    double num;

    num = FormulaUtils.costInsuranceContainers( materialList,
budgetBase, proposal);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A20 ;*/
public void testCostDerivedSeaFreight() throws Exception {
    double expected = 14973;
    double num;

    num = FormulaUtils.costDerivedSeaFreight( budgetBase,
proposal, containers);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A21 ;*/
public void testCostInternalTransportDestination() throws
Exception {
    double expected = 5175;
    double num;

    num = FormulaUtils.costInternalTransportDestination(
containers, budgetBase);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A22 ;*/
public void testSumElementsReferencedContainers() throws
Exception {
    double expected = 67669.64;
    double num;

    num = FormulaUtils.sumElementsReferencedContainers(
materialList, budgetBase, proposal, containers);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A23 ;*/
public void testCostMaritimeFreightPercentage() throws
Exception {
    double expected = 31.51;
    double num;

    num = FormulaUtils.costMaritimeFreightPercentage(
materialList, budgetBase, proposal, containers);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A24 ;*/
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
public void testCostInternalTransportDestinationPercentage()
throws Exception {
    double expected = 10.89;
    double num;

    num =
FormulaUtils.costInternalTransportDestinationPercentage(
materialList, budgetBase, proposal, containers);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A27 ;*/
public void testMountPriceEquipment() throws Exception {
    double expected = 7.50;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.mountPriceEquipment( material,
budgetBase );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A28 ;*/
public void testCostPricesEquipment() throws Exception {
    double expected = 1.11;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.costPricesEquipment( material );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A29 ;*/
public void testMarginEquipmentSource() throws Exception {
    double expected = 0.17;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.marginEquipmentSource( material,
budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A30 ;*/
public void testSalePriceEquipmentSource() throws Exception {
    double expected = 1.28;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.salePriceEquipmentSource( material,
budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
@Test /*A31 ;*/
public void testCostMaterialDestinationCountry() throws
Exception {
    double expected = 1.79;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.costMaterialDestinationCountry(
material, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A32 ;*/
public void testCostHandLaborDestinationCountry() throws
Exception {
    double expected = 10.50;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.costHandLaborDestinationCountry(
material, budgetBase );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A33 ;*/
public void testCostPriceEquipmentDestinationCountry() throws
Exception {
    double expected = 12.2871;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.costPriceEquipmentDestinationCountry(
material, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A34 ;*/
public void testValueForCustomsClearance() throws Exception {
    double expected = 0.19;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.valueForCustomsClearance( material,
budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A35 ;*/
public void testValueForRights() throws Exception {
    double expected = 0;
    double num;
    Material material;
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
        material = materialList.get(2);
        num = FormulaUtils.valueForRights( material, budgetBase,
proposal );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A36 ;*/
    public void testValueForIRT_Retention() throws Exception {
        double expected = 0.43;
        double num;
        Material material;

        material = materialList.get(2);
        num = FormulaUtils.valueForIRT_Retention( material,
budgetBase, proposal );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A37 ;*/
    public void testValueForStampDuty() throws Exception {
        double expected = 0.12;
        double num;
        Material material;

        material = materialList.get(2);
        num = FormulaUtils.valueForStampDuty( material, budgetBase,
proposal );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A38 ;*/
    public void testCostTotalPriceDestinationCountry() throws
Exception {
        double expected = 2.06;
        double num;
        Material material;

        material = materialList.get(2);
        num = FormulaUtils.costTotalPriceDestinationCountry(
material, budgetBase, proposal );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A39 ;*/
    public void testCostTotalPriceHandLaborDestinationCountry()
throws Exception {
        double expected = 10.97;
        double num;
        Material material;

        material = materialList.get(2);
        num =
FormulaUtils.costTotalPriceHandLaborDestinationCountry( material,
budgetBase );
        Assert.assertEquals(expected, num, 0.01);
    }
}
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
@Test /*A40 ;*/
public void testTotalCostPriceDestinationCountry() throws
Exception {
    double expected = 13.03;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.totalCostPriceDestinationCountry(
material, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A41 ;*/
public void testMarginEquipmentDestinationCountry() throws
Exception {
    double expected = 0.51;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.marginEquipmentDestinationCountry(
material, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A42 ;*/
public void testMarginHandLaborDestinationCountry() throws
Exception {
    double expected = 2.74;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.marginHandLaborDestinationCountry(
material, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A43 ;*/
public void testMarginTotalDestinationCountry() throws
Exception {
    double expected = 3.26;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.marginTotalDestinationCountry( material,
budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A44 ;*/
public void testTotalSalePriceEquipmentDestinationCountry()
throws Exception {
    double expected = 2.57;
```



## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
double num;
Material material;

material = materialList.get(2);
num =
FormulaUtils.totalSalePriceEquipmentDestinationCountry( material,
budgetBase, proposal );
Assert.assertEquals(expected, num, 0.01);
}

@Test /*A45 */
public void testTotalSalePriceHandLaborDestinationCountry()
throws Exception {
double expected = 13.72;
double num;
Material material;

material = materialList.get(2);
num =
FormulaUtils.totalSalePriceHandLaborDestinationCountry( material,
budgetBase, proposal );
Assert.assertEquals(expected, num, 0.01);
}

@Test /*A46 */
public void testTotalSaleDestinationCountry() throws Exception
{
double expected = 16.28;
double num;
Material material;

material = materialList.get(2);
num = FormulaUtils.totalSaleDestinationCountry( material,
budgetBase, proposal );
Assert.assertEquals(expected, num, 0.01);
}

@Test /*A47 */
public void testItemSalePrice() throws Exception {
double expected = 162.85;
double num;
Material material;

material = materialList.get(2);
num = FormulaUtils.itemSalePrice( material, budgetBase,
proposal);
Assert.assertEquals(expected, num, 0.01);
}

@Test /*A48 */
public void testAmountSpentOtherCosts() throws Exception {
double expected = 16.28;
double num;
Material material;

material = materialList.get(2);
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
        num = FormulaUtils.AmountSpentOtherCosts( material,
budgetBase, proposal );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A49 ;*/
    public void testItemSpentOtherCosts() throws Exception {
        double expected = 162.85;
        double num;
        Material material;

        material = materialList.get(2);
        num = FormulaUtils.itemSpentOtherCosts( material,
budgetBase, proposal);
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A50 ;*/
    public void testItemSpentTransportInsurance() throws Exception
    {
        double expected = 0.31;
        double num;
        Material material;

        material = materialList.get(2);
        num = FormulaUtils.itemSpentTransportInsurance( material,
budgetBase, proposal);
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A51 ;*/
    public void testItemSpentTransportSeaFreight() throws Exception
    {
        double expected = 51.31;
        double num;
        Material material;

        material = materialList.get(2);
        num =
FormulaUtils.itemSpentTransportSeaFreight(materialList, material,
budgetBase, proposal, containers);
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A52 ;*/
    public void testItemSpentInternalTransportDestinationCountry()
throws Exception {
        double expected = 17.73;
        double num;
        Material material;

        material = materialList.get(2);
        num =
FormulaUtils.itemSpentInternalTransportDestinationCountry(materialL
ist, material, budgetBase, proposal, containers);
        Assert.assertEquals(expected, num, 0.01);
    }
}
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
@Test /*A53 ;*/
public void
testDoublesalePriceMaterialsAndTransportDestinationCountry() throws
Exception {
    double expected = 9.50;
    double num;
    Material material;

    material = materialList.get(2);
    num =
FormulaUtils.doublesalePriceMaterialsAndTransportDestinationCountry
(materialList, material, budgetBase, proposal, containers);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A54 ;*/
public void testHandLaborAndShippingPriceDestinationCountry()
throws Exception {
    double expected = 13.72;
    double num;
    Material material;

    material = materialList.get(2);
    num =
FormulaUtils.handLaborAndShippingPriceDestinationCountry( material,
budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A55 ;*/
public void
testTotalSalePricePlusTransportDestinationCountryUnitary() throws
Exception {
    double expected = 23.22;
    double num;
    Material material;

    material = materialList.get(2);
    num =
FormulaUtils.TotalSalePricePlusTransportDestinationCountryUnitary(
materialList, material, budgetBase, proposal, containers);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A56 ;*/
public void testTotalSalePricePlusTransportDestinationCountry()
throws Exception {
    double expected = 232.2;
    double num;
    Material material;

    material = materialList.get(2);
    num =
FormulaUtils.TotalSalePricePlusTransportDestinationCountry(
materialList, material, budgetBase, proposal, containers);
    Assert.assertEquals(expected, num, 0.01);
}
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
}

@Test /*A57 ;*/
public void testCostPriceLabor1() throws Exception {
    double expected = 0;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.costPriceLabor1( material, budgetBase
);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A58 ;*/
public void testCostPriceLabor2() throws Exception {
    double expected = 0;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.costPriceLabor2( material, budgetBase
);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A59 ;*/
public void testCostPriceLabor3() throws Exception {
    double expected = 7.50;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.costPriceLabor3( material, budgetBase
);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A60 ;*/
public void testSumPricesEequipment() throws Exception {
    double expected = 77.20;
    double num;

    num = FormulaUtils.sumPricesEequipment( materialList);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A61 ;*/
public void testSumPricesHandLabor() throws Exception {
    double expected = 25825;
    double num;

    num = FormulaUtils.sumPricesHandLabor( materialList,
budgetBase);
    Assert.assertEquals(expected, num, 0.01);
}
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
@Test /*A62 ;*/
public void testSumPricesEquipmentAccessoris() throws
Exception {
    double expected = 48.40;
    double num;

    num = FormulaUtils.sumPricesEquipmentAccessoris(
materialList);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A63 ;*/
public void testSumPricesEquipmentTransportation() throws
Exception {
    double expected = 0;
    double num;

    num = FormulaUtils.sumPricesEquipmentTransportation(
materialList);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A64 ;*/
public void testSumPricesCostEquipment() throws Exception {
    double expected = 125.60;
    double num;

    num = FormulaUtils.sumPricesCostEquipment( materialList);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A65 ;*/
public void testSumPricesMarginEquipmentSource() throws
Exception {
    double expected = 18.84;
    double num;

    num = FormulaUtils.sumPricesMarginEquipmentSource(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01 );
}

@Test /*A66 ;*/
public void testTotalSellingPricesSource() throws Exception {
    double expected = 144.44 ;
    double num;

    num = FormulaUtils.totalSellingPricesSource( materialList,
budgetBase, proposal );
    Assert.assertEquals(expected, num , 0.001);
}

@Test /*A67 ;*/
public void testSumCostMaterialDestinationCountry() throws
Exception {
    double expected = 202.22;
    double num;
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
        num = FormulaUtils.sumCostMaterialDestinationCountry(
materialList, budgetBase, proposal );
        Assert.assertEquals(expected, num, 0.01 );
    }

    @Test /*A68 ;*/
    public void testSumHandLaborDestinationCountry() throws
Exception {
        double expected = 36155 ;
        double num;

        num = FormulaUtils.sumHandLaborDestinationCountry(
materialList, budgetBase);
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A69 ;*/
    public void testSumCostPriceEquipmentDestinationCountry()
throws Exception {
        double expected = 36357.22;
        double num;

        num = FormulaUtils.sumCostPriceEquipmentDestinationCountry(
materialList, budgetBase, proposal );
        Assert.assertEquals(expected, num, 0.01 );
    }

    @Test /*A70 ;*/
    public void testSumPricesEequipmentDestination() throws
Exception {
        double expected = 0;
        double num;

        num = FormulaUtils.sumPricesEequipmentDestination(
materialList );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A71 ;*/
    public void testSumValueForCustomsClearance() throws Exception
{
        double expected = 21.23;
        double num;

        num = FormulaUtils.sumValueForCustomsClearance(
materialList, budgetBase, proposal );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A72 ;*/
    public void testSumValueForRights() throws Exception {
        double expected = 0;
        double num;

        num = FormulaUtils.sumValueForRights( materialList,
budgetBase, proposal );
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A73 ;*/
    public void testSumValueForIRT_Retention() throws Exception {
        double expected = 1272.50;
        double num;

        num = FormulaUtils.sumValueForIRT_Retention( materialList,
budgetBase, proposal );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A74 ;*/
    public void testSumValueForStampDuty() throws Exception {
        double expected = 363.57;
        double num;

        num = FormulaUtils.sumValueForStampDuty( materialList,
budgetBase, proposal );
        Assert.assertEquals(expected, num, 0.01 );
    }

    @Test /*A75 ;*/
    public void testSumCostTotalPriceDestinationCountry() throws
Exception {
        double expected = 232.55;
        double num;

        num = FormulaUtils.sumCostTotalPriceDestinationCountry(
materialList, budgetBase, proposal );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A76 ;*/
    public void testSumCostTotalPriceHandLaborDestinationCountry()
throws Exception {
        double expected = 37781.98;
        double num;

        num =
FormulaUtils.sumCostTotalPriceHandLaborDestinationCountry(
materialList, budgetBase );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A77 ;*/
    public void testSumTotalCostPriceDestinationCountry() throws
Exception {
        double expected = 38014.52;
        double num;

        num = FormulaUtils.sumTotalCostPriceDestinationCountry(
materialList, budgetBase, proposal );
        Assert.assertEquals(expected, num, 0.01);
    }
}
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
@Test /*A78 ;*/
public void testSumMarginEquipmentDestinationCountry() throws
Exception {
    double expected = 58.14;
    double num;

    num = FormulaUtils.sumMarginEquipmentDestinationCountry(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A79 ;*/
public void testSumMarginHandLaborDestinationCountry() throws
Exception {
    double expected = 9445.49;
    double num;

    num = FormulaUtils.sumMarginHandLaborDestinationCountry(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A80 ;*/
public void testSumMarginTotalDestinationCountry() throws
Exception {
    double expected = 9503.63;
    double num;

    num = FormulaUtils.sumMarginTotalDestinationCountry(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A81 ;*/
public void testSumTotalSalePriceEquipmentDestinationCountry()
throws Exception {
    double expected = 290.69;
    double num;

    num =
FormulaUtils.sumTotalSalePriceEquipmentDestinationCountry(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A82 ;*/
public void testSumTotalSalePriceHandLaborDestinationCountry()
throws Exception {
    double expected = 47227.47;
    double num;

    num =
FormulaUtils.sumTotalSalePriceHandLaborDestinationCountry(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}
```



## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
@Test /*A83 ;*/
public void testSumTotalSaleDestinationCountry() throws
Exception {
    double expected = 47518.15 ;
    double num;

    num = FormulaUtils.sumTotalSaleDestinationCountry(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A84 ;*/
public void testTotalSalesPricesEquipmentDestination() throws
Exception {
    double expected = 47518.15;
    double num;

    num = FormulaUtils.totalSalesPricesEquipmentDestination(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A85 ;*/
public void testSumAmountSpentOtherCosts() throws Exception {
    double expected = 947.68;
    double num;

    num = FormulaUtils.sumAmountSpentOtherCosts( materialList,
budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A86 ;*/
public void testSumItemSpentOtherCosts() throws Exception {
    double expected = 47518.15;
    double num;

    num = FormulaUtils.sumItemSpentOtherCosts( materialList,
budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A87 ;*/
public void testSumItemSpentTransportInsurance() throws
Exception {
    double expected = 3.49;
    double num;

    num = FormulaUtils.sumItemSpentTransportInsurance(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A88 ;*/
public void testSumItemSpentTransportSeaFreight() throws
Exception {
    double expected = 14973;
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
        double num;

        num = FormulaUtils.sumItemSpentTransportSeaFreight (
materialList, budgetBase, proposal, containers );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A89 ;*/
    public void
testSumItemSpentInternalTransportDestinationCountry() throws
Exception {
        double expected = 5175;
        double num;

        num =
FormulaUtils.sumItemSpentInternalTransportDestinationCountry(materi
allList, budgetBase, proposal, containers);
        Assert.assertEquals(expected, num, 0.01 );
    }

    @Test /*A90 ;*/
    public void
testSumDoublesalePriceMaterialsAndTransportDestinationCountry()
throws Exception {
        double expected = 20442.17;
        double num;

        num =
FormulaUtils.sumDoublesalePriceMaterialsAndTransportDestinationCoun
try( materialList, budgetBase, proposal, containers );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A91 ;*/
    public void
testSumHandLaborAndShippingPriceDestinationCountry() throws
Exception {
        double expected = 47227.47;
        double num;

        num =
FormulaUtils.sumHandLaborAndShippingPriceDestinationCountry(
materialList, budgetBase, proposal );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A92 ;*/
    public void
testSumTotalSalePricePlusTransportDestinationCountryUnitary()
throws Exception {
        double expected = 67669.64;
        double num;

        num =
FormulaUtils.sumTotalSalePricePlusTransportDestinationCountryUnitar
y( materialList, budgetBase, proposal, containers );
        Assert.assertEquals(expected, num, 0.01);
    }
}
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
}

@Test /*A94 ;*/
public void testTotalAssemblyTimeType1() throws Exception {
    double expected = 833.33;
    double num;

    num = FormulaUtils.totalAssemblyTimeType1(materialList );
    Assert.assertEquals(expected , num, 0.004);
}

@Test /*A95 ;*/
public void testSumCostPriceLabor1() throws Exception {
    double expected = 25000;
    double num;

    num = FormulaUtils.sumCostPriceLabor1( materialList,
budgetBase );
    Assert.assertEquals(expected, num , 0);
}

@Test /*A96 ;*/
public void testTotalAssemblyTimeType2() throws Exception {
    double expected = 25;
    double num;

    num = FormulaUtils.totalAssemblyTimeType2( materialList );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A97 ;*/
public void testSumcostPriceLabor2() throws Exception {
    double expected = 750;
    double num;

    num = FormulaUtils.sumcostPriceLabor2( materialList,
budgetBase );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A98 ;*/
public void testTotalAssemblyTimeType3() throws Exception {
    double expected = 2.50;
    double num;

    num = FormulaUtils.totalAssemblyTimeType3( materialList);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A99 ;*/
public void testSumcostPriceLabor3() throws Exception {
    double expected = 75;
    double num;

    num = FormulaUtils.sumcostPriceLabor3( materialList,
budgetBase );
    Assert.assertEquals(expected, num, 0.01);
}
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
    }

    @Test /*A100 ;*/
    public void testTotalSalePriceCountryOrigin() throws Exception
    {
        double expected = 10841.93;
        double num;
        Material material;

        material = materialList.get(0);
        num = FormulaUtils.totalSalePriceCountryOrigin(
materialList, material, budgetBase, proposal, containers );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A101 ;*/
    public void testTotalSalePrice_CIF_DestinationCountry() throws
Exception {
        double expected = 67669.64;
        double num;

        num = FormulaUtils.totalSalePrice_CIF_DestinationCountry(
materialList, budgetBase, proposal, containers );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A102 ;*/
    public void testTotalSalePrice_FOB_DestinationCountry() throws
Exception {
        double expected = 47518.15;
        double num;

        num = FormulaUtils.totalSalePrice_FOB_DestinationCountry(
materialList, budgetBase, proposal );
        Assert.assertEquals(expected, num, 0.01);
    }

    @Test /*A103 ;*/
    public void testTotalCostEquipmentCountryOrigin() throws
Exception {
        double expected = 125.60;
        double num;

        num = FormulaUtils.totalCostEquipmentCountryOrigin(
materialList );
        Assert.assertEquals(expected, num, 0.01 );
    }

    @Test /*A104 ;*/
    public void testTotalHandLaborDestinationCountry() throws
Exception {
        double expected = 36155;
        double num;

        num = FormulaUtils.totalHandLaborDestinationCountry(
materialList, budgetBase );
        Assert.assertEquals(expected, num, 0.01);
    }
}
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
}

@Test /*A105 ;*/
public void testTotalMarginCountryOrigin() throws Exception {
    double expected = 18.84;
    double num;

    num = FormulaUtils.totalMarginCountryOrigin( materialList,
budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01 );
}

@Test /*A106 ;*/
public void testTotalMarginDestinationCountry() throws
Exception {
    double expected = 9503.63;
    double num;

    num = FormulaUtils.totalMarginDestinationCountry(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A107 ;*/
public void testTotalValue_IRT_Retention() throws Exception {
    double expected = 1272.50;
    double num;

    num = FormulaUtils.totalValue_IRT_Retention( materialList,
budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01 );
}

@Test /*A108 ;*/
public void testTotalValueStampDutyRetention() throws Exception
{
    double expected = 363.57;
    double num;

    num = FormulaUtils.totalValueStampDutyRetention(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A109 ;*/
public void testTotalValueOfTheRights() throws Exception {
    double expected = 0;
    double num;

    num = FormulaUtils.totalValueOfTheRights( materialList,
budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A110 ;*/
public void testTotalPriceContainerDestinationCountry() throws
Exception {
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
double expected = 13020;
double num;

    num = FormulaUtils.totalPriceContainerDestinationCountry(
budgetBase, containers );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A111 ;*/
public void
testTotalValueInternalTransportContainerDestinationCountry() throws
Exception {
    double expected = 5175.0;
    double num;

    num =
FormulaUtils.totalValueInternalTransportContainerDestinationCountry
( budgetBase, containers );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A112 ;*/
public void testTotalValueForCustomsClearance() throws
Exception {
    double expected = 21.23;
    double num;

    num = FormulaUtils.totalValueForCustomsClearance(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A113 ;*/
public void
testTotalPriceInsuranceContainersDestinationCountry() throws
Exception {
    double expected = 3.49;
    double num;

    num =
FormulaUtils.totalPriceInsuranceContainersDestinationCountry(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A114 ;*/
public void testTotalCostEquipmentCountryOriginPercentage()
throws Exception {
    double expected = 0.48;
    double num;

    num =
FormulaUtils.totalCostEquipmentCountryOriginPercentage(
materialList, budgetBase );
    Assert.assertEquals(expected, num, 0.01);
}
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
@Test /*A115 ;*/
public void testTotalCostHandLaborCountryOriginPercentage()
throws Exception {
    double expected = 99.52;
    double num;

    num =
FormulaUtils.totalCostHandLaborCountryOriginPercentage(
materialList, budgetBase );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A116 ;*/
public void testTotalMarginCountryOriginPercentage() throws
Exception {
    double expected = 15;
    double num;

    num = FormulaUtils.totalMarginCountryOriginPercentage(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A117 ;*/
public void testTotalMarginDestinationCountryPercentage()
throws Exception {
    double expected = 25;
    double num;

    num = FormulaUtils.totalMarginDestinationCountryPercentage(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01 );
}

@Test /*A118 ;*/
public void testFullAmountOfOceanFreight() throws Exception {
    double expected = 20148;
    double num;

    num = FormulaUtils.fullAmountOfOceanFreight( budgetBase,
proposal, containers );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A119 ;*/
public void testFinalValueOnTheDestinationCountry() throws
Exception {
    double expected = 67669.64;
    double num;

    num = FormulaUtils.finalValueOnTheDestinationCountry(
materialList, budgetBase, proposal, containers );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A120 !Atenção valor variável! ;*/
```

## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
public void testPartialPriceEquipmentOriginCountry() throws
Exception {
    double expected = 12.77;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.partialPriceEquipmentOriginCountry(
material, budgetBase, proposal);
    Assert.assertEquals(expected, num, 0.005 );
}

@Test /*A121 ;*/
public void testInsuranceValueOriginCountry() throws Exception
{
    double expected = 2.49;
    double num;

    num = FormulaUtils.insuranceValueOriginCountry(
materialList, budgetBase, proposal );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A122 ;*/
public void testValueofocanfreightOriginCountry() throws
Exception {
    double expected = 10695;
    double num;

    num = FormulaUtils.valueofocanfreightOriginCountry(
budgetBase, proposal, containers );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A123 ;*/
public void testTotalAmountProFormaNumber5() throws Exception {
    double expected = 10841.93;
    double num;

    num = FormulaUtils.totalAmountProFormaNumber5 (
materialList, budgetBase, proposal, containers );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A124 !Atenção valor variável! ;*/
public void testUnitPriceAssembly() throws Exception {
    double expected = 7.50;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.UnitPriceAssembly( material, budgetBase
);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A125 !Atenção valor variável! ;*/
```



## Apêndice nº4 – Teste de assertividade

### 4.2 – T Programa de teste aos inputs

```
public void testPartialPriceAssembly() throws Exception {
    double expected = 75;
    double num;
    Material material;

    material = materialList.get(2);
    num = FormulaUtils.partialPriceAssembly( material,
budgetBase);
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A126 */
public void testSumPartialPriceAssembly() throws Exception {
    double expected = 25825;
    double num;

    num = FormulaUtils.sumPartialPriceAssembly( materialList,
budgetBase );
    Assert.assertEquals(expected, num, 0.01);
}

@Test /*A127 */
public void testTotalAmountProFormaNumber7() throws Exception {
    double expected = 25969.44;
    double num;

    num = FormulaUtils.totalAmountProFormaNumber7(
materialList, budgetBase, proposal);
    Assert.assertEquals(expected, num, 0.01);
}
}
```