

# Projecto *eMudar*

Critical Software | iTGROW

Autor:  
**Ricardo Quirino**  
ricardojvquirino@gmail.com

Orientador:  
**Prof. Bruno Cabral**

Porto, 07/10/2016





# Índice

|   |           |
|---|-----------|
| <b>1. INTRODUÇÃO</b>                                      | <b>1</b>  |
| <b>2. PROJECTO <i>EMUDAR</i></b>                          | <b>3</b>  |
| 2.1 ENQUADRAMENTO   | 3         |
| 2.2 <i>STAKEHOLDERS</i>                                   | 3         |
| 2.2.1 <i>Equipa</i>                                       | 3         |
| 2.3 FUNCIONALIDADES                                       | 5         |
| 2.4 ESTRUTURA   | 5         |
| <b>3. ESTÁGIO</b>   | <b>9</b>  |
| 3.1 OBJECTIVOS  | 9         |
| 3.2 TECNOLOGIAS UTILIZADAS                                | 9         |
| 3.2.1 <i>Java</i>   | 9         |
| 3.2.2 <i>IBM BPM</i>                                      | 10        |
| 3.2.3 <i>IBM WebSphere Application Server</i>             | 10        |
| 3.2.4 <i>FileNet</i>                                      | 10        |
| 3.2.5 <i>Proxy</i>  | 10        |
| 3.3 TRABALHOS DESENVOLVIDOS                               | 11        |
| 3.3.1 <i>Ramp-up</i>                                      | 11        |
| 3.3.2 <i>Desenvolvimento</i>                              | 14        |
| 3.3.2.1 Apresentação e Análise Funcional                  | 14        |
| 3.3.2.2 Análise Detalha dos Requisitos                    | 14        |
| 3.3.2.3 Estimativas                                       | 14        |
| 3.3.2.4 Desenvolvimento                                   | 15        |
| 3.3.2.5 Conclusão   | 15        |
| 3.3.3 <i>Módulos específicos desenvolvidos</i>            | 15        |
| 3.3.3.1 Processo Caixas - Módulo Bloco A                  | 15        |
| 3.3.3.2 Processo Caixas – Módulo Bloco C                  | 20        |
| 3.3.3.3 Configuração Automática de Limites de Autorização | 24        |
| <b>4. CONCLUSÃO</b>                                       | <b>25</b> |
| <b>REFERÊNCIAS</b>  | <b>26</b> |



## 1. Introdução

Este relatório pretende descrever o meu recente percurso numa nova área profissional: a área das tecnologias de informação.

Em 2014, agastado pelo estado da minha situação profissional, decidi arriscar com um novo programa de formação, da responsabilidade conjunta da Critical Software, Banco BPI e Universidade de Coimbra: Acertar o Rumo.

Depois de, praticamente, um ano de formação académica intensiva, iniciei um estágio profissional de 12 meses na empresa Critical Software.

O primeiro projecto no qual estive envolvido foi na área do *business intelligence*. Os meus conhecimentos sobre a área eram escassos, mas depois de integrado no projecto, pude constatar que é sem dúvida uma actividade com um futuro promissor no contexto empresarial. No entanto, e porque queria acima de tudo consolidar tudo o que havia aprendido ao longo do ano de formação em Coimbra, verifiquei que não era, no momento, a melhor opção, pois não iria conseguir trabalhar com as ferramentas e tecnologias que pretendia.

Com a ajuda dos elementos responsáveis, quer na Critical Software quer na iTGroW, tive a oportunidade de abraçar um novo projecto: *eMudar*.

Neste relatório procuro, de forma sucinta, reunir todos os pormenores do estágio que se iniciou no dia 26 de Outubro de 2015.

Num primeiro capítulo é feito o enquadramento do projecto, com referência às entidades e pessoas envolvidas, nomeadamente o cliente e a equipa de desenvolvimento.

O capítulo seguinte (Estágio) descreve o processo de estágio, com a descrição das tecnologias utilizadas e os trabalhos em que estive envolvido directamente.

No último capítulo, *Conclusão*, é feita uma retrospectiva sobre todo este processo de aprendizagem e inserção no mercado de trabalho, desde a formação académica até ao estágio.



## 2. Projecto *eMudar*

### 2.1 Enquadramento

O objectivo do projecto *eMudar* é a criação de um sistema de *front-end* que permita integrar e uniformizar o acesso aos principais sistemas disponibilizados nos balcões e *back-office*.

Este novo sistema assenta numa base já existente, o *BANKA*, a ferramenta utilizada para realizar as operações do banco, como a gestão da economia, documental e processual dos seus clientes.

Esta ferramenta suporta, na sua maioria, as operações supracitadas, embora não seja apropriada para a realização de processos complexos que ocorrem diariamente neste tipo de entidades, pois permite apenas a execução de tarefas simples e atómicas, não existindo um suporte à automatização, controlo e monitorização de processos, alguns dos quais com múltiplos intervenientes. Outro problema é a complexidade da interface de utilização e a dispersão da informação por múltiplos ecrãs.

O novo sistema tira partido de todas as operações já definidas no *BANKA* para, com recurso às novas tecnologias, fornecer um serviço mais adequado ao cliente nas suas actividades de *front-office* e *back-office*. Os dois sistemas são, por isso, complementares.

### 2.2 Stakeholders

Este sistema é desenvolvido e mantido pela Critical Software (CSW), por uma equipa do pelotão *Machimbombo*, dividida entre Porto e Coimbra. Esta equipa integra, actualmente, mais de 30 pessoas: programadores, *testers*, analistas funcionais e gestores.

O cliente é o Banco de Fomento Angola, com actividade no sector bancário. Esta entidade conta com:

- Perto de 1.500.000 clientes;
- 191 balcões;

Ocupando, actualmente, a primeira posição entre os vários bancos privados em Angola, mercado onde opera há 22 anos. [1]

#### 2.2.1 Equipa

Como foi referido, a equipa do projecto *eMudar* é constituída por cerca de 30 elementos distribuídos, maioritariamente, por Coimbra e Porto.

A distribuição é a seguinte:

| Posição                                     | Elementos |
|---|-----------|
| <i>Suporte</i>                              | 7         |
| <i>Infra-estrutura e Gestão de Releases</i> | 2         |
| <i>Desenvolvimento (Porto)</i>              | 9         |
| <i>Desenvolvimento (Coimbra)</i>            | 4         |
| <i>SPAE<sup>1</sup></i>                     | 4         |
| <i>Análise Funcional e Requisitos</i>       | 4         |
| <i>Gestão</i>                               | 4         |
| <b>Total</b>                                | <b>34</b> |

Tabela 1 - Distribuição de elementos da equipa

O organograma dos responsáveis do projecto é o seguinte:



Figura 1 - Organograma da equipa (responsáveis)

A equipa com que interagi directamente é a equipa que se encontra no Porto. Esta equipa é constituída por:

- 9 elementos na parte de desenvolvimento (incluindo o *team leader*)
- 1 elemento que ocupa a posição de *technical manager* do projecto
- 2 SPAEs
- 2 elementos na equipa de análise funcional e requisitos
- 1 elemento que ocupa a posição de assistente do gestor de projecto

<sup>1</sup> SPAE – Software Product Assurance Engineer é a pessoa responsável por garantir a qualidade do *software* desenvolvido, nomeadamente através da realização de testes



## 2.3 Funcionalidades

O sistema criado e desenvolvido permite, essencialmente, a execução de processos de negócio associados à actividade do cliente, como:

- Transferências
- Depósitos
- Constituição de conta
- Cheques e Cartões
- Etc..

Inerentes a estes processos estão tarefas como movimentações de divisas, associação e gestão documental, aprovações ou anulações desses mesmos processos, bem como tarefas de operador:

- Abertura/Fecho de Caixa
- Consulta de processos
- Consulta de certificações de operações
- Reclamação de tarefas
- Cálculo de operações (Calculadora)

## 2.4 Estrutura

A principal tecnologia de desenvolvimento do *software* é JAVA, sobre a plataforma JAVA EE (*Enterprise Edition*). Esta plataforma é usada para a criação de aplicações, maioritariamente empresariais, baseadas em serviços pois permite o desenvolvimento sólido e sustentado recorrendo a ferramentas e *APIs*, como:

- *Java Servlets*
- *Enterprise Javabeans Components (EJB)*
- *Java Persistence API (JPA)*
- *Java API for XML Binding (JAX-B)*
- *Java Message Service (JMS)*
- Etc...

Aplicações desenvolvidas sobre esta plataforma recorrem aos chamados servidores aplicativos, para que possam ser implementadas. No caso específico deste projecto, o servidor utilizado é o *IBM WebSphere Application Server*.

Este tipo de aplicações são desenvolvidas com base numa arquitectura de três camadas:

- Apresentação
- Negócio
- Dados

A divisão em camadas tem o propósito de dar ao sistema maior flexibilidade, ao tornar independentes cada um dos três principais componentes que constituem uma aplicação *web*. Esta separação permite que a aplicação seja escalável pelo simples facto de que é, assim, possível alterar/substituir qualquer um dos elementos sem prejuízo do sistema global, ou seja, possibilita um maior controlo no crescimento da aplicação. Este tipo arquitectura assenta num sistema modelo cliente-servidor. A camada de apresentação encontra-se disponível do lado do cliente, aquele que utiliza a aplicação, e a comunicação com as restantes camadas é feita através de pedidos executados ao servidor. A camada de apresentação é responsável pelo ambiente gráfico mostrado ao cliente. A camada de negócio está incumbida de executar de todas as regras de negócio (lógica) da aplicação e, finalmente, a camada de dados serve como o repositório de informação da aplicação.

A arquitectura da aplicação é também caracterizada por ser baseada em serviços. Este tipo de arquitectura permite a integração de diferentes sistemas, representando esses sistemas como serviços, que podem ser utilizados por diferentes aplicações.

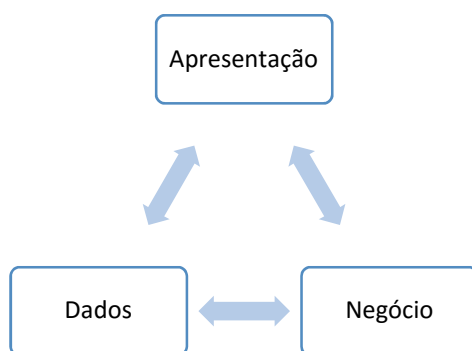


Figura 2 - Divisão em Camadas

A aplicação é composta por vários componentes. Um deles, e que representa a camada de apresentação, é denominado por *FrontEnd* (FE) no projecto, e que se traduz na interface em contacto com o utilizador, permitindo que este interaja com o sistema, executando as suas funcionalidades.

O *FrontEnd* tem integração com outros componentes. Um desses componentes é motor *Business Process Manager*. Este motor traduz-se numa forma estruturada de representação de tarefas e interações que constituem um processo de uma organização. Estes processos incluem, na sua maioria, diversos intervenientes, e o recurso a um motor deste tipo permite assegurar que o processo segue as regras definidas pela entidade, bem como a sua monitorização e consulta do seu estado. É possível definir nestes processos todas as interações com os utilizadores e sistemas existentes, e regras de controlo.

Outro componente com integração com a camada *FrontEnd* é o repositório de documentos. Este repositório suporta a circulação electrónica dos documentos específicos dos processos, e está disponível para todos os sistemas que necessitem de interagir com documentos, como por exemplo, o processo de abertura de uma nova conta no banco, que exige, no seu decurso, que o utilizador associe os documentos necessários (como comprovativos de identificação da entidade) ao processo, documentos esses que são mais tarde validados, em *back-office*. Portanto para além da associação de documentos é possível também realizar a validação dos mesmos. A associação e validação são feitas no mesmo processo, embora por intervenientes diferentes, representando uma das vantagens da implementação deste novo sistema, combatendo a deficiência já apresentada ao *BANKA*.

O repositório de documentos está dividido em duas partes:

- Repositório Local
- Repositório Central

O repositório local é um repositório de documentos presente em cada uma das agências bancárias. É o primeiro local para onde é enviado o documento, depois de associado ao processo pelo utilizador. Por se encontrar no mesmo local onde é realizada a associação, o processo de *upload* é muito rápido e permite que a tarefa seja desempenhada de forma mais eficiente, situação válida também para a consulta de documentos. Este documento é depois enviado para um repositório central, agregador de todos os documentos. Este envio é feito através de um *proxy*<sup>2</sup> existente em cada uma das agências. Este *proxy* permite que a operação que engloba o *upload* do documento para o repositório local e seu envio para o repositório central seja feita de forma assíncrona, garantindo que o processo não fica pendente de toda esta operação, condicionada ao estado da rede existente.

O sistema electrónico que suporta o repositório documental denomina-se por *FILENET*.

Outro elemento que importa referir é denominado de *Integration (INT)*. Esta camada contempla uma plataforma de serviços, desenvolvida para a exposição da funcionalidade dos sistemas existentes. No fundo esta camada permite, como o seu próprio nome indica, a integração dos componentes no sistema global, através da criação e distribuição de funcionalidades através de serviços.

---

<sup>2</sup> Proxy – servidor que funciona de intermediário para os pedidos do cliente a outros servidores, ou seja, o cliente em vez de executar os pedidos directamente a um servidor, seja para que propósito for, conecta-se a um proxy, e será o proxy, como intermediário do cliente, a executar o pedido do cliente ao servidor final

Finalmente, é importante mencionar também que sistema suporta assinaturas digitais, recorrendo a um repositório de imagens para o efeito, o *Finsigna*. Este componente é utilizado, fundamentalmente, nos processos para a validação de assinaturas.

### 3. Estágio

#### 3.1 Objectivos

O objectivo do estágio foi aplicar e consolidar o conhecimento adquirido ao longo de quase um ano de formação na Universidade de Coimbra, no âmbito das novas tecnologias, e concretamente programação em Java, através de um contexto empresarial.

No fundo, o estágio serviu para dar continuidade aos trabalhos que foram desenvolvidos na formação, aplicando os conceitos adquiridos sobre desenvolvimento de software, redes, bases de dados e sistemas, desta vez num contexto real de trabalho.

#### 3.2 Tecnologias Utilizadas

Este capítulo é dedicado às tecnologias que considerei importante destacar.

##### 3.2.1 Java

A primeira tecnologia que quero destacar é, impreterivelmente, o Java.

Esta foi a linguagem que foi leccionada durante a formação e é o “motor” da aplicação.

Este tipo de aplicações são desenvolvidas sobre a plataforma Java EE (*Enterprise Edition*).

O Java é utilizado nas três camadas que já referi: apresentação, negócio e dados, embora não seja, naturalmente, exclusiva.

Na camada de apresentação é utilizado, fundamentalmente, nos controladores, componentes responsáveis pelos ecrãs apresentados ao utilizador.

A camada de negócio é onde o Java assume um papel mais importante, sendo utilizado quase em toda a sua extensão, desde a implementação das regras de negócio, configurações e servindo como base para a criação e chamada dos serviços.

Na camada de dados, importa destacar o JPA (*Java Persistence API*), uma API<sup>3</sup> responsável pela gestão de dados em plataformas que recorram ao Java, como o Java EE.

---

<sup>3</sup> API – Application Programming Interface, constitui um conjunto de definições, padrões e funções utilizadas na construção de software. No fundo podem ser consideradas bibliotecas a que se recorrem para auxiliar na implementação de funcionalidades em aplicações

### 3.2.2 IBM BPM

O IBM Business Process Manager (BPM) é uma plataforma utilizada para a gestão de processos de negócio (já anteriormente referidos), permitindo a sua concepção, execução monitorização e optimização.

A criação destes processos pode ser feita através de uma ferramenta disponibilizada pela IBM: IBM Process Designer. Com esta ferramenta, o utilizador pode desenhar um processo, definindo tarefas, permissões e acessos ao processo, tarefas, fluxos de processo e integrar o processo com outras tecnologias, através do recurso, por exemplo, a *web services*.

Resumidamente, um processo é composto por um conjunto de tarefas que são executadas de forma sequencial, criando um fluxo, com um início e um fim definidos. Este fluxo pode variar de acordo com as opções definidas dentro do processo, e portanto com resultados variáveis. As tarefas podem ser atribuídas a utilizadores distintos, de acordo, por exemplo, com as suas responsabilidades específicas dentro de um determinado processo.

### 3.2.3 IBM WebSphere Application Server

O servidor aplicational utilizado para a aplicação do projecto é o IBM WebSphere Application Server (*WAS*).

O *WAS* é um produto, que à semelhança do *BPM*, é fornecido pela IBM. Este servidor tem a vantagem de ser o único que é capaz de ser executado nas plataformas mais importantes como Windows, Linux e IBM i. Este último representa, talvez, o factor decisivo pois era o sistema que servia de base para as operações executadas pelo cliente, através da interface *BANKA*, permitindo assim uma mais fácil integração desse sistema com a aplicação *eMudar*.

Durante o meu período de estágio, registou-se a migração do servidor da versão 7.5 para 8.5.

### 3.2.4 FileNet

*FileNet* é uma plataforma que permite a gestão de documentos, numa aplicação. É utilizado no recorrer de processos que obriguem à associação e validação de documentos.

### 3.2.5 Proxy

Por último, destacar a utilização, nas agências de servidores *proxy*. Este tipo de servidores servem como intermediário nos pedidos executados entre outros servidores e clientes.

O seu propósito no âmbito da aplicação *eMudar* é fornecer uma maior fiabilidade ao sistema de envio de documentos para os repositórios digitais, face às condições adversas em termos de comunicações digitais que o sistema está sujeito. Esta tecnologia permite realizar de forma assíncrona esse envio de documentos, permitindo ao utilizador que o executa continuar a executar as suas funções sem que necessite de aguardar pela conclusão do envio do documento da agência para o repositório central. Essa transferência é feita quando possível pelo *proxy*, desde o repositório local para o destino.

### 3.3 Trabalhos desenvolvidos

#### 3.3.1 Ramp-up

O projecto *eMudar* conta, à data, com 5 anos de desenvolvimento e manutenção, sendo portanto um projecto de grande dimensão e complexidade.

Por forma a ajudar a adaptação de novos elementos, especialmente de categoria *junior*, procura-se inserir estes num processo de aprendizagem, denominado por *ramp up*.

Este processo procura assegurar uma evolução gradual do novo elemento dentro do projecto, dando-lhe a conhecer o projecto bem como as tecnologias e metodologias presentes.

No início do estágio tive oportunidade de realizar exercícios e tarefas, inseridas neste processo de aprendizagem, que me permitiram evoluir de forma acelerada, mas sustentada.

Em primeiro lugar, tive oportunidade de conhecer o propósito e funcionamento, a um alto nível, do produto/aplicação, através de vários documentos presentes no repositório do projecto – desde a arquitectura, tecnologias, módulos específicos, etc. Este processo foi complementado com reuniões com o *team leader*, com vista a garantir um processo correcto e bem direccionado.

Ao mesmo tempo, com o auxílio da equipa, preparei as ferramentas necessárias para o desempenho das minhas funções na equipa de desenvolvimento:

- *IDE* - IntelliJ
- Servidor Aplicacional – IBM WebSphere Application Server
- Bases de Dados – DB2
- Sistema de Controlo de Versões - SVN
- IBM Process Designer (ferramenta para a construção de processos)

A ajuda da equipa foi essencial, logo desde início, para a resolução destas primeiras tarefas, dada a minha inexperiência profissional, nesta área.

A segunda parte do processo de *ramp up* foi direccionada para a familiarização de uma das plataformas mais importantes do projecto: IBM Business Process Manager. Este tipo de plataformas não fez parte do programa académico durante a formação, e era por isso um tema totalmente desconhecido.

Nesta fase, tive oportunidade de realizar alguns exercícios práticos na ferramenta IBM Process Designer.



Figura 3 - IBM Process Designer

Esta ferramenta é utilizada na criação e modelação de processos de negócios, parte fundamental da aplicação desenvolvida neste projecto.

A minha formação teve por base o desenvolvimento de um simples processo, desde o *design* até à implementação.

Este processo tem por objectivo calcular a multiplicação de dois valores numéricos introduzidos pelo utilizador num primeiro ecrã, e apresentar o resultado num novo ecrã. Apesar de simples e de pequena dimensão, o processo reúne uma parte considerável dos principais componentes dos processos que são desenvolvido para a aplicação:

- *Workflow*
- Tarefas
- Criação de Ecrãs
- Recurso a *web services* (serviços) do componente de integração (já referido anteriormente)



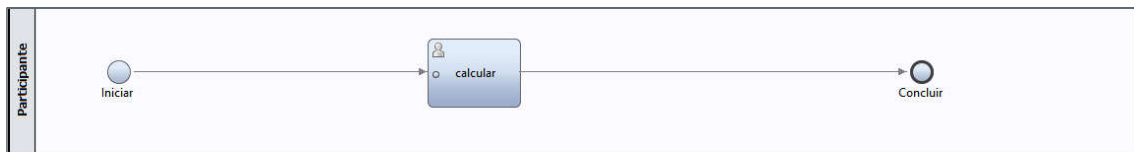


Figura 4 - Diagrama do processo

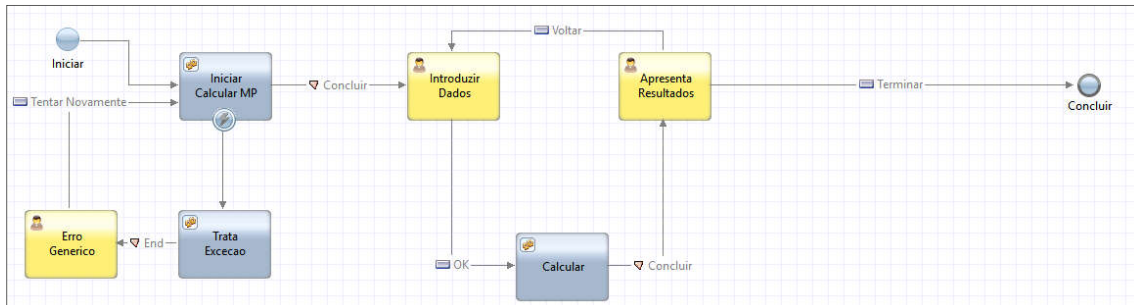


Figura 5 - Tarefa de cálculo da multiplicação

O desenvolvimento dos ecrãs do processo permitiu também estabelecer o primeiro contacto com plataforma ZK. Este passo foi importante pois não possuía, naquele momento, quaisquer conhecimentos sobre a plataforma.

A conclusão do processo de *ramp up* deu-se em duas partes:

- Desenvolvimento de uma aplicação *standalone*<sup>4</sup> como ferramenta de apoio à equipa de *testing* para a clonagem de *issues*<sup>5</sup>

O desenvolvimento desta aplicação permitiu-me aprofundar os conhecimentos, já adquiridos ao longo do período de formação académica, sobre os protocolos utilizados em contexto de web services (SOAP e REST) e ao mesmo tempo familiarizar-me com a ferramenta de gestão de problemas Jira.

- Resolução de pequenos problemas de manutenção

Esta tarefa teve como objectivo a resolução de pequenos problemas e implementação de alterações na aplicação que surgiram durante trabalhos de manutenção, e constituiu o primeiro contacto mais profundo com a estrutura e código desenvolvido no projecto.

<sup>4</sup> Por *standalone*, pretende-se transmitir a ideia de uma aplicação que é desenvolvida como um único módulo, e não está dependente de um servidor aplicacional para ser executada

<sup>5</sup> *Issues* são os problemas relatados pelos consumidores da aplicação, numa ferramenta criada para o efeito – Jira (Atlassian)

### 3.3.2 Desenvolvimento

Concluída a primeira fase de aprendizagem, com a aquisição das bases sobre a aplicação e o projecto, iniciei a fase de desenvolvimento de módulos da aplicação.

O desenvolvimento, neste projecto, tem várias fases:

- Apresentação e Análise Funcional
- Análise detalhada dos requisitos
- Estimativas
- Desenvolvimento
- Conclusão

Todos estes passos são, de um modo geral, transversais ao componente a desenvolver para a aplicação, e tive a oportunidade, durante o período de estágio, de participar em todos eles.

#### 3.3.2.1 Apresentação e Análise Funcional

A primeira parte do processo de desenvolvimento consiste na apresentação, por parte dos elementos da equipa de análise funcional, do objectivo e funcionamento do componente/módulo a desenvolver, com a explicação de conceitos de negócio inerentes ao módulo, requisitos do cliente e estrutura pretendida. Este processo de apresentação é complementado com uma análise entre os vários elementos presentes, abordando a possibilidade de várias soluções primárias.

#### 3.3.2.2 Análise Detalha dos Requisitos

Depois de apresentado o componente a desenvolver, a equipa de desenvolvimento reúne-se para, de forma detalhada, analisar os requisitos do cliente.

Esta reunião traduz-se num processo de debate e discussão sobre a forma de abordar os problemas e a procura de consenso nas soluções a utilizar de acordo com o que é pretendido pelo cliente.

#### 3.3.2.3 Estimativas

Com a conclusão da análise aos requisitos exigidos pelo cliente, é pedido à equipa que elabore as estimativas de desenvolvimento do módulo, através da análise aos vários componentes que constituem o constituem. Mais uma vez, a equipa reúne-se para procurar estabelecer estimativas adequadas. É um processo fundamental para a elaboração do esforço a apresentar ao cliente, e extremamente difícil numa fase inicial de carreira. Senti dificuldades, principalmente no primeiro módulo em que estive envolvido, fruto da minha inexperiência, embora não considere que esta situação constitua um problema, mas sim mais uma oportunidade de aprendizagem, que é contínua.

#### 3.3.2.4 Desenvolvimento

No projecto *eMudar*, um aspecto que considerei bastante importante foi a “exigência”, a todos os envolvimento neste processo, de um desenvolvimento *full stack*, isto é, procurar garantir que os programadores estejam envolvidos na criação de todos os elementos dos módulos a desenvolver (camada de apresentação, negócio e dados), assegurando assim que a evolução é feita de forma global, ao nível das diversas tecnologias.

No que diz respeito ao desenvolvimento, mais propriamente dito, o *team leader* estabelece, de acordo com os trabalhos que foram estimados no ponto anterior, um catálogo de tarefas, que no conjunto representam o módulo a construir. Estas tarefas são distribuídas pelos elementos de acordo com critérios estabelecidos em concreto para o módulo específico, por exemplo, se determinado elemento esteve envolvido no desenvolvimento de um componente que é utilizado numa fracção do novo módulo, fará sentido que esse mesmo elemento se foque mais nessa parte. Esta atribuição é sempre feita em conjunto com toda a equipa.

#### 3.3.2.5 Conclusão

No que diz respeito à equipa de desenvolvimento, a conclusão da construção do novo componente atinge-se quando todas as tarefas estão concluídas. Normalmente as últimas tarefas a realizar são as revisões de código e testes de desenvolvimento (testes especificados pela equipa de *testing* e realizados pelos programadores).

Depois de concluído o processo de desenvolvimento, o módulo é entregue à equipa de *testing* e posto à prova para a detecção de problemas que possam ainda existir.

### 3.3.3 Módulos específicos desenvolvidos

Ao longo do estágio tive oportunidade de participar no desenvolvimento, de raiz, de dois módulos da aplicação, e portanto, trabalhei em todos os pontos descritos na secção anterior.

#### 3.3.3.1 Processo Caixas - Módulo Bloco A

O primeiro módulo no qual estive inserido tem o nome de *Caixas Bloco A*.

Este módulo faz parte de um novo componente da aplicação: Caixas. Este componente é composto por mais três blocos: B, C e D.

O componente de Caixas diz respeito às diversas operações desenvolvidas em postos de caixas, nos balcões e agências da entidade:

- Abertura / Encerramento de Caixa
- Entrega / Recepção de Valores Monetários ao / do cofre
- Depósitos em Numerário de clientes

- Etc...

Concretamente, o bloco A do módulo de Caixas engloba as seguintes operações:

- Abertura de Caixa
- Fecho de Caixa (Processo)
- Recepção de valores do Cofre / Entrega ao Caixa (Processo)
- Anulação da Entrega ao Caixa (Processo)
- Entrega de valores da Caixa / Recepção do Cofre (Processo)
- Validação do saldo de Caixa
- Validação do saldo de Cofre

Neste bloco, parte das operações representavam processos criados e modelados no BPM, sendo que as restantes era apenas tarefas directas.

Ao longo do desenvolvimento deste bloco, estive envolvido directamente com os processos de *Fecho de Caixa*, *Anulação da Entrega ao Caixa* e com a tarefa de *Validação do saldo do Cofre*.

#### Fecho de Caixa

Este processo tem por objectivo realizar a operação de fecho de caixa de um operador da agência. É um processo que requiere a intervenção de dois indivíduos:

- Operador de Caixa, utilizador que inicia o processo
- Tesoureiro, utilizador que valida a operação

E está dividido em três tarefas:

- Início de Fecho de Caixa, a cabo do operador de caixa
- Aprovação do Fecho de Caixa, executada pelo Tesoureiro
- Conclusão do processo de Fecho de Caixa, novamente pelo operador

É um processo com alguma complexidade pois o fluxo que é seguido ao longo de toda a operação pode ser diferente consoante alguns condicionantes, principalmente relacionados com a situação da caixa a ser fechada, por exemplo, o processo será imediatamente aprovado na primeira tarefa, sem a intervenção do tesoureiro, caso o saldo da caixa esteja saldado (saldo de valor igual a zero). Se pelo contrário existirem diferenças de caixa (o saldo de caixa, depois de todas as operações, for positivo ou negativo), o percurso terá obrigatoriamente de passar pela aprovação. Eu desenvolvi uma parte significativa da primeira tarefa, onde se inclui o *web service* que executa a operação de fecho, e parte da última tarefa,

A segunda tarefa, a aprovação, foi desenvolvida por outro elemento da equipa, embora tenha acompanhado o seu desenvolvimento. Nesta tarefa, caso a operação de

fecho seja aprovada, são executadas operações de regularização do saldo de caixa, eliminando as diferenças e executando o fecho, através de *web services*.

Finalmente a última tarefa é a de conclusão do processo, onde o utilizador (operador de caixa que iniciou o processo) é informado do resultado da validação/aprovação e onde terá que imprimir a certificação do processo de fecho (caso tenha sido aprovado na tarefa anterior).

Foi um processo que contou com algumas alterações, exigidas pelo cliente, no decorrer do seu desenvolvimento. Este aspecto provocou alguns atrasos no decorrer dos trabalhos.

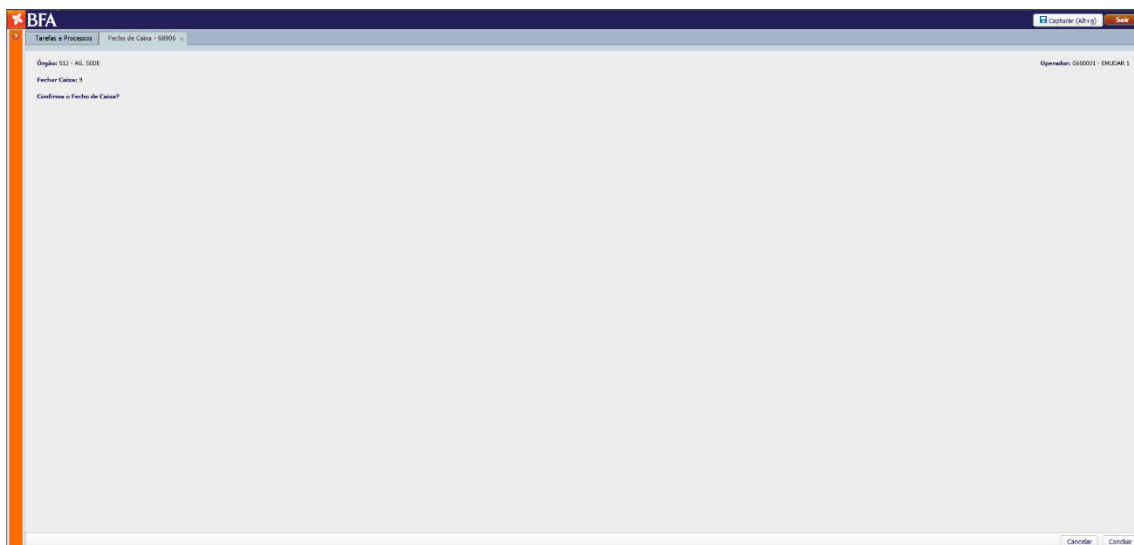


Figura 6 - Primeiro ecrã do processo de Fecho de Caixa

### Anulação de Entrega ao Caixa

Depois de participar no desenvolvimento do processo descrito no ponto anterior, fiquei responsável por criar o processo de anulação da operação de entrega ao caixa (apenas esta operação é possível de ser anulada, no bloco A).

Este processo tem dois intervenientes:

- Tesoureiro, que inicia o processo de anulação
- Indivíduo responsável pela validação da anulação de operações em Caixa

E está dividido em três tarefas:

- Início do processo de anulação
- Aprovação/Recusa da operação
- Tomada de conhecimento e conclusão do processo

O processo de anulação existe para tornar operações que são certificadas sem efeito, na aplicação. Na verdade, este processo não elimina a sua contraparte, apenas a marca como anulada e cria nova certificação dessa anulação. Este registo é feito também na operação a ser anulada.

A consulta de certificações de operações é feita através de um módulo já desenvolvido denominado de Jornal Electrónico. Ao consultar o detalhe de uma dada certificação de uma operação passível de ser anulada, é dada a possibilidade ao utilizador (caso este cumpra alguns requisitos específicos) de a anular (Figura 7). É a partir deste ponto que se pode dar início a um processo de anulação.

No componente de Caixas, o processo de anulação começa com a introdução do motivo para a realização da operação (Figura 8), e de seguida o processo é encaminhado para o grupo de utilizadores com permissões para avaliar o pedido. Os utilizadores deste grupo receberão uma mensagem de notificação na sua área de trabalho da aplicação (Figura 9) onde poderá reclamar a tarefa. De seguida terá que decidir se pretende confirmar ou recusar o pedido (Figura 10).

Após da tomada de decisão, o utilizador que iniciou o processo deverá reclamar a tarefa de conclusão do processo de anulação, onde lhe é transmitida a decisão do utilizador anterior, e caso o pedido seja aprovado, terá que imprimir a certificação da operação.

Como referi no início deste sub-capítulo, tive a responsabilidade de modelar este processo, bem como desenvolver todos os serviços que lhes estão associados, e também o formulário de impressão da certificação da operação. Este aspecto foi importantíssimo pois permitiu-me adquirir conhecimentos mais profundos sobre a aplicação, e perceber melhor a sua estrutura e o seu funcionamento.

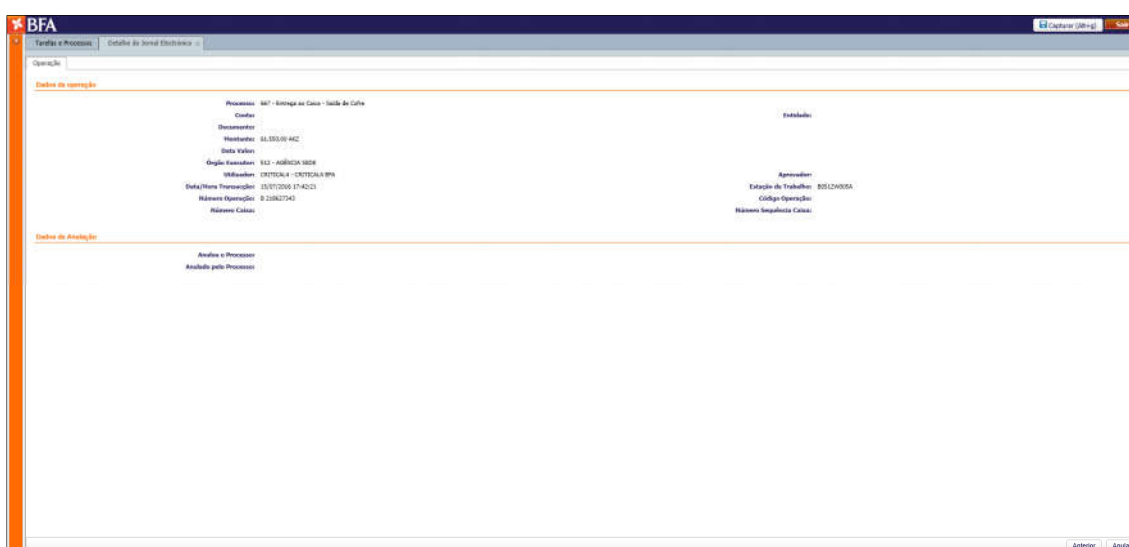


Figura 7 - Ecrã de detalhe de certificação de um processo de Entrega ao Caixa

Figura 8 - Primeiro ecrã do processo de anulação

| Processo | Número da operação   | Número da folha       | Atividade | Data Início | MDC | Órgão | Situação do Processo |
|----------|----------------------|-----------------------|-----------|-------------|-----|-------|----------------------|
| 67918    | Deposito Cheques ODC | Depositar Cheques ODC | 040909    | 25-01-2018  | 0   | 0     |                      |
| 67953    | Deposito Cheques ODC | Depositar Cheques ODC | 040909    | 25-01-2018  | 0   | 0     |                      |
| 67954    | Deposito Cheques ODC | Depositar Cheques ODC | 040909    | 25-01-2018  | 0   | 0     |                      |

Figura 9 - Exemplo de notificação gerada para um processo de anulação

Figura 10 - Ecrã de aprovação de anulação de processo

## Validação do Saldo do Cofre

A operação de validação do saldo do cofre, cujo desenvolvimento foi da minha responsabilidade, não envolve um processo pois é uma tarefa com apenas um ecrã.

Esta operação tem, por isso, apenas um interveniente: o tesoureiro da agência do cofre cujo saldo se pretende validar.

Nesta operação, é pedido ao utilizador que escolha a moeda e introduza o saldo respectivo, sendo que existem apenas dois resultados possíveis: *saldo correcto* ou *saldo incorrecto* (Figura 11).

A validação é feita através de um *web service* que compara o saldo introduzido pelo utilizador para a moeda escolhida, com o saldo existente no cofre nesse momento.

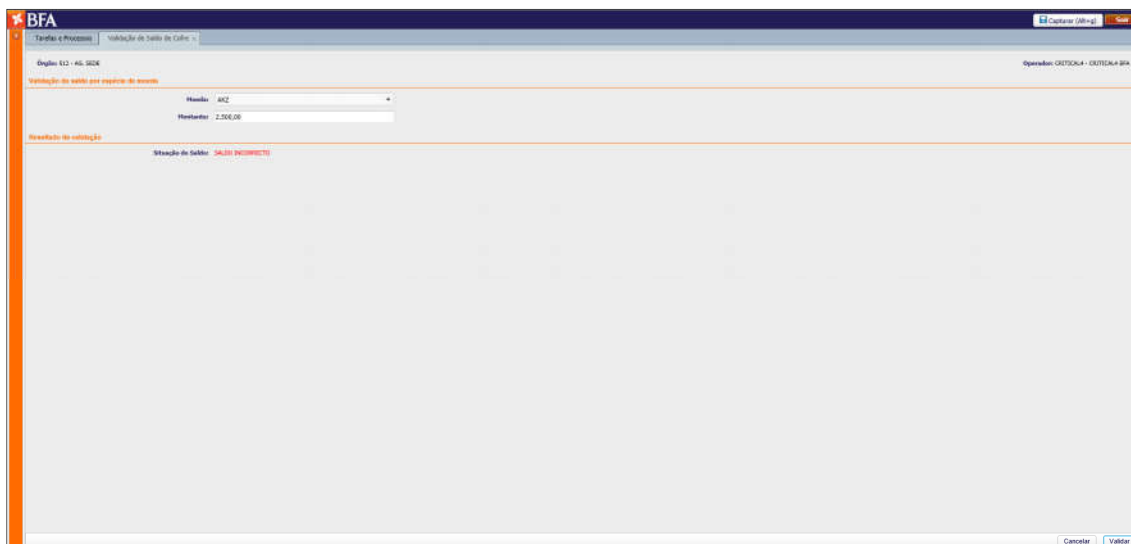


Figura 11 - Ecrã de Validação de Saldo de Cofre

### 3.3.3.2 Processo Caixas – Módulo Bloco C

O segundo módulo em que participei foi o módulo Bloco C, de Caixas.

O desenvolvimento de todo módulo, foi feito em cooperação com a colega de equipa Joana Sousa.

As tarefas a desenvolver neste módulo foram:

- Depósito em Numerário (Moeda Nacional e Estrangeira)
- Anulação do processo de Depósito
- Criação de uma Calculadora, com dois modos – normal e auxiliar

Depois de feitas as estimativas, as tarefas foram divididas e cada elemento trabalhou em componentes específicos do módulo. Eu fiquei responsável pelo desenvolvimento do processo de Anulação e pela criação do componente da calculadora que contém ambos os modos, bem como o desenvolvimento da calculadora em modo auxiliar.

O objectivo deste módulo é acrescentar a funcionalidade para realizar a operação de depósito de numerário em caixa, seja em moeda nacional ou moeda estrangeira.

O processo de depósito tem um interveniente: o operador de caixa onde é realizado o depósito. Tem por isso apenas uma tarefa.



### Anulação do Depósito em Numerário

No desenvolvimento deste processo procurou-se reformular o processo anterior, desenvolvido para o bloco A, de forma a torná-lo genérico para todos os processos de Caixa, ou seja, independentemente do processo que se pretendia anular, a operação para o anular seria apenas uma. Esta abordagem vai permitir que, na eventualidade da criação de processos no futuro que sejam passíveis de serem anulados, a operação para o fazer já existe e pode ser reutilizada.

Para o efeito, foi necessário adaptar o que foi criado no processo original, nomeadamente ecrãs, serviços e tarefas, para um contexto genérico. Para além deste ponto, foi necessário criar ferramentas, como configurações, que permitissem adaptar o novo processo à especificidade da operação que se pretende anular, pois existirão variáveis que são distintas entre processos, como por exemplo, o grupo de utilizadores responsáveis pela aprovação da anulação, ou até os formulários a imprimir na conclusão do processo.

Resumidamente, as tarefas a desenvolver foram:

- Reformulação de ecrãs, controladores, serviços e elementos do BPM
- Criação de configurações, com as características de cada tipo de operação a ser anulada
- Criação de novos serviços, adaptados à genericidade do processo, nomeadamente aqueles que servem o encaminhamento de tarefas e notificações para o grupo de utilizadores definidos nas configurações

Ao trabalhar neste módulo, denotei uma clara evolução para o anterior, nomeadamente ao nível do conhecimento da arquitectura e estrutura do projecto. Esta evolução deveu-se, em grande parte, à forma como o *team leader* levou a cabo o meu processo de integração na equipa de desenvolvimento: a delegação de responsabilidades para a criação de componentes foi, talvez, o ponto mais importante neste aspecto. Obrigou-me, no bom sentido da palavra, a desenvolver as minhas competências e apetências, e para além disso cimentou a minha relação com os outros elementos de equipa, que foram essenciais para o meu progresso, pois mesmo estando responsável pelo desenvolvimento de fracções dos módulos, a comunicação é constante no seio da equipa.

### Calculadora – *Wrapper* e modo Auxiliar

A calculadora foi um requisito imposto pelo cliente, e um componente completamente novo na aplicação. Um dos objectivos propostos para o desenvolvimento deste componente foi o de o criar de forma a que pudesse ser aproveitado, no futuro, em novos elementos da aplicação, isto é, que não fosse desenhado apenas para este módulo de Caixas – Bloco C. Outro objectivo a atingir era garantir que todas as

operações realizadas na calculadora fossem feitas apenas do lado do cliente, de forma a tornar o componente o mais eficiente possível, sem chamadas ao servidor.

No caso específico da calculadora para este módulo, a sua função é realizar cálculos directamente na aplicação e depois possibilitar ao utilizador transferir o resultado final para o processo de depósito, resultado esse que representa o valor do numerário a depositar.

A calculadora tem dois modos disponíveis:

- Modo normal – Calculadora assume as funções de uma calculadora simples, como por exemplo a que está disponível no sistema operativo Windows. De resto, foi pedido que fosse desenvolvida tendo como base o comportamento desta (**Error! Reference source not found.**).

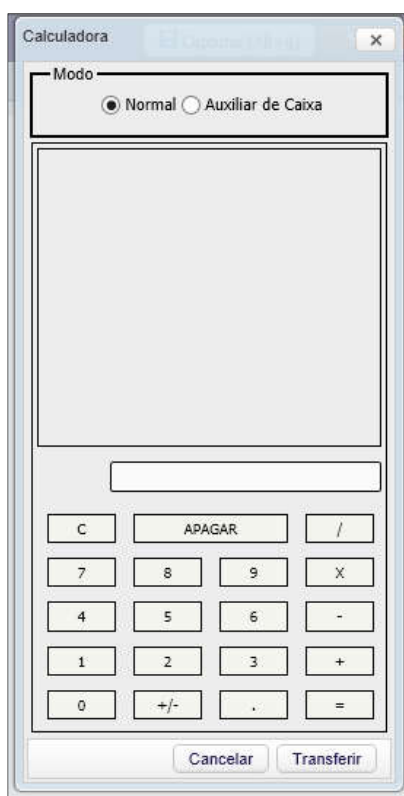


Figura 12 - Calculadora Modo Normal

- Modo auxiliar – A calculadora disponibiliza as várias espécies da moeda de depósito (seja para notas ou moedas) e o utilizador pode introduzir a quantidade de divisas ou o montante específico, sendo depois **calculado** o montante ou a quantidade, respectivamente. O valor total é calculado com a soma de todos os valores introduzidos pelo utilizador (**Error! Reference source not found.**)

The screenshot shows a window titled 'Calculadora' with a close button in the top right corner. Inside the window, there are two main sections for configuration:

- Modo:** A section with two radio buttons: 'Normal' (unselected) and 'Auxiliar de Caixa' (selected).
- Opção:** A section with two radio buttons: 'Quantidade' (selected) and 'Montante' (unselected).

Below these sections is a table with three columns: 'Espécie', 'Quantidade', and 'Montante'. The table contains two groups of entries:

- Notas (Notes):** A group of entries with values 50.00, 100.00, 200.00, 500.00, 1000.00, 2000.00, and 5000.00 in the 'Espécie' column, and empty input fields in the 'Quantidade' column.
- Moedas (Coins):** A group of entries with values 0.50, 1.00, 5.00, 10.00, 20.00, 50.00, and 100.00 in the 'Espécie' column, and empty input fields in the 'Quantidade' column.

At the bottom of the table, there are two summary rows:

- Total - Notas:** A row with a label 'Total - Notas' and an empty input field.
- Total - Moedas:** A row with a label 'Total - Moedas' and an empty input field.

At the very bottom of the window, there are three buttons: 'Cancelar', 'Limpar', and 'Transferir'.

Figura 13 - Calculadora Modo Auxiliar

Eu estive responsável por criar o elemento que controla os dois modos funcionais requeridos para esta calculadora. Este componente denomina-se por *wrapper*. É a partir deste elemento que o utilizador consegue escolher qual o modo que pretende operar, e depois transferir o valor para o processo a decorrer. Este *wrapper* foi desenhado de modo a aceitar novos modos que possam ser criados no futuro, sendo por isso escalável.

O segundo componente desenvolvido foi o que contemplava o modo auxiliar. Como foi referido, a função deste modo é a de calcular o valor total introduzido pelo utilizador, seja em função da quantidade de cada espécie da moeda seleccionada no processo, seja em função do montante dessa mesma espécie. Se o utilizador, por exemplo, introduzir uma quantidade de cinco notas de espécie “5.00”, o montante calculado seria de “25.00” e representava o resultado final. Caso o utilizador pretender introduzir o montante directamente, e segundo o exemplo anterior, “25.00”, seria calculada a quantidade de notas para a espécie “5.00”, ou seja, cinco notas e o resultado final seria o montante introduzido.

O desenvolvimento da calculadora representou um desafio dada a natureza de funcionamento pretendida: operando apenas do lado do cliente, e por forma a

garantir isso mesmo, toda a funcionalidade implementada foi feita em *Javascript*. O contacto com esta linguagem ao longo do período de formação académico foi escasso, e portanto o meu conhecimento no assunto também o era. Felizmente a equipa conta com elementos já com elevada experiência em diversas linguagens de programação, e o auxílio que me foi prestado foi inexcedível, permitindo-me concluir com sucesso o desenvolvimento do componente.

No final, foi necessário garantir que a integração dos dois modos, sendo que ambos os modos foram desenvolvidos de forma independente, no componente que os continha, o *wrapper*, decorresse sem problemas.

### 3.3.3.3 Configuração Automática de Limites de Autorização

No decorrer do estágio tive também a responsabilidade de desenvolver um componente para a criação/actualização de forma automática dos limites de autorização.

Os limites de autorização são limites estabelecidos que definem acções a executar se forem atingidos. Os limites podem ser atribuídos a operações como, por exemplo, transferências, definindo barreiras para o valor a transferir (se o valor atingir um dado limite, a acção a tomar pelo operador deverá estar de acordo com o estabelecido pelo banco), ou por exemplo relativamente a taxas de juro de uma qualquer aplicação financeira.

De modo a tornar o processo de definição de limites automático, foi criada uma configuração com todas as características de cada limite de autorização. Esta configuração divide os limites em grupos de autorização, que representam o tipo de operação sobre a qual incidem tais limites (transferências, depósitos, etc...).

A configuração deverá ser actualizada pelo utilizador sempre que se pretende criar ou alterar limites e depois será transferida para a base de dados da aplicação através do *upload* de um ficheiro global de configurações (DAR<sup>6</sup>) existente na aplicação.

Para desenvolver esta aplicação foi criado um serviço que recorrendo às configurações criadas para os limites, executa as validações necessárias no momento em que o ficheiro *DAR* é enviado e cria ou actualiza os limites se necessário.

---

<sup>6</sup> DAR – O DAR é um ficheiro que contém algumas das mais importantes configurações da aplicação, como configurações do *front-end*, módulo de integração, dos ficheiros de certificação, etc.

## 4. Conclusão

Este estágio representa um percurso de 9 meses de intenso trabalho e formação.

Antes de dar início ao estágio frequentei a formação intensiva em Java do programa “Acertar o Rumo – 2ª Edição” que me deu as bases necessárias para enfrentar o desafio de relançar a minha carreira profissional numa nova área.

De facto, não posso concluir este relato sobre o meu novo percurso profissional sem referir o programa que me possibilitou abraçar novos desafios e trabalhar numa área com futuro e cheia de oportunidades. O programa exige muito esforço e dedicação, mas que, reflectindo agora, é sem dúvida recompensador.

A transição do contexto académico para o profissional deu-se com algumas reservas e receios, naturais de quem está a começar a sua actividade. Felizmente, considero que todo o conhecimento que foi passado aos formandos no decorrer do programa e o nível de exigência que nos habituou foram suficientes para dar a confiança necessária para arriscar neste novo desafio.

No que ao estágio diz respeito, não posso imaginar como poderia ter corrido de melhor forma.

Desde o início do estágio todas as pessoas envolvidas no projecto ofereceram o seu total apoio na minha integração, o que facilitou em muito o meu percurso até final. A grande experiência de todos os elementos envolvidos provou ser um aspecto crucial na minha constante aprendizagem ao longo deste último ano.

Um factor a destacar é também a forma como é gerida a equipa, referindo-me a todos os que constituem a equipa de gestão e liderança do projecto. A gestão e as acções de liderança foram, sem dúvida, um dos aspectos mais positivos do estágio. Serviram também de fonte para o meu crescimento, quer a nível pessoal, como profissional.

O projecto, em si, contém todos os ingredientes de uma receita fantástica para quem, como eu, deseja aprender ao máximo neste nova, exigente, e até há pouco tempo, por mim desconhecida área profissional. Desde o elevado grau de complexidade, às tecnologias utilizadas, que de resto, vão de encontro ao que foi abordado ao longo da formação académica, tudo contribui para garantir o sucesso de quem está apenas a começar.

Em retrospectiva, considero que esta aposta, quer na formação na Universidade de Coimbra, quer no projecto no qual foi inserido, foi uma aposta ganha. No total, foram dois anos onde consegui, com muito trabalho, dedicação e apoio dos que acompanharam neste percurso, definitivamente, acertar o meu rumo profissional.

## Referências

- [1 "BFA@glance," 2016. [Online].  
] <http://www.bfa.ao/Conteudos/Artigos/detalhe.aspx?sidc=2652&idc=2851&idsc=2856&idi=6586&idl=1>