



FLM Guide - Roteiro do Festival Literário da Madeira

Carolina Sofia Martins Sousa

(Licenciada)

*Relatório de Estágio Submetido à Universidade da Madeira para a
Obtenção do Grau de Mestre em Engenharia Informática*

Funchal - Portugal

Janeiro 2016

Orientador Interno:

Professor Doutor Eduardo Leopoldo Fermé

Professor Associado à Faculdade de Ciências Exatas e da Engenharia da Universidade da Madeira

Orientador Externo:

Engenheiro Hélder Gouveia Figueira Pestana

Responsável pelo Departamento de Informática da Nova Delphi

RESUMO

O presente relatório surge de modo a expor toda a experiência vivenciada, assim como todo o trabalho desenvolvido ao longo do estágio curricular inserido no plano de conclusão do Mestrado em Engenharia Informática, da Universidade da Madeira.

A ideia de realizar um estágio curricular para a conclusão do 2º ciclo de estudos em Engenharia Informática surgiu da vontade de conhecer a dinâmica de trabalho de uma empresa real, com necessidades e prazos reais, e ao mesmo tempo ganhar alguma experiência profissional, pondo em prática muitas das competências adquiridas durante a formação académica.

Neste relatório serão descritas as metodologias e tecnologias utilizadas para o desenvolvimento de uma aplicação móvel, na qual é possível consultar o roteiro/programa de eventos do Festival Literário da Madeira, organizado pela empresa Nova Delphi.

Será exposta a análise previamente feita sobre algumas tecnologias e ferramentas *crossplatform* (multiplataforma) pesquisadas com o intuito de serem utilizadas no projeto. Posteriormente, serão também apresentadas as fases e decisões tomadas em relação aos procedimentos escolhidos e utilizados durante o processo de desenvolvimento e implementação das duas versões disponibilizadas da aplicação móvel para a 3ª e 4ª edições do Festival Literário da Madeira – versões 1.0 e 2.0, respetivamente.

PALAVRAS-CHAVE

Engenharia de Requisitos

Aplicação Móvel

Tecnologias Multiplataforma

Festival Literário da Madeira

Escalabilidade

AGRADECIMENTOS

Em primeiro lugar, gostaria de agradecer a toda a minha família, amigos e colegas, pelo apoio, força e ânimo demonstrados desde o início desta fase da minha formação académica. Sem o apoio daqueles que me são mais queridos, esta caminhada teria sido, com certeza, mais difícil.

Quero agradecer também aos meus orientadores, o Prof. Eduardo Fermé e o Eng. Hélder Pestana, por se terem demonstrado sempre disponíveis e bastante prestáveis durante todo este percurso do estágio curricular, contribuindo sempre que necessário com conselhos, orientações e sugestões fundamentais. Foram essenciais para a conclusão desta etapa.

Gostaria também de agradecer à Nova Delphi e a toda equipa da empresa pela oportunidade de ter integrado e poder ter contribuído para um projeto tão cativante como é o Festival Literário da Madeira.

De um modo geral, agradeço também a todos os docentes da Faculdade de Ciências Exatas e da Engenharia da Universidade da Madeira, que me acompanharam tanto na Licenciatura como no Mestrado e que contribuíram para a minha formação académica e pessoal, através da transmissão de conhecimentos, experiências e vivências, durante o meu percurso académico.

Muito obrigada a todos!

ÍNDICE

I. Introdução	1
I.1. Contexto do Estágio	3
I.2. Festival Literário da Madeira	4
I.2.1. Aplicação móvel	5
I.3. Objetivos	7
I.4. Estrutura do Relatório	8
II. Estado da Arte	9
II.1. Tecnologias multiplataforma	11
II.1.1. <i>Websites</i> e utilização de <i>shells</i> nativas	12
II.1.2. Uma única linguagem de programação, vários ambientes móveis	13
II.1.3. Ferramentas mais utilizadas	14
II.1.4. Considerações finais/análise global	16
II.2. Aplicações semelhantes	19
II.2.1. Festival de Cannes 2014	19
II.2.2. Festival Letteratura	21
II.2.3. Optimus Alive 2013	22
II.2.4. Tiff 2013	23
II.2.5. Feeling Madeira	24
II.2.6. Considerações finais/análise global	25
III. Desenvolvimento da Aplicação	27
III.1. Introdução	29
III.2. Levantamento de requisitos	31
III.2.1. Entrevistas	31
III.2.2. Principais funcionalidades	32
III.2.3. Requisitos da aplicação	33
III.3. Casos de utilização	40
III.3.1. Atores	40
III.3.2. Diagrama de casos de utilização	40
III.4. Diagramas de actividade	44
III.5. Arquitetura	48
III.6. Estrutura da Base de dados	50
III.6.1. Esquema Relacional	51
III.7. Interface do Utilizador	58
III.7.1. Mockups	58
III.7.2. Protótipos Abstratos Canónicos	59
III.7.3. Protótipos de Alta Fidelidade	60
IV. Implementação	63
IV.1. Introdução	65
IV.2. Linguagens e tecnologias utilizadas	67

IV.2.1. MySQL Workbench e SQLite Browser.....	67
IV.2.2. Appcelerator Titanium.....	68
IV.2.3. GitHub.....	71
IV.3. Versão 1.0	72
IV.3.1. Requisitos/funcionalidades implementados.....	72
IV.3.2. Testes	75
IV.4. Versão 2.0	77
IV.4.1. Normas de desenvolvimento para aplicações Android	78
IV.4.2. Atualização dos dados locais	81
IV.4.3. Novas funcionalidades implementadas	85
V. Conclusão	89
<hr/>	
V.1. Sobre a aplicação	91
V.2. Sobre o estágio curricular	93
V.3. Perspetivas Futuras.....	94
Referências	95
Anexos	99
<hr/>	
Anexo 1. Documento de Requisitos.....	100
Anexo 2. Esquema Relacional da Base de Dados	110
Anexo 3. Protótipos Abstratos Canónicos (CAPs)	111
Anexo 4. Protótipos de Alta Fidelidade.....	114

LISTA DE FIGURAS

Figura I.1 - Logo da DIKE Madeira	3
Figura I.2 - Logo da Nova Delphi	3
Figura I.3 - Programa da 3ª edição do Festival Literário da Madeira (2013) [1].....	4
Figura I.4 - Principais focos da aplicação móvel.....	6
Figura II.1 - Custos, Acesso ao <i>Hardware</i> e ao <i>Software</i> e Desempenho de aplicações nativas ou desenvolvidas com tecnologias multiplataforma [2]	12
Figura II.2 - Exemplo de um pequeno <i>script</i> em Lua, retirado do <i>site</i> da Corona Labs [3].....	14
Figura II.3 - Excerto de uma função em JavaScript implementada com o WidgetPag [5]	15
Figura II.4 - Exemplo de um componente apresentado nas diferentes plataformas [12].....	15
Figura II.5 - Logo da aplicação móvel Festival de Cannes 2014	19
Figura II.6 - Ecrã inicial da aplicação móvel Festival de Cannes 2014.....	20
Figura II.7 - a) Programa de eventos a) e b) mapa com os locais do Festival de Cannes 2014	20
Figura II.8 - Logo da aplicação móvel Festivaletteratura	21
Figura II.9 - Programa de eventos da aplicação móvel Festivaletteratura	21
Figura II.10 - Logo da aplicação móvel Optimus Alive 2013	22
Figura II.11 - a) Programa de concertos do Optimus Alive 2013 organizados por dia, b) palco e c) artista.....	22
Figura II.12 - Logo da aplicação móvel TIFF 2013	23
Figura II.13 - a) Menu inicial, b) notícias dos pontos principais e c) programa de eventos da aplicação móvel TIFF 2013	23
Figura II.14 - Logo da aplicação móvel Feeling Madeira	24
Figura II.15 - a) Mapa e b) programa de eventos da aplicação móvel Feeling Madeira	24
Figura III.1 - Sequência de entrevistas realizadas para levantamento de requisitos	31
Figura III.2 - Representação da divisão do projecto em módulos	33
Figura III.3 - Diagrama de casos de utilização	41
Figura III.4 - Diagrama de atividades do processo localizar evento no mapa	44
Figura III.5 - Diagrama de atividades do processo de atualização de dados	46
Figura III.6 - Diagrama representativo da arquitetura Cliente-Servidor	48
Figura III.7 - Diagrama representativo da arquitetura MVC	49
Figura III.8 - Exemplo prático da solução encontrada para o armazenamento das traduções	50
Figura III.9 - Esquema relacional da base de dados remota (<i>backoffice</i>).....	51
Figura III.10 - Tabelas para o armazenamento de dados relativos aos eventos	52
Figura III.11 - Tabelas para o armazenamento de dados relativos aos convidados participantes.....	53

Figura III.12 - Tabelas para o armazenamento de dados relativos aos locais	54
Figura III.13 - Tabelas para o armazenamento de dados relativos aos parceiros	55
Figura III.14 - Tabelas para o armazenamento de dados de contactos	56
Figura III.15 - Tabelas para o armazenamento de dados de contactos	56
Figura III.16 - <i>Mockup</i> do ecrã inicial	58
Figura III.17 - <i>Mockup</i> dos ecrãs da lista de eventos e do evento em detalhe (descrição e convidados)	59
Figura III.18 - CAP do ecrã inicial	59
Figura III.19 - CAPs dos ecrãs da lista de eventos e do evento em detalhe (descrição e convidados)	60
Figura III.20 - Protótipo de alta fidelidade do ecrã inicial	60
Figura III.21 - Protótipo de alta fidelidade dos ecrãs da lista de eventos do evento em detalhe (descrição e convidados)	61
Figura IV.1 - <i>Timeline</i> da evolução da aplicação FLM	66
Figura IV.2 - Editor de esquemas relacionais do MySQL Workbench	67
Figura IV.3 - Editor DB Browser for SQLite	68
Figura IV.4 - Logo da Appcelerator e Titanium	68
Figura IV.5 - Esquema do funcionamento do Titanium Studio [17]	69
Figura IV.6 - Titanium SDK como ponte entre o JavaScript e o código nativo [17]	69
Figura IV.7 - <i>Framework Alloy</i> do Appcelerator	70
Figura IV.8 - Exemplo: aplicação Alloy vs aplicação Titanium tradicional	70
Figura IV.9 - Logo e <i>splashscreen</i> da versão 1 da aplicação	72
Figura IV.10 - Alguns dos ecrãs desenvolvidos na 1.0 versão da aplicação	75
Figura IV.11 - Logo e <i>splashscreen</i> da versão 2 da aplicação	77
Figura IV.12 - Navegação em <i>tabs</i> usada a partir da versão 2.0	79
Figura IV.13 - <i>Action bar</i> nativa com o título do ecrã e botões de voltar atrás e menu	79
Figura IV.14 - Barra de pesquisa incorporada na <i>action bar</i> a partir da versão 2.0	80
Figura IV.15 - Exemplo da aplicação da regra dos 48dp numa interface	80
Figura IV.16 - Tipo de letra Roboto nas várias variantes	81
Figura IV.17 - Exemplo da utilização do tipo de letra Roboto	81
Figura IV.18 - Logo da <i>framework Sails.js</i>	81
Figura IV.19 - Processo de atualização de dados da aplicação móvel	83
Figura IV.20 - Algoritmo de atualização dos dados recebidos pelo <i>webservice</i>	84
Figura IV.21 - Resultados da pesquisa organizados por Eventos, Pasrticipantes, Locais e Parceiros	85
Figura IV.22 - Mapa com a localização geográfica dos locais onde decorrem os eventos	86
Figura IV.23 - Área com as Atualizações Recentes	87

LISTA DE TABELAS

Tabela II.1 - Aspectos positivos e negativos das várias abordagens analisadas	17
Tabela III.1 - Requisitos funcionais do Módulo I - Guia do FLM e respetivas prioridades	34
Tabela III.2 - Requisitos funcionais do Módulo II - Associados do Festival e respetiva prioridade	35
Tabela III.3 - Requisitos funcionais do Módulo III - Roteiro Turístico/Literário e respetivas prioridades	36
Tabela III.4 - Requisitos funcionais do Módulo IV - Loja e respetivas prioridades	36
Tabela III.5 - Requisitos não funcionais - Portabilidade e respetivas prioridades	37
Tabela III.6 - Requisitos não funcionais - Usabilidade e respetivas prioridades	37
Tabela III.7 - Requisitos não funcionais - Acessibilidade e respetivas prioridades	37
Tabela III.8 - Requisitos não funcionais - Disponibilidade e respetivas prioridades	38
Tabela III.9 - Requisitos não funcionais - Manutenção e respetivas prioridades	38
Tabela III.10 - Requisitos não funcionais - Escalabilidade e respetivas prioridades	38
Tabela IV.1 - Requisitos implementados na versão 1.0.....	73

ACRÓNIMOS

CAP - *Canonical Abstract Prototype*

CINM - *Centro Internacional de Negócios da Madeira*

CSS - *Cascading Style Sheets*

DB - *Database*

FLI - *Festival Literário Infantil*

FLM - *Festival Literário da Madeira*

GPS - *Global Positioning System*

HTML - *Hyper Text Markup Language*

HTTP - *HyperText Transfer Protocol*

IDE - *Integrated Development Environment*

iOS - *iOperation System*

MVC - *Model View Controller*

MySQL - *My Structured Query Language*

OS - *Operating System*

PHP - *Hypertext Preprocessor*

SDK - *Software Development Kit*

SDM - *Sociedade de Desenvolvimento da Madeira*

SQL - *Structured Query Language*

TSS - *Titanium Stylesheets*

UI - *User Interface*

URL - *Uniform Resource Locator*

VCS - *Version Control System*

XML - *eXtensible Markup Language*

I. INTRODUÇÃO

“Tell me and I forget. Teach me and I remember. Involve me and I learn.”

– Benjamin Franklin

I.1. CONTEXTO DO ESTÁGIO

O presente relatório e respetivo estágio, inseridos na unidade curricular Projeto/Estágio/Dissertação prevista no plano do Mestrado em Engenharia Informática da Universidade da Madeira, são apresentados no âmbito da conclusão deste 2º ciclo de estudos.

A experiência de um estágio curricular representa para mim a oportunidade de uma convivência num ambiente empresarial e num mercado real, que só por si proporcionam novos desafios que resultam, certamente, tanto consolidação das competências apreendidas durante a formação académica e postas agora em prática, como no desenvolvimento de outras novas.

A oportunidade da realização de um estágio curricular surgiu após contacto, via e-mail, com a Sociedade de Desenvolvimento da Madeira (SDM) que, por sua vez, reencaminhou a minha candidatura espontânea de estágio curricular a um conjunto de empresas do Centro Internacional de Negócios da Madeira (CINM). Neste conjunto de empresas está inserida a DIKE Madeira, que através do Eng. Hélder Pestana, responsável pelo Departamento de Informática, apresentou a proposta do projeto a desenvolver durante o estágio curricular.



Figura I.1 - Logo da DIKE Madeira

A DIKE Madeira, sediada na Rua da Carreira nº115/117, além de prestar serviços na área da consultadoria de gestão, é também empresa sócia de um projeto editorial, com o nome Euthalia Editora, representada pelas chancelas editoriais Nova Delphi e As Joaninhas by Nova Delphi. Este é um projeto que aposta na expansão cultural da região, sendo que uma das estratégias foi a criação do Festival Literário da Madeira.



Figura I.2 - Logo da Nova Delphi

I.2. FESTIVAL LITERÁRIO DA MADEIRA

O Festival Literário da Madeira (FLM) tem vindo a realizar-se todos os anos desde 2010, indo já na sua 5ª edição. O festival caracteriza-se por ser um conjunto de momentos culturais, não só ligados à literatura ou à escrita, procurando divulgar também outras manifestações de cultura como as artes plásticas ou a música.

The image shows a program poster for the 3rd edition of the Madeira Literary Festival (2013). The poster has a dark background with a large, stylized, multi-colored geometric pattern on the left side. The text is arranged in a list format, detailing various events from April 3rd to 6th. Each event includes its title, time, date, and day of the week. Some events also list the speakers or organizers. A note at the bottom states that all events take place at the Teatro Baltazar Dias, except for the ones specifically marked with an asterisk.

PROGRAMA	
CONFERÊNCIA DE ABERTURA	18.00 3 abril (qua)
CONVERSA À MESA Naomi Wolf e Rui Tavares	19.00
ESPETÁCULO Jazz, a Minha Arte (SCAT*) Massimo Cavalli / CEPAM Esc. Artes	21.00
CONVERSA CRUZADA A Arte de Morrer Longe	18.00 4 abril (qui)
João Tordo, Raquel Ochoa, Tiago Patrício e Tiago Salazar M: Cláudia Rodrigues	
EVENTO Multimédia, a Minha Arte (UMa, Colégio dos Jesuítas*)	21.30
ABERTURA DA EXPOSIÇÃO A Arte de Outro Lugar	11.00 5 abril (sex)
CONVERSA CRUZADA A Arte de Lidar com as Mulheres	18.00
Ana Luísa Amaral, Filipa Leal, Inês Fonseca Santos, João Paulo Cotrim e Waldir Araújo M: Paula Moura Pinheiro	
CONCERTO «Da minha língua vê-se o mar»	21.30
Mariano Deidda / Sérgio Godinho Entrada: 15€ Bilhetes à venda no Teatro Baltazar Dias	
CONVERSA CRUZADA A Arte da Guerra	10.00 6 abril (sáb)
Antonio Scurati, Carlos Vaz Marques, João Luís Barreto Guimarães e Pedro Mexia M: Ricardo Miguel Oliveira	
CONVERSA CRUZADA A Arte da Libertação	11.45
Anselmo Borges, Gina Picart, Lídio Araújo e Tabish Kahir M: Sílvio Fernandes	
CONVERSA CRUZADA A Arte de Pagar as Suas Dívidas	15.30
Carlos Quiroga, Maria do Rosário Pedreira, Raquel Varela e Rui Zink M: Carlos Vaz Marques	
CONVERSA À MESA Zygmunt Bauman e José Rodrigues dos Santos	17.15
ESPETÁCULO Poesia, a Minha Arte (Estalagem da Ponta do Sol*)	21.00

* À exceção dos eventos assinalados, todas as iniciativas têm lugar no Teatro Baltazar Dias.

Figura I.3 - Programa da 3ª edição do Festival Literário da Madeira (2013) [1]

Durante sensivelmente uma semana, estão reunidas, principalmente, no Teatro Municipal Baltazar Dias, denominado como o 'epicentro' do festival, personalidades de várias culturas e nacionalidades, num conjunto de conferências e conversas, proporcionando à população em geral da região debates agradáveis sobre temáticas atuais, inseridas num panorama cultural.

Como já foi referido anteriormente, também de música vive o Festival Literário da Madeira. A Nova Delphi tem por norma organizar dois espetáculos musicais, um no início e

outro já na reta final do festival, tendo já trazido à região artistas internacionais como a Adriana Calcanhotto, Rodrigo Leão, entre outros. No FLM, e como não poderia deixar de ser, existe também espaço para os artistas e projetos musicais regionais, sendo esta outra forma de promover o que é nosso.

Na 3ª edição, em 2013, surgiu pela primeira vez a inclusão do Festivalinho Literário Infantil (FLI) no FLM, numa tentativa de estimular um maior contacto dos mais pequenos com os livros e os seus escritores e ilustradores. O FLI leva os autores e ilustradores às escolas por toda a região, num ambiente bastante informal, onde são lidos e mostrados alguns excertos de obras infantojuvenis ao público mais jovem.

Para a realização do festival, a Nova Delphi conta com a colaboração de vários parceiros, que tendem a repetir-se ao longo das edições. Entre patrocinadores, *media partners* e outros tipos de parceiros são mais de dez as entidades que contribuem para que o festival tenha vindo a crescer, tendo já atingido um patamar de relevo cultural em termos nacionais.

Com todas estas ofertas, começa a surgir associado ao festival um novo nicho de turismo na região, o turismo cultural. Em parceria com a Agência Abreu, parceiro habitual do festival ao longo das últimas edições, para a 4ª edição do FLM, foi criado um pacote especial com viagem e estadia na região durante os dias do festival.

Com o FLM a atingir um grau de maturidade considerável, a Nova Delphi acho adequado a criação de uma aplicação que permitisse aumentar a experiência do público antes e no decorrer do festival.

I.2.1. Aplicação móvel

Na conjuntura atual, o uso de dispositivos móveis, como smartphones e tablets, está associado às tarefas do nosso dia a dia. Existe um vasto leque de aplicações que permitem que o uso destes dispositivos substitua o uso de objetos com o despertador, cronómetro, livros, rádios, etc. Num único dispositivo podemos reunir tanta informação quanta a que temos em agendas, blocos de notas, livros.

E nada mais simples para fazer chegar às pessoas a informação do festival do que uma aplicação móvel com todos os conteúdos chave do festival, de uma forma resumida e objetiva que pode ser atualizada a cada instante, através de uma ligação à internet.

A aplicação móvel para o Festival Literário da Madeira deverá apresentar todos os eventos do festival, dando a conhecer os locais onde se realizam, o contexto e os intervenientes dos mesmos. O intuito da aplicação é também familiarizar o público sobre os intervenientes do festival, exibindo alguma da sua história e obra. De destacar ainda, que deverão ser

referenciadas todas as entidades parceiras e patrocinadoras que tornam possível a realização do festival ano após ano.

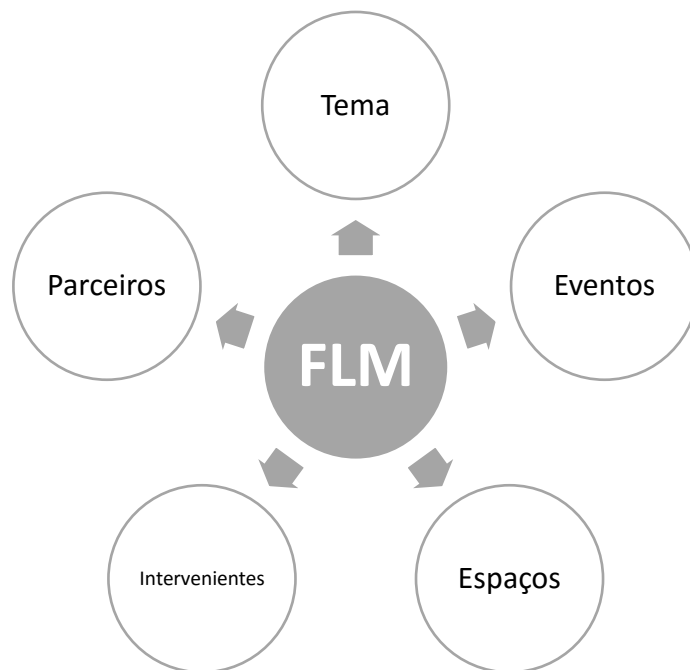


Figura I.4 - Principais focos da aplicação móvel

I.3. OBJETIVOS

Feita a contextualização da aplicação a desenvolver, são agora apresentados os principais objetivos do projeto a alcançar durante este estágio curricular:

- i. Desenvolvimento de uma aplicação móvel para *smartphones* e *tablets* que permita a consulta do roteiro/programa do Festival Literário da Madeira.
- ii. Desenvolvimento de uma plataforma *web* onde deve ser feita a gestão dos conteúdos da aplicação móvel (*backoffice*).

O desenvolvimento descrito no segundo objetivo será da responsabilidade da empresa Nova Delphi, sendo que será prestado apoio sempre que necessário, uma vez que a aplicação móvel e a plataforma *web* devem estar devidamente integradas.

Para este projeto de estágio tenho também como objetivos pessoais o colocar em prática das competências adquiridas, tanto durante o Mestrado como a Licenciatura em Engenharia Informática, contactar com um ambiente profissional e com a realidade de uma empresa, assim como a adquirir novas competências através desta nova experiência profissional.

Proponho-me também a projetar uma solução escalável através do planeamento de uma estrutura robusta, prevendo um possível crescimento da aplicação e a implementação de funcionalidades posteriores a este estágio.

I.4. ESTRUTURA DO RELATÓRIO

Este relatório está dividido em cinco grandes capítulos, correspondentes às cinco principais fases do desenvolvimento do estágio e deste projeto.

No primeiro capítulo, a **I. Introdução**, é feita a contextualização do estágio, na qual são apresentados a motivação pela escolha de um estágio curricular, uma breve descrição sobre a empresa Nova Delphi e a necessidade da criação de uma aplicação móvel para o Festival Literário da Madeira. Também serão mencionados os objetivos deste projeto.

No capítulo seguinte, o **II. Estado da Arte** serão introduzidas as várias abordagens de tecnologias multiplataforma estudadas e apresentada a tecnologia escolhida para ser utilizada neste projeto. Ainda neste capítulo serão apresentadas algumas aplicações com características semelhantes às pretendidas neste projeto.

Já no capítulo **III. Desenvolvimento da Aplicação**, serão descritos os procedimentos desde o levantamento de requisitos até ao desenho da interface gráfica, passando pela elaboração dos casos de utilização e dos diagramas de atividade e a estruturação da arquitetura e da base de dados do projeto.

No capítulo **IV. Implementação** serão mencionadas as linguagens e tecnologias utilizadas em todo o processo de implementação, assim como as funcionalidades disponibilizadas nos lançamentos das versões 1.0 e 2.0 da aplicação móvel.

As conclusões e outras apreciações sobre a aplicação móvel e sobre o estágio curricular serão descritas no capítulo **V. Conclusão**, assim como também serão apresentadas algumas perspetivas futuras, ainda em relação ao projeto.

Por fim, serão apresentadas as **Referências** e os **Anexos** referentes ao projeto e a este relatório, concretamente.

II. ESTADO DA ARTE

"640K ought to be enough for anybody."

– Bill Gates (1981)

II.1. TECNOLOGIAS MULTIPLATAFORMA

Um dos principais objetivos para o lançamento da aplicação móvel para o Festival Literário da Madeira é fazer com que o público do festival aumente, e para que o produto final possa chegar a um maior número de pessoas, as restrições e/ou incompatibilidades entre a aplicação e a marca ou o sistema operativo do dispositivo móvel de cada utilizador deverão ser mínimas ou inexistentes. De maneira a cobrir, maioritariamente, a variada gama de dispositivos móveis existentes no mercado, será necessário que a aplicação seja disponibilizada para as versões mais utilizadas dos principais sistemas operativos móveis.

Através de um raciocínio simples conseguimos concluir que os custos associados ao desenvolvimento de aplicações móveis aumentam proporcionalmente com o número de sistemas operativos que queremos abranger - e quando se fala em custos inclui-se também a necessidade de recursos físicos/humanos e de tempo. Surgirão então as tecnologias multiplataforma (em inglês *cross-platform*) cujo princípio básico consiste em reduzir tanto os recursos necessários como o tempo de desenvolvimento das aplicações para vários sistemas operativos, que nos dias de hoje, aparentam ser sempre escassos devido ao grande volume de ideias e projetos que surgem a cada instante.

No gráfico da **Figura II.1**, elaborado pela DB Best Technologies, uma companhia que desenvolve sistemas *web* e *mobile*, conhecida internacionalmente (www.dbbest.com), é apresentada uma comparação feita entre várias abordagens para o desenvolvimento multiplataforma, tendo como termo de comparação os custos de desenvolvimento, o acesso ao *hardware* e sistema operativo do dispositivo móvel e o desempenho de uma aplicação nativa.

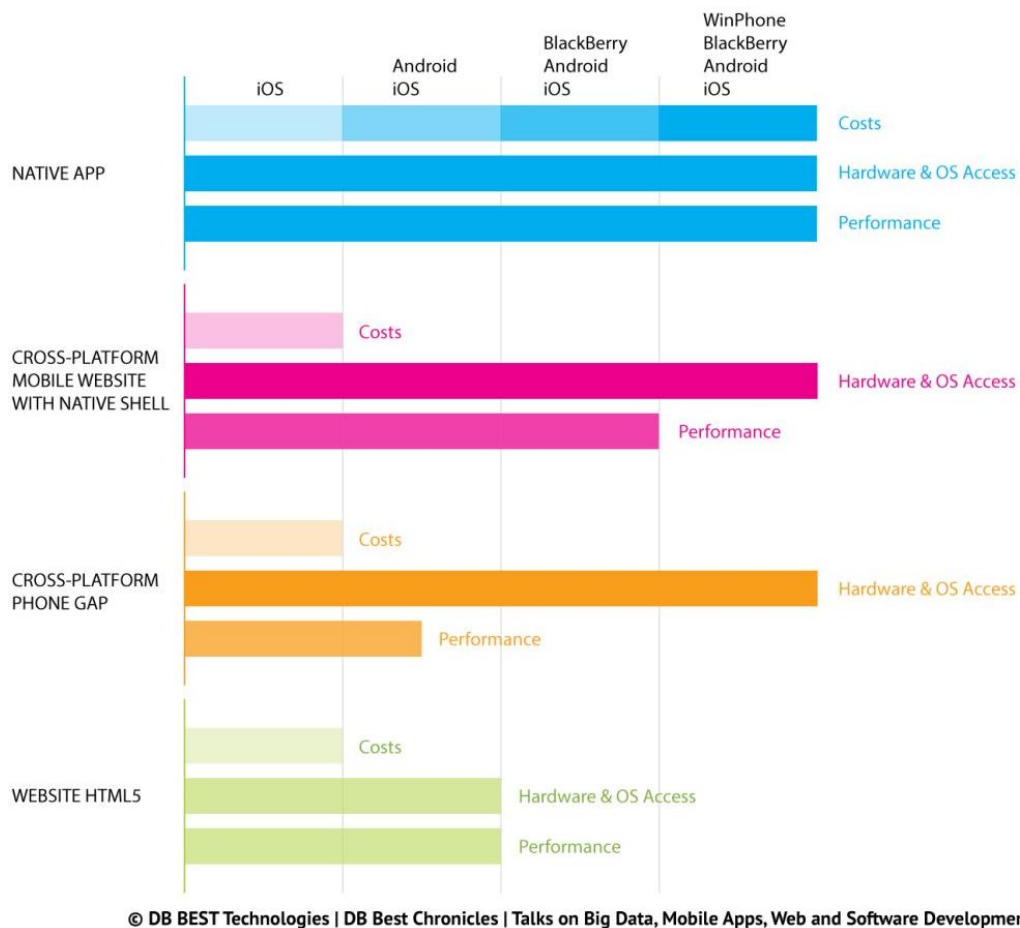


Figura II.1 - Custos, Acesso ao *Hardware* e ao *Software* e Desempenho de aplicações nativas ou desenvolvidas com tecnologias multiplataforma [2]

Através de uma análise rápida, conclui-se instantaneamente que existe uma redução dos custos de desenvolvimento de uma aplicação móvel utilizando tecnologias multiplataforma quando comparada com uma aplicação nativa. Esta redução dos recursos e do tempo necessários ao desenvolvimento ocorre, principalmente, pois a implementação do código fonte é feita utilizando a mesma linguagem de programação para os vários sistemas operativos disponíveis no mercado, evitando-se que o processo de implementação seja repetido. [2] Por outro lado, a utilização destas tecnologias pode condicionar parcialmente os níveis de desempenho das aplicações não nativas.

II.1.1. *Websites* e utilização de *shells* nativas

Das várias abordagens para o desenvolvimento multiplataforma de aplicações para dispositivos móveis, a mais simples passa pela implementação de um *website* ou aplicação *web* que poderá correr num simples *browser* do dispositivo móvel, ou então incorporado num *containers* ou *shell* nativos, permitindo que haja acesso aos recursos do dispositivo móvel, tanto a nível de *hardware* como de *software*. [2]

Os baixos custos inerentes ao desenvolvimento de *websites* contrastam com o facto de estas aplicações não estarem disponíveis em modo *offline*, o que representa a principal limitação da sua utilização. Do ponto de vista do utilizador, esta é uma das razões da maior preferência pela utilização de aplicações ao invés de *websites*, sem referir o facto de que para abrir uma aplicação basta selecioná-la, enquanto que para abrir um *website* é necessário abrir o *browser* e escrever o URL. [2]

As *shells* nativas com *websites* incorporados resolvem a questão em relação ao número de passos necessários para abrir a aplicação e permitem o acesso quase total ao *hardware* e ao *software* do dispositivo móvel, ao contrário dos *websites* que correm nos *browsers* [2], apresentando-se assim numa solução mais completa.

II.1.2. Uma única linguagem de programação, vários ambientes móveis

A ferramenta Corona SDK da Corona Labs (www.coronalabs.com) que utiliza a linguagem de programação *Lua* (www.lua.org) [3] é um exemplo de outra abordagem com um novo conceito para o desenvolvimento recorrendo a tecnologias multiplataforma.

A implementação do código fonte é realizada com a utilização de uma única linguagem de programação, que durante a sua compilação dará origem ao código nativo para cada sistema operativo pretendido: iOS, Android Windows Phone, por exemplo. [3]

Recorrendo ao mesmo exemplo, a implementação em *Lua* leva a uma diminuição significativa de linhas de código quando comparada com Java por exemplo, embora por outro lado, a sua sintaxe seja menos clara, sendo necessário algum tempo extra para a leitura e estudo da sua documentação.[3] Apesar de a curto prazo parecer que a utilização deste tipo de ferramentas seja mais demorada e trabalhosa, devido à necessidade de aprendizagem de novas tecnologias, os resultados começam realmente a surgir a médio e longo prazo, após o período de adaptação ao contexto novo ter sido ultrapassado.



Figura II.2 - Exemplo de um pequeno *script* em Lua, retirado do *site* da Corona Labs [3]

Outra desvantagem, neste caso diretamente relacionada com a ferramenta Corona SDK, é que é paga. Apesar de existir uma versão gratuita, esta não é contemplada com todas as funcionalidades que a versão paga disponibiliza.¹

II.1.3. Ferramentas mais utilizadas

Entre as ferramentas mais conhecidas e mais utilizadas para o desenvolvimento de aplicações multiplataforma encontram-se a RhoMobile, WidgwtPad, PhoneGap e Appcelerator Titanium.[4] [5] [6] [7] De uma forma resumida, será feita de seguida, uma breve descrição de cada uma destas ferramentas multiplataforma.

A RhoMobile disponibiliza uma *framework*, a Rhodes [8], que permite o desenvolvimento de aplicações nativas para um vasto conjunto de dispositivos e sistemas operativos, como Android, iOS e Windows Phone, utilizando Ruby (www.ruby-lan.org) como linguagem de programação. [5]

O WidgetPad utiliza tecnologias como HTML5, CSS3 e JavaScript para o desenvolvimento de aplicações móveis para Android e iOS e disponibiliza ainda um conjunto de ferramentas como um gestor de projecto, um editor de código e um sistema de versões. [5] [6] O WidgetPad é neste momento um projeto beta privado, cujo acesso é restrito e só pode ser concedido pelos próprios criadores.

¹ Atualmente, esta ferramenta já é disponibilizada gratuitamente

```

1  var canvas;
2  WidgetPad.ready(function() {
3  try {
4  /* To prevent scrolling, call WidgetPad.noscroll(true); */
5  WidgetPad.noscroll(true);
6
7  canvas = document.getElementById('canvas').getContext('2d');
8  var lincrad = canvas.createLinearGradient(0,0,0,600);
9  lincrad.addColorStop(0, '#000');
10 lincrad.addColorStop(1, '#888');
11 canvas.fillStyle = lincrad;
12 canvas.fillRect(0, 0, 320, 420);
13
14 /* for MobileSafari */
15 document.body.addEventListener("touchstart", paintstart, false);
16 document.body.addEventListener("touchend", paintend, false);
17 document.body.addEventListener("touchmove", paintmove, false);
18
19 /* for PC browser */
20 document.body.addEventListener("mousedown", paintstart, false);
21 document.body.addEventListener("mouseup", paintend, false);
22 document.body.addEventListener("mousemove", paintmove, false);
23 }

```

Figura II.3 - Excerto de uma função em JavaScript implementada com o WidgetPag [5]

A PhoneGap é grátis, *open-source* e uma das ferramentas mais conhecidas para o desenvolvimento de aplicações móveis. Esta ferramenta utiliza linguagens *web* como HTML e JavaScript e permite o desenvolvimento para iPhone, Android, Windows Phone, Symbian, BlackBerry, entre outro. [5][6]

À semelhança da ferramenta anterior, o Appcelerator Titanium é também grátis e *open-source*. O Titanium utiliza JavaScript como linguagem de programação que quando compilado é transformado em código nativo para Android, iOS ou Windows Phone, entre outros. Tendo em conta que uma boa aplicação multiplataforma deve manter a aparência base da plataforma na qual está a ser usada [9], utilizar o Appcelerator Titanium é realmente uma mais-valia, pois o *'look and feel'* nativo é mantido, enquanto que as aplicações desenvolvidas utilizando a PhoneGap apresentam uma interface gráfica consistente para todas as plataformas. [10][11]

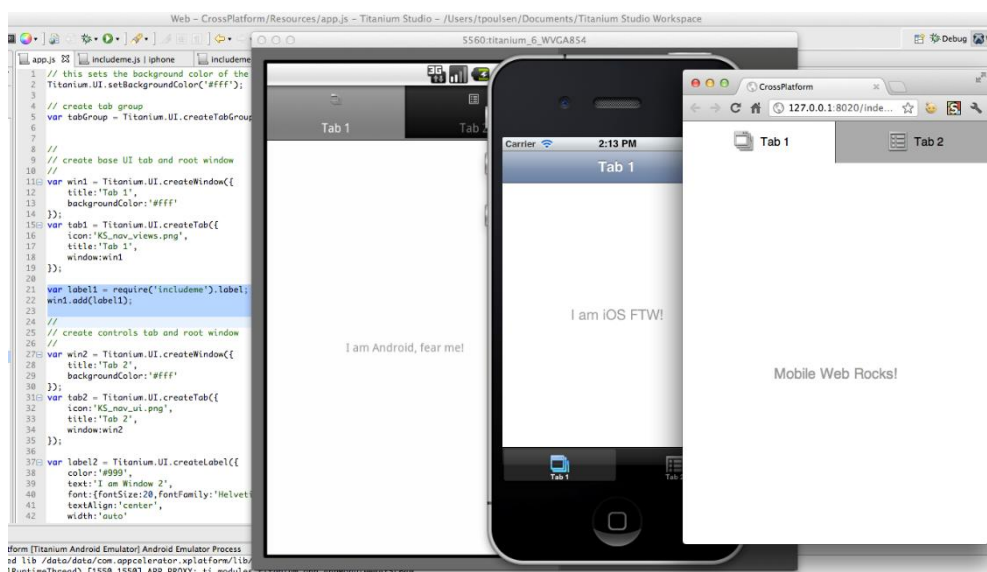


Figura II.4 - Exemplo de um componente apresentado nas diferentes plataformas [12]

II.1.4.Considerações finais/análise global

Expostas algumas das abordagens e ferramentas multiplataforma disponíveis para utilização, será feita agora uma análise de modo a concluir qual deverá ser a escolha sobre qual a ferramenta a utilizar para a implementação deste projeto, durante o estágio curricular.

Nesta análise serão tidos em conta vários critérios importantes tanto no planeamento e desenvolvimento de aplicações, como na introdução a novas tecnologias. Para que seja feito um bom planeamento, e conseqüentemente, um bom desenvolvimento dentro dos prazos estimados, é necessário ter em atenção os custos e tempos necessários para a aprendizagem de novas ferramentas e tecnologias que serão utilizadas. A análise deste primeiro critério resultará da avaliação conjunta de fatores como: o tipo de distribuição e os custos de aquisição associados às ferramentas a utilizar, uma vez que se trata de um estágio curricular e não existem fundos destinados a suportar este projeto; e as linguagens utilizadas e o conhecimento já adquirido sobre as mesmas. Será dada prioridade a ferramentas grátis que utilizem linguagens de programação que nas quais já tenha alguma experiência e conhecimento.

Outro critério é a disponibilidade da aplicação com a existência ou inexistência de conexão à internet. As aplicações que dependem de conexão à internet para o seu funcionamento têm o seu acesso limitado, uma vez que estarão indisponíveis para o utilizador em ambientes/situações em que a conexão não exista. Por outro lado, a conexão à internet permite que os conteúdos disponibilizados pela aplicação sejam dinâmicos e atualizáveis. Uma aplicação que não dependa de todo da conexão à internet será estática e rapidamente estará desatualizada. Assim, o ideal será desenvolver uma aplicação que funcione independentemente da conexão à internet, mas que recorra à mesma, pontualmente, apenas para a atualização dos conteúdos.

Da análise feita inicialmente sobre tecnologias multiplataforma, concluiu-se que algumas das abordagens existentes comprometem, em parte, o desempenho das aplicações geradas. Este fator também será tido em conta na escolha da ferramenta a usar, sendo que, idealmente será dada prioridade àquelas que proporcionarem maiores níveis de desempenho.

Por fim, o *'look and feel'* final da aplicação, que terá conseqüências na experiência do utilizador, será outro fator a ter em consideração na escolha da tecnologia e ferramenta a ser utilizada. Algumas das abordagens mencionadas anteriormente utilizam linguagens *web* que não são compiladas em código nativo e, como tal, originam aplicações que apresentam um *'look and feel'* *web*, que facilmente induzirão no utilizador a sensação de estar a navegar numa página *web* e não numa aplicação nativa.

Perante os critérios *'Custos/tempos de aprendizagem'*, *'Disponibilidade'*, *'Níveis de desempenho'* e *'Look and feel'*, será apresentada de seguida uma tabela síntese com os aspetos

positivos e negativos das várias abordagens que foram analisadas anteriormente e que pesaram na escolha da ferramenta a ser utilizada neste projeto.

Tabela II.1 - Aspetos positivos e negativos das várias abordagens analisadas

Critério Abordagem	Custos/tempos de aprendizagem	Disponibilidade	Níveis de desempenho	'Look and Feel'
Websites / Shells nativas	Linguagens <i>web</i> ★ ★ ★	Depende da conexão à internet ★ ☆ ☆	Bons níveis de desempenho ★ ★ ★	<i>Web</i> ★ ☆ ☆
Corona SDK	Ferramenta paga <i>Lua</i> ☆ ☆ ☆	Sempre disponível ★ ★ ★	Razoáveis níveis de desempenho ★ ★ ☆	Nativo ★ ★ ★
RhoMobile	Ferramenta não paga <i>Ruby</i> ★ ☆ ☆			Nativo ★ ★ ★
WidgetPad	Ferramenta com acesso restrito Linguagens <i>web</i> ★ ★ ☆			<i>Web</i> ★ ☆ ☆
PhoneGap	Ferramenta não paga Linguagens <i>web</i> ★ ★ ★			<i>Web</i> ★ ☆ ☆
Appcelerator Titanium	Ferramenta não paga <i>JavaScript</i> ★ ★ ★			Nativo ★ ★ ★

Perante os dados apresentados na tabela, primeiramente em relação ao critério 'Custos/tempos de aprendizagem', destaca-se a abordagem de *website/shell* nativa, o PhoneGap e o Appcelerator Titanium, uma vez que, além de serem gratuitas, utilizam linguagens minhas conhecidas como HTML, CSS e JavaScript, o que faria com que o tempo previsto para a aprendizagem e introdução à ferramenta fosse reduzido.

Quando falamos de disponibilidade, destacam-se pela negativa os *websites/shells* nativas uma vez que as aplicações geradas por todas as outras abordagens apresentadas não necessitam de conexão à internet para estarem disponíveis ao utilizador, embora necessitem da mesma para a atualização dos conteúdos. Segundo este critério, a opção *websites/shells* nativas seria eliminada pois pretende-se que a aplicação esteja sempre disponível para o utilizador.

Ao contrário do critério anterior, os *websites/shells* nativas destacam-se positivamente em relação às restantes abordagens quando são comparados os níveis de desempenho das aplicações. O facto de o código fonte não ser convertido, faz com que seja executado exatamente o que o desenvolvedor pretende e não uma aproximação generalizada das ações pretendidas, convertida automaticamente. Um componente gerado e compilado na sua linguagem nativa, apresentará sempre melhor desempenho e um código 'mais limpo' (partindo do princípio que o desenvolvedor é adepto de boas práticas de programação), do que outro semelhante, mas que tenha passado por processos de conversão automática de código. Assim, segundo apenas este critério, a escolha recairia sobre os *websites/shells* nativas.

Sendo o '*look and feel*' da aplicação também importante para o utilizador, destacam-se as opções Corona SDK, RhoMobile, e Appcelerator Titanium. Estas três abordagens disponibilizam aplicações com '*look and feel*' nativos, ao contrário das restantes que deixariam no utilizador uma sensação de estar a navegar num *website* ou numa página *web*.

Em suma e perante as ferramentas multiplataforma analisadas e apresentadas anteriormente, o Appcelerator Titanium destaca-se positivamente por permitir a criação de aplicações nativas com um '*look and feel*' também ele nativo. Esta vantagem traduz-se numa melhor usabilidade e experiência do utilizador, fatores que estão de acordo com as necessidades do projeto a ser desenvolvido. A principal vantagem em relação à utilização de *shells* nativas, que apresentam melhores níveis de desempenho, é o facto de poder estar disponível mesmo quando o dispositivo móvel não está conectado à internet.

Além do que foi referido anteriormente, o facto da linguagem de programação utilizada ser JavaScript representa uma mais-valia a nível pessoal, devido aos conhecimentos já adquiridos sobre esta linguagem.

II.2. APLICAÇÕES SEMELHANTES

Existem no mercado vários tipos de exemplares de aplicações móveis para festivais culturais, tanto literários, de cinema, teatro ou música, que apresentam alguns aspetos e funcionalidades semelhantes aos pretendidos para o projeto. Antes de dar início ao desenvolvimento da aplicação, foi feita uma pesquisa de aplicações do mesmo género e uma recolha dos principais elementos que as compõem, com o intuito de conhecer melhor o âmbito em que se inserem e o que transmitem ao utilizador. Algumas das aplicações pesquisadas foram sugeridas pelo Eng. Hélder, uma vez que já eram do conhecimento da Nova Delphi, por serem de festivais literários internacionais.

Como já foi referido, foram essencialmente pesquisadas aplicações de festivais literários, de cinema ou teatro e de música por se aproximarem mais do conceito da aplicação a desenvolver para o Festival Literário da Madeira, uma vez que a organização e estrutura deste tipo de festivais têm vários pontos em comum, como por exemplo, a lista de eventos em vários locais, nos quais participam vários convidados. De seguida são apresentadas, das várias aplicações analisadas, aquelas, que para mim, apresentam os elementos que mais se destacaram.

II.2.1. Festival de Cannes 2014

Desenvolvida para o Festival de Arte Cinematográfica de Cannes, esta aplicação está disponível em francês e inglês, é suportada apenas em *landscape* e reúne um conjunto de fotos, vídeos, gravações de áudio e artigos que podem ser partilhas nas redes sociais. [13]



Figura II.5 - Logo da aplicação móvel Festival de Cannes 2014

Nesta aplicação destaca-se o uso abundante de imagens acompanhadas de legendas sucintas, que encaminham o utilizador para os vários conteúdos disponíveis.

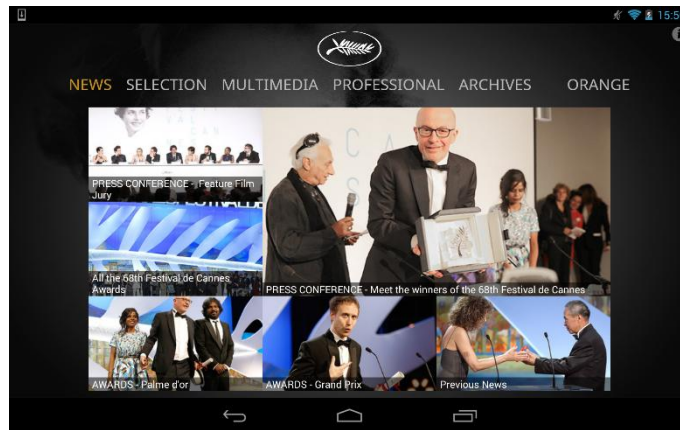
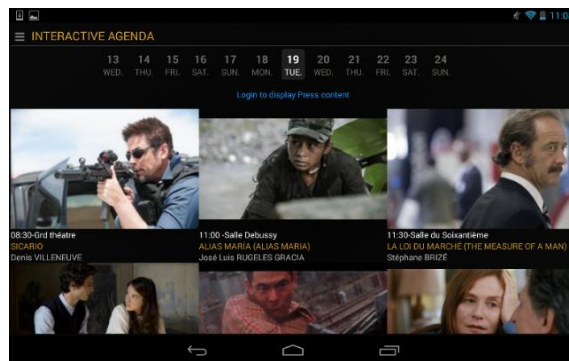
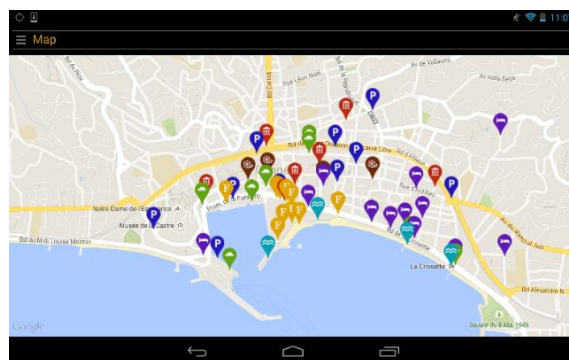


Figura II.6 - Ecrã inicial da aplicação móvel Festival de Cannes 2014

É disponibilizado uma agenda com o programa de eventos dos vários dias do festival e um mapa no qual são sinalizados vários locais do festival. A aplicação permite ainda consultar os conteúdos das edições anteriores do festival.



a)



b)

Figura II.7 - a) Programa de eventos a) e b) mapa com os locais do Festival de Cannes 2014

II.2.2. Festivaletteratura

Esta aplicação desenvolvida para o Festival de Literatura em Mantua (Mantova em italiano), Itália, que se realiza todos os anos em setembro apresenta, entre outros conteúdos, o programa de eventos, autores convidados, locais dos eventos, bilheteiras, e ainda informações úteis de como chegar aos locais.



Figura II.8 - Logo da aplicação móvel Festivaletteratura

O programa de eventos é apresentado sob a forma de lista que é filtrada por dia e por parte do dia (manhã, tarde ou noite). Os eventos apresentam, por vezes, ícones cuja legenda pode ser consultada à direita. Através desta legenda, podemos identificar, por exemplo, quais os eventos destinados a crianças ou adultos, eventos gratuitos, entre outros.



Figura II.9 - Programa de eventos da aplicação móvel Festivaletteratura

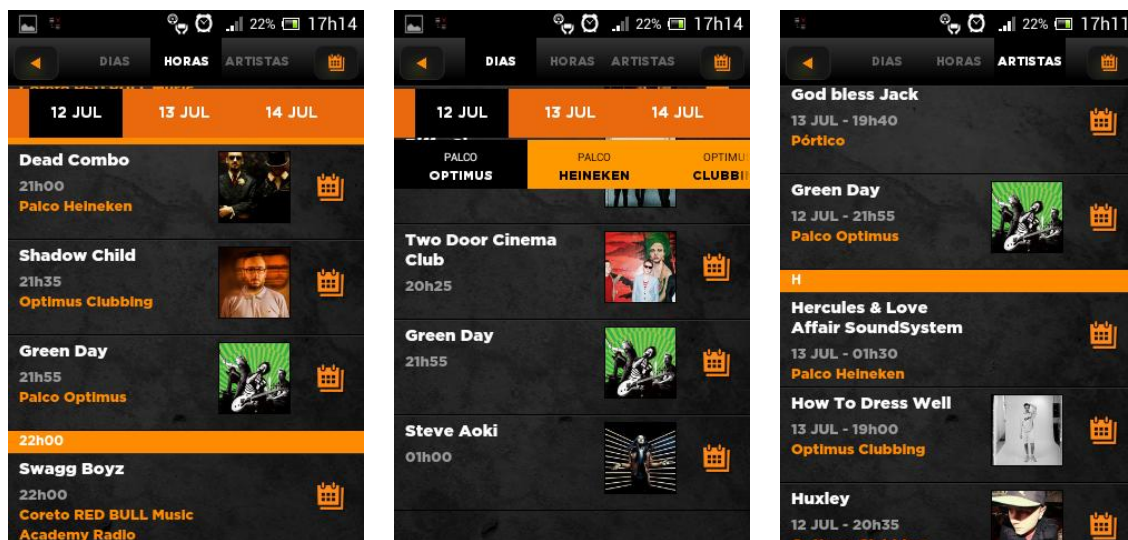
II.2.3.Optimus Alive 2013

O Optimus Alive, agora NOS Alive, acontece todos os anos em Oeiras, desde 2007, e durante 3 dias sobem ao palco bandas musicais nacionais e internacionais. Na aplicação para a edição de 2013 era apresentado o cartaz, o recinto, notícias e, entre outros, informações úteis de como chegar, onde ficar, FAQ, etc.



Figura II.10 - Logo da aplicação móvel Optimus Alive 2013

Os artistas são apresentados numa lista ordenada alfabeticamente, na qual se pode consultar a data, hora e palco do concerto. Os concertos podem ainda ser listados por dia e palco ou por dia e hora.



a)

b)

c)

Figura II.11 - a) Programa de concertos do Optimus Alive 2013 organizados por dia, b) palco e c) artista

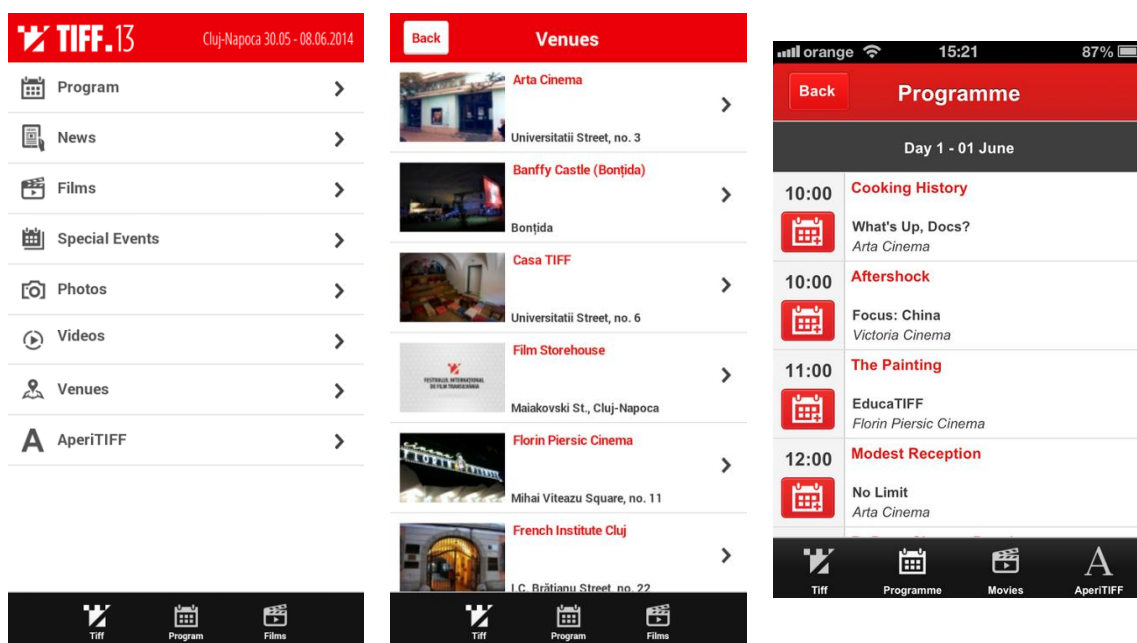
II.2.4. Tiff 2013

O TIFF é o primeiro festival internacional de cinema da Roménia e tem como principal objectivo a promoção da arte cinematográfica.



Figura II.12 - Logo da aplicação móvel TIFF 2013

Na aplicação móvel é possível consultar o programa de eventos do festival que podem ser filtrados por dia, tipo ou local, notícias relacionadas e uma galeria de fotos dos eventos, entre outros, sendo que os conteúdos são maioritariamente apresentados sob a forma de lista.



a)

b)

c)

Figura II.13 - a) Menu inicial, b) notícias dos pontos principais e c) programa de eventos da aplicação móvel TIFF 2013

II.2.5. Feeling Madeira

Além do estilo convencional de aplicações para festivais, foram também pesquisados outro tipo de aplicações que disponibilizam funcionalidades específicas que podem ser introduzidas no contexto da aplicação do Festival Literário da Madeira. Exemplo disso é a aplicação Feeling Madeira, que além de disponibilizar uma série de informação sobre promoções e serviços da região, pode também ser usada como guia de eventos e de locais de interesse da Madeira.



Figura II.14 - Logo da aplicação móvel Feeling Madeira

De destacar nesta aplicação, nas secções de Eventos e Aqui Perto, o programa de eventos que podem ser consultados sobre a forma de lista ou de pontos dispostos sobre o mapa da região.



a)



b)

Figura II.15 - a) Mapa e b) programa de eventos da aplicação móvel Feeling Madeira

II.2.6. Considerações finais/análise global

Após uma breve análise das aplicações apresentadas anteriormente, retenho três elementos principais que são comuns a quase todas as aplicações analisadas: conteúdos organizados e apresentados sob a forma de lista, nomeadamente o programa de eventos; utilização de elementos gráficos (cores ou ícones) que diferenciam os tipos de eventos (isto nas aplicações que divulgam tipos de eventos diferentes); e por fim, a disponibilização da localização geográfica dos vários eventos, através da utilização de um mapa.

A utilização de listas para a disponibilização de conteúdos facilita bastante na organização da informação que se pretende apresentar em cada um dos elementos da mesma. Por ser atribuído a cada elemento da lista um espaço limitado, a disposição da informação a apresentar tem de ser sintetizada e objetiva, de modo a apresentar ao utilizador apenas o essencial.

Ainda relacionado com o ponto anterior, e de modo a diferenciar os vários tipos de eventos, surgem por vezes pequenos elementos gráficos, nomeadamente, ícones ou cores variados, tanto nas listas como nos mapas, que facilitam ao utilizador a identificação do tipo de eventos, sem que para tal seja apresentada a designação do mesmo. É do conhecimento geral que uma imagem chega e é processada mais rapidamente pelo cérebro humano do que informação textual. Logo, se os ícones utilizados foram bastante intuitivos, além de ganharmos espaço físico na nossa interface gráfica (porque normalmente o texto ocupa mais espaço do que um simples ícone), conseguimos com que facilmente o utilizador entenda o que queremos transmitir, sem que este tenha de realizar muito esforço.

E por se tratar de uma aplicação que pretende apresentar vários eventos que não se concentram num único local, é importante informar o utilizador da localização de cada um dos diferentes locais. Nada mais intuitivo do que apresentar um mapa da região, sinalizado a posição atual do utilizador e a localização do evento pretendido.

Posto isto, estes serão alguns dos elementos que serão tidos em conta aquando do processo de desenvolvimento da aplicação do Festival Literário da Madeira, descrito no próximo capítulo.

III. DESENVOLVIMENTO DA APLICAÇÃO

“Programming without an overall architecture or design in mind is like exploring a cave with only a flashlight: You don’t know where you’ve been, you don’t know where you’re going, and you don’t know quite where you are.”

– Danny Thorpe

III.1. INTRODUÇÃO

O desenvolvimento de *software* pressupõe um conjunto de etapas preliminares à implementação propriamente dita, que têm como objetivo responder as questões como 'O quê?', 'Porquê?', 'Quem?', 'Quando?' e 'Como?'. O desenvolvimento da aplicação para o Festival Literário da Madeira fará parte de um processo iterativo e incremental, através do qual se pretenderá encontrar as respostas a todas estas questões, passando por várias etapas: o levantamento de requisitos, a elaboração de casos de utilização e diagramas de atividade, o estudo da arquitetura da aplicação e da estrutura da base de dados e, por fim, a criação da interface do utilizador.

Nesta fase inicial será fundamental a permuta de ideias com aos responsáveis da empresa, através da troca de *e-mails*, para questões mais pontais, ou através de reuniões presenciais para a discussão e obtenção de *feedback* dos resultados produzidos até então. Sendo apologista da máxima 'dividir para conquistar', acredito que a melhor abordagem será dividir o projeto em partes/módulos e desenvolvê-los à vez, concentrando todos os esforços em cada um deles. Daí dizer que o processo será interativo e incremental.

É fundamental desde cedo identificar e compreender qual o propósito do produto a desenvolver, respondendo a questões como 'O quê?', 'O que deve ser?' e 'O que deve fazer?'. Estas questões serão desvendadas durante o Levantamento de Requisitos através de reuniões e de entrevistas aos vários representantes das várias áreas do festival, indicados pelo Eng. Hélder, assim como da leitura de alguma documentação existente sobre os festivais de outros anos. A partir da análise destes elementos serão identificadas as funcionalidades/competências e características pretendidas para a aplicação, as respostas aos 'Porquês?'.

Identificados os requisitos e principais funcionalidades pretendidas para a aplicação, o passo seguinte consistirá em identificar e definir quais as sequências dos passos e procedimentos necessários para as ações que poderão ser executadas para as funcionalidades da aplicação, assim como quem as deverá executar. Estas ações serão descritas através dos casos de utilização, e a questão 'Quem?' permitirá que os respetivos atores sejam identificados. Através da elaboração dos diagramas de atividade serão definidas as sequenciais temporais dos passos que constituem cada ação, obtendo respostas à questão 'Quando?'.

Sendo esta uma aplicação que funcionará em cooperação com outros elementos, nesta fase começarão a surgir questões como 'Como será a comunicação entre os vários elementos do sistema?' que ajudará na construção da arquitetura do sistema e a questão 'Como serão

armazenados os dados?' que levará à estruturação da base de dados. Por fim, será definida a interface do utilizador cujos protótipos de alta fidelidade serão elaborados pela designer da empresa, Dr. Andreia Encarnação.

III.2. LEVANTAMENTO DE REQUISITOS

Tendo como objetivo o início do levantamento de requisitos da aplicação móvel a desenvolver, foram marcadas várias entrevistas com os representantes de cada uma das áreas da empresa diretamente relacionadas com este projeto, com o intuito de serem recolhidas as primeiras impressões sobre os conteúdos e funcionalidades pretendidos para a aplicação, que foram, mais tarde, traduzidos em requisitos.

De seguida será feita referência às entrevistas que, previamente preparadas, decorreram todas elas nas instalações da empresa, sendo que o respetivo áudio foi gravado e, posteriormente, analisado, juntamente com as notas tiradas. A preparação de cada entrevista teve por base uma pequena pesquisa sobre cada área da empresa e a elaboração de um guião com várias questões a colocar aos representantes entrevistados.

III.2.1. Entrevistas

Como foi referido, anteriormente, cada entrevista teve um foco especial tendo em conta área da empresa em questão e as funções desempenhadas pelos entrevistados. Foram entrevistados membros da direção do FLM, responsáveis pela distribuição e contactos com as livrarias, responsáveis pela organização e logística do FLM, e responsáveis pela parte editorial, comunicação e gestão de autores.



Figura III.1 - Sequência de entrevistas realizadas para levantamento de requisitos

Na primeira entrevista, realizado no dia 18 de outubro de 2013, foi-me apresentado, pelo Dr. Francesco Valentini, membro da direção do FLM, o contexto no qual está inserido o Festival Literário da Madeira. Foram abordados vários pontos como o conceito geral do festival, o programa de eventos, os locais onde ocorrem, os tipos de parceiros do festival, a diferença entre os macroeventos e os microeventos, os tipos de eventos existentes: conferências, mesas, eventos

musicais, exposições de artes plásticas, entre outros. Falou-se também de a necessidade da aplicação ser proativa, no sentido em que deve notificar o utilizador sobre os conteúdos que são do seu interesse, sem que tenha que ser este a tomar a iniciativa.

Totalmente virada para a área de distribuição e vendas, a segunda entrevista que decorreu no dia 25 de outubro de 2013, cujos intervenientes foram o Dr. Rúben Santos e João Lotscher, responsáveis pela distribuição e contactos com as livrarias. Falou-se, principalmente, sobre quais os canais de vendas da empresa, da alteração do fluxo de vendas durante o festival, os tipos de parcerias com editoras e livrarias e, por fim, a ponderação da venda e reserva de livros físicos e digitais através da aplicação móvel e que outros procedimentos esta funcionalidade implicaria.

Na terceira entrevista, que aconteceu no dia 1 de novembro de 2013, foram postos sobre a mesa vários pontos relacionados com as parcerias e trocas de serviços, dos quais se destacou a diferenciação entre os parceiros do festival e os associados do festival e respetivas responsabilidades. Na segunda parte da entrevista, falou-se sobre a ideia de ser desenvolvido um roteiro turístico literário que reúna locais marcantes para autores nacionais e internacionais, que passaram pela região.

Na última entrevista, realizada no dia 20 de novembro de 2013, foram focados dois pontos distintos do projeto: o Festivalinho Literário Infantil (FLI) e os associados do festival. O FLI pretende a aproximação entre os mais novos e os livros através da realização de eventos tanto nas escolas para as crianças, como conferências destinadas aos pais. Em relação aos associados do festival, foi abordada uma outra vertente desejada para a aplicação, que visa a disponibilização de informações sobre vários serviços úteis para o utilizador que se encontra de passagem, sozinho ou acompanhado da família, e não só, passando a aplicação móvel de um simples produto a um serviço fundamental.

III.2.2.Principais funcionalidades

As primeiras impressões retiradas após as entrevistas demonstraram que dada a versatilidade de funcionalidades ambicionadas pelos membros da empresa para a aplicação móvel, a melhor opção seria um desenvolvimento por etapas e a organização destas funcionalidades por módulos. Ou seja, cada módulo representa um conjunto de funcionalidades relacionadas entre si que dizem apenas respeito a um dos âmbitos do projeto.



Figura III.2 - Representação da divisão do projecto em módulos

Assim, fazem parte do módulo referente ao Festival Literário da Madeira todas as funcionalidades relacionadas com a consulta do programa do festival, assim como a pesquisa, filtragem e navegação entre conteúdos, como eventos, locais, participantes e parceiros. Está também incluído neste módulo a implementação do *feed* de atualizações do festival e respetivas notificações.

Outro módulo reúne todas as funcionalidades referentes aos associados do festival como a pesquisa por categorias, a disponibilização das vantagens oferecidas e informação sobre a área de negócio, os contactos e a informação da localização.

Os restantes módulos incluem, cada um, as funcionalidades das restantes ideias para a aplicação móvel. Assim, todas as funcionalidades relacionadas com o roteiro turístico/literário e navegação entre os respetivos locais constituem um módulo e, por fim, as funcionalidades relacionadas com a venda e reserva de livros irão constituir o último módulo previsto.

III.2.3.Requisitos da aplicação

Após a análise aprofundada das informações recolhidas nas entrevistas, que se traduzem nas ambições dos membros da empresa, foi elaborado um conjunto de requisitos que devem ser tidos em conta aquando do desenvolvimento da aplicação. Os requisitos de uma aplicação representam ações, condições ou necessidades que o sistema a desenvolver deverá satisfazer. [14]

Os requisitos foram classificados consoante o nível de prioridade de implementação, tendo em conta a vontade da empresa e outros fatores como custos ou recursos disponíveis. Assim sendo, foi atribuída prioridade Alta aos requisitos considerados mais importantes e que as primeiras versões da aplicação devem abranger, e prioridade Baixa, aos requisitos menos importantes que podem vir a ser implementados apenas em versões futuras da aplicação. Os requisitos de prioridade Média dependem apenas de fatores como custos ou recursos disponíveis para serem ou não implementados.

A acrescentar apenas que as seguintes listas de requisitos funcionais e não funcionais fizeram parte de um documento de requisitos que foi submetido a análise por parte da empresa, que pode ser consultado no final deste relatório, nos anexos em **Anexo 1. Documento de Requisitos**.

III.2.3.1.Requisitos Funcionais

Os requisitos funcionais descrevem as acções (ou funções) que o sistema deverá satisfazer [14] descrevendo as funcionalidades, os cálculos, a manipulação ou processamento de dados que este deve ser capaz de executar. Nesta secção, os requisitos estão organizados segundo os módulos aos quais pertencem, como se pode observar de seguida.

Tabela III.1 - Requisitos funcionais do Módulo I – Guia do FLM e respetivas prioridades

MÓDULO I - Guia do FLM	PRIORIDADE
RF1 - A aplicação deve permitir ao utilizador a visualização do programa de eventos do festival, ordenado cronologicamente.	Alta ★ ★ ★
RF2 - A aplicação deve permitir ao utilizador pesquisar os eventos do festival e filtrá-los por categorias/tipos, por local ou por participantes convidados.	Média ★ ★ ☆
RF3 - A aplicação deve disponibilizar informação relativa a cada evento do festival, como tema, título, descrição, local, data/horário e participantes convidados.	Alta ★ ★ ★
RF4 - A aplicação deve diferenciar visualmente os eventos do FLI dos restantes eventos do FLM.	Alta ★ ★ ★
RF5 - A aplicação deve informar sobre qual o público-alvo de cada evento do festival.	Média ★ ★ ☆
RF6 - A aplicação deve disponibilizar informação sobre o preço e locais onde podem ser comprados os bilhetes para os eventos pagos.	Baixa ★ ☆ ☆
RF7 - A aplicação deve disponibilizar ao utilizador a possibilidade de efectuar a reserva de bilhetes para os eventos pagos.	Baixa ★ ☆ ☆
RF8 - A aplicação deve disponibilizar a lista dos locais onde decorrem os eventos do festival.	Alta ★ ★ ★

RF9 - A aplicação deve permitir ao utilizador visualizar uma pequena descrição, a localização no mapa e informações úteis sobre cada local, como o <i>link</i> para a página do local, os contactos e como lá chegar.	Alta ★ ★ ★
RF10 - A aplicação deve disponibilizar informação relativa a cada participante convidado do festival, como foto, uma pequena biografia sobre o participante convidado e os seus principais feitos/obras publicadas, acompanhados de <i>links</i> para sítios que complementem esta informação.	Alta ★ ★ ★
RF11 - A aplicação deve disponibilizar, para cada participante convidado, a lista de eventos do festival nos quais este participa.	Alta ★ ★ ★
RF12 - A aplicação deve disponibilizar um feed de actualizações das informações relativas ao programa do festival.	Média ★ ★ ☆
RF13 - A aplicação deve notificar o utilizador sempre que seja feita alguma actualização em relação ao programa do festival.	Média ★ ★ ☆
RF14 - A aplicação deve permitir ao utilizador configurar a periodicidade das notificações do feed de actualizações.	Baixa ★ ☆ ☆
RF15 - A aplicação deve disponibilizar uma lista dos parceiros e promotores do festival.	Alta ★ ★ ★
RF16 - A aplicação deve disponibilizar informação sobre cada parceiro: pequena descrição, informações úteis (localização, site).	Alta ★ ★ ★

Tabela III.2 - Requisitos funcionais do Módulo II - Associados do Festival e respetiva prioridade

MÓDULO II - Associados do Festival	PRIORIDADE
RF17 - A aplicação deve disponibilizar uma lista dos Associados do Festival.	Baixa ★ ☆ ☆
RF18 - A aplicação deve permitir ao utilizador pesquisar os associados, ordená-los alfabeticamente e por classificação, e filtrá-los por categorias consoante a sua área de negócio.	Baixa ★ ☆ ☆
RF19 - A aplicação deve permitir a classificação de cada associado por parte	Baixa

do utilizador.	★ ☆ ☆
RF20 - A aplicação deve indicar os associados mais próximos geograficamente do local de cada evento.	Baixa ★ ☆ ☆
RF21 - A aplicação deve disponibilizar informação sobre as vantagens que cada associado proporciona.	Baixa ★ ☆ ☆
RF22 - A aplicação deve disponibilizar a localização e informações sobre como chegar ao local onde se pode usufruir dos serviços que cada associado proporciona.	Baixa ★ ☆ ☆
RF23 - A aplicação deve disponibilizar os contactos de cada associado assim como o <i>link</i> para a respectiva página externa.	Baixa ★ ☆ ☆

Tabela III.3 - Requisitos funcionais do Módulo III - Roteiro Turístico/Literário e respetivas prioridades

MÓDULO III - Roteiro Turístico/Literário	PRIORIDADE
RF24 - A aplicação deve disponibilizar uma lista dos principais locais turísticos da região, organizados por categorias.	Baixa ★ ☆ ☆
RF25 - A aplicação deve permitir ao utilizador pesquisar os locais do roteiro, ordená-los alfabeticamente e filtrá-los por tipo.	Baixa ★ ☆ ☆
RF26 - A aplicação deve apresentar uma pequena descrição local.	Baixa ★ ☆ ☆
RF27 - A aplicação deve apresentar ao utilizador a opção 'ver mais' e redirecioná-lo para respectiva página no <i>site</i> .	Baixa ★ ☆ ☆
RF28 - A aplicação deve apresentar uma lista de locais em destaque, sugeridos pelos responsáveis pelo roteiro.	Baixa ★ ☆ ☆

Tabela III.4 - Requisitos funcionais do Módulo IV - Loja e respetivas prioridades

MÓDULO IV - Loja	PRIORIDADE
RF29 - A aplicação deve permitir ao utilizador a compra de livros digitais ou	Baixa

físicos e a reserva de livros físicos.	★ ☆ ☆
RF30 - A aplicação deve disponibilizar uma lista de livros disponíveis para compra.	Baixa ★ ☆ ☆
RF31 - A aplicação deve permitir ao utilizador pesquisar os livros disponíveis, ordená-los alfabeticamente ou por preço e filtrá-los por tipo ou autor.	Baixa ★ ☆ ☆
RF32 - A aplicação deve permitir ao utilizador consultar informação sobre cada livro, como um pequeno resumo, o(s) autor(es), a capa e o preço.	Baixa ★ ☆ ☆

III.2.3.2.Requisitos Não Funcionais

Os requisitos não funcionais referem-se a aspectos transversais, normalmente aplicáveis a todo o sistema, tais como: desempenho, robustez, fiabilidade, distribuição, segurança, integração com a Internet, facilidade de manutenção ou usabilidade [14]. Neste relatório, os requisitos não funcionais estão organizados nas categorias de Portabilidade, Acessibilidade, Disponibilidade, Manutenção e Escalabilidade, sendo que, podem existir outras.

Tabela III.5 - Requisitos não funcionais – Portabilidade e respetivas prioridades

PORTABILIDADE	PRIORIDADE
RNF1 - A aplicação deve ser desenvolvida para Android (2.3.3 ou superiores) e iOS (7).	Média ★ ★ ☆
RNF2 - A aplicação deve ser desenvolvida para <i>smartphones</i> e <i>tablets</i> .	Alta ★ ★ ★

Tabela III.6 - Requisitos não funcionais – Usabilidade e respetivas prioridades

USABILIDADE	PRIORIDADE
RNF3 - A aplicação deve ser multilingue.	Média ★ ★ ☆

Tabela III.7 - Requisitos não funcionais – Acessibilidade e respetivas prioridades

ACESSIBILIDADE	PRIORIDADE
----------------	------------

RNF4 - Algumas funcionalidades da aplicação requerem que o dispositivo esteja conectado à internet.	Média ★ ★ ☆
RNF5 - Algumas funcionalidades da aplicação requerem acesso à tecnologia GPS.	Média ★ ★ ☆

Tabela III.8 - Requisitos não funcionais – Disponibilidade e respetivas prioridades

DISPONIBILIDADE	PRIORIDADE
RNF6 - A aplicação deve funcionar mesmo que o dispositivo não se encontre conectado à internet.	Alta ★ ★ ★
RNF7 - A aplicação deve disponibilizar sempre os conteúdos relacionados com o festival, como programa previsto, informação sobre os eventos, locais, participantes convidados e parceiros, sem que seja necessária conexão à internet.	Alta ★ ★ ★
RNF8 - A aplicação necessita de conexão à internet para efectuar as actualizações, assim como para permitir ao utilizador realizar reservas e classificações.	Média ★ ★ ☆

Tabela III.9 - Requisitos não funcionais – Manutenção e respetivas prioridades

MANUTENÇÃO	PRIORIDADE
RNF9 - A aplicação deve disponibilizar informação atualizada que deverá ser gerida pelos respetivos responsáveis.	Alta ★ ★ ★

Tabela III.10 - Requisitos não funcionais – Escalabilidade e respetivas prioridades

ESCALABILIDADE	PRIORIDADE
RNF10 - A aplicação deve suportar toda a informação relacionada com o festival e suportar futuras extensões.	Alta ★ ★ ★

Os requisitos anteriormente apresentados incluem já as alterações que foram realizadas tendo por base a análise realizada por parte da empresa ao documento de requisitos submetidos - **Anexo 1. Documento de Requisitos**. Por fim, após uma análise conjunta, alguns requisitos foram descartados, outros apenas corrigidos, surgiram alguns por aglutinação de

outros já existentes e foram ainda acrescentados novos requisitos. As principais alterações a que os requisitos funcionais e não funcionais foram sujeitos são apresentadas de seguida.

Assim sendo, começando pelos requisitos funcionais, o **RF2**, que inicialmente previa apenas a ordenação dos eventos do festival cronologicamente, por local ou por participante convidado, foi alterado, modificando-se para filtragem por local ou por participante convidado, mantendo a ordenação cronológica, uma vez que faria mais sentido. O **RF18**, relativo aos associados do festival, sofreu uma alteração semelhante.

Ao **RF10** apenas acrescentou-se a indicação de que os principais feitos/obras de cada participante convidado deveriam ser acompanhados de *links* para sítios externos que completem aquela informação. O **RF23**, relativo a cada associado do festival, sofreu também uma alteração semelhante.

Em relação ao **MÓDULO IV - Loja**, foram removidos alguns requisitos que não se apresentavam claros sobre alguns aspetos que necessitam ainda de serem analisados e estudados, como a sugestão ou o destaque de livros por parte da aplicação baseados em critérios que ainda não foram pensados ou definidos.

Por fim e de forma a complementar e a clarificar as necessidades apresentadas pela empresa, foram ainda adicionados os requisitos funcionais **RF19** no **MÓDULO II - Associados do Festival**, **RF25** e **RF28** no **MÓDULO III - Roteiro Turístico** e, finalmente, **RF31** no **MÓDULO IV - Loja**.

Em relação aos requisitos não funcionais, poucas foram as alterações, sendo que do **RNF1** foi removida a plataforma Windows Phone, uma vez que não visto como objetivo da empresa, pelo menos nesta fase inicial da aplicação e alterada a versão da plataforma Android.

As seguintes fases do processo de desenvolvimento da aplicação incidiram principalmente nos requisitos com maior prioridade e naqueles cujos recursos necessários estavam disponíveis.

III.3. CASOS DE UTILIZAÇÃO

Após a conclusão da análise dos requisitos da aplicação, teve lugar a elaboração dos casos de utilização. Os casos de utilização consistem na representação do conjunto de ações que os atores podem executar num sistema.

III.3.1. Atores

Neste contexto, um ator representa um papel que um utilizador desempenha relativamente ao sistema em análise, sendo que o conjunto total de atores reflete todos os elementos que interatuam com o sistema.[14] Os atores da aplicação a desenvolver são:

Utilizador - utilizador que descarrega a aplicação para o seu *smartphone* ou *tablet*, instala-a e utiliza-a.

Aplicação - a aplicação móvel é responsável pelo processamento das ações solicitadas pelo utilizador.

Administrador - gestor dos conteúdos inseridos na plataforma *web* que serve de *backoffice* da aplicação móvel.

Backoffice - processa os conteúdos inseridos pelo administrador e responde aos pedidos da aplicação.

III.3.2. Diagrama de casos de utilização

Os diagramas de casos de utilização descrevem a relação entre atores e casos de utilização, representando a visão do sistema na perspetiva dos seus utilizadores.[14] De seguida é apresentado o diagrama de casos de utilização com os principais casos de utilização provenientes dos requisitos apresentados anteriormente.

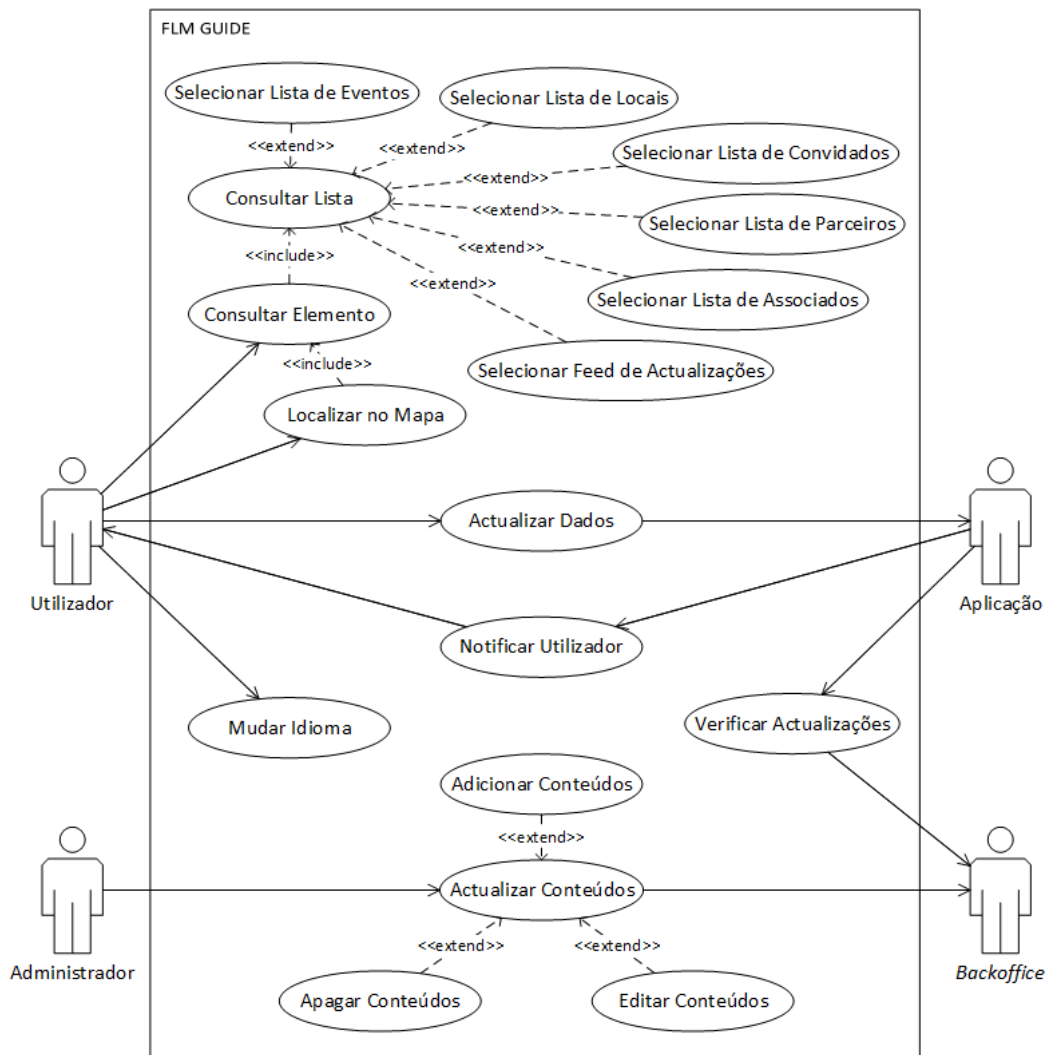


Figura III.3 - Diagrama de casos de utilização

Para alguns dos casos de utilização presentes no diagrama anterior, foi feita a especificação textual dos mesmos, indicando quais os cenários possíveis para cada um.

Especificação textual do caso de utilização “Consultar Elemento”

<p>Nome: Consultar Elemento</p> <p>Cenário</p> <p><u>Incluir caso de utilização “Consultar Lista de Elementos”</u>. Percorrer a lista de elementos escolhida através de <i>scroll</i> vertical. Escolher o elemento pretendido através da seleção do mesmo.</p>

O caso de utilização “Consultar Elemento” representa os possíveis casos de utilização “Consultar Evento”, “Consultar Local”, “Consultar Convidado”, “Consultar Parceiro” e “Consultar Associados” que contêm cenários semelhantes. O resultado deste caso de utilização

consistirá na apresentação de um elemento do mesmo tipo que a lista que tenha sido consultada, como é explicado na próxima especificação textual.

Especificação textual do caso de utilização “Consultar Lista”

Nome: Consultar Lista

Cenário Principal

Após o *splash screen* da aplicação desaparecer, selecionar de entre a lista de eventos, a lista de locais, a lista de participantes e a lista de parceiros, a pretendida.

Cenário Alternativo

Selecionado um elemento de uma das listas, selecionar a lista de outro tipo de elementos relacionado com o primeiro.

O **Cenário Alternativo** do caso de utilização “Consultar Lista” pretende representar as seguintes situações: 1) para cada evento selecionado será apresentada a lista de convidados participantes; 2) para cada convidado selecionado será apresentada a lista de eventos nos quais este participa; 3) para cada local selecionado será apresentada a lista de eventos que decorrem no mesmo.

Especificação textual do caso de utilização “Atualizar Dados”

Nome: Atualizar Dados

Cenário

O utilizador acede ao menu de opções, seleciona a opção para atualizar os dados da aplicação. É verificada a existência de ligação à Internet. Se não houver ligação, o utilizador é alertado. Caso exista ligação, a atualização de dados é solicitada. Consoante a resposta recebida pela aplicação, o utilizador é alertado caso não existam dados a atualizar ou pode então confirmar a atualização.

Cenário Alternativo 1

Idêntico ao Cenário Principal. Em qualquer momento, antes de confirmar a atualização, o utilizador pode cancelar toda a ação.

Especificação textual do caso de utilização “Verificar Atualizações”

Nome: Verificar Atualizações

Cenário

Caso exista ligação à Internet, é feito um pedido ao servidor remoto para verificar se existem dados novos. O servidor emite uma resposta: caso a resposta seja negativa, a ação é terminada; se a resposta for positiva, o servidor pode receber novo pedido da aplicação a solicitar os dados novos. Caso isso aconteça, os dados são enviados para a aplicação que os guarda na sua base de dados local.

Como foi referido no **Cenário** do caso de utilização “Verificar Atualizações”, é feita comunicação entre a aplicação móvel e o servidor seguindo uma arquitetura Cliente-Servidor, que será abordada mais à frente neste relatório.

Especificação textual do caso de utilização “Notificar Utilizador”

Nome: Notificar Utilizador

Cenário

Em cada dia do festival, o utilizador será notificado através de um alerta de quanto eventos estão agendados para o dia. Ao clicar na notificação, o utilizador será redirecionado para a lista de eventos da aplicação, com a data atual selecionada.

III.4. DIAGRAMAS DE ATIVIDADE

Elaborados os casos de utilização, seguem-se, nesta secção, os diagramas de atividade de alguns casos de utilização da aplicação. Os diagramas de atividade descrevem o comportamento de um processo ou função pela especificação da sequência de operações e decisões que permitem determinar quando e como elas são realizadas. [14] Dos casos de utilização anteriormente apresentados foram escolhidos alguns para proceder à elaboração dos seguintes diagramas de atividade.

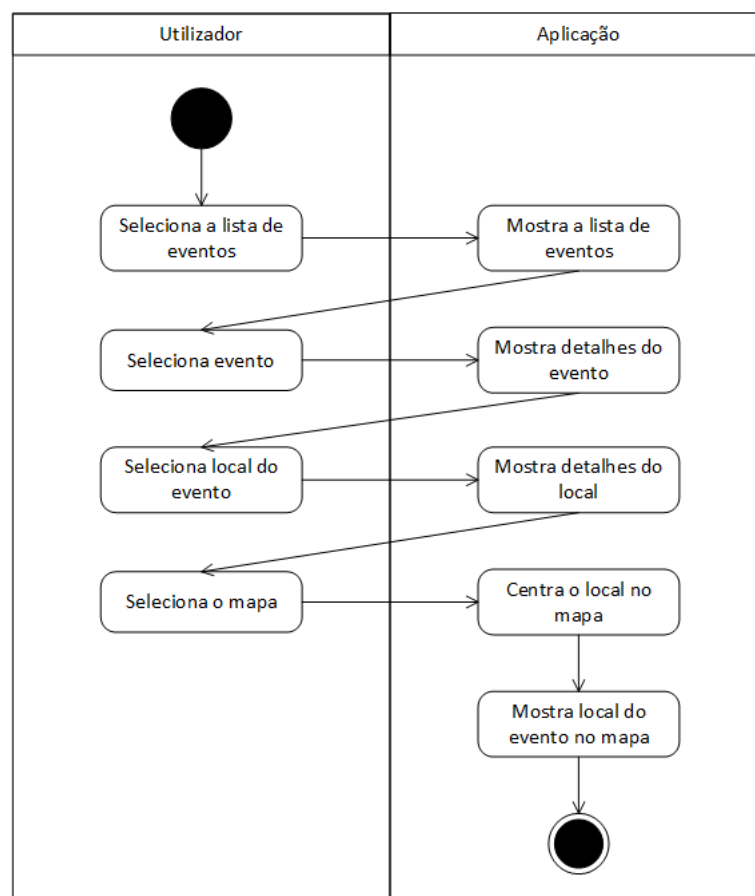


Figura III.4 - Diagrama de atividades do processo localizar evento no mapa

O processo de localizar evento no mapa, representado no diagrama de atividades anterior, reúne um conjunto de casos de utilização: “Localizar no Mapa”, que inclui “Consultar Elementos”, que inclui “Consultar Lista” que pode estender em “Selecionar Lista de Eventos.” Os passos deste processo são:

- O utilizador seleciona a lista de eventos.
- A aplicação mostra a lista de eventos.
- Da lista apresentada, o utilizador seleciona um evento.
- Os detalhes do evento selecionado são mostrados pela aplicação.
- O utilizador seleciona o local do evento.
- São mostrados os detalhes do local pela aplicação.
- O utilizador seleciona o mapa.
- A aplicação centra a local do evento no mapa e apresenta-o ao utilizador.

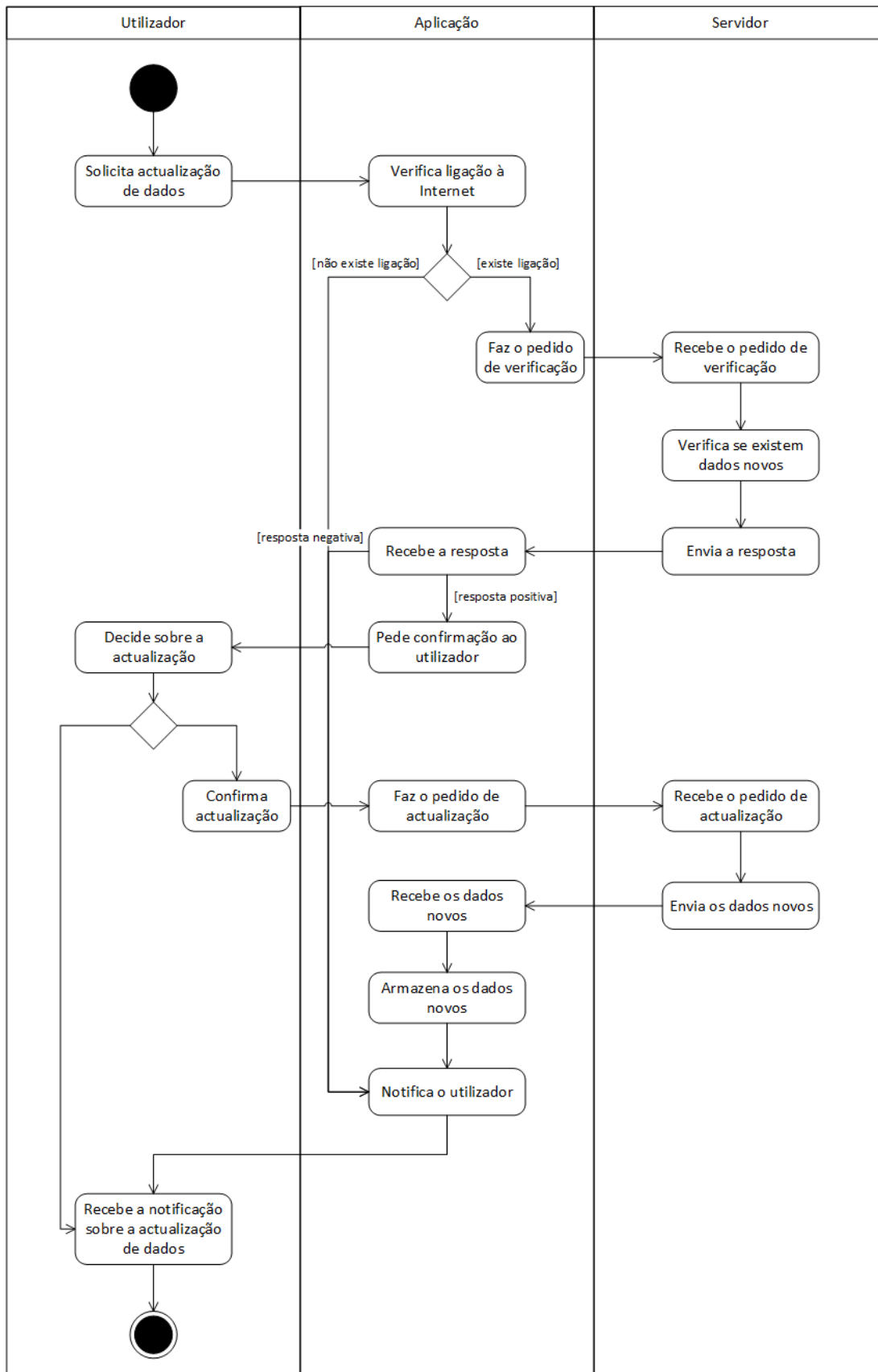


Figura III.5 - Diagrama de atividades do processo de atualização de dados

O diagrama de atividades anterior é referente aos casos de utilização “Atualizar Dados”, cujo ator que inicia o caso é o utilizador, e “Verificar Atualizações” iniciado pela aplicação. Os passos representados neste diagrama são:

- O utilizador seleciona na aplicação a opção de atualizar os dados referentes ao festival.
- A aplicação verifica se existe ligação à Internet através de Wi-Fi ou dados móveis.
- Caso não existe ligação, o utilizador é notificado e o processo termina.
- Caso existe ligação à Internet, a aplicação envia um pedido ao servidor para verificar se existem novos dados.
- O servidor recebe o pedido e verifica se existem novos dados.
- Casos não existam, a resposta do servidor é negativa, o utilizador é notificado e o processo termina.
- Casos existam dados novos, a aplicação recebe-os, armazena-os na estrutura de dados local, o utilizador é notificado e o processo termina.

III.5. ARQUITETURA

Os sistemas computacionais são, normalmente, constituídos por elementos distintos, que interligados entre si funcionam como um todo. Surge então a necessidade de definir e diferenciar a natureza de cada um dos elementos, o papel que desempenham e como o fazem. A arquitetura de *software* consiste na estrutura do sistema, no qual se incluem os elementos de *software* que o compõem, as propriedades externamente visíveis e as relações entre eles. [15]

Apesar de a aplicação móvel funcionar quase de um modo independente, a atualização dos dados do festival faz-nos, instantaneamente, pensar no estilo arquitetural Cliente-Servidor, no qual vários clientes efetuam pedidos a um único servidor.

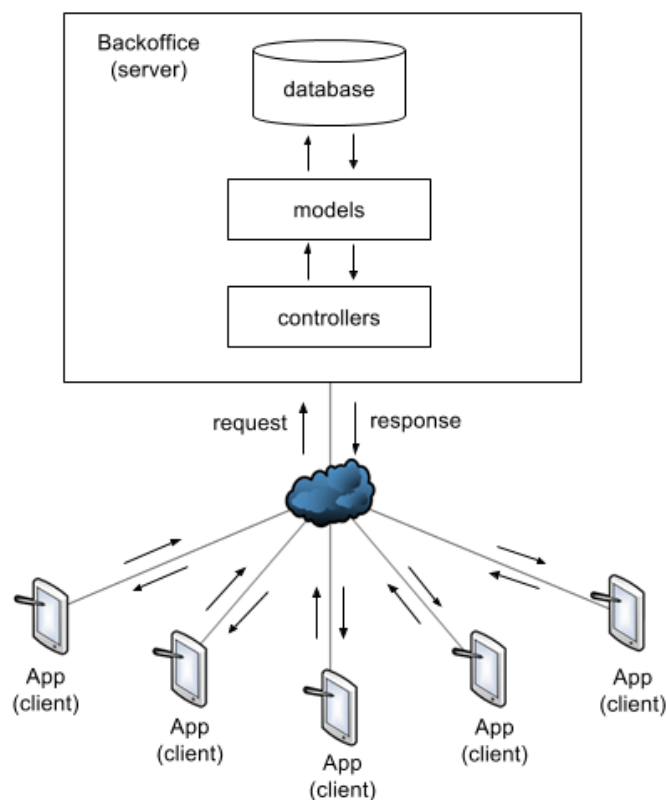


Figura III.6 - Diagrama representativo da arquitetura Cliente-Servidor

Como já foi esquematizado anteriormente, o utilizador poderá solicitar pedidos de atualização de dados a um servidor remoto, utilizando uma ligação à internet. Nesta comunicação, a aplicação instalada nos dispositivos móveis é considerada o Cliente, uma vez que partem dela os pedidos para o Servidor, representado por uma aplicação *web/webservice* alojado num servidor remoto, que atenderá o pedido, fará o tratamento necessário e encaminhará a resposta.

A grande vantagem da utilização deste estilo arquitetural neste projeto, é mesmo a centralização do armazenamento dos dados num único servidor, o que facilita a sua atualização remota e, posteriormente, localmente.

A implementação da aplicação móvel em si seguirá uma arquitetura MVC (*Model-View-Controller*) bem definida pela *framework* que será utilizada, a Alloy do Appcelerator Titanium, que será mencionada novamente no capítulo seguinte.

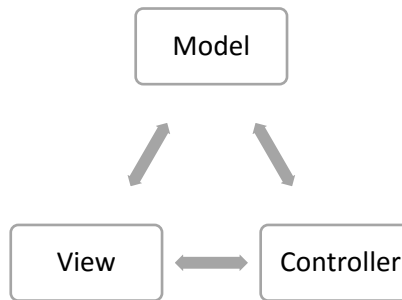


Figura III.7 - Diagrama representativo da arquitetura MVC

Além de tornar a aplicação escalável, a arquitetura MVC permite o reaproveitamento do código, através da reutilização de métodos e *views* por exemplo, o que facilita na organização, manutenção e limpeza do código.

III.6. ESTRUTURA DA BASE DE DADOS

A base de dados de uma aplicação representa uma parte fundamental da robustez da mesma. A maneira como os dados são armazenados determina, em parte, o desempenho da aplicação.

Nesta fase, a base de dados será modelada tendo em conta, principalmente, os requisitos do Módulo I - Guia do festival, sendo que serão estudadas algumas soluções que permitirão a escalabilidade da mesma, para que a introdução futura dos restantes módulos não implique alterações profundas na estrutura.

Assim sendo, para este projeto é necessária uma estrutura de dados capaz de armazenar e relacionar dados da lista de eventos, de locais, de participantes convidados e de parceiros, assim como a informação de cada um dos elementos de cada uma destas listas.

De acordo com o RNF3, que nos diz que a aplicação deverá ser disponibilizada em vários idiomas, a solução encontrada passa por incluir todas as traduções dos conteúdos na base de dados, seguindo a lógica apresentada no esquema relacional seguinte.

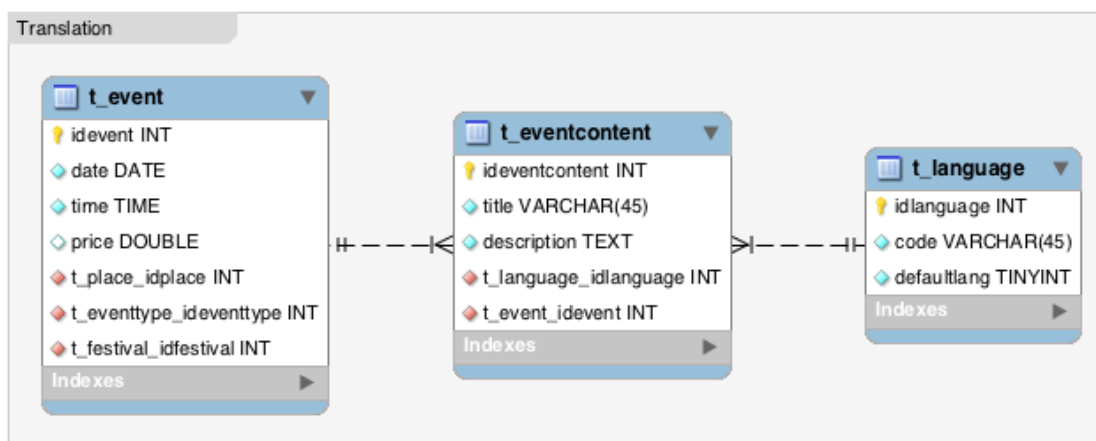


Figura III.8 - Exemplo prático da solução encontrada para o armazenamento das traduções

Na tabela **t_language** estarão armazenados todos os idiomas para os quais será feita a tradução dos dados da aplicação.

Para cada conjunto de dados que necessite de tradução serão criadas duas tabelas:

- Uma primeira que armazenará os dados não traduzíveis como datas, preços, estados, urls, chaves estrangeiras, etc - tabela **t_event**, por exemplo;

- E uma outra - tabela **t_eventcontent**, por exemplo - com os dados traduzíveis como títulos e textos/descrições.

A relação de muitos-para-muitos estabelecida entre as tabelas **t_language** e **t_event**, conseguida através da tabela de relação **t_eventcontent**, permite que para o mesmo evento existam várias traduções, nos vários idiomas disponíveis.

Esta é uma solução escalável, uma vez que sempre que necessário adicionar novos idiomas à plataforma, basta introduzi-los na tabela **language** e os conteúdos traduzidos na tabela **t_eventcontent**, por exemplo, sem que seja necessário alterar a estrutura da base de dados ou repetir os dados não traduzíveis, evitando assim a redundância de dados.

Dada a arquitetura do sistema, a base de dados da aplicação móvel será semelhante ao da base de dados remota. Foi ainda analisada a opção de utilizar uma base de dados não relacional para o armazenamento dos dados na aplicação móvel, o que não se verificou viável pois traria dificuldades na atualização dos dados devidos às questões de redundância associadas, causadas pela inexistência de relações e de normalização os dados armazenados.

III.6.1. Esquema Relacional

O esquema relacional ou diagrama entidade-relação de uma base de dados representa as relações entre as entidades/tabelas que contêm os vários tipos de dados que compõem a informação dinâmica necessária, neste caso da aplicação móvel e do *backoffice*. No esquema relacional seguinte (**Figura III.9**), estão representadas as principais tabelas e respetivas relações que irão constituir a base de dados do *backoffice*, gerido pelo *staff* do festival, e servirá de fonte de dados para as atualizações de dados da aplicação móvel.

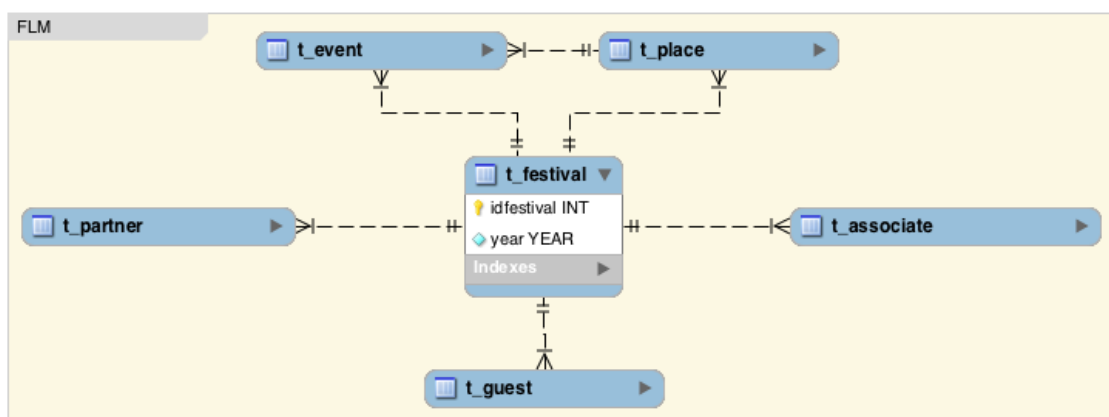


Figura III.9 - Esquema relacional da base de dados remota (*backoffice*)

- **t_eventtype** e **t_etypecontent** - dados relativos aos vários tipos de eventos (conferência, conversa cruzada, espetáculo, etc) e respetiva tradução;
- **t_esupertype** - dados relativos aos supertipos associados aos tipos de eventos (FLM e FLI);
- **t_eventguest** - guarda a relação de muitos para muitos entre os eventos e os convidados, e tipo de participação destes últimos em cada evento (participante, moderador, artista, etc).

III.6.1.2. Convidados

Os dados relativos aos participantes convidados do festival, que seguem os requisitos RF10 e RF11, serão armazenados no conjunto de tabelas representado na **Figura III.11**.

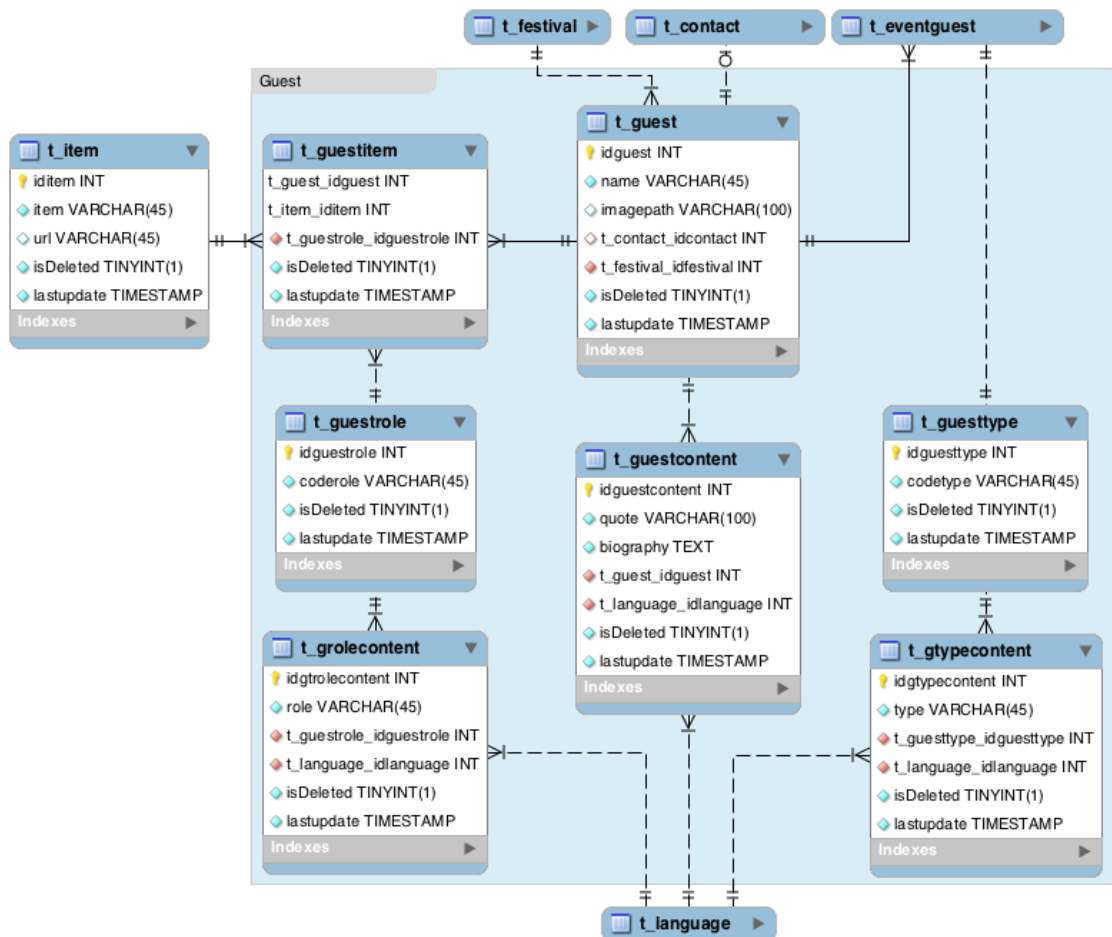


Figura III.11 - Tabelas para o armazenamento de dados relativos aos convidados participantes

A informação afeta aos participantes convidados do festival será armazenada nas tabelas **t_guest** e **t_guestcontent**, **t_guesttype** e **t_gtypecontent**, **t_guestitem** e **t_item**, **t_guestrole** e **t_grolecontent** e distribuída da seguinte forma:

- **t_guest** e **t_guestcontent** - dados relativos a cada convidado como o nome, pequena frase descritiva, biografia, url da foto e contactos, e respetiva tradução;
- **t_guesttype** e **t_gtypecontent** - dados relativos aos vários tipos de participação que os convidados podem desempenhar em cada evento, e respetiva tradução;
- **t_guestitem** - guarda a relação de muitos para muitos entre os convidados e algumas das suas obras mais importantes, e o tipo de papel do convidado perante a obra (escritor, ilustrador, etc);
- **t_item** - dados relativos às obras, álbuns, etc do reportório de cada convidado;
- **t_guestrole** e **t_grolecontent** - dados relativos aos vários tipos de papéis que os convidados desempenham em às obras, e respetiva tradução.

III.6.1.3.Locais

Para os locais dos eventos do festival e de acordo com os requisitos **RF8** e **RF9**, a informação relacionada será guardada nas tabelas representadas na **Figura III.12**.

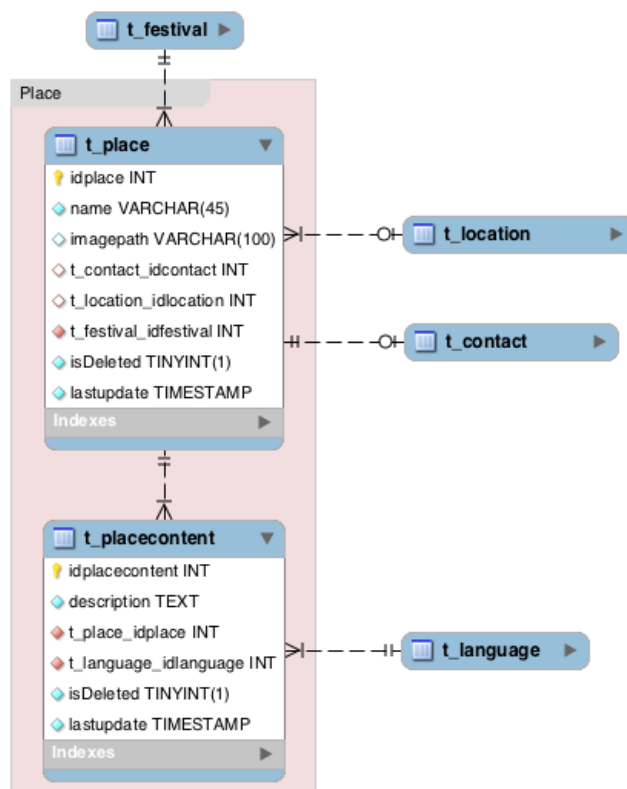


Figura III.12 - Tabelas para o armazenamento de dados relativos aos locais

Nas tabelas **t_place** e **t_placecontent** serão armazenados dados relativos a cada local como designação e respetiva tradução, url da imagem, localização e contactos.

III.6.1.4.Parceiros

Os dados relativos aos parceiros do festival, de acordo com os requisitos **RF15** e **RF16**, serão armazenados nas tabelas representadas na **Figura III.13**.

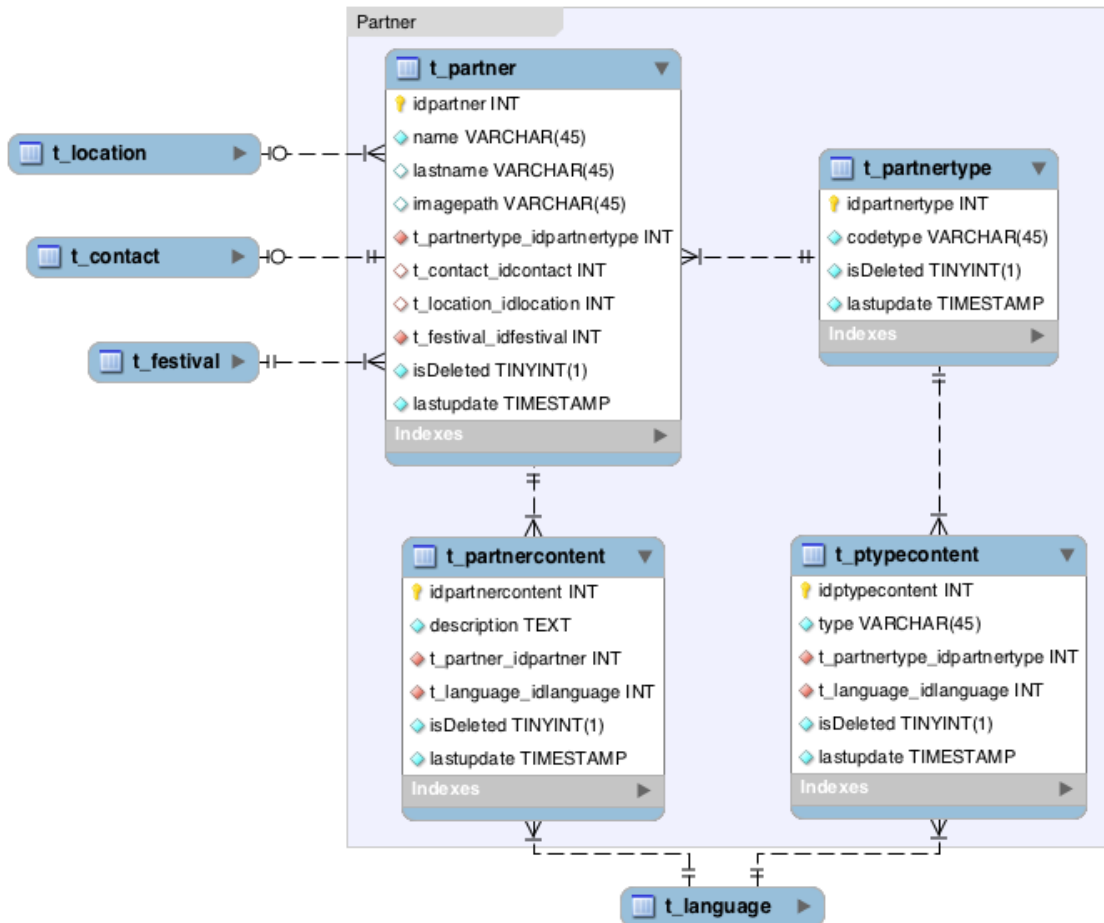


Figura III.13 - Tabelas para o armazenamento de dados relativos aos parceiros

A informação dos parceiros será guardada nas tabelas **t_partner** e **t_partnercontent**, e **t_partnertype** e **t_ptypecontent**:

- **t_partner** e **t_partnercontent** - dados relativos a cada parceiro do festival como nome, descrição, URL da imagem, tipo de parceiro, localização e contactos, e respetiva tradução.
- **t_partnertype** e **t_ptypecontent** - dados relativos aos vários tipos de parceiros que o festival pode ter (patrocinador, apoio, etc), e respetiva tradução.

III.6.1.5.Localização

Os dados relativos à localização geográfica como as coordenadas geográficas latitude e longitude, morada, código postal e país, tanto para locais, parceiros e associados do festival serão armazenados na tabela **t_location**.

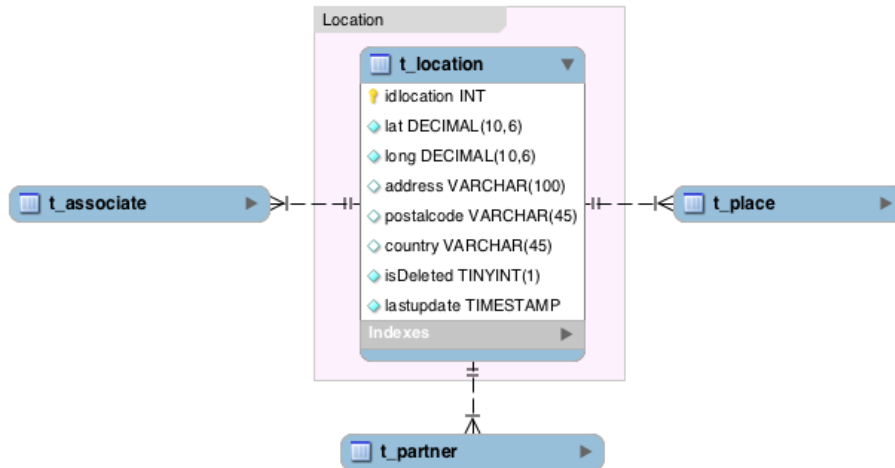


Figura III.14 - Tabelas para o armazenamento de dados de contactos

III.6.1.6.Contactos

Tanto os contactos (telefone, email e *link* para o *website*) dos participantes convidados, parceiros, associados e locais do festival serão armazenados na tabela **t_contact**, fazendo cumprir os requisitos **RF9**, **RF16**, **RF20** e **RF22**.

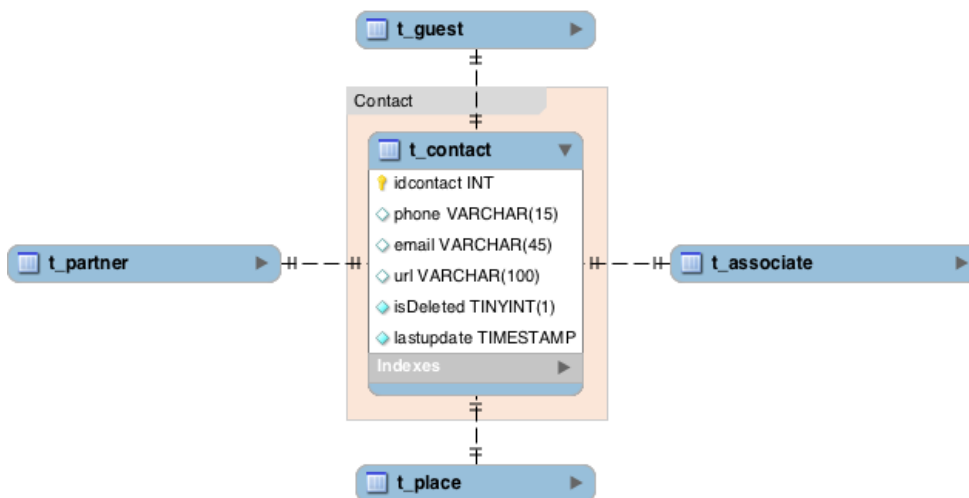


Figura III.15 - Tabelas para o armazenamento de dados de contactos

III.6.1.7.Outros

Além da informação relativa a cada evento, convidado, local e parceiro, a base de dados estará preparada para armazenar outros dados como os dos associados do festival e utilizadores, satisfazendo os requisitos **RF17**, **RF18**, **RF19**, **RF20**, **RF21**, **RF22**, **RF23** e **RNF10**, apesar de não serem utilizados na versão inicial da aplicação. Esta informação complementar será distribuída pelas seguintes tabelas:

- **t_associate** e **t_associatecontent**, e **t_associatetype** e **t_atypecontent** - seguindo a mesma ordem de ideias utilizada para os eventos, nestas tabelas serão guardados relativos aos associados do festival como nome, descrição, tipo de associado, localização e contactos, e respetiva tradução;
- **t_benefitcontent** - dados relativos aos benefícios oferecidos por cada associado do festival;
- **t_user** - dados relativos ao utilizador da aplicação, como o código identificador;
- **t_rating** - a relação muitos para muitos entre os utilizadores da aplicação e os associados do festival e respetiva classificação atribuída;

Por fim, cada tabela contém ainda um campo **isDeleted**, que indica se o tuplo foi descartado ou não, e um **updatetime** que guarda a data e hora da última atualização do mesmo. Estes dois valores serão utilizados durante o processo de atualização das bases de dados dos dispositivos móveis da segunda versão da aplicação e voltarão a ser mencionados mais à frente neste relatório.

III.7. INTERFACE DO UTILIZADOR

O passo seguinte do processo de desenvolvimento da aplicação é a prototipagem, que consiste na representação do aspeto gráfico da aplicação, sendo que existem protótipos de baixa e de alta fidelidade.

Para a conceção dos protótipos de baixa fidelidade foram utilizadas duas técnicas de prototipagem: *Mockups* e CAP (*Canonical Abstract Prototype*, em português Protótipos Abstratos Canónicos).

III.7.1. Mockups

Os *mockups* são técnicas de prototipagem de baixa fidelidade onde são criados esboços das interfaces, feitos à mão em papel, normalmente a lápis, de modo a sua alteração possa ser feita de uma maneira rápida e fácil.

Seguida serão apresentados alguns dos *mockups* iniciais desenhados para a aplicação do Festival Literário da Madeira (**Figura III.16** e **Figura III.17**), sendo que os restantes ecrãs da aplicação do FLM podem ser consultados nos anexos, em **Anexo 3. Mockups**.



Figura III.16 - *Mockup* do ecrã inicial

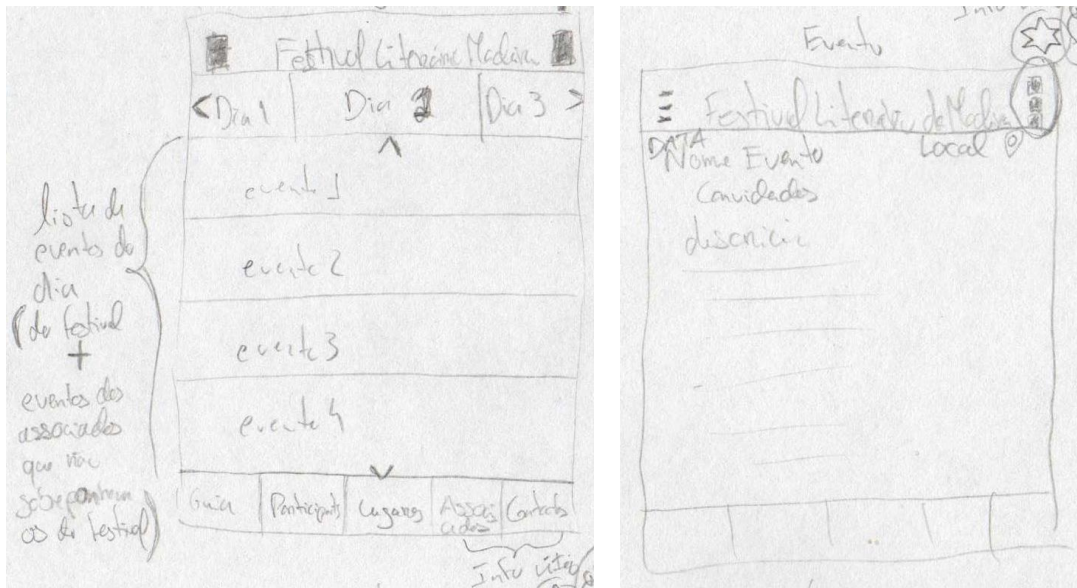


Figura III.17 - Mockup dos ecrãs da lista de eventos e do evento em detalhe (descrição e convidados)

III.7.2. Protótipos Abstratos Canónicos

Os Protótipos Abstratos Canónicos consistem numa forma de representação gráfica intermédia entre modelos abstratos e os protótipos de alta fidelidade, que usa um conjunto próprio e intuitivo de símbolos que representam as várias interações possíveis, sem que se especifique detalhes de aparência ou comportamento. [16]



Figura III.18 - CAP do ecrã inicial

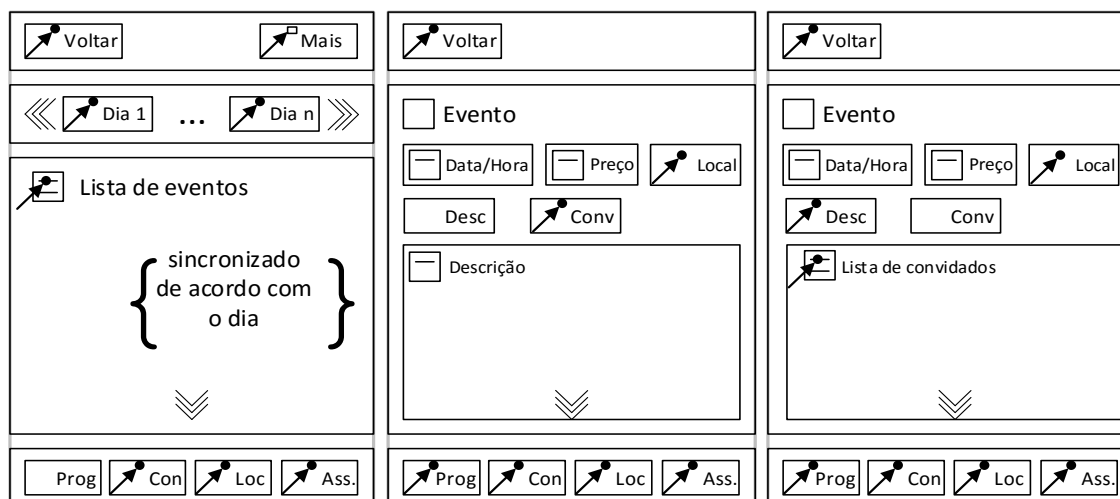


Figura III.19 - CAPs dos ecrãs da lista de eventos e do evento em detalhe (descrição e convidados)

Os CAPs dos restantes ecrãs da aplicação do FLM podem ser consultados nos anexos, em **Anexo 4. Protótipos Abstratos Canónicos (CAPs)**.

III.7.3. Protótipos de Alta Fidelidade

Os protótipos de alta fidelidade são uma representação mais próxima do que se pretende na realidade, tanto a nível de navegação entre vistas e disposição dos componentes da aplicação, como das cores, tamanhos dos componentes, tamanho e tipos de letras, sombras, brilhos, entre outros. Os protótipos que se seguem foram elaborados pela *designer* da empresa, Dra. Andreia Encarnação, que teve por base os CAPs anteriormente cedidos.



Figura III.20 - Protótipo de alta fidelidade do ecrã inicial

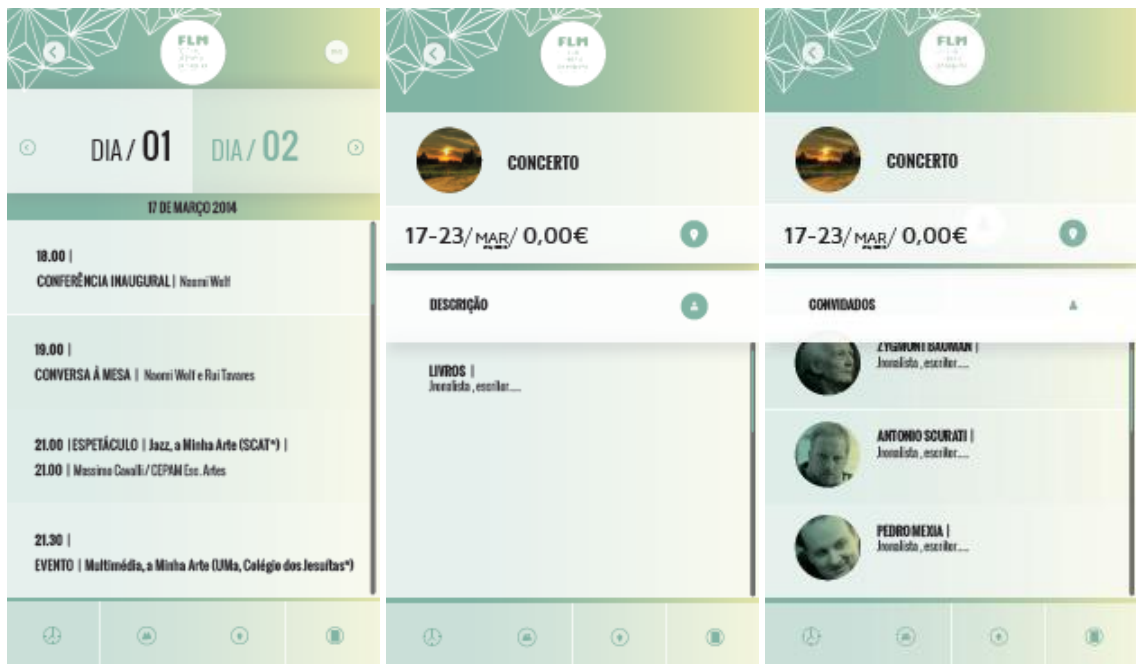


Figura III.21 - Protótipo de alta fidelidade dos ecrãs da lista de eventos do evento em detalhe (descrição e convidados)

Os protótipos de alta fidelidade dos restantes ecrãs da aplicação do FLM podem ser consultados nos anexos, em **Anexo 5. Protótipos de Alta Fidelidade**.

IV. IMPLEMENTAÇÃO

“First, solve the problem. Then, write the code.”

– John Johnson

IV.1. INTRODUÇÃO

À fase de modelação, onde toda a estrutura lógica foi estudada e definida, precede a fase de implementação da aplicação que, por sua vez, inclui a escolha e ambientação com as ferramentas escolhidas para todo o processo de implementação.

Sendo este um projeto no qual os dados serão armazenados em base de dados SQL, MySQL para o *backoffice* e SQLite para a aplicação móvel, surge a necessidade da utilização de ferramentas para a elaboração e gestão do esquema relacional de ambas, definido na fase anterior.

Já citado anteriormente no Estado da Arte, foi também feito um estudo sobre qual a ferramenta a ser utilizada para a implementação do código, que será nesta secção apresentada com mais algum pormenor.

A vantagem de manter várias versões do código, que facilmente serão recuperáveis, e o facto de existir para o projeto uma equipa de desenvolvimento composta por mais do que um elemento, colmataram na necessidade da utilização de um sistema de controlo de versões do código da aplicação.

Em suma, neste capítulo serão citadas as ferramentas utilizadas nesta fase, nomeadamente, para a construção do esquema relacional tanto da base de dados MySQL do *backoffice*, o MySQL Workbench (www.mysql.com/products/workbench), como da base de dados em SQLite da aplicação móvel, o DB Browser for SQLite (www.sqlitebrowser.org); para o desenvolvimento e complicação do código da aplicação móvel, o Appcelerator Titanium e para a gestão e controlo de versões do código propriamente dito, o GitHub (www.github.com).

Uma vez que será a primeira vez que terei contacto com a plataforma Appcelerator Titanium, optei por analisar e seguir alguns tutoriais de iniciação de utilização da plataforma que se encontram disponíveis na área da documentação e suporte no site da plataforma (www.docs.appcelerator.com). Relativamente às outras ferramentas, apenas foram feitos alguns testes antes da utilização propriamente dita para a implementação do projeto, uma vez que já eram do meu conhecimento ou de utilização simples.

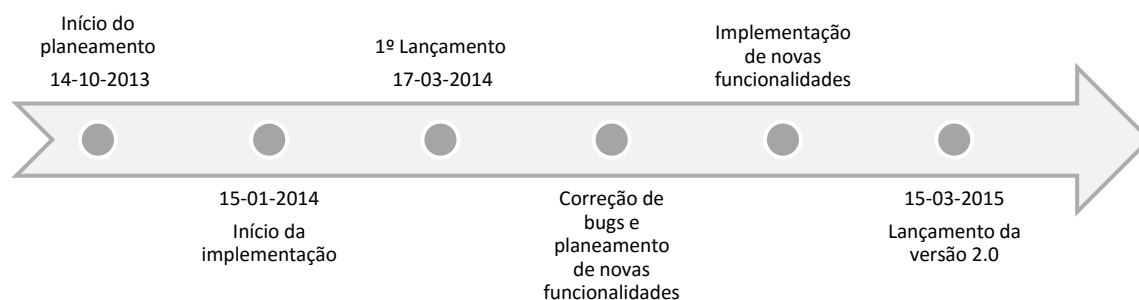


Figura IV.1 - *Timeline* da evolução da aplicação FLM

Ainda neste capítulo serão descritas e apresentadas as funcionalidades que foram desenvolvidas para a primeira versão da aplicação móvel que foi lançada no dia 17 de março de 2014, assim como as alterações e melhorias implementadas e disponibilizadas a 15 de março de 2015 aquando do lançamento da segunda versão.

IV.2. LINGUAGENS E TECNOLOGIAS UTILIZADAS

IV.2.1. MySQL Workbench e SQLite Browser

O MySQL Workbench é a ferramenta mais utilizada para a criação de esquemas de base de dados relacionais que permite também a administração e monitorização de base de dados alojadas em servidores remotos, sendo por isto, utilizada por arquitetos, desenvolvedores e administradores de projetos.

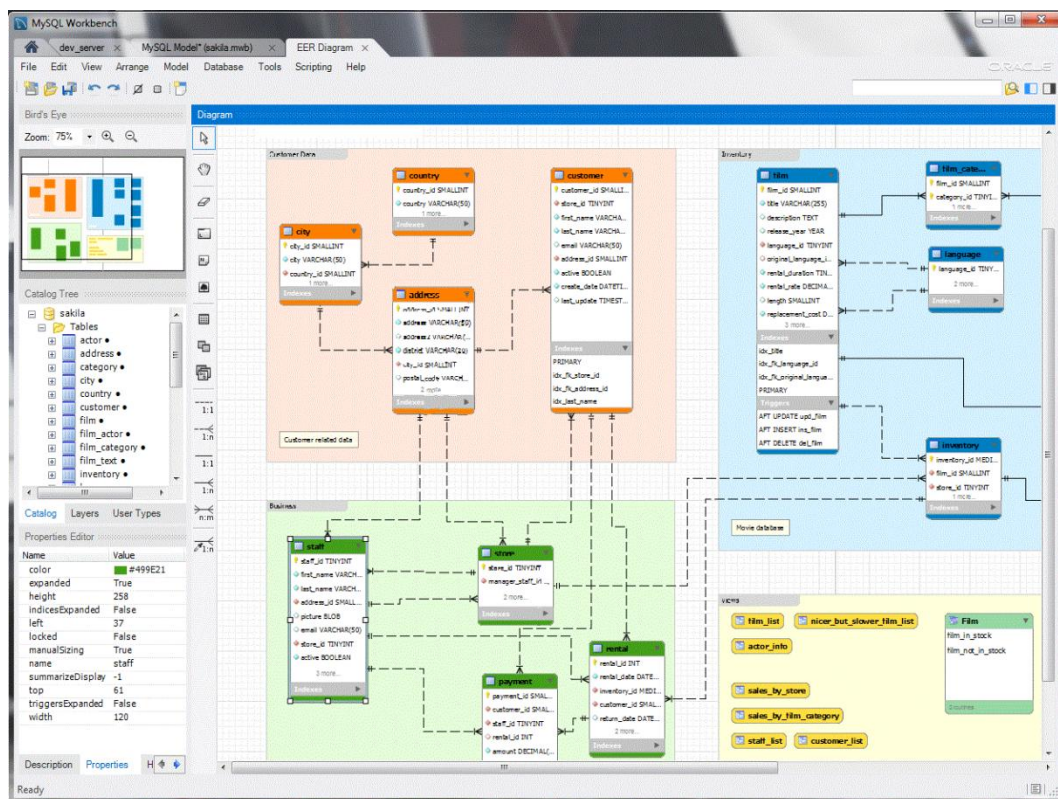


Figura IV.2 - Editor de esquemas relacionais do MySQL Workbench [17]

Para além do desenho do esquema relacional da base de dados, esta ferramenta servirá também para a monitorização dos dados da *backoffice* da aplicação, e para tal deverá ser configurada com os dados de acesso ao servidor da Nova Delphi, onde estará alojada a base de dados remota para testes.

Mais simples e menos conhecido, o *DB Browser for SQLite* consiste numa ferramenta *open source* como uma interface para a criação e edição de ficheiros compatíveis com o SQLite, que será utilizado para a construção e manutenção do ficheiro da base de dados da aplicação móvel.

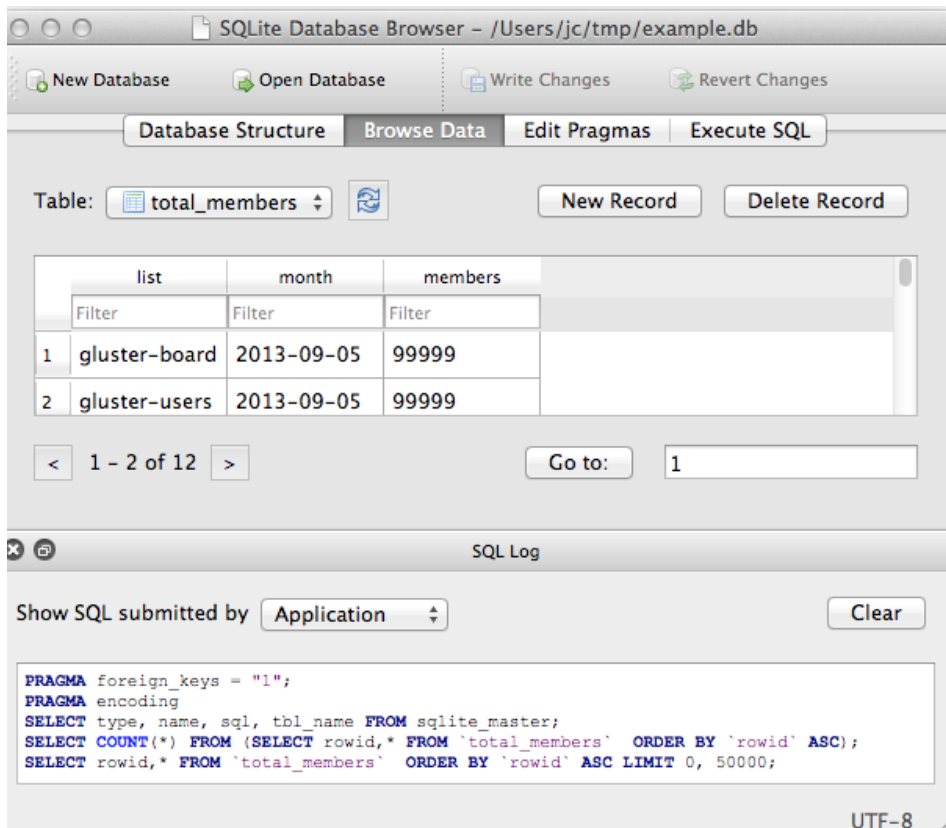


Figura IV.3 - Editor DB Browser for SQLite [18]

IV.2.2.Appcelerator Titanium

A plataforma Titanium da Appcelerator permite a criação de aplicações móveis nativas para dispositivos iOS e Android através de tecnologias multiplataforma e conta, neste momento, com mais de 60 000 aplicações a correr em cerca de 170 000 000 de dispositivos. [19] Disponibiliza ainda uma vasta documentação sobre a plataforma, instalação e configuração, o seu editor de código e SDK, entre outros, apresentando alguns tutoriais e exemplos práticos sempre que aplicável.



Figura IV.4 - Logo da Appcelerator e Titanium

O Appcelerator Titanium disponibiliza um IDE grátis para o desenvolvimento das aplicações, o Titanium Studio, que além de editor de código, permite ainda simulá-lo através de emuladores ou de dispositivos móveis e testá-lo. O Studio também contém *templates* e aplicações exemplo que ajudam no início do desenvolvimento de aplicações. [20]

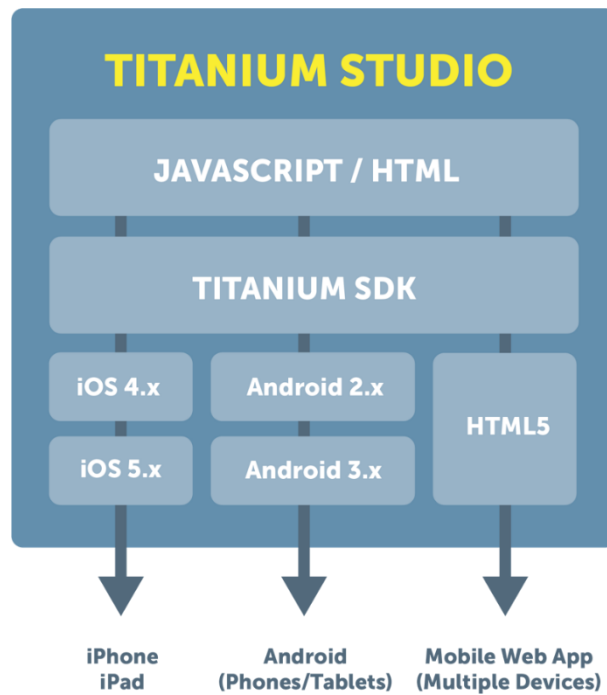


Figura IV.5 - Esquema do funcionamento do Titanium Studio [21]

O Titanium SDK atualmente na versão 3.5.1 (lançado a 6 de março de 2015), consiste num conjunto de ferramentas baseadas em Node.js que trabalham com os SDKs (*Software Development Kit*) nativos e ajudam na construção das aplicações multiplataforma nativas a partir do código-fonte desenvolvido em JavaScript. [22]

O Titanium SDK funciona como uma ponte entre o código da aplicação em JavaScript e as operações nativas do sistema, ou seja, traduzindo as chamadas em JavaScript nas nativas equivalentes. [21]

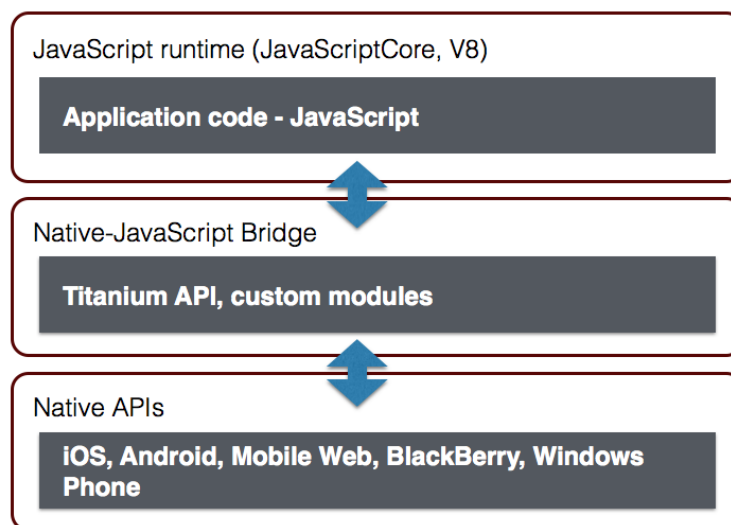


Figura IV.6 - Titanium SDK como ponte entre o JavaScript e o código nativo [21]

IV.2.2.1. Framework Alloy

O Appcelerator disponibiliza ainda uma *framework*, a Alloy, que torna o desenvolvimento das aplicações mais rápido [23], aumentando a qualidade das mesmas e permitindo a reutilização de componentes.



Figura IV.7 - Framework Alloy do Appcelerator

A Alloy, atualmente na versão 1.5.1 (2 de outubro de 2014) utiliza uma arquitetura MVC, (*Model-View-Controller*, Modelo-Vista-Controleador), separando as aplicações nas três componentes: [24]

Modelos – a manipulação dos dados é feita através de *Backbone Models* e *Collections*.

Vistas – as componentes da interface de utilizador são desenhadas em XML e estilizadas nas Titanium Stylesheets, TSS, bastante semelhantes ao HTML e CSS.

Controladores – a lógica da aplicação, o tratamento dos dados e a respetiva ponte entre os modelos e as componentes das vistas é feita em JavaScript.

Na figura seguinte são apresentadas as diferenças entre a estrutura de uma aplicação que usa Alloy (à esquerda) e uma aplicação Titanium tradicional (à direita).

```
index.tss
1- |.container": {
2-   backgroundColor:"white"
3- },
4- "Label": {
5-   width: Ti.UI.SIZE,
6-   height: Ti.UI.SIZE,
7-   color: "#000"
8- }

index.js
1- function doClick(e) {
2-   alert($.label.text);
3- }
4-
5- $.index.open();
6-

index.xml
1- <Alloy>
2-   <Window class="container">
3-     <Label id="label" onClick="doClick">Hello, World</Label>
4-   </Window>
5- </Alloy>

app.js
1- var win = Titanium.UI.createWindow({
2-   backgroundColor: 'white'
3- });
4- // Platform specific label
5- if (Ti.Platform.osname == "android") {
6-   var label = Titanium.UI.createLabel({
7-     top: '50dp',
8-     color: 'black',
9-     text: 'Hello Android user'
10- });
11- } else {
12-   var label = Titanium.UI.createLabel({
13-     top: '100dp',
14-     color: 'red',
15-     text: 'Hello iPhone user'
16- });
17- }
18- win.addEventListener("click", function() {
19-   alert(label.text);
20- });
21-
22- win.add(label);
23- win.open();
24-
```

Figura IV.8 - Exemplo: aplicação Alloy vs aplicação Titanium tradicional [25]

IV.2.3.GitHub

Segundo as boas práticas de desenvolvimento de *software*, para qualquer projeto desenvolvido deve ser utilizado um sistema de controlo de versões, em inglês *Version Control System*, VCS. O VCS permite manter um histórico das várias versões do sistema de ficheiros do projecto num repositório, que pode ser acedido pelos elementos da equipa de desenvolvimento do projeto. Além de facilitar o trabalho em equipa, uma vez que faz a junção das alterações de cada elemento ao projeto global, permite, sempre que necessário, reaver do repositório uma versão mais antiga de um ficheiro em particular, ou de todo o projeto.

Uma vez que o projeto será composto por dois sistemas, a aplicação móvel e a plataforma web, que serão desenvolvidas com a colaboração de várias pessoas, a utilização de um sistema de controlo de versões é realmente necessária para que se evite a geração de conflitos ou perda de código por algumas das partes. A evolução dos vários componentes do projeto pode também ser acompanhada por toda a equipa de trabalho, conforme serão feitos os *commits* no repositório, o que considero ser uma mais-valia, pois é do interesse de todos os colaboradores do projeto.

O sistema de controlo de versões escolhido para a ser utilizado durante o desenvolvimento do projeto será o incontestável GitHub (<https://github.com>), sugerido pelo Eng, Hélder.

IV.3. VERSÃO 1.0

Durante a implementação da primeira versão da aplicação móvel, foram selecionadas as funcionalidades cujos requisitos funcionais e não funcionais apresentam prioridade alta, e que serão enumerados na seção seguinte.



Figura IV.9 - Logo e *splashscreen* da versão 1 da aplicação

Devido a questões relacionadas com a escassez de tempo para a implementação da primeira versão, foi necessário encontrar uma solução, que embora provisória, fosse de rápida implementação para a atualização dos dados da aplicação móvel. Assim, optou-se por gerar réplicas da base de dados remota, converte-las para MySQLite e instalá-las no dispositivo móvel sempre que este fizesse um pedido de atualização de dados e que se verificasse a existência de dados novos (através do campo *update* das tabelas).

IV.3.1. Requisitos/funcionalidades implementados

De uma forma muito resumida, a primeira versão implementada da aplicação permitia ao utilizador navegar e pesquisar nas listas de eventos, convidados, locais e parceiros, consultar os dados associados a cada um dos elementos de cada uma destas listas e a atualização destes

dados. Na tabela seguinte são listados os requisitos funcionais e não funcionais implementados na sua totalidade ou parcialmente.

Tabela IV.1 - Requisitos implementados na versão 1.0

REQUISITO	PRIORIDADE	IMPLEM.
RF1 - A aplicação deve permitir ao utilizador a visualização do programa de eventos do festival, ordenado cronologicamente.	Alta ★ ★ ★	Total ★ ★ ★
RF2 - A aplicação deve permitir ao utilizador pesquisar os eventos do festival e filtrá-los por categorias/tipos , por local ou por participantes convidados.	Média ★ ★ ☆	Parcial ★ ★ ☆
RF3 - A aplicação deve disponibilizar informação relativa a cada evento do festival, como tema, título, descrição, local, data/horário e participantes convidados.	Alta ★ ★ ★	Total ★ ★ ★
RF4 - A aplicação deve diferenciar visualmente os eventos do FLI dos restantes eventos do FLM.	Alta ★ ★ ★	Total ★ ★ ★
RF6 - A aplicação deve disponibilizar informação sobre o preço e locais onde podem ser comprados os bilhetes para os eventos pagos.	Baixa ★ ☆ ☆	Parcial ★ ★ ☆
RF8 - A aplicação deve disponibilizar a lista dos locais onde decorrem os eventos do festival.	Alta ★ ★ ★	Total ★ ★ ★
RF9 - A aplicação deve permitir ao utilizador visualizar uma pequena descrição, a localização no mapa e informações úteis sobre cada local, como o <i>link</i> para a página do local, os contactos e como lá chegar .	Alta ★ ★ ★	Parcial ★ ★ ☆
RF10 - A aplicação deve disponibilizar informação relativa a cada participante convidado do festival, como foto, uma pequena biografia sobre o participante convidado e os seus principais feitos/obras publicadas, acompanhados de <i>links</i> para sítios que complementem esta informação.	Alta ★ ★ ★	Total ★ ★ ★
RF11 - A aplicação deve disponibilizar, para cada participante convidado, a lista de eventos do festival nos quais este participa.	Alta ★ ★ ★	Total ★ ★ ★

RF15 - A aplicação deve disponibilizar uma lista dos parceiros e promotores do festival.	Alta ★ ★ ★	Total ★ ★ ★
RF16 - A aplicação deve disponibilizar informação sobre cada parceiro: pequena descrição, informações úteis (<i>localização, site</i>).	Alta ★ ★ ★	Parcial ★ ★ ☆
RNF1 - A aplicação deve ser desenvolvida para Android (2.3.3 ou superiores) e iOS (7).	Média ★ ★ ☆	Parcial ★ ★ ☆
RNF2 - A aplicação deve ser desenvolvida para <i>smartphones</i> e <i>tablets</i> .	Alta ★ ★ ★	Total ★ ★ ★
RNF3 - A aplicação deve ser multilingue.	Média ★ ★ ☆	Total ★ ★ ★
RNF4 - Algumas funcionalidades da aplicação requerem que o dispositivo esteja conectado à internet.	Média ★ ★ ☆	Total ★ ★ ★
RNF6 - A aplicação deve funcionar mesmo que o dispositivo não se encontre conectado à internet.	Alta ★ ★ ★	Total ★ ★ ★
RNF7 - A aplicação deve disponibilizar sempre os conteúdos relacionados com o festival, como programa previsto, informação sobre os eventos, locais, participantes convidados e parceiros, sem que seja necessária conexão à internet.	Alta ★ ★ ★	Total ★ ★ ★
RNF8 - A aplicação necessita de conexão à internet para efetuar as atualizações, assim como para permitir ao utilizador realizar reservas e classificações.	Média ★ ★ ☆	Parcial ★ ★ ☆
RFN9 - A aplicação deve disponibilizar informação atualizada que deverá ser gerida pelos respetivos responsáveis.	Alta ★ ★ ★	Parcial ★ ★ ☆
RFN10 - A aplicação deve suportar toda a informação relacionada com o festival e suportar futuras extensões.	Alta ★ ★ ★	Total ★ ★ ★

A implementação do requisito **RF2** foi parcial uma vez que a primeira versão da aplicação apenas permitia a pesquisa de eventos pelo título dos mesmos, sem que fosse possível a filtragem dos mesmos.

Já em relação ao **RF6**, que embora fosse de prioridade baixa, foi parcialmente implementado uma vez que a disponibilização do preço dos eventos na aplicação era de rápida e simples implementação.

Para esta primeira versão, não foram implementadas as funcionalidades relacionadas com o mapa e sinalização da localização geográfica dos eventos e dos parceiros, o que limitou a implementação total dos requisitos **RF9** e **RF16**.

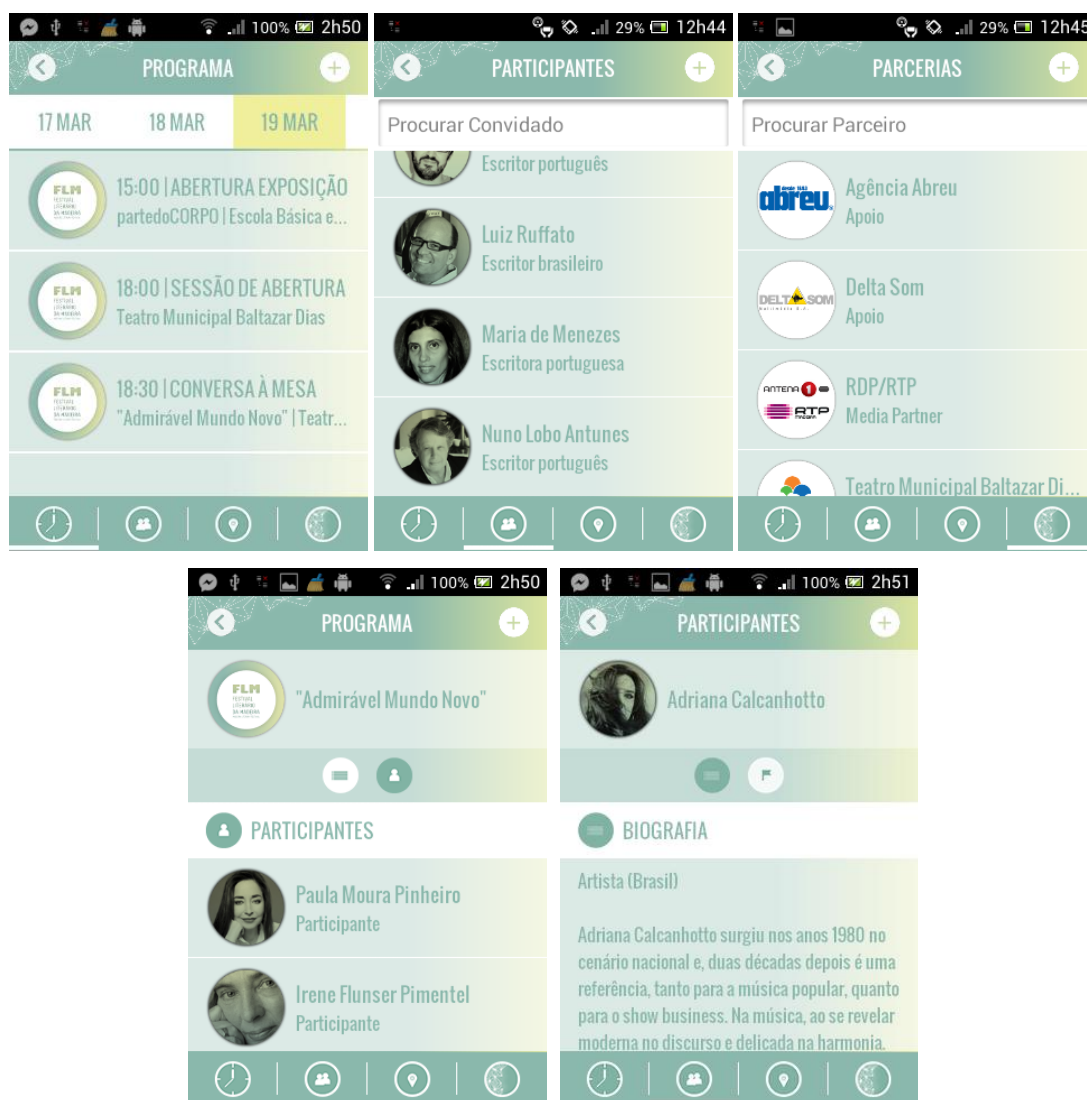


Figura IV.10 - Alguns dos ecrãs desenvolvidos na 1.0 versão da aplicação

IV.3.2. Testes

Uma vez que a edição de 2013 do Festival Literário da Madeira realizou-se de 17 a 23 de março, deixando pouco mais do que dois meses para a aprendizagem da nova ferramenta e implementação da aplicação, decidiu-se não implementar um plano de testes, limitando o mesmo a provas de funcionamento. Algumas das situações de incoerência ou *bugs* que foram surgindo, após o lançamento, foram corrigidas conforme foram detetadas pelos utilizadores.

Embora não tenha existido uma fase de testes à aplicação, durante a implementação da primeira versão da aplicação, foram realizadas algumas verificações entre o que era pretendido na fase de planeamento e o que foi realmente implementado, ou seja, entre o tipo de informação e dados solicitados em cada um dos requisitos funcionais e os conteúdos e ações disponíveis nos vários ecrãs da aplicação, de modo a confirmar que a estrutura criada para a base de dados e as *queries* construídas cumpriam com os mesmos, de uma maneira coerente e coesa.

IV.4. VERSÃO 2.0

Terminada a edição de 2014 do festival, foram realizadas várias reuniões com o intuito de se fazer um balanço das etapas percorridas e do trabalho concretizado até então, de modo a que se pudessem planear as fases que se seguiriam no desenvolver do projeto.

Destas reuniões resultaram, essencialmente, três pontos principais para serem tidos em atenção antes da implementação de novas versões da aplicação: Android *friendly design* da aplicação, atualização dos dados na base de dados da aplicação local e novas funcionalidades a desenvolver, que serão explorados nas secções seguintes.

Para a edição de 2015 foi desenvolvida uma nova imagem nos tons de dourado, preto e branco para o Festival Literário da Madeira, e como tal, a aparência da aplicação também sofreu alterações.



Figura IV.11 – Logo e splashscreen da versão 2 da aplicação

A aplicação pode ser descarregada do Google Play através do seguinte link:
<https://play.google.com/store/apps/details?id=com.novadelphi.flm>.

IV.4.1. Normas de desenvolvimento para aplicações Android

Uma vez que não estavam reunidos recursos suficientes para o desenvolvimento da aplicação para a plataforma iOS, achou-se que a aplicação deveria ter um aspeto mais próximo de uma aplicação nativa Android, de forma a aumentar a experiência do público-alvo limitado aos utilizadores Android. Para tal, foram estudadas e seguidas algumas das principais normas e princípios de *design* para o desenvolvimento de aplicações Android.

IV.4.1.1. Princípios de *Design* e Interface Gráfica

A Equipa de Experiência de Utilizador Android (a *Android User Experience Team* - AUET) enumera alguns princípios de *design* estudados, que têm por base os interesses dos utilizadores e que são aconselhados a serem considerados no desenvolvimento de novas aplicações. Por exemplo, a AUET reforça que é importante surpreender o utilizador com pequenos pormenores súbtis, como pequenas animações ou efeitos sonoros, que ao surgirem no momento certo, contribuem para uma boa experiência e trazem ao utilizador uma sensação de satisfação no executar de uma ação. [26]

A utilização de objetos manipuláveis ao toque, ao invés dos tradicionais botões e menus, reduz o esforço cognitivo necessário à realização das tarefas e a possibilidade de personalizar a gosto pequenos detalhes da aplicação produzem uma sensação de poder e de controlo ao utilizador, deixando-o mais à vontade e tornando a aplicação mais satisfatória emocionalmente. [26]

As imagens e as frases curtas são sempre mais apelativas e relavam-se muito mais eficientes que frases longas ou descrições, que provavelmente serão ignoradas pelo utilizador. A apresentação de demasiada informação de uma só vez pode trazer ao utilizador uma sensação de confusão e restringir o sucesso da concretização das ações pretendidas. A divisão das ações em partes mais pequenas e a apresentação apenas das opções essenciais simplificam a execução das tarefas e trazem confiança ao utilizador. [26]

O utilizador não deve sentir-se perdido durante a navegação, daí a importância da utilização de transições entre ecrãs e de *feedback* sobre as tarefas a decorrer. A consistência no comportamento de componentes com aparência semelhante permite ao utilizador se familiarizar com a mecânica da aplicação.

Considerando todos estes princípios de *design*, tentou-se que esta versão se apresentasse muito mais *Android user friendly*, e para tal, optou-se pela utilização de componentes nativos como a *action bar*, as *tabs*, o menu lateral, entre outros.

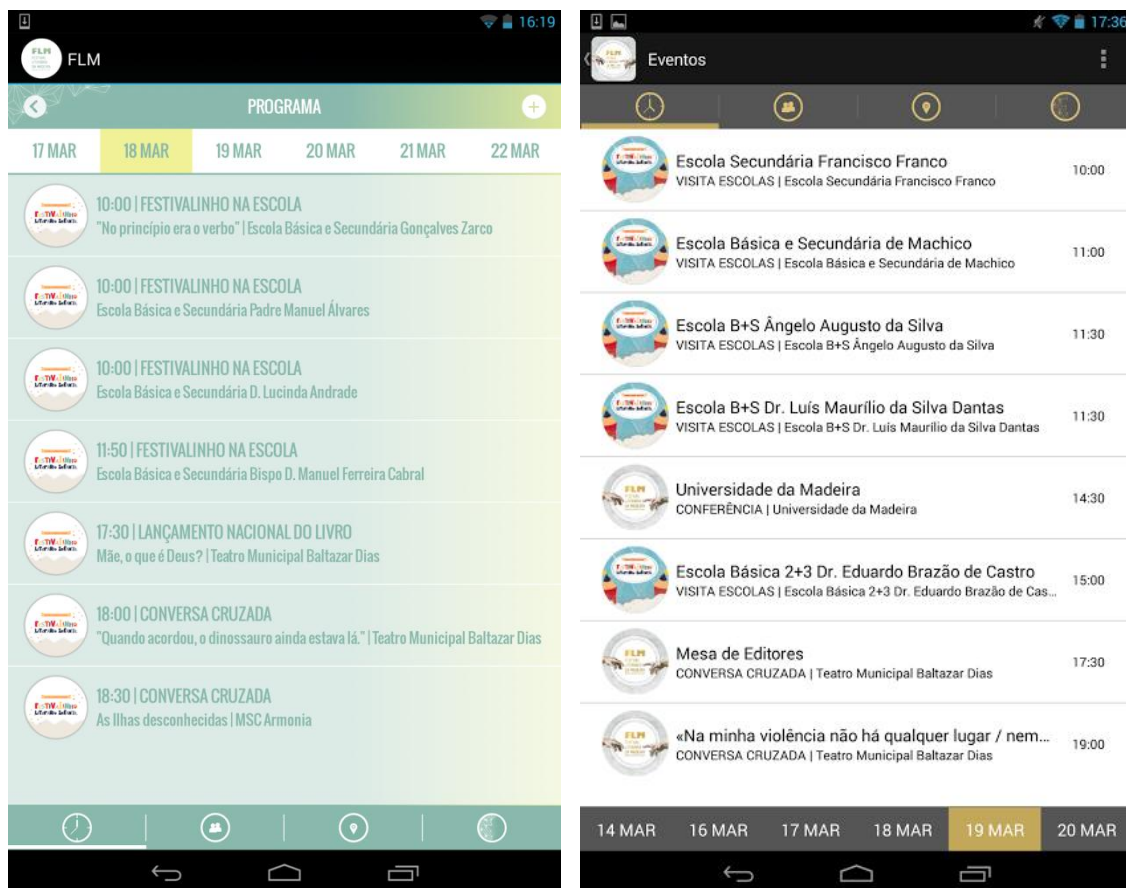


Figura IV.12 - Navegação em *tabs* usada a partir da versão 2.0

A interface gráfica das aplicações Android contém, na maior parte das vezes, três elementos principais: a *Action Bar*, o *Navigation Drawer* que consiste no painel de navegação e que surge normalmente do lado esquerdo do ecrã, e a área de conteúdos.

Action Bar e Barra de Pesquisa

A *Action Bar* é normalmente composta pelo ícone da aplicação, o título da vista apresentada e por botões de outras opções. Através do título da *action bar*, o utilizador consegue identificar em que parte da aplicação se encontra, o que previne que este tenha a sensação de estar perdido. De modo a facilitar a navegação, é também comum ser apresentado na *action bar* o botão de voltar atrás, para que seja possível voltar ao ecrã anterior.



Figura IV.13 - *Action bar* nativa com o título do ecrã e botões de voltar atrás e menu

A barra de pesquisa é também ela incorporada na *action bar*, situação comum nas aplicações android, o que não acontecia na versão anterior, ocupando um espaço desnecessário, como se pode verificar nas imagens seguintes.

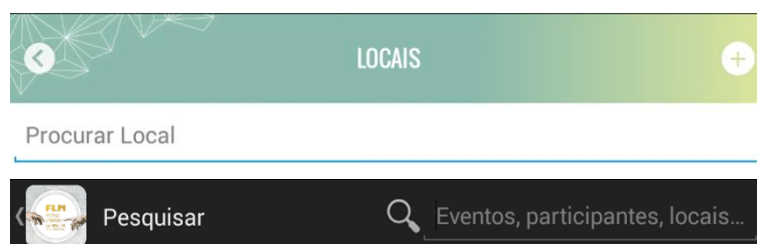


Figura IV.14 - Barra de pesquisa incorporada na *action bar* a partir da versão 2.0

Regra dos 48dp

Na área de conteúdos e em relação às métricas, as aplicações Android seguem a regra dos 48dp (*density-independent pixels*). Os 48dp correspondem a cerca de 9mm que está entre o intervalo de 7 e 10mm, tamanho recomendado para objetos *touchscreen* para que possam ser manuseados pelos dedos dos utilizadores. O espaço entre cada elemento do ecrã deve ser de 8dp. [27]

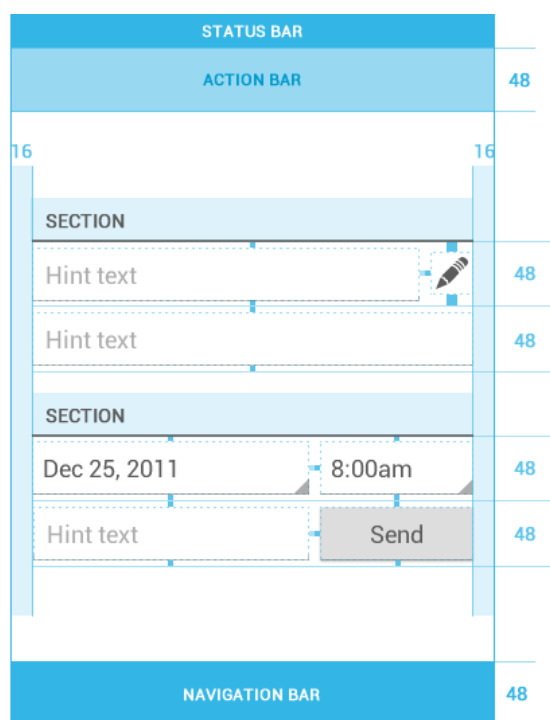


Figura IV.15 - Exemplo da aplicação da regra dos 48dp numa interface

Tipografia

Já em relação à tipografia, o tipo de letra utilizado nas aplicações Android é o Roboto nas suas variantes Thin, Light, Regular, Medium, Bold, Black, Condensed, Condensed Light e Condensed Bold. [28] Na primeira versão, o tipo de letra utilizado na aplicação foi o Oswald, que foi utilizado também em toda a imagem da edição de 2014 do Festival Literário da Madeira.

Roboto Thin
Roboto Light
Roboto Regular
Roboto Medium
Roboto Bold
Roboto Black
Roboto Condensed Light
Roboto Condensed
Roboto Condensed Bold

Figura IV.16 - Tipo de letra Roboto nas várias variantes

Para que a interface de utilizador não fique demasiado confusa, as aplicações Android costumam ter no máximo 4 tamanhos de texto diferentes: Micro 12sp, Small 14sp, Medium 18sp e Large 22sp (*scale-independent pixels*).

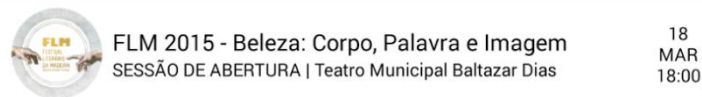


Figura IV.17 - Exemplo da utilização do tipo de letra Roboto

Futuramente, se forem reunidas as condições para o desenvolvimento para iOS, essa transição é de fácil implementação, uma vez o Appceletor permite a criação de componentes exclusivos para cada tipo de sistema operativo.

IV.4.2. Atualização dos dados locais

A solução utilizada na 1ª versão da aplicação para a atualização dos dados locais da aplicação móvel não se apresentava como sendo uma boa prática, pelo que mesmo desde o início da sua implementação, sabia-se que se teria de planear outra solução.

Para colmatar esta necessidade, pensou-se em criar um pequeno *webservice* que receberia os pedidos da aplicação móvel, consultaria os dados remotos e enviava-os de volta à aplicação. Na aplicação móvel seria necessário apenas de atualizar a base de dados local com os novos dados recebidos.

Para o desenvolvimento do *webservice* foi sugerida, pelo Eng. Hélder a utilização da *web framework* Sails, já numa perspetiva futurista de integrar no projeto a parte do *backoffice* de gestão da aplicação.



Figura IV.18 - Logo da *framework* Sails.js

O Sails consiste numa *framework realtime* para o desenvolvimento de aplicações *web*, que segue uma estrutura MVC e que usa JavaScript como linguagem de programação. Suporta vários tipos de base de dados (MySQL, MongoDB e PostgreSQL, entre outros), e apresenta uma nova abordagem sobre o esquema relacional que torna a modelação dos dados muito mais prática. É possível, por exemplo, associar e fazer *joins* entre uma tabela de uma base de dados em MySQL com uma *collection* NoSQL, em MongoDB.

Uma vez que na base de dados remota é guardada a data da última atualização de cada tuplo de todas tabelas, no campo **lastupdate**, para se obter os dados atualizados, basta fazer uma pesquisa pelos tuplos cujo **lastupdate** é superior ao da última atualização da aplicação móvel. No diagrama de atividades seguinte é representado todo o processo de atualização de dados da aplicação móvel, desencadeado pelo utilizador.

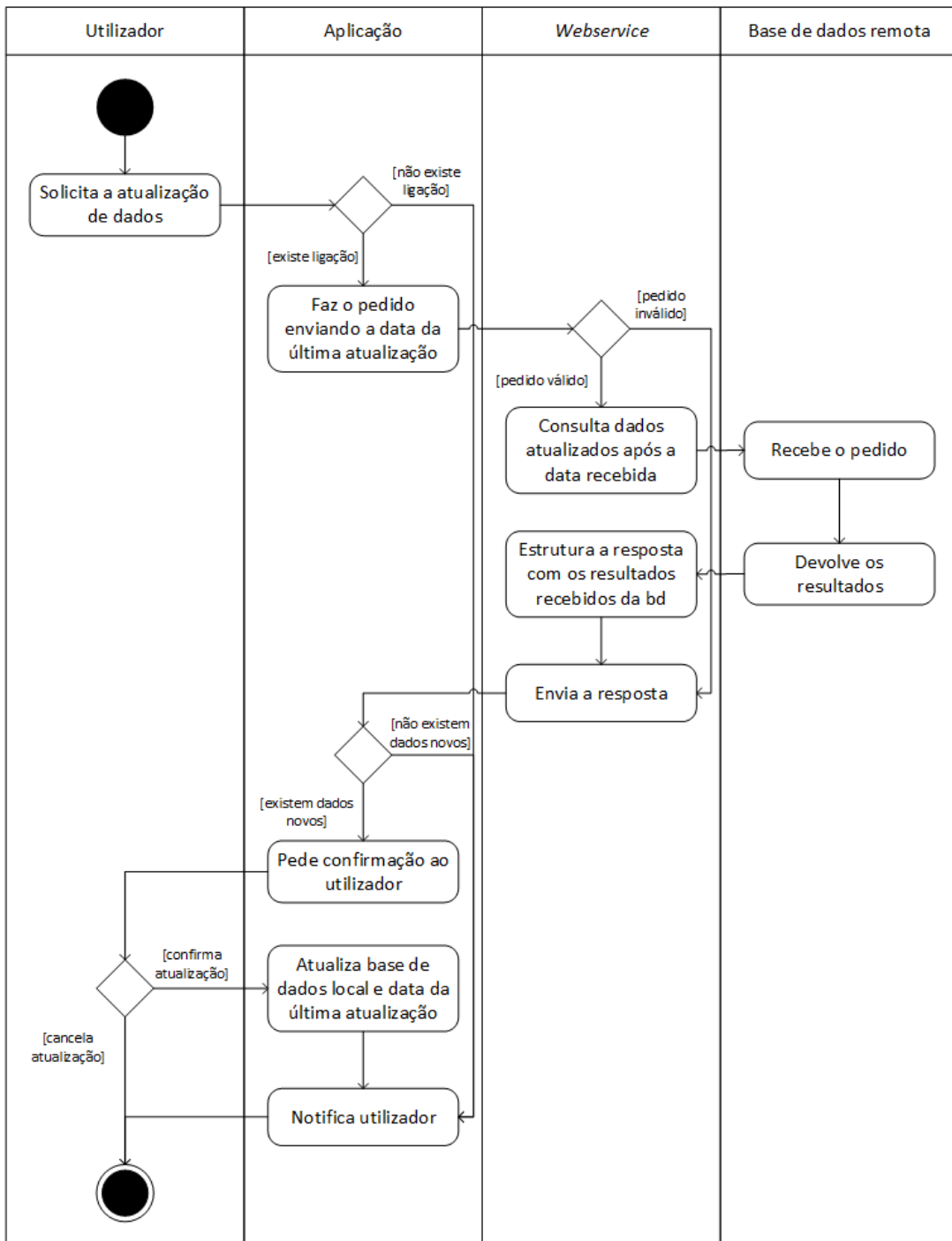


Figura IV.19 - Processo de atualização de dados da aplicação móvel

De salientar que este processo também é executado em *background* e automaticamente quando a aplicação é inicializada, sem que sejam apresentadas as notificações de erro de ligação ou inexistência de dados novos ao utilizador, sendo que apenas é apresentado o pedido de confirmação de atualização, caso existam dados a atualizar.

A atualização da base de dados local da aplicação móvel com os dados provenientes do *webservice*, no diagrama anterior representado pelo processo 'Atualiza base de dados local e data da última atualização' pode ser descrito através do fluxograma seguinte.

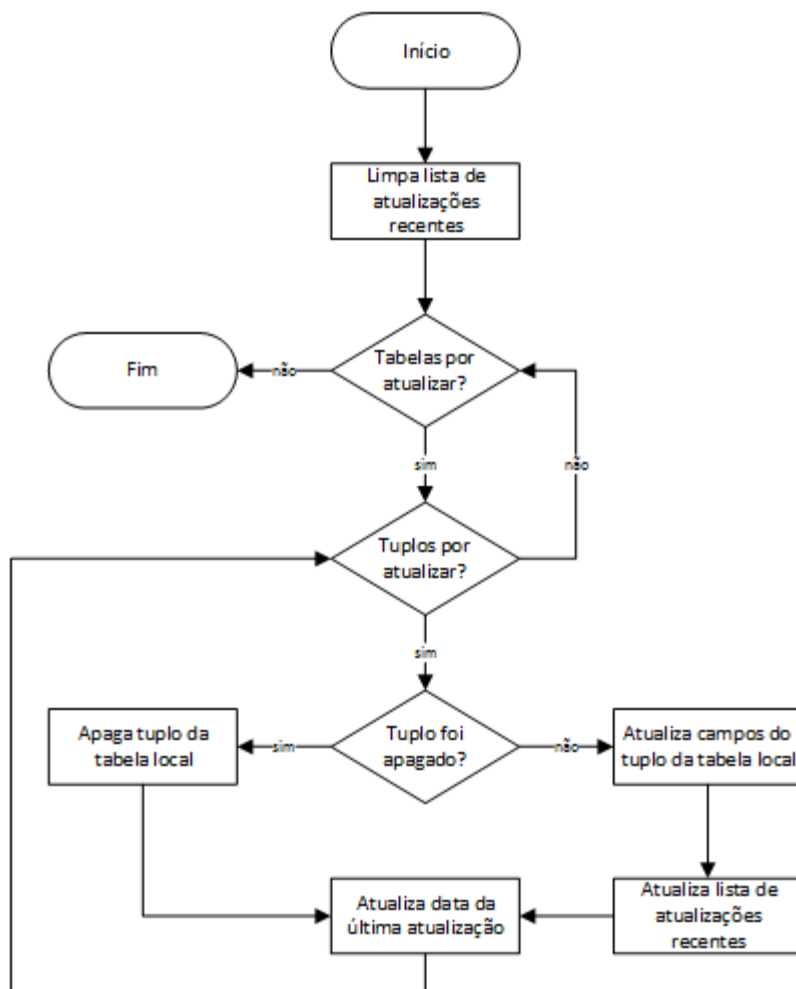


Figura IV.20 - Algoritmo de atualização dos dados recebidos pelo *webservice*

A resposta ao pedido feito pela aplicação móvel é composta pelos dados do festival atualizados e organizados por tuplos e tabelas da base de dados, de modo a facilitar o tratamento dos mesmos. Durante este tratamento de dados, são percorridas todas as tabelas e para cada tuplo é feita a verificação do estado do mesmo, ou seja, verifica-se se foi apagado ou atualizado e procede-se à remoção ou atualização do tuplo na tabela correspondente na base de dados local.

Após a atualização na base de dados local, a lista de atualizações recentes da aplicação é também ela atualizada (nova funcionalidade descrito na secção seguinte), assim como, a variável local que guarda a data da última atualização (caso a data de atualização do tuplo em questão seja superior), garantindo desta forma que no próximo pedido de atualização de dados ao *webservice*, não sejam devolvidos dados repetidos.

IV.4.3. Novas funcionalidades implementadas

Para a segunda versão da aplicação foram repensadas algumas das funcionalidades já implementadas, para além do processo de atualização dos dados que já foi referido anteriormente.

Foi criada uma nova área destina só a realização de pesquisas, que até então era feita separadamente por eventos, convidados, locais ou parceiros. A partir da versão 2.0, nos resultados da pesquisa são listados todos os eventos, convidados, locais ou parceiros que coincidam com a expressão pesquisada.

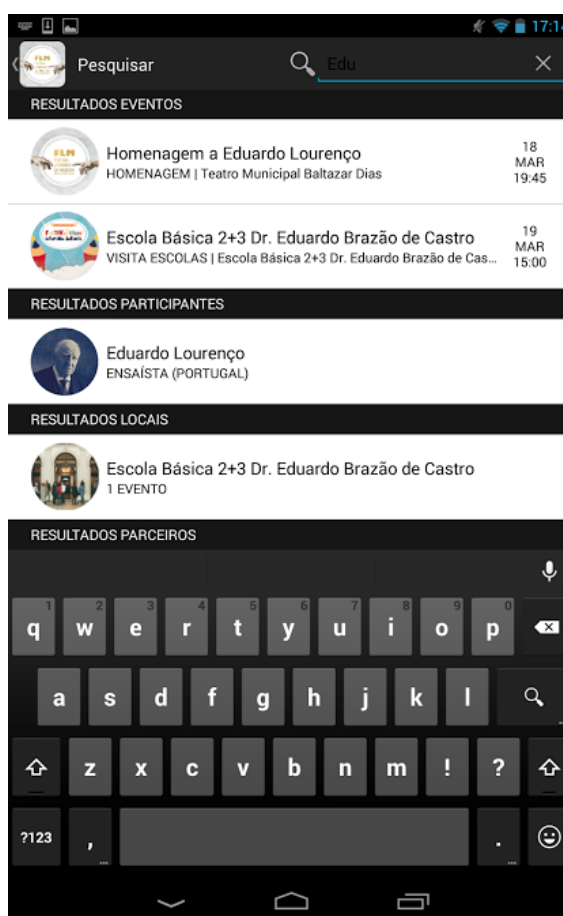


Figura IV.21 - Resultados da pesquisa organizados por Eventos, Partcipantes, Locais e Parceiros

Foram também implementadas as funcionalidades relacionadas com a localização geográfica dos eventos e parceiros. Na segunda versão da aplicação já é possível ao utilizador identificar num mapa a localização geográfica dos eventos ou da sede dos parceiros do festival, através da integração com a API do Google Maps v2.

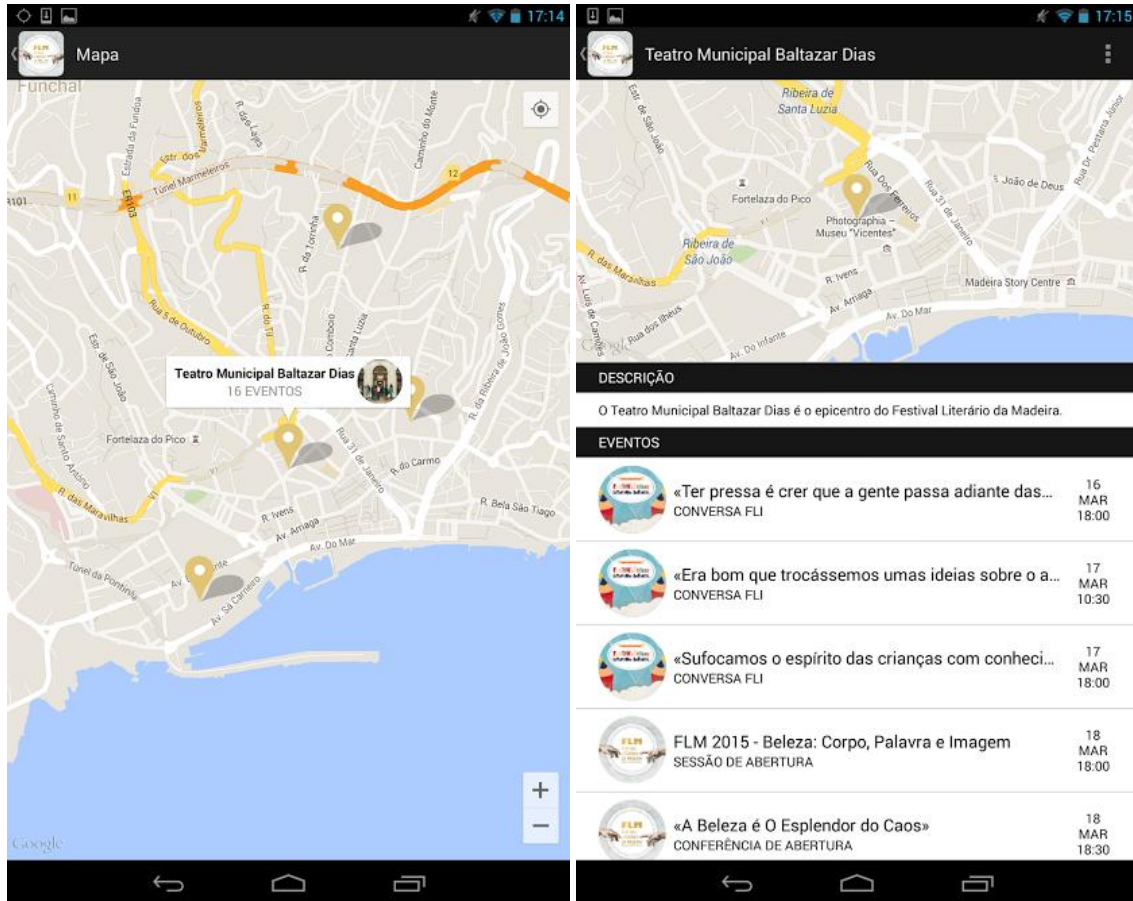


Figura IV.22 - Mapa com a localização geográfica dos locais onde decorrem os eventos

E como já foi mencionado na secção anterior, aplicação disponibiliza agora uma lista das últimas atualizações efetuadas. Esta lista facilita ao utilizador encontrar os eventos, convidados, locais ou parceiros cuja informação foi atualizada.

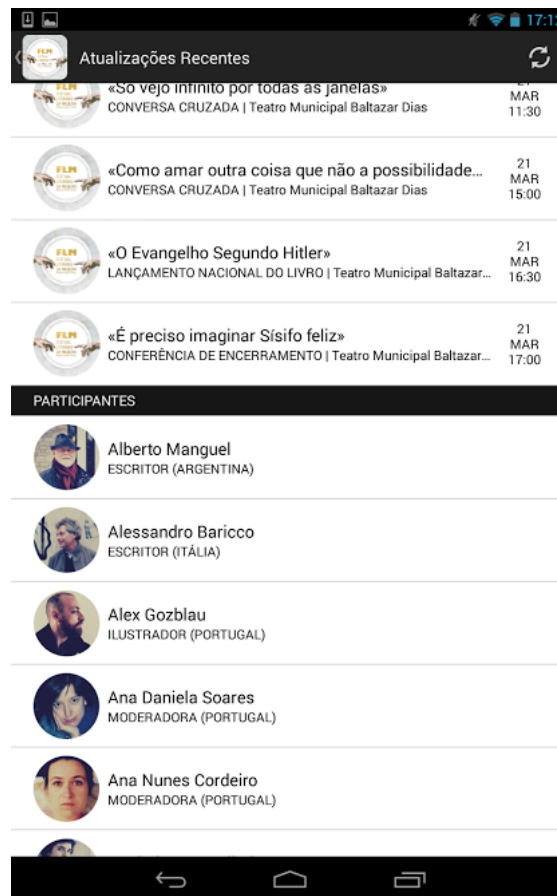


Figura IV.23 - Área com as Atualizações Recentes

V. CONCLUSÃO

“Imagination is more important than knowledge. For knowledge is limited, whereas imagination embraces the entire world, stimulating progress, giving birth to evolution.”

– Albert Einstein

V.1. SOBRE A APLICAÇÃO

Em jeito de conclusão, as tecnologias multiplataforma, e neste caso em específico, o Appcelerator Titanium, representam uma solução válida para o desenvolvimento rápido de aplicações móveis para vários ambientes, embora os problemas de *performance* diretamente proporcionais com o número de camadas existentes entre o código fonte produzido e o resultado final que é apresentado ao utilizador, serem visíveis e limitarem a experiência deste último.

Embora não se resolva o problema dos custos e recursos elevados para o desenvolvimento para vários ambientes móveis, creio que a solução para esta questão de menor desempenho das aplicações passará mesmo por desenvolver código na linguagem nativa. Colocando os dois pratos na balança, acredito que pesará sempre mais a qualidade da aplicação e a satisfação do utilizador durante a utilização da mesma, do que o número de utilizadores abrangidos. A meu ver e falando de um modo geral, creio que a aposta deverá ser feita na qualidade e não na quantidade, embora reconheça que existam sempre interesses comerciais e que o mercado seja bastante competitivo.

Em relação aos recursos utilizados para a implementação da aplicação e apesar da mesma ter sido desenvolvida apenas para Android, é de salientar que foi verificada uma redução significativa quando comparado o tempo de desenvolvimento da versão inicial com o tempo gasto para a segunda versão. Esta redução confirma o esperado a médio prazo com a utilização de tecnologias multiplataforma, neste caso específico do Appcelerator Titanium: o conhecimento e a experiência provenientes da aprendizagem inicial do uso desta tecnologia permitiram que o tempo e esforço necessários para o desenvolvimento para segunda versão fossem menores.

As alterações efetuadas para a segunda versão da aplicação relacionadas com ao *design* da mesma que seguem alguns dos Princípios de Design Android, tornaram-na mais próxima de uma aplicação nativa. E esta aproximação, se por um lado, se refletiu numa navegação e usabilidade mais fluídas e numa maior experiência de utilizador, por outro lado, levou também a um aumento de *performance* da aplicação que foi provocado pela utilização de componentes nativos.

A criação do *webservice*, que embora ainda se encontre num estado embrionário de uma API, revela-se numa mais valia para desenvolvimentos futuros no âmbito do Festival Literário da Madeira. Assim como a aplicação móvel, que já o utilizava para a obtenção dos dados do

festival, tanto o próprio *website* do festival como outras aplicações/projetos que poderão ser desenvolvidas futuramente, poderão comunicar com este, mantendo a informação centralizados.

Com tudo isto, acredito que o principal objetivo do projeto, que seria o desenvolvimento de uma aplicação móvel tanto para *smartphones* como para *tablets*, na qual fosse possível consultar o programa de eventos do Festival Literário da Madeira, foi cumprido.

Outro dos objetivos, este de carácter mais pessoal, que seria a criação de algo escalável, acredito que também tenha sido cumprido tanto a nível da estrutura da base de dados com do *front-end*, embora a aplicação não tenha tido muita projeção e adesão devido à pouca divulgação da mesma. De futuro, e tendo em conta um dos requisitos não funcionais iniciais, caso se pretenda disponibilizar a aplicação para iOS e se se optar pela utilização destas mesmas tecnologias, as alterações a serem efetuadas serão apenas a nível de *layout*, sendo que o *core* será o mesmo, comprovando, em parte, a escalabilidade da aplicação.

V.2. SOBRE O ESTÁGIO CURRICULAR

Esta foi sem dúvida uma experiência bastante enriquecedora com exigências, prazos e ritmo de trabalho reais, de uma empresa bem sucedida no mercado, que me permitiu adquirir alguma experiência a nível profissional, o que ainda não tinha sucedido durante a restante formação académica. A nível pessoal, foi também bastante motivador poder conhecer e fazer parte de um evento cultural em crescimento que consegue mover centenas de pessoas na região.

Permitiu-me também conhecer e cooperar com uma equipa de profissionais de várias áreas, que me foram transmitindo conhecimentos e experiências variados, ao longo de todo o percurso do estágio. Consequentemente, aumentei também a minha rede de contactos profissionais, que de certeza, me será útil no futuro.

Pude também colocar em prática neste projeto algumas das competências que fui adquirindo ao longo da Licenciatura e Mestrado em Engenharia Informática, e adquirir outras novas, por exemplo, técnicas, nestes caso, diretamente relacionadas com as tecnologias multiplataforma, nomeadamente o Appcelerator Titanium e a *framework* Alloy, e outras relacionadas com o desenvolvimento *web*, neste último caso relacionados com a *framework* Sails.js.

Por fim, é também bastante gratificante o fato de estar disponível a qualquer pessoa com acesso à Google Store, uma aplicação móvel modelada e criada por mim, para um evento público a nível regional.

Por toda esta experiência, o contato aconselharia a qualquer colega a realização de um estágio curricular.

V.3. PERSPETIVAS FUTURAS

Apesar de ter uma clara noção de que esta cooperação foi, sem dúvida, mais benéfica para mim como finalista de mestrado do que para a empresa em questão, uma vez que os resultados provenientes do lançamento da aplicação não terem sido de certo os esperados no início do projeto, devido aos prazos curtos e à pouca divulgação e projeção da mesma, acredito que deixei a minha contribuição através de um projeto-base que foi deixado e que, futuramente, poderá ser complementado e tomar outra dimensão, mais próxima das expectativas da empresa.

Uma das questões importantes caso seja dada continuidade a este projeto, seria a criação de testes (unitários se possível) para toda a aplicação e testá-la exaustivamente. Seria importante também testar a aplicação no máximo de dispositivos móveis possíveis (além dos que já foram utilizados).

Outra situação que ficou pendente foi o desenvolvimento da plataforma *web* para a atualização dos conteúdos da aplicação, que poderia representar também um desafio interessante, uma vez que poderia ser integrado com o *webservice* desenvolvido em Sails.js.

De todo o modo e num futuro próximo, gostaria de continuar ligada ao projeto, tentando inovar através da introdução de novas tecnologias e ferramentas - até porque o Appcelerator Titanium neste momento já não é *freeware* - dando a minha contribuição para o seu crescimento e expansão, tanto em quantidade e qualidade com novas ideias e conceitos.

REFERÊNCIAS

- [1] "FLM - Newsletter," 20 Março 2013. [Online]. Available: <http://2013.festivalliterariodamadeira.pt/imprensa/newsletters>.
- [2] A. Goldenberg, "Pros and Cons of Developing Native vs. Cross-Platform Web-Based Mobile Application," 30 Julho 2013. [Online]. Available: <http://www.dbbest.com/blog/pros-and-cons-of-developing-native-vs-cross-platform-web-based-mobile-application/>. [Accessed 04 Outubro 2013].
- [3] W. J. Francis, "Cross-platform vs. native development: Corona SDK," 03 Julho 2013. [Online]. Available: <http://www.techrepublic.com/blog/software-engineer/cross-platform-vs-native-development-corona-sdk/>. [Acedido em 03 Outubro 2013].
- [4] B. Sandu, "Mobile Frameworks To Develop Multi-Platform Apps - 26 Items," Design Your Way, [Online]. Available: <http://www.designyourway.net/blog/resources/mobile-frameworks-to-develop-multi-platform-apps-26-items/>. [Acedido em 03 Outubro 2013].
- [5] J. O'dell, "5 Cross-Platform Mobile Development Tools You Should Try," Mashable, 11 Agosto 2010. [Online]. Available: <http://mashable.com/2010/08/11/cross-platform-mobile-development-tools/#Qos0hWlp2kkc>. [Acedido em 09 Outubro 2013].
- [6] P. Viswanathan, "Top 5 Tools for Multi-Platform Mobile App Development," about tech, [Online]. Available: <http://mobiledevices.about.com/od/mobileappbasics/tp/Top-5-Tools-Multi-Platform-Mobile-App-Development.htm>. [Acedido em 09 Outubro 2013].
- [7] Freak Sense, [Online]. Available: <http://codemink.com/cross-platform-mobile-development-framework-which-one-should-you-choose/>. [Acedido em 09 Outubro 2013].
- [8] "RhoMobile Suite Documentation," RhoMobile, [Online]. Available: <http://docs.rhobile.com/en/2.2.0/rhodes/introduction>.
- [9] C. Warren, "The Pros and Cons of Cross-Platform App Design," Mashable, 16 Fevereiro 2012. [Online]. Available: <http://mashable.com/2012/02/16/cross-platform-app-design-pros-cons/#qIM5CeJxDqkQ>. [Acedido em 04 Outubro 2013].
- [10] M. Elkharyshy, "PhoneGap VS Titanium, in few points...," BADR, 10 Abril 2013. [Online]. Available: <http://badrit.com/blog/2013/4/10/phonegap-vs-titanium-in-few-points#.Vf02v99Viko>. [Acedido em 09 Outubro 2013].
- [11] "Appcelerator Titanium vs. Phonegap: Which is the better Mobile Development Platform?," DZone, 26 Maio 2011. [Online]. Available: <http://mobile.dzone.com/news/appcelerator-titanium-vs>. [Acedido em 09 Outubro 2013].
- [12] Appcelerator, "Supporting Multiple Platforms in a Single Codebase," Appcelerator, [Online]. Available: http://docs.appcelerator.com/titanium/3.0/#!/guide/Supporting_Multiple_Platforms_in_a_Single_Codebase.
- [13] [Online]. Available: <https://play.google.com/store/apps/details?id=fr.festivaldecannes>.
- [14] A. Silva e C. Videira, UML - Metodologias e Ferramentas CASE, 2ª ed., vol. 1, Lisboa: Centro Atlântico, 2005.
- [15] "Software Engineering Institute - Carnegie Mellon University," 06 06 2015. [Online]. Available: <http://www.sei.cmu.edu/architecture/definitions.html>. [Acedido em 06 06

2015].

- [16] L. Lockwood e L. Constantine, Usability by Inspection: Collaborative Techniques for Software and Web Applications, 2003.
- [17] “MySQL Workbench,” [Online]. Available: <http://www.mysql.com/products/workbench/>.
- [18] “DB Browser for SQLite,” [Online]. Available: <http://sqlitebrowser.org/>.
- [19] Appcelerator, “The Appcelerator Platform,” Appcelerator, [Online]. Available: <http://www.appcelerator.com/>. [Acedido em 14 Outubro 2013].
- [20] Appcelerator, “Titanium Platform Overview,” Appcelerator, [Online]. Available: http://docs.appcelerator.com/platform/latest/#!/guide/Titanium_Platform_Overview.
- [21] Appcelerator, “Titanium Mobile Overview,” Appcelerator, [Online]. Available: http://docs.appcelerator.com/platform/latest/#!/guide/Titanium_Mobile_Overview. [Acedido em 14 Outubro 2013].
- [22] Appcelerator, “Titanium SDK,” Appcelerator, [Online]. Available: http://docs.appcelerator.com/platform/latest/#!/guide/Titanium_SDK.
- [23] Appcelerator, “Alloy Framework,” Appcelerator, [Online]. Available: http://docs.appcelerator.com/platform/latest/#!/guide/Alloy_Framework.
- [24] Appcelerator, “Alloy,” Appcelerator, [Online]. Available: <http://docs.appcelerator.com/platform/latest/#!/api/Alloy>.
- [25] “Android Design Principles,” Google, [Online]. Available: <http://developer.android.com/design/get-started/principles.html>.
- [26] “Android Design Principles,” Google, [Online]. Available: <http://developer.android.com/design/get-started/principles.html>.
- [27] Google, “Layout- Metrics & keylines,” [Online]. Available: <http://developer.android.com/design/style/metrics-grids.html>.
- [28] Google, “Style- Typography,” [Online]. Available: <http://developer.android.com/design/style/typography.html>.

ANEXOS

ANEXO 1. DOCUMENTO DE REQUISITOS



APLICAÇÃO - FESTIVAL LITERÁRIO DA MADEIRA

Documento de Requisitos

Sofia Sousa
csofi@sousa@gmail.com

Índice

Introdução	1
Objectivos	2
Público-alvo.....	2
Principais Funcionalidades.....	3
Requisitos da Aplicação.....	4
Requisitos Funcionais.....	4
Requisitos Não Funcionais.....	6
Proposta Versão 1.0	7
Conclusão	8

Introdução

Ao longo dos meses de Outubro e Novembro, foram preparadas e realizadas entrevistas aos representantes das várias áreas da empresa, com o objectivo de serem recolhidas as impressões iniciais sobre os conteúdos que a aplicação móvel deverá abranger. As entrevistas tiveram em vista o levantamento e a elaboração dos requisitos funcionais e não funcionais da aplicação a ser desenvolvida, que serão apresentados nas secções seguintes deste relatório e deverão ser analisados e sujeitos a aprovação por parte da empresa. Cada uma das entrevistas teve um foco principal tendo em conta a área e funções desempenhadas pelos entrevistados na empresa. Todas as entrevistas decorreram nas instalações da empresa e o respectivo áudio foi gravado para posterior análise.

Na primeira entrevista, realizada no dia 18 de Outubro, foi apresentado o contexto no qual está inserido o Festival Literário da Madeira (FLM) e foram abordados vários pontos como o conceito do festival, o programa de eventos, a diferença entre macroevento e microevento, os tipos de eventos: conferências, mesas, eventos musicais, exposições de artes plásticas, entre outros; os locais onde ocorrem, os tipos de parceiros do festival, entre outros. Falou-se também da necessidade da aplicação ser proactiva, no sentido em que deve notificar o utilizador sobre os conteúdos que são do seu interesse, sem que tenha que ser este a tomar iniciativa.

Totalmente virada para a área de distribuição e vendas, na segunda entrevista, que decorreu no dia 25 de Outubro, falou-se, principalmente, sobre quais os canais de vendas da empresa, da alteração do fluxo de vendas durante o festival, os tipos de parcerias com editoras e livrarias e, por fim, a ponderação da venda e reserva de livros físicos e digitais através da aplicação móvel e que outros procedimentos esta funcionalidade implicaria.

Na terceira entrevista, que aconteceu no dia 1 de Novembro, foram postos sobre a mesa vários pontos relacionados com as parcerias e trocas de serviços, dos quais se destacou a diferenciação entre os parceiros do festival e os associados do festival e respectivas responsabilidades. Na segunda parte da entrevista, falou-se sobre a ideia de ser desenvolvido um roteiro turístico literário que reúna locais marcantes para autores nacionais e internacionais, que passaram pela região.

Na última entrevista, realizada no dia 20 de Novembro, foram focados dois pontos distintos do projecto: o Festivalinho Literário Infantil (FLI) e os associados do festival. O FLI pretende a aproximação entre os mais novos e os livros através da realização de eventos tanto nas escolas para as crianças, como conferências destinadas aos pais das crianças. Em relação aos associados do festival, foi abordada uma outra vertente desejada para a aplicação, que visa a disponibilização de informações sobre vários serviços úteis na região para o utilizador que se encontra cá de passagem, sozinho ou acompanhado da respectiva família e não só, passando a aplicação móvel de um simples produto a um serviço fundamental.

Objectivos

Dada a natureza deste projecto e relativamente ao Festival Literário da Madeira, o principal objectivo consiste, de um modo geral, na criação de uma aplicação móvel, na qual os utilizadores possam consultar informações referentes ao programa, eventos, locais e participantes do festival. Pretende-se também dar a conhecer ao utilizador os parceiros e os associados do festival, assim como as suas áreas de negócio e valências.

Uma vez que surgiram e surgirão outras ideias para a aplicação, que podem ou não estar directamente relacionadas com o festival, mas que não serão desenvolvidas e implementadas ao mesmo tempo, pretende-se que seja criada uma base robusta, tanto a nível de estruturas de dados como de *design*, à qual seja possível acrescentar posteriormente outros conteúdos, assegurando sempre a consistência, a coerência e o bom funcionamento da aplicação.

Público-alvo

O público-alvo consiste num conjunto de pessoas que reúnem características e interesses semelhantes em relação ao produto ou serviço que se está a desenvolver, ou seja, neste caso, representa o conjunto dos vários grupos de futuros utilizadores da aplicação.

Sendo que, como foi falado durante as entrevistas, o festival apresenta-se como um conjunto de eventos que não se focam somente na literatura, o público-alvo da aplicação móvel abrange, de uma maneira geral, todos utilizadores de *smartphones* e outros dispositivos móveis que se interessem por eventos culturais, acompanhados, ou não, das respectivas famílias e, de um ponto de vista mais específico, abrange também potenciais parceiros e associados, participantes convidados do festival, assim como, o todo o *staff*.

A plataforma *web*, que fará a gestão dos conteúdos da aplicação móvel do festival, será desenvolvida tendo em conta que os seus utilizadores serão membros da empresa e estão, por isso, familiarizados com a temática do festival e com a dinâmica da empresa.

Principais Funcionalidades

Dada a versatilidade de funcionalidades ambicionadas pelos membros da empresa para a aplicação móvel a ser desenvolvida, foi decidido que a sua implementação será feita através do desenvolvimento de módulos. Cada módulo irá reunir um conjunto de funcionalidades relacionadas entre si e que dizem respeito a um âmbito do projecto.

Assim, do módulo referente ao Festival Literário da Madeira farão parte todas as funcionalidades relacionada com a consulta do programa do festival, assim como a pesquisa, filtragem e navegação entre os conteúdos, como eventos, locais, participantes convidados e parceiros. Está também incluída neste módulo a implementação do *feed* de actualizações do festival e respectivas notificações espontâneas ao utilizador.

Outro módulo reunirá todas a funcionalidades referentes aos associados do festival como a pesquisa por categorias, a disponibilização das vantagens oferecidas e informação sobre a área de negócio, os contactos e a informação da localização e de como lá chegar.

Os restantes módulos irão incluir, cada um, as funcionalidades das restantes ideias para a aplicação móvel. Assim, o roteiro turístico, roteiro literário, consulta e navegação entre os respectivos locais e todas as funcionalidades relacionadas irão constituir um módulo e, por fim, as funcionalidades relacionadas com a venda e reserva de livros irá constituir o último módulo previsto.



Figura 1 - Representação da divisão do projecto em módulos

Requisitos da Aplicação

Descritas as principais funcionalidades dos módulos da aplicação, seguem-se os requisitos da aplicação que representam as propriedades, funções ou comportamentos que o sistema requer e que foram expressos durante as entrevistas. As seguintes listas de requisitos funcionais e não funcionais deverão ser alvo de análise por parte da empresa, uma vez que a sua aprovação permite que se inicie a próxima fase do desenvolvimento do projecto.

Requisitos Funcionais

Os requisitos funcionais representam o que a aplicação deve fazer: descrevem as funções, os cálculos, a manipulação ou processamento de dados que esta deve ser capaz de realizar. Estes requisitos estão organizados segundo os módulos aos quais pertencem, como se pode observar de seguida.

MÓDULO I - Guia do FLM

RF1 - A aplicação deve permitir ao utilizador a visualização do programa de eventos do festival.

RF2 - A aplicação deve permitir ao utilizador pesquisar os eventos do festival, ordená-los cronologicamente e filtrá-los por categorias/tipos, por local ou por participantes convidados.

RF3 - A aplicação deve disponibilizar informação relativa a cada evento do festival, como tema, título, descrição, local, data/horário e participantes convidados.

RF4 - A aplicação deve diferenciar visualmente os eventos do FLI dos restantes eventos do FLM.

RF5 - A aplicação deve informar sobre qual o público-alvo de cada evento do festival.

RF6 - A aplicação deve disponibilizar informação sobre o preço e locais onde podem ser comprados os bilhetes para os eventos pagos.

RF7 - A aplicação deve disponibilizar ao utilizador a possibilidade de efectuar a reserva de bilhetes para os eventos pagos.

RF8 - A aplicação deve disponibilizar a lista dos locais onde decorrem os eventos do festival.

RF9 - A aplicação deve permitir ao utilizador visualizar uma pequena descrição, a localização no mapa e informações úteis sobre cada local, como o *link* para a página do local, os contactos e como lá chegar.

RF10 - A aplicação deve disponibilizar informação relativa a cada participante convidado do festival, como foto, uma pequena biografia sobre o participante convidado e os seus principais feitos/obras publicadas, acompanhados de *links* para sítios que complementem esta informação.

RF11 - A aplicação deve disponibilizar, para cada participante convidado, a lista de eventos do festival nos quais este participa.

RF12 - A aplicação deve disponibilizar um *feed* de actualizações das informações relativas ao programa do festival.

RF13 - A aplicação deve notificar o utilizador sempre que seja feita alguma actualização em relação ao programa do festival.

RF14 - A aplicação deve permitir ao utilizador configurar a periodicidade das notificações do *feed* de actualizações.

RF15 - A aplicação deve disponibilizar uma lista dos parceiros e promotores do festival.

RF16 - A aplicação deve disponibilizar informação sobre cada parceiro: pequena descrição, informações úteis (localização, site).

MÓDULO II - Associados do Festival

RF17 - A aplicação deve disponibilizar uma lista dos Associados do Festival.

RF18 - A aplicação deve permitir ao utilizador pesquisar os associados, ordená-los alfabeticamente e por classificação, e filtrá-los por categorias consoante a sua área de negócio.

RF19 - A aplicação deve permitir a classificação de cada associado por parte do utilizador.

RF20 - A aplicação deve indicar os associados mais próximos geograficamente do local de cada evento.

RF21 - A aplicação deve disponibilizar informação sobre as vantagens que cada associado proporciona.

RF22 - A aplicação deve disponibilizar a localização e informações sobre como chegar ao local onde se pode usufruir dos serviços que cada associado proporciona.

RF23 - A aplicação deve disponibilizar os contactos de cada associado assim como o *link* para a respectiva página externa.

MÓDULO III - Roteiro Turístico

RF24 - A aplicação deve disponibilizar uma lista dos principais locais turísticos da região, organizados por categorias.

RF25 - A aplicação deve permitir ao utilizador pesquisar os locais do roteiro, ordená-los alfabeticamente e filtrá-los por tipo.

RF26 - A aplicação deve apresentar uma pequena descrição local.

RF27 - A aplicação deve apresentar ao utilizador a opção 'ver mais' e redirecioná-lo para respectiva página no *site*.

RF28 - A aplicação deve apresentar uma lista de locais em destaque, sugeridos pelos responsáveis pelo roteiro.

MÓDULO IV - Loja

RF29 - A aplicação deve permitir ao utilizador a compra de livros digitais ou físicos e a reserva de livros físicos.

RF30 - A aplicação deve disponibilizar uma lista de livros disponíveis para compra.

RF31 - A aplicação deve permitir ao utilizador pesquisar os livros disponíveis, ordená-los alfabeticamente ou por preço e filtrá-los por tipo ou autor.

RF32 - A aplicação deve permitir ao utilizador consultar informação sobre cada livro, como um pequeno resumo, o(s) autor(es), a capa e o preço.

Requisitos Não Funcionais

Os requisitos não funcionais descrevem como a aplicação deve se comportar a nível de restrições, atributos e objectivos de qualidade. Neste relatório, os requisitos não funcionais estão organizados nas categorias de Usabilidade, Portabilidade, Acessibilidade, Disponibilidade, Manutenção e Escalabilidade, sendo que, podem existir outras.

Portabilidade

RNF1 - A aplicação deve ser desenvolvida para Android (2.1 ou superiores) e iOS (7).

RNF2- A aplicação deve ser desenvolvida para *smartphones* e *tablets*.

Usabilidade

RNF3 - A aplicação deve ser multilingue.

Acessibilidade

RNF4 - Algumas funcionalidades da aplicação requerem que o dispositivo esteja conectado à *internet*.

RNF5 - Algumas funcionalidades da aplicação requerem acesso à tecnologia GPS.

Disponibilidade

RNF6 - A aplicação deve funcionar mesmo que o dispositivo não se encontre conectado à *internet*.

RNF7 - A aplicação deve disponibilizar sempre os conteúdos relacionados com o festival, como programa previsto, informação sobre os eventos, locais, participantes convidados e parceiros, sem que seja necessária conexão à *internet*.

RNF8 - A aplicação necessita de conexão à *internet* para efectuar as actualizações, assim como para permitir ao utilizador realizar reservas e classificações.

Manutenção

RNF9 - A aplicação deve disponibilizar informação actualizada que deverá ser gerida pelos respectivos responsáveis.

Escalabilidade

RNF10 - A aplicação deve suportar toda a informação relacionada com o festival e suportar futuras extensões.

Proposta Versão 1.0

Analisada toda a documentação e informação recolhida tanto nas reuniões como nas entrevistas, através das opiniões, expectativas e necessidades expressadas pelos membros da empresa, foi feita uma selecção sobre quais os pontos essenciais que uma primeira versão da aplicação deverá contemplar, e que serão de seguida apresentados.

Assim, numa primeira fase será dada mais relevância às funcionalidades que permitirão ao utilizador o acompanhamento do festival, a obtenção de informação sobre os eventos, locais e participantes convidados. Ou seja, será desenvolvido em primeiro lugar, maioritariamente, o primeiro módulo - Guia do FLM, sendo que, será sempre tido em conta a construção de uma base robusta capaz de suportar possíveis alterações futuras, expansões e inclusões dos restantes módulos. Os aspectos deste módulo que requerem mais tempo de implementação, como a reserva de bilhetes, *feed* de actualizações ou a incorporação de mapas na aplicação, darão prioridade aos requisitos, tanto do módulo I como do módulo II, que possam ser implementados em períodos de tempo mais reduzidos.

A segunda prioridade será a implementação do módulo II - Associados do Festival, que está diretamente relacionado com o módulo I e que pretende dar a conhecer e oferecer ao utilizador um conjunto de serviços úteis de uma forma espontânea, proporcionados pelos associados.

Alguns pontos dos módulos, que neste momento ainda estão a ser trabalhados e por isso não estão bem definidos, farão, eventualmente, parte de novas versões que serão implementadas sobre a mesma base.

Conclusão

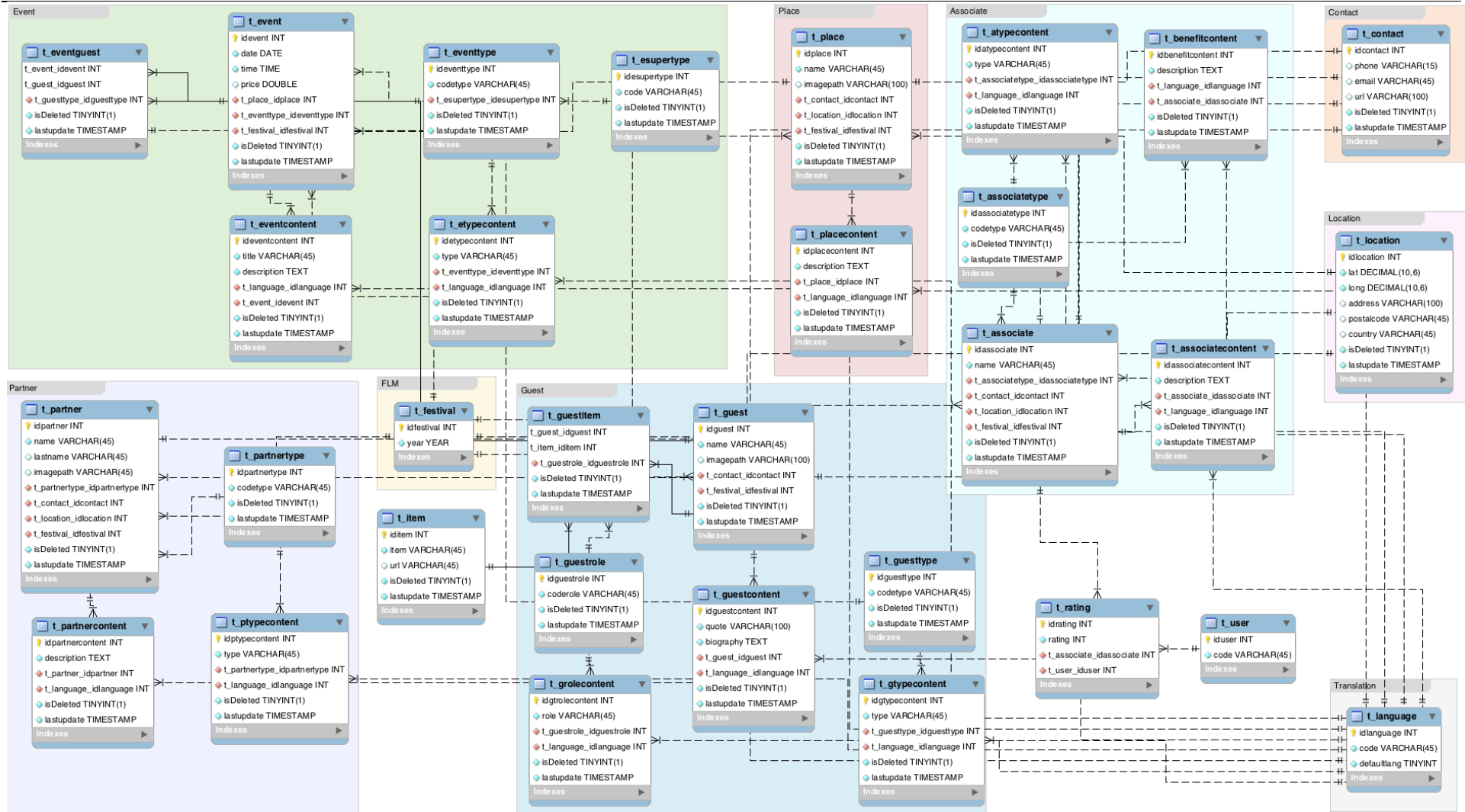
Após a recolha e estudo dos factos, conclui-se que, apesar do principal foco ser realmente o Festival Literário da Madeira, o propósito da aplicação móvel a ser desenvolvida vai mais além do que ser um simples guia para o festival.

Tendo em conta as proporções que o projecto tem vindo a tomar, resultantes da ambição e criatividades dos membros da empresa, conclui-se que, de facto que, a divisão da implementação da aplicação móvel em módulos apresenta-se como uma boa prática e a mais adequada perante a situação. A utilização de módulos permite um desenvolvimento gradual da aplicação, que leva a uma maior organização, estruturação, definição e consolidação dos conteúdos do sistema que têm de ser trabalhados.

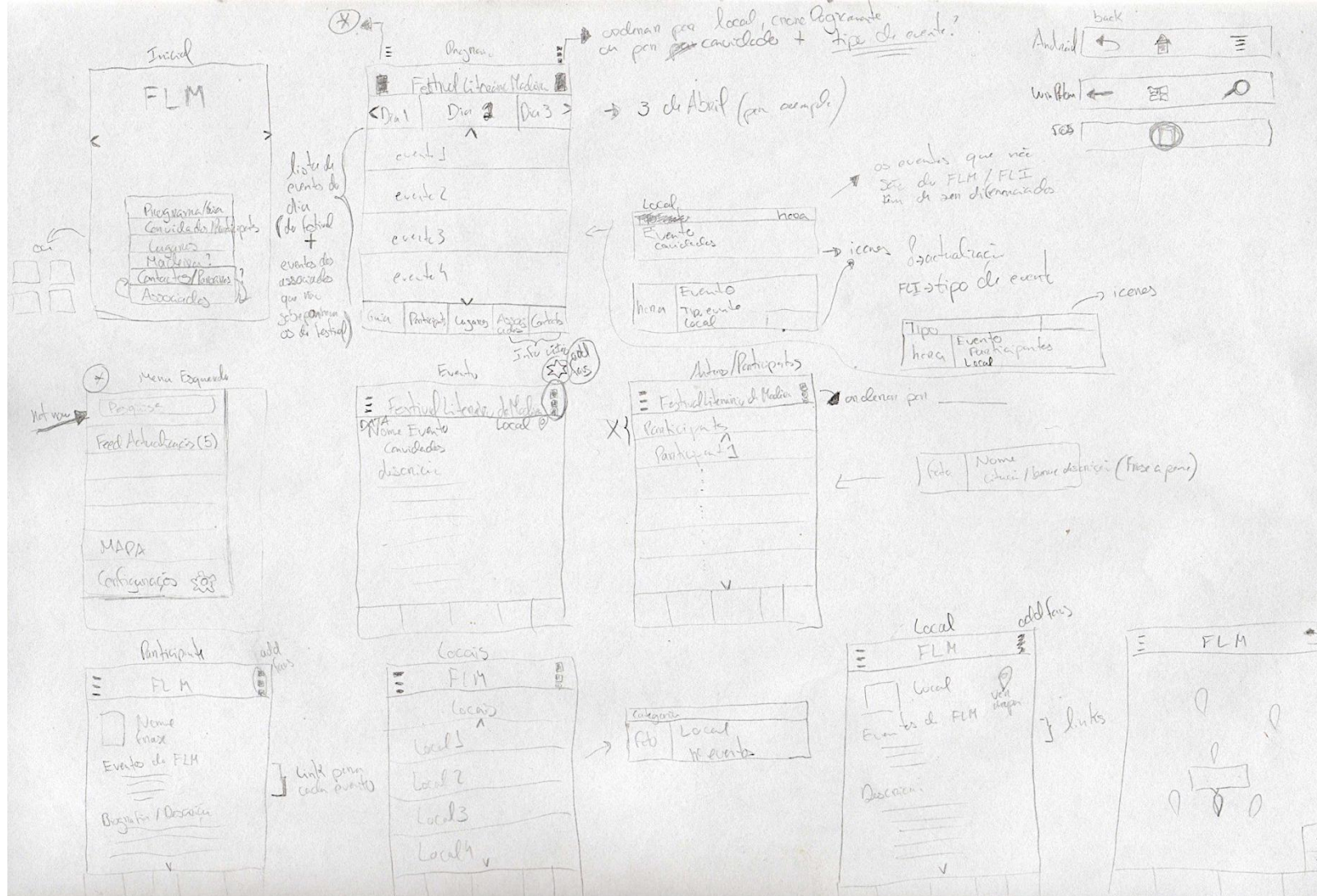
O desenvolvimento previsto utilizando tecnologias multiplataforma também ajudará a que a aplicação chegue a um maior número de utilizadores que possuam *smartphones* ou *tablets* com os vários sistemas operativos móveis, o que representa uma mais-valia.

Atingido um consenso em relação aos requisitos funcionais e não funcionais da aplicação, dar-se-á início às próximas fases de desenvolvimento do projecto, a fase da modelação do desenho conceptual, seguido do desenho gráfico e protótipos.

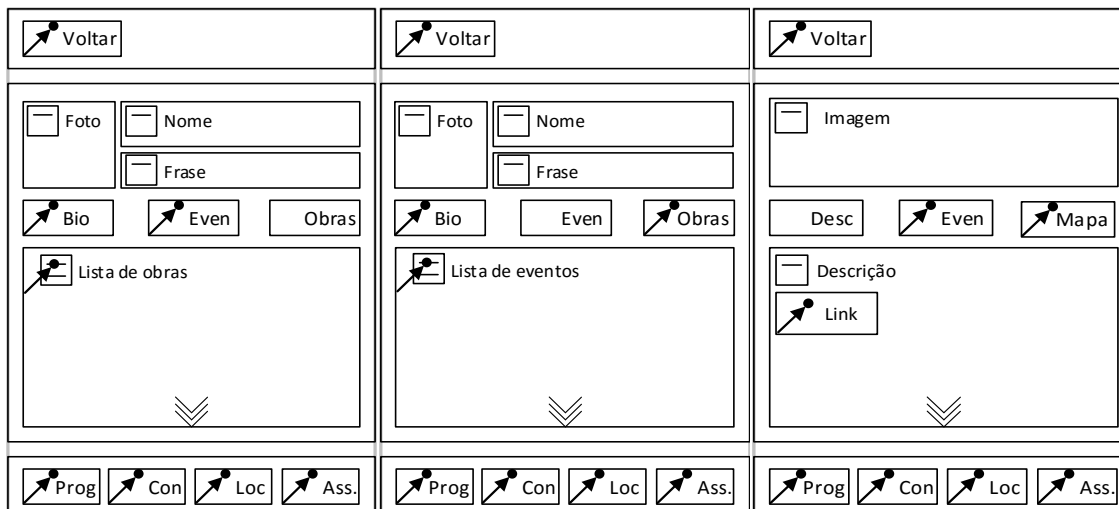
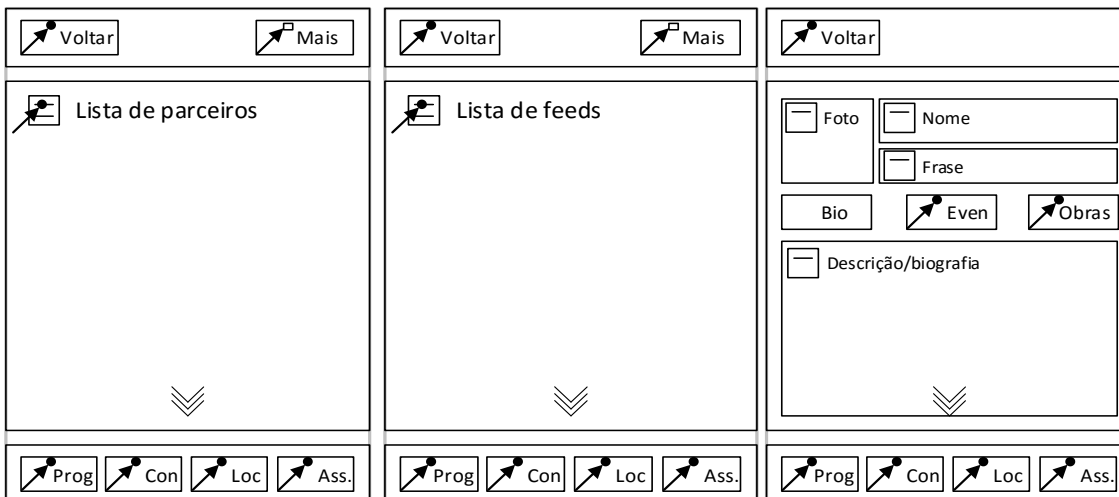
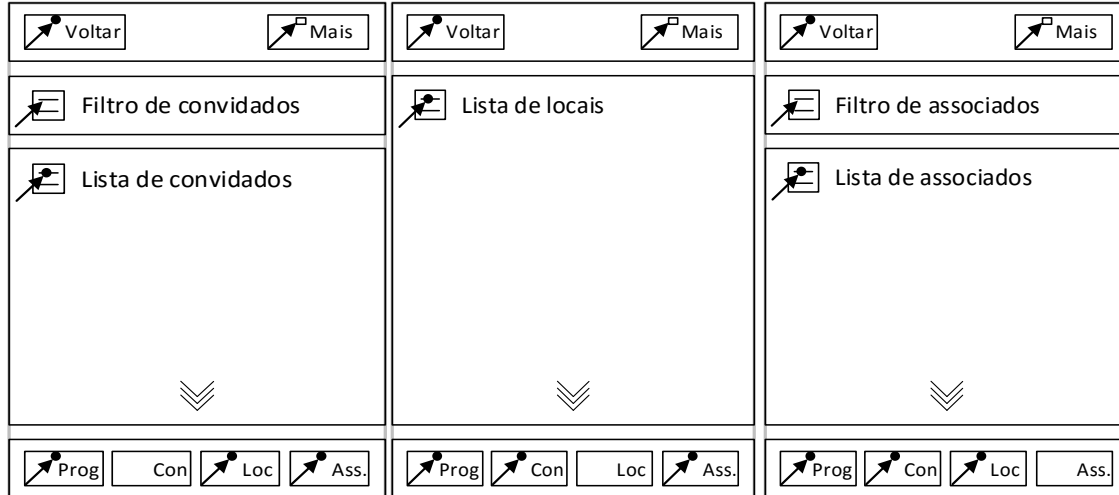
ANEXO 2. ESQUEMA RELACIONAL DA BASE DE DADOS

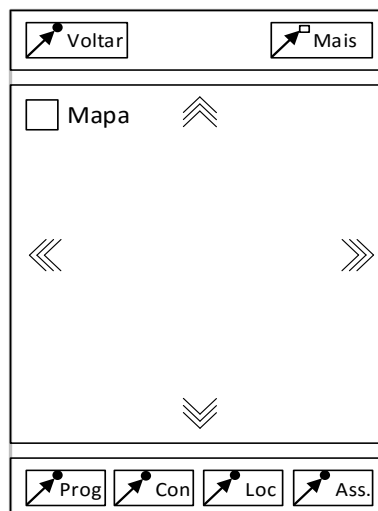
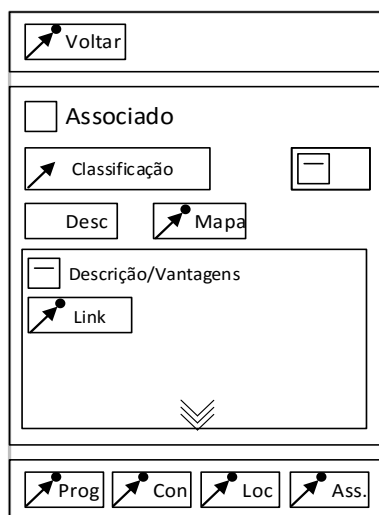
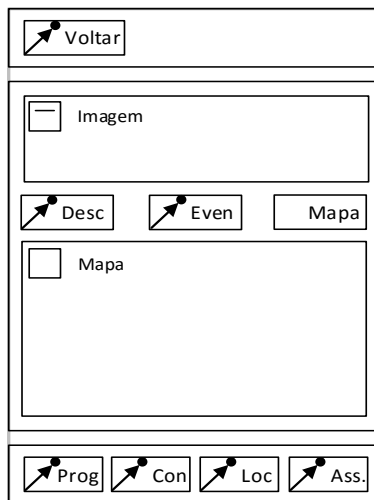


ANEXO 3. MOCKUPS



ANEXO 4. PROTÓTIPOS ABSTRATOS CANÓNICOS (CAPS)





ANEXO 5. PROTÓTIPOS DE ALTA FIDELIDADE

