

Mestrado em Engenharia Informática
Estágio
Relatório Final

Plataforma de Distribuição de Conteúdos Georreferenciados

João Gabriel Gama Pinto Bessa
jbessa@student.dei.uc.pt

Orientador DEI:

Prof. Doutor Paulo Simões

Orientador Enso Origins:

Eng. Carlos Ramos

Data: 28, Agosto de 2014



FCTUC DEPARTAMENTO
DE ENGENHARIA INFORMÁTICA
FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Resumo

O presente documento descreve os resultados do trabalho desenvolvido na empresa Enso Origins, no âmbito do estágio do Mestrado em Engenharia Informática.

O âmbito deste estágio consistiu na criação de um sistema com duas aplicações *web* e uma aplicação móvel para distribuição de conteúdos georreferenciados. O objectivo passa por conseguir orientar um utilizador num dado local, quer seja um espaço interior ou exterior, tendo ao seu dispor informações sobre o local (em forma de texto, imagens, *links* e/ou vídeos), opiniões de outros visitantes e mapas com funcionalidades de navegação. A plataforma web vai permitir criar zonas, onde se pode adicionar, personalizar e editar conteúdos com informações diversas. A plataforma móvel, através da utilização do WIFI e GPS para geolocalização, servirá para consulta de conteúdos previamente adicionados numa das aplicações *web*.

É apresentado detalhadamente, ao longo de todo o documento, cada fase do projecto, desde o estado da arte à arquitectura, sustentando as várias fases de especificação que auxiliaram na implementação do projecto. Ainda engloba os detalhes da implementação e testes efectuados com vista à avaliação dos requisitos definidos.

Palavras-Chave

“Distribuição de conteúdos”, “georreferenciação”, “aplicação móvel”, “ubiquidade”
“aplicação web”.

Agradecimentos

Durante a minha vida académica foram várias as pessoas que me apoiaram, tendo ajudado em muito o meu percurso.

Um enorme agradecimento ao meu orientador e Professor, Doutor Paulo Simões, pelo seu apoio, disponibilidade, críticas construtivas, paciência e dedicação que demonstrou ao longo de todo o percurso desta dissertação.

Um sincero agradecimento ao meu orientador Eng. Carlos Ramos, pela paciência, dedicação, apoio, disponibilidade, ensinamentos e amizade demonstrada ao longo dos últimos anos, em especial durante o estágio, cuja contribuição foi muito importante para ultrapassar as adversidades encontradas ao longo da especificação e desenvolvimento do projecto.

Um especial agradecimento ao Professor, Doutor Mário Zenha Relá, pela ajuda e disponibilidade demonstrada, não só durante este estágio mas pelos ensinamentos ao longo do curso.

Um enorme agradecimento aos meus pais, não só pelo apoio incondicional e compreensão perante todas as minhas escolhas ao longo do tempo, mas também por me proporcionarem esta experiência de vida e me tornarem todos os dias uma pessoa melhor.

Um sentido agradecimento à minha irmã, pelo apoio incondicional, amizade e compreensão demonstrada sempre que necessário.

Um grande agradecimento à minha avó, pela ajuda e ensinamentos ao longo da minha vida.

Um especial agradecimento à Maria Matos, pelo incentivo, apoio, paciência, dedicação, compreensão, amizade e amor demonstrado nos bons e maus momentos.

Um grande agradecimento aos meus colegas e amigos de curso, pelas horas passadas a trabalhar e pelo apoio constante em momentos mais complicados.

Um especial agradecimento aos meus amigos Ana Lages, Cátia Guerra, Daniel Bessa, Gonçalo Maria, João Bernardo, João Gomes, Mafalda Meireles, Melanie Rodrigues, Mónica Dias, Rui Gomes, Samuel Lages, Sérgio Rodrigues e Tiago Bessa, pela amizade única e especial demonstrada ao longo da minha vida, nos bons e maus momentos.

Um sentido agradecimento ao Paulo Valdez, pela amizade, apoio e incentivo ao longo dos últimos anos.

Um agradecimento à Helena Duarte, pelo apoio e incentivo ao longo deste ano.

Por fim, mas não menos importante, agradeço à Enso Origins, aos seus colaboradores internos (Diogo Bhovan, Filipe Pires, Nuno Fidalgo, Pedro Marques e Pedro Sá) e aos externos (Pedro Loureiro e Marco Ferreira) por me terem acolhido e ajudado, sempre que necessário, na realização do meu estágio.

A todas estas pessoas o meu muito OBRIGADO!

Índice

Capítulo 1 - Introdução.....	19
1.1. Contexto.....	19
1.2. Objectivos.....	19
1.3. Sistema.....	21
1.4. Entidade acolhedora.....	22
1.5. Estrutura do documento.....	22
Capítulo 2 - Plano de trabalho.....	23
2.1. Plano inicial.....	23
2.2. Desvios ao plano.....	25
Capítulo 3 - Estado da arte.....	26
3.1. Aplicações concorrentes.....	26
3.2. Escolha e estudo das tecnologias de suporte.....	30
Capítulo 4 - Especificação do sistema.....	43
4.1. Descrição da plataforma.....	43
4.2. Levantamento de requisitos.....	47
4.3. Diagrama de casos de uso.....	54
4.4. Mockups.....	56
4.5. Arquitectura.....	57
Capítulo 5 - Implementação.....	59
5.1. Metodologia de Desenvolvimento.....	59
5.2 Aplicações.....	60
Capítulo 6 – Testes e Avaliação.....	70
6.1. Ciclo de vida dos testes.....	70
6.2. Testes unitários e Testes de integração.....	70
6.3. Testes de sistema.....	71
6.4. Testes de aceitação.....	72
6.5. Testes de usabilidade.....	73
6.6. Testes de performance.....	74
6.7. Testes de escalabilidade.....	76
Capítulo 7 – Conclusões e Trabalho Futuro.....	77
7.1. Conclusões.....	77
7.2. Trabalho Futuro.....	79
Referências.....	80
Anexos.....	83
A – Diagrama de Casos de Uso.....	85
B – Diagrama de Flow & Mockups.....	91
B.1 – Diagrama de Flow.....	91
B.2 – Mockups.....	95
C - Arquitectura.....	119

C.1 – Diagrama físico.....	119
C.2 – Diagrama de estados.....	120
C.3 – Diagrama Entidade-Relação.....	125
D – Diagrama de Gantt.....	127
D.1 – Versão 1.0.....	127
D.2 – Versão 2.0.....	129
D.3 – Versão 3.0.....	131
E - Testes.....	133
E.1 – Sistema.....	133
E.2 – Testes de Aceitação.....	162
E.3 – Testes de Usabilidade.....	170
E.4 – Testes de Performance.....	173
E.5 – Testes de Escalabilidade.....	175
F – Screenshots da Plataforma.....	182
F.1 – Aplicações web.....	182
F.2 – Aplicação móvel.....	190
G – Scrum.....	193
G.1 – Product Backlog & Sprint Planning.....	193

Lista de Figuras

Índice de Figuras

Figura 1: Arquitectura simplificada do sistema.....	21
Figura 2: Utilização de web servers [17].....	30
Figura 3: Processo de pedido de um token.....	34
Figura 4: Processo de comunicação recorrendo a tokens.....	35
Figura 5: Programação do lado do servidor[40].....	36
Figura 6: Exemplo de um pedido REST [46].....	37
Figura 7: Fluxo de dados MVC com codeigniter[50].....	38
Figura 8: Percentagem de dispositivos que correm uma dada versão Android [64].....	41
Figura 9: Arquitectura geral do sistema.....	43
Figura 10: Tipos de utilizador.....	44
Figura 11: Objectos core.....	44
Figura 12: Acesso à aplicação de gestão administrativa.....	45
Figura 13: Personalização de itens.....	45
Figura 14: Layout de Autenticação.....	46
Figura 15: Exemplo do caso de uso de personalização de um canal.....	54
Figura 16: Exemplo do caso de uso de personalização de uma planta.....	55
Figura 17: Mockup para personalizar canal.....	56
Figura 18: Mockup para personalizar planta/regiões.....	56
Figura 19: Diagrama físico.....	57
Figura 20: Diagrama de estados - personalização dos dados do canal e dos portais.....	58
Figura 21: Metodologia de desenvolvimento scrum [65].....	59
Figura 22: Comunicações existentes na plataforma.....	63
Figura 23: Modelo MVC.....	65
Figura 24: Código utilizado no servidor Apache para configurar Expires Header.....	75
Figura 25: Exemplo de teste com a ferramenta ApacheBench.....	76
Figura 26: Diagrama de Casos de Uso - Aplicação de Gestão Administrativa.....	85
Figura 27: Diagrama de Casos de Uso - Aplicação de Gestão para Clientes.....	87
Figura 28: Diagrama de Casos de Uso – Aplicação Móvel.....	89
Figura 29: Diagrama de flow da plataforma web (Administração e Cliente).....	91
Figura 30: Diagrama de Flow da aplicação móvel.....	93
Figura 31: Mockup M1.00.....	96
Figura 32: Mockup M1.01.....	96
Figura 33: Mockup M1.02.....	97
Figura 34: Mockup M1.03.....	97
Figura 35: Mockup M1.04.....	98
Figura 36: Mockup M1.05.....	98
Figura 37: Mockup M1.06.....	99
Figura 38: Mockup M1.07.....	99
Figura 39: Mockup M1.08.....	100
Figura 40: Mockup M1.09.....	100
Figura 41: Mockup M1.10.....	101
Figura 42: Mockup M1.11.....	101
Figura 43: Mockup M2.01.....	103
Figura 44: Mockup M2.02.....	103
Figura 45: Mockup M2.03.....	104

Figura 46: Mockup M2.04.....	104
Figura 47: Mockup M2.05.....	105
Figura 48: Mockup M2.06.....	105
Figura 49: Mockup M2.07.....	106
Figura 50: Mockup M2.08.....	106
Figura 51: Mockup M2.09.....	107
Figura 52: Mockup M2.10.....	107
Figura 53: Mockup M2.11.....	108
Figura 54: Mockup M2.12.....	108
Figura 55: Mockup M2.13.....	109
Figura 56: Mockup M2.14.....	109
Figura 57: Mockup M2.15.....	110
Figura 58: Mockup M2.16.....	110
Figura 59: Mockup M2.17.....	111
Figura 60: Mockup M2.18.....	111
Figura 61: Mockup M3.0.....	113
Figura 62: Mockup M3.1.....	113
Figura 63: Mockup M3.2.....	114
Figura 64: Mockup M3.3.....	114
Figura 65: Mockup M3.4.....	115
Figura 66: Mockup M3.5.....	115
Figura 67: Mockup M3.6.....	116
Figura 68: Mockup M3.7.....	116
Figura 69: Mockup M3.8.....	117
Figura 70: Diagrama físico.....	119
Figura 71: Autenticação plataforma web.....	120
Figura 72: Personalização de um canal e/ou portal.....	120
Figura 73: Personalização de uma planta.....	121
Figura 74: Personalização de uma região.....	121
Figura 75: Personalização de um item e respectivo conteúdo.....	122
Figura 76: Autenticação plataforma móvel.....	122
Figura 77: Acções num canal - plataforma móvel.....	123
Figura 78: Diagrama entidade-relação.....	125
Figura 79: Diagrama de Gantt - versão 1.....	127
Figura 80: Diagrama de Gantt - versão 2.....	129
Figura 81: Diagrama de Gantt - versão 3.....	131
Figura 82: Questionário de usabilidade.....	170
Figura 83: Acessos antes e depois de aplicadas as regras do Steve Souders.....	173
Figura 84: Avaliação da performance antes e depois de aplicadas as regras do Steve Souders	173
Figura 85: Testes - Pedidos por segundo.....	175
Figura 86: Testes - Tempo por pedido.....	179
Figura 87: Autenticação.....	182
Figura 88: Formulário de registo.....	182
Figura 89: Administração - Listagem de todos os canais da plataforma.....	183
Figura 90: Administração - canais para aprovação.....	183
Figura 91: Administração - sem canais para aprovação.....	183
Figura 92: Administração - Listagem de todos os utilizadores.....	184
Figura 93: Administração - Editar perfil de um utilizador.....	184
Figura 94: Administração - Visualizar um canal pertencente a um cliente.....	185
Figura 95: Administração - Relatório de sistema.....	185
Figura 96: Administração - Relatório de sistema (continuação).....	185

Figura 97: Lista de sem canais.....	186
Figura 98: Lista de um cliente sem gestores.....	186
Figura 99: Formulário para adicionar um gestor.....	186
Figura 100: Pedido de criação de canal pendente.....	186
Figura 101: Definir área de abrangência de um canal.....	187
Figura 102: Acesso à gestão de um canal.....	187
Figura 103: Edição dos dados de um canal.....	187
Figura 104: Menu de criação e edição de conteúdos do portal público.....	188
Figura 105: Menu de criação e edição de conteúdos do portal privado.....	188
Figura 106: Personalização da planta (regiões, itens e conteúdos dos itens).....	188
Figura 107: Moderação de comentários referentes a um canal.....	189
Figura 108: Relatório de estatísticas para clientes	189
Figura 109: Login Facebook.....	190
Figura 110: Autenticação.....	190
Figura 111: Lista canais.....	190
Figura 112: Portal Público.....	190
Figura 113: Portal Privado.....	191
Figura 114: Planta.....	191
Figura 115: Navigation Mode.....	191
Figura 116: Planta.....	191

Lista de Tabelas

Índice de tabelas

Tabela 1: Planeamento Primeiro Semestre.....	23
Tabela 2: Planeamento Segundo Semestre.....	23
Tabela 3: Comparação entre as aplicações concorrentes.....	29
Tabela 4: Ranking de Sistemas de Gestão de Bases de Dados [28].....	32
Tabela 5: Comparação entre o Canvas e a livraria Leaflet.....	39
Tabela 6: Requisitos funcionais (must) para a aplicação de gestão administrativa.....	49
Tabela 7: Requisitos funcionais (should e nice to have) para a aplicação de gestão administrativa.....	49
Tabela 8: Requisitos funcionais (must) para a aplicação de gestão para clientes.....	51
Tabela 9: Requisitos funcionais (should e nice to have) para a aplicação de gestão para clientes.....	52
Tabela 10: Requisitos funcionais (must) para a aplicação móvel.....	53
Tabela 11: Requisitos funcionais (should e nice to have) para a aplicação móvel.....	53
Tabela 12: Requisitos não funcionais para todas as aplicações.....	53
Tabela 13: Códigos de resposta utilizados pela REST API.....	61
Tabela 14: Exemplo de tabela para apresentação dos testes de sistema.....	71
Tabela 15: Regras do High Performance Websites do Steve Souders [70].....	74
Tabela 16: Comparação de ganhos na transferência de dados.....	75
Tabela 17: Comparação de ganhos no número de pedidos ao servidor.....	75
Tabela 18: Cenário de validação	76
Tabela 19: Métricas a atingir.....	76
Tabela 20: Lista de mockups da Aplicação de Gestão Administrativa.....	95
Tabela 21: Lista de mockups da Aplicação de Gestão para Clientes.....	102
Tabela 22: Lista de mockups da Aplicação Móvel.....	112
Tabela 23: Testes de aceitação dos requisitos funcionais (must) para a aplicação de gestão administrativa.....	164
Tabela 24: Testes de aceitação dos requisitos funcionais (should) para a aplicação de gestão administrativa.....	164
Tabela 25: Testes de aceitação dos requisitos funcionais (nice to have) para a aplicação de gestão administrativa.....	165
Tabela 26: Testes de aceitação dos requisitos funcionais (must) para a aplicação de gestão para clientes.....	166
Tabela 27: Testes de aceitação dos requisitos funcionais (should e nice to have) para a aplicação de gestão para clientes.....	167
Tabela 28: Testes de aceitação dos requisitos funcionais (must) para a aplicação móvel.....	168
Tabela 29: Testes de aceitação dos requisitos funcionais (should e nice to have) para a aplicação móvel.....	168
Tabela 30: Testes de aceitação dos requisitos não funcionais para todas as aplicações.....	169
Tabela 31: Teste de usabilidade na aplicação web a utilizadores da área informática.....	171
Tabela 32: Teste de usabilidade na aplicação web a utilizadores de outras áreas.....	171
Tabela 33: Teste de usabilidade na aplicação móvel a utilizadores da área informática.....	172
Tabela 34: Teste de usabilidade na aplicação móvel a utilizadores da área informática.....	172
Tabela 35: Resumo dos sprint backlogs.....	195

Glossário

Área de abrangência, é um polígono georreferenciado num mapa, referindo-se à área pertencente de um canal.

Canal, é um local georreferenciado constituído por portais (público e/ou privado).

Cloud, é um modelo para acesso a rede *on-demand*, ubíquo e conveniente para uma *pool* partilhada de recursos computacionais configuráveis (exemplos: redes, servidores, aplicações, serviços) que podem ser rapidamente fornecidos com mínimo esforço de gestão ou interacção com o fornecedor de serviços. Podem ser classificadas como públicas, privadas ou híbridas.

Conteúdo multimédia, é uma imagem, texto, *link* ou vídeo.

Framework, é um conjunto de ferramentas que promove a abstracção entre vários projectos de modo a fornecer uma funcionalidade genérica.

Heatmap, é uma representação gráfica de dados onde os valores contidos na matriz são representados por cores.

Interoperabilidade, é a capacidade de um sistema para interagir e comunicar com outro.

Item, é um objecto georreferenciado numa região constituído por conteúdo multimédia.

Model-View-Controller, é um modelo de arquitectura de software que separa a representação da informação da interacção com o utilizador.

Open Source, código de domínio público, de distribuição livre.

PHP, linguagem de programação utilizada no desenvolvimento web.

Planta, imagem com vários itens, onde cada item pode ter um local.

Portal, é uma derivação do canal podendo ser públicos ou privados.

Proxy, é um servidor intermédio que permite aos clientes fazer conexões de rede indirectas a outros serviços de rede.

Região, é um polígono georreferenciado numa área de abrangência.

Serverless, refere-se a uma arquitectura que não utiliza qualquer tipo de servidor central.

Timestamp, é um tipo de dados que permite guardar uma referência para um qualquer instante no tempo.

Acrónimos

ACID – Atomicidade, Consistência, Isolamento e Durabilidade

AP – Access Point

API – Application Programming Interface

BSD – Berkeley Software Distribution

CDN – Content Delivery Network

CEO – Chief Executive Office

CSS – Cascading Style Sheets

CSV – Comma-separated values

DEI – Departamento de Engenharia Informática

DNS – Domain Name System

ER – Entity-Relationship

FCTUC – Faculdade de Ciências e Tecnologia da Universidade de Coimbra

GPS – Global Positioning System

GSM – Global System for Mobile Communications

HTML – HyperText Markup Language

HTTP – Hypertext Transfer Protocol

HTTPS – Hypertext Transfer Protocol Secure

JSON – JavaScript Object Notation

MEI – Mestrado em Engenharia Informática

MIT – Massachusetts Institute of Technology

MVC – Model-View-Controller

NFC – Near Field Communication

OGC – Open Geospatial Consortium

PDF – Portable Document Format

QR-code – Quick Response Code

RAM – Random Access Memory

RBAC – Role-Based Access Control

REST – Representational State Transfer

RFID – Radio-Frequency IDentification

SGDB – Sistema de Gestão de Bases de Dados

SSL – Secure Socket Layer

SUS – System Usability Scale

UC – Universidade de Coimbra

URI – Uniform Resource Identifier

URL – Uniform Resource Locator

WBS – Work Breakdown Structure

WLAN – Wireless Local Area Network

XML – eXtensible Markup Language

Capítulo 1 - Introdução

O presente documento apresenta o relatório do Estágio de Mestrado em Engenharia Informática, efectuado pelo aluno João Gabriel Gama Pinto Bessa, da Faculdade de Ciências e Tecnologia da Universidade de Coimbra (FCTUC), no ano lectivo 2013/2014.

O estágio foi realizado em dois semestres, sendo que no primeiro foi definido o planeamento, efectuou-se o levantamento de requisitos, o estudo do estado da arte e desenho da arquitectura, para servir de suporte ao desenvolvimento do projecto efectuado no segundo semestre.

Neste capítulo serão abordados o contexto no qual se insere o projecto, o objectivo pretendido, uma breve introdução ao sistema, o âmbito da entidade acolhedora para o qual este projecto está a ser desenvolvido e qual a estrutura deste documento.

1.1. Contexto

Frequentemente, ao visitarmos um determinado local: museu, jardim, parque, cidade ou um qualquer outro ponto turístico, sentimos a falta de orientação e de um acompanhamento mais personalizado, ficando muitas vezes sem se saber onde ir e o que ver, não usufruindo ao máximo da nossa visita.

A constante interacção com dispositivos móveis, permite, com grande frequência, recorrer a estes para, de certa forma, potenciar cada experiência do nosso dia-a-dia. Posto isto, as aplicações móveis são criadas de modo a acompanhar o desenvolvimento tecnológico de acordo com as necessidades do utilizador, sendo assim uma importante solução tecnológica para a plataforma a desenvolver.

É desta forma que nasce este projecto, pretendendo ir de encontro à solução ao criar uma plataforma de disponibilização de conteúdos, que apoie o turista oferecendo informação georreferenciada útil e pertinente. Acessível através de uma aplicação móvel, esta permite ao utilizador não só descobrir o que o rodeia – restaurantes, museus, marcos históricos – como ainda o pode auxiliar caso se desloque a um desses espaços fornecendo toda a informação disponível com recurso a texto, imagem e vídeo, enriquecendo desta forma toda a experiência do utilizador.

Simultaneamente, ao representar uma mais valia para quem a utiliza, esta plataforma fornece ainda às entidades que exploram a plataforma comercialmente, a possibilidade de criarem uma via de comunicação privilegiada com o turista, sendo esta via ideal à divulgação de eventos e à promoção de locais/acontecimentos.

1.2. Objectivos

O objectivo deste estágio passa por encontrar soluções com recurso a diversas tecnologias, explicadas na secção 3.2, que visem a criação de um produto que apoie um visitante (utilizador) durante uma visita a algum local.

O utilizador irá conseguir descobrir mais informações sobre o local onde se encontra, qual a opinião de outros utilizadores e terá ao seu dispor um conjunto de ferramentas colaborativas que lhe permitem participar desta base de dados de conhecimento a disponibilizar a cada visitante.

Desta forma e para a realização deste estágio é necessário preencher os seguintes objectivos:

- Estado da arte:
 - Estudo das aplicações concorrentes;
 - Estudo e escolha das tecnologias.
- Especificação do sistema:
 - Levantamento de requisitos;
 - Casos de uso;
 - Mockups;
 - Arquitectura.
- Criação de um modelo de dados;
- Criação de uma aplicação *web* para gestão administrativa:
 - Gestão de clientes;
 - Gestão de canais;
 - Relatórios de sistema.
- Criação de uma aplicação *web* de gestão para clientes:
 - Gestão de canais;
 - Marcação da área de abrangência de um canal;
 - Gestão dos portais;
 - Gestão das permissões dos canais;
 - Disponibilização de conteúdos multimédia (texto, áudio, vídeo, imagens, link);
 - Relatórios de canais;
 - Identificação das zonas mais visitadas recorrendo a modelos visuais, como por exemplo *heatmaps* – não é para implementação no estágio.
- Criação de uma aplicação móvel:
 - Disponibilização de mapas com funcionalidades de navegação;
 - Visualização de conteúdos multimédia (texto, áudio, vídeo, imagens, link);
 - Disponibilização de conteúdos por posição geográfica;
 - Utilização de sinais WI-FI e/ou GPS para localização do dispositivo móvel;
 - Capacidade de utilização da câmara:
 - Análise de imagens para reconhecimento de locais – não é para implementação no estágio;
 - Tirar fotos:
 - Partilha na plataforma móvel;
 - Partilha em redes sociais – não é para implementação no estágio.
 - Leitura de QR-Codes.
 - Recolha de informações estatísticas.

A aplicação móvel a ser desenvolvida irá disponibilizar ao utilizador conteúdos previamente inseridos na plataforma web, sendo esta aplicação responsável por apresentar o utilizador, de uma forma simples, ao local que está a visitar, recomendando pontos de interesse e

permitindo uma consulta extensiva de todos os conteúdos disponibilizados. De notar que a recomendação de pontos de interesse não será efectuada para efeitos de estágio.

Toda a integração entre as plataformas web e móveis deve seguir uma arquitectura *RESTful* suportada em tecnologias que irão ser abordadas e aprofundadas na secção 3.2.

1.3. Sistema

De modo a dar a conhecer o sistema, de seguida são apresentadas as aplicações que constituem o sistema, colmatando com a explicação de cada uma das aplicações.

Infraestrutura

Da infraestrutura, como é possível ver pela figura 1, faz parte um servidor *web* que comunica com a base de dados e com as diferentes aplicações da plataforma. Existem duas aplicações *web* de gestão da plataforma e uma aplicação móvel para consulta de conteúdos. De modo a garantir a segurança e a integridade dos dados, todas as comunicações entre o servidor web e as diferentes aplicações e/ou base de dados são feitas de forma segura, recorrendo à tecnologia SSL (*Secure Socket Layer*), mais concretamente através do protocolo HTTPS (*Hypertext Transfer Protocol Secure*).

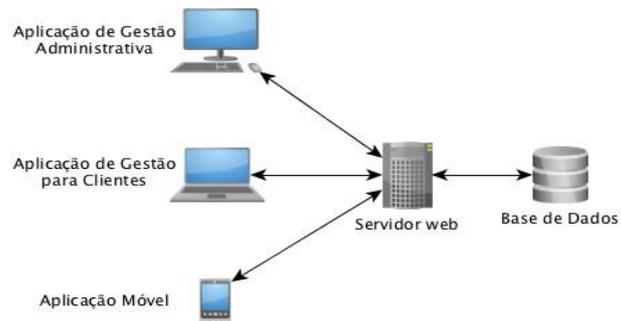


Figura 1: Arquitectura simplificada do sistema

Aplicações

O objectivo deste estágio passa pela criação de um produto de divulgação de informação sobre o espaço envolvente ao qual um visitante (utilizador) se encontra. Desta forma, é necessário disponibilizar essa informação através da aplicação móvel, sendo essa informação introduzida previamente numa das aplicações de gestão.

A aplicação de gestão administrativa tem como intuito administrar toda a plataforma. É possível fazer uma gestão de utilizadores passando pela criação, edição ou remoção de acessos à aplicação de gestão para clientes. Esta aplicação tem a capacidade da criação de vias de divulgação, sendo as mesmas uma forma prática de fazer chegar informação aos utilizadores da aplicação móvel. Esta aplicação tem ainda a particularidade de visualização de relatórios sobre a plataforma.

A aplicação de gestão para clientes tem como objectivo a gestão (introdução, edição e/ou remoção) de informação a ser visualizada na aplicação móvel. Para tal fornece a possibilidade de fazer pedidos, à aplicação de gestão administrativa, para criar uma via de divulgação para a aplicação móvel. Esta via de comunicação permite aos utilizadores, desta aplicação, introduzirem informações sobre um local colmatando-a com um mapa do local.

A aplicação móvel serve para consulta de informações sobre um ou mais locais, desde que previamente introduzidos numa das aplicações de gestão. O utilizador desta aplicação, consegue ver que locais à sua volta têm informações disponíveis sendo auxiliados com um mapa. É possível ainda tecer comentários sobre o local conseguindo desta forma dar *feedback* a outros tantos utilizadores que possam por ali passar.

1.4. Entidade acolhedora

A Enso Origins [1] é uma empresa vocacionada para o desenho e implementação de soluções de software, com especial enfoque em tecnologias móveis e plataformas web.

A Enso nasce de um pequeno grupo de pessoas: Eng. Carlos Ramos, Nuno Fidalgo e Diogo Bhovan, que juntando três áreas distintas (sistemas, software e suporte) encontraram um projecto comum e fundaram esta empresa. Perante a necessidade de criar soluções de software, que abordassem e solucionassem problemas reais com que lidavam em cada organização por onde passaram, criaram a Enso, que apesar de ainda jovem, conta com um grupo de trabalho com uma vasta experiência, conseguindo assim juntar valências bastante disparemas mas consolidadas que permitem ajustar cada solução a cada projecto que é apresentado.

Alguns dos projectos que foram elaborados até ao momento, encontram-se na área de controlo de entradas em locais públicos, sistemas de gestão de processos e aplicações móveis na área de transportes e mobilidade.

1.5. Estrutura do documento

Este documento, tem a seguinte estrutura:

- O capítulo 2 aborda o planeamento inicialmente definido para o estágio e os respectivos desvios ao mesmo;
- No capítulo 3 encontra-se o estado da arte onde é feito um estudo das aplicações concorrentes à desenvolvida no âmbito do estágio, bem como um estudo das tecnologias usadas;
- No capítulo 4 encontra-se a especificação do sistema, onde é possível encontrar uma descrição da plataforma, o levantamento de requisitos, diagrama de casos de uso, *mockups* e o respectivo diagrama de *flow*, bem como diagramas (diagrama físico do sistema, diagramas de estados e o diagrama Entidade-Relação) que definem a arquitectura da plataforma;
- No capítulo 5 apresenta-se em detalhe as decisões de implementação referentes às plataformas desenvolvidas;
- No capítulo 6 são abordados os diversos testes efectuados às plataformas, assim como é feita uma validação da implementação dos requisitos;
- Finalmente, no capítulo 7 encontram-se as conclusões referentes ao trabalho realizado e é feita uma reflexão sobre o trabalho futuro para optimização das aplicações.

Capítulo 2 - Plano de trabalho

Deste capítulo consta o planeamento inicial projectado, bem como todas as alterações/desvios ao longo de todo o ano.

2.1. Plano inicial

Fase 00 – Planeamento do Projecto

Para a organização de todo o estágio, efectuou-se um planeamento separando o mesmo por fases, estimando uma duração para cada fase. Assim, nas tabelas 1 e 2 são apresentados os planeamentos para o primeiro e segundo semestre, respectivamente, podendo no anexo D ser consultados os diagramas de *Gantt* com o planeamento inicial (figura 79), o com as devidas alterações relativas ao primeiro semestre (figura 80) e o final (figura 81).

Na reunião inicial com o orientador da Enso, Eng. Carlos Ramos, foi acordado que semanalmente haveria uma reunião onde iria ser mostrado o que tinha sido feito e o que se iria fazer na semana seguinte. Dessa reunião seria sempre elaborada uma acta para posteriormente ser enviada para o orientador do DEI (Dr. Paulo Simões), de modo a haver sempre um acompanhamento do estágio.

Existiram também reuniões com uma equipa externa da Enso, equipa essa residente em Moçambique, que apoiou na especificação dos requisitos e seguiu todo o processo da definição da arquitectura, de forma a que as componentes desenvolvidas possam vir a ser incorporadas num produto que a Enso se encontra a desenvolver e cujo público alvo actual é o mercado Português e Moçambicano.

Tarefa	Duração Estimada
Fase 00 – Plano do Projecto	20 horas
Fase 01 – Levantamento de Requisitos	60 horas
Fase 02 – Casos de Uso	40 horas
Fase 03 – <i>Mockups</i>	40 horas
Fase 04 – Arquitectura	40 horas
Fase 05 – Estudo e Escolha de Tecnologias	100 horas
Fase 06 – Relatório Intermédio	80 horas

Tabela 1: Planeamento Primeiro Semestre

Tarefa	Duração Estimada
Fase 07 – Desenvolvimento	560 horas
Fase 08 – Testes	160 horas
Fase 09 – Performance e Escalabilidade	80 horas
Relatório Final	120 horas

Tabela 2: Planeamento Segundo Semestre

Fase 01 – Levantamento de Requisitos

A fase de levantamento de requisitos é bastante importante, visto levar a perceber com toda a exactidão os requisitos funcionais e não funcionais que o sistema deve suportar, bem como perceber quais os requisitos fundamentais de implementar (*musts*), os que devem ser implementados (*should*) e os que são de valor ter no projecto (*nice to have*).

Fase 02 – Casos de Uso

Esta fase é de fundamental importância sendo que, através dos casos de uso, por um lado fica-se a saber quais as acções suportadas por cada actor envolvido em cada plataforma e por outro lado servem para dar uma melhor noção do que é esperado como produto final.

Fase 03 – Mockups

A fase de *mockups* serve para mostrar uma visualização de um protótipo do produto final com todas as funcionalidades requeridas no levantamento de requisitos.

Fase 04 – Arquitectura

Esta fase contempla toda a arquitectura do sistema, onde foi possível perceber através da criação de vários diagramas (físico, estados, ER), como cada componente do sistema interage com as restantes, e consequentemente perceber quais as tecnologias necessárias para a implementação da solução final deste estágio.

Fase 05 – Estudo e Escolha de Tecnologias

O estudo das tecnologias, ajudou a compreender o que existe de semelhante no mercado e de que forma é possível criar um produto distinto dos demais. Após o término desta fase, consegue-se concluir quais as ferramentas necessárias para a criação do produto em questão.

Fase 06 – Relatório Intermédio

O relatório intermédio, foi ao longo do primeiro semestre no decorrer de cada fase, minimizando o peso do mesmo no final do semestre. Ainda assim, as horas atribuídas a esta tarefa foram utilizadas para revisão de todos os conteúdos produzidos até esta fase, bem como adequar os mesmos ao perfil técnico e à linguagem requerida para este relatório.

Fase 07 – Desenvolvimento

A metodologia de trabalho seguida é baseada em *Scrum*. Esta escolha deve-se ao facto de ser uma metodologia ágil (conjunto de estratégias para gestão de projectos incremental e iterativa com ciclos rápidos de desenvolvimento), permitindo fazer uma lista das funcionalidades a serem implementadas (*product backlog*). Antes de se dar início a cada ciclo de desenvolvimento (*sprint*), efectua-se uma reunião (*sprint planning*) para se saber o que vai ser implementado no *sprint*. No final de cada *sprint* existe uma nova reunião para fazer uma revisão do que foi desenvolvido (*sprint review*) e planeia-se o *sprint* seguinte.

Fase 08 – Testes

Durante o desenvolvimento serão feitos diversos tipos de testes, de modo a perceber se o resultado do processo de desenvolvimento está coerente e sem falhas. Sempre que um módulo terminar o seu ciclo de desenvolvimento, serão feitos testes de integração para se encontrar falhas na integração dos componentes no sistema. Proceder-se-á à realização de testes de sistema, tendo estes o objectivo de perceber se existem ou não falhas, do ponto de vista do utilizador. No final da fase de desenvolvimento serão realizados testes de aceitação, servindo estes para fazer uma ultima validação do desenvolvimento e dos requisitos definidos. Para colmatar serão feitos testes de usabilidade, por um grupo diverso de utilizadores, tendo por objectivo perceber se o sistema é facilmente compreendido e usável por um utilizador.

Fase 09 – Performance e Escalabilidade

Com o intuito de perceber qual a capacidade de reacção da aplicação face à quantidade de conexões simultâneas e avaliar o desempenho, serão feitos testes de performance e de escalabilidade. A especificação dos testes de escalabilidade podem ser visualizados na secção 6.7.

Fase 10 – Relatório Final

À semelhança do primeiro semestre, o relatório será escrito no decorrer de cada etapa, com um conjunto de revisões finais.

2.2. Desvios ao plano

2.2.1 – Primeiro Semestre

Visto que o estágio começou uma semana depois do previsto, devido a questões burocráticas, isso levou a um ligeiro atraso nas tarefas. Assim, foi aumentada a carga horária da primeira semana de forma a recuperar algum tempo e manter o planeamento global do projecto.

Foi ainda necessário fazer uma alteração no planeamento, passando a fase do estudo da tecnologia para uma fase anterior/coincidente com a fase do desenho da arquitectura, de modo a ser possível completar com mais informação o diagrama físico bem como conseguir fundamentar o mesmo com dados mais concretos.

Como referido anteriormente, o diagrama com as devidas alterações encontra-se na secção D.2 dos anexos na figura 80.

2.2.1 – Segundo Semestre

Devido a não estar implementado tudo o que se pretendia de forma a conseguir entregar o estágio até ao final de Junho, foi decidido adiar a defesa para Setembro. A alteração resultante levou a que o mês de Junho fosse ainda dedicado à implementação, o mês de Julho para testes e o mês de Agosto para escrita do relatório e preparação da apresentação final.

O diagrama com as devidas alterações encontra-se na secção D.3 dos anexos na figura 81.

Capítulo 3 - Estado da arte

Na primeira secção deste capítulo, é apresentada uma comparação entre algumas aplicações concorrentes, tais como o *Google Places*, o *Foursquare*, o *Waze*, o *e-Museum* e o *TripAdvisor*. Esta comparação surge com o intuito de ser possível analisar em que é que a aplicação desenvolvida se difere das demais.

Para além da comparação entre aplicações concorrentes, são também apresentadas, na segunda secção deste capítulo, o estudo e a escolha da tecnologia com base nas funcionalidades que a plataforma necessita para o seu bom funcionamento. A partir deste estudo é também possível perceber o motivo de terem sido escolhidas entre as demais existentes no mercado.

3.1. Aplicações concorrentes

Nesta secção serão apresentadas algumas aplicações que serão concorrentes à que será desenvolvida no âmbito deste estágio. Em cada sub-secção, são abordadas cada uma das aplicações individualmente, sendo que no final é feita uma comparação entre cada uma e a aplicação a desenvolver neste estágio.

O autor decidiu experimentar todas as aplicações concorrentes referidas, não só para as conhecer melhor individualmente, mas também para perceber quais os pontos diferenciadores existentes entre cada uma em relação à plataforma proposta para o estágio.

De referir que as escolhidas não são as únicas aplicações concorrentes existentes, e que as eleitas para comparação são principais no mercado e englobam diversas tecnologias interessantes. Apesar de não ser uma aplicação, mas sim uma plataforma para a qual são desenvolvidas aplicações, uma das que não foi mencionada como concorrente, é o *Google Glass* [2]. Não foi visto como necessidade mencionar esta concorrente, devido a ter preços nada convidativos [3] e ainda só foi lançado nos Estados Unidos [4].

3.1.1. Google places

O objectivo do *Google Places* [5] serve para disponibilizar informações de uma empresa, na pesquisa do *Google*, no *Google Maps*, no *Google+* e em dispositivos móveis. Esta aplicação é multiplataforma, estando disponível para *browsers*, dispositivos android e iOS. É permitido aos clientes comentarem e fazerem avaliações sobre o a empresa, permitindo uma aproximação entre a empresa e os clientes.

“O Google Places para Empresas permite-lhe aceder a ferramentas gratuitas que ajudam a colocar a sua empresa online, a ser encontrada na Pesquisa Google e no Google Maps e a aproximar-se dos clientes. É a forma mais fácil de gerir a sua empresa no Google e é gratuito.” [6]

Tanto o produto do âmbito do estágio, como o *Google Places* têm um objectivo em comum, que passa com o tentar aproximar as empresas dos clientes, disponibilizando conteúdos com informação sobre as empresas. Porém, o *Google Places* é um bom produto, embora tenha objectivos distintos dos propostos para a plataforma de estágio como já foi mencionado neste documento anteriormente. Desta forma, vê-se uma clara diferença na maneira como são disponibilizados os conteúdos, sendo que no *Google Places* fornecem dados sobre as empresas facilitando a pesquisa online destas, ao invés da plataforma do âmbito do estágio que servirá para promover o espaço disponibilizando informações à medida que se deseje e

do público-alvo que se queira englobar, tendo oportunidade de mostrar a qualquer utilizador do que trata a empresa e onde se situa. Para o utilizador que se encontra no interior do espaço/local o que será mostrador, servirá mais para promover e mostrar o que se pode encontrar, orientando-o pelo local onde se encontra.

3.1.2 – Foursquare

O objectivo do *Foursquare* [7] é permitir aos utilizadores dizerem onde estão (fazendo *check-in*), partilhando essa informação com os seus amigos. O âmbito inicial passava por ser um jogo, onde os utilizadores à medida que iam partilhando a sua localização ganhavam pontos e *badges* (insígnias). Com a evolução e com a forte adesão à escrita de comentários sobre os lugares visitados, fez com que actualmente o *Foursquare* seja quase como um guia cultural e gastronómico. Esta aplicação é multiplataforma, estando disponível para *browsers*, dispositivos android, iOS e Blackberry.

“Foursquare is a free app that helps you and your friends make the most of where you are. When you're out and about, use Foursquare to share and save the places you visit. And, when you're looking for inspiration for what to do next, we'll give you personalized recommendations and deals based on where you, your friends, and people with your tastes have been.” [8]

Apesar do sucesso do *Foursquare*, há diversos detalhes que diferem entre o mesmo e a plataforma proposta. Começando pelos objectivos que transparecem da mesma, passando pelo público alvo e perante o uso dado à tecnologia, são visíveis claras diferenças na forma como é usada a plataforma quanto ao facto de se limitar a georreferenciar objectos, não dando ênfase à verdadeira envolvência do espaço em relação ao utilizador, focando-se essencialmente nos comentários dos utilizadores não permitindo criar um canal devidamente orientado ao que o espaço comercial pretende anunciar/promover.

3.1.3 – Waze

O objectivo do *Waze* [9] é ser uma aplicação *crowdsourcing* (modelo de inteligência artificial baseado em conhecimentos colectivos e voluntários da comunidade) de mapas e cálculo de rotas, conseguindo juntar informações de trânsito dada pelos utilizadores, permitindo com isto melhorar os itinerários e fazer com que os utilizadores poupem tempo e combustível. Esta aplicação está disponível em diversas plataformas, sendo elas *browsers*, dispositivos android, iOS, Windows Phone, Blackberry, Symbian (Nokia) e Windows Mobile.

“By connecting drivers to one another, we help people create local driving communities that work together to improve the quality of everyone's daily driving. That might mean helping them avoid the frustration of sitting in traffic, cluing them in to a police trap or shaving five minutes off of their regular commute by showing them new routes they never even knew about.” [10]

Facilmente se percebe que toda a estrutura que suporta o *Waze* é completamente diferente da plataforma do âmbito do estágio, visto que para o *Waze* funcionar é necessário que quem tenha a aplicação no dispositivo móvel participe activamente reportando com o que se lhe depara, enquanto que na plataforma proposta o objectivo é diferente sendo que o essencial é cativar utilizadores, guiando-os pelos locais onde se encontram. Para além disso há algumas semelhanças como é o caso de se conseguir referenciar objectos que estejam no nosso raio de acção. No entanto, o *Waze* faz essa referenciação para objectos apenas no exterior de edifícios, enquanto que a plataforma proposta faz a referenciação tanto no exterior como no interior de edifícios.

3.1.4 – Aplicação e-Museum

A aplicação e-Museum [11] é constituída por dois produtos, a aplicação de gestão de conteúdos (*E-museum Producer*) e a aplicação móvel para dispositivos Android (*E-Museum Mobile Assistant Guide*). Esta aplicação móvel foi lançada para o Mosteiro de Santa Clara a Velha (Coimbra, Portugal) [12] tendo como intuito permitir a qualquer utilizador do museu ter um guia virtual.

Esta aplicação está disponível em vários idiomas e oferece ao utilizador uma “visita guiada” ao museu, onde mostra diversa informação associada. Mostra todas as colecções ou então, caso o visitante deseje, é possível fazer uma pesquisa onde se escreve um número que representa a peça ou por voz onde se dita o número. Quando é apresentada a informação sobre uma peça, essa informação pode ser lida ou narrada.

Apesar desta plataforma parecer semelhante à do âmbito deste estágio, estas diferem bastante, tanto a nível do público alvo, como da tecnologia que será usada na aplicação proposta. O intuito com a aplicação móvel proposta, num cenário semelhante como é o caso de um museu, é o facto dos utilizadores se sentirem verdadeiramente apoiados pela aplicação e não servir apenas para dar informações. Esse apoio, advém de ser possível mostrar um mapa do local, conseguido situar o utilizador, guiando-o verdadeiramente pelo museu.

3.1.5 – TripAdvisor

O objectivo do *TripAdvisor* [13] é fornecer aos utilizadores informações sobre locais. Para além de agregarem diversas informações sobre uma dada cidade (hotéis, restaurantes, atracções, voos) com comentários/opiniões de outros utilizadores e classificações dos locais, ainda fornecem a possibilidade de, por exemplo, reservar um hotel. Com isto, para além de um guia gastronómico e cultural, tornam mais simples e cómodo para os utilizadores finais a possibilidade de planearem uma viagem. Esta aplicação é suportada em diversas plataformas, sendo elas *browsers*, dispositivos android, iOS, Windows Phone 7 e 8, Nokia touchscreen (que corram o *software* Symbian Series 60 v5) e Kindle Fire de 2ª e 3ª geração.

“TripAdvisor® is the world's largest travel site, enabling travelers to plan and have the perfect trip. TripAdvisor offers trusted advice from real travelers and a wide variety of travel choices and planning features with seamless links to booking tools. (...) TripAdvisor also includes TripAdvisor for Business, a dedicated division that provides the tourism industry access to millions of monthly TripAdvisor visitor” [14].

A principal diferença entre o *TripAdvisor*, e apesar das múltiplas vantagens do *TripAdvisor*, este não consegue fazer orientação *indoor*, nem identificar objectos *indoor*.

3.1.6 – Análise comparativa

Como características/objectivos semelhantes considera-se a aproximação entre a empresa e os clientes/utilizadores finais, a disponibilização de conteúdos georreferenciados e o facto de conseguir auxiliar os utilizadores no seu dia-a-dia dependendo do propósito de cada aplicação. Na tabela 3, encontra-se uma análise comparativa entre as diferentes aplicações, de notar que os pontos apresentados na tabela são pontos diferenciadores.

	Público-alvo	Multiplataforma	Orientação indoor	Personalizável	Temática	Identificação de objectos indoor
Google Places	Empresas	Sim	Não	Não	Não	Não
Foursquare	Comunidades online	Sim	Não	Não	Não	Não
Waze	Condutores	Sim	Não	Não	Sim	Não
E-Museum	Visitantes de museus	Não	Não	Não	Sim	Sim
TripAdvisor	Turistas	Sim	Não	Não	Sim	Não
Plataforma proposta	Utilizador temático	Sim	Sim	Sim	Sim	Sim

Tabela 3: Comparação entre as aplicações concorrentes

Os pontos diferenciadores considerados são público-alvo, multiplataforma, orientação *indoor*, personalizável, temática e identificação de objectos *indoor*.

Como público-alvo do estágio, considera-se utilizador temático, visto depender do contexto envolvente de onde a aplicação estiver a ser usada, assim pode-se ter uma temática para diferentes tipos de visitantes sendo exemplos disso, os visitantes de museus, de jardins, turistas numa cidade ou até mesmo público em concertos ou festivais.

3.1.7 – Em suma

Todas as plataformas descritas nas sub-seções anteriores, são consideradas concorrentes não só por terem os mesmos objectivos que a plataforma proposta, mas também por serem bem sucedidos quanto ao propósito que servem. Assim, acima de tudo utilizar-se-á o conhecimento que se obteve da análise de cada uma destas aplicações, utilizando o mesmo para apoiar e moldar as soluções tecnológicas a usar.

Quanto às tecnologias usadas serão abordadas, com uma respectiva análise, na escolha e estudo das tecnologias que se encontra neste documento na secção 2 deste mesmo capítulo.

3.2. Escolha e estudo das tecnologias de suporte

Nesta secção serão apresentadas as ferramentas e tecnologias adoptadas para a execução deste projecto, de modo a garantir que são as que melhor apoiam a solução para a implementação desejada.

Assim, são apresentadas as escolhas para a infraestrutura de suporte, servidor web, motor de base de dados, linguagens de desenvolvimento, *frameworks*, técnicas de georreferenciação interior e exterior e qual a plataforma móvel para o qual será desenvolvida o projecto no âmbito do estágio.

3.2.1. Infraestrutura de Suporte

Para a infraestrutura de suporte, foi tido em consideração os requisitos para o servidor, nomeadamente a capacidade de processamento, a memória necessária, a largura de banda necessária e o espaço de armazenamento. De modo evitar problemas com os requisitos referenciados, foi escolhida uma implementação em *cloud*. Para tal, já havia na empresa, uma VPS (Virtual Private Server) alojada numa entidade externa, tendo como sistema operativo o CentOS 6.5 [15] com 50Gb de espaço em disco, 2 processadores virtuais (*cores*) e 2Gb de memória RAM, que será usada para o efeito.

Este modelo torna-se adequado para o desenvolvimento desta plataforma, tendo como vantagens o facto de se poder crescer “elasticamente” à medida das necessidades e a possibilidade de utilizar softwares sem que estejam instalados no computador. Ainda há que ter em conta as vantagens de consumo energético, refrigeração e espaço físico.

A nível de desenvolvimento será necessário garantir um acesso constante à plataforma em causa para garantir que não existem perturbações significativas no processo de desenvolvimento e *deployment*.

Para servidor *web*, e como já era usado na empresa, será usado o Apache [16]. Este, representa o mais bem sucedido servidor *Open Source* conseguindo assim ser o mais usado como podemos ver pela figura 2.

Contudo, esta escolha parece acertada visto ser eficiente, flexível devido à sua estrutura modular, fácil de administrar, ser de código aberto e oferecer uma boa portabilidade, podendo ser instalado e operado em diferentes plataformas. Há um bom suporte por parte da comunidade *Open Source* e a estrutura interna da empresa já tem conhecimento a nível de instalação e configuração deste tipo de servidores.

Em suma, para a infraestrutura será usado um servidor com o CentOS e o Apache.

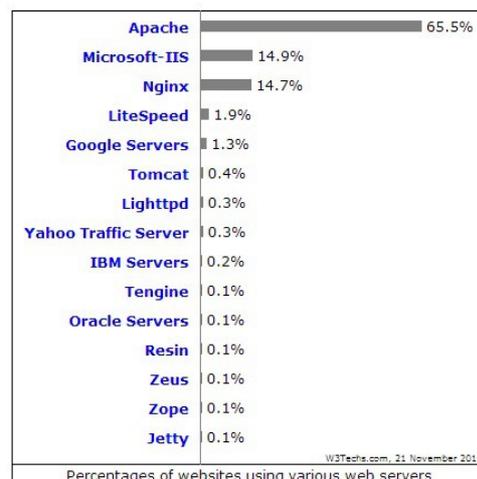


Figura 2: Utilização de *web servers* [17]

3.2.2. Base de dados

Num projecto desta dimensão, é útil recorrer-se a um Sistema de Gestão de Bases de Dados (SGBD), visto poderem armazenar grandes quantidades de informação, na facilidade de acesso, adição e/ou edição da informação guardada. Para além disso, esta pode ser acedida por mais do que um utilizador ao mesmo tempo.

Assim, foi efectuada uma pesquisa de modo a conseguir-se perceber quais os mais usados devido à sua popularidade. A medição da popularidade [18] é baseada em vários parâmetros, sendo eles:

- o número de sites que mencionam o SGBD em consultas efectuadas nos motores de busca do Google [19] e do Bing [20];
- o interesse geral do sistema, sendo usado a frequência das procuras no Google Trends [21];
- a frequência de discussões técnicas sobre o SGBD (número de questões e número de utilizadores interessados, sendo estes vistos através do Stack Overflow [22] e do DBA Stack Exchange [23];
- o número de oferta de empregos em que o sistema é mencionado, sendo medido nos principais motores de procura de emprego, o Indeed [24] e o Simply Hired [25];
- o número de perfis de redes profissionais em que o sistema é mencionado, sendo usado o LinkedIn [26];
- a relevância nas redes sociais, contando o número de *tweets* no Twitter [27].

De referir ainda, que este *ranking* não mede o número de instalações/utilização dos SGBD.

Os resultados apresentados na tabela 4 foram filtrados, de modo a ser possível a escolha e análise dos SGBD de interesse para a plataforma desenvolvida. Assim, e como modo de exclusão, não serão analisados os que tenham licença comercial (Oracle, Microsoft SQL Server, DB2, Microsoft Access, Sybase, Teradata), visto não estar abrangido no actual projecto investimento para a aquisição de licenciamento a nível de base de dados. Ainda foi excluído o MongoDB, tendo em conta que se trata de um SGBD para armazenar documentos.

Rank	DBMS	Database Model	Score
1	Oracle	Relational DBMS	1617,19
2	MySQL	Relational DBMS	1254,27
3	Microsoft SQL Server	Relational DBMS	1234,46
4	PostgreSQL	Relational DBMS	190,83
5	DB2	Relational DBMS	165,9
6	MongoDB	Document store	161,87
7	Microsoft Access	Relational DBMS	141,6
8	SQLite	Relational DBMS	78,78
9	Sybase	Relational DBMS	77,75
10	Teradata	Relational DBMS	60,12

Tabela 4: Ranking de Sistemas de Gestão de Bases de Dados [28]

Assim, a análise recaiu sobre o MySQL, PostgreSQL e SQLite, sistemas que respeitam o conceito ACID (Atomicidade, Consistência, Isolamento, Durabilidade), ou seja, garantem que as transações são processadas de forma fiável.

SQLite

O SQLite [29] não necessita de configuração, nem de nenhum instalador, sendo esta a sua vantagem, no entanto é considerado *serverless* tendo em conta que não necessita de usar qualquer tipo de comunicação entre processos (normalmente TCP/IP) para enviar solicitações. Assim, lê e escreve directamente no disco. O âmbito do SQLite é orientado para plataformas que não requerem a integração com outros módulos aplicativos externos e dado a sua natureza *serverless* dificulta todo e qualquer tipo de acesso à base de dados que não seja feito a partir do local original onde o motor se encontra.

MySQL

Dos motores considerados, o MySQL [30] é o SGBD mais popular da actualidade segundo o db-engines [28]. Este sucesso deve-se essencialmente à sua portabilidade, desempenho, estabilidade e compatibilidade com diversas linguagens de programação. Para além de ter uma versão gratuita (*MySQL Community Edition*), requer poucos recursos e a sua utilização é fácil e intuitiva.

PostgreSQL

O PostgreSQL [31, 32] é um servidor de base de dados estando disponível para várias plataformas. Um claro benefício deste motor de base de dados é o facto de ser *Open Source* e o seu licenciamento ser baseado em licenciamento BSD [33] ou MIT [34], podendo ser instalado, configurado e distribuído sem qualquer tipo de custos. Este sistema é conhecido por se manter por largos períodos requerendo pouca ou até mesmo nenhuma manutenção. Ainda se destaca pela sua arquitectura cliente-servidor, é altamente concorrente (escritas e leituras não bloqueantes) e robusto. O motor ainda tem uma excelente escalabilidade, performance e segurança.

Conclusão

Perante o que foi apresentado, é facilmente perceptível que não há uma clara diferença entre o PostgreSQL e o MySQL. A diferença entre estes dois motores prende-se com o facto do PostgreSQL ser optimizado para aplicações complexas, ao invés do MySQL que se foca na agilidade. Contudo, actualmente estes dois SGBD têm ambos aspectos muito bons, tais como desempenho, recursos, documentação e suporte.

Feita esta análise, foi dada preferência ao PostgreSQL visto impor menos restrições a nível de licenciamento do produto final desenvolvido e ser o SGBD usado na empresa, tornando-se vantajoso para o projecto.

3.2.3. Segurança

Tendo em conta a quantidade de ataques diários [35] que existem, é importante ter especial atenção à segurança da plataforma. Deste modo, deu-se especial enfoque à questão da confidencialidade, autenticação e autorização para se conseguir garantir a integridade das comunicações entre o servidor e as diferentes aplicações cliente.

Confidencialidade

A confidencialidade é trabalhada recorrendo a um conjunto de técnicas, baseadas num sistema de *tokens* desenvolvido internamente, suportado no protocolo *Secure Socket Layer* (SSL). De momento está a ser usado um certificado *self-signed*, no entanto quando a plataforma entrar em produção o mesmo será substituído por um certificado adequado, emitido por uma *trusted CA*, sendo que de um ponto de vista técnico nada irá alterar à solução desenvolvida.

Autenticação

De modo a garantir um método de autenticação de um utilizador ao *provider*, recorreu-se a uma livreria de *tokens* (ou acessos) desenvolvida por uma equipa interna. Esta livreria garante de forma fácil e segura a *tokens* limitando temporalmente os mesmos. O processo de *tokens* negocia uma chave para utilização numa sessão criptográfica, tendo ainda a capacidade de transferir um “cartão identificativo” do *token* (designado por NetworkToken). O processo utilizado para o efeito, é dividido em duas fases, sendo a fase 1 o pedido de um *token* de não autenticação, figura 3, e a fase 2 a utilização do *token* por parte de um utilizador, figura 4.

Processo de pedido de um token:

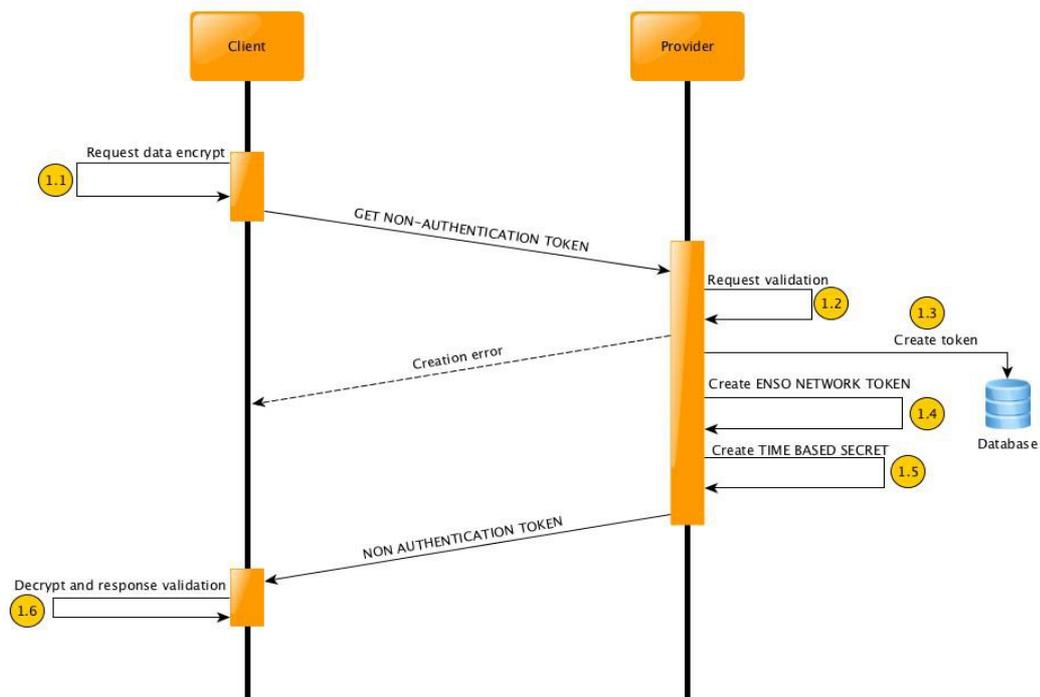


Figura 3: Processo de pedido de um token

- 1.1 Um determinado utilizador requisita ao *provider* a criação de um *token* inicial;
- 1.2 O *provider* valida o pedido recebido e ignora o mesmo caso exista qualquer problema criptográfico;
- 1.3 O *provider* cria um *token* inicial no modelo de dados;
- 1.4 É criada uma mensagem, por parte do *provider* de modo a poder enviá-la ao utilizador, com um NetworkToken;
- 1.5 Visto que o *token* já foi validado, é criado um *secret* pseudo-aleatório baseado num algoritmo temporal que servirá de chave criptográfica para este *token* a partir da conclusão deste processo;
- 1.6 Assim que o utilizador recebe a mensagem enviada pelo *provider* deverá descriptar a mesma validando a resposta. Caso a mesma seja validada com sucesso, o NetworkToken recebido é armazenado, sendo que o mesmo contém já a nova chave a utilizar no processo de encriptação a partir deste instante.

Processo de comunicação recorrendo ao token:

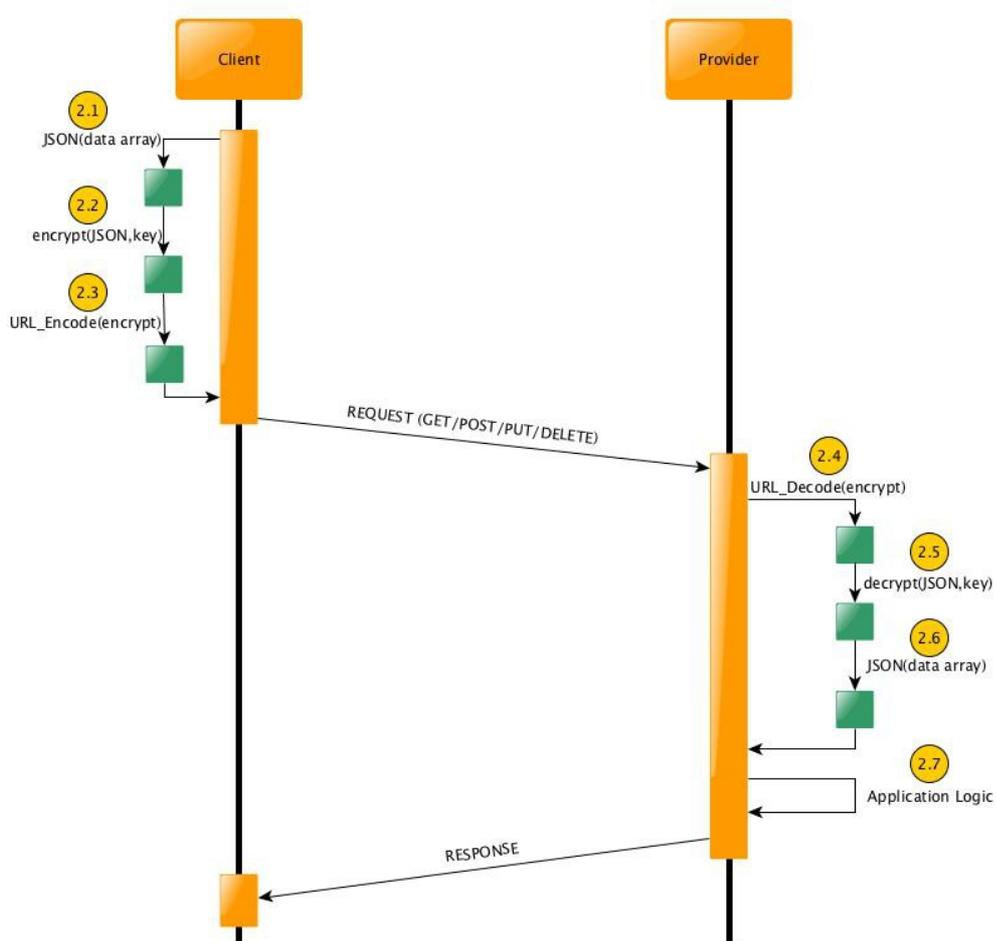


Figura 4: Processo de comunicação recorrendo a tokens

- 2.1 Este processo inicia-se convertendo os dados de entrada (todo o *input* do processo HTTP) recorrendo à codificação JSON;
- 2.2 O resultado da fase 2.1 é agora encriptado recorrendo à chave definida para o processo de encriptação (*secret* proveniente da fase 1.5);
- 2.3 É feita a codificação dos dados, de modo a poderem ser transferidos na rede;
- 2.4 É feita a descodificação dos dados transferidos na rede;
- 2.5 O *provider* através da *secret* previamente armazenada para este *token*, descripta os dados recebidos;
- 2.6 Os respectivos dados são descodificados de acordo com o formato JSON e entregues ao respectivo processo implementado;
- 2.7 É implementada a lógica pretendida.

Autorização

Por fim, mas não menos importante, para o processo de autorização recorreu-se à utilização da livreria *Role-Based Access Control* (RBAC) [36]. Este sistema baseia-se em três premissas:

- Utilizadores;
- *Roles*;
- Acções.

Os utilizadores estão integrados com o sistema de utilizadores da plataforma desenvolvida, sendo que lhes é atribuído automaticamente um *role* de acordo com o tipo de utilizador que é. Os *roles* estão relacionados com as acções, sendo que as mesmas representam acções directas sobre a plataforma. Por exemplo, caso o utilizador Zé tenha privilégios para adicionar novos utilizadores, teríamos:

- Utilizador: Zé;
- *Role* atribuída ao utilizador Zé: administrador;
- Acção: CAN CREATE USER atribuída à *role* administrador.

Desta forma, ao implementar a funcionalidade de implementar utilizadores, a mesma filtra os acessos recorrendo à acção CAN CREATE USER.

3.2.4. Linguagens de desenvolvimento

Server Side

A nível de linguagens *server side* temos diversas alternativas como podemos ver na figura 5, independentemente da popularidade do PHP [36, 37, 38], optou-se por esta linguagem devido à sua portabilidade suportando diversos sistemas operativos (Windows, Linux, FreeBSD, Unix, etc) e à facilidade de acesso à base de dados.

Adicionalmente a estas vantagens, existe já conhecimento interno na empresa a nível de desenvolvimento para a linguagem PHP, não justificando assim o investimento necessário à introdução de linguagens alternativas no processo de desenvolvimento.

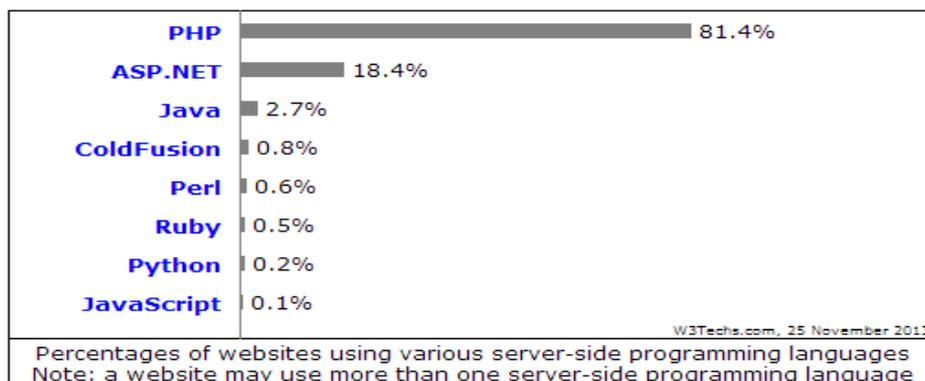


Figura 5: Programação do lado do servidor[40]

Client Side

A componente de cliente será desenvolvida baseando a mesma em HTML5 [41] e CSS3 [42], recorrendo a JS[43] para a estruturação lógica da aplicação.

Acesso Server-Client

Roy T. Fielding [44] definiu na sua tese de doutoramento [45] o termo REST (*Representational State Transfer*), para além disso foi um dos principais a contribuir para o desenvolvimento de muitos protocolos *web* essenciais, incluindo o HTTP e URIs. Conforme descrito na sua dissertação o REST é um estilo de arquitectura para a construção de aplicações que geralmente correm sobre HTTP, sendo a transferência de dados feita tipicamente recorrendo a JSON ou XML, como pode ser visto na figura 6.

Desta forma irá ser utilizado este modelo para fornecer acesso ao modelo de dados por parte das aplicações cliente. O formato JSON apesar de ser mais difícil de ser lido por um humano quando comparado com outras alternativas (exemplo: XML), optimiza o consumo de largura de banda e toda a representação de dados levando a que haja um menor *overhead* computacional.

Trazendo esta metodologia para este projecto ganha-se algumas vantagens estratégicas de onde destaca:

- comunicação sem estados garantindo uma boa capacidade de escalabilidade;
- altos níveis de interoperabilidade (capacidade de um sistema interagir e comunicar com outro);
- facilidade de utilização e de implementação, desenvolvendo ou recorrendo a bibliotecas já existentes.

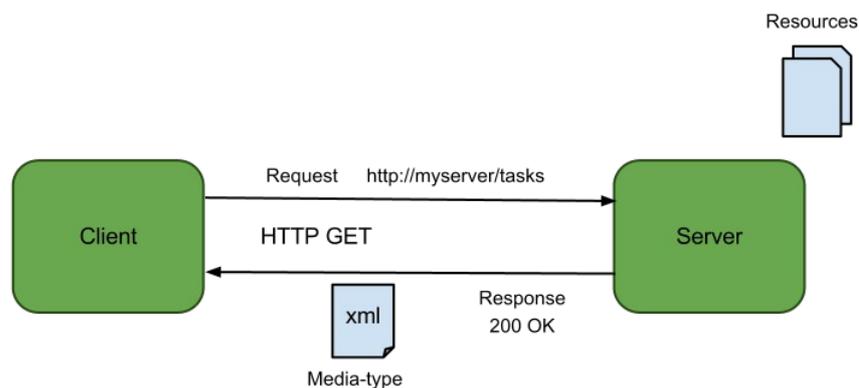


Figura 6: Exemplo de um pedido REST [46]

3.2.5. Frameworks/APIs

Implementação Server Side

A *framework* de desenvolvimento para a plataforma *web* escolhida foi o *Codeigniter* [47]. Esta escolha deveu-se ao facto de ser *Open Source* e de já haver conhecimento interno na empresa sobre esta *framework*, levando a uma curva de aprendizagem de grau menos acentuado.

Esta *framework* desenvolvida por Rick Ellis (CEO da EllisLab Inc [48]) assenta sobre uma arquitectura MVC (*Model-View-Controller*) e aposta principalmente na rapidez, em ser leve e simples. Devido a não requerer muitos recursos torna-se leve, visto carregar as bibliotecas dinamicamente apenas quando são necessárias, levando a que a base do sistema seja muito rápida.

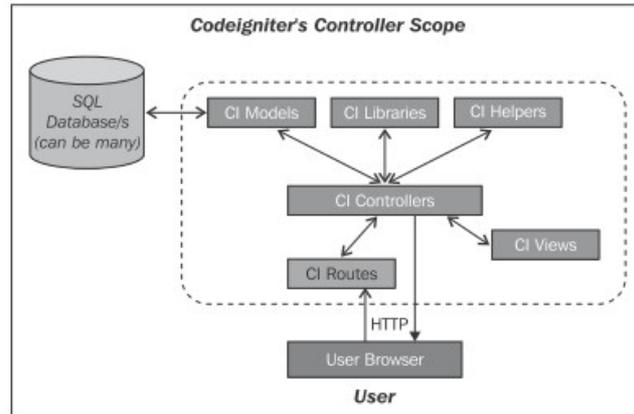


Figura 7: Fluxo de dados MVC com codeigniter[50]

“A arquitectura MVC é uma solução para desacoplar as camadas de lógica e regras de negócio da camada de apresentação.” [49]

- O modelo é composto por funções que servem para interagir com a base de dados.
- A vista contém por norma uma página *web*, servindo para apresentar a informação ao utilizador final.
- O controlador é o responsável pela comunicação entre o *Model*, a *View* e todos os pedidos HTTP.

Na figura 7 pode-se ver como funciona o fluxo de dados da arquitectura MVC que o *Codeigniter* faz uso.

Mapas

É necessário recorrer a *frameworks* de integração para visualização do mapa nas diferentes plataformas, marcar pontos e definir polígonos nas aplicações *web* e aplicação móvel.

Para a visualização de mapas nas diferentes plataformas, foi tido em conta a API do *Google Maps* [51] e a do *Mapbox* [52].

Ambas as APIs são boas e estão bem documentadas, sendo a grande desvantagem do *Google Maps* perante o *MapBox* prender-se ao facto de ser pago. Mesmo assim, o *Google Maps*, disponibiliza a possibilidade dos programadores poderem usar a sua API com a restrição de apenas poderem ser feitos 2000000 pedidos por dia.

Deu-se preferência à API do *Google Maps*, visto que a versão gratuita é suficiente e contém diversas funcionalidades interessantes, como por exemplo, o *auto complete* para se poder pesquisar escrevendo o nome de uma rua, cidade ou país.

Integração Mapas com aplicação WEB

Para a marcação de pontos e desenho de polígonos na plataforma *web*, foi tido em conta o uso do Canvas do HTML5 ou fazer-se uso da livreria *Leaflet* [53].

O Canvas, é um elemento que surgiu com o HTML5, sendo usado principalmente para animações ou na criação da possibilidade de desenhar numa página *web*, no entanto após uma análise mais detalhada e comparativa com a livreria *Leaflet* a mesma revelou algumas fragilidades como se pode ver na tabela 5.

	Compatibilidade browsers	Georreferenciação da planta	Layers
Canvas	Firefox 30+, Chrome 27+, Safari 5+, Opera 23+, IE (superior a 9.0) [54]	Necessita ser manualmente implementado	Não suportado
Leaflet	Firefox, Chrome, Safari 5+, Opera 12+, IE (superior a 7) [55]	Suportado Nativamente	Suportado Nativamente

Tabela 5: Comparação entre o Canvas e a livreria Leaflet

Optámos pelo *Leaflet* de acordo com a tabela em cima, e também devido a ser uma livreria *open source* suportada em JavaScript que permite interagir facilmente com mapas, sendo até utilizada por organizações reconhecidas como o *Flickr* ou o *Foursquare*.

A documentação disponível é bastante completa e permite facilmente compreender como carregar mapas e adicionar *features* a esse mapa (ex: marcadores ou polígonos), permitindo assim marcar os itens no mapa ou desenhar áreas de abrangência e regiões na nossa plataforma *web*, respectivamente.

Facebook

Em relação à autenticação na plataforma móvel através de redes sociais foi decidido nos requisitos que seria necessário integrar a aplicação móvel tanto com o *Facebook* [56], como com o *Google* [57]. De notar que apenas o *Facebook* será tido em conta no âmbito do estágio.

QR-Codes

Para se conseguir fazer leituras de *QR-Codes* a escolha recaiu no uso do *zbar* SDK para android, devido à facilidade do uso da mesma.

YouTube

Ainda será usada a API do *Youtube*, para haver facilidade na integração, visualização e no carregamento de vídeos provenientes do youtube.

3.2.6. Georreferenciação

Para este projecto é fundamental conseguir-se georreferenciar os utilizadores da plataforma móvel, os canais e os itens criados pelos clientes para serem disponibilizados para consulta na plataforma móvel.

Georreferenciação de Utilizadores

Para a localização dos utilizadores da plataforma móvel no exterior de edifícios, recorre-se naturalmente ao GPS (*Global Positioning System*), A-GPS (*Assisted Global Positioning System*) e ao GSM (*Global System for Mobile Communications*), sistemas que funcionam com recurso a satélites fornecendo assim a posição geográfica do dispositivo.

Entre o GPS e o A-GPS a grande diferença existente consiste na velocidade em que um *smartphone* consegue obter posições geográficas. Para além disso, o A-GPS permite que a informação de localização seja adquirida não só de satélites, mas também de antenas de comunicação de uma operadora.

No âmbito do estágio, apenas se irá ter em conta, o uso do GPS e A-GPS para geolocalização exterior e para a geolocalização interior recorre-se ao sinal recebido por um AP, sem nenhum sistema de apoio para ajudar na precisão da localização de um utilizador.

No entanto, e para um trabalho futuro, visto não haver tempo suficiente para se conseguir implementar durante o período do estágio, há algumas técnicas que se podem usar para ajudar na precisão da localização de um dispositivo. Na localização por GSM, o sistema usado é a triangulação de antenas GSM. Esta técnica depende do número de antenas disponíveis e sendo que a precisão do mesmo oscila entre os 500 metros e os 1000 metros [58].

Para a detecção da posição dos utilizadores da plataforma móvel dentro de edifícios recorre-se a uma WLAN, sendo esta uma tecnologia de rede local que através de técnicas como a triangulação dos AP's permite saber a localização de um dispositivo móvel, visto a localização por GPS apenas ser possível em locais abertos.

Georreferenciação de Objectos

Para a localização de objectos dentro de um local as tecnologias que podem ser usadas são o NFC (*Near Field Communication*) e/ou QR-Code (*Quick Response Code*). Não há necessidade de abordar o RFID (Identificação por radio frequência), devido aos seus custos e porque não apresenta melhorias perante o NFC ou QR-code.

O NFC permite de forma segura, a troca de informações automaticamente, sem necessidade de configurações, entre dois dispositivos (*smartphones*, cartões electrónicos ou qualquer dispositivo com um chip NFC) compatíveis com esta tecnologia desde que estejam próximos um do outro.

O QR-Code é um código de barras bi-dimensional, código esse que pode conter um endereço URI (*Uniform Resource Identifier*), uma localização, texto, sendo o mesmo lido através da captura pela câmara do telemóvel.

No âmbito do estágio e para auxiliar na localização de objectos no interior de um edifício, apenas vai ser tido em conta o QR-Code, visto não ser necessário haver investimento monetário por parte da empresa (compra de chip NFC).

Georreferenciação e Armazenamento de dados

A área de abrangência e as regiões de um canal, serão georreferenciadas na plataforma *web*, sendo isto possível com o uso da API do *Google Maps*.

Para ser possível o armazenamento da georreferenciação dos objectos e da localização dos utilizadores da plataforma móvel, recorrer-se-á à extensão PostGIS [59] do PostgreSQL. Esta extensão será vantajosa, tendo em conta que permite adicionar suporte a objectos geográficos ao PostgreSQL, seguindo a especificação do OGC (*Open Geospatial Consortium*) [60]. Este suporte contempla funções espaciais tais como a distância, área, união, intersecção permitindo assim ver se um ponto (utilizador com dispositivo móvel) se encontra numa dada região ou a que distância este se encontra da área de abrangência de um canal. Para além disso permite armazenar dados na base de dados, quer sejam pontos, polígonos ou outras formas geométricas [61].

3.2.7. Plataforma móvel

Estão disponíveis hoje em dia diversas alternativas a nível de sistemas operativos para dispositivos móveis, sendo os líderes do mercado actualmente o Android [62] e Apple iOS [63].

No âmbito do estágio, a escolha recaiu sobre o sistema operativo Android, visto a curva de aprendizagem ser menor e tendo em conta que o autor já tinha tido contacto prévio com esta tecnologia. Para além disso, para se poder programar para iOS, é necessário ter equipamento adequado (computador Apple). Mesmo tendo este equipamento, é necessário ter uma licença que permita o desenvolvimento para iOS, visto ser necessário testar a aplicação efectuada ou num dispositivo virtual ou num iPhone.

A aplicação móvel a ser desenvolvida, será orientada para os sistemas operativos Android mais usados no mercado, que como se pode observar pela Figura 8 é a que contempla a versão 4.x.

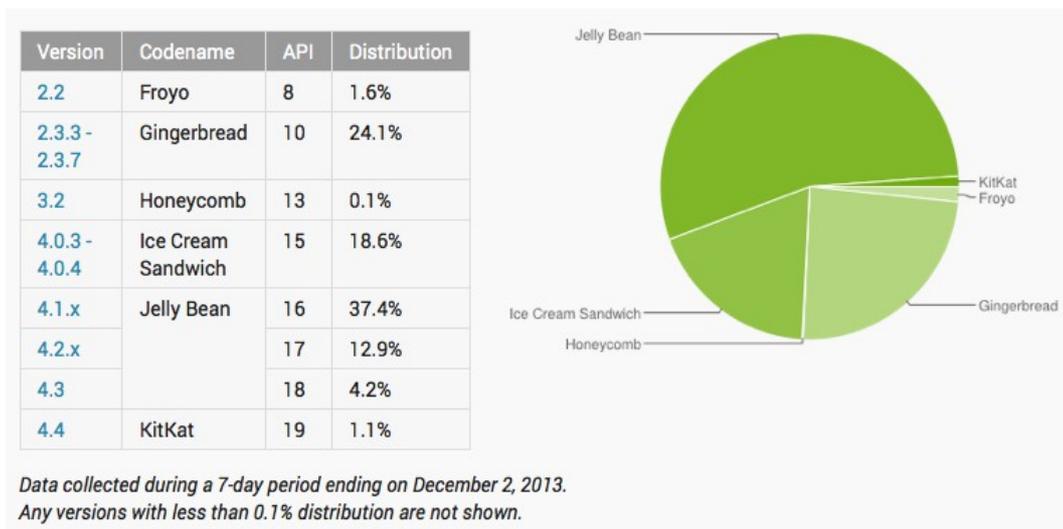


Figura 8: Percentagem de dispositivos que correm uma dada versão Android [64]

3.2.8. Conclusões

Ao longo da secção 3.2, foi feito um levantamento e estudo de ferramentas que apoiam o desenvolvimento do projecto proposto no âmbito do estágio. Desta forma, as tecnologias adoptadas consistem em:

- VPS com o CentOS 6.5;
- Apache;
- PostgreSQL;
- livraria de *tokens* interna;
- PHP e Codeigniter;
- HTML5, JS, CSS3;
- API do Google Maps e Leaflet;
- API do YouTube;
- API do Facebook;
- zbar SDK;
- GPS, WIFI, GSM;
- Android.

Para além disso, no levantamento anterior foram não só analisadas técnicas e tecnologias que serão utilizadas neste estágio, mas também vistas técnicas que no futuro poderão ser implementadas para melhorar e evoluir a presente plataforma.

Capítulo 4 - Especificação do sistema

Este capítulo tem como principal objectivo a compreensão das necessidades do sistema a implementar, tal será possível através da descrição da plataforma e do levantamento de requisitos, apresentados nas secções 4.1 e 4.2 respectivamente. Com base no levantamento de requisitos efectuado, será apresentado nas secções 4.3, 4.4 e 4.5 os casos de uso, os *mockups* e a arquitectura do sistema, respectivamente.

4.1. Descrição da plataforma

De modo a perceber-se melhor a plataforma, representada na figura 9, há que compreender em primeiro lugar quais as relações entre os diferentes tipos de utilizadores existentes com as diferentes aplicações e quais os objectos *core* pertencentes ao sistema.

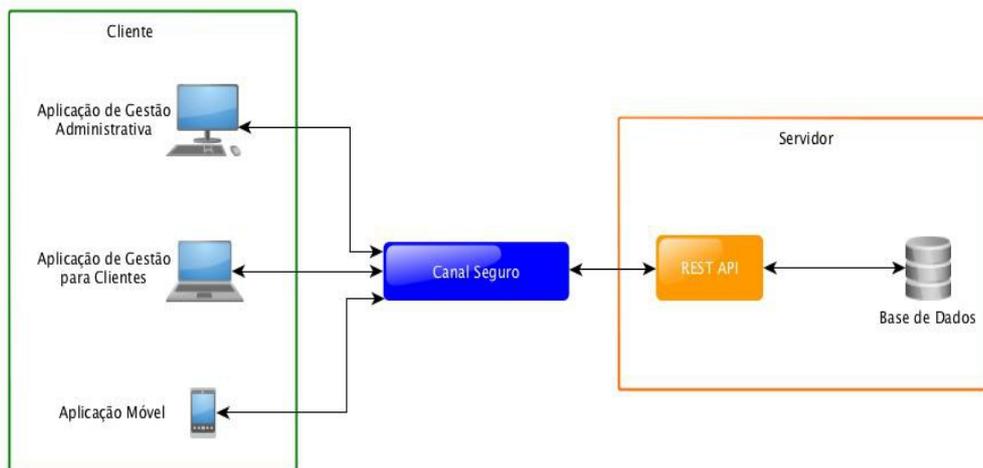


Figura 9: Arquitectura geral do sistema

Relação utilizador-plataforma

Para a autenticação nas diferentes aplicações, existem vários níveis de utilizadores, sendo eles do tipo L1 (super-administradores e administradores), L2 (clientes e gestores) e L3 (autenticação através de rede social e autenticação anónima). Na plataforma *web*, que engloba as duas aplicações de gestão do sistema, apenas existem utilizadores do tipo L1 e/ou L2, enquanto que para a aplicação móvel existem utilizadores do tipo L3, como se observa na figura 10.

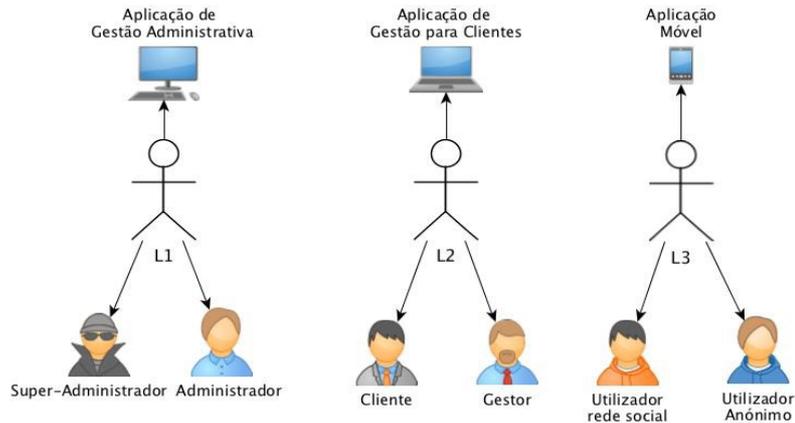


Figura 10: Tipos de utilizador

Objectos Base do Sistema:

Na figura 11 é possível visualizar de que maneira os objectos *core* estão relacionados.

Cada local georreferenciado tem um canal associado, onde podem existir dois tipos de portais, que não são mais do que derivações dos canais, sendo que os portais podem ser públicos ou privados. Cada tipo de portal é constituído por conteúdo multimédia, sendo estes sempre constituídos por *links*, imagens, vídeos e/ou texto. O portal privado ainda tem a capacidade de ter a(s) planta(s), imagem do local, podendo estas conter regiões (polígonos para georreferenciar uma área numa planta), que podem conter itens, sendo estes objectos georreferenciados numa região.

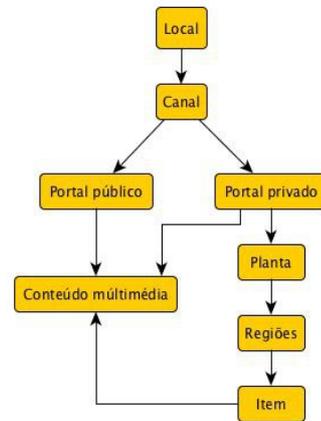


Figura 11: Objectos core

Exemplo:

Local	Coimbra
Canal	Jardins de Coimbra
Portal	Área com informação sobre todos os jardins de Coimbra
Planta	Representa os jardins existentes em Coimbra
Região	Jardim da Sereia
Item	Fonte da Nogueira
Conteúdo	Imagem com a fonte ou texto descritivo da fonte

Plataforma

A plataforma web divide-se em duas aplicações: a aplicação de gestão administrativa e a aplicação de gestão para clientes. Na primeira o utilizador que se está a autenticar tem de ser do tipo L1, enquanto que para a segunda o utilizador tem de ser do tipo L2.

Aplicação de Gestão Administrativa

A aplicação de gestão administrativa tem como finalidade permitir aos utilizadores L1 a gestão de utilizadores do tipo L2 e de canais. Na gestão de utilizadores, é possível a criação, edição ou remoção de utilizadores, desde que o tipo de utilizador seja igual ou inferior ao do utilizador L1 em causa. Na gestão de canais é possível aceitar/rejeitar pedidos pendentes de criação de novos canais efectuados pelos utilizadores L2, a criação de novos canais, apagar (pôr o canal inactivo) ou cessar (pôr o canal em manutenção) os canais. Esta aplicação ainda permite aos utilizadores do tipo L1 assumirem perfis de utilizadores, desde que o tipo de utilizador seja igual ou inferior, bem como verem relatórios do sistema onde se pode ver a utilização do armazenamento dos canais, a quantidade de canais existentes no sistema, quantidade de itens no sistema, quantidade de acessos às diferentes aplicações e a quantidade de gestores registados no sistema, podendo ainda exportar estes relatórios para um ficheiro CSV. Os *screenshots* desta aplicação podem ser vistos no anexo F.1.

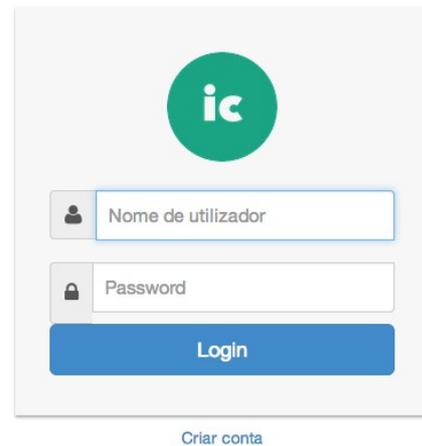


Figura 12: Acesso à aplicação de gestão administrativa

Aplicação de Gestão para Clientes

A aplicação de gestão para clientes, permite aos mesmos fazerem pedidos para criarem o seu próprio canal, sendo que mais tarde pode ser personalizado à sua medida. O canal criado apenas será visível para os utilizadores da aplicação móvel quando algum administrador aceitar o canal. Enquanto o cliente não definir a área de abrangência do canal não é possível fazer mais nada no canal. É possível a criação de gestores e adicionarem ou removerem

Gestão de Canais: Jardins de Coimbra

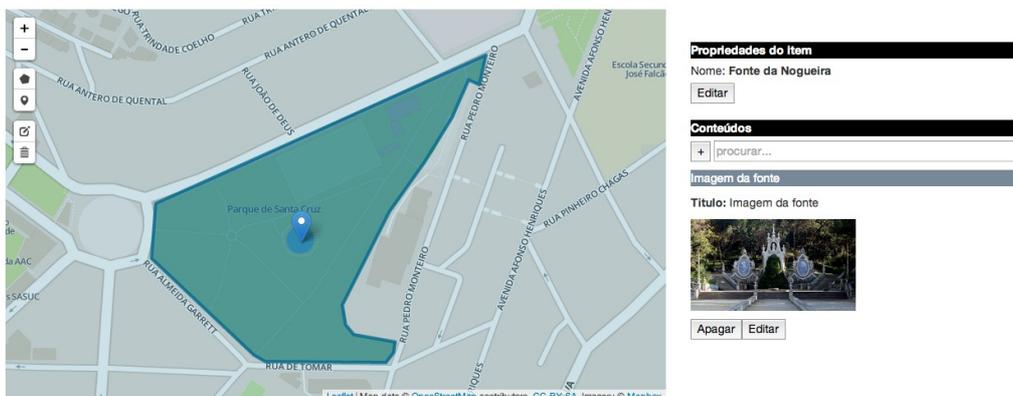


Figura 13: Personalização de itens

gestores de cada canal, sendo que cada gestor pode gerir múltiplos canais pertencentes ao mesmo cliente. Tanto os gestores como os clientes, podem editar os portais (público e privado) inserindo conteúdo multimédia, editar as permissões dos canais e dos respectivos portais. Em cada canal é também possível editar as plantas, editarem os dados do canal permitindo cessar o canal ou fechá-lo para manutenção e exportar relatórios, em CSV, associados ao uso do canal (utilização do armazenamento dos canais, quantidade de canais, quantidade de itens e quantidade de gestores). Os utilizadores do tipo L2 ao editarem a planta, podem criar regiões, tendo estas de estar na área de abrangência do canal, onde poderão posteriormente colocar marcadores pertencentes a itens deste canal. Cada item pode ser constituído por um ou mais conteúdos multimédia, caracterizando o mesmo. A aprovação de comentários é outra *feature* presente, permitindo assim um controlo sobre os itens e sobre os portais. Os *screenshots* desta aplicação podem ser vistos no anexo F.1.

Aplicação Móvel

A aplicação móvel permite ao utilizador, do tipo L3, visualizar quais os canais que o rodeiam. Ao entrar num canal, que está no seu raio de acção, o utilizador visualiza o portal público do canal, onde estão disponíveis os conteúdos multimédia previamente introduzido pelos utilizadores do tipo L2, de modo a darem uma visão/descrição do canal. Quando o utilizador está na área geográfica de abrangência de um canal, tem acesso a um conjunto de informação diversificada sobre o mesmo, como por exemplo, a consulta do portal privado desse canal. Adicionalmente, estão disponíveis outro tipo de conteúdos, de onde se destaca a(s) planta(s) do local, eventos e/ou exposições. Cada item pode ser abrangido socialmente por comentários dos utilizadores sendo necessária a aprovação posterior de algum utilizador do tipo L2 para poder ser mostrado a qualquer utilizador desta aplicação. Os comentários apenas poderão ser efectuados caso as permissões do canal/portal o permitam. Os *screenshots* desta aplicação podem ser vistos no anexo F.2.

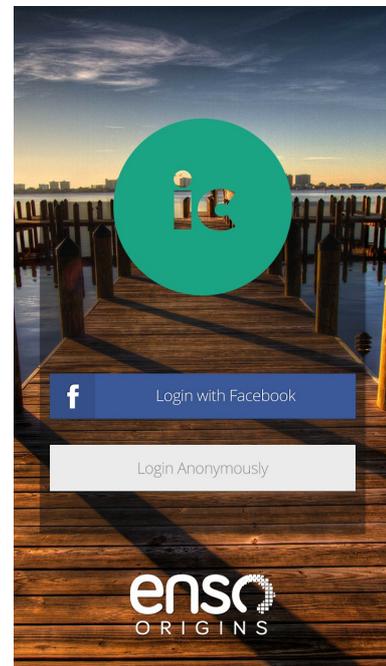


Figura 14: Layout de Autenticação

4.2. Levantamento de requisitos

O levantamento de requisitos foi dividido em diferentes módulos, sendo cada módulo referente a uma aplicação diferente. Nas secções 4.2.1 e 4.2.2 encontra-se os requisitos funcionais e não funcionais para as diferentes plataformas e com a respectiva classificação, para o âmbito do estágio.

Para a identificação dos requisitos houve diversas reuniões com a entidade visto que o software a desenvolver é uma ideia interna, a mesma é considerada como um cliente na fase de especificação de requisitos. A partir dessas reuniões, o autor identificou alguns requisitos, sendo depois essa lista aumentada com uma reunião final com o orientador Carlos Ramos para validar, adicionar e classificar os requisitos.

4.2.1. Requisitos Funcionais

Os requisitos funcionais diferem de aplicação para aplicação. Assim, nas tabelas 6 e 7, podemos ver os que dizem respeito à aplicação de gestão administrativa, na tabela 8 e 9 os que dizem respeito à aplicação de gestão para clientes e nas tabelas 10 e 11 os que dizem respeito à aplicação móvel. Cada tabela contém ainda o requisito funcional e a respectiva classificação no âmbito do estágio.

ID	Requisito funcional	Classificação
RF.1	O sistema deve permitir um utilizador L1 autenticar-se com <i>username</i> e <i>password</i> .	Must
RF.2	O sistema deve possibilitar o acesso a administração de canais/clientes.	Must
RF.3	O nível de acesso de super-administrador permite o acesso e controlo de todos os componentes.	Must
RF.4	O nível de acesso de administrador permite o acesso a: gestão de clientes/gestores/utilizadores e gestão de canais.	Must
RF.5	Todos os acessos ao sistema devem ser registados para consulta futura.	Must
RF.6	Todas as criações de acessos devem ser registadas.	Must
RF.7	Todos os registos de segurança devem conter sempre: <i>timestamp</i> , autor, acção.	Must
RF.8	O super-administrador tem a capacidade para criar qualquer tipo de utilizador da plataforma.	Must
RF.9	O super-administrador tem a capacidade para criar qualquer tipo de utilizador da plataforma.	Must
RF.10	O super-administrador tem a capacidade para remover qualquer tipo de utilizador da plataforma.	Must
RF.11	O super-administrador tem a capacidade para editar qualquer tipo de utilizador da plataforma.	Must

ID	Requisito funcional	Classificação
RF.12	Os utilizadores são classificados em 3 níveis, L1 (Super-Administradores, Administradores), L2 (Clientes, Gestores), L3 (Utilizadores, Utilizadores Anónimos).	Must
RF.13	O perfil de administrador tem capacidade para criar utilizadores L2 dentro da plataforma.	Must
RF.14	O perfil de administrador tem capacidade para cessar utilizadores L2 dentro da plataforma.	Must
RF.15	O perfil de administrador tem capacidade para editar utilizadores L2 dentro da plataforma.	Must
RF.16	Tem de ser possível criar clientes descrevendo os mesmos com as seguintes informações: - Nome; - Morada; - Data Nascimento/Criação; - Documento de Identificação; - Contactos Telefónicos; - Contactos de Email; - URL WEB; Identificação Fiscal; - Descrição Geral.	Must
RF.17	Tem de ser possível fazer reset às credenciais de acesso de clientes/gestores respeitando sempre a confidencialidade sendo as mesmas enviadas para o cliente/gestor diretamente.	Must
RF.18	Os utilizadores L1 podem assumir a identidade de um utilizador L2.	Must
RF.19	O utilizador L1 tem a capacidade para criar canais associados a um cliente	Must
RF.20	O utilizador L1 tem a capacidade para cessar um canal, sendo sempre necessário indicar o motivo de tal.	Must
RF.21	O canal tem de ser caracterizado pela seguinte informação: - Título; Descrição; - Data de criação.	Must
RF.22	Um canal tem de ter associado a si um local ou mais locais que limitam a acção do mesmo.	Must
RF.23	Um canal pode ter múltiplos gestores associados.	Must
RF.24	Cada canal pode ter um portal privado e um portal público.	Must
RF.25	O portal privado pode ser acedido por qualquer tipo de utilizador, desde que estejam na área geográfica da acção do mesmo.	Must
RF.26	O portal privado pode ser acedido por qualquer tipo de utilizador L3.	Must
RF.27	O conteúdo Multimédia pode ser composto pelos seguintes objectos: Imagens; - Vídeos; - Sons; - Texto; - Links.	Must
RF.28	Deve ser suportado o youtube como um plataforma externa de vídeo.	Must
RF.29	Aceitar pedidos para adicionar canais.	Must
RF.30	Os acessos à API devem poder ser feitos de modo encriptado.	Must
RF.31	O acesso à API deve ser devidamente autenticado, recorrendo à utilização de <i>tokens</i> .	Must
RF.32	Tem de ser possível obter estatísticas/relatórios detalhados da utilização de armazenamento.	Must

ID	Requisito funcional	Classificação
RF.33	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade de canais.	Must
RF.34	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade itens.	Must
RF.35	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade acessos.	Must
RF.36	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade de gestores.	Must
RF.37	Relatório exportáveis em CSV.	Must

Tabela 6: Requisitos funcionais (must) para a aplicação de gestão administrativa

ID	Requisito funcional	Classificação
RF.38	O utilizador L1 tem a capacidade para remover todo o conteúdo de um canal, sendo necessário indicar o motivo para tal.	Should
RF.39	O objecto vídeo deve suportar tanto o armazenamento local do vídeo, bem como a representação do mesmo através de plataformas externas de vídeo.	Should
RF.40	O objecto áudio deve suportar tanto o armazenamento local do áudio, bem como a representação do mesmo através de plataformas externas de áudio.	Should
RF.41	Relatório exportáveis em PDF, XML e JSON.	Should
RF.42	Os dados pessoais como por exemplo o nome, morada, telefone e email, devem poder se encriptados no armazenamento de dados.	Nice to Have
RF.43	Qualquer utilizador L1 deve poder ter múltiplos perfis associados à sua conta.	Nice to Have
RF.44	Deve ser suportado o vimeo como um plataforma externa de vídeo.	Nice to Have
RF.45	Um canal deve poder ser classificado quanto a conteúdos adultos ou não.	Nice to Have
RF.46	O objecto imagem deve suportar tanto o armazenamento local da imagem, bem como a representação da mesma através de plataformas externas de imagem.	Nice to Have
RF.47	Deve ser suportado o <i>instagram</i> como plataforma externa de imagens.	Nice to Have
RF.48	Deve ser suportado o <i>soundcloud</i> como plataforma externa de áudio.	Nice to Have
RF.49	Tem de ser possível obter estatísticas/relatórios detalhados dos acessos de gestão à plataforma.	Nice to Have
RF.50	Todas as estatísticas devem poder ser geradas aglomerando um ou múltiplos utilizadores, e devem ser temporais e com uma granularidade configurável.	Nice to Have

Tabela 7: Requisitos funcionais (should e nice to have) para a aplicação de gestão administrativa

ID	Requisito funcional	Classificação
RF.51	Qualquer utilizador L2 apenas pode visualizar canais, portais, mapas, itens, multimédia respeitantes ao canal/canais que lhe estão atribuídos.	Must
RF.52	O sistema deve permitir um cliente ou gestor autenticar-se com <i>username</i> e <i>password</i> .	Must
RF.53	Podem existir múltiplos gestores para um só canal.	Must
RF.54	O perfil de cliente tem todos os privilégios que um gestor tem.	Must
RF.55	Possibilitar o acesso a administração de canais/gestores.	Must
RF.56	Apenas o cliente tem acesso à criação de gestores.	Must
RF.57	Todas as operações/interacções dos clientes/gestores devem ser devidamente registadas.	Must
RF.58	Todos os registos de segurança devem conter sempre: <i>Timestamp</i> , Autor, Acção.	Must
RF.59	Tem de ser possível criar gestores descrevendo os mesmos com as seguintes informações: Nome; Morada; Data Nascimento/Criação; Documento de Identificação; Contactos Telefónicos; Contactos de Email.	Must
RF.60	O sistema deve permitir ao L2 editar as permissões (de visualização) do seu canal.	Must
RF.61	Tem de ser possível fazer <i>reset</i> às credenciais de acesso de gestores respeitando sempre a confidencialidade sendo as mesmas enviadas para o gestor diretamente.	Must
RF.62	O sistema deve permitir ao L2 editar as permissões (de visualização) dos portais.	Must
RF.63	Um cliente pode criar elementos numa região.	Must
RF.64	Cada portal tem de ser personalizável com os seguintes conteúdos: - Plantas; - Multimédia.	Must
RF.65	O objecto planta tem como base uma imagem rectangular.	Must
RF.66	O objecto planta tem de ser passível de ser dividido em regiões, regiões essas definidas por um polígono.	Must
RF.67	O objecto planta irá assumir que o interior das regiões é considerado a zona onde a planta efectivamente é reconhecida.	Must
RF.68	As plantas do portal são acessíveis quando abandonamos o mesmo e nos mantemos dentro do canal.	Must
RF.69	Dentro de cada canal temos a capacidade de ter múltiplos itens definidos.	Must
RF.70	um local é um ponto geo-referenciado ou um polígono (area) georreferenciada utilizado para caracterizar diferentes componentes desta plataforma.	Must
RF.71	Um item é caracterizado pelo seguinte conteúdo: Local; Título; Multimédia.	Must

ID	Requisito funcional	Classificação
RF.72	Permitir fazer um pedido aos administradores para a criação de um novo canal.	Must
RF.73	O sistema deve permitir ao cliente editar os dados/conteúdos multimédia de cada portal.	Must
RF.74	Um canal tem de ter a seguinte informação: - Título; Descrição; - Data de criação.	Must
RF.75	Os itens são georreferenciados nas regiões definidas nas plantas.	Must
RF.76	Cada planta deve permitir ao L2 fazer regiões na mesma.	Must
RF.77	Editar conteúdo multimédia associado aos itens.	Must
RF.78	O sistema deve permitir ao cliente moderar os utilizadores do canal.	Must
RF.79	O sistema deve permitir uma gestão do canal por parte do L2 do canal.	Must
RF.80	O sistema deve permitir ao cliente editar os dados de um canal.	Must
RF.81	Tem de ser possível obter estatísticas/relatórios detalhados da utilização de armazenamento	Must
RF.82	Tem de ser possível obter estatísticas/relatórios detalhados da utilização de quantidade de canais.	Must
RF.83	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade itens.	Must
RF.84	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade de gestores.	Must
RF.85	O sistema deve permitir ao L2 ver estatísticas do uso do canal.	Must

Tabela 8: Requisitos funcionais (must) para a aplicação de gestão para clientes

ID	Requisito funcional	Classificação
RF.86	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade de acessos.	Should
RF.87	Tem de ser possível obter estatísticas sociais sobre os utilizadores de nível L3 baseando-se na informação que conseguimos recolher nas plataformas móveis fornecendo assim dados relevantes e completos sobre o perfil dos mesmos.	Should
RF.88	O sistema deve permitir ver o portal público de um canal através de um URL publicamente acessível.	Should
RF.89	Tem de ser possível obter estatísticas/relatórios detalhados das zonas mais visitadas no seu canal.	Should
RF.90	O sistema deve permitir ao L2 ver gráficos das estatísticas.	Should
RF.91	Tem de ser possível obter estatísticas/relatórios detalhados dos acessos de gestão à plataforma.	Nice to Have

ID	Requisito funcional	Classificação
RF.92	O sistema deve permitir pôr um espaço temporal associado a cada conteúdo multimédia. Caso o cliente não preencha o espaço temporal assume-se que o conteúdo ficará disponível para sempre. Caso dois conteúdos estejam no mesmo espaço temporal, prevalece o que tem menor espaço temporal.	Nice to Have
RF.93	Todas as estatísticas devem poder ser geradas aglomerando um ou múltiplos canais, e devem ser temporais e com uma granularidade configurável.	Nice to Have
RF.94	Relatório exportáveis em PDF, CSV, XML e JSON.	Nice to Have

Tabela 9: Requisitos funcionais (should e nice to have) para a aplicação de gestão para clientes

ID	Requisito funcional	Classificação
RF.95	O sistema deve permitir uma identificação do utilizador L3 através de redes sociais (facebook).	Must
RF.96	Caso o utilizador L3 não se queira identificar, poderá na mesma aceder à plataforma sendo criado um perfil anónimo. Este tipo de utilizador será tratado da mesma forma que um utilizador com perfil social.	Must
RF.97	O sistema deve pedir permissões para guardar a posição GPS do utilizador L3.	Must
RF.98	O sistema deve mostrar os termos de utilização quando um utilizador L3 aquando da criação da conta rápida.	Must
RF.99	O sistema tem de permitir que o utilizador saiba quais os canais que o rodeiam.	Must
RF.100	O sistema tem de permitir que o utilizador consiga ver a informação do portal público dos canais que o rodeiam.	Must
RF.101	Assim que se entra num canal o portal público deve ser apresentado ao utilizador.	Must
RF.102	Quando o L3 se encontra numa região geo-referenciada pertencente ao canal em questão, o portal privado passa a estar visível.	Must
RF.103	Caso o utilizador L2 dê permissões, o utilizador de nível L3 pode comentar conteúdos multimédia.	Must
RF.104	Caso o utilizador L2 dê permissões, o utilizador de nível L3 pode introduzir conteúdo multimédia que necessitará da aprovação de L2.	Must
RF.105	Os comentários são constituídos por conteúdos multimédia.	Must
RF.106	O utilizador L3 poderá “navegar” pelas diversas plantas de modo a permitir dar-lhe uma visão global para onde pode ir e do que pode fazer.	Must
RF.107	Na navegação pelas plantas um utilizador L3 pode ver os itens que estão definidos na região.	Must
RF.108	O sistema deve permitir uma procura pelo nome do canal	Must

ID	Requisito funcional	Classificação
RF.109	O sistema deve permitir apresentar uma listagem de todos os canais existentes	Must
RF.110	O sistema deve permitir apresentar informações de um canal - detalhes/descrição do canal.	Must
RF.111	O sistema deve permitir a integração da aplicação com a API do facebook.	Must
RF.112	O sistema deve permitir a integração da aplicação com a API do Google.	Must

Tabela 10: Requisitos funcionais (must) para a aplicação móvel

ID	Requisito funcional	Classificação
RF.113	Aceder a conteúdo de outros canais que têm itens nas imediações, sem necessariamente sair do canal em que se encontra	Should
RF.114	O sistema permite enviar estatísticas das acções que um utilizador L3 faz num canal.	Should
RF.115	A aplicação deve adaptar-se graficamente recorrendo aos objectos gráficos que caracterizam o canal após a entrada no mesmo.	Nice to Have
RF.116	O sistema deve permitir uma procura pelo tipo do canal	Nice to have

Tabela 11: Requisitos funcionais (should e nice to have) para a aplicação móvel

4.2.2. Requisitos Não Funcionais

Os requisitos não funcionais envolvem todas as aplicações do sistema, podendo ser visualizados na tabela 12.

ID	Requisito não funcional	Classificação
RNF.1	O sistema deve permitir o acesso à aplicação de gestão administrativa através de <i>browser web</i> .	Must
RNF.2	Os acessos de administração e de gestão devem sempre recorrer a protocolos encriptados.	Must
RNF.3	Toda a plataforma tem de ser passível de ser suportada em <i>clouds</i> públicas sem recorrer a infra-estrutura dedicada para um determinado cliente (exceptua-se hardware necessário para a geolocalização).	Must
RNF.4	A API tem de ser facilmente acessível usando http/https.	Must
RNF.5	O sistema deve permitir o acesso à aplicação de gestão para clientes através de <i>browser web</i> .	Must
RNF.6	O sistema deve permitir comunicações com ligações seguras através de encriptação, entre o cliente e o servidor.	Should

Tabela 12: Requisitos não funcionais para todas as aplicações

4.3. Diagrama de casos de uso

Este diagrama é importante uma vez que permite dar uma visão global de todas as acções que os actores podem executar; foi criado a partir do levantamento de requisitos, onde cada acção levou à criação de um caso de uso. A lógica para a criação dos casos de uso, prendeu-se com a separação por módulos (aplicação de gestão administrativa, aplicação de gestão para clientes e aplicação móvel).

De seguida são explicados dois exemplo:

- Exemplo 1: Personalização de um canal na aplicação de gestão para clientes (ver figura 15);
- Exemplo 2: Personalização de uma planta na aplicação de gestão para clientes (ver figura 16).

De referir que o diagrama completo pode ser visualizado no anexo A.

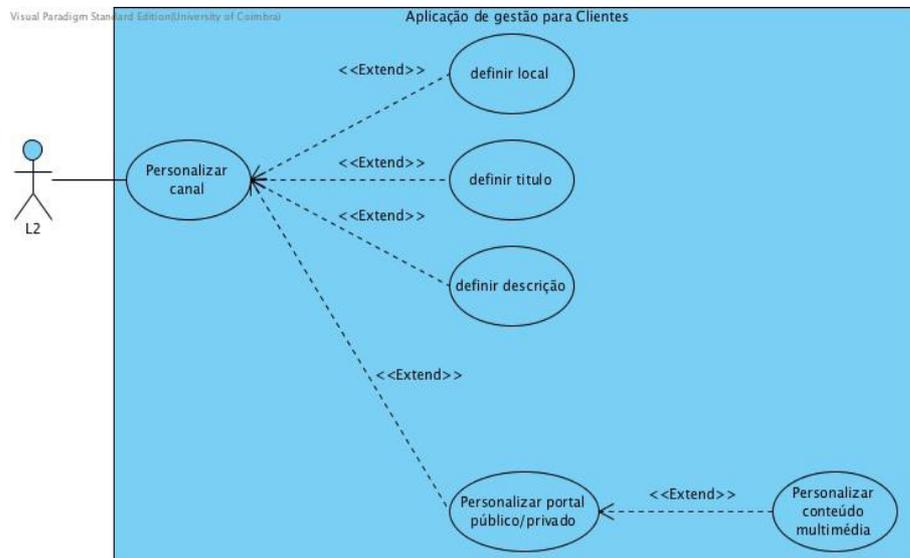


Figura 15: Exemplo do caso de uso de personalização de um canal

Os requisitos que envolvem o exemplo 1, visível na figura 15, prendem-se com os vistos na tabela 6, tendo a referência RF.24 e na tabela 8 com as referências RF.73 e RF.80.

No exemplo 1 apresentado, repara-se que o utilizador do tipo L2, pode personalizar canal, conseguindo definir o local, o título e a descrição. O canal pode ter dois tipos de portais, o público e o privado, sendo estes constituídos por conteúdos multimédia. Este tipo de conteúdos são tipicamente imagens, texto, vídeos e/ou *links*, podendo ser sempre personalizáveis pelo utilizador.

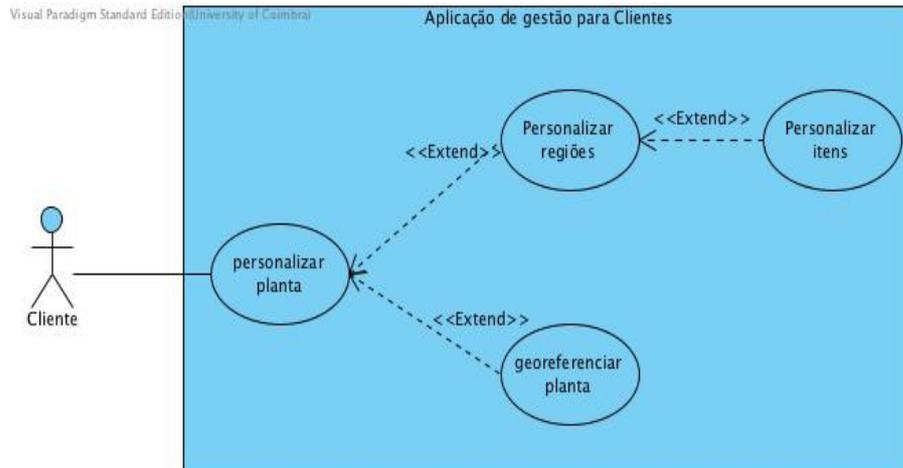


Figura 16: Exemplo do caso de uso de personalização de uma planta

Os requisitos que envolvem o exemplo 2, visível na figura 16, prendem-se com os vistos no levantamento de requisitos na tabela 8 tendo como referências RF.63, RF.64, RF.65, RF.66, RF.70, RF.75 e RF.76.

No exemplo 2 apresentado, o utilizador do tipo L2 (cliente) pode personalizar a planta, sendo para tal necessário ter a área de abrangência do canal definida (“gerenciar planta”). Após isso, a personalização da planta consiste na adição, remoção ou edição de regiões e caso tenha regiões consiste ainda na personalização de itens (adição, edição ou remoção). Os itens são constituídos por conteúdos multimédia, que como explicado no exemplo 1, são do tipo imagens, texto, vídeos e/ou *links*.

4.4. Mockups

Para facilitar a compreensão do modo como é possível “navegar” pelos *mockups*, foi criado um diagrama de *flow*. Deste diagrama constam todas as transições possíveis, sendo apresentado no anexo B.1.

Nas figuras 17 e 18 é possível ver os *mockups* referente aos exemplos 1 e 2, respectivamente, apresentados e explicados neste mesmo capítulo na secção 3 (figuras 15 e 16, respectivamente). Ainda de mencionar que os restantes *mockups* podem ser consultados no anexo B.2.

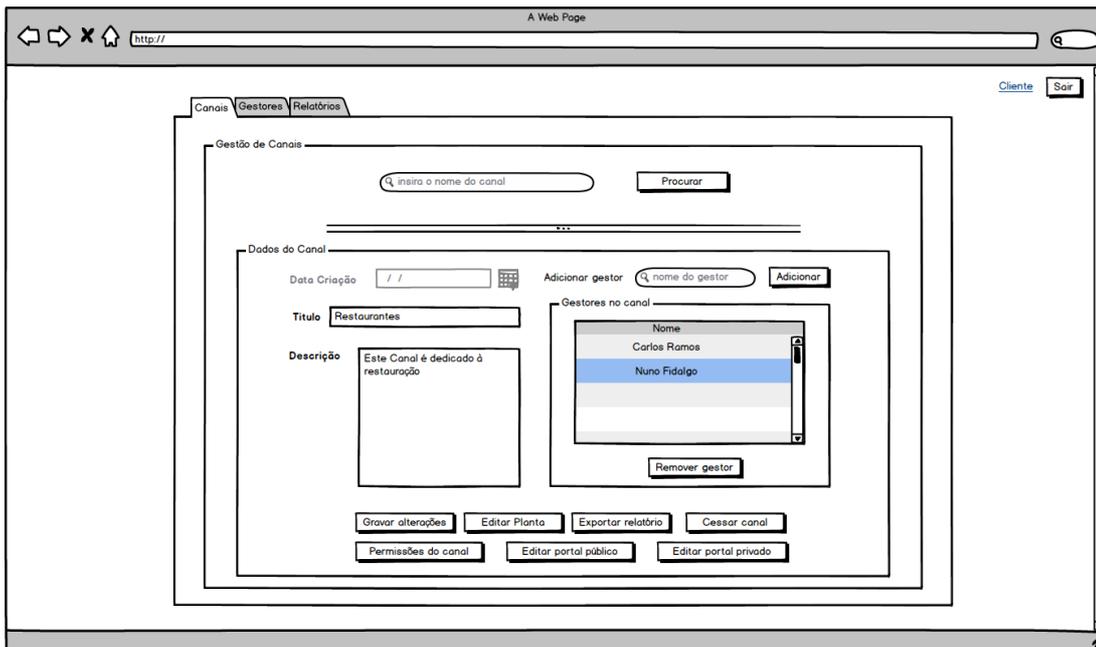


Figura 17: Mockup para personalizar canal



Figura 18: Mockup para personalizar planta/regiões

4.5. Arquitectura

Esta secção apresenta as diferentes componentes que compõem a arquitectura do sistema através da apresentação e explicação do diagrama físico, diagramas de estados e diagrama entidade-relação. Para tal, foi necessário ter em conta o levantamento de requisitos apresentado na primeira secção deste capítulo.

De notar que as tecnologias que foram usadas e que constam do diagrama físico do sistema, foram estudadas no capítulo 3 – Estado da Arte - na secção 2, deste documento.

Todos os diagramas desenvolvidos no âmbito da arquitectura podem ser consultados no anexo C, mais concretamente encontramos o diagrama físico no anexo C.1, o diagrama de estados no anexo C.2 e o diagrama entidade-relação no anexo C.3.

4.5.1. Diagrama físico

O diagrama físico da aplicação mostra quais os componentes físicos que a plataforma necessita e de que forma estão ligadas. Na figura 19, encontra-se este diagrama, onde é possível ver que a aplicação *web* que corre através de um *browser*, a aplicação móvel que foi desenhada para correr em qualquer plataforma móvel embora para efeitos de estágio apenas iremos ter em conta a plataforma *Android*. Existe ainda uma base de dados em *PostgreSQL* e um servidor que terá um módulo de *rest*, acessível através do servidor HTTP apache. As ligações entre o servidor e a aplicação *web* ou a aplicação móvel são feitas recorrendo a SSL, e a troca de informação é feita recorrendo a *JSON*.

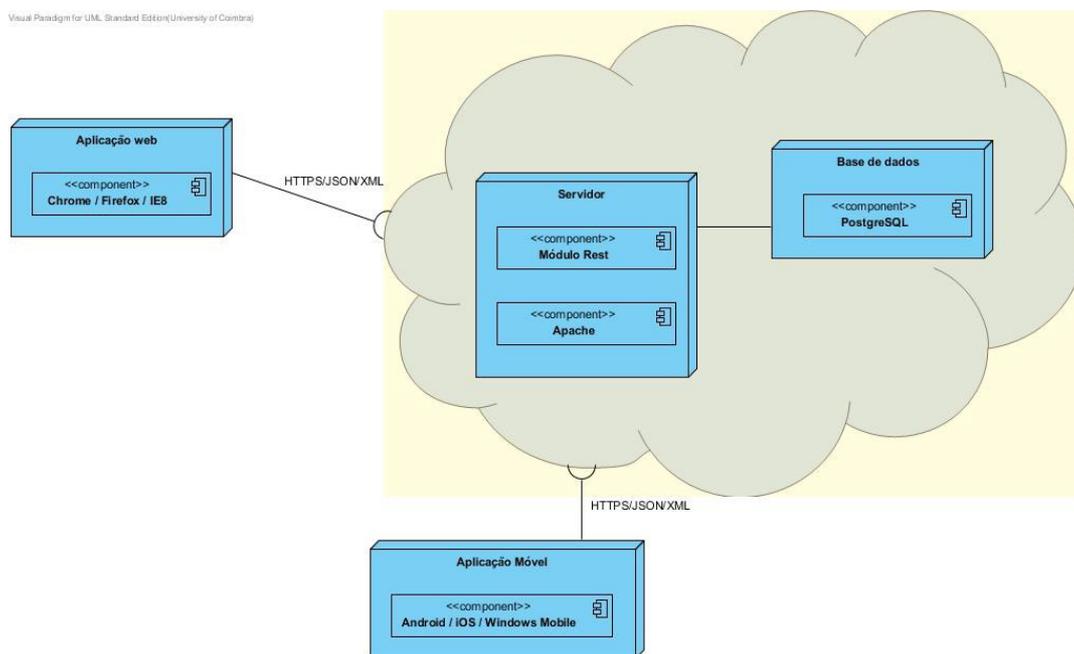


Figura 19: Diagrama físico

4.5.2. Diagramas de estados

Este tipo de diagramas são orientados aos objectos e são usados para descrever o comportamento dos mesmos.

Tal como explicado na secção dos casos de uso do presente capítulo, um dos exemplos descrito foi sobre a personalização de um canal (exemplo 1). Assim, o diagrama de estados da figura 20, representa esse exemplo. Quando um utilizador quer personalizar um canal, pode editar o local, o título e/ou a descrição ou então editar o portal privado ou o público. Assim, se o utilizador apenas alterar o local, o título e/ou a descrição gravando posteriormente as suas alterações encontram-nos num estado final. Outra forma de atingir o estado final deste caso de uso acontece quando o utilizador edita qualquer tipo de portal, personalizando ou criando qualquer tipo de conteúdo multimédia do mesmo.

No anexo C.2, podem ser vistos todos os diagramas de estados que foram criados.

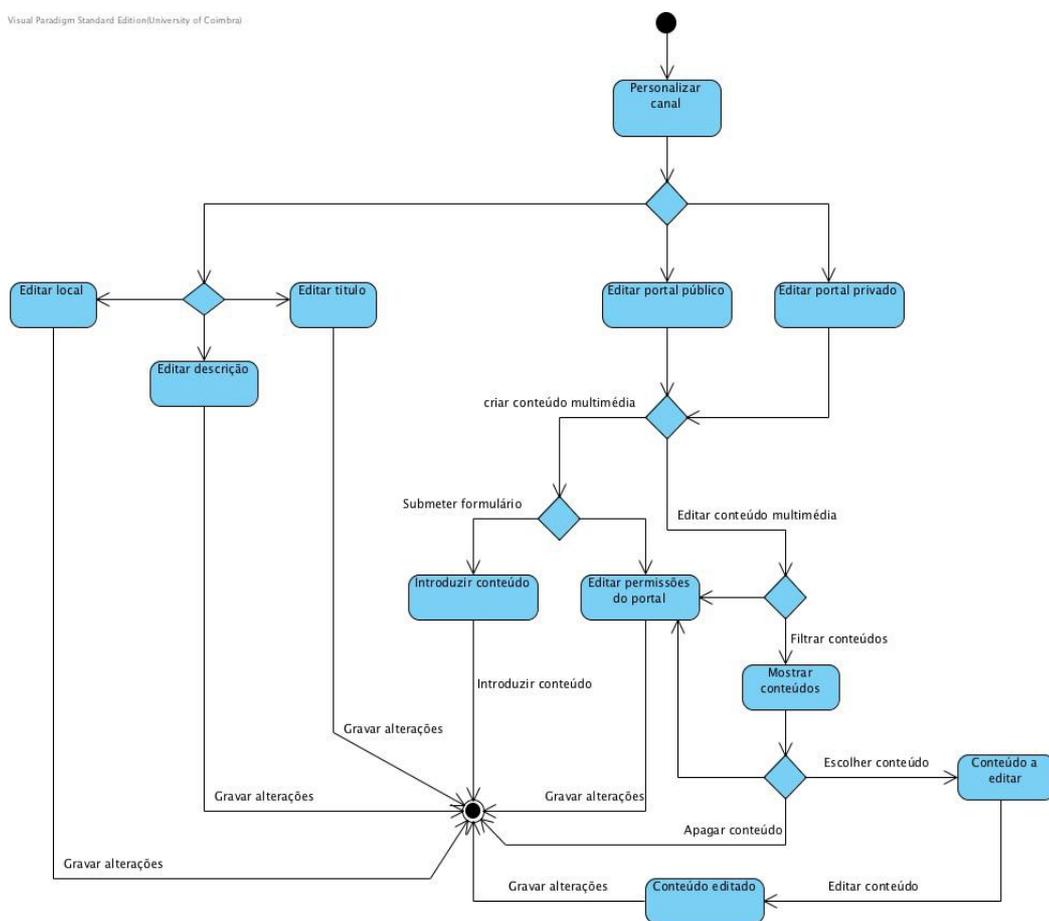


Figura 20: Diagrama de estados - personalização dos dados do canal e dos portais

4.5.3. Diagrama ER

Este diagrama descreve a base de dados usada, mostrando como se relacionam as tabelas, quais os campos que contêm e qual a cardinalidade das relações existentes. No anexo C.3 é possível ver este diagrama.

Capítulo 5 - Implementação

Este capítulo focar-se-à na descrição da metodologia de desenvolvimento usada, na descrição técnica da implementação das diferentes aplicações e na forma como estas interagem entre si. Pretende-se dar a conhecer não só a forma de como foi realizado o desenvolvimento, mas também as decisões tomadas com o objectivo de garantir a completude do projecto e a sua validação.

5.1. Metodologia de Desenvolvimento

A metodologia de desenvolvimento seguida foi baseada em *Scrum*, figura 21, tendo como fases o *product backlog*, *sprint backlog*, *sprint* e o *product increment*. Prescindiui-se do *daily scrum*, reunião diária do *sprint*, colmatando com reuniões semanais de avaliação do *sprint*. Caso surgissem dúvidas, no dia seguinte era feita uma *daily scrum*.

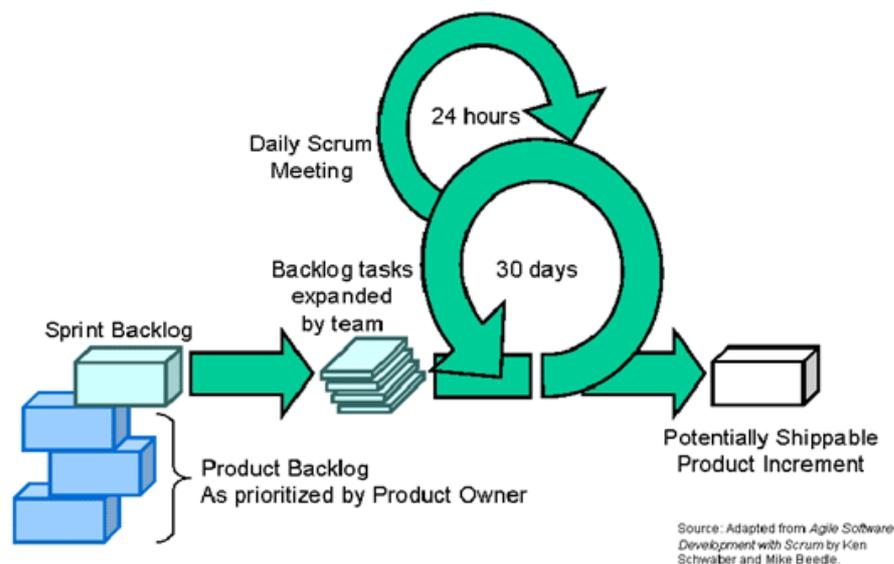


Figura 21: Metodologia de desenvolvimento scrum [65]

Segundo o guia do *Scrum*[66] da autoria de Ken Schwaber e Jeff Sutherland, “Cada *Sprint* pode ser considerado como um projecto com um horizonte de menos de um mês”. O espaço temporal escolhido foi de duas semanas, visto se fossem ciclos menores iria aumentar o número de tarefas pois teriam de ser tarefas mais pequenas, se fossem ciclos maiores seria complicado de situá-las no tempo levando a uma gestão de trabalhos mais complexa.

Assim, inicialmente definiu-se o *product backlog*, com base no levantamento de requisitos efectuado que se encontra na secção 4.2.

Antes de se iniciar o *sprint* era efectuada uma reunião onde, em conjunto com o orientador da empresa, era feito o *sprint planning*. De referir ainda que no final de cada *sprint* tudo o que foi implementado foi revisto e validado pelo orientador da empresa.

No anexo G.1 é possível ver o *product backlog*, bem como em que *sprint* foi iniciada a *user story* e em que *sprint* foi finalizada.

5.2 Aplicações

Na presente secção começa-se por apresentar a API de suporte, passando posteriormente por cada aplicação, finalizando com a explicação das comunicações entre as aplicações e o servidor que contém a API de suporte.

5.2.1 REST API

A REST API serve de suporte a cada uma das aplicações, sendo que por esta passa cada pedido efectuado. Para se perceber melhor em que consiste, de seguida é apresentado os verbos existentes, as respostas possíveis de serem utilizadas e a estrutura utilizada em cada função REST.

Verbos HTTP

Há algumas convenções para o uso de verbos no REST. A convenção utilizada neste projecto foi decidida, numa reunião na empresa, sendo que é a usada actualmente pela entidade.

Desta forma, os quatro verbos HTTP utilizados são o GET, POST, PUT e DELETE.

GET – utilizado para a leitura de recursos, como por exemplo para fazer uma procura de utilizadores ou de canais.

POST – utilizado para criar um novo recurso, como por exemplo para criar um utilizador ou um canal.

PUT – utilizado para fazer *uptade* de um recurso, como por exemplo editar dados de um utilizador ou de um canal.

DELETE – utilizado para apagar um recurso, como por exemplo eliminar uma conta de utilizador ou um canal.

De referir ainda, que apesar de se utilizar o verbo DELETE em alguns casos, o que realmente acontece é um *update* na base de dados, visto não se pretender realmente apagar dados da base de dados. Isto acontece não só por ser convenção na empresa, mas também para manter a integridade na base de dados, pois caso uma conta fosse eliminada realmente da base de dados, então teria de se remover tudo o que era referente a esse utilizador.

Códigos de resposta

Quando é efectuado um pedido HTTP ao serviço REST é devolvido com a resposta um código associado. Assim, são vários os tipos de resposta possíveis de serem dados consoante o pedido efectuado. Foi decidido em reunião, na empresa, que os diferentes códigos de resposta que devem ser usados são os que se encontram na tabela 13. Caso fosse necessário utilizar outro que não se encontrasse definido, era possível utilizar outro que se adequasse, bastando que para tal o mesmo fosse devidamente fundamentado, caso este que não se veio a verificar.

Assim sendo, podemos dividir estes códigos em três grandes grupos, sendo os códigos 2xx de sucesso, os 4xx códigos de erro do lado do cliente e os 5xx de erro do lado do servidor.

Desta forma, todos os códigos 2xx são códigos de sucesso, dependendo do pedido que foi realizado. O código 200 é utilizado para todos os casos de sucesso, isto é, quando tudo foi processado correctamente e sem falhas. Quando é necessário, há códigos mais específicos como é o caso do código 201 e 202 para o sucesso de inserção e editar ou apagar dados da base de dados, respectivamente. A resposta dada é 400, quando houver algum pedido mal formado, o 401 é devolvido quando o utilizador em questão não tem permissões ou não está

autenticado para fazer um pedido, enquanto que o 404 é devolvido quando não foram encontrados dados para devolver ao utilizador. Finalmente usa-se o código 500, caso algo corra mal do lado do servidor.

Códigos de Erro	
200	OK
201	Created
202	Updated/Deleted
400	Bad Request
401	Not Authorized
404	Not Found
500	Internal Server Error

Tabela 13: Códigos de resposta utilizados pela REST API

Livraria Logs

Foi criada um livraria no âmbito do estágio e que está a ser utilizada pela empresa, para adicionar um sistema de *logs* a qualquer aplicação desenvolvida.

Esta livraria tem uma função `addLog()`, que recebe como argumentos o id do utilizador que está a executar a acção, a acção que está a ser executada, a plataforma de onde está a ser executada a acção, o nível de severidade e a facility. O funcionamento desta função permite a qualquer altura registar um evento, relacionando o mesmo com um determinado utilizador e indicando a acção que foi executada pelo mesmo. Esse evento é organizado de acordo com a plataforma onde está a executar, bem como é classificado de acordo com uma sub-área (*facility*) e com um dado nível de severidade.

Como indicado a *facility* serve para num futuro, quando se quiser fazer uma plataforma onde se possa ver os *logs* de uma dada aplicação, estes poderem ser visto por grupos, a título de exemplo, se a acção for feita sobre os canais, então a *facility* será canais, se for feita sobre utilizadores então será utilizadores.

Os níveis de severidade, são níveis de erro que foram definidos também internamente em reunião, sendo os níveis os seguintes:

- 0 – Emergency – o erro é crítico e influencia grande parte do sistema;
- 1 – Alert – deve ser visto urgentemente;
- 2 – Critical – condições críticas específicas;
- 3 – Error – condições de erro de reduzida criticidade;
- 4 – Warning – condições de aviso;
- 5 – Notice – condição normal, mas com informação significativa;
- 6 – Informational – condição normal, mas apenas serve de informação sem necessidade de análise;
- 7 – Debug – nível de *debug* para apoio ao desenvolvimento.

De referir ainda que os níveis podem ser usados consoante o bom senso de cada programador, não havendo ainda nenhuma restrição exacta para o uso de cada nível.

Estrutura REST

Para não haver equívocos ou falhas desnecessárias, foi definida uma estrutura para cada função de REST. Assim, cada pedido efectuado passa por algumas etapas importantes, sendo elas:

1. Leitura de variáveis;
2. Verificação do *token*;
3. Verificação da acção;
4. Execução das operações;
5. Adição de um log;
6. Resposta.

A leitura de variáveis (**fase 1**), serve para se ler o *token* de quem está a realizar o pedido e qualquer outra variável que venha no URL. Após isto procede-se a uma verificação do *token* (**fase 2**) para se saber se é válido. Caso seja válido, é desta forma que se consegue identificar quem está a fazer o pedido, caso não o seja é dada uma resposta do tipo 401 visto não ter autorização para fazer a chamada pretendida.

Caso o *token* seja válido, de seguida é verificado se o utilizador em causa tem permissões para executar a acção que está a tentar (**fase 3**). Isto só é possível, visto ter sido utilizado um controlo de acessos baseado em níveis hierárquicos dos utilizadores (RBAC). Caso se verifique que o utilizador não tem permissões para a execução da acção é enviado uma resposta com o código 401. Desta forma, consegue-se filtrar todos os pedidos efectuados à API criada, não permitindo a ninguém que não esteja registado no sistema e que mesmo que esteja mas não tenha permissões, fazer pedidos à base de dados. Caso se passe com sucesso na verificação do *token* e na verificação das permissões para a acção pretendida, então chega-se à etapa da execução das acções (**fase 4**) onde se faz a chamada pretendida à base de dados, que consoante a resposta dada devolve um código de resposta ao utilizador que está a executar a acção. A resposta final (**fase 6**) nunca é devolvida ao utilizador, sem antes ser adicionado um *log* à base de dados (**fase 5**).

Esta estrutura foi criada de modo a garantir que todas as funções REST são criadas de uma forma consistente e segura.

5.2.2 Interação aplicação-servidor

Esta secção tem o intuito de demonstrar a especificação comportamental da solução desenvolvida. Na figura 22 estão representadas as comunicações existentes entre as diferentes aplicações e o servidor. O servidor é composto pela API de suporte criada, pelas camadas do modelo MVC criado e pela base de dados. Em todas as comunicações existe um indicador numérico para auxiliar na compreensão e acompanhamento na explicação da figura.

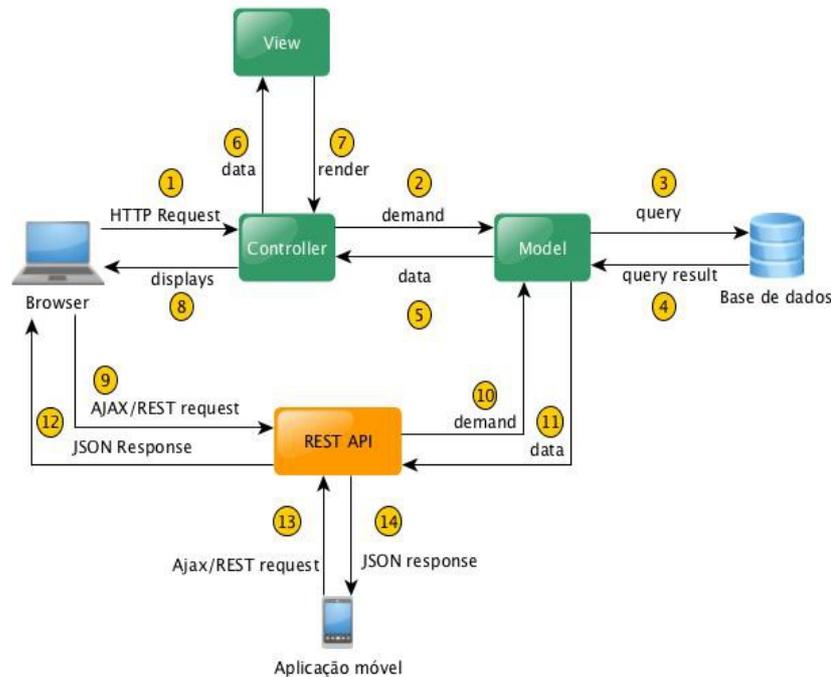


Figura 22: Comunicações existentes na plataforma

Explicação do Processo de chamadas Cliente-Servidor

- Quando acedemos à plataforma por um browser *web* é efectuada uma chamada HTTP ao servidor *web* que a reencaminha para o controlador (1), sendo este o responsável pelo fluxo da aplicação.
- Após a chamada o controlador recebe o pedido efectuado pelo utilizador e decide qual a vista que irá ser mostrada ao utilizador (6,7,8).
- Um exemplo desta situação acontece quando um utilizador acede a <https://neala.enso-origins.com/ic>, sendo-lhe apresentada a página de autenticação. Neste exemplo específico, apenas há um pedido ao controlador que envia a vista para o utilizador. Após a autenticação, o controlador necessita de carregar alguns modelos (*tokens*, *rbac* e *logs*), visto verificar a validade do *token* de quem fez o pedido, saber se quem fez o pedido tem permissões para o fazer e adicionar um *log* ao sistema a informar que acção foi efectuada (seja de erro ou de sucesso), quem é que a executou e a que horas (2,3,4,5).

Transferência de Dados em pedidos REST

- Sempre que o utilizador faz um pedido AJAX/REST, usando um dos verbos definidos e explicados na sub-secção 5.2.1 (9) e que não seja para mudar de vista, este é processado pela API de suporte criada. A API apenas vai carregar os modelos necessários para executar o pedido efectuado pelo utilizador seguindo sempre uma estrutura para tal, como explicado na sub-secção 5.2.1 – estrutura REST.
- Na fase 4 (execução das operações), da estrutura REST, é onde se faz os pedidos necessários ao modelo, que ao comunicar com a base de dados devolve o resultado da(s) operação(ões) (10,3,4,11). A resposta para o *browser* é dado por JSON, onde é colocado o código de resposta adequado, consoante o sucesso ou insucesso da acção pedida, bem como em caso de sucesso os dados referentes ao pedido efectuado (12).
- Exemplo prático – autenticação numa das aplicações *web*:
 - São usadas várias fases, tendo em conta que para além de haver confirmação das credenciais também há a mudança da vista;
 - O utilizador após preencher o formulário de autenticação clica no botão de “Login”, que faz um pedido AJAX/REST à API de suporte;
 - A API de suporte, vai seguir o procedimento, que se encontra explicado na sub-secção 5.2.1, e em caso de insucesso na verificação do *token* ou na verificação da acção devolve o código de erro 401, visto o *token* não ser válido ou não tenha autorização para executar uma dada acção;
 - A autenticação é um caso particular visto haver uma verificação das credenciais introduzidas pelo utilizador, que caso estejam incorrectas é devolvido o código 404 pela API de suporte ao utilizador, não o permitindo entrar na aplicação. Caso estejam correctas, é verificado se já existe um *token* atribuído. Se não houver é gerado um novo, sendo essa a resposta enviada ao utilizador acompanhada do código de sucesso 200;
 - Para a verificação das credenciais e a atribuição ou re-validação do *token* são feitos pedidos ao modelo, que responde consoante o encontrado. Por exemplo, se as credenciais estiverem erradas, responde apenas a dar *false*, informando a API de suporte que existe um erro com o pedido efectuado. Caso as credenciais estejam correctas, então o modelo devolve à API a informação de qual é o id do utilizador;
 - A API pede ao modelo para verificar se já existe um *token* atribuído a esse utilizador, caso exista então vai revalidar e devolve o *token*, caso não exista é criado um novo *token* para ser incluído na resposta para o *browser*. Se houver algum erro, na criação do *token*, é devolvido um erro do tipo 500;
 - Quando a resposta de sucesso chega ao *browser*, este “informa” o controlador que já tem um *token* e faz o pedido HTTP para aceder à aplicação à qual tem permissões, consoante o tipo de utilizador e as permissões que tem.

Fluxo aplicação móvel

Quanto à aplicação móvel, o controlo de fluxo da aplicação é gerido na própria aplicação, não sendo necessário fazer pedidos ao servidor para mudar de vista. A aplicação apenas se liga à API de suporte para poder mostrar informação ao utilizador ou poder comentar conteúdos ou itens. Os pedidos efectuados pela aplicação móvel à API de suporte são tratados exactamente como os das aplicações *web* (13,10,3,4,11,14).

5.2.3 Aplicações web

As aplicações *web* desenvolvidas assentam sobre uma arquitectura MVC. Esta arquitectura, representada na figura 23, permite dividir as funcionalidades do sistema em camadas separando a representação da informação da interacção do utilizador com o sistema. As três camadas que o constituem são o modelo, a vista e o controlador. O modelo é utilizado para manipular e aceder a toda e qualquer informação vinda da base de dados, a vista é responsável pela informação mostrada e de que maneira é organizada essa informação para ser apresentada ao utilizador, enquanto que o controlador é responsável por controlar todo o fluxo de informação que passa pelo sistema.

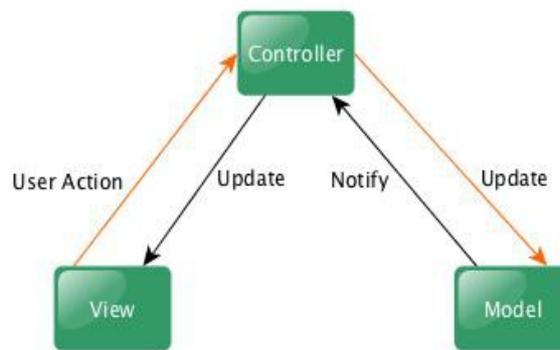


Figura 23: Modelo MVC

Passando à explicação da figura 23, quando o utilizador faz uma acção num *browser*, esta é recebida pelo controlador que, caso seja necessário, pede ao modelo para realizar uma consulta na base de dados. Após essa consulta, é devolvido ao controlador os resultados que, posteriormente, são enviados para a vista do utilizador. Explicando com um exemplo prático, quando um utilizador faz um pedido para visualizar todos os canais existentes no sistema (acção despoletada na vista), este pedido é feito ao controlador que faz o pedido ao modelo. O modelo executa o pedido, executando uma *query* de *select* na base de dados, devolvendo uma lista com os resultados encontrados ao controlador. Este envia o resultado para *browser* (vista), de modo ao utilizador poder visualizar o resultado do pedido efectuado. De notar que é apenas um exemplo explicativo de como funciona o modelo MVS, não representando o que realmente acontece na aplicação, visto os pedidos efectuados pelo utilizador passarem pela API de REST criada. Como explicado na sub-secção 5.2.2, o controlador serve para controlar o fluxo da aplicação, sendo que os pedidos que necessita de fazer ao modelo passam por:

- saber a validade do *token* recebido;
- saber, através do *token*, que utilizador está a fazer o pedido;
- ver se o utilizador tem permissões para executar a acção que foi pedida;
- adicionar um *log* à base de dados.

5.2.3.1 Funcionalidades implementadas

Das funcionalidades implementadas, para cada aplicação *web*, há quatro grandes focos fundamentais sendo eles: utilizadores, canais, moderação de comentários e relatórios.

Utilizadores

Em ambas as aplicações *web* é possível criar, visualizar, editar ou remover utilizadores. Porém existe uma clara diferença entre as duas aplicações, sendo que na de gestão administrativa, as operações efectuadas podem ser sobre qualquer tipo de utilizador sem haver restrições, enquanto que na de gestão para clientes, as operações só podem ser ou sobre o próprio utilizador ou então sobre os gestores criadores pelo cliente.

Na aplicação de gestão administrativa, é possível assumir a conta de um utilizador. Esta funcionalidade permite a um administrador auxiliar um cliente ou gestor com maior facilidade, visto ser a única forma de ajudar à gestão do seu canal.

Canais

O módulo referente aos canais acaba por ser um dos mais complexos, visto ter muitas diferenças entre as duas aplicações. A possibilidade de criação, visualização, edição ou remoção de um canal, em ambas as plataformas, funciona essencialmente da mesma forma, tendo a particularidade de na aplicação de gestão administrativa esta funcionalidade ser abrangente a todo o sistema, enquanto que na aplicação de gestão para clientes apenas afecta o que seja referente ao cliente. Um canal criado por um cliente, antes de poder ser visualizado futuramente na aplicação móvel, tem de ser aprovado por um administrador. Assim, quando um cliente cria um canal, ele na verdade está a fazer um pedido para a criação e não é possível estar activo enquanto um administrador não o aceitar.

Antes de poder gerir qualquer uma das funcionalidades descritas em seguida o cliente ou administrador tem de definir uma área de abrangência para o seu canal, “desbloqueando” assim a gestão do canal. Funcionalidades implementadas:

1. Personalização de uma imagem para o canal

A personalização da imagem do canal, consiste em definir uma imagem que será visualizada na aplicação móvel. O detentor do canal, pode escolher ter ou não uma foto associada ao canal.

2. Personalização de gestores associados ao canal

A cada canal é possível associar um ou mais gestores, de modo a ajudarem a administrar o mesmo. Esta funcionalidade permite ao cliente, criar contas de gestor e adicioná-las ao(s) canal(is) que deseje. Caso um cliente, verifique que um gestor já não é necessário, este pode removê-lo da lista de gestores do canal em questão tirando-lhe desta forma os privilégios de visualizar e gerir o canal.

3. gestão de permissões referentes ao canal

As permissões do canal permitem, ao proprietário ou a quem for gestor do mesmo, definir se o canal pode ser visualizado na aplicação móvel e/ou se permite aos utilizadores fazerem comentários em qualquer parte do canal.

4. Edição dos dados do canal

A edição dos dados do canal, passa por poder alterar o nome, descrição e estado sendo os diferentes estados possíveis:

- activo
- fechado
- manutenção.

Um canal activo é um canal que está em uso, estando visível a qualquer utilizador da aplicação móvel, caso as permissões do canal o permitam;

O estado fechado significa que não é visível para ninguém e que já não está em uso;

O estado de manutenção significa que quem gere o canal, está a fazer “remodelação” ao mesmo, não permitindo ser visível na aplicação móvel mas que também não está em desuso.

5. Gestão dos portais público e privado e permissões dos mesmos

Um canal é constituído por dois portais, público e privado.

Portal Público

Este portal está acessível a qualquer utilizador da aplicação móvel e apenas pode ser visualizado, caso as permissões do portal ou do canal o permitam. É possível a criação de conteúdos, sendo para isso necessário introduzir um título, um tipo de conteúdo (link, video, imagem ou texto) e introduzir o conteúdo que se deseje. Ainda se pode editar os conteúdos ou até mesmo apagá-los. Na edição de conteúdos não é possível mudar o tipo de conteúdo, sendo para isso necessário proceder à remoção e à criação de um novo. É possível editar as permissões e permitir ou não comentários nos conteúdos introduzidos neste portal. Este portal pode ser visualizado recorrendo a um URL público.

Portal Privado

No portal privado a criação, edição e remoção, bem como o controlo de permissões do portal funciona da mesma maneira que no portal público. A grande diferença entre estes portais é que este se destina apenas a quando, um utilizador da aplicação móvel, se encontra no interior de um local. Para aceder a este portal é necessário ou ler o código QR correcto ou introduzir um código. Assim, neste portal é possível a criação ou renovação do código de acesso, sendo este código gerado de forma aleatória e transformado, recorrendo ao uso da livraria zbar, num código QR.

6. Gestão da planta

Uma das funcionalidades mais importantes, visto ser o que vai apoiar os utilizadores da aplicação móvel no local onde se encontram, é a gestão da planta do canal.

É possível criar regiões na área de abrangência do canal e nessas regiões definir itens. As regiões podem ser editadas ou removidas, sendo que se removidas os itens nela contida também o são, de modo a garantir uma integridade na aplicação.

Os itens podem ser removidos todos ou isoladamente ou podem ser movidos para outra região. A cada item é possível:

- Definir/Alterar um título;
- Adicionar/Alterar/Remover vários conteúdos (texto, imagem, video ou link).
- Os conteúdos do tipo vídeo são suportados recorrendo à API do YouTube.

Moderação

Esta funcionalidade apenas faz sentido, quando os utilizadores da aplicação móvel, fazem comentários em conteúdos ou itens. Estes comentários podem ser do tipo texto ou imagens, e para poderem ser visualizados por outros utilizadores da aplicação móvel, têm de passar pela aprovação do proprietário (administrador ou cliente) ou de um gestor desse canal.

Relatórios

Ambas as aplicações permitem a visualização de um relatório, sendo que no caso da aplicação de gestão administrativa abrange todos os dados da plataforma, enquanto que na aplicação de gestão para clientes apenas abrange os canais referentes ao cliente ou os canais em que um gestor esteja inserido. Ambas as aplicações permitem a exportação dos dados no formato CSV.

5.2.4 Aplicação móvel

A aplicação móvel, para além da comunicação com a API de suporte, que pode ser vista na secção 5.2.2, tem algumas funcionalidades que permitem auxiliar o utilizador. As funcionalidades implementadas passam por: autenticação; listagem de canais; visualização dos portais; visualização e navegação das plantas; comentar conteúdos.

Autenticação

A autenticação nesta aplicação pode ser feita de forma anónima ou ser feita recorrendo à API do facebook.

Listagem de canais

Há duas maneiras de ver os canais que se encontram à volta do utilizador. Uma delas é ser visualizada em forma de lista e noutra dos casos é visualizada num mapa. Em ambos os casos o utilizador pode definir um filtro, que permite ver apenas os canais que estão num determinado raio ou ainda filtrar por nome os canais ou ainda juntar ambas as coisas.

Visualização dos portais

A visualização dos portais apenas surge quando um utilizador desta aplicação clica num canal. Ao entrar no canal é mostrado, ao utilizador, o portal público com os conteúdos que foram previamente introduzidos pelo proprietário ou gestor do canal, numa das aplicações de gestão *web*. Ainda é possível entrar no portal privado, sendo para isso necessário ler um código QR ou introduzir um código. Quando o utilizador entra no portal privado aparece uma lista com os conteúdos inseridos previamente pelo proprietário ou gestor do canal.

Visualização e navegação das plantas

Quando um utilizador se encontra no portal privado, para além da visualização de conteúdos tem ainda acesso à planta do local onde se encontra. A planta permite ao utilizador uma navegação de modo a perceber o que se encontra à sua volta – itens. Para a visualização dos itens, o utilizador clica num item, sendo-lhe mostrado uma lista com os conteúdos associados ao item. Estes conteúdos foram inseridos previamente numa das aplicações *web*, pelo proprietário ou gestor do canal.

Comentar conteúdos

Tanto nos portais como nos itens existem conteúdos, podendo estes ser do tipo texto, imagem, video ou link. Um utilizador pode comentar cada conteúdo, sendo o comentário enviado para aprovação e apenas ficando visível após a aprovação do proprietário ou gestor do canal. Os tipos de comentários possíveis são do tipo texto ou imagem. A imagem pode ser uma que já se encontre no dispositivo móvel ou então tirar uma foto no dado momento em que se quer fazer o comentário.

Capítulo 6 – Testes e Avaliação

Esta fase consiste na realização de um conjunto de testes às plataformas desenvolvidas, validando desta forma que estas cumprem os objectivos para o âmbito no qual se insere o projecto.

6.1. Ciclo de vida dos testes

O ciclo de vida dos testes foi composto por 4 fases: Planeamento, Especificação, Execução e Entrega.

Na fase de Planeamento foi elaborada uma estratégia para efectuar os testes. A estratégia definida, consistiu em seguir os seguintes passos sequencialmente:

- testes unitários;
- testes de integração;
- testes de sistema;
- testes de aceitação;
- testes de usabilidade;
- testes de performance;
- testes de escalabilidade.

Todo o processo decorrido ao longo destes testes, será explicado nas secções seguintes deste mesmo capítulo.

Após a fase de planeamento, foi efectuada uma elaboração dos testes necessários. Posteriormente os testes foram executados, caso houvesse *bugs* no final dos testes terem sido executados, eram enviados para correcção para depois voltarem a ser executados (este ciclo repete-se até o teste passar).

Por fim, na fase da entrega, obteve-se o produto final, no âmbito do estágio.

6.2. Testes unitários e Testes de integração

Foi tomada a decisão de não serem feitos testes unitários, visto ser necessário algum tempo adicional para os implementar, sendo que em alternativa, foram efectuadas reuniões semanais, onde era mostrado o que estava implementado, como tinha sido implementado e testadas as funcionalidades.

Cada vez que eram adicionadas novas *features* à aplicação, eram feitos testes de modo a ver se a plataforma continuava a funcionar correctamente. Caso fossem encontradas falhas provenientes da integração, procedia-se à correcção do(s) problema(s).

6.3. Testes de sistema

O objectivo deste tipo de testes é executar o produto do ponto de vista do utilizador final, percorrendo assim todas as funcionalidades de modo a se perceber se há falhas em relação aos objectivos propostos.

O modelo de apresentação destes testes pode ser visto na tabela 14, onde é especificada qual a funcionalidade que se está a testar, qual a plataforma onde se está a testar, quais as acções efectuadas pelo utilizador, qual o resultado esperado, bem como o resultado obtido. Se no final o resultado obtido for igual ao esperado, então o resultado do teste dá “Sucesso”, senão dá “Falhou”.

ID Teste: XX
Funcionalidade:
Plataforma: Plataforma Y
Acções:
Resultado esperado:
Resultado obtido:
Resultado do teste: Sucesso / Falhou

Tabela 14: Exemplo de tabela para apresentação dos testes de sistema

No anexo E.1 podem ser vistos todos os testes efectuados às diferentes plataformas. De referir que da primeira vez que todos os testes foram efectuados (92), passaram com sucesso 76, como pode ser visto no anexo E.1.1.

Os casos de teste que falharam foram revistos, tendo sido todos corrigidos. A repetição dos casos de teste pode ser visto no anexo E.1.2.

6.4. Testes de aceitação

Este tipo de testes são usados para validar e verificar se o produto desenvolvido está de acordo com o que foi estabelecido nos requisitos. Para tal foi efectuada uma reunião com o orientador Carlos Ramos, a fim de verificar que todos os requisitos se encontravam implementados e o comportamento dos mesmos era o esperado, de acordo com o anteriormente especificado na sub-secção 4.2, deste mesmo documento.

O resultado desta reunião pode ser consultado na tabela no anexo E.2.

De referir ainda que o resultado referente aos requisitos funcionais foi de 94 implementados de 116, sendo que a divisão por aplicação é a seguinte:

- Aplicação de gestão administrativa:
 - *Must* – 37/37;
 - *Should* – 1/4;
 - *Nice to have* – 0/9;
- Aplicação de gestão para clientes:
 - *Must* – 35/35;
 - *Should* – 2/5;
 - *Nice to have* – 1/4;
- Aplicação móvel:
 - *Must* – 18/18;
 - *Should* – 0/2;
 - *Nice to have* – 0/2.

Em relação aos requisitos não funcionais todos os *musts* foram validados com sucesso.

6.5. Testes de usabilidade

Para qualquer software desenvolvido, torna-se indispensável fazer uma avaliação do mesmo do ponto de vista do utilizador. Pretende-se então obter uma indicação geral do nível de usabilidade deste sistema, não só com vista à melhoria de certos aspectos mas também com a finalidade de verificar a facilidade de um utilizador compreender e manipular o software.

Para o efeito, o questionário é baseado na *System Usability Scale* (SUS), criado por *John Brooke* em 1986 [67], foi traduzido, adaptado e aplicado a 20 pessoas com idades compreendidas entre os 18 e os 60 anos, dos quais 10 dos inquiridos são da área de informática e os restantes 50% estão ligados a outras áreas de conhecimento. Ainda de referir que nenhum utilizador conhecia a plataforma em causa e que todos os testes efectuados foram em ambiente informal, para deixar os utilizadores mais à vontade.

O cálculo da pontuação do teste é efectuado da seguinte forma [68]:

- cada questão tem uma contribuição variável entre 0 (“discordo totalmente”) e 4 (“concordo totalmente”);
- para as respostas ímpares, a pontuação refere-se ao número na escala menos 1;
- para as respostas pares, a pontuação obtida diz respeito a 5 menos a posição na escala.

Finalmente, todas as contribuições individuais são somadas e multiplicadas por 2.5. A pontuação final estará compreendida entre 0 e 100.

No anexo E.3, referente aos testes de usabilidade, encontra-se o resultado dos testes efectuados.

Os resultados médios obtidos nos testes foram:

- aplicação *web* (gestão administrativa e gestão para clientes):
 - grupo área de informática – 85,75
 - grupo de outras áreas de conhecimento – 76,25
- aplicação móvel:
 - grupo área de informática – 78
 - grupo de outras áreas de conhecimento – 69,5

“Based on research, a SUS score above a 68 would be considered above average and anything below 68 is below average, however the best way to interpret your results involves “normalizing” the scores to produce a percentile ranking.” [67]

Posto isto, podemos considerar que a usabilidade das aplicações foi verificada e aceite pela empresa, visto todas as pontuações terem dado acima da média dos 68 pontos.

6.6. Testes de performance

Este tipo de testes dizem respeito ao tempo que uma página demora a ser carregada. Segundo Steve Souders, pelo menos 80% do tempo de exibição de uma página *web* acontece depois do *download* do HTML [69]. Desta forma, identifica quais as razões que levam uma página a demorar a ser carregada, criando para tal uma série de regras, ver tabela 15, que visam melhorar este aspecto.

Regra	Descrição da regra
1	Make Fewer HTTP Requests
2	Use a Content Delivery Network (CDN)
3	Add an Expires Header
4	Gzip Components
5	Put Stylesheets at the Top
6	Put Scripts at the bottom
7	Avoid CSS Expressions
8	Make JavaScript and CSS External
9	Reduce DNS Lookups
10	Minify JavaScript
11	Avoid Redirects
12	Remove Duplicate Scripts
13	Configure ETags
14	Make AJAX Cacheable

Tabela 15: Regras do High Performance Websites do Steve Souders [70]

Algumas regras passam apenas por boas práticas, por exemplo: colocar os ficheiros CSS no início da página, os ficheiros JS no final. Contudo, outras regras implicam algumas modificações no servidor Apache, como é o caso da regra *Add an Expires Header*, onde a configuração pode ser vista na figura 24. Esta, permite que os ficheiros CSS e JS possam ficar em cache no *browser*, havendo desta forma melhorias significativas aquando de um segundo acesso a uma página.

```

<IfModule mod_expires.c>
  ExpiresActive on
  ExpiresDefault "access plus 10 days"
  ExpiresByType text/css "access plus 1 month"
  ExpiresByType text/plain "access plus 1 month"
  ExpiresByType image/gif "access plus 1 month"
  ExpiresByType image/png "access plus 1 month"
  ExpiresByType image/jpeg "access plus 1 month"
  ExpiresByType application/x-javascript "access plus 1 month"
  ExpiresByType application/javascript "access plus 1 month"
  ExpiresByType application/x-icon "access plus 1 year"
</IfModule>

```

Figura 24: Código utilizado no servidor Apache para configurar Expires Header

A concordância com as regras enunciadas na tabela 15 permitiram melhorias de *performance* significativas, como podemos ver nas tabelas 16 e 17. Por forma a validar as melhorias das alterações introduzidas, foi usada a ferramenta Yslow [71]. Com esta, foram realizados testes antes e depois de introduzidas as alterações, havendo notórias melhorias acentuando-se mais num segundo acesso, na medida que fica em *cache* os elementos CSS e JS, evitando assim pedidos desnecessários.

As alterações introduzidas permitiram mudar a classificação atribuída de B(83) para A(95), podendo o resultado dos testes ser consultados no anexo E.4.

	Sem Cache (KB)	Com Cache (KB)
Antes de aplicadas as regras	561.3	530.2
Depois de aplicadas as regras	315.2	0.3
Ganhos	43.8%	99.9%

Tabela 16: Comparação de ganhos na transferência de dados

	Sem Cache (Pedidos)	Com Cache (Pedidos)
Antes de aplicadas as regras	13	11
Depois de aplicadas as regras	12	2
Ganhos	7.7%	81.8%

Tabela 17: Comparação de ganhos no número de pedidos ao servidor

6.7. Testes de escalabilidade

Em engenharia de *software* a escalabilidade é uma característica bastante importante, visto indicar a capacidade que uma aplicação tem para reagir e acompanhar uma variação no volume de acessos e volume de informação processada. O cenário de validação e as métricas a atingir encontram-se na tabela 18 e 19, respectivamente.

Cenário de Validação	
Servidor	VPS CentOS 6.5, 2 <i>cores</i> com 2 Gb RAM
Cliente	MacBook Pro 2.4GHz Intel Core i5
Website	http://neala.enso-origins.com/ic
Ligação (internet)	29.63Mbps de <i>download</i> e 3.34Mbps de <i>upload</i> com <i>ping</i> de 27ms
Notas	O teste de ligação foi efectuado no <i>website</i> Speedtest [72].

Tabela 18: Cenário de validação

Métricas a atingir
Estudar o limite de clientes que a aplicação suporta, num tempo de resposta inferior a 1seg.
A plataforma deverá apresentar as mesmas características quer estejam 10 ou 100 utilizadores a usar a plataforma.

Tabela 19: Métricas a atingir

Em relação às métricas a atingir, o tempo de resposta não pode ser superior a 1 segundo, pois segundo Robert B. Miller [73], um segundo é o tempo limite para que no decorrer de interações humano computador, o utilizador sinta que está a interagir livremente com as informações. Ainda no mesmo estudo é realçado que o tempo de resposta pode ir até dois segundos, sem que o utilizador perca a atenção sobre o sistema.

Para avaliar a escalabilidade, foi usada a ferramenta *ApacheBench* [74] do Apache. Os testes de escalabilidade permitiram aferir qual o limite de clientes possível, no servidor disponibilizado pela empresa ENSO para a implementação do projecto.

A realização do teste consistiu na implementação de um *script* que permitiu variar o número de clientes em paralelo, bem como o número de pedidos que estes realizavam à aplicação. Um exemplo do teste, pode ser consultado na figura 25, correspondendo as *flags* -n e -c ao número de pedidos a realizar e à concorrência no número de pedidos a fazer ao servidor, respectivamente.

```
ab -n 100 -c 10 https://neala.enso-origins.com/ic
```

Figura 25: Exemplo de teste com a ferramenta *ApacheBench*

Através dos vários testes efectuados, foi possível verificar que a aplicação efectivamente cumpre o objectivo de suportar, pelo menos 100 utilizadores com uma responsividade inferior a 1 segundo na actual plataforma de *hardware*. Os testes podem ser consultados detalhadamente no anexo E.5.

Capítulo 7 – Conclusões e Trabalho Futuro

O presente capítulo visa uma reflexão do trabalho realizado ao longo deste estágio, bem como uma análise ao futuro da plataforma implementada.

7.1. Conclusões

Este documento representa o esforço, empenho e dedicação ao longo de um ano lectivo.

Durante todo o período de estágio, que decorreu na empresa Enso Origins, onde em relação ao meu trabalho em particular estive em contacto com duas equipas diferentes. Por um lado, tive o apoio do meu orientador de estágio (Eng. Carlos Ramos) e de outro elemento residente da equipa da Enso (Nuno Fidalgo) para a especificação do sistema, por outro lado, mas com o mesmo âmbito, apoiar na especificação, estive em contacto com a equipa residente em Moçambique (Marco Ferreira e Pedro Loureiro).

Ao longo do documento é visível, de forma clara, o processo que levou à construção das diferentes aplicações. Este consistiu na definição do planeamento, o estudo do estado da arte, a especificação do sistema, a implementação, testes e validação.

Planeamento

O plano de trabalho, presente no capítulo 2, permitiu fazer um maior controlo de todas as tarefas do estágio, conseguindo assim uma estimativa para a especificação do sistema. Houve algumas alterações ao planeamento, sendo que a que afectou mais foi a decisão de adiar a defesa para Setembro. Esta alteração apenas surgiu, tendo em conta que o autor, devido a razões de natureza pessoal, ainda se encontrava a trabalhar em *part-time* e que devido a tal, levou a um atraso na implementação.

Estado da arte

Através do estado da arte, presente no capítulo 3, foi apresentado um estudo entre aplicações concorrentes e a plataforma do âmbito do estágio (secção 3.1), dando uma noção da existência de várias aplicações concorrentes, com objectivos bem distintos da do âmbito do estágio mas que permitiu aproveitar o conhecimento que se obteve para apoiar e moldar as soluções tecnológicas usadas. No seguimento, é apresentado um estudo e a escolha das tecnologias (secção 3.2) usadas no âmbito do estágio.

Especificação do sistema

A especificação do sistema, presente no capítulo 4, engloba uma breve introdução ao sistema, o levantamento de requisitos, o diagrama de casos de uso, os *mockups* e a arquitectura do sistema (diagrama físico, diagramas de estados e diagrama entidade-relação).

O levantamento de requisitos (secção 4.1) permitiu especificar as funcionalidades e características que eram desejadas para o sistema. Por forma a priorizar os diferentes requisitos, foi usada a seguinte nomenclatura:

- *Must*, quando o requisito é essencial para o funcionamento do sistema;
- *Should*, quando o requisito não é essencial ao funcionamento do sistema, porém deveria ser implementado;
- *Nice to have*, quando era desejável que o requisito fosse implementado, no entanto não é de todo essencial para o sistema.

O diagrama de casos de uso (secção 4.2) é importante para dar uma visão global de todas as acções que os actores podem executar no sistema, tendo sido criado através do levantamento de requisitos.

Os *mockups* (secção 4.3) são importantes para que se possa visualizar de que maneira as acções que os utilizadores podem executar (diagrama de casos de uso) se conciliam com as diferentes aplicações implementadas.

A arquitectura do sistema (secção 4.4) está dividida em três sub-secções:

- o diagrama físico (sub-secção 4.4.1) mostrando-nos quais as componentes físicas do nosso sistema, de que maneira comunicam, quais as tecnologias que serão usadas e como estão as plataformas interligadas;
- os diagramas de estados (sub-secção 4.4.2) que descrevem o comportamento dos objectos;
- o diagrama ER (sub-secção 4.4.3) que nos mostra como se relacionam as tabelas, quais os campos que contém e qual a cardinalidade das relações existentes.

Implementação

A metodologia de desenvolvimento seguida foi baseada em *scrum*. Esta permitiu fazer uma divisão das tarefas referentes aos requisitos necessários à plataforma, fazendo com que fosse possível a implementação de cada uma das aplicações (aplicação gestão administrativa, aplicação gestão para clientes e aplicação móvel), da API de suporte e da base de dados.

Foi criada uma livraria no âmbito do estágio e que está a ser utilizada na empresa, para adicionar um sistema de *logs* a qualquer aplicação desenvolvida.

Testes e validação

Antes de finalizar o documento, foram efectuados testes, presentes no capítulo 6, e procedeu-se à validação do cumprimento da solução com os requisitos idealizados.

A fase de testes foi composta por quatro fases (planeamento, especificação, execução e entrega). A estratégia passou por fazer revisões de código e testes às funcionalidades implementadas, ao invés de testes unitários e testes de integração.

Em suma

Em modo de conclusão, o trabalho desenvolvido no âmbito do estágio, encontra-se de acordo com o expectável, tendo havido algumas alterações ao planeamento que não são boas, mas que levou ao aperfeiçoamento de tudo o que foi efectuado.

Dados os resultados das validações dos requisitos, concluiu-se que foram atingidos com sucesso os objectivos definidos.

De referir que o autor nunca tinha planeado e implementado uma plataforma com esta dimensão, sendo que tudo o que aprendeu ou aprofundou deve-se ao acompanhamento de ambos os orientadores, Doutor Paulo Simões e Eng. Carlos Ramos, por toda a ajuda quer na documentação, quer nos ensinamentos que levaram à implementação da presente solução, e que sem ambos o autor teria tido muito mais dificuldade em alcançar os objectivos propostos.

7.2. Trabalho Futuro

Algum trabalho futuro pode valorizar ainda mais a plataforma, trabalho futuro que passa por:

- Partilha de conteúdos em redes sociais, como por exemplo no Facebook;
- Identificação das zonas mais visitadas recorrendo a modelos visuais, como por exemplo *heatmaps*;
- Aumentar o tipo de conteúdos multimédia suportados:
 - Adicionar a possibilidade de ter áudio, seja recorrendo a API's (exemplo: SoundCloud), seja suportando áudio gravado no momento (aplicação móvel) ou até a possibilidade de carregar áudio que se encontre no dispositivo onde corre a aplicação cliente;
 - Alargar o suporte para vídeo gravado no momento (aplicação móvel) ou a possibilidade de carregar vídeos que se encontrem no dispositivo onde corre a aplicação cliente.
- Reconhecimento de imagens para localização de objectos;
- Recorrer ao uso a tecnologias de NFC e RFID para identificação de objectos;
- Recorrer a sistemas de triangulação de AP's e antenas GSM, por forma a melhorar a geolocalização indoor e outdoor;
- Implementação da aplicação móvel em outros sistemas operativos.

Referências

- [1] Enso Origins. <http://www.enso-origins.com>, Janeiro 2014.
- [2] Google Glass. <http://www.google.com/glass>, Agosto 2014.
- [3] Google Glass review. <http://www.techradar.com/reviews/gadgets/google-glass-1152283/review>, Agosto 2014.
- [4] “Now Anyone Can Buy Google Glass”. <http://time.com/98945/google-glass-on-sale-now>, Agosto 2014.
- [5] Google Places. www.google.com/business/placesforbusiness, Novembro 2013.
- [6] Funcionalidades gratuitas Google Places. <http://www.google.com/business/placesforbusiness/free-features>, Novembro 2013.
- [7] Foursquare. <https://foursquare.com>, Novembro 2013.
- [8] About Foursquare. <https://foursquare.com/about>, Novembro 2013.
- [9] Waze. <https://www.waze.com>, Novembro 2013.
- [10] About Waze. <https://www.waze.com/about>, Novembro 2013.
- [11] e-Museum. <https://www.facebook.com/emuseumpm>, Novembro 2013.
- [12] Convento de Santa Clara-a-Velha. <http://santaclaraavelha.drcc.pt>, Novembro 2013.
- [13] TripAdvisor. <http://www.tripadvisor.com>, Julho 2014.
- [14] About TripAdvisor. http://www.tripadvisor.com/PressCenter-c6-About_Us.html, Julho 2014.
- [15] CentOS. www.centos.org, Julho 2014.
- [16] Apache. <http://www.apache.org>, Novembro 2013.
- [17] W3Techs “Usage of web servers for websites”. <http://w3techs.com>, Novembro 2013.
- [18] DB-engines Ranking definition. http://db-engines.com/en/ranking_definition, Julho 2014.
- [19] Google. <http://www.google.com>, Julho 2014.
- [20] Bing. <http://www.bing.com>, Julho 2014.
- [21] Google Trends. <http://www.google.com/trends>, Julho 2014.
- [22] Stackoverflow. <http://stackoverflow.com>, Julho 2014.
- [23] DBA Stack Exchange. <http://dba.stackexchange.com>, Julho 2014.
- [24] Indeed. <http://www.indeed.pt>, Julho 2014.
- [25] Simply Hired. <http://www.simplyhired.pt>, Julho 2014.
- [26] LinkedIn. <https://www.linkedin.com>, Julho 2014.
- [27] Twitter. <https://twitter.com>, Julho 2014.
- [28] DB-engines Ranking. <http://db-engines.com/en/ranking>, Novembro 2013.
- [29] SQLite. <http://www.sqlite.org>, Novembro 2013.
- [30] MySQL. <http://www.mysql.com>, Novembro 2013.
- [31] PostgreSQL. <http://www.postgresql.org>, Novembro 2013.
- [32] S. Riggs; H. Krosing. “PostgreSQL 9 – Administration Cookbook”. PACKT Publishing, 2010.

- [33] “Open Source Initiative OSI - ‘The BSD License’”. <http://www.opensource.org/licenses/bsd-license.php>, Novembro 2013.
- [34] “Open Source Initiative OSI - ‘The MIT License’”. <http://opensource.org/licenses/MIT>, Novembro 2013.
- [35] Hackmageddon – Cyber Attacks Statistics. <http://hackmageddon.com/category/security/cyber-attacks-statistics>, Julho 2014.
- [36] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinsteins and Charles E. Youman. “Role-Based Access Control Models”. <http://csrc.nist.gov/rbac/sandhu96.pdf>, Julho 2014.
- [37] PHP. <http://www.php.net>, Novembro 2013.
- [38] David Sklar. “Learning PHP5”. O'Reilly Media, Junho 2004.
- [39] Adam Trachtenberg. “PHP Cookbook”. O'Reilly Media, 2nd Edition 2002.
- [40] W3Techs. “Programming Languages Analysis by W3techs”. http://w3techs.com/technologies/overview/programming_language/all, Novembro 2013.
- [41] Bruce Lawson and Remy Sharp. “Introducing HTML5”. 2nd Edition, 2011.
- [42] Peter Gasston. “The Book of CSS3: A Developer's Guide to the Future of Web Design”. 2011.
- [43] David Flanagan. “JavaScript: The Definite Guide”. O'Reilly Media. 6th Edition, 2011.
- [44] Roy T. Fielding. <http://www.ics.uci.edu/~fielding>, Novembro 2013.
- [45] Tese de doutoramento de Roy T. Fielding. “Architectural Styles and the Design of Network -base Softwares”. <http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>, Novembro 2013.
- [46] Client-Server Communication with REST. <http://www.prideparrot.com/Source-Codes/Images/REST.png>, Agosto 2014.
- [47] Codeigniter. <http://www.ellislab.com/codeigniter>, Dezembro 2013.
- [48] EllisLab Inc, <http://ellislab.com>, Dezembro 2013.
- [49] Ademir Cristiano Gabardo. “PHP e MVC com codeigniter”. Novatec Editora Lda, 2012.
- [50] MVC com codeigniter. http://www.packtpub.com/sites/default/files/9781849694704_Chapter-03.pdf, Dezembro 2013.
- [51] Google Maps Developers. <https://developers.google.com/maps>, Julho 2014.
- [52] MapBox Developers. <https://www.mapbox.com/developers>, Julho 2014.
- [53] Leaflet. <http://leafletjs.com>, Julho 2014.
- [54] Canvas – basic support. <http://caniuse.com/canvas>, Julho 2014.
- [55] Leaflet – features. <http://leafletjs.com/features.html>, Julho 2014.
- [56] Facebook Developers. <https://developers.facebook.com>, Novembro 2013
- [57] Google Developers. <https://developers.google.com>, Novembro 2013.
- [58] Axel Küpper. “Location-Based Services: Fundamentals and operation”, John Wiley & Sons, 2005.
- [59] PostGIS. <http://postgis.net>, Julho 2014.
- [60] Open Geospatial Consortium. <http://www.opengeospatial.org>, Julho 2014.
- [61] “How to setup and user PostGis with Postgres Plus Standard Server”. <http://www.enterprisedb.com/resources-community/tutorials-quickstarts/all-platforms/how-setup-and-use-postgis-postgres-plus-stan>, Julho 2014.
- [62] Android. <http://www.android.com>, Novembro 2013.

- [63] Apple iOS. <http://www.apple.com/pt/ios>, Novembro 2013.
- [64] Android Developers. “Platforms Versions”.
<http://developer.android.com/about/dashboards/index.html>, Dezembro 2013.
- [65] Ken Schwaber and Mike Beedle. “Agile Software Development with Scrum”. 2002.
- [66] Ken Schwaber e Jeff Sutherland. “O Guia do Scrum”,
https://www.scrum.org/Portals/0/Documents/Scrum%20Guides/Scrum%20Guide%20-%20Portuguese_European.pdf, Julho 2014.
- [67] John Brooke. System Usability Scale (SUS),
<http://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>, Agosto 2014.
- [68] John Brooke. “SUS – A quick and dirty usability scale”,
http://cui.unige.ch/isi/icle-wiki/_media/ipm:test-sus-chapt.pdf, Agosto 2014.
- [69] Steve Souders. “High Performance Web Sites”. O'REILLY, 2007.
- [70] “High Performance Web Sites”, <http://stevesouders.com/hpws/rules.php>, Agosto 2014.
- [71] Yahoo. <http://developer.yahoo.com/yslow>, Agosto 2014.
- [72] SpeedTest. <http://www.speedtest.net>, Agosto 2014.
- [73] Robert B. Miller. “Response time in man-computer conversational transactions”,
<http://theixdlibrary.com/pdf/Miller1968.pdf>, Agosto 2014.
- [74] ApacheBench. <https://httpd.apache.org/docs/2.2/programs/ab.html>, Agosto 2014.

Anexos

A – Diagrama de Casos de Uso

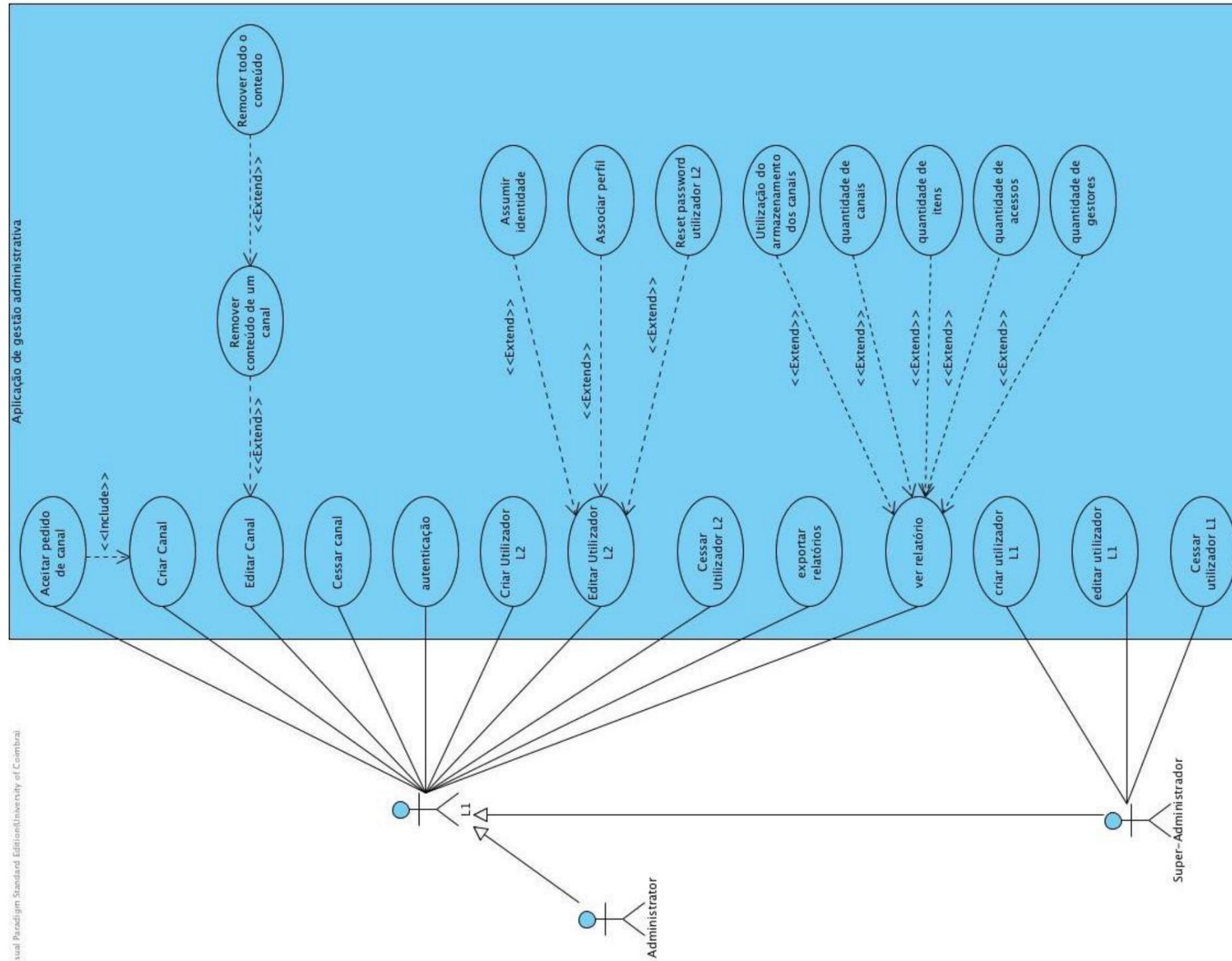


Figura 26: Diagrama de Casos de Uso - Aplicação de Gestão Administrativa

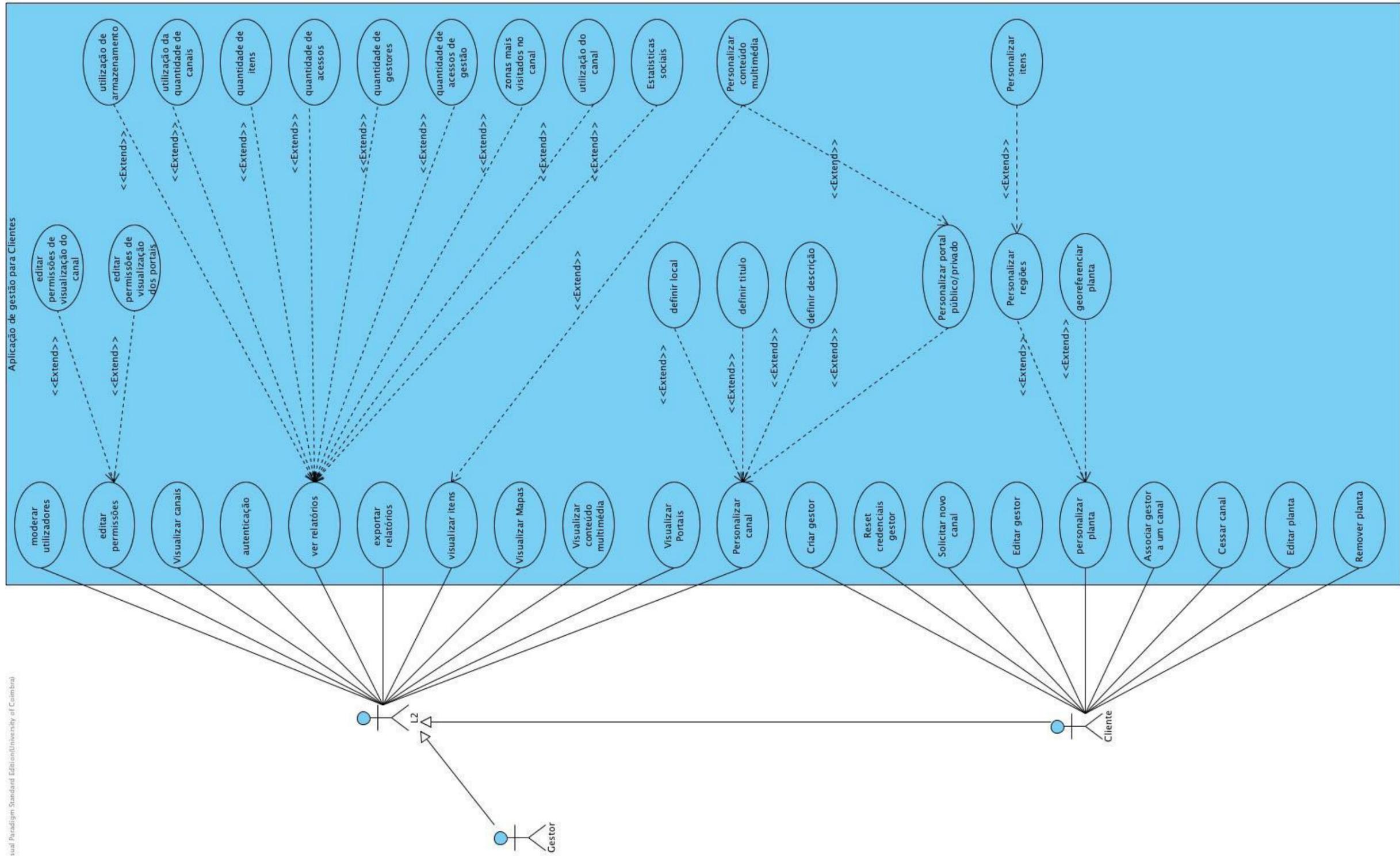


Figura 27: Diagrama de Casos de Uso - Aplicação de Gestão para Clientes

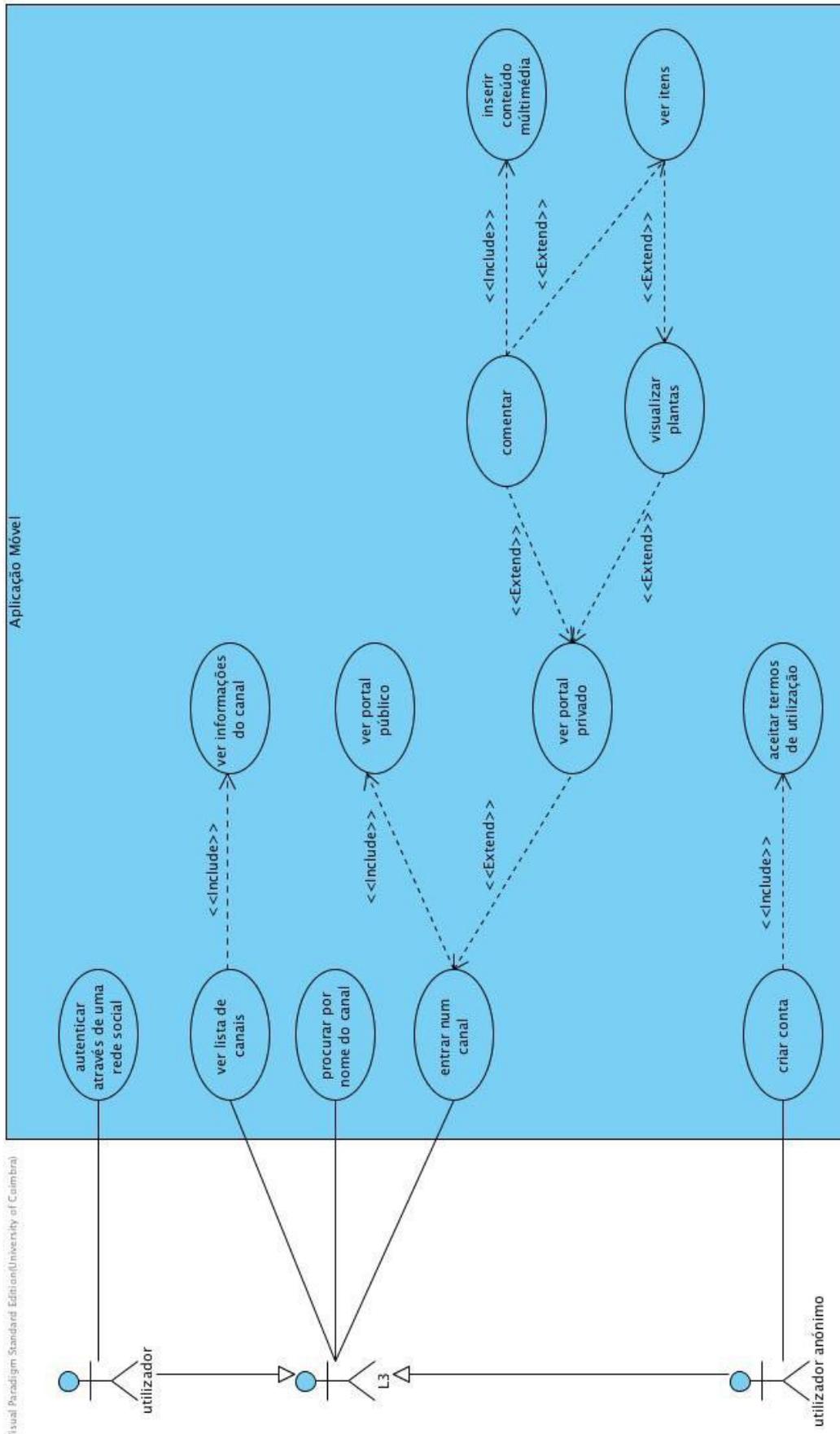


Figura 28: Diagrama de Casos de Uso – Aplicação Móvel

B – Diagrama de Flow & Mockups

B.1 – Diagrama de Flow



Figura 29: Diagrama de flow da plataforma web (Administração e Cliente)

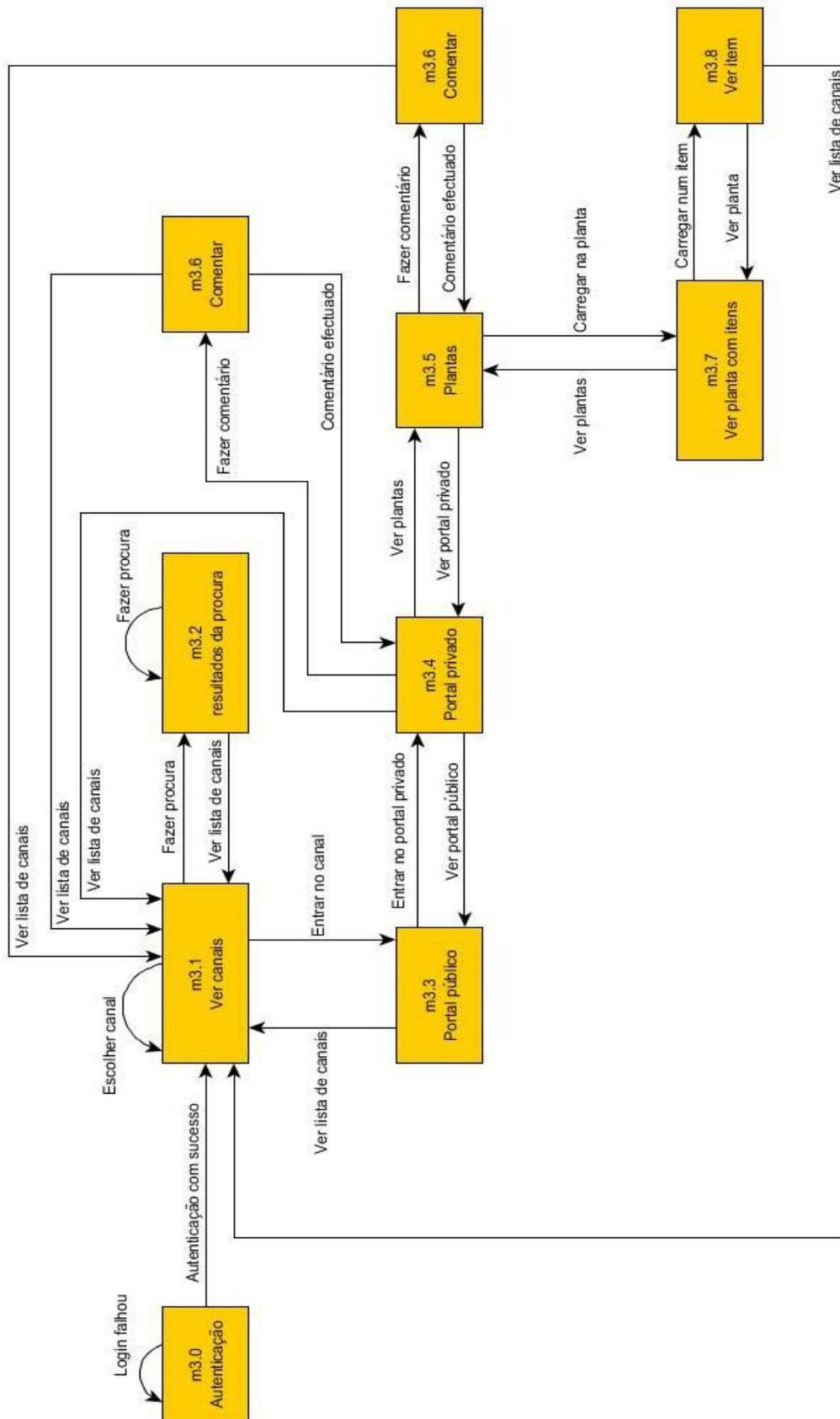


Figura 30: Diagrama de Flow da aplicação móvel

B.2 – Mockups

Módulo 1 – Aplicação de Gestão Administrativa

Código	Descrição
M1.00	Autenticação
M1.01	Procurar/criar novo canal
M1.02	Resultados da procura de canais
M1.03	Editar/cessar canal/editar conteúdos/exportar relatórios
M1.04	Apagar um conteúdo/apagar todos os conteúdos do canal
M1.05	Aceitar pedidos de criação de canal
M1.06	Procurar/criar novo utilizador L2
M1.07	Resultados da procura de utilizadores
M1.08	Editar/cessar conta utilizador/reset credenciais/associar perfil/assumir identidade
M1.09	Relatório de sistema
M1.10	Procurar/criar utilizadores L1,L2
M1.11	Editar/cessar conta utilizador/reset credenciais/associar perfil/assumir identidade

Tabela 20: Lista de mockups da Aplicação de Gestão Administrativa

Autenticação

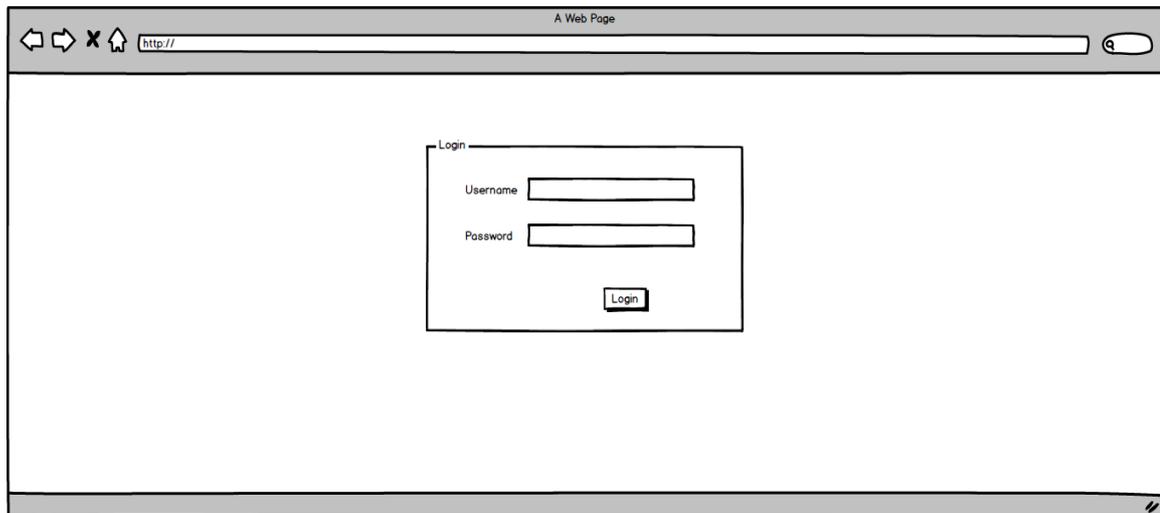


Figura 31: Mockup M1.00

Procurar/criar novo canal

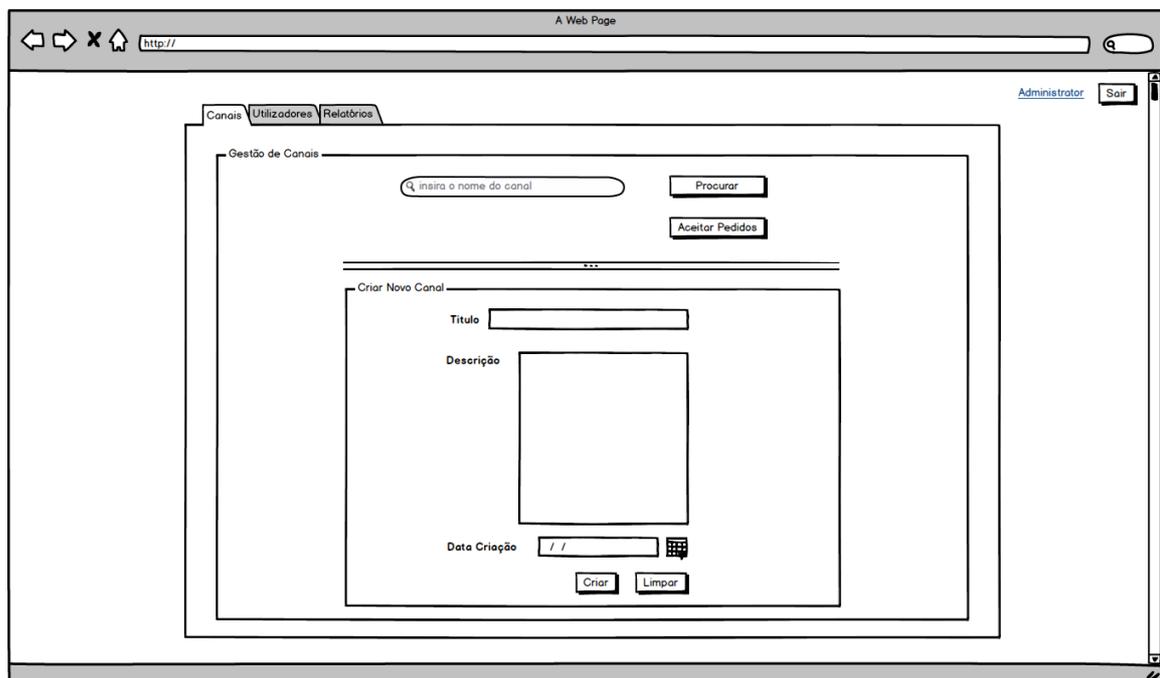


Figura 32: Mockup M1.01

Resultados da procura de canais

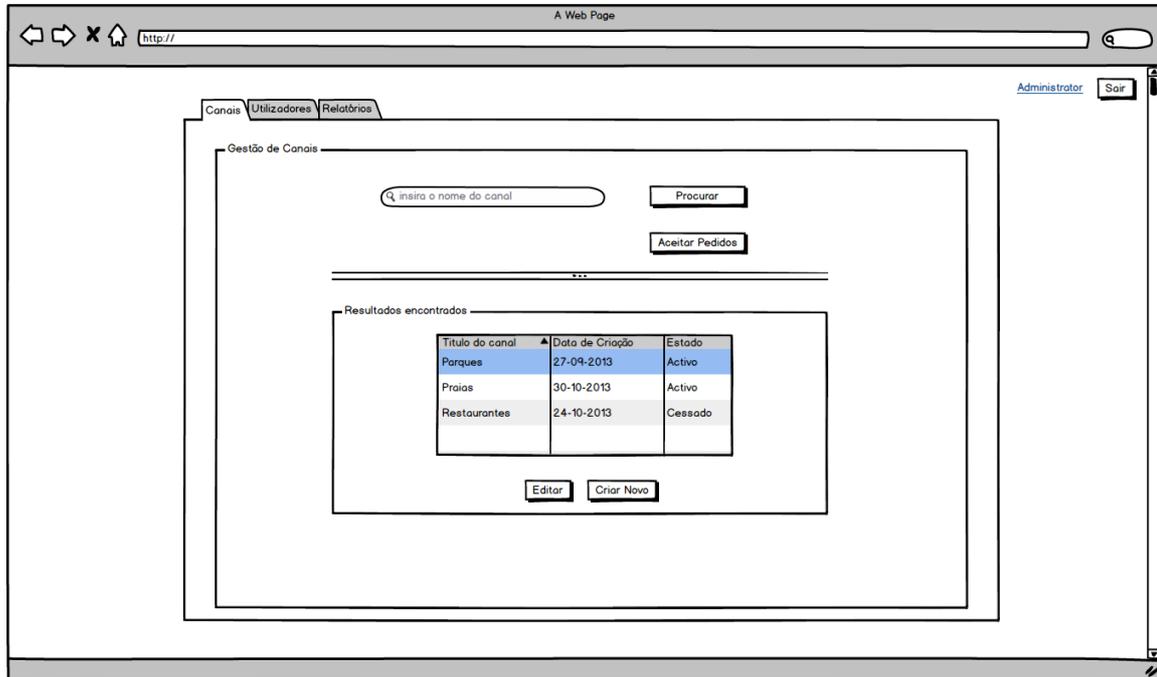


Figura 33: Mockup M1.02

Personalizar canal

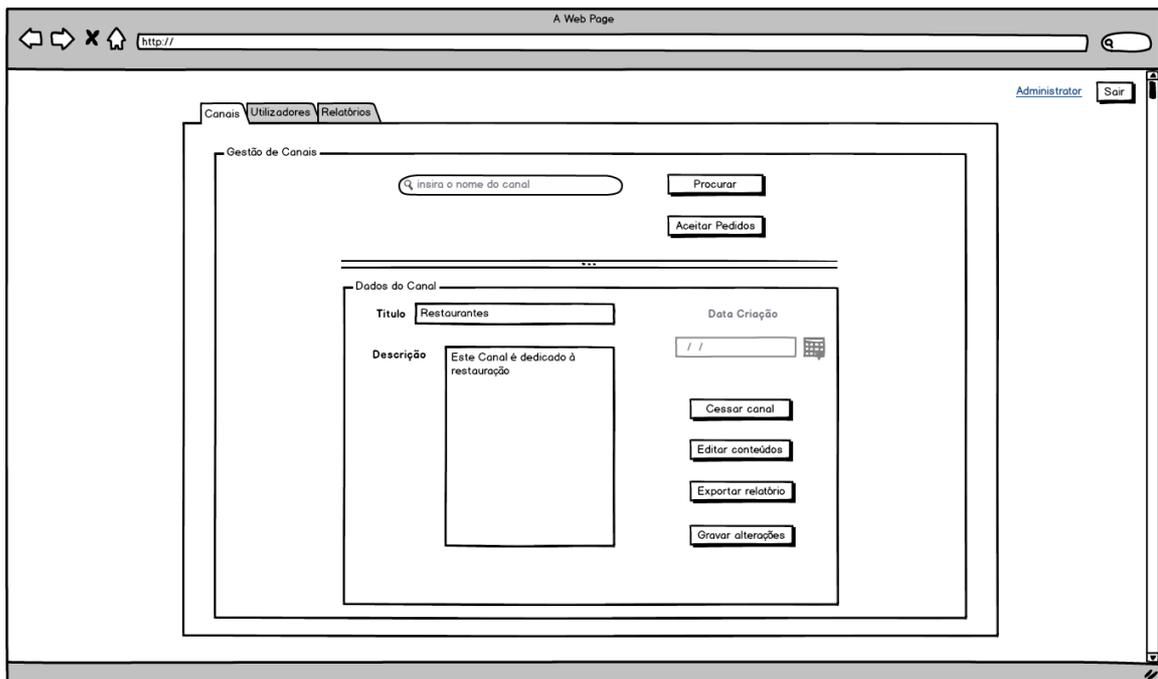


Figura 34: Mockup M1.03

Apagar conteúdos de um canal

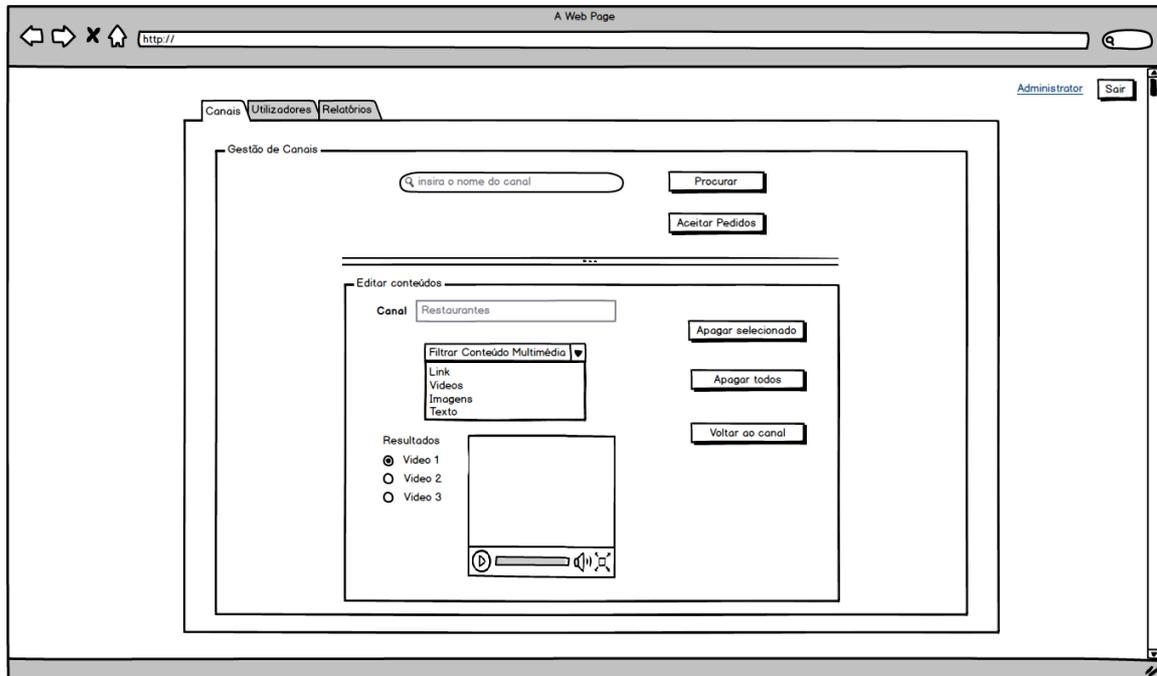


Figura 35: Mockup M1.04

Aceitar pedidos para a criação de novos canais

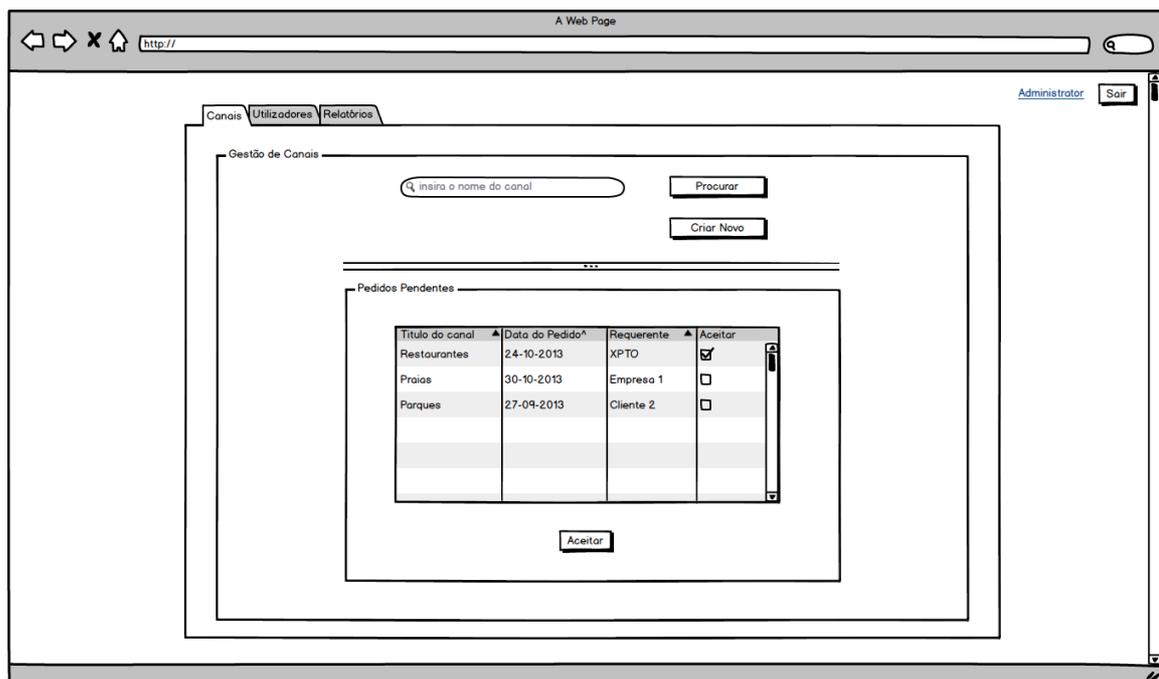


Figura 36: Mockup M1.05

Procurar/criar novo utilizador L2

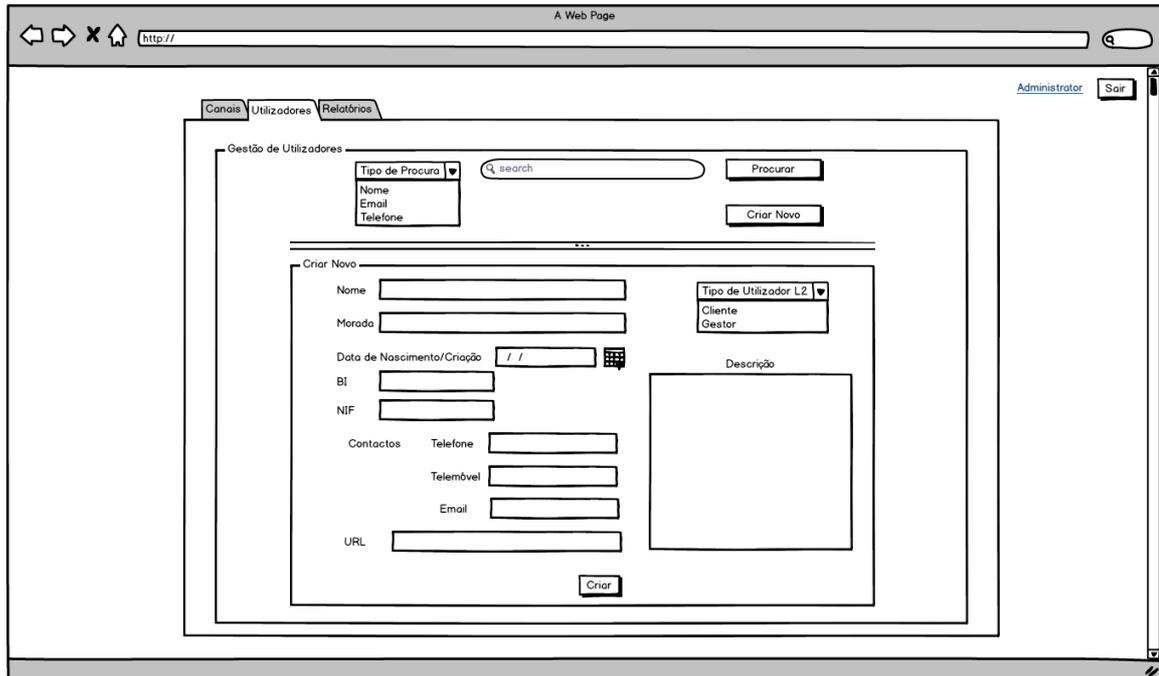


Figura 37: Mockup M1.06

Resultados da procura de utilizadores

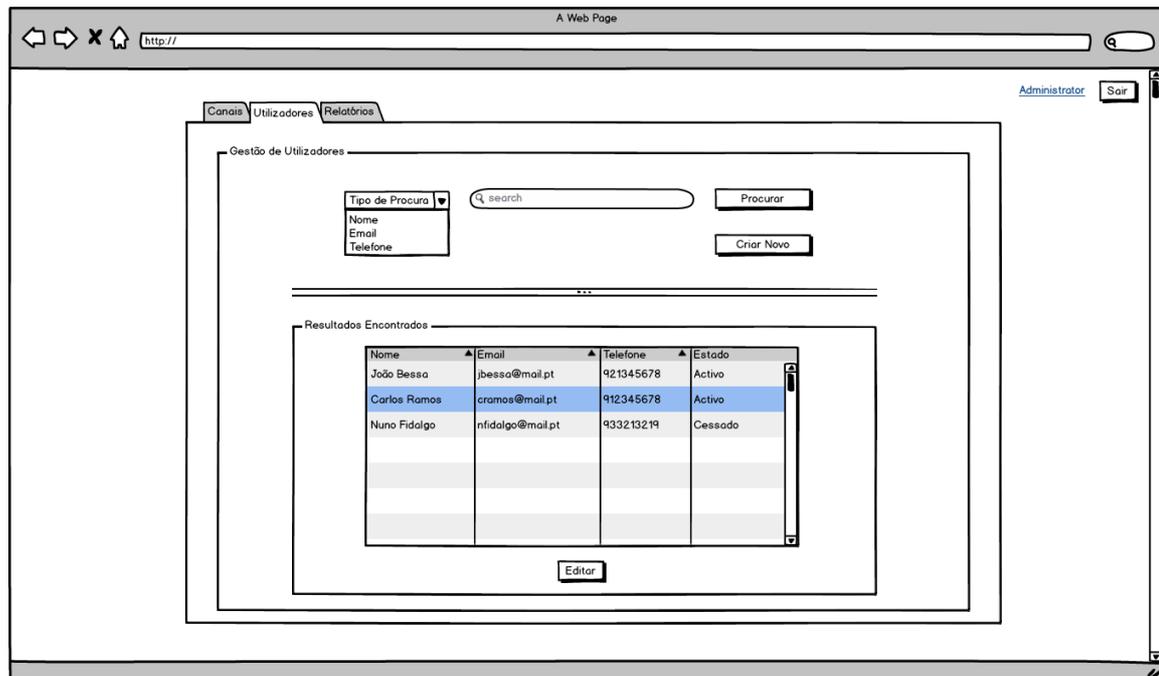


Figura 38: Mockup M1.07

Personalizar dados dos utilizadores

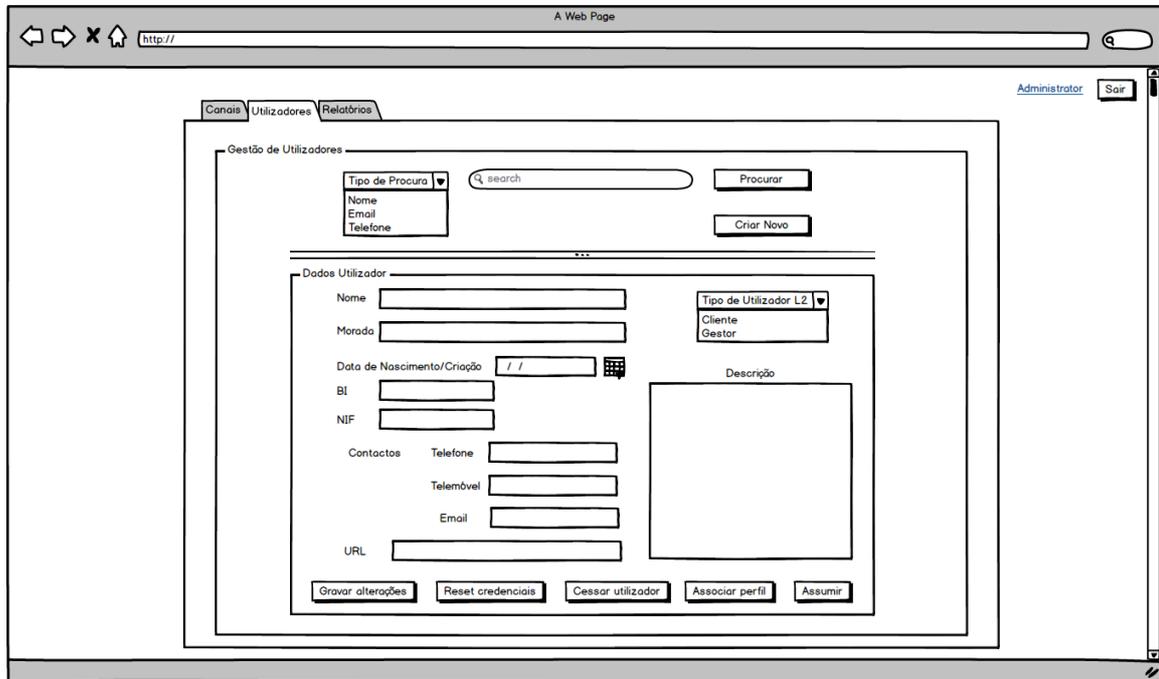


Figura 39: Mockup M1.08

Relatório de sistema

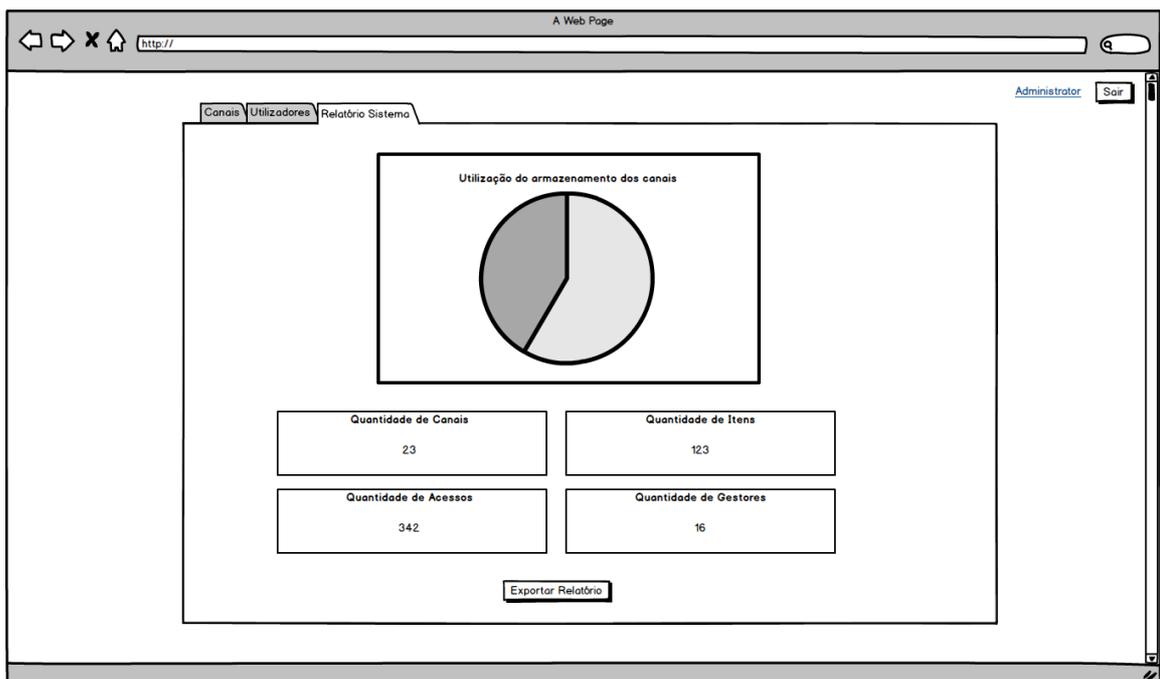


Figura 40: Mockup M1.09

Procurar/criar novo utilizador L1/L2

The mockup shows a web browser window with a navigation bar containing 'Canais', 'Utilizadores', and 'Relatórios'. The main content area is titled 'Gestão de Utilizadores'. It features a search section with a dropdown menu for 'Tipo de Procura' (options: Nome, Email, Telefone), a search input field, and 'Procurar' and 'Criar Novo' buttons. Below this is the 'Criar Novo' form, which includes fields for 'Nome', 'Morada', 'Data de Nascimento/Criação' (with a date picker), 'BI', 'NIF', 'Contactos' (with sub-fields for 'Telefone', 'Telemóvel', and 'Email'), and 'URL'. A 'Tipo de Utilizador' dropdown menu is set to 'L1 - Super-Administrador' (other options: L1 - Administrador, L2 - Cliente, L2 - Gestor). A large 'Descrição' text area is also present. A 'Criar' button is located at the bottom right of the form.

Figura 41: Mockup M1.10

Personalizar dados de utilizadores L1/L2

The mockup shows the same web browser window as Figure 41, but the main content area is titled 'Dados Utilizador'. The search section remains the same. The form fields for 'Nome', 'Morada', 'Data de Nascimento/Criação', 'BI', 'NIF', 'Contactos', and 'URL' are identical to the 'Criar Novo' form. The 'Tipo de Utilizador' dropdown menu is also the same. The 'Descrição' text area is present. At the bottom of the form, there are five buttons: 'Gravar alterações', 'Reset credenciais', 'Cessar utilizador', 'Associar perfil', and 'Assumir'.

Figura 42: Mockup M1.11

Módulo 2 – Aplicação de Gestão para Clientes

Código	Descrição
M2.01	Procurar/solicitar novo canal
M2.02	Resultados da procura
M2.03	Cessar canal/adicionar gestor a um canal/permisões do canal/editar planta/exportar relatórios
M2.04	Procura/criar gestor
M2.05	Resultados da procura
M2.06	Editar gestor/ <i>reset</i> credenciais/cessar gestor
M2.07	Relatório de sistema (utilização de armazenamento/utilização da quantidade de canais/quantidade de itens/quantidade de acessos/quantidade de gestores/quantidade de acessos de gestão/utilização do canal/zonas mais visitadas no canal/estatísticas sociais)
M2.08	Personalizar canal/visualizar canal
M2.09	Editar permisões de visualização do canal
M2.10	Personalizar portal público (criar)/editar permisões de visualização
M2.11	Personalizar portal público (editar)/editar permisões de visualização
M2.12	Personalizar portal privado (criar)/editar permisões de visualização
M2.13	Personalizar portal privado (editar)/editar permisões de visualização
M2.14	Personalizar planta/georreferenciar a planta
M2.15	Procurar/resultados da procura
M2.16	Personalizar regiões
M2.17	Personalizar itens
M2.18	Relatório do canal

Tabela 21: Lista de mockups da Aplicação de Gestão para Clientes

Procurar/solicitar novo canal

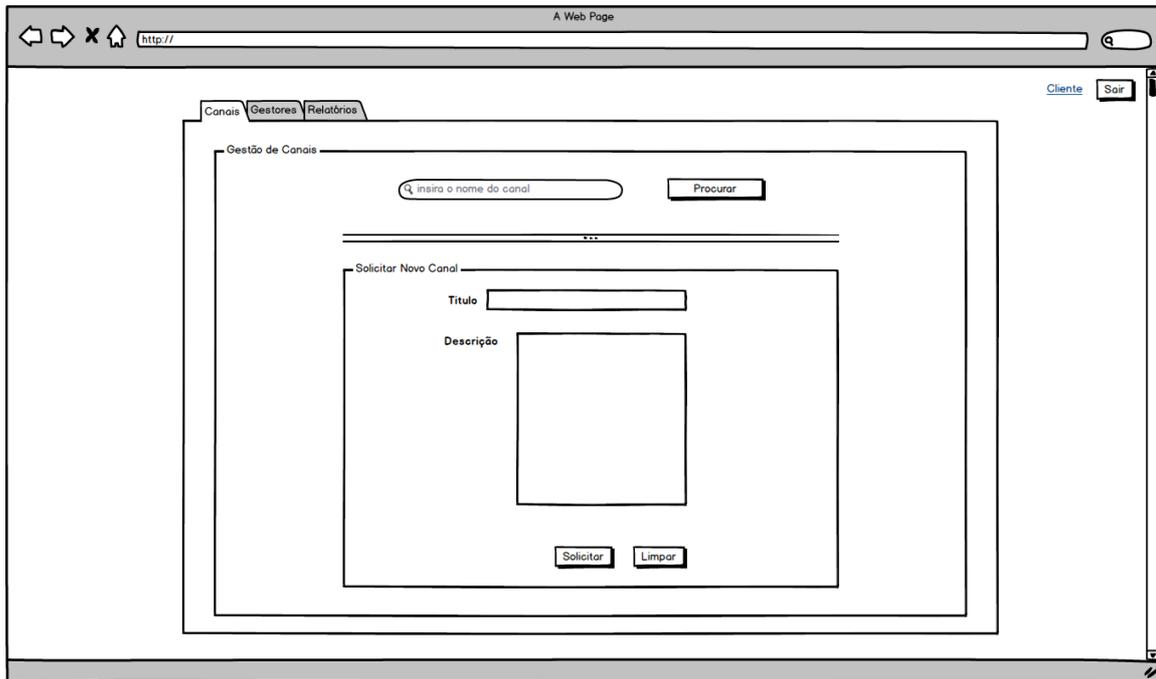


Figura 43: Mockup M2.01

Resultados da procura de canais

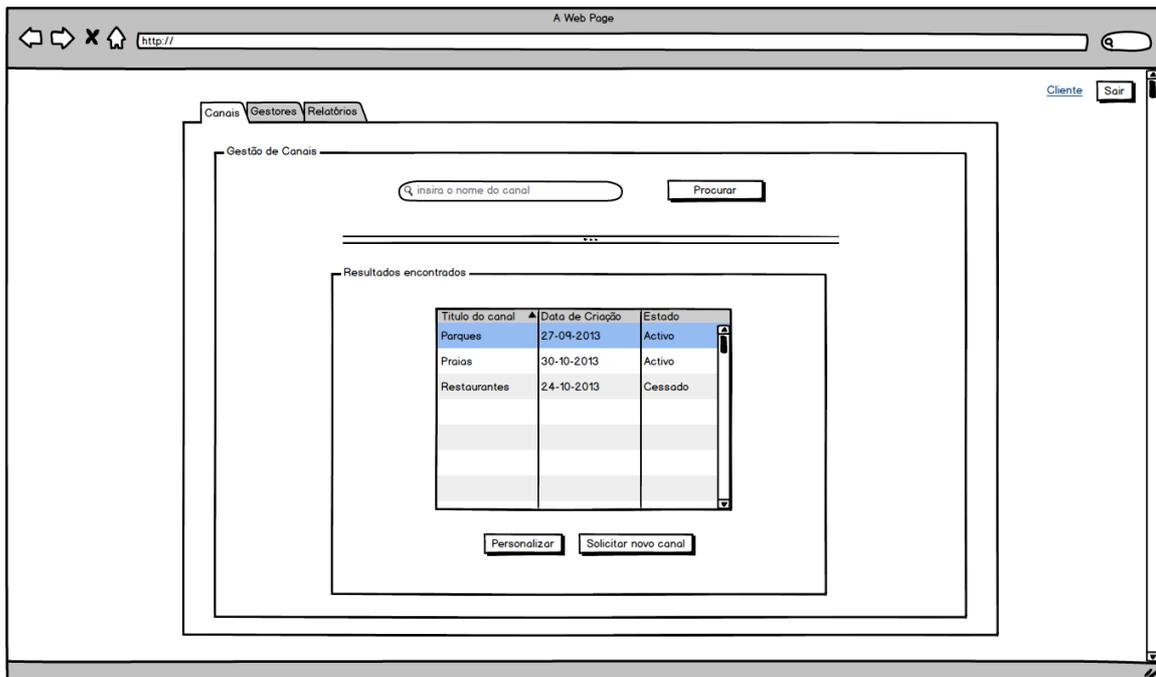


Figura 44: Mockup M2.02

Personalizar canal

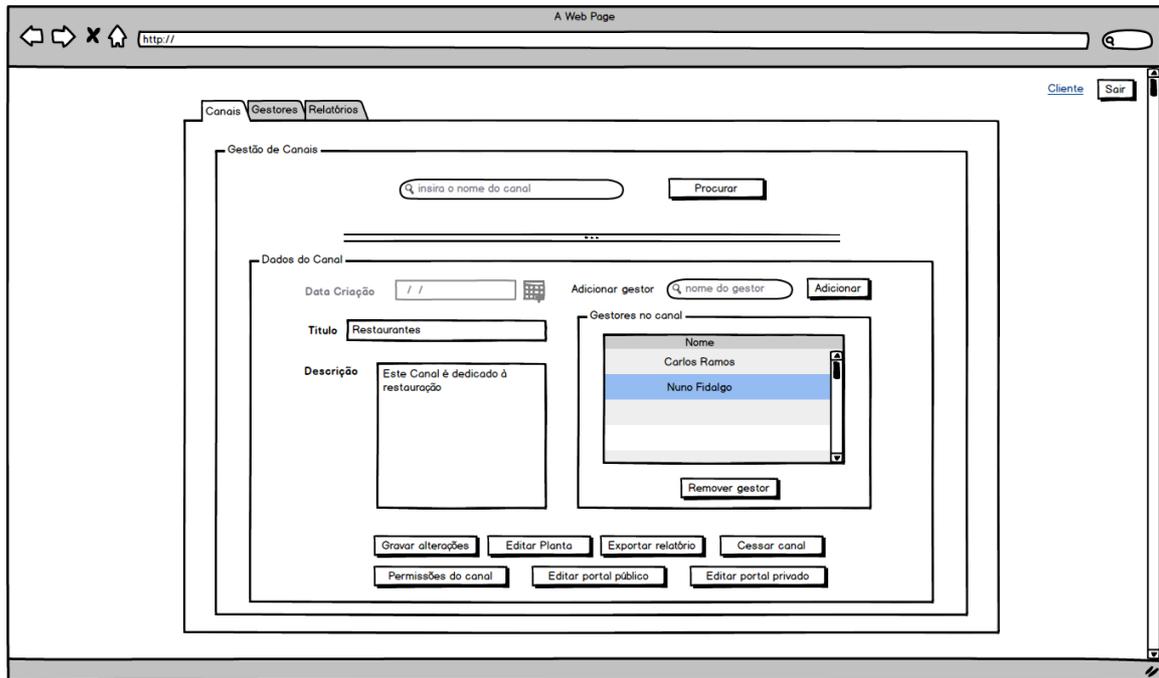


Figura 45: Mockup M2.03

Procurar/criar gestor

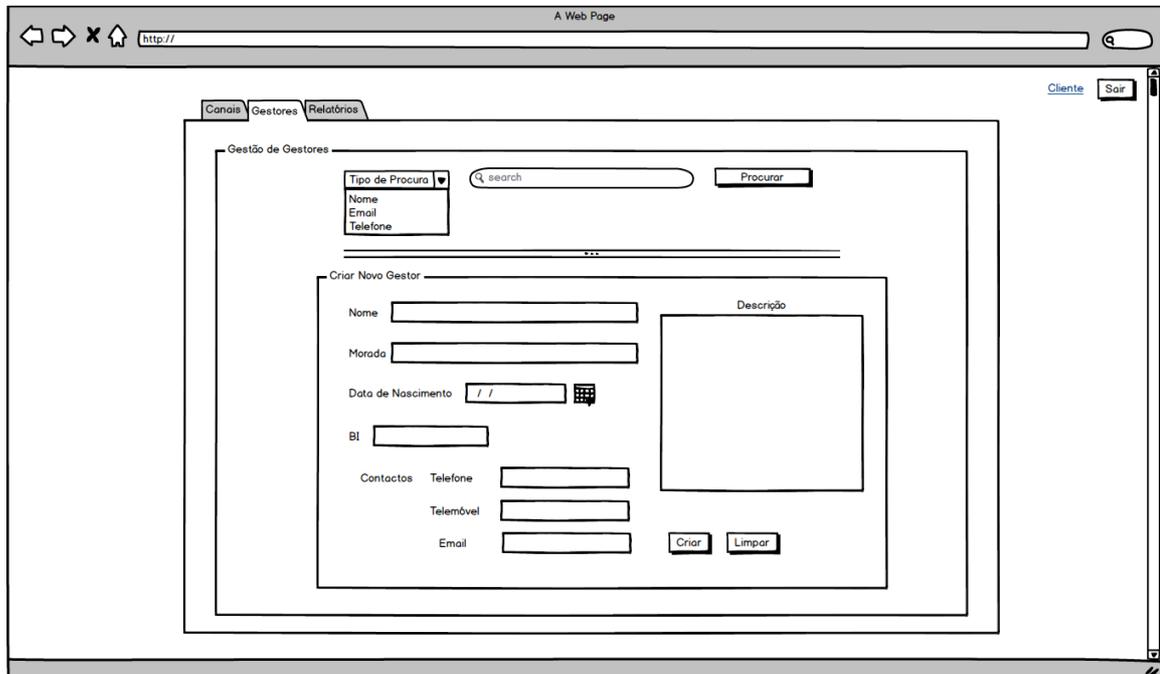


Figura 46: Mockup M2.04

Resultados da procura de utilizadores

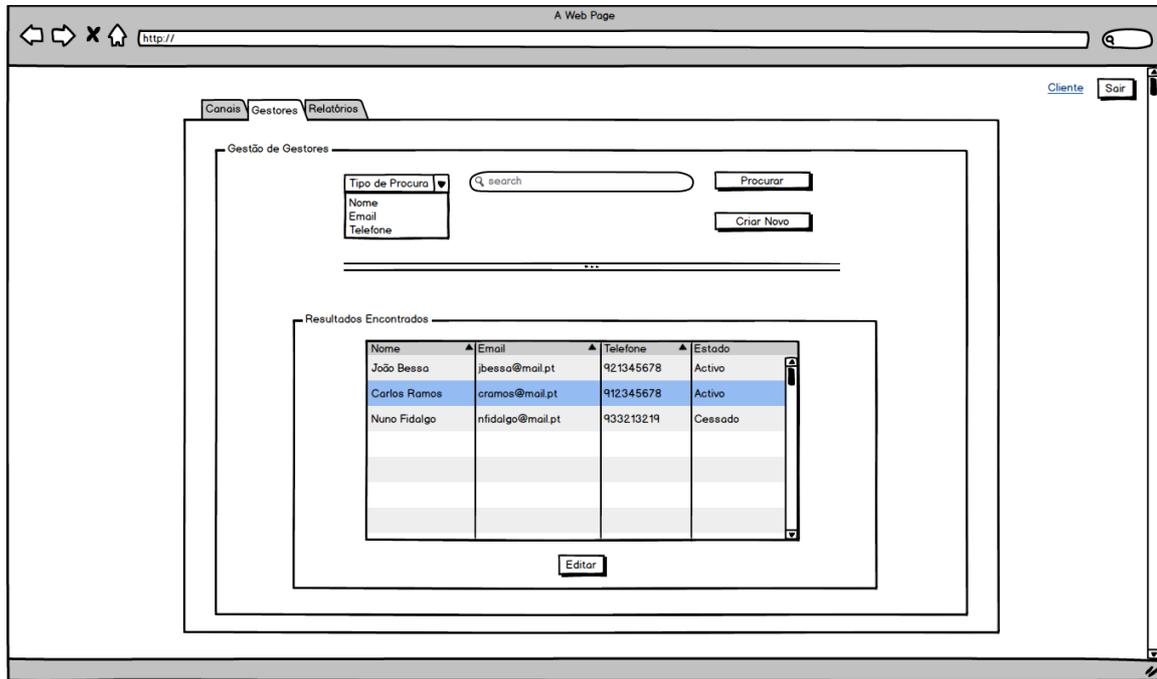


Figura 47: Mockup M2.05

Personalizar dados de um gestor

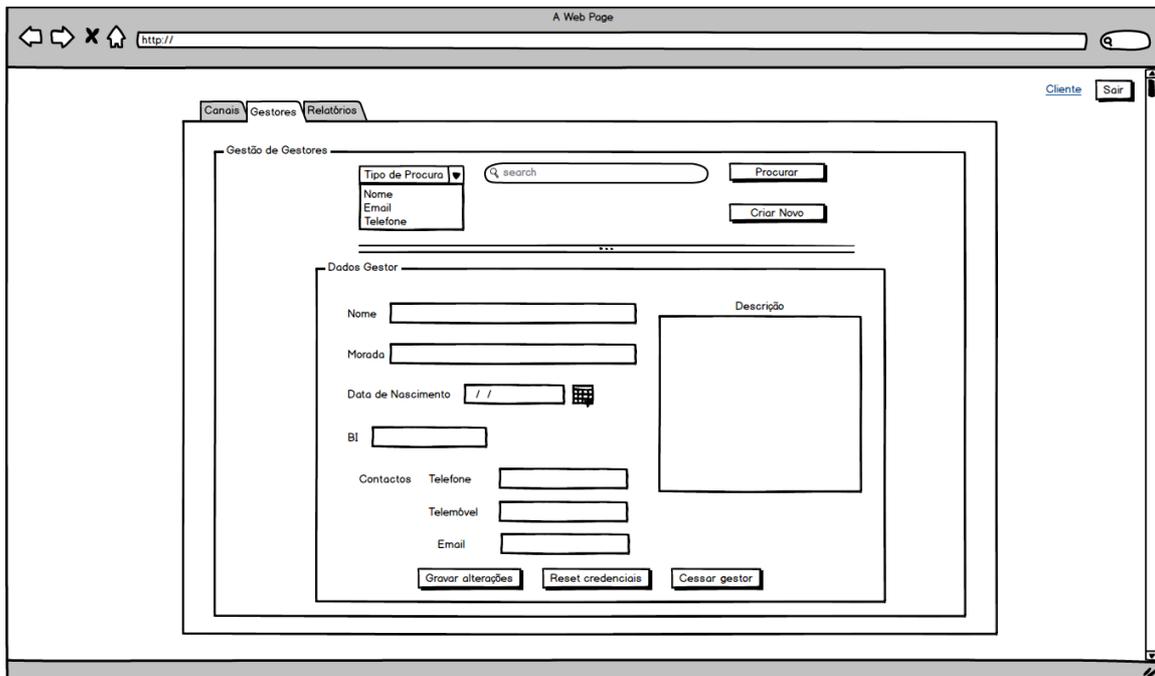


Figura 48: Mockup M2.06

Relatório de sistema

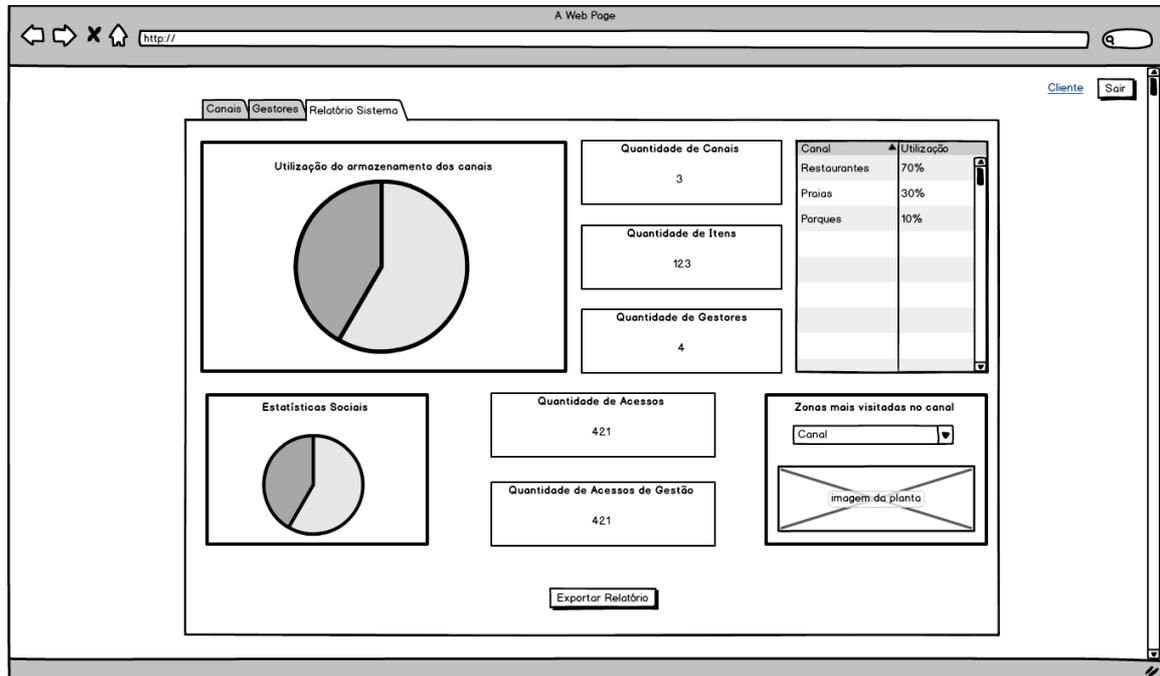


Figura 49: Mockup M2.07

Personalizar canal

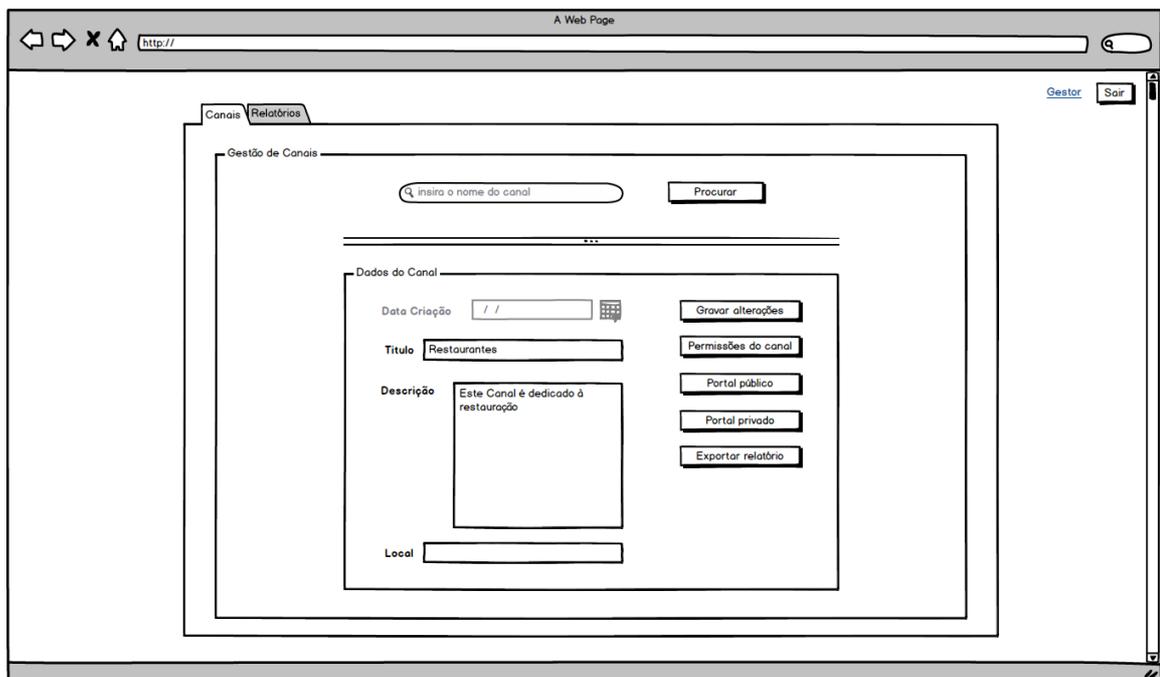


Figura 50: Mockup M2.08

Editar permissões de um canal

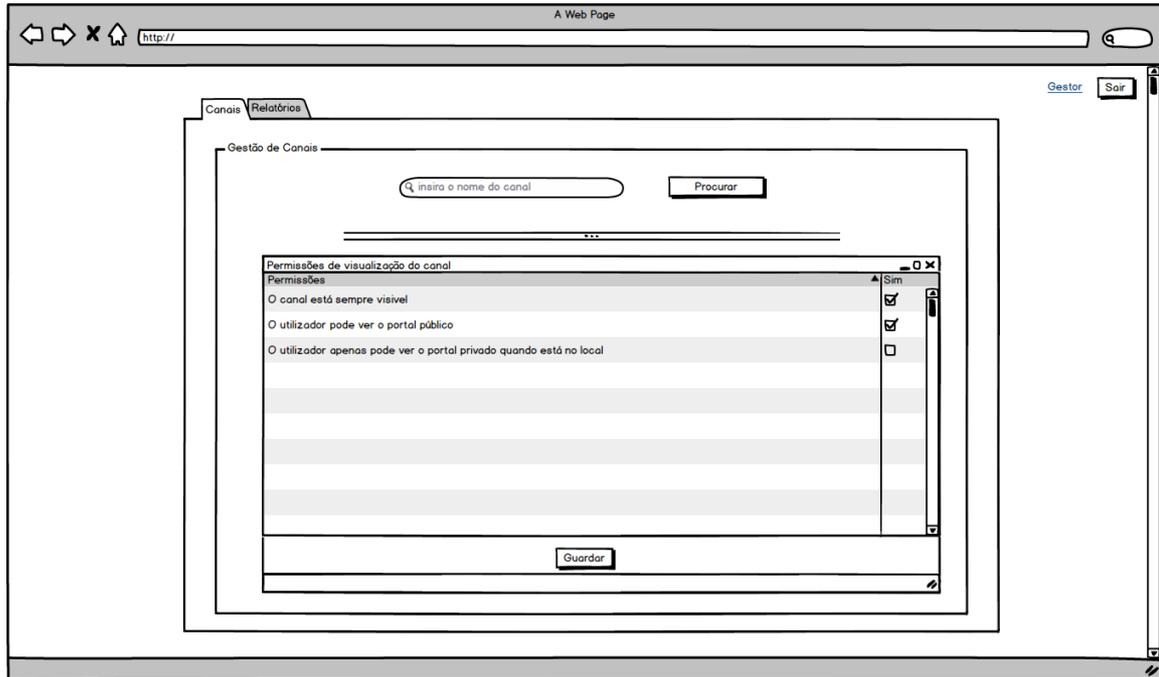


Figura 51: Mockup M2.09

Criar portal público

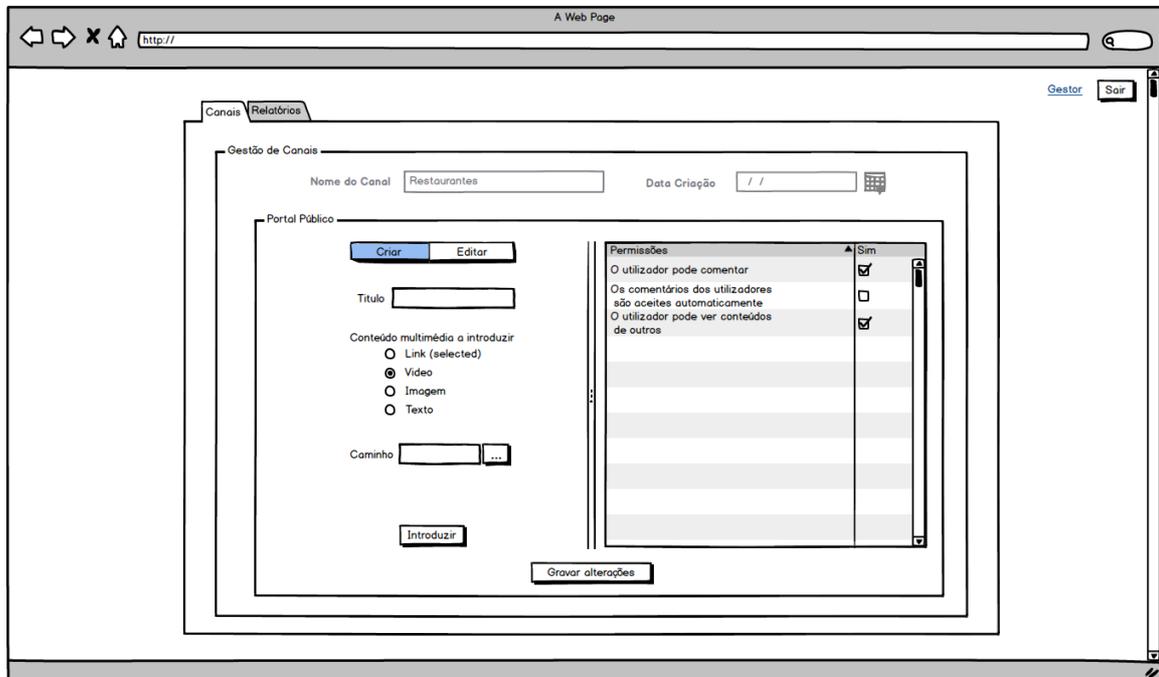


Figura 52: Mockup M2.10

Editar portal público

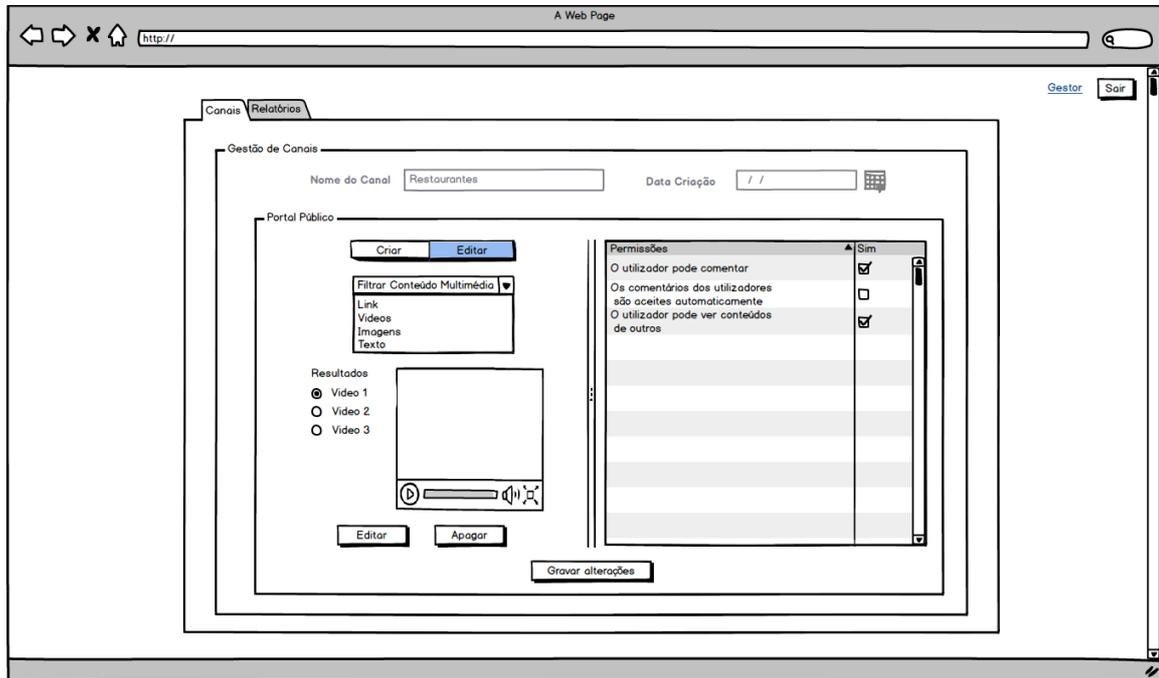


Figura 53: Mockup M2.11

Criar portal privado

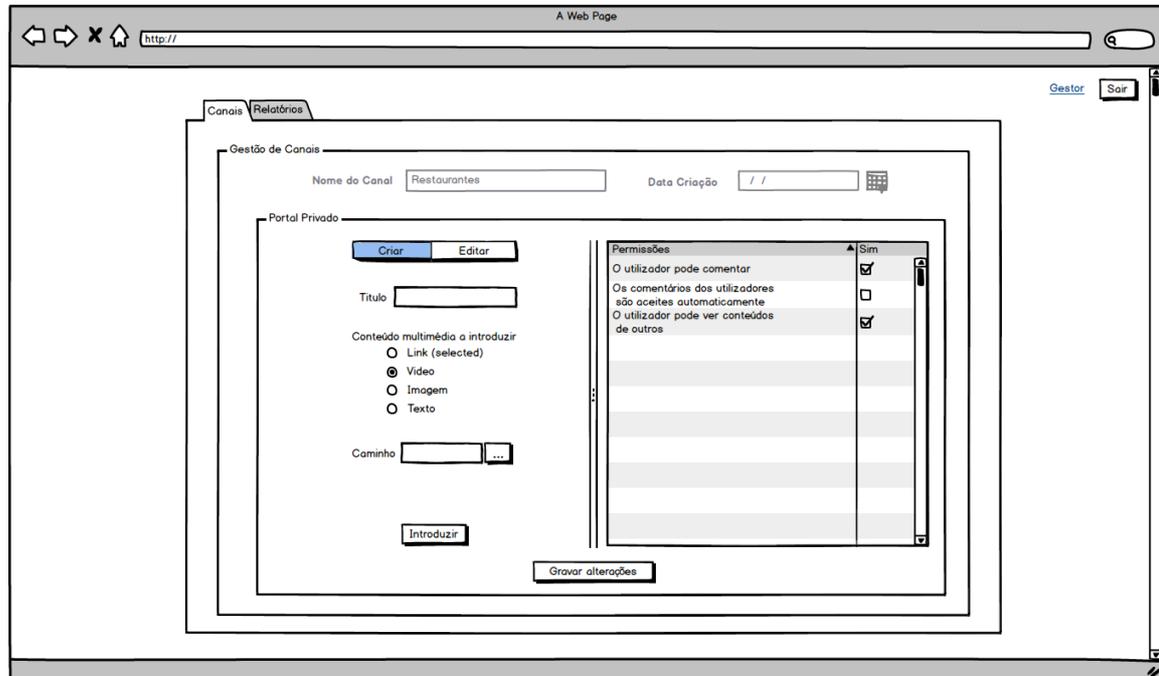


Figura 54: Mockup M2.12

Editar portal privado

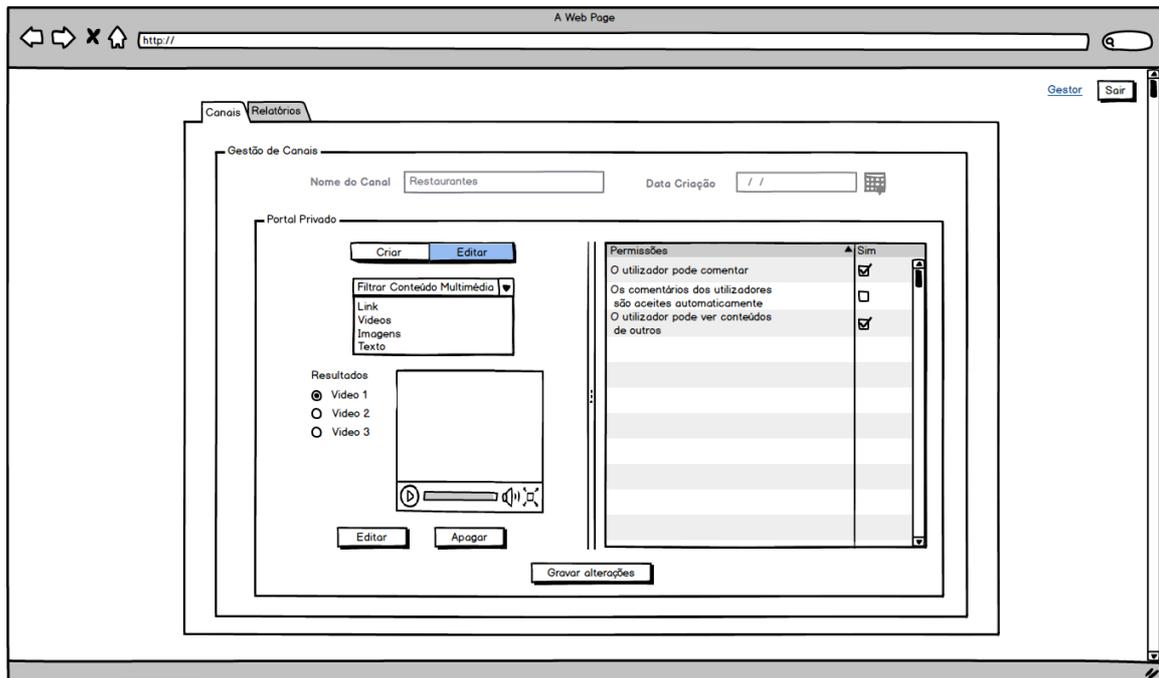


Figura 55: Mockup M2.13

Personalizar planta

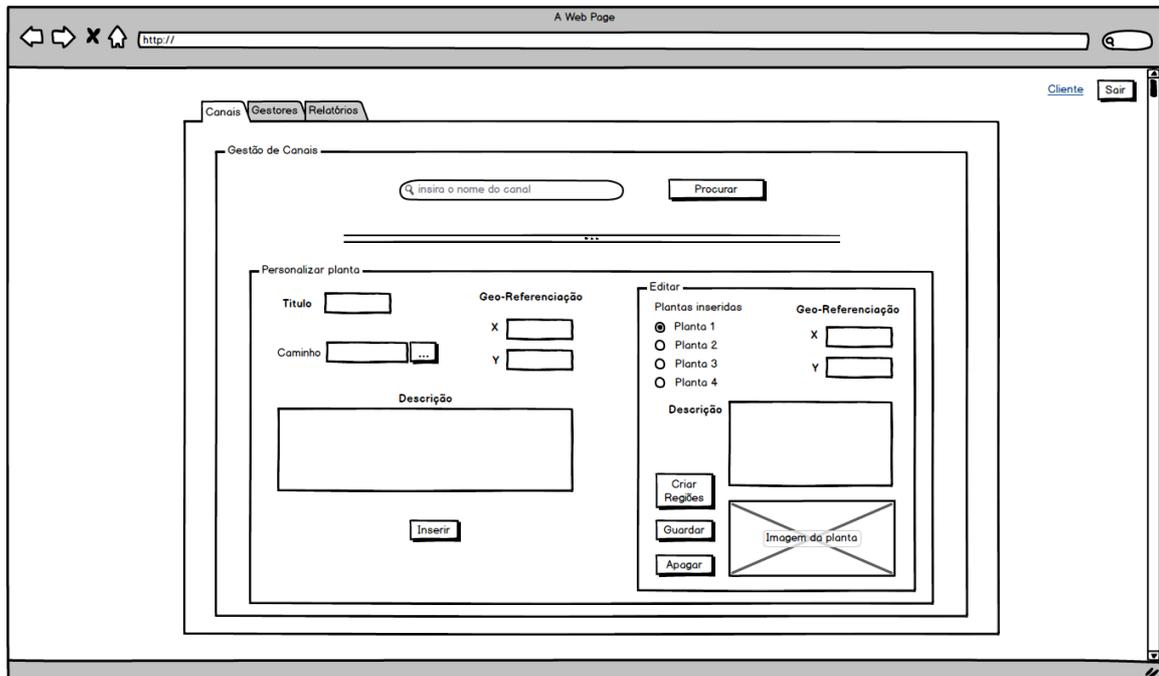


Figura 56: Mockup M2.14

Resultados da procura de canais

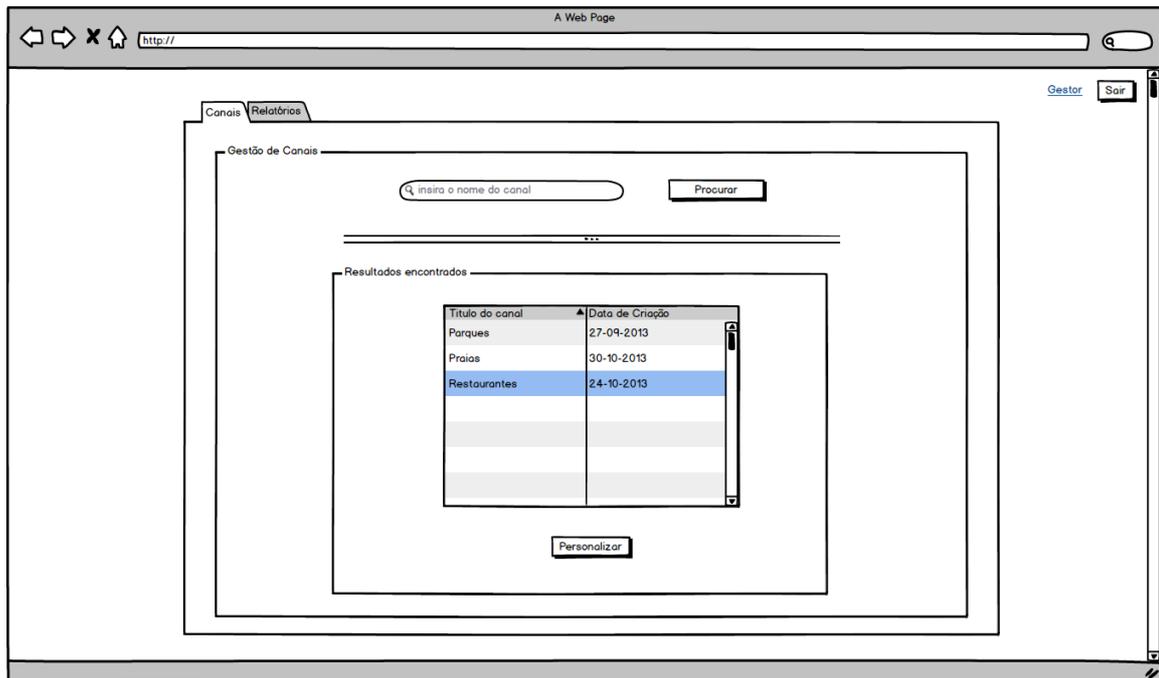


Figura 57: Mockup M2.15

Personalizar regiões

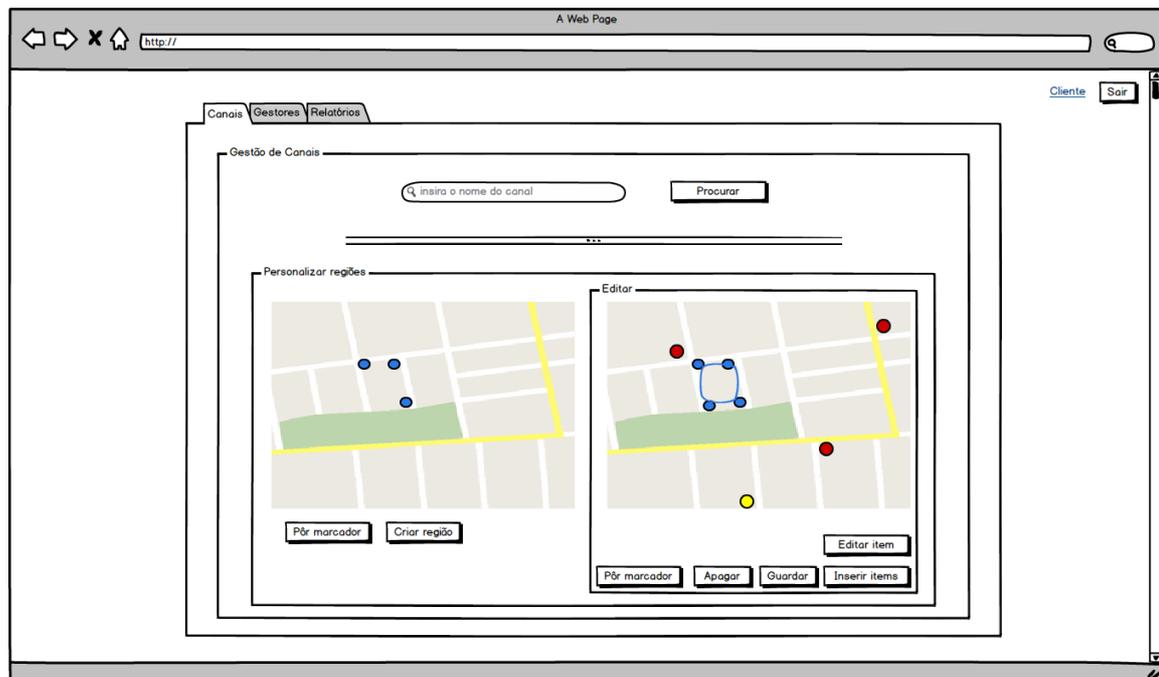


Figura 58: Mockup M2.16

Personalizar itens

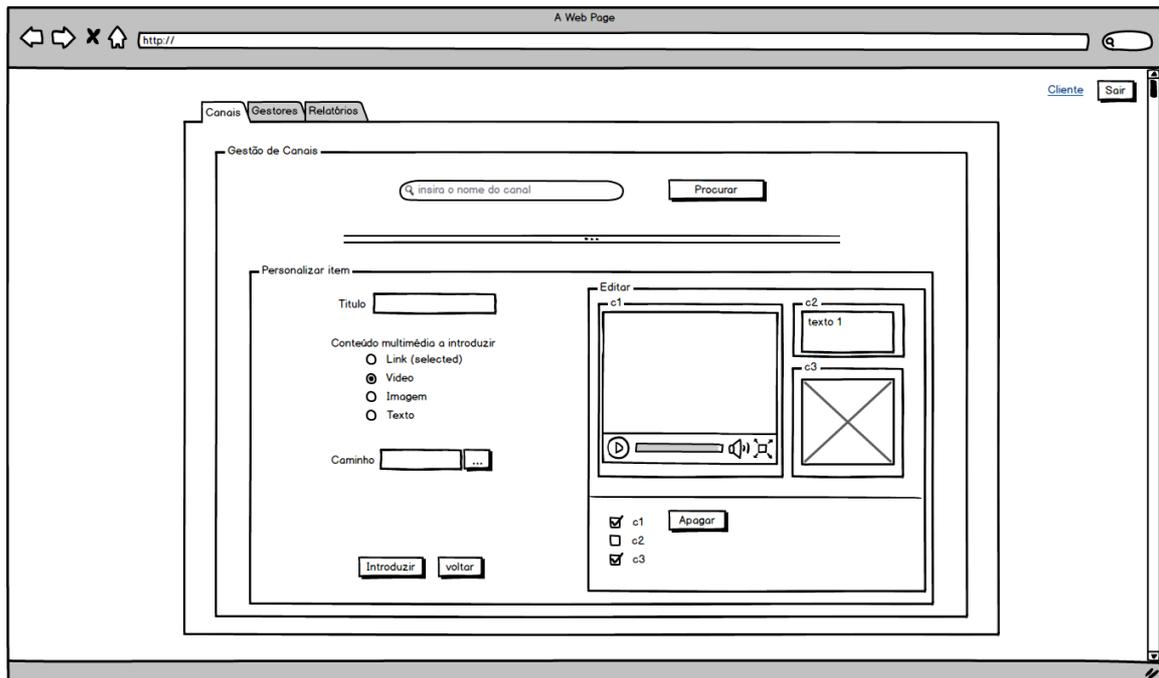


Figura 59: Mockup M2.17

Relatório do canal

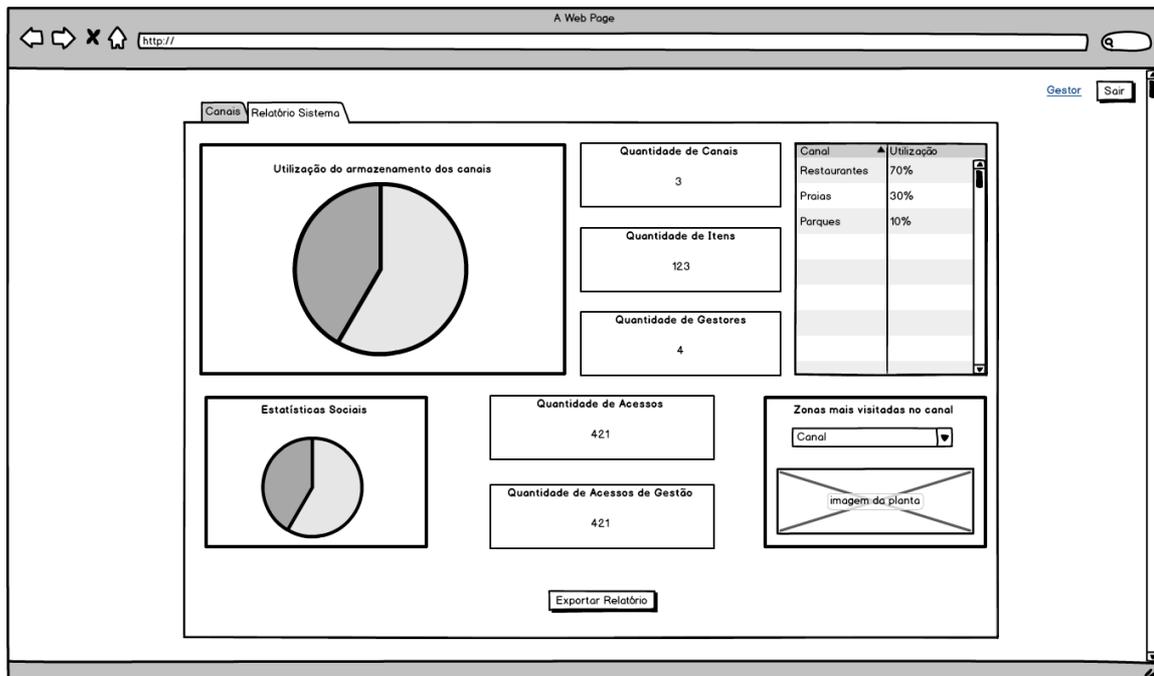


Figura 60: Mockup M2.18

Módulo 3 – Aplicação móvel

Código	Descrição
M3.0	Criar conta/autenticação por rede social
M3.1	Procura por tipo/procura por nome/lista dos canais mais próximos
M3.2	Resultados da procura
M3.3	Portal público
M3.4	Portal privado
M3.5	Ver plantas
M3.6	Comentar
M3.7	Ver planta com itens
M3.8	Ver/comentar item

Tabela 22: Lista de mockups da Aplicação Móvel

Autenticação



Figura 61: Mockup M3.0

Procurar canal



Figura 62: Mockup M3.1

Resultados da procura



Figura 63: Mockup M3.2

Portal público

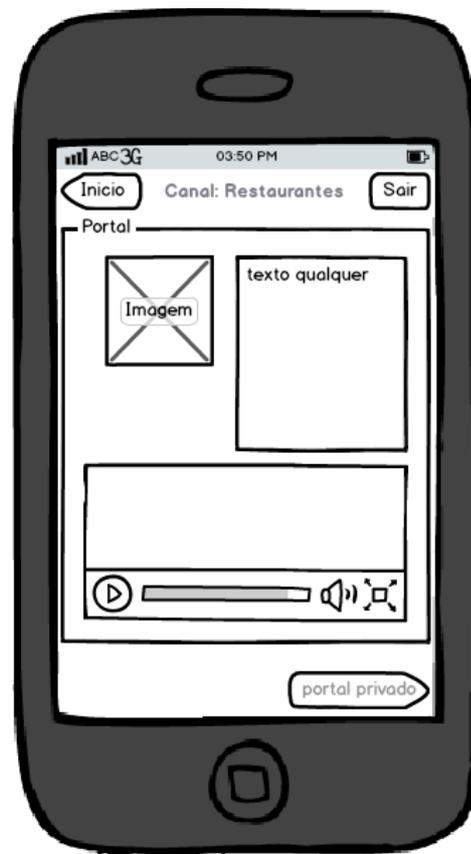


Figura 64: Mockup M3.3

Portal privado

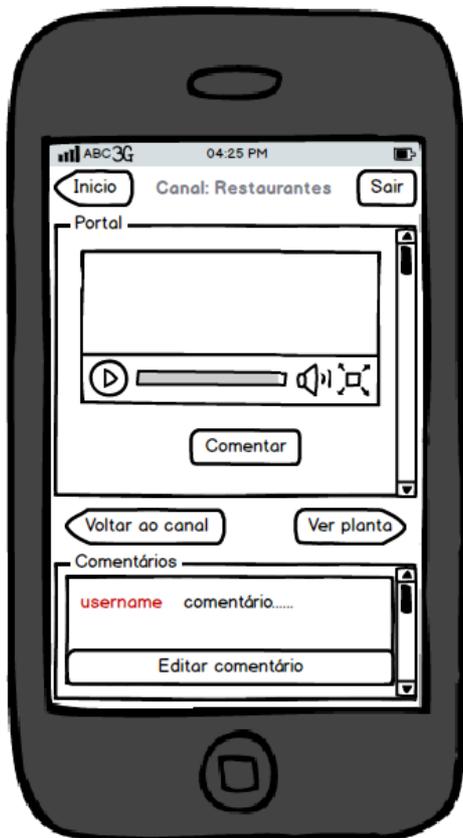


Figura 65: Mockup M3.4

Visualizar plantas

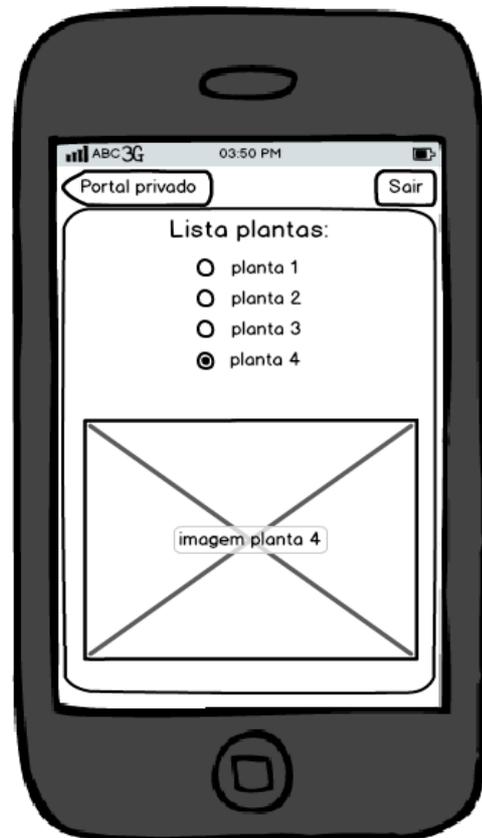


Figura 66: Mockup M3.5

Comentar



Figura 67: Mockup M3.6

Visualizar planta com itens

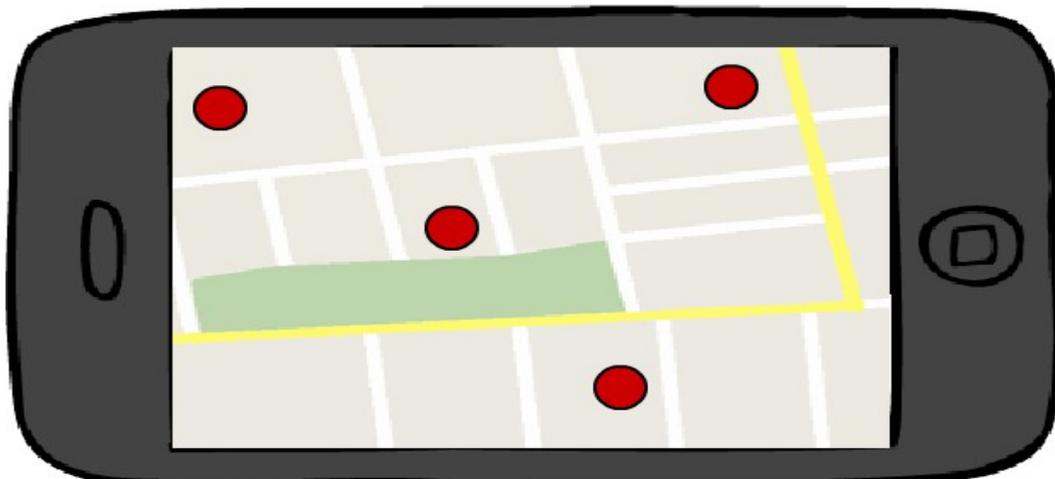


Figura 68: Mockup M3.7

Visualizar planta com itens

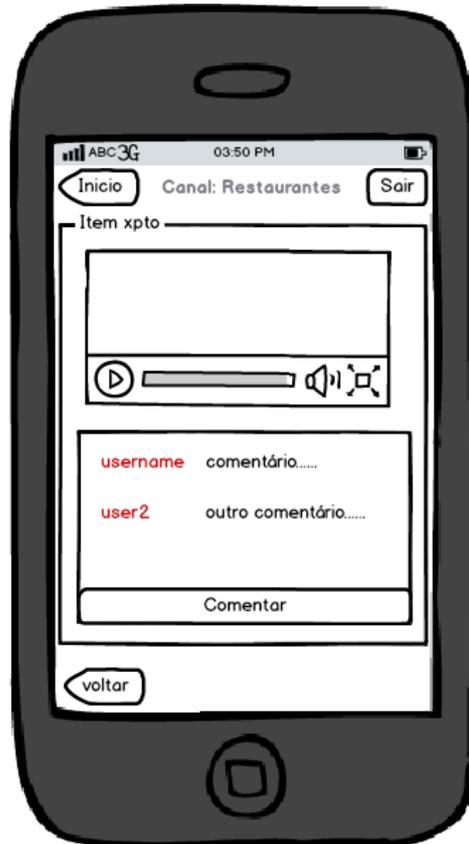


Figura 69: Mockup M3.8

C - Arquitectura

C.1 – Diagrama físico

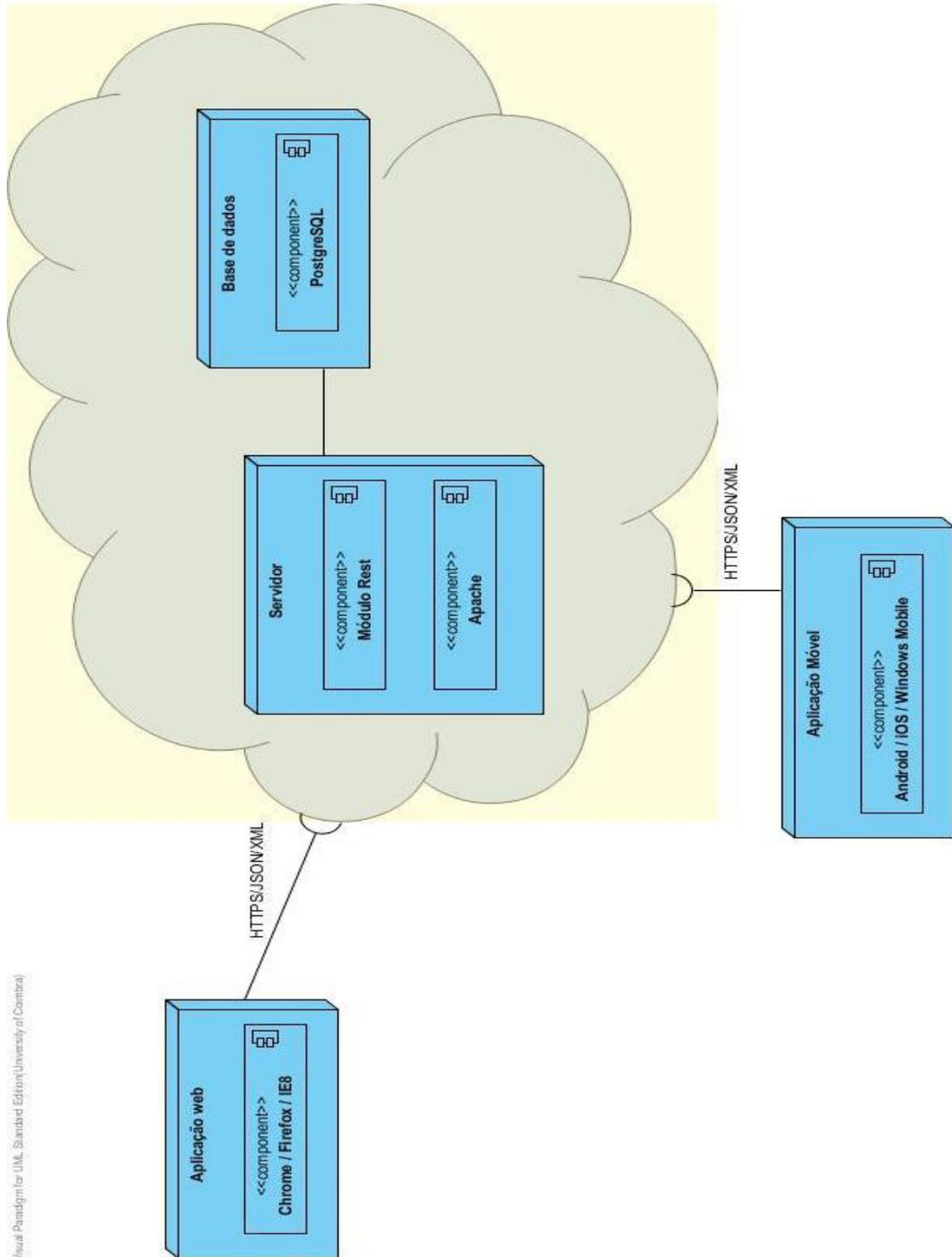


Figura 70: Diagrama físico

C.2 – Diagrama de estados

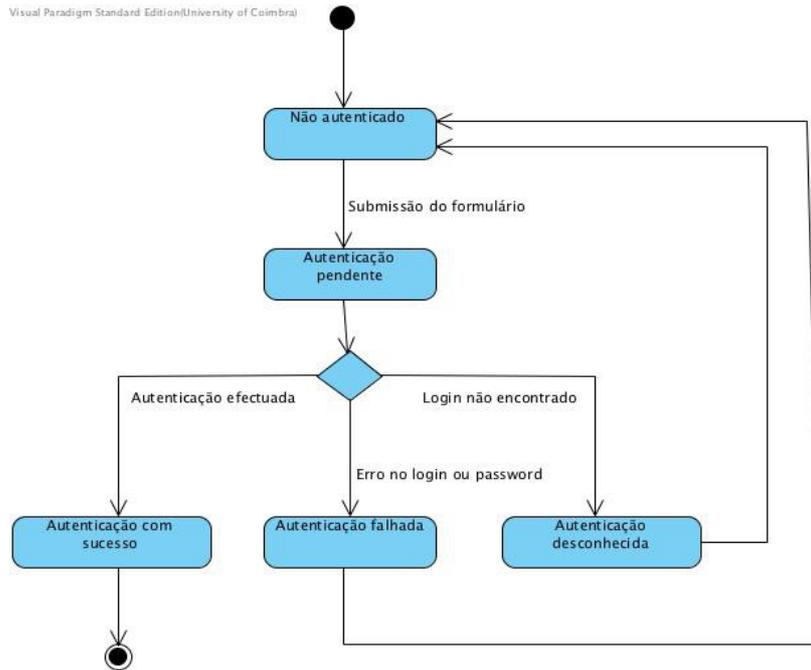


Figura 71: Autenticação plataforma web

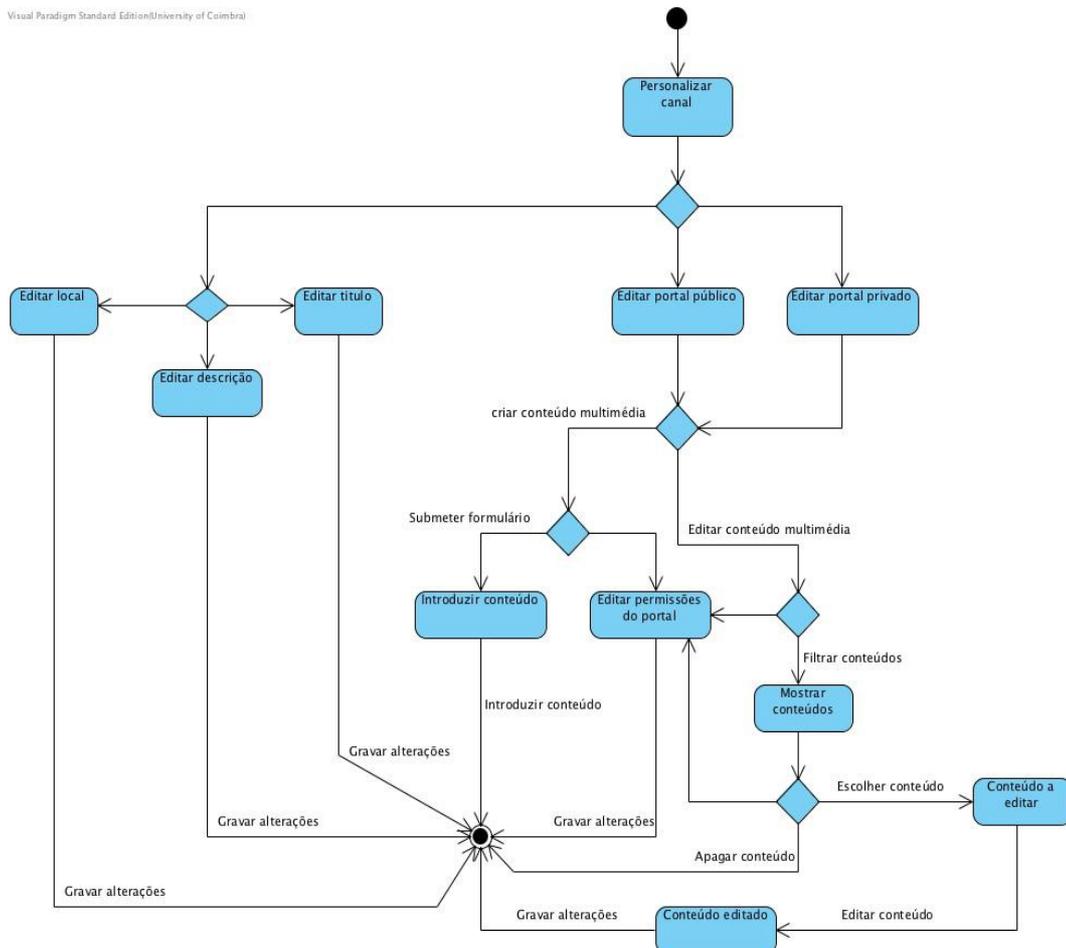


Figura 72: Personalização de um canal e/ou portal

Visual Paradigm Standard Edition(University of Coimbra)

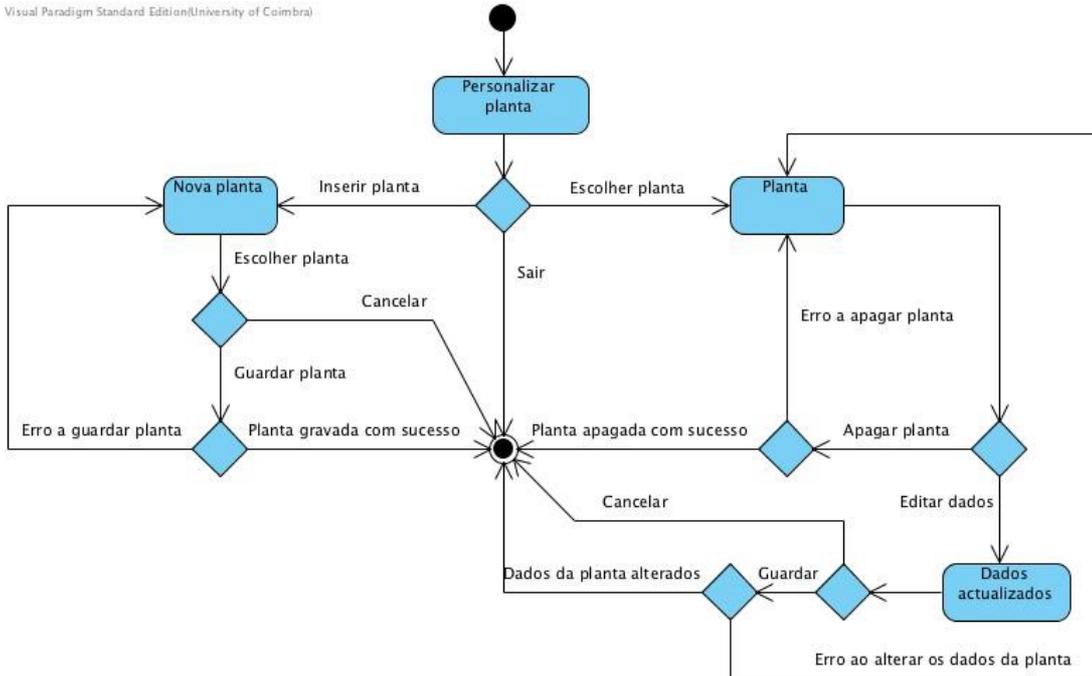


Figura 73: Personalização de uma planta

Visual Paradigm Standard Edition(University of Coimbra)

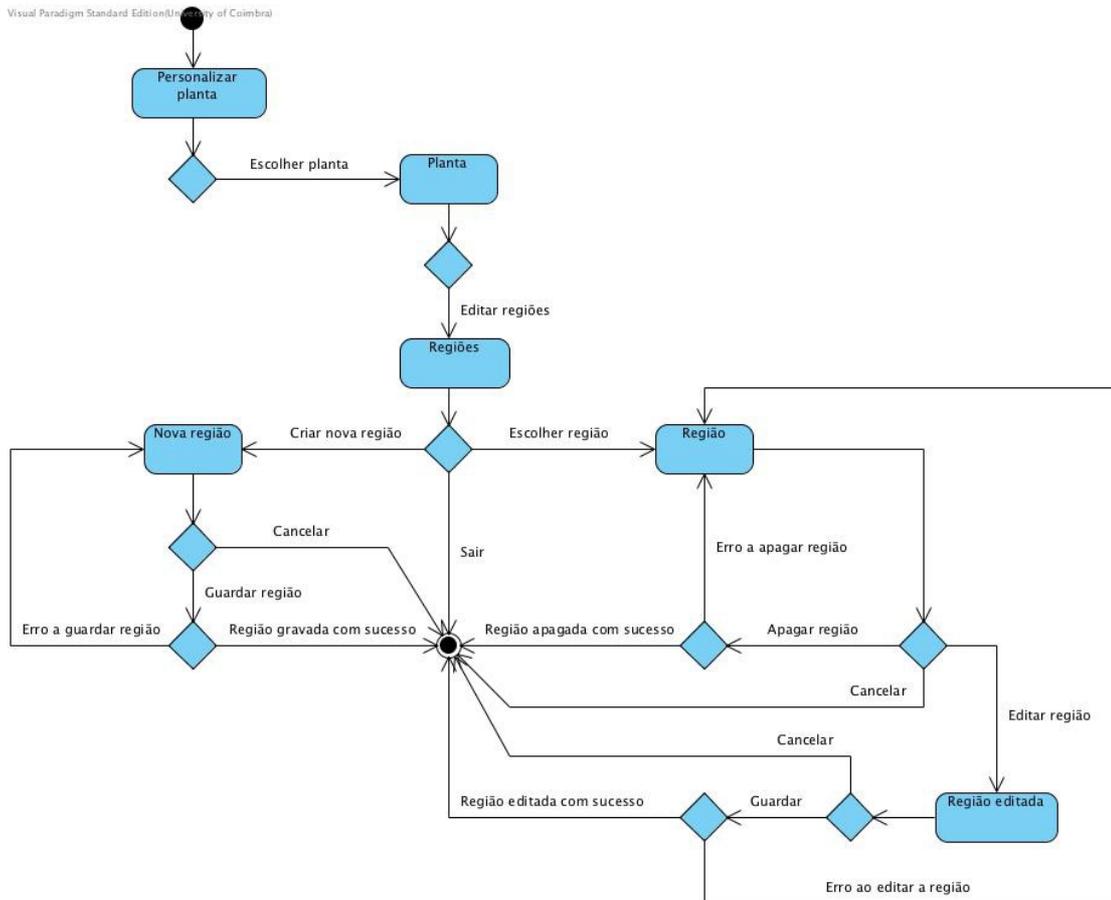


Figura 74: Personalização de uma região

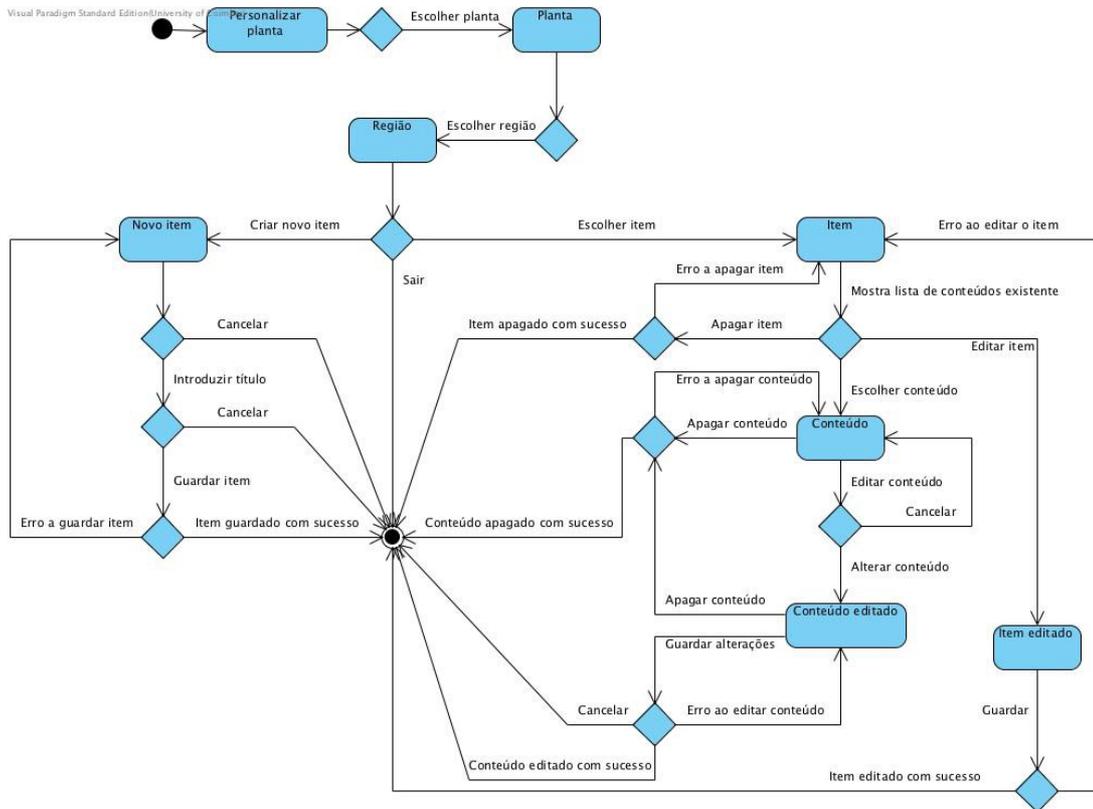


Figura 75: Personalização de um item e respectivo conteúdo

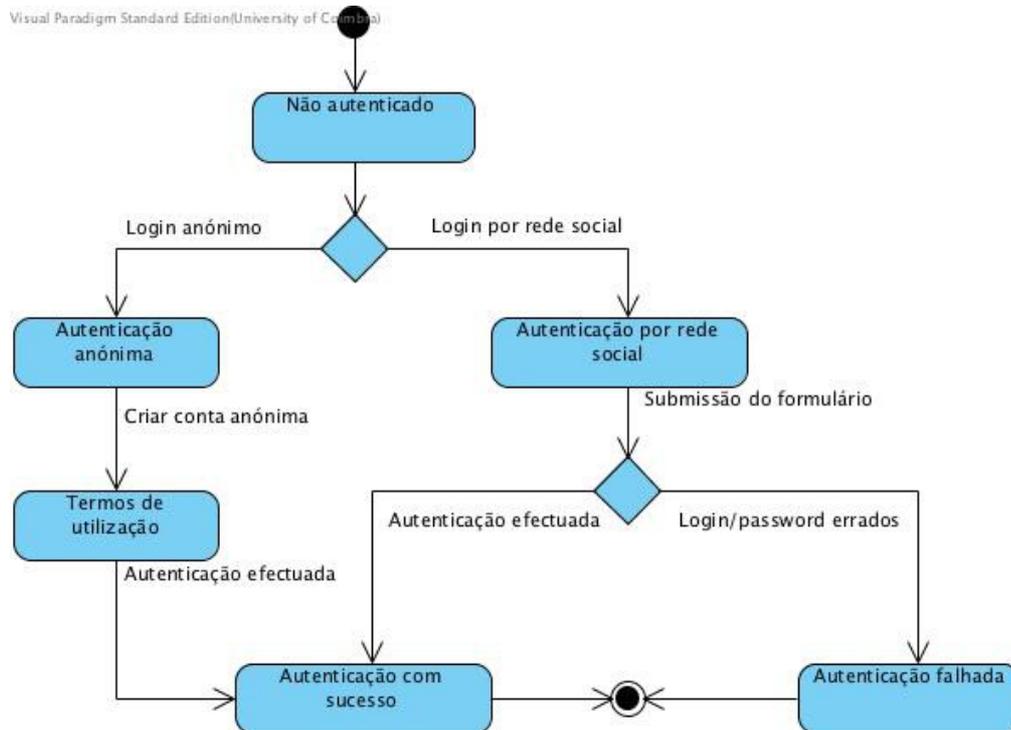


Figura 76: Autenticação plataforma móvel

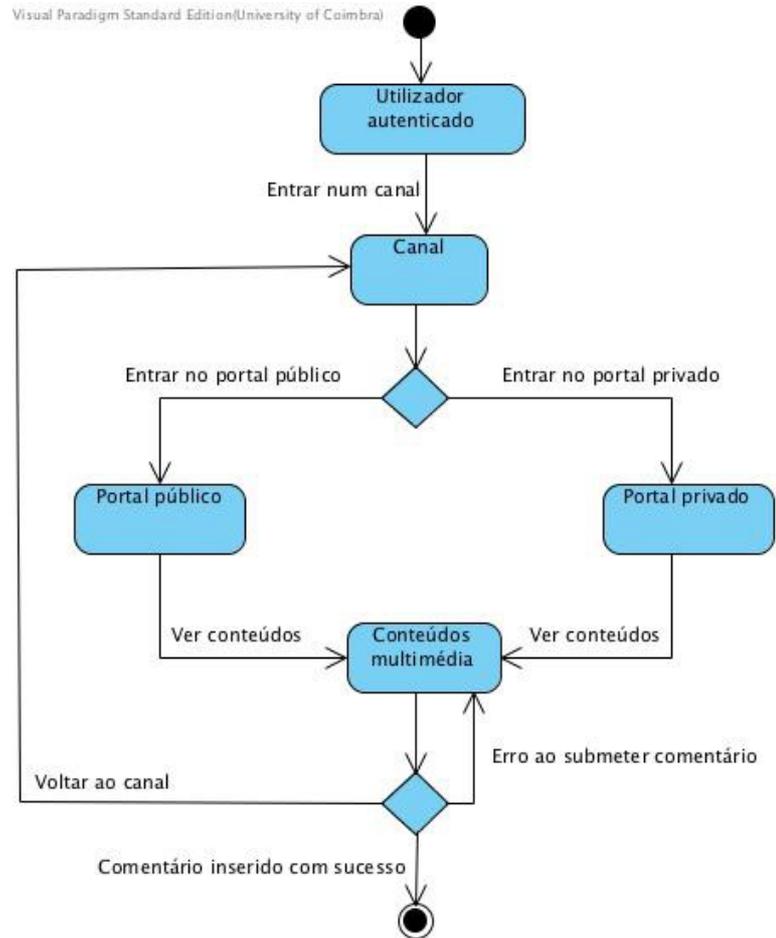


Figura 77: Ações num canal - plataforma móvel

D – Diagrama de Gantt

D.1 – Versão 1.0

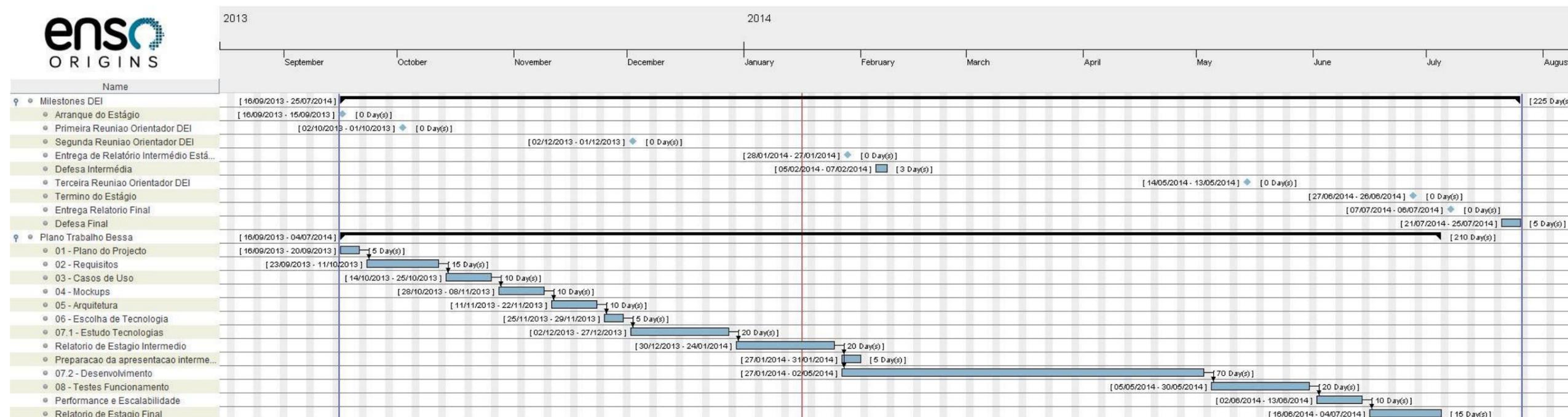


Figura 79: Diagrama de Gantt - versão 1

D.2 – Versão 2.0

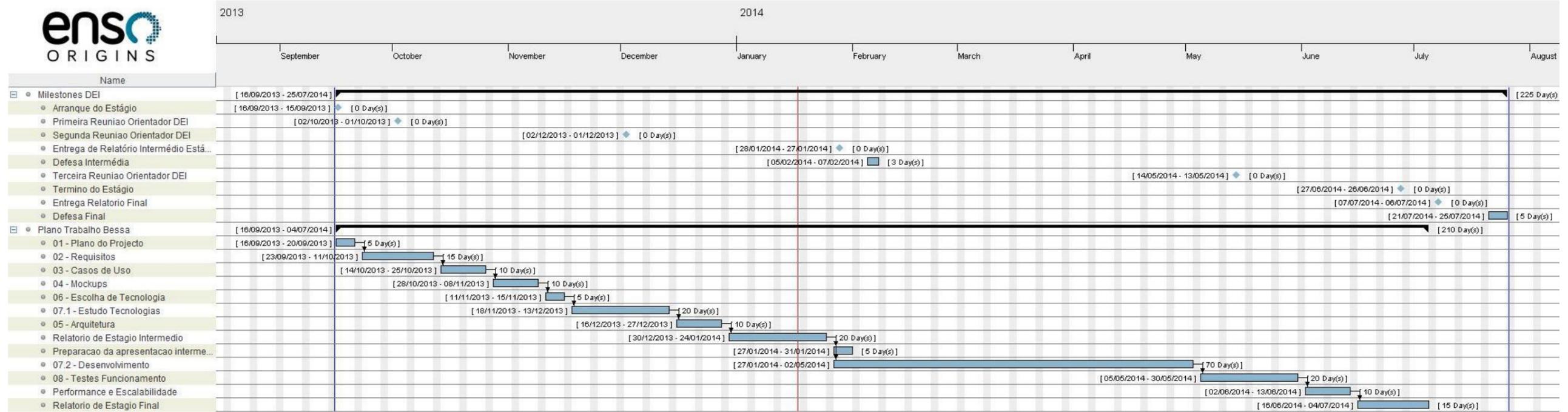


Figura 80: Diagrama de Gantt - versão 2

D.3 – Versão 3.0

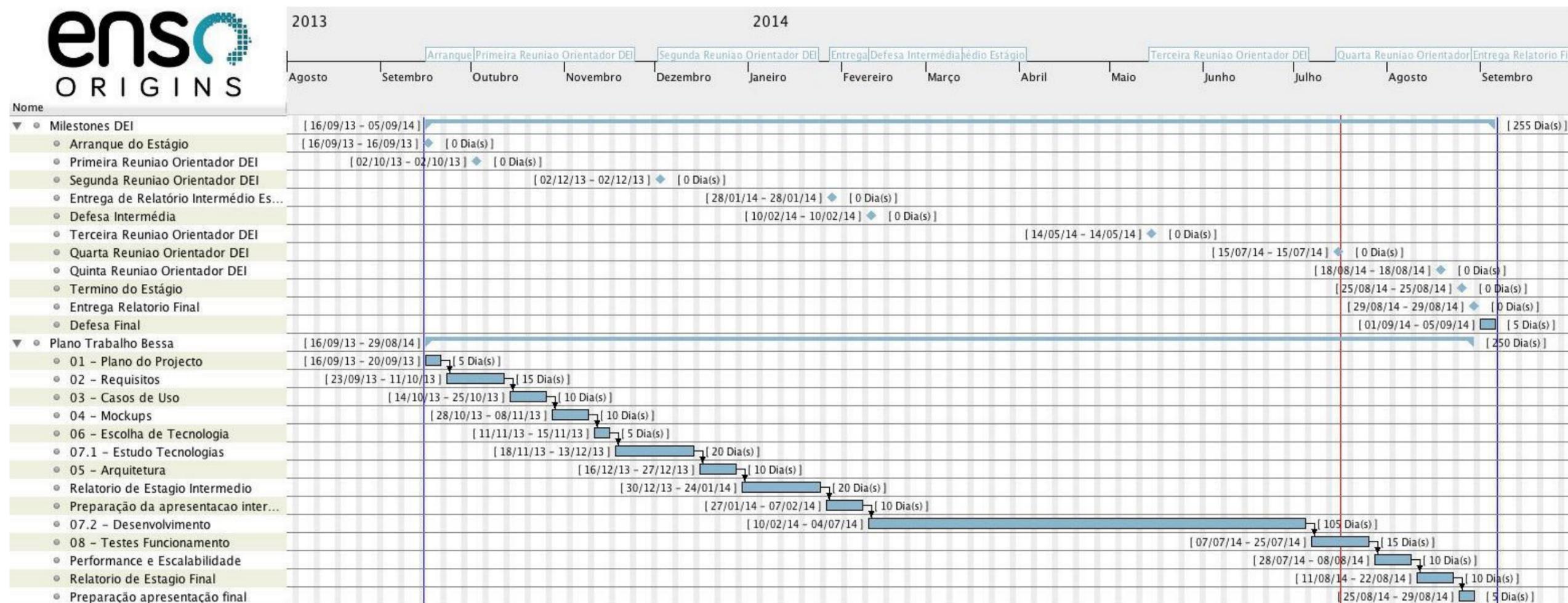


Figura 81: Diagrama de Gantt - versão 3

E - Testes

E.1 – Sistema

E.1.1 - Iniciais

ID Teste: 01
Funcionalidade: Autenticação falhada
Plataforma: Plataforma de gestão - web
Ações: O utilizador põe um nome de utilizador e/ou a <i>password</i> errada ou que não existam na base de dados.
Resultado esperado: Assinalar campos como inválidos.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 02
Funcionalidade: Autenticação com sucesso
Plataforma: Plataforma de gestão - web
Ações: O utilizador põe o nome de utilizador e a <i>password</i> correctamente.
Resultado esperado: Redireccionar para a plataforma de gestão administrativa ou para a gestão para clientes, consoante o tipo de utilizador.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 03
Funcionalidade: Registo na plataforma falhada
Plataforma: Plataforma de gestão - web
Ações: Registo na plataforma sem preencher todos os campos ou com tipos de dados inválidos.
Resultado esperado: Assinalar campos obrigatórios inválidos a vermelho.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 04
Funcionalidade: Registo na plataforma com sucesso
Plataforma: Plataforma de gestão - web
Ações: O utilizador faz um registo na plataforma com os dados todos válidos.
Resultado esperado: Retorna mensagem de sucesso e limpa todos os campos.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 05
Funcionalidade: Listar canais
Plataforma: Plataforma de gestão administrativa - web
Ações: O utilizador entra na plataforma de administração, sendo-lhe de seguida apresentado uma lista dos canais.
Resultado esperado: O sistema mostra a lista de todos os canais registados no sistema.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 06
Funcionalidade: Procura de canais por nome
Plataforma: Plataforma de gestão administrativa - web
Ações: O utilizador faz uma procura por nome do canal.
Resultado esperado: O sistema filtra os resultados da listagem de canais, consoante o que o utilizador escrever na procura.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 07
Funcionalidade: Aceitar pedidos para a criação de novos canais
Plataforma: Plataforma de gestão administrativa - web
Ações: O utilizador clica para aceitar pedidos, podendo posteriormente aceitar ou rejeitar esses pedidos.
Resultado esperado: O sistema mostra a lista de canais para serem aprovados ou rejeitados, redireccionando o utilizador para a lista de canais mal sejam aprovados ou rejeitados.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 08
Funcionalidade: Criação de um novo canal
Plataforma: Plataforma de gestão administrativa - web
Açções: O utilizador clica para criar novo canal, preenchendo correctamente os campos disponíveis (título e descrição).
Resultado esperado: Criação do novo canal com sucesso, sucedida por uma mensagem de sucesso.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 09
Funcionalidade: Editar próprio perfil
Plataforma: Plataforma de gestão administrativa - web
Açções: O utilizador clica no seu username para editar os dados do seu perfil.
Resultado esperado: Os dados são actualizados com sucesso, sendo apresentado uma mensagem de sucesso.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 10
Funcionalidade: Dado(s) do próprio perfil, de preenchimento obrigatório, apagado(s)
Plataforma: Plataforma de gestão administrativa - web
Açções: O utilizador apaga um ou mais dados do próprio perfil que seja de preenchimento obrigatório.
Resultado esperado: É mostrada uma mensagem de erro e assinalado a vermelho o(s) campo(s) de preenchimento obrigatórios, quando apagados.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 11
Funcionalidade: Reset da password do próprio perfil
Plataforma: Plataforma de gestão administrativa - web
Açções: O utilizador faz reset à password do próprio perfil.
Resultado esperado: É enviada uma mensagem de sucesso sucedido de um email para o email registado no perfil do utilizador.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 12
Funcionalidade: Apagar/desactivar a própria conta
Plataforma: Plataforma de gestão administrativa - web
Acções: O utilizador quer apagar/desactivar a própria conta.
Resultado esperado: Quando o utilizador apaga a sua própria conta, é reencaminhado para a página de login não podendo mais aceder à sua conta até alguém a activar.
Resultado obtido: Apaga a conta, mas não é reencaminhado para a página de login.
Resultado do teste: Falhou

ID Teste: 13
Funcionalidade: Listar utilizadores
Plataforma: Plataforma de gestão administrativa - web
Acções: O utilizador entra no menu de utilizadores, sendo-lhe de seguida apresentado uma lista dos utilizadores.
Resultado esperado: O sistema mostra a lista de todos os utilizadores registados no sistema.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 14
Funcionalidade: Procura de utilizadores por nome
Plataforma: Plataforma de gestão administrativa - web
Acções: O utilizador faz uma procura por nome de utilizador ou nome real.
Resultado esperado: O sistema filtra os resultados da listagem de utilizadores, consoante o que o utilizador escrever na procura. A procura tanto é feita por nome de utilizador como por nome real.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 15
Funcionalidade: Editar perfil de um utilizador
Plataforma: Plataforma de gestão administrativa - web
Acções: O administrador clica num nome de utilizador para editar os dados do perfil do mesmo, editando posteriormente os dados e guardando no final.
Resultado esperado: Os dados são actualizados com sucesso, sendo apresentado uma mensagem de sucesso.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 16
Funcionalidade: Dado(s) do perfil de um utilizador, de preenchimento obrigatório, apagado(s)
Plataforma: Plataforma de gestão administrativa - web
Acções: O administrador apaga um ou mais dados do perfil de um utilizador que seja de preenchimento obrigatório.
Resultado esperado: É enviada uma mensagem de erro e assinalado a vermelho o(s) campo(s) de preenchimento obrigatórios, quando apagados.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 17
Funcionalidade: Apagar/desactivar a conta de um utilizador
Plataforma: Plataforma de gestão administrativa - web
Acções: O administrador faz reset à password do perfil de um utilizador.
Resultado esperado: É enviada uma mensagem de sucesso sucedido de um email para o email registado no perfil do utilizador.
Resultado obtido: Erro a enviar email.
Resultado do teste: Falhou

ID Teste: 18
Funcionalidade: Reset da password do perfil de um utilizador
Plataforma: Plataforma de gestão administrativa - web
Acções: O administrador faz reset à password do perfil de um utilizador.
Resultado esperado: É enviada uma mensagem de sucesso sucedido de um email para o email registado no perfil do utilizador.
Resultado obtido: Erro a enviar email.
Resultado do teste: Falhou

ID Teste: 19
Funcionalidade: Assumir conta de um utilizador
Plataforma: Plataforma de gestão administrativa - web
Acções: O administrador assume a conta de um utilizador.
Resultado esperado: O administrador ao tentar assumir a conta de um utilizador, tem de voltar a introduzir as suas credenciais. Ao fazê-lo com sucesso, entra na plataforma como sendo esse utilizador.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 20
Funcionalidade: Visualizar relatórios
Plataforma: Plataforma de gestão administrativa - web
Ações: O administrador acede ao menu de relatórios.
Resultado esperado: O administrador ao aceder ao menu de relatórios, visualiza relatórios de todo o sistema.
Resultado obtido: Não está implementado.
Resultado do teste: Falhou

ID Teste: 21
Funcionalidade: Exportar relatórios
Plataforma: Plataforma de gestão administrativa - web
Ações: No menu de relatórios, o administrador clica em exportar.
Resultado esperado: O relatório é exportado em csv para onde o administrador desejar.
Resultado obtido: Não está implementado.
Resultado do teste: Falhou

ID Teste: 22
Funcionalidade: Terminar sessão
Plataforma: Plataforma de gestão - web
Ações: O utilizador clica para terminar sessão.
Resultado esperado: O utilizador ao terminar a sua sessão é reencaminhado para a página de login, perdendo acesso à plataforma onde estava.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 23
Funcionalidade: Listar canais
Plataforma: Plataforma de gestão para clientes - web
Ações: O utilizador entra na plataforma de gestão para clientes, sendo-lhe de seguida apresentado uma lista dos canais.
Resultado esperado: O sistema mostra a lista de todos os canais do cliente registados no sistema.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 24
Funcionalidade: Procura de canais por nome
Plataforma: Plataforma de gestão para clientes - web
Ações: O utilizador faz uma procura por nome do canal.
Resultado esperado: O sistema filtra os resultados da listagem de canais pertencentes ao cliente, consoante o que o utilizador escrever na procura.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 25
Funcionalidade: Fazer pedido de novo canal
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente faz um pedido para poder ter um novo canal, escrevendo o nome do canal que deseja e a respectiva descrição.
Resultado esperado: O canal é criado, ficando em estado pendente enquanto espera a aprovação de um administrador do sistema.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 26
Funcionalidade: Editar próprio perfil
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente clica no seu nome de utilizador para editar os dados do seu perfil
Resultado esperado: Os dados são actualizados com sucesso, sendo apresentado uma mensagem de sucesso.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 27
Funcionalidade: Apagar dado(s) do próprio perfil que seja de preenchimento obrigatório
Plataforma: Plataforma de gestão para clientes - web
Ações: O utilizador apaga um ou mais dados do próprio perfil que seja de preenchimento obrigatório.
Resultado esperado: É mostrada uma mensagem de erro e assinalado a vermelho o(s) campo(s) de preenchimento obrigatórios, quando apagados.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 28
Funcionalidade: Visualizar gestores criados pelo cliente
Plataforma: Plataforma de gestão para clientes - web
Ações: O utilizador entra no menu de utilizadores, sendo-lhe de seguida apresentado uma lista dos gestores criados pelo cliente.
Resultado esperado: O sistema mostra a lista de todos os gestores registados no sistema pelo cliente.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 29
Funcionalidade: Fazer procura de gestores
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente faz uma procura por nome de utilizador ou nome real.
Resultado esperado: O sistema filtra os resultados da listagem de gestores, consoante o que o cliente escrever na procura. A procura tanto é feita por nome de utilizador como por nome real.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 30
Funcionalidade: Reset da própria password
Plataforma: Plataforma de gestão para clientes - web
Ações: O utilizador faz reset à password do próprio perfil.
Resultado esperado: É enviada uma mensagem de sucesso sucedido de um email para o email registado no perfil do utilizador.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 31
Funcionalidade: Apagar a própria conta
Plataforma: Plataforma de gestão para clientes - web
Ações: O utilizador quer apagar/desactivar a própria conta.
Resultado esperado: Quando o utilizador apaga a sua própria conta, é reencaminhado para a página de login não podendo mais aceder à sua conta até alguém a activar.
Resultado obtido: Apaga a conta, mas não é reencaminhado para a página de login.
Resultado do teste: Falhou

ID Teste: 32
Funcionalidade: Editar dados de um gestor
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente clica num username para editar os dados do perfil de um gestor, editando posteriormente os dados e guardando no final.
Resultado esperado: Os dados são actualizados com sucesso, sendo apresentado uma mensagem de sucesso.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 33
Funcionalidade: Dado(s) do perfil de um gestor, de preenchimento obrigatório, apagado(s)
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente apaga um ou mais dados do perfil de um gestor que seja de preenchimento obrigatório.
Resultado esperado: É enviada uma mensagem de erro e assinalado a vermelho o(s) campo(s) de preenchimento obrigatórios, quando apagados.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 34
Funcionalidade: Apagar a conta de um gestor
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente faz reset à password do perfil de um gestor.
Resultado esperado: É enviada uma mensagem de sucesso sucedido de um email para o email registado no perfil do utilizador.
Resultado obtido: Erro a enviar email.
Resultado do teste: Falhou

ID Teste: 35
Funcionalidade: Reset da password do perfil de um gestor
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente faz reset à password do perfil de um gestor.
Resultado esperado: É enviada uma mensagem de sucesso sucedido de um email para o email registado no perfil do utilizador.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 36
Funcionalidade: Procurar por nome de uma rua, cidade ou país para definir a área de abrangência do canal
Plataforma: Plataforma de gestão para clientes - web
Ações: O gestor ao clicar no nome do canal que deseja, caso a área de abrangência do mesmo ainda não esteja definida, escreve o nome da rua, cidade ou país e clica em procurar.
Resultado esperado: A imagem presente no mapa, tem por <i>default</i> a cidade de Coimbra, mas após ser escrito o nome de uma rua, cidade ou país, o mapa é reencaminhado para onde o cliente deseja.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 37
Funcionalidade: Definir área de abrangência de uma canal
Plataforma: Plataforma de gestão para clientes - web
Ações: O gestor ao clicar no nome do canal que deseja, caso a área de abrangência do mesmo ainda não esteja definida, tem de definir a área de abrangência do canal.
Resultado esperado: Só é possível definir a área de abrangência, caso se tenha marcado no mínimo 3 pontos no mapa.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 38
Funcionalidade: Adicionar gestor a um canal
Plataforma: Plataforma de gestão para clientes - web
Ações: O gestor ao clicar no nome do canal que deseja, caso a área de abrangência do mesmo esteja definida, mostra os dados e as opções do canal. Neste menu, o cliente clica em adicionar gestor, aparecendo uma lista dos seus gestores de onde pode escolher quais os gestores que quer adicionar.
Resultado esperado: Na lista de gestores, aparecem os gestores que estão/foram adicionados ao canal em questão.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 39
Funcionalidade: Remover gestor a um canal
Plataforma: Plataforma de gestão para clientes - web
Ações: O gestor ao clicar no nome do canal que deseja, caso a área de abrangência do mesmo esteja definida, mostra os dados e as opções do canal. Neste menu, o cliente clica no gestor que quer remover do canal e clica em apagar gestor.
Resultado esperado: O gestor é removido do canal em causa.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 40
Funcionalidade: Criar novo gestor
Plataforma: Plataforma de gestão para clientes - web
Ações: O gestor ao clicar no nome do canal que deseja, caso a área de abrangência do mesmo esteja definida, mostra os dados e as opções do canal. Neste menu, o cliente clica em novo gestor, preenchendo correctamente os campos obrigatórios para conseguir criar um novo gestor.
Resultado esperado: Caso o cliente preencha correctamente todos os campos, o gestor é criado com sucesso, sendo enviado um email com a password para o email do novo gestor.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 41
Funcionalidade: Editar dados do canal
Plataforma: Plataforma de gestão para clientes - web
Ações: O gestor ao clicar no nome do canal que deseja, caso a área de abrangência do mesmo esteja definida, mostra os dados e as opções do canal. Ao clicar no editar dados canal, o cliente pode alterar o estado e/ou nome do canal e/ou a descrição do canal, guardando as alterações no final.
Resultado esperado: Caso o cliente preencha correctamente todos os campos, as alterações são guardadas com sucesso. Caso exista algum campo que deixou por preencher, é mostrado o campo a vermelho, de modo ao cliente perceber que não preencheu um dado campo que é obrigatório.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 42
Funcionalidade: Mostrar regiões existentes de um canal
Plataforma: Plataforma de gestão para clientes - web
Ações: O gestor ao clicar no nome do canal que deseja, caso a área de abrangência do mesmo esteja definida, mostra os dados e as opções do canal. Ao clicar no editar planta do canal, são mostradas as regiões, que já foram definidas pelo cliente para esse canal.
Resultado esperado: O utilizador vê as regiões definidas na área de abrangência do canal.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 43
Funcionalidade: Mostrar itens existentes de um canal
Plataforma: Plataforma de gestão para clientes - web
Ações: O gestor ao clicar no nome do canal que deseja, caso a área de abrangência do mesmo esteja definida, mostra os dados e as opções do canal. Ao clicar no editar planta do canal, são mostrados os itens, que já foram definidas pelo cliente para esse canal.
Resultado esperado: O utilizador vê os itens definidos nas regiões do canal.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 44
Funcionalidade: Remover todos os itens
Plataforma: Plataforma de gestão para clientes - web
Ações: O utilizador ao estar no menu de editar a planta do canal, clica no eliminar todos os itens.
Resultado esperado: Todos os itens pertencentes ao canal em questão são apagados.
Resultado obtido: Não está implementado.
Resultado do teste: Falhou

ID Teste: 45
Funcionalidade: Criar nova região
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente ao clicar no mapa, começa a desenhar uma região. Esta região apenas pode ser gravada quando tem um mínimo de três pontos definidos.
Resultado esperado: É criada uma nova região no canal por parte do cliente.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 46
Funcionalidade: Apagar uma região
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente selecciona a região que deseja apagar, podendo eliminá-la ao clicar num botão (remover região).
Resultado esperado: É eliminada uma região do canal por parte do cliente.
Resultado obtido: Não está implementado.
Resultado do teste: Falhou

ID Teste: 47
Funcionalidade: Editar uma região
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente selecciona a região que deseja editar, podendo alterar os pontos definidos dessa região ou criar novos pontos para a região, guardando no final.
Resultado esperado: É editada uma região do canal por parte do cliente.
Resultado obtido: Não está implementado.
Resultado do teste: Falhou

ID Teste: 48
Funcionalidade: Adicionar novo item a uma região
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente ao clicar numa região, cria um novo item. A este item tem de ser atribuído um título podendo ser guardado no final.
Resultado esperado: É adicionado um novo item a uma região.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 49
Funcionalidade: Remover item de uma região
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente ao seleccionar um item aparecem várias hipóteses, de onde escolher apagar.
Resultado esperado: O item é eliminado da região por parte do cliente.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 50
Funcionalidade: Editar posição de um item numa região
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente muda um item de posição na mesma região.
Resultado esperado: O item é alterado de posição, sendo guardada a sua nova posição.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 51
Funcionalidade: Mudar item de região
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente muda um item de região.
Resultado esperado: O item é alterado de posição e de região, sendo guardada a sua nova posição. De notar que não é permitido o cliente deixar o item fora de uma região.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 52
Funcionalidade: Editar dados de um item
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente ao seleccionar um item e clicando em editar, é-lhe permitido alterar o nome do item.
Resultado esperado: O título do item seleccionado pelo cliente é alterado.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 53
Funcionalidade: Adicionar conteúdo a um item
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente selecciona um item e clica em “+” para poder adicionar um novo conteúdo ao item seleccionado. Após isso, tem de escrever um título, escolher o tipo de conteúdo que deseja adicionar e adiciona o conteúdo que pretende, gravando no final.
Resultado esperado: É adicionado um novo conteúdo a um item pelo cliente.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 54
Funcionalidade: Remover conteúdo de um item
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente selecciona um item e clica no conteúdo que pretende eliminar. Ao clicar em eliminar, o conteúdo é eliminado.
Resultado esperado: É removido um conteúdo de um item pelo cliente.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 55
Funcionalidade: Editar conteúdo de um item
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente selecciona um item e clica no conteúdo que pretende editar. Ao clicar em editar, é possível editar o título, o conteúdo ou ambas as coisas.
Resultado esperado: É editado um conteúdo de um item pelo cliente.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 56
Funcionalidade: Procurar conteúdos por nome num item
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente faz uma procura por título de conteúdo.
Resultado esperado: O sistema filtra os resultados da listagem dos conteúdos, consoante o que o cliente escrever na procura.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 57
Funcionalidade: Visualização do portal público através de um URL público
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente ao aceder ao portal público, clica no ver portal.
Resultado esperado: É mostrada uma nova página ao cliente contendo todos os conteúdos do portal público, caso existam.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 58
Funcionalidade: Edição das permissões do portal público
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente ao aceder ao portal público, selecciona quais as permissões que quer activas, guardando no final as opções que deseja activas.
Resultado esperado: As opções seleccionadas pelo cliente são guardadas.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 59
Funcionalidade: Adicionar novo conteúdo ao portal público
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente ao aceder ao portal público, escreve um título selecciona o tipo de conteúdo e coloca o conteúdo que deseja adicionar.
Resultado esperado: O conteúdo adicionado pelo cliente, é adicionado ao portal público do canal. Caso não preencha correctamente algum campo é notificado.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 60
Funcionalidade: Procura de conteúdos no portal público
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente ao aceder ao portal público, selecciona o editar e procura o conteúdo pelo tipo que deseja.
Resultado esperado: São apresentados os resultados, desde que existam, da procura desejada.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 61
Funcionalidade: Editar conteúdo do portal público
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente ao aceder ao portal público, selecciona o editar e procura o conteúdo pelo tipo que deseja. Caso existam conteúdos, pode editar o que deseja, tendo de guardar no final para as alterações terem efeito.
Resultado esperado: O conteúdo é editado sendo enviado uma mensagem de sucesso ao cliente.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 62
Funcionalidade: Apagar conteúdo do portal público
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente ao aceder ao portal público, selecciona o editar e procura o conteúdo pelo tipo que deseja. Caso existam conteúdos, pode apagar o que deseja.
Resultado esperado: O conteúdo é apagado sendo enviado uma mensagem de sucesso ao cliente.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 63
Funcionalidade: Edição das permissões do portal privado
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente ao aceder ao portal privado, selecciona quais as permissões que quer activas, guardando no final as opções que deseja activas.
Resultado esperado: As opções seleccionadas pelo cliente são guardadas.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 64
Funcionalidade: Adicionar novo conteúdo ao portal privado
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente ao aceder ao portal privado, escreve um título selecciona o tipo de conteúdo e coloca o conteúdo que deseja adicionar.
Resultado esperado: O conteúdo adicionado pelo cliente, é adicionado ao portal privado do canal. Caso não preencha correctamente algum campo é notificado.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 65
Funcionalidade: Procura de conteúdos no portal privado
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente ao aceder ao portal privado, selecciona o editar e procura o conteúdo pelo tipo que deseja.
Resultado esperado: São apresentados os resultados, desde que existam, da procura desejada.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 66
Funcionalidade: Editar conteúdo do portal privado
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente ao aceder ao portal privado, selecciona o editar e procura o conteúdo pelo tipo que deseja. Caso existam conteúdos, pode editar o que deseja, tendo de guardar no final para as alterações terem efeito.
Resultado esperado: O conteúdo é editado sendo enviado uma mensagem de sucesso ao cliente.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 67
Funcionalidade: Apagar conteúdo do portal privado
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente ao aceder ao portal privado, selecciona o editar e procura o conteúdo pelo tipo que deseja. Caso existam conteúdos, pode apagar o que deseja.
Resultado esperado: O conteúdo é apagado sendo enviado uma mensagem de sucesso ao cliente.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 68
Funcionalidade: Editar permissões de um canal
Plataforma: Plataforma de gestão para clientes - web
Ações: O cliente ao aceder ao canal, clica em “permissões do canal” e selecciona quais as permissões que quer activas, guardando no final as opções que deseja activas.
Resultado esperado: As opções seleccionadas pelo cliente são guardadas.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 69
Funcionalidade: Visualização de relatórios
Plataforma: Plataforma de gestão para clientes - web
Ações: Clicar no menu de relatórios.
Resultado esperado: São mostradas estatísticas de todos os canais que o cliente detenha.
Resultado obtido: Não está implementado.
Resultado do teste: Falhou

ID Teste: 70
Funcionalidade: Exportar relatórios
Plataforma: Plataforma de gestão para clientes - web
Acções: Ir ao menu de relatórios e clicar em exportar relatório.
Resultado esperado: É perguntado ao cliente onde deseja guardar o relatório, sendo este gravado onde o cliente deseja.
Resultado obtido: Não está implementado.
Resultado do teste: Falhou

ID Teste: 71
Funcionalidade: Autenticação por rede social (Facebook)
Plataforma: Plataforma móvel
Acções: O utilizador clica na autenticação por Facebook para poder entrar na aplicação.
Resultado esperado: A autenticação é efectuada com sucesso, deixando o utilizador entrar na aplicação.
Resultado obtido: Não está implementado.
Resultado do teste: Falhou

ID Teste: 72
Funcionalidade: Autenticação por conta anónima
Plataforma: Plataforma móvel
Acções: O utilizador clica na autenticação por conta anónima para poder entrar na aplicação.
Resultado esperado: A autenticação é efectuada com sucesso, deixando o utilizador entrar na aplicação. Nota: é necessário o caso de teste 73 ser executado com sucesso.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 73
Funcionalidade: Mostrar/Aceitar termos e condições
Plataforma: Plataforma móvel
Acções: O utilizador tem de aceitar os termos e condições para poder registar-se pela primeira vez na aplicação móvel.
Resultado esperado: A autenticação é efectuada com sucesso, deixando o utilizador entrar na aplicação.
Resultado obtido: Não está implementado.
Resultado do teste: Falhou

ID Teste: 74
Funcionalidade: Mostrar lista de canais que estão à volta do utilizador
Plataforma: Plataforma móvel
Acções: Autenticar-se na plataforma ou com conta do facebook ou com conta anónima.
Resultado esperado: É mostrada uma lista, ao utilizador, com canais que estão no seu raio de acção (tipicamente 5Km no máximo).
Resultado obtido: Não está implementado.
Resultado do teste: Falhou

ID Teste: 75
Funcionalidade: Mostrar lista com todos os canais
Plataforma: Plataforma móvel
Acções: Autenticar-se na plataforma ou com conta do facebook ou com conta anónima.
Resultado esperado: É mostrada uma lista, ao utilizador, com todos os canais existentes.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 76
Funcionalidade: Procurar canal por nome na lista de canais
Plataforma: Plataforma móvel
Acções: O utilizador pesquisa canais por nome.
Resultado esperado: É devolvida a lista de canais filtrada consoante o que o utilizador escreva e caso exista algum resultado. Quando não existem resultados, o utilizador é informado com a mensagem “Não existem canais com a procura que pretende”.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 77
Funcionalidade: Visualização do portal público
Plataforma: Plataforma móvel
Acções: Seleccionar um canal da lista de canais.
Resultado esperado: É apresentado ao utilizador, uma lista com todos os conteúdos presentes no portal público.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 78
Funcionalidade: Visualização do portal privado
Plataforma: Plataforma móvel
Acções: Seleccionar um canal da lista de canais e clicar no portal privado, quando disponível.
Resultado esperado: É apresentado ao utilizador, uma lista com todos os conteúdos presentes no portal público. Nota: é necessário o caso de teste 88 ser executado com sucesso.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 79
Funcionalidade: Comentar conteúdo de um portal (texto)
Plataforma: Plataforma móvel
Acções: Escolher um conteúdo de um portal, escrever o texto e clicar em enviar.
Resultado esperado: O comentário é enviado para aprovação por parte do cliente que detêm o canal.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 80
Funcionalidade: Visualização de comentários de um conteúdo de um portal
Plataforma: Plataforma móvel
Acções: Seleccionar um canal da lista de canais, clicando depois num dos portais. Caso existam mais do que dois comentários, o utilizador pode clicar em “carregar mais comentários”.
Resultado esperado: Visualização dos últimos comentários dois comentários se apenas for seleccionado o portal onde deseja ver os comentários. Caso o utilizador clique em “carregar mais comentários”, é mostrado todos os comentários ao conteúdo onde clicou.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 81
Funcionalidade: Visualização da planta de um canal
Plataforma: Plataforma móvel
Açções: Seleccionar um canal da lista de canais, ir ao portal privado e clicar em “ver planta”.
Resultado esperado: É mostrado a planta do canal ao utilizador.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 82
Funcionalidade: Visualização das regiões pertencentes a um canal
Plataforma: Plataforma móvel
Açções: Seleccionar um canal da lista de canais, ir ao portal privado e clicar em “ver planta”.
Resultado esperado: É mostrado as regiões na planta do canal ao utilizador.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 83
Funcionalidade: Visualização dos itens que estão numa região de um canal
Plataforma: Plataforma móvel
Açções: Seleccionar um canal da lista de canais, ir ao portal privado e clicar em “ver planta”.
Resultado esperado: É mostrado os itens que estão nas regiões da planta do canal ao utilizador.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 84
Funcionalidade: Visualizar o conteúdo de um item
Plataforma: Plataforma móvel
Açções: Escolher um canal da lista de canais, ir ao portal privado, clicar em “ver planta” e seleccionar um item.
Resultado esperado: É mostrado ao utilizador o conteúdo que o item escolhido tem.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 85
Funcionalidade: Comentar conteúdo de um item (texto)
Plataforma: Plataforma móvel
Acções: Escolher um canal da lista de canais, ir ao portal privado, clicar em “ver planta” e seleccionar um item. Escolher um conteúdo de um item, escrever o texto e clicar em enviar.
Resultado esperado: O comentário é enviado para aprovação por parte do cliente que detêm o canal.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 86
Funcionalidade: Visualização dos comentários a um conteúdo de um item
Plataforma: Plataforma móvel
Acções: Escolher um canal da lista de canais, ir ao portal privado, clicar em “ver planta” e seleccionar um item. Caso existam mais do que dois comentários, o utilizador pode clicar em “carregar mais comentários”.
Resultado esperado: Visualização dos últimos comentários dois comentários se apenas for seleccionado o item onde se deseja ver os comentários. Caso o utilizador clique em “carregar mais comentários”, é mostrado todos os comentários ao conteúdo onde clicou.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 87
Funcionalidade: Navegação na planta
Plataforma: Plataforma móvel
Acções: O utilizador pode fazer <i>zoom in</i> ou <i>zoom out</i> , rodar o mapa, mover-se “livremente” pelo mapa ou clicar nos itens.
Resultado esperado: Consoante a acção que o utilizador execute a aplicação deixa o utilizador fazer <i>zoom in</i> ou <i>zoom out</i> , rodar o mapa, mover-se livremente pelo mapa ou clicar nos itens.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 88
Funcionalidade: Leitura de QR-Code ou introduzir um código para poder visualizar o portal privado
Plataforma: Plataforma móvel
Acções: O utilizador tem de ler um QR-Code ou introduzir um código para poder entrar no portal privado.
Resultado esperado: Se a leitura do QR-Code ou a introdução do código estiver certa, o utilizador poderá ver os conteúdos presentes no portal privado de um determinado canal.
Resultado obtido: Não está implementado.
Resultado do teste: Falhou

ID Teste: 89
Funcionalidade: Comentar conteúdo de um portal (foto da galeria)
Plataforma: Plataforma móvel
Acções: Escolher um conteúdo de um portal, clicar no botão para enviar foto e seleccionar “escolher da galeria”. Escolher uma foto e clicar em enviar.
Resultado esperado: O comentário é enviado para aprovação por parte do cliente que detêm o canal e é mostrado uma mensagem ao utilizador a informar que o comentário foi enviado para aprovação.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 90
Funcionalidade: Comentar conteúdo de um portal (tirar foto)
Plataforma: Plataforma móvel
Acções: Escolher um conteúdo de um portal, clicar no botão para enviar foto e seleccionar “tirar foto”. O utilizador tira uma foto e clicar em enviar.
Resultado esperado: O comentário é enviado para aprovação por parte do cliente que detêm o canal e é mostrado uma mensagem ao utilizador a informar que o comentário foi enviado para aprovação.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 91
Funcionalidade: Comentar conteúdo de um item (foto da galeria)
Plataforma: Plataforma móvel
Acções: Escolher um canal da lista de canais, ir ao portal privado, clicar em “ver planta” e seleccionar um item. Escolher um conteúdo de um item, clicar no botão para enviar foto e seleccionar “escolher da galeria”. Escolher uma foto e clicar em enviar.
Resultado esperado: O comentário é enviado para aprovação por parte do cliente que detêm o canal e é mostrado uma mensagem ao utilizador a informar que o comentário foi enviado para aprovação.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 92
Funcionalidade: Comentar conteúdo de um item (tirar foto)
Plataforma: Plataforma móvel
Acções: Escolher um canal da lista de canais, ir ao portal privado, clicar em “ver planta” e seleccionar um item. Escolher um conteúdo de um item, clicar no botão para enviar foto e seleccionar “tirar foto”. O utilizador tira uma foto e clicar em enviar.
Resultado esperado: O comentário é enviado para aprovação por parte do cliente que detêm o canal e é mostrado uma mensagem ao utilizador a informar que o comentário foi enviado para aprovação.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

E.1.2 – Repetições

ID Teste: 12
Funcionalidade: Apagar/desactivar a própria conta
Plataforma: Plataforma de gestão administrativa - web
Acções: O utilizador quer apagar/desactivar a própria conta.
Resultado esperado: Quando o utilizador apaga a sua própria conta, é reencaminhado para a página de login não podendo mais aceder à sua conta até alguém a activar.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 17
Funcionalidade: Apagar/desactivar a conta de um utilizador
Plataforma: Plataforma de gestão administrativa - web
Acções: O administrador faz reset à password do perfil de um utilizador.
Resultado esperado: É enviada uma mensagem de sucesso sucedido de um email para o email registado no perfil do utilizador.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 18
Funcionalidade: Reset da password do perfil de um utilizador
Plataforma: Plataforma de gestão administrativa - web
Acções: O administrador faz reset à password do perfil de um utilizador.
Resultado esperado: É enviada uma mensagem de sucesso sucedido de um email para o email registado no perfil do utilizador.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 20
Funcionalidade: Visualizar relatórios
Plataforma: Plataforma de gestão administrativa - web
Acções: O administrador acede ao menu de relatórios.
Resultado esperado: O administrador ao aceder ao menu de relatórios, visualiza relatórios de todo o sistema.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 21
Funcionalidade: Exportar relatórios
Plataforma: Plataforma de gestão administrativa - web
Acções: No menu de relatórios, o administrador clica em exportar.
Resultado esperado: O relatório é exportado em csv para onde o administrador desejar.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 31
Funcionalidade: Apagar a própria conta
Plataforma: Plataforma de gestão para clientes - web
Acções: O utilizador quer apagar/desactivar a própria conta.
Resultado esperado: Quando o utilizador apaga a sua própria conta, é reencaminhado para a página de login não podendo mais aceder à sua conta até alguém a activar.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 34
Funcionalidade: Apagar a conta de um gestor
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente faz reset à password do perfil de um gestor.
Resultado esperado: É enviada uma mensagem de sucesso sucedido de um email para o email registado no perfil do utilizador.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 44
Funcionalidade: Remover todos os itens
Plataforma: Plataforma de gestão para clientes - web
Acções: O utilizador ao estar no menu de editar a planta do canal, clica no eliminar todos os itens.
Resultado esperado: Todos os itens pertencentes ao canal em questão são apagados.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 46
Funcionalidade: Apagar uma região
Plataforma: Plataforma de gestão para clientes - web
Acções: O cliente selecciona a região que deseja apagar, podendo eliminá-la ao clicar num botão (remover região).
Resultado esperado: É eliminada uma região do canal por parte do cliente.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 47
Funcionalidade: Editar uma região
Plataforma: Plataforma de gestão para clientes - web
Açções: O cliente selecciona a região que deseja editar, podendo alterar os pontos definidos dessa região ou criar novos pontos para a região, guardando no final.
Resultado esperado: É editada uma região do canal por parte do cliente.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 69
Funcionalidade: Visualização de relatórios
Plataforma: Plataforma de gestão para clientes - web
Açções: Clicar no menu de relatórios.
Resultado esperado: São mostradas estatísticas de todos os canais que o cliente detenha.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 70
Funcionalidade: Exportar relatórios
Plataforma: Plataforma de gestão para clientes - web
Açções: Ir ao menu de relatórios e clicar em exportar relatório.
Resultado esperado: É perguntado ao cliente onde deseja guardar o relatório, sendo este gravado onde o cliente deseja.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 71
Funcionalidade: Autenticação por rede social (Facebook)
Plataforma: Plataforma móvel
Açções: O utilizador clica na autenticação por Facebook para poder entrar na aplicação.
Resultado esperado: A autenticação é efectuada com sucesso, deixando o utilizador entrar na aplicação.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 73
Funcionalidade: Mostrar/Aceitar termos e condições
Plataforma: Plataforma móvel
Acções: O utilizador ao fazer login na aplicação aceita de imediato os termos e condições, podendo desta forma registar-se pela primeira vez na aplicação móvel.
Resultado esperado: A autenticação é efectuada com sucesso, deixando o utilizador entrar na aplicação.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 74
Funcionalidade: Mostrar lista de canais que estão à volta do utilizador
Plataforma: Plataforma móvel
Acções: Autenticar-se na plataforma ou com conta do facebook ou com conta anónima.
Resultado esperado: É mostrada uma lista, ao utilizador, com canais que estão no seu raio de acção (<i>default</i> 2000 metros). Posteriormente o utilizador pode aumentar ou diminuir este filtro.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

ID Teste: 88
Funcionalidade: Leitura de QR-Code ou introduzir um código para poder visualizar o portal privado
Plataforma: Plataforma móvel
Acções: O utilizador tem de ler um QR-Code ou introduzir um código para poder entrar no portal privado.
Resultado esperado: Se a leitura do QR-Code ou a introdução do código estiver certa, o utilizador poderá ver os conteúdos presentes no portal privado de um determinado canal.
Resultado obtido: Igual ao esperado.
Resultado do teste: Sucesso

E.2 – Testes de Aceitação

E.2.1 – Requisitos Funcionais

ID	Requisito funcional	Classificação	Verificado
RF.1	O sistema deve permitir um utilizador L1 autenticar-se com <i>username</i> e <i>password</i> .	Must	Sucesso
RF.2	O sistema deve possibilitar o acesso a administração de canais/clientes.	Must	Sucesso
RF.3	O nível de acesso de super-administrador permite o acesso e controlo de todos os componentes.	Must	Sucesso
RF.4	O nível de acesso de administrador permite o acesso a: gestão de clientes/gestores/utilizadores e gestão de canais.	Must	Sucesso
RF.5	Todos os acessos ao sistema devem ser registados para consulta futura.	Must	Sucesso
RF.6	Todas as criações de acessos devem ser registadas.	Must	Sucesso
RF.7	Todos os registos de segurança devem conter sempre: <i>timestamp</i> , autor, acção.	Must	Sucesso
RF.8	O super-administrador tem a capacidade para criar qualquer tipo de utilizador da plataforma.	Must	Sucesso
RF.9	O super-administrador tem a capacidade para criar qualquer tipo de utilizador da plataforma.	Must	Sucesso
RF.10	O super-administrador tem a capacidade para remover qualquer tipo de utilizador da plataforma.	Must	Sucesso
RF.11	O super-administrador tem a capacidade para editar qualquer tipo de utilizador da plataforma.	Must	Sucesso
RF.12	Os utilizadores são classificados em 3 níveis, L1 (Super-Administradores, Administradores), L2 (Clientes, Gestores), L3 (Utilizadores, Utilizadores Anónimos).	Must	Sucesso
RF.13	O perfil de administrador tem capacidade para criar utilizadores L2 dentro da plataforma.	Must	Sucesso
RF.14	O perfil de administrador tem capacidade para cessar utilizadores L2 dentro da plataforma.	Must	Sucesso
RF.15	O perfil de administrador tem capacidade para editar utilizadores L2 dentro da plataforma.	Must	Sucesso
RF.16	Tem de ser possível criar clientes descrevendo os mesmos com as seguintes informações: - Nome; - Morada; - Data Nascimento/Criação; - Documento de Identificação; - Contactos Telefónicos; - Contactos de Email; - URL WEB; Identificação Fiscal; - Descrição Geral.	Must	Sucesso

ID	Requisito funcional	Classificação	Verificado
RF.17	Tem de ser possível fazer reset às credenciais de acesso de clientes/gestores respeitando sempre a confidencialidade sendo as mesmas enviadas para o cliente/gestor diretamente.	Must	Sucesso
RF.18	Os utilizadores L1 podem assumir a identidade de um utilizador L2.	Must	Sucesso
RF.19	O utilizador L1 tem a capacidade para criar canais associados a um cliente	Must	Sucesso
RF.20	O utilizador L1 tem a capacidade para cessar um canal, sendo sempre necessário indicar o motivo de tal.	Must	Sucesso
RF.21	O canal tem de ser caracterizado pela seguinte informação: - Título; Descrição; - Data de criação.	Must	Sucesso
RF.22	Um canal tem de ter associado a si um local ou mais locais que limitam a acção do mesmo.	Must	Sucesso
RF.23	Um canal pode ter múltiplos gestores associados.	Must	Sucesso
RF.24	Cada canal pode ter um portal privado e um portal público.	Must	Sucesso
RF.25	O portal privado pode ser acedido por qualquer tipo de utilizador, desde que estejam na área geográfica da acção do mesmo.	Must	Sucesso
RF.26	O portal privado pode ser acedido por qualquer tipo de utilizador L3.	Must	Sucesso
RF.27	O conteúdo Multimédia pode ser composto pelos seguintes objectos: Imagens; - Vídeos; - Sons; -Texto; - Links.	Must	Sucesso
RF.28	Deve ser suportado o youtube como um plataforma externa de vídeo.	Must	Sucesso
RF.29	Aceitar pedidos para adicionar canais.	Must	Sucesso
RF.30	Os acessos à API devem poder ser feitos de modo encriptado.	Must	Sucesso
RF.31	O acesso à API deve ser devidamente autenticado, recorrendo à utilização de <i>tokens</i> .	Must	Sucesso
RF.32	Tem de ser possível obter estatísticas/relatórios detalhados da utilização de armazenamento.	Must	Sucesso
RF.33	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade de canais.	Must	Sucesso
RF.34	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade itens.	Must	Sucesso
RF.35	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade acessos.	Must	Sucesso

ID	Requisito funcional	Classificação	Verificado
RF.36	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade de gestores.	Must	Sucesso
RF.37	Relatório exportáveis em CSV.	Must	Sucesso

Tabela 23: Testes de aceitação dos requisitos funcionais (must) para a aplicação de gestão administrativa

ID	Requisito funcional	Classificação	Verificado
RF.38	O utilizador L1 tem a capacidade para remover todo o conteúdo de um canal, sendo necessário indicar o motivo para tal.	Should	Sucesso
RF.39	O objecto vídeo deve suportar tanto o armazenamento local do vídeo, bem como a representação do mesmo através de plataformas externas de vídeo.	Should	Falhou
RF.40	O objecto audio deve suportar tanto o armazenamento local do audio, bem como a representação do mesmo através de plataformas externas de audio.	Should	Falhou
RF.41	Relatório exportáveis em PDF, XML e JSON.	Should	Falhou

Tabela 24: Testes de aceitação dos requisitos funcionais (should) para a aplicação de gestão administrativa

ID	Requisito funcional	Classificação	Verificado
RF.42	Os dados pessoais como por exemplo o nome, morada, telefone e email, devem poder se encriptados no armazenamento de dados.	Nice to Have	Falhou
RF.43	Qualquer utilizador L1 deve poder ter múltiplos perfis associados à sua conta.	Nice to Have	Falhou
RF.44	Deve ser suportado o vimeo como um plataforma externa de video.	Nice to Have	Falhou
RF.45	Um canal deve poder ser classificado quanto a conteúdos adultos ou não.	Nice to Have	Falhou
RF.46	O objecto imagem deve suportar tanto o armazenamento local da imagem, bem como a representação da mesma através de plataformas externas de imagem.	Nice to Have	Falhou
RF.47	Deve ser suportado o <i>instagram</i> como plataforma externa de imagens.	Nice to Have	Falhou
RF.48	Deve ser suportado o <i>soundcloud</i> como plataforma externa de áudio.	Nice to Have	Falhou
RF.49	Tem de ser possível obter estatísticas/relatórios detalhados dos acessos de gestão à plataforma.	Nice to Have	Falhou

ID	Requisito funcional	Classificação	Verificado
RF.50	Todas as estatísticas devem poder ser geradas aglomerando um ou múltiplos utilizadores, e devem ser temporais e com uma granularidade configurável.	Nice to Have	Falhou

Tabela 25: Testes de aceitação dos requisitos funcionais (nice to have) para a aplicação de gestão administrativa

ID	Requisito funcional	Classificação	Verificado
RF.51	Qualquer utilizador L2 apenas pode visualizar canais, portais, mapas, itens, multimédia respeitantes ao canal/canais que lhe estão atribuídos.	Must	Sucesso
RF.52	O sistema deve permitir um cliente ou gestor autenticar-se com <i>username</i> e <i>password</i> .	Must	Sucesso
RF.53	Podem existir múltiplos gestores para um só canal.	Must	Sucesso
RF.54	O perfil de cliente tem todos os privilégios que um gestor tem.	Must	Sucesso
RF.55	Possibilitar o acesso a administração de canais/gestores.	Must	Sucesso
RF.56	Apenas o cliente tem acesso à criação de gestores.	Must	Sucesso
RF.57	Todas as operações/interacções dos clientes/gestores devem ser devidamente registadas.	Must	Sucesso
RF.58	Todos os registos de segurança devem conter sempre: <i>Timestamp</i> , Autor, Acção.	Must	Sucesso
RF.59	Tem de ser possível criar gestores descrevendo os mesmos com as seguintes informações: Nome; Morada; Data Nascimento/Criação; Documento de Identificação; Contactos Telefónicos; Contactos de Email.	Must	Sucesso
RF.60	O sistema deve permitir ao L2 editar as permissões (de visualização) do seu canal.	Must	Sucesso
RF.61	Tem de ser possível fazer <i>reset</i> às credenciais de acesso de gestores respeitando sempre a confidencialidade sendo as mesmas enviadas para o gestor diretamente.	Must	Sucesso
RF.62	O sistema deve permitir ao L2 editar as permissões (de visualização) dos portais.	Must	Sucesso
RF.63	Um cliente pode criar elementos numa região.	Must	Sucesso
RF.64	Cada portal tem de ser personalizável com os seguintes conteúdos: - Plantas; - Multimédia.	Must	Sucesso
RF.65	O objecto planta tem como base uma imagem rectangular.	Must	Sucesso
RF.66	O objecto planta tem de ser passível de ser dividido em regiões, regiões essas definidas por um polígono.	Must	Sucesso

ID	Requisito funcional	Classificação	Verificado
RF.67	O objecto planta irá assumir que o interior das regiões é considerado a zona onde a planta efectivamente é reconhecida.	Must	Sucesso
RF.68	As plantas do portal são acessíveis quando abandonamos o mesmo e nos mantemos dentro do canal.	Must	Sucesso
RF.69	Dentro de cada canal temos a capacidade de ter múltiplos itens definidos.	Must	Sucesso
RF.70	um local é um ponto geo-referenciado ou um polígono (area) georreferenciada utilizado para caracterizar diferentes componentes desta plataforma.	Must	Sucesso
RF.71	Um item é caracterizado pelo seguinte conteúdo: Local; Título; Multimédia.	Must	Sucesso
RF.72	Permitir fazer um pedido aos administradores para a criação de um novo canal.	Must	Sucesso
RF.73	O sistema deve permitir ao cliente editar os dados/conteúdos multimédia de cada portal.	Must	Sucesso
RF.74	Um canal tem de ter a seguinte informação: - Título; Descrição; - Data de criação.	Must	Sucesso
RF.75	Os itens são georreferenciados nas regiões definidas nas plantas.	Must	Sucesso
RF.76	Cada planta deve permitir ao L2 fazer regiões na mesma.	Must	Sucesso
RF.77	Editar conteúdo multimédia associado aos itens.	Must	Sucesso
RF.78	O sistema deve permitir ao cliente moderar os utilizadores do canal.	Must	Sucesso
RF.79	O sistema deve permitir uma gestão do canal por parte do L2 do canal.	Must	Sucesso
RF.80	O sistema deve permitir ao cliente editar os dados de um canal.	Must	Sucesso
RF.81	Tem de ser possível obter estatísticas/relatórios detalhados da utilização de armazenamento	Must	Sucesso
RF.82	Tem de ser possível obter estatísticas/relatórios detalhados da utilização de quantidade de canais.	Must	Sucesso
RF.83	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade itens.	Must	Sucesso
RF.84	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade de gestores.	Must	Sucesso
RF.85	O sistema deve permitir ao L2 ver estatísticas do uso do canal.	Must	Sucesso

Tabela 26: Testes de aceitação dos requisitos funcionais (must) para a aplicação de gestão para clientes

ID	Requisito funcional	Classificação	Verificado
RF.86	Tem de ser possível obter estatísticas/relatórios detalhados da quantidade acessos.	Should	Falhou
RF.87	Tem de ser possível obter estatísticas sociais sobre os utilizadores de nível L3 baseando-se na informação que conseguimos recolher nas plataformas móveis fornecendo assim dados relevantes e completos sobre o perfil dos mesmos.	Should	Falhou
RF.88	O sistema deve permitir ver o portal público de um canal através de um URL publicamente acessível.	Should	Sucesso
RF.89	Tem de ser possível obter estatísticas/relatórios detalhados das zonas mais visitadas no seu canal.	Should	Falhou
RF.90	O sistema deve permitir ao L2 ver gráficos das estatísticas.	Should	Sucesso
RF.91	Tem de ser possível obter estatísticas/relatórios detalhados dos acessos de gestão à plataforma.	Nice to Have	Falhou
RF.92	O sistema deve permitir pôr um espaço temporal associado a cada conteúdo multimédia. Caso o cliente não preencha o espaço temporal assume-se que o conteúdo ficará disponível para sempre. Caso dois conteúdos estejam no mesmo espaço temporal, prevalece o que tem menor espaço temporal.	Nice to Have	Falhou
RF.93	Todas as estatísticas devem poder ser geradas aglomerando um ou múltiplos canais, e devem ser temporais e com uma granularidade configurável.	Nice to Have	Falhou
RF.94	Relatório exportáveis em CSV.	Nice to Have	Sucesso
RF.94	Relatório exportáveis em PDF, XML e JSON.	Nice to Have	Falhou

Tabela 27: Testes de aceitação dos requisitos funcionais (should e nice to have) para a aplicação de gestão para clientes

ID	Requisito funcional	Classificação	Verificado
RF.95	O sistema deve permitir uma identificação do utilizador L3 através de redes sociais (facebook).	Must	Sucesso
RF.96	Caso o utilizador L3 não se queira identificar, poderá na mesma aceder à plataforma sendo criado um perfil anónimo. Este tipo de utilizador será tratado da mesma forma que um utilizador com perfil social.	Must	Sucesso
RF.97	O sistema deve pedir permissões para guardar a posição GPS do utilizador L3.	Must	Sucesso
RF.98	O sistema deve mostrar os termos de utilização quando um utilizador L3 aquando da criação da conta rápida.	Must	Sucesso
RF.99	O sistema tem de permitir que o utilizador saiba quais os canais que o rodeiam.	Must	Sucesso

ID	Requisito funcional	Classificação	Verificado
RF.100	O sistema tem de permitir que o utilizador consiga ver a informação do portal público dos canais que o rodeiam.	Must	Sucesso
RF.101	Assim que se entra num canal o portal público deve ser apresentado ao utilizador.	Must	Sucesso
RF.102	Quando o L3 se encontra numa região geo-referenciada pertencente ao canal em questão, o portal privado passa a estar visível.	Must	Sucesso
RF.103	Caso o utilizador L2 dê permissões, o utilizador de nível L3 pode comentar conteúdos multimédia.	Must	Sucesso
RF.104	Caso o utilizador L2 dê permissões, o utilizador de nível L3 pode introduzir conteúdo multimédia que necessitará da aprovação de L2.	Must	Sucesso
RF.105	Os comentários são constituídos por conteúdos multimédia.	Must	Sucesso
RF.106	O utilizador L3 poderá “navegar” pelas diversas plantas de modo a permitir dar-lhe uma visão global para onde pode ir e do que pode fazer.	Must	Sucesso
RF.107	Na navegação pelas plantas um utilizador L3 pode ver os itens que estão definidos na região.	Must	Sucesso
RF.108	O sistema deve permitir uma procura pelo nome do canal	Must	Sucesso
RF.109	O sistema deve permitir apresentar uma listagem de todos os canais existentes	Must	Sucesso
RF.110	O sistema deve permitir apresentar informações de um canal - detalhes/descrição do canal.	Must	Sucesso
RF.111	O sistema deve permitir a integração da aplicação com a API do facebook.	Must	Sucesso
RF.112	O sistema deve permitir a integração da aplicação com a API do Google.	Must	Sucesso

Tabela 28: Testes de aceitação dos requisitos funcionais (must) para a aplicação móvel

ID	Requisito funcional	Classificação	Verificado
RF.113	Acéder a conteúdo de outros canais que têm itens nas imediações, sem necessariamente sair do canal em que se encontra	Should	Falhou
RF.114	O sistema permite enviar estatísticas das acções que um utilizador L3 faz num canal.	Should	Falhou
RF.115	A aplicação deve adaptar-se graficamente recorrendo aos objectos gráficos que caracterizam o canal após a entrada no mesmo.	Nice to Have	Falhou
RF.116	O sistema deve permitir uma procura pelo tipo do canal	Nice to have	Falhou

Tabela 29: Testes de aceitação dos requisitos funcionais (should e nice to have) para a aplicação móvel

E.2.2 – Requisitos Não Funcionais

ID	Requisito não funcional	Classificação	Verificado
RNF1	O sistema deve permitir o acesso à aplicação de gestão administrativa através de <i>browser web</i> .	Must	Sucesso
RNF2	Os acessos de administração e de gestão devem sempre recorrer a protocolos encriptados.	Must	Sucesso
RNF3	Toda a plataforma tem de ser passível de ser suportada em <i>clouds</i> públicas sem recorrer a infra-estrutura dedicada para um determinado cliente (exceptua-se hardware necessário para a geo-localização).	Must	Sucesso
RNF4	A API tem de ser facilmente acessível usando http/https.	Must	Sucesso
RNF5	O sistema deve permitir o acesso à aplicação de gestão para clientes através de <i>browser web</i> .	Must	Sucesso
RNF6	O sistema deve permitir comunicações com ligações seguras através de encriptação, entre o cliente e o servidor.	Should	Falhou

Tabela 30: Testes de aceitação dos requisitos não funcionais para todas as aplicações

E.3 – Testes de Usabilidade

		Discordo totalmente			Concordo totalmente	
1	Penso que gostaria de usar esta aplicação frequentemente	1	2	3	4	5
2	Considero a aplicação desnecessariamente complexa	1	2	3	4	5
3	Penso que a aplicação foi fácil de utilizar	1	2	3	4	5
4	Considero que necessitaria de suporte técnico para poder utilizar a aplicação	1	2	3	4	5
5	Penso que as várias funções da aplicação estavam bem integradas	1	2	3	4	5
6	Penso que existiam demasiadas inconsistências na aplicação	1	2	3	4	5
7	Penso que a maioria das pessoas poderia aprender a utilizar a aplicação rapidamente	1	2	3	4	5
8	Considero que a aplicação é muito inconveniente do ponto de vista da utilização	1	2	3	4	5
9	Senti-me bastante confiante ao usar a aplicação	1	2	3	4	5
10	Precisaria de adquirir muitas competências antes de continuar a usar a aplicação	1	2	3	4	5

Figura 82: Questionário de usabilidade

Utilizador	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total
1	4	1	4	1	2	1	4	1	4	1	82,5
2	5	2	5	1	3	2	5	1	5	1	90
3	4	2	5	1	3	2	4	2	4	1	80
4	5	1	5	1	3	2	5	1	5	1	92,5
5	5	2	4	1	4	1	5	1	4	1	90
6	4	1	4	1	3	1	5	1	4	1	87,5
7	5	1	3	1	4	2	4	1	4	1	85
8	4	2	4	1	3	1	4	2	4	1	80
9	4	1	5	1	3	2	4	2	4	1	82,5
10	5	1	5	1	3	2	5	2	4	1	87,5
Média											85,75

Tabela 31: Teste de usabilidade na aplicação web a utilizadores da área informática

Utilizador	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total
1	5	2	4	2	4	2	5	1	4	1	85
2	4	3	4	3	3	2	4	2	3	2	65
3	4	3	4	3	3	3	4	2	3	2	62,5
4	5	2	5	2	4	1	5	1	5	1	92,5
5	4	3	4	3	3	3	4	1	4	1	70
6	4	3	4	2	3	2	4	2	4	2	70
7	4	2	3	2	3	3	5	2	3	2	67,5
8	4	2	5	2	3	2	4	1	3	1	77,5
9	5	2	5	1	4	1	5	2	4	1	90
10	5	1	5	2	4	3	4	2	4	1	82,5
Média											76,25

Tabela 32: Teste de usabilidade na aplicação web a utilizadores de outras áreas

Utilizador	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total
1	5	2	4	2	3	2	4	1	4	2	77,5
2	5	2	4	2	3	2	5	1	4	2	80
3	4	3	3	3	4	1	5	2	3	1	72,5
4	4	3	4	2	3	2	5	1	4	2	75
5	5	3	4	1	5	2	4	2	4	1	82,5
6	5	3	4	1	3	2	4	2	4	1	77,5
7	4	2	3	2	3	2	4	2	3	2	67,5
8	3	2	4	2	4	2	3	2	4	2	70
9	4	3	5	1	3	2	5	1	5	1	85
10	5	2	5	1	4	2	5	1	5	1	92,5
Média											78

Tabela 33: Teste de usabilidade na aplicação móvel a utilizadores da área informática

Utilizador	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total
1	4	3	3	2	4	1	2	2	4	2	67,5
2	3	3	4	2	4	2	3	2	4	2	67,5
3	5	2	4	3	3	1	3	2	4	1	75
4	3	2	4	2	4	1	2	1	3	2	70
5	4	2	4	3	3	2	2	1	3	2	65
6	5	3	3	3	3	2	3	2	4	2	65
7	4	3	3	2	4	2	3	1	3	1	70
8	3	2	4	2	4	2	3	2	4	2	70
9	4	4	5	3	4	2	2	2	4	2	65
10	4	3	5	2	4	1	2	1	5	1	80
Média											69,5

Tabela 34: Teste de usabilidade na aplicação móvel a utilizadores da área informática

E.4 – Testes de Performance

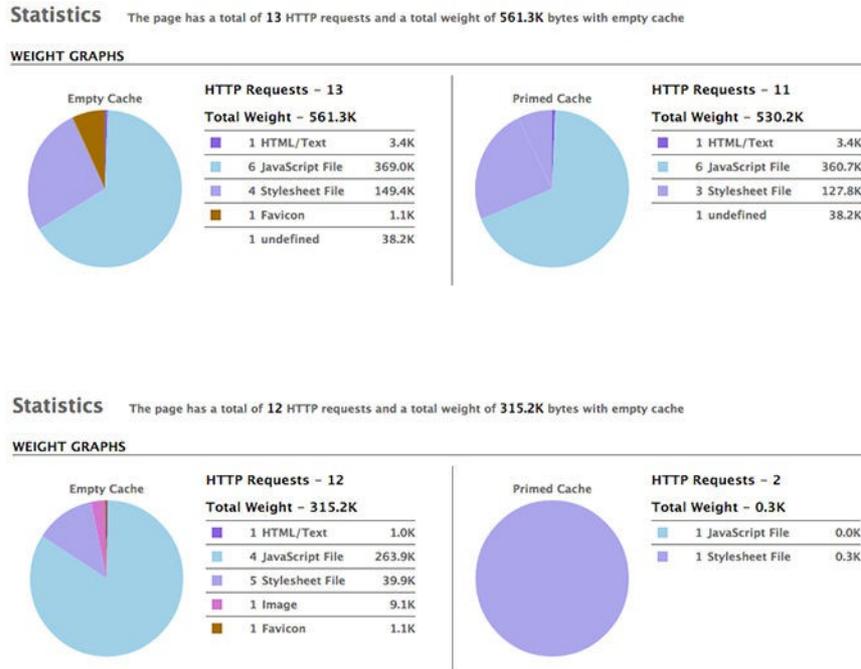


Figura 83: Acessos antes e depois de aplicadas as regras do Steve Souders

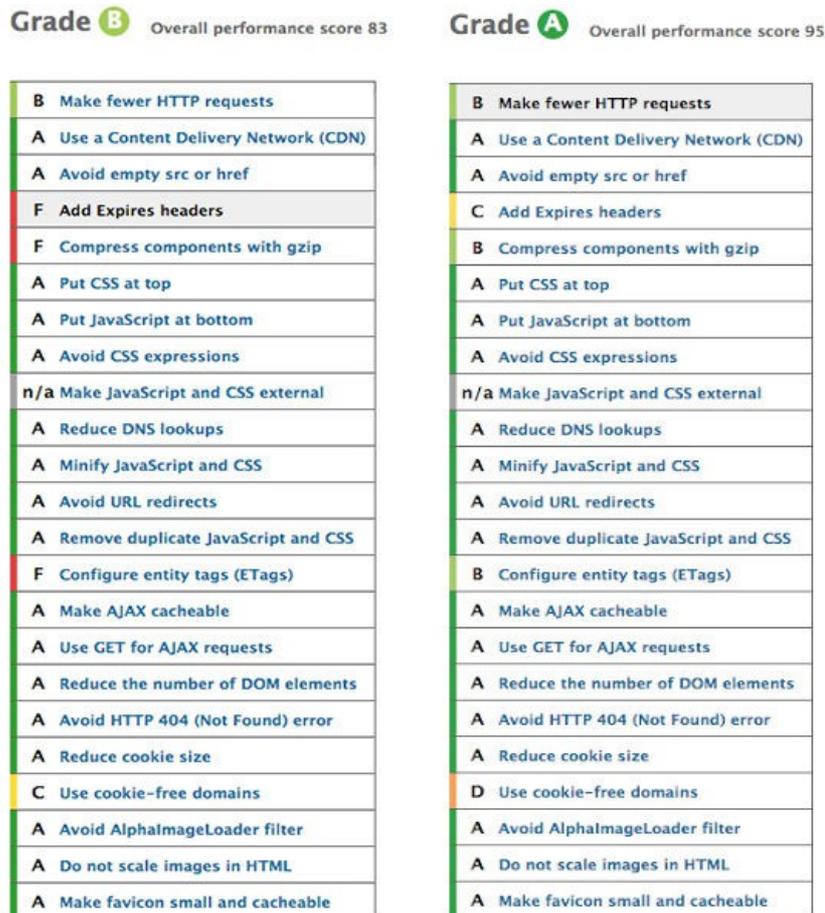


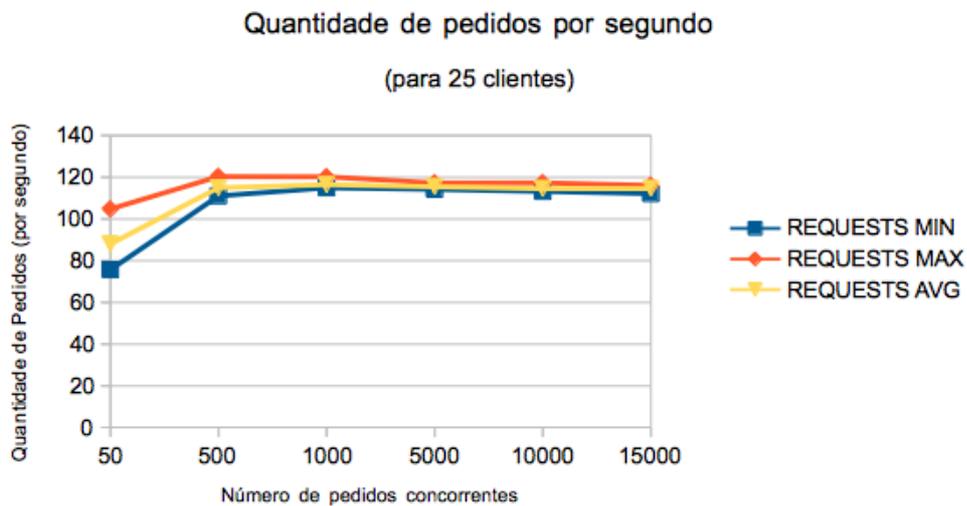
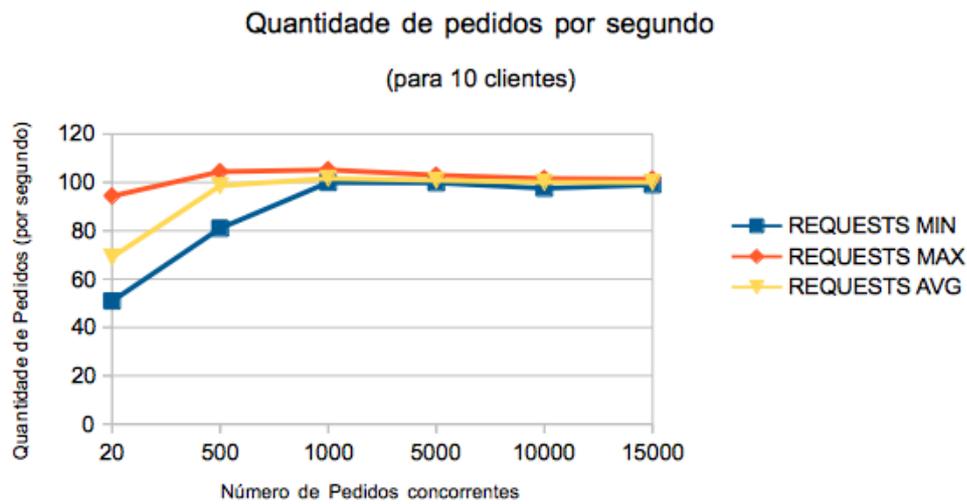
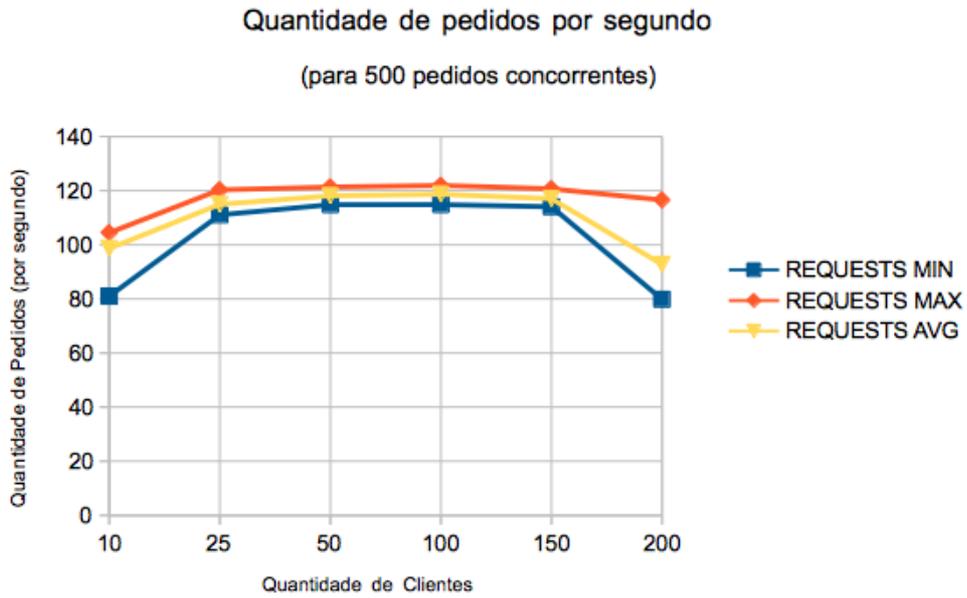
Figura 84: Avaliação da performance antes e depois de aplicadas as regras do Steve Souders

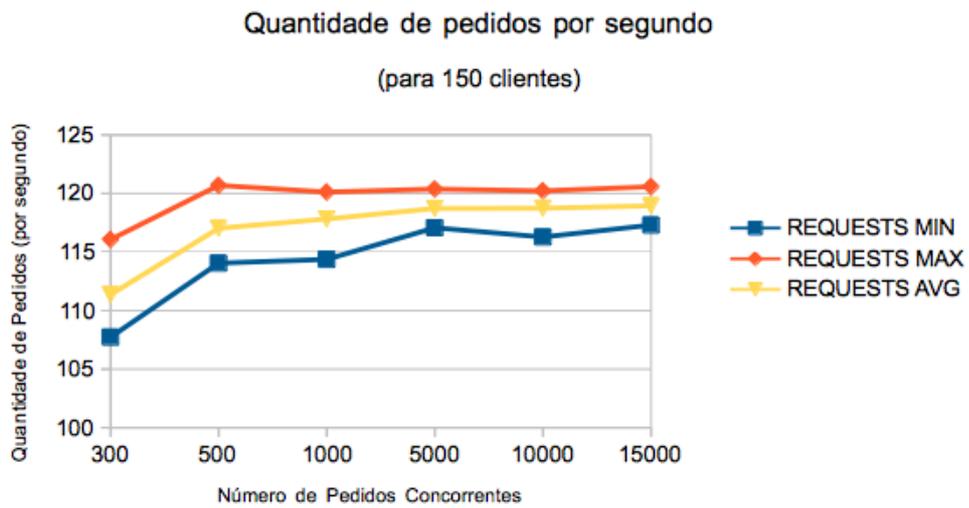
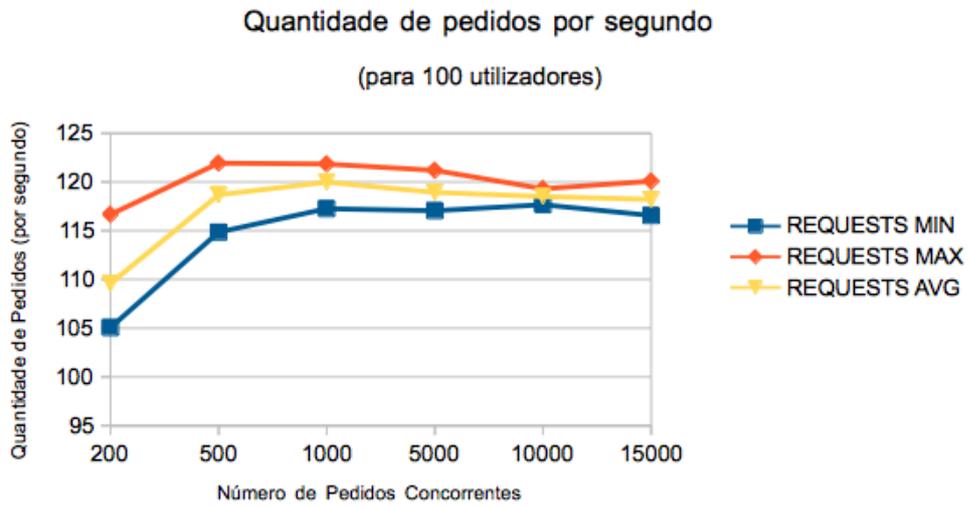
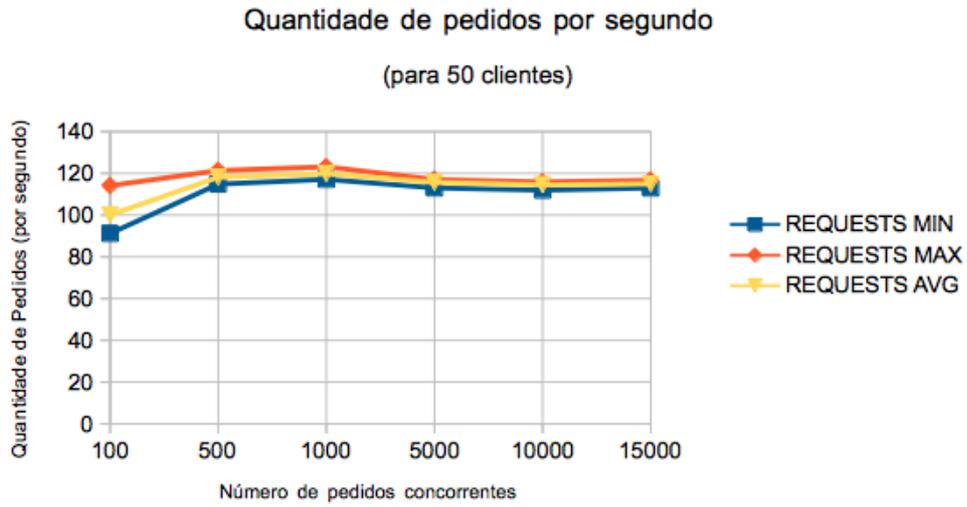
E.5 – Testes de Escalabilidade

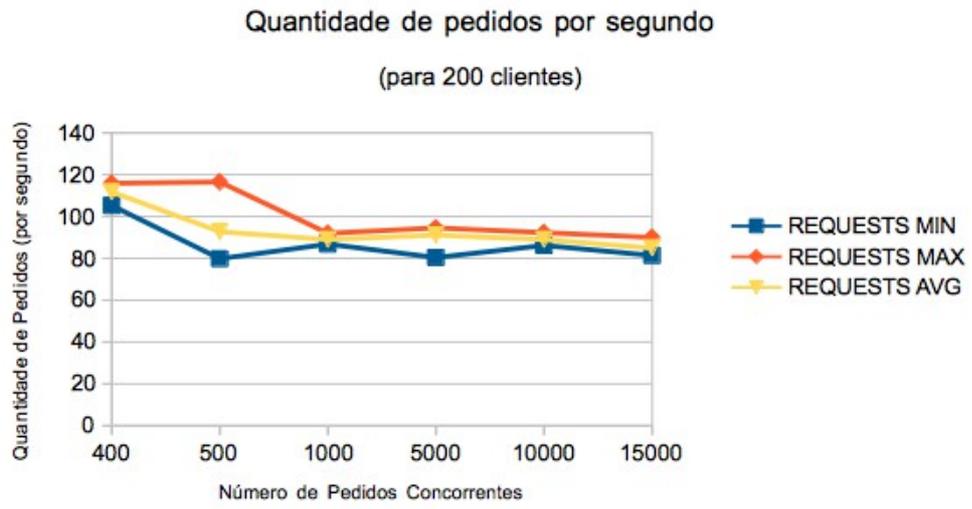
E.5.1 – Pedidos por segundo

TAG	Clients	Requests	Failed Request	Request AVG	REQUESTS MIN	REQUESTS MAX	REQUESTS AVG
1	10	20	0		50,87	94,32	69,072
2	25	50	0		75,69	104,75	88,031
3	50	100	0		91,18	114,09	100,066
4	100	200	0		105,06	116,67	109,585
5	150	300	0		107,73	116,07	111,366
6	200	400	0		105,25	115,87	111,759
TAG	Clients	Requests	Failed Request	Request AVG	REQUESTS MIN	REQUESTS MAX	REQUESTS AVG
7	10	500	0		80,98	104,43	98,623
8	25	500	0		111,03	120,31	114,946
9	50	500	0		114,82	121,32	118,108
10	100	500	0		114,85	121,93	118,677
11	150	500	0		114,05	120,69	117,029
12	200	500	0		79,85	116,53	92,749
TAG	Clients	Requests	Failed Request	Request AVG	REQUESTS MIN	REQUESTS MAX	REQUESTS AVG
13	10	1000	0		99,94	105,19	101,623
14	25	1000	0		114,91	120,21	116,423
15	50	1000	0		117,09	123,13	119,806
16	100	1000	0		117,26	121,84	119,981
17	150	1000	0		114,37	120,12	117,826
18	200	1000	0		86,81	91,97	88,975
TAG	Clients	Requests	Failed Request	Request AVG	REQUESTS MIN	REQUESTS MAX	REQUESTS AVG
19	10	5000	0		99,82	102,96	100,865
20	25	5000	0		114,16	117,27	115,406
21	50	5000	0		112,99	117	115,621
22	100	5000	0		117,05	121,19	118,93
23	150	5000	0		117,07	120,38	118,718
24	200	5000	1		80,42	94,57	91,103
TAG	Clients	Requests	Failed Request	Request AVG	REQUESTS MIN	REQUESTS MAX	REQUESTS AVG
25	10	10000	0		97,56	101,57	100,006
26	25	10000	0		113,19	117,31	114,5
27	50	10000	0		111,89	115,89	114,418
28	100	10000	0		117,67	119,3	118,504
29	150	10000	0		116,28	120,22	118,742
30	200	10000	44,6		86,18	92,31	88,981
TAG	Clients	Requests	Failed Request	Request AVG	REQUESTS MIN	REQUESTS MAX	REQUESTS AVG
31	10	15000	0		99,02	101,33	100,047
32	25	15000	0		112,12	116,06	114,556
33	50	15000	0		112,97	116,7	114,93
34	100	15000	0		116,56	120,08	118,204
35	150	15000	0		117,3	120,58	118,945
36	200	15000	77,4		81,39	90,02	84,916

Figura 85: Testes - Pedidos por segundo



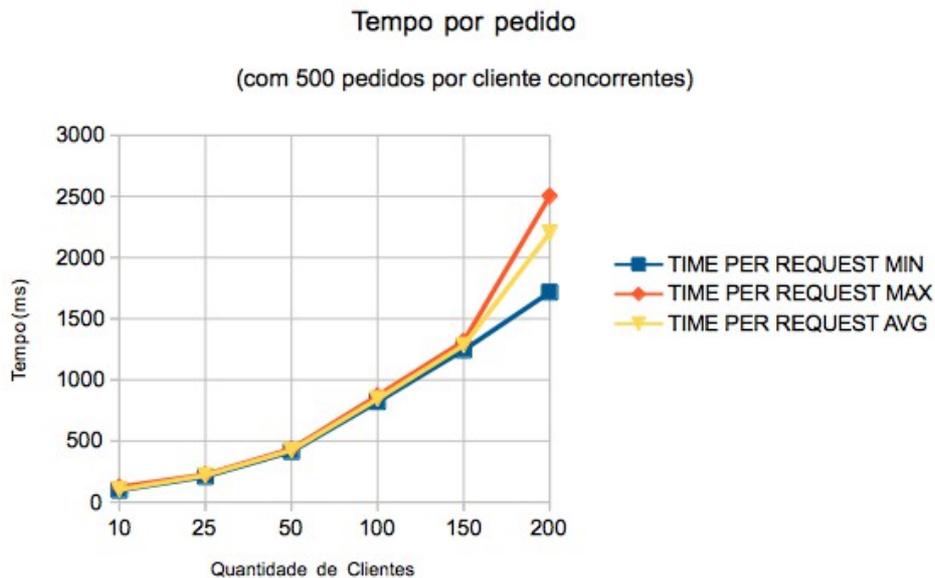




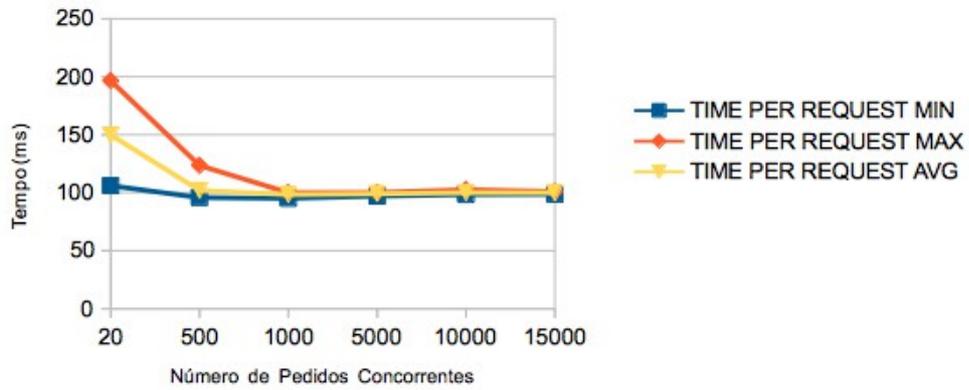
E.5.2 – Tempo por Pedido

TAG	Clients	Requests	TIME PER REQUEST MIN	TIME PER REQUEST MAX	TIME PER REQUEST AVG
1	10	20	106,027	196,566	149,8441
2	25	50	238,654	330,309	285,5707
3	50	100	438,258	649,878	525,2345
4	100	200	857,097	951,797	913,7276
5	150	300	1292,299	1392,426	1347,563
6	200	400	1726,016	1900,297	1790,7988
TAG	Clients	Requests	TIME PER REQUEST MIN	TIME PER REQUEST MAX	TIME PER REQUEST AVG
7	10	500	95,759	123,49	101,9016
8	25	500	207,8	225,156	217,6512
9	50	500	412,138	435,459	423,4766
10	100	500	820,139	870,667	842,9904
11	150	500	1242,894	1315,243	1282,1328
12	200	500	1716,349	2504,63	2202,4092
TAG	Clients	Requests	TIME PER REQUEST MIN	TIME PER REQUEST MAX	TIME PER REQUEST AVG
13	10	1000	95,062	100,057	98,4255
14	25	1000	207,976	217,564	214,7724
15	50	1000	406,065	425,725	417,2399
16	100	1000	820,744	862,105	836,8371
17	150	1000	1248,7	1311,547	1273,2379
18	200	1000	2174,696	2340,739	2255,7418
TAG	Clients	Requests	TIME PER REQUEST MIN	TIME PER REQUEST MAX	TIME PER REQUEST AVG
19	10	5000	97,123	100,177	99,1544
20	25	5000	213,185	218,983	216,6442
21	50	5000	427,356	442,5	432,5039
22	100	5000	825,177	854,355	840,9361
23	150	5000	1246,088	1335,913	1269,7618
24	200	5000	2114,921	2486,874	2199,6195
TAG	Clients	Requests	TIME PER REQUEST MIN	TIME PER REQUEST MAX	TIME PER REQUEST AVG
25	10	10000	98,459	102,503	100,008
26	25	10000	213,105	220,863	218,3576
27	50	10000	431,454	446,887	437,049
28	100	10000	838,239	849,809	843,864
29	150	10000	1247,675	1289,993	1263,3365
30	200	10000	2166,522	2320,593	2248,9118
TAG	Clients	Requests	TIME PER REQUEST MIN	TIME PER REQUEST MAX	TIME PER REQUEST AVG
31	10	15000	98,686	100,99	99,958
32	25	15000	215,398	222,965	218,2542
33	50	15000	428,459	442,595	435,1037
34	100	15000	832,786	887,536	850,4149
35	150	15000	1243,967	1278,766	1261,1767
36	200	15000	2221,796	2626,636	2386,1707

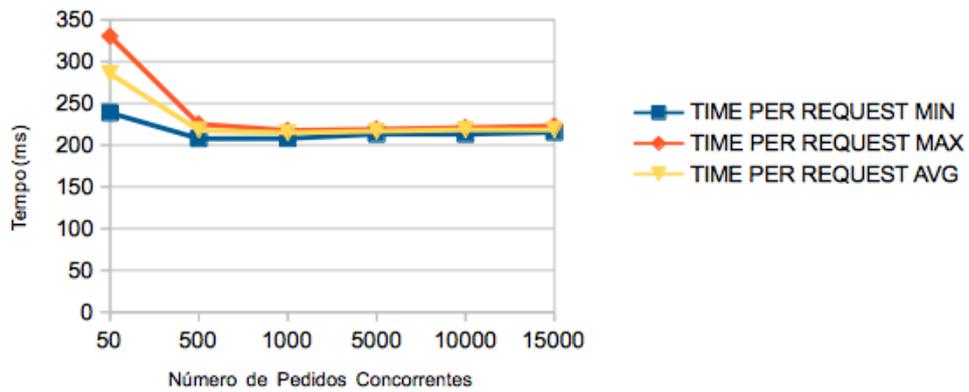
Figura 86: Testes - Tempo por pedido



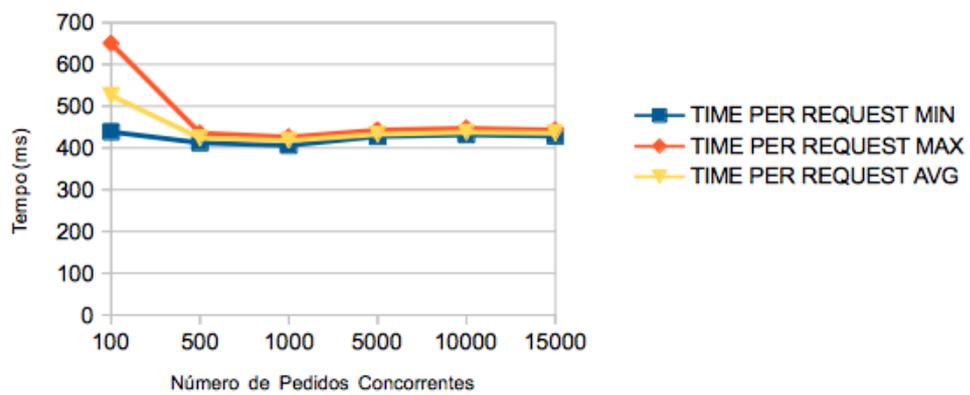
Tempo por pedido
(para 10 clientes)



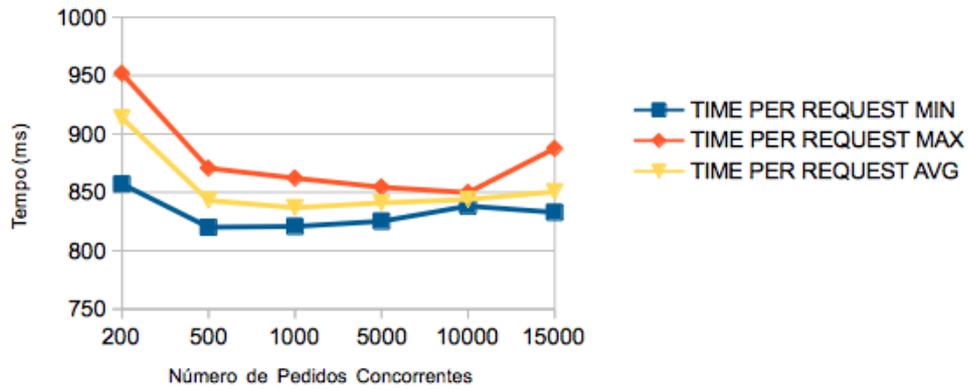
Tempo por pedido
(para 25 clientes)



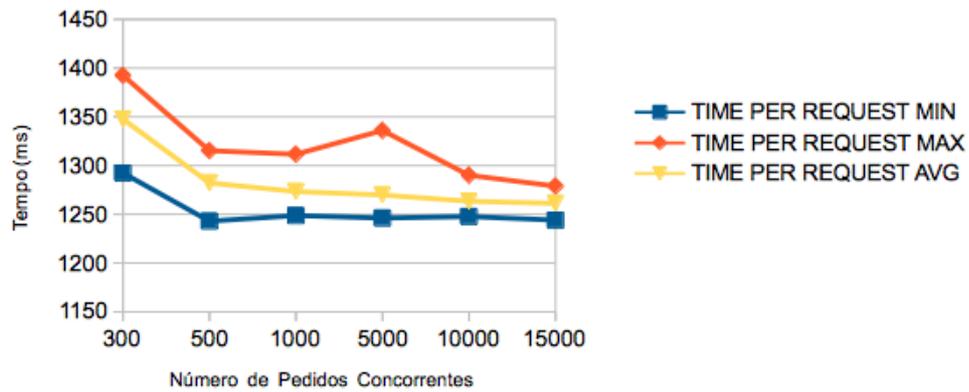
Tempo por pedido
(para 50 clientes)



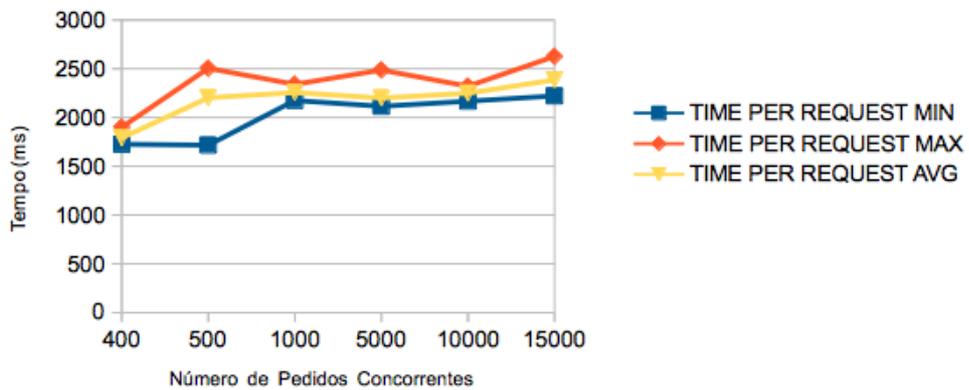
Tempo por pedido
(para 100 clientes)



Tempo por pedido
(para 150 clientes)

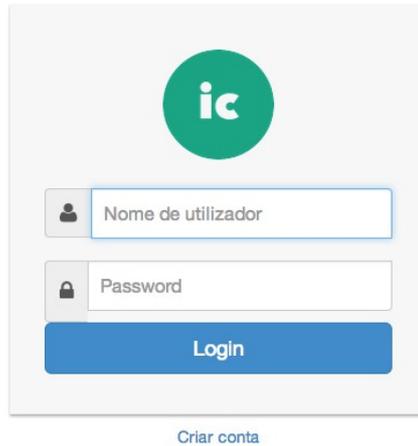


Tempo por pedido
(para 200 clientes)



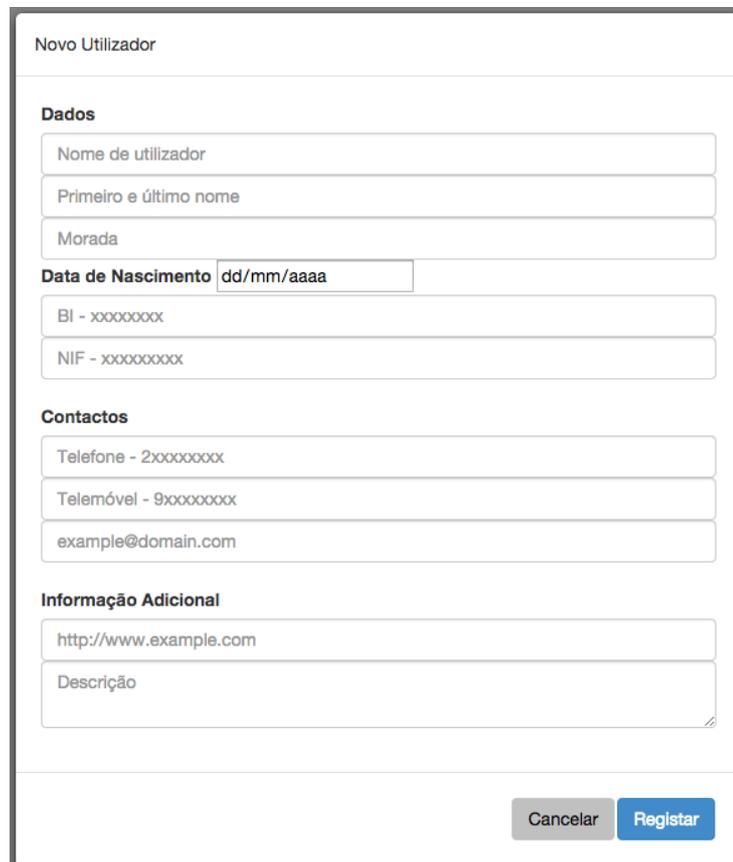
F – Screenshots da Plataforma

F.1 – Aplicações web



The login form features a green circular logo with the letters 'ic' at the top center. Below the logo are two input fields: the first is labeled 'Nome de utilizador' and the second is labeled 'Password'. A blue 'Login' button is positioned below the password field. At the bottom of the form, there is a blue link labeled 'Criar conta'.

Figura 87: Autenticação



The registration form is titled 'Novo Utilizador' and is organized into several sections. The 'Dados' section includes fields for 'Nome de utilizador', 'Primeiro e último nome', 'Morada', 'Data de Nascimento' (with a 'dd/mm/aaaa' placeholder), 'BI - xxxxxxxx', and 'NIF - xxxxxxxx'. The 'Contactos' section includes fields for 'Telefone - 2xxxxxxxx', 'Telemóvel - 9xxxxxxxx', and 'example@domain.com'. The 'Informação Adicional' section includes a field for 'http://www.example.com' and a larger text area for 'Descrição'. At the bottom right, there are two buttons: 'Cancelar' and 'Registar'.

Figura 88: Formulário de registo

Canais Utilizadores Relatórios Moderação 0

Bem-vindo, joabessa! Terminar Sessão

Canais

Procurar... Aceitar Pedidos Novo Canal

<p>Nome: Cafés de Coimbra</p> <p>Estado: Activo</p> <p>Descrição: Canal com os cafés de Coimbra</p> <p>Pertence a: mary</p> <p>Pedido efectuado em: 2014-08-28</p>	<p>Nome: Estádio Cidade Coimbra</p> <p>Estado: Manutencao</p> <p>Descrição: Estádio Cidade de Coimbra</p> <p>Pertence a: joabessa</p> <p>Pedido efectuado em: 2014-08-06</p>
<p>Nome: Jardins de Coimbra</p> <p>Estado: Activo</p> <p>Descrição: Jardins de Coimbra</p> <p>Pertence a: cramos</p> <p>Pedido efectuado em: 2014-08-25</p>	<p>Nome: Restaurantes</p> <p>Estado: Activo</p> <p>Descrição: Restaurantes de Maputo</p> <p>Pertence a: joabessa</p> <p>Pedido efectuado em: 2014-08-27</p>
<p>Nome: Restaurantes Coimbra</p> <p>Estado: Rejeitado</p> <p>Descrição: Restaurantes existentes na baixa de Coimbra</p> <p>Pertence a: mary</p> <p>Pedido efectuado em: 2014-08-28</p>	<p>Nome: Sétima Academia de Música</p> <p>Estado: Activo</p> <p>Descrição: Escola de Música de Coimbra - muito fixe!! :)</p> <p>Pertence a: Goncalo92</p> <p>Pedido efectuado em: 2014-08-16</p>

Figura 89: Administração - Listagem de todos os canais da plataforma

Canais Utilizadores Relatórios Moderação 0

Bem-vindo, joabessa! Terminar Sessão

Pedidos de Novos Canais

<p>Nome: Cafés de Coimbra</p> <p>Estado: Pendente</p> <p>Descrição: Canal com os cafés de Coimbra</p> <p>Pertence a: mary</p> <p>Pedido efectuado em: 2014-08-28</p>	<p>Nome: Restaurantes Coimbra</p> <p>Estado: Pendente</p> <p>Descrição: Restaurantes existentes na baixa de Coimbra</p> <p>Pertence a: mary</p> <p>Pedido efectuado em: 2014-08-28</p>
---	---

Figura 90: Administração - canais para aprovação

Canais Utilizadores Relatórios Moderação 0

Bem-vindo, joabessa! Terminar Sessão

Pedidos de Novos Canais

Não foram encontrados resultados!

Figura 91: Administração - sem canais para aprovação

Canais Utilizadores Relatórios Moderação 0 Bem-vindo, joabessa! Terminar Sessão

Utilizadores

Procurar... Novo Utilizador

<p>Nome: Gonçalo Filipe</p> <p>Estado: Activo</p> <p>Tipo: Cliente</p> <p>Nome de utilizador: Goncalo92</p>	<p>Nome: Sétima AM</p> <p>Estado: Activo</p> <p>Tipo: Gestor</p> <p>Nome de utilizador: SetimaAM</p>	<p>Nome: Carlos Ramos</p> <p>Estado: Activo</p> <p>Tipo: Cliente</p> <p>Nome de utilizador: cramos</p>
<p>Nome: Filipe Pires</p> <p>Estado: Activo</p> <p>Tipo: Gestor</p> <p>Nome de utilizador: fi</p>	<p>Nome: Goncalo Maria</p> <p>Estado: Activo</p> <p>Tipo: Gestor</p> <p>Nome de utilizador: goncalo</p>	<p>Nome: Maria Matos</p> <p>Estado: Activo</p> <p>Tipo: Cliente</p> <p>Nome de utilizador: mary</p>
<p>Nome: Nuno Fidalgo</p> <p>Estado: Activo</p> <p>Tipo: Administrador</p> <p>Nome de utilizador: nfidalgo</p>	<p>Nome: Samuel Lages</p> <p>Estado: Activo</p> <p>Tipo: Gestor</p> <p>Nome de utilizador: sam</p>	

Figura 92: Administração - Listagem de todos os utilizadores

Canais Utilizadores Relatórios Moderação 0 Bem-vindo, joabessa! Terminar Sessão

Dados do Utilizador

Estado: **Activo**

Nome de utilizador: mary

Nome: Maria Matos

Morada: Ladeira das Alpendurac

Data de Nascimento/Criação: 09/05/1984

BI: 67543598

NIF: 764312653

Contactos

Telefone: 230000000

Telemóvel: 930045422

Email: mary@teste.com

URL:

Descrição:

Gravar Alterações Reset Password Apagar Conta Assumir Conta

Figura 93: Administração - Editar perfil de um utilizador

Canais Utilizadores Relatórios Moderação 0 Bem-vindo, joabessa! Terminar Sessão

Gestão do Canal: Jardins de Coimbra

Dados do Canal

Estado: **Activo**

Nome do utilizador: cramos

Nome do canal: Jardins de Coimbra

Data do pedido: 25-08-2014

Descrição do canal: Jardins de Coimbra

Foto do canal:



Lista de Gestores



Utilizador: sam **Nome:** Samuel Lages
Email: jpbessa@gmail.com

Cessar Canal Apagar Canal Activar Canal Portal Público

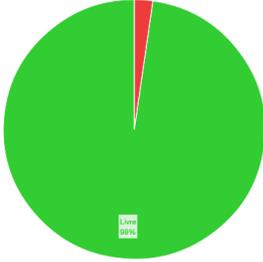
Figura 94: Administração - Visualizar um canal pertencente a um cliente

Canais Utilizadores Relatórios Moderação 0 Bem-vindo, joabessa! Terminar Sessão

Relatórios

EXPORTAR

Utilização do armazenamento dos canais



Quantidade de Canais

Quantidade de Itens

Figura 95: Administração - Relatório de sistema

Quantidade de Canais

11

Quantidade de Itens

10

Quantidade de Acessos à plataforma

24

Quantidade de Gestores

5

Figura 96: Administração - Relatório de sistema (continuação)



Figura 97: Lista de sem canais



Figura 98: Lista de um cliente sem gestores

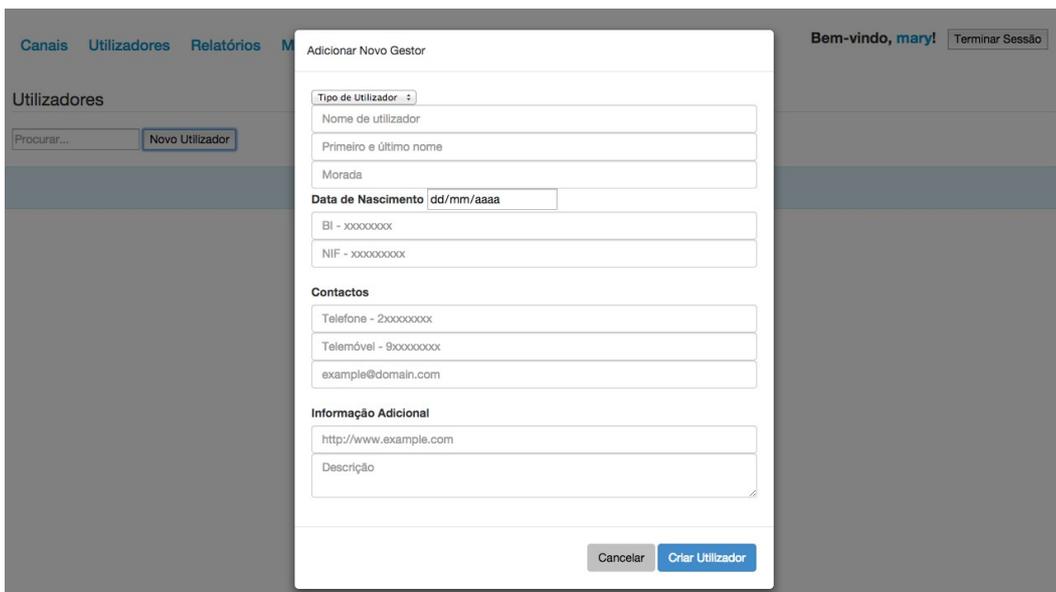


Figura 99: Formulário para adicionar um gestor



Figura 100: Pedido de criação de canal pendente

Gestão do Canal: Cafés de Coimbra

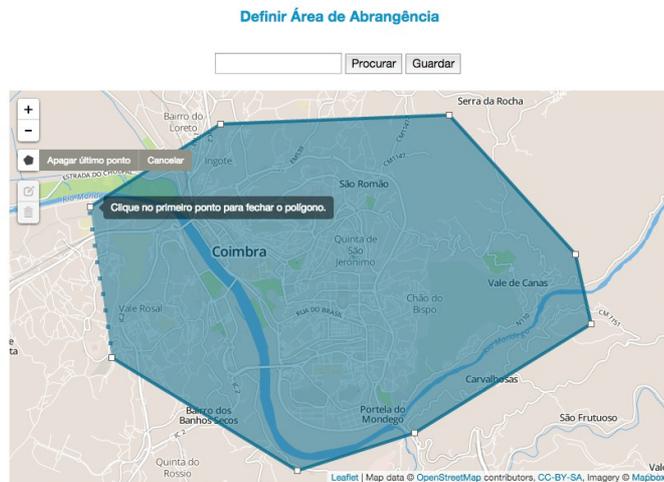


Figura 101: Definir área de abrangência de um canal

Gestão do Canal: Cafés de Coimbra



Figura 102: Acesso à gestão de um canal

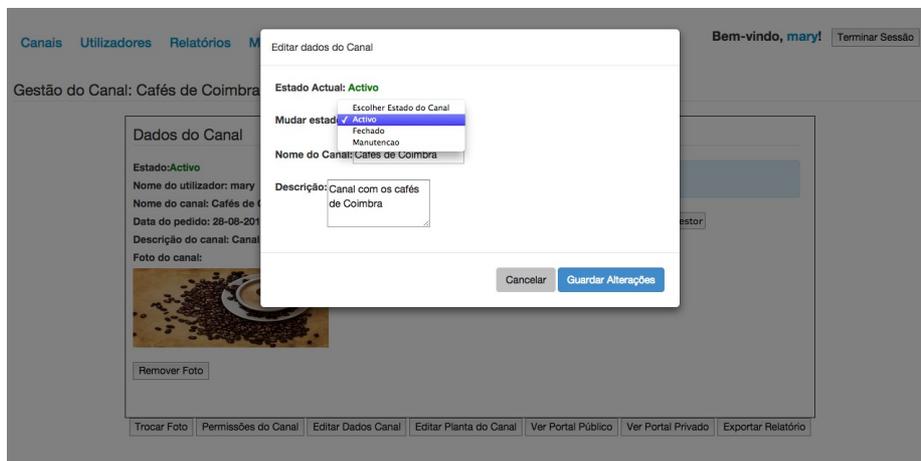


Figura 103: Edição dos dados de um canal

Gestão do Portal Público

Criar
Editar

Título:

Permissões do Portal

O utilizador pode comentar

O utilizador pode visualizar

Gravar Permissões Voltar ao Canal

Tipo de conteúdo multimédia:

Link

Video

Imagem

Texto

Adicionar Conteúdo

URL Portal

Ver Portal

Figura 104: Menu de criação e edição de conteúdos do portal público

Gestão do Portal Privado

Criar
Editar

Título:

Permissões do Portal

O utilizador pode comentar

O utilizador pode visualizar

Gravar Permissões Voltar ao Canal

Tipo de conteúdo multimédia:

Link

Video

Imagem

Texto

Adicionar Conteúdo

Códigos de acesso ao portal

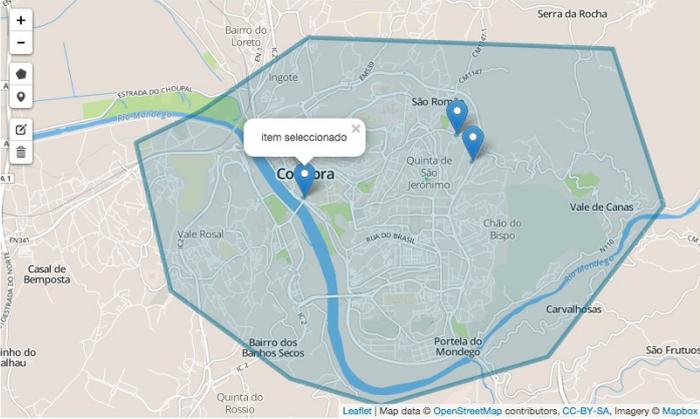


ZV6MAPPpf0

Renovar

Figura 105: Menu de criação e edição de conteúdos do portal privado

Gestão de Canais: Cafés de Coimbra



Remover todos Items

Propriedades do Item

Nome: Montanha

Editar

Conteúdos

+
procurar...

Café Montanha

Título: Café Montanha



Apagar Editar

Figura 106: Personalização da planta (regiões, itens e conteúdos dos itens)

Canais Utilizadores Relatórios **Moderação** 6 Bem-vindo, **cramos!** Terminar Sessão

Moderação de Comentários

Canal: Jardins de Coimbra

Portal Público

Comentário 1 Ver

Comentário 2 Ver

Comentário 3 Ver

A SER VISTO Comentário 4 Ver

Comentário 5 Ver

Comentário 6 Ver

Conteúdo Comentado:

Canal: Jardins de Coimbra
 Portal: PUBLICO
 Título Conteúdo: Parque Verde
 Conteúdo:



Comentário a aprovar:

Utilizador: Anónimo
 Comentário: este parque parece ser muito giro. fica onde mesmo?

Aceitar Rejeitar

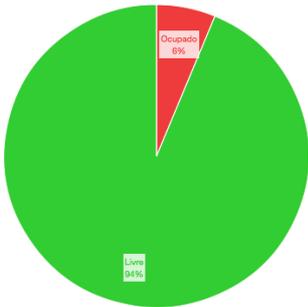
Figura 107: Moderação de comentários referentes a um canal

Canais Utilizadores Relatórios **Moderação** 6 Bem-vindo, **cramos!** Terminar Sessão

Relatórios

EXPORTAR

Utilização do armazenamento dos canais



Metric	Value
Quantidade de Canais	4
Quantidade de Items	3
Quantidade de Gestores	1

Figura 108: Relatório de estatísticas para clientes

F.2 – Aplicação móvel

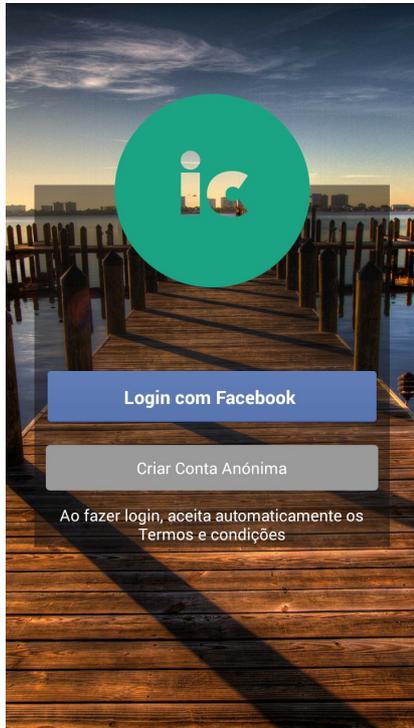


Figura 110: Autenticação



Figura 109: Login Facebook

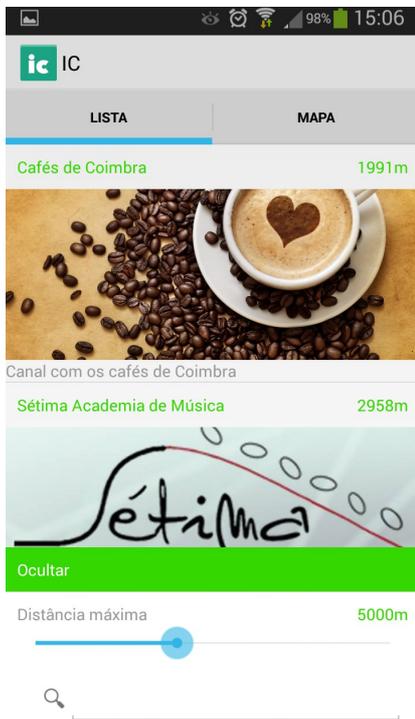


Figura 111: Lista canais

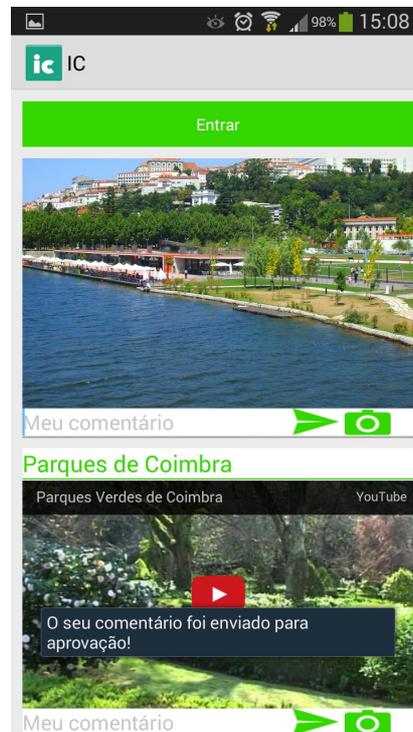


Figura 112: Portal Público



Figura 113: Portal Privado

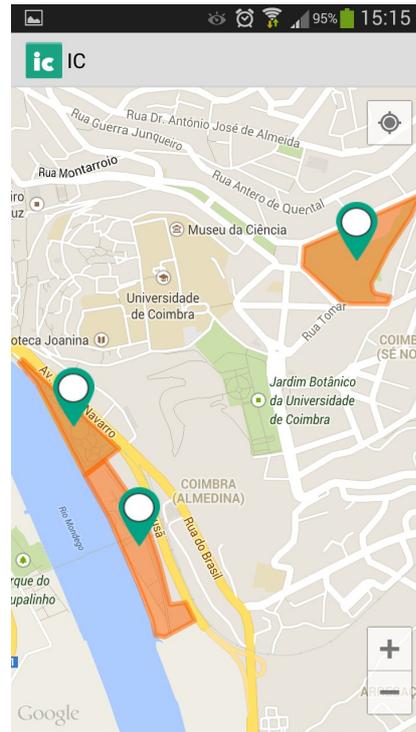


Figura 114: Planta

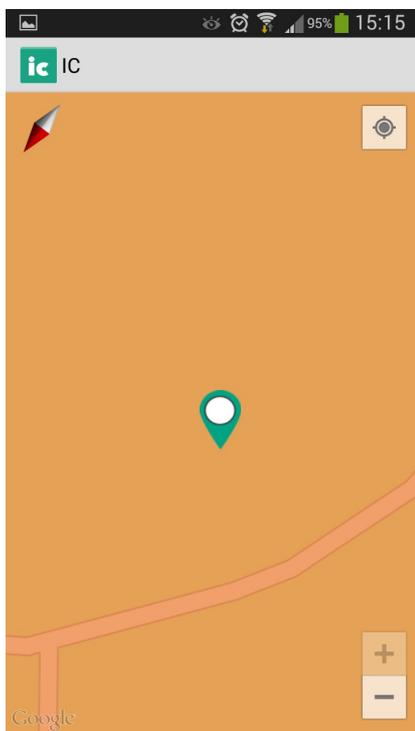


Figura 115: Navigation Mode

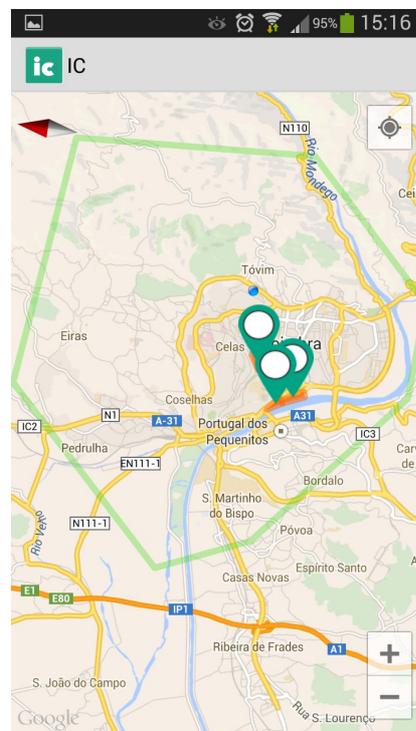


Figura 116: Planta

G – Scrum

G.1 – Product Backlog & Sprint Planning

Story	Sprint Init	Sprint End	Status
Tem de existir um modelo de dados que suporte a aplicação	1	1	Done
Deve ser possível os utilizadores L1 e L2 acederem à sua plataforma autenticando-se com username e password.	1	1	Done
Deve ser possível os utilizadores L1 terem acesso à administração de canais/clientes.	1	1	Done
Deve ser possível o sistema registar todos os acessos ou a criação de acessos ao sistema (logs).	1	1	Done
Deve ser possível os utilizadores L1 criação de acessos à plataforma.	1	1	Done
Deve ser possível ao super-administrador remover qualquer tipo de utilizador da plataforma.	1	1	Done
Deve ser possível ao super-administrador editar qualquer tipo de utilizador da plataforma.	1	1	Done
Deve ser possível a qualquer utilizador L1 poder ter múltiplos perfis associados à sua conta.	2	2	Done
Deve ser possível ao administrador criar utilizadores L2.	2	2	Done
Deve ser possível ao administrador cessar utilizadores L2.	2	2	Done
Deve ser possível ao administrador editar utilizadores L2.	2	2	Done
Deve ser possível ao administrador fazer reset às credenciais de acesso de utilizadores L2.	2	2	Done
Revisões base de dados/pgmodeler/rest/tokens/rbac.	2	3	Done
Deve ser possível ao administrador assumir a identidade de um utilizador L2.	2	5	Done
Deve ser possível os utilizadores L1 criarem canais associados a um cliente.	3	3	Done
Deve ser possível os utilizadores L1 cessarem um canal.	3	3	Done
Deve ser possível os utilizadores L1 removerem um canal.	3	3	Done
Deve ser possível um canal ter múltiplos gestores associados.	3	3	Done

Story	Sprint Init	Sprint End	Status
Deve ser possível um canal ter múltiplos gestores associados.	3	3	Done
Deve ser possível um canal ter um portal privado e/ou portal público.	3	3	Done
Deve ser possível os utilizadores L2 visualizarem todo o conteúdo associado ao seu canal.	3	4	Done
Deve ser possível ao cliente criar gestores.	4	4	Done
Deve ser possível ao cliente fazer reset às credenciais de acesso de gestores.	4	4	Done
Deve ser possível os utilizadores L2 editarem permissões.	4	4	Done
Deve ser possível os utilizadores L2 personalizarem os portais.	4	5	Done
Deve ser possível os utilizadores L2 personalizarem uma planta.	4	5	Done
Deve ser possível os utilizadores L2 definirem múltiplos itens dentro de cada canal.	4	5	Done
Deve ser possível ao cliente fazer pedido para a criação de um novo canal.	4	4	Done
Deve ser possível os utilizadores L2 personalizar um canal.	4	4	Done
Deve ser possível ao cliente criar elementos numa região.	5	5	Done
Deve ser possível os utilizadores L2 editarem conteúdo multimédia associado aos itens.	5	5	Done
Deve ser possível os utilizadores L2 moderarem os utilizadores do canal.	5	5	Done
Deve ser possível a visualização do portal público através de um URL.	5	5	Done
Deve ser possível autenticação do utilizador L3 na plataforma móvel.	6	6	Done
Deve ser possível ao utilizador L3 visualizar os canais que o rodeiam.	6	8	Done
Deve ser possível ao utilizador L3 visualizar a informação do portal público dos canais que o rodeiam.	6	6	Done
Deve ser possível ao utilizador L3 a visualização do portal privado pertencente ao canal.	6	6	Done
Deve ser possível ao utilizador L3 comentar conteúdos multimédia.	6	6	Done

Story	Sprint Init	Sprint End	Status
Deve ser possível ao utilizador L3 introduzir conteúdo multimédia.	6	6	Done
Deve ser possível ao utilizador L3 navegar com as diversas plantas.	7	7	Done
Deve ser possível ao utilizador L3 a visualização de itens definidos na região da planta.	7	7	Done
Deve ser possível à plataforma móvel apresentar uma listagem de todos os canais existentes.	7	7	Done
Deve ser possível ao utilizador L3 fazer pesquisa de canais.	7	7	Done
Deve ser possível à plataforma móvel apresentar informações de um canal.	7	7	Done
Deve ser possível os utilizadores L1 obterem estatísticas/relatórios detalhados.	8	8	Done
Deve ser possível os utilizadores L1 exportarem estatísticas/relatórios detalhados.	8	8	Done
Deve ser possível os utilizadores L2 obterem estatísticas/relatórios detalhados.	8	8	Done
Deve ser possível os utilizadores L2 exportarem estatísticas/relatórios detalhados.	8	8	Done
Deve ser possível os utilizadores L2 visualizarem gráficos das estatísticas.	8	8	Done
Deve ser possível à plataforma móvel recolher e enviar estatísticas.	8		ToDo

Tabela 35: Resumo dos sprint backlogs