



Mestrado em Informática e Sistemas

---

## **Plataforma de Reuniões Online**

Relatório apresentado para a obtenção do grau de  
Mestre em Engenharia Informática e Sistemas  
Especialização em Desenvolvimento de Software

**Autor**

**Telmo Dias Raimundo**

**Orientadora**

**Ana Rosa Pereira Borges**

Professora Coordenadora do Departamento de Informática e Sistemas  
Instituto Superior de Engenharia de Coimbra

**Supervisora**

**Marta Sofia Cabral e Cunha**

AIRC

**Coimbra, Outubro, 2017**



# Agradecimentos

A realização deste estágio de mestrado contou com importantes apoios e incentivos, sem os quais não se teria tornado uma realidade e aos quais estarei eternamente grato.

Em primeiro lugar, cumpre-me expressar o meu sincero agradecimento à AIRC, na pessoa do Sr. Jorge Coimbra, pela oportunidade que me concedeu para a realização deste estágio, pelo acolhimento, pelas condições prestadas e por toda a vontade e disponibilidade demonstrada.

Também à Eng<sup>a</sup>. Marta Cunha, ao Eng<sup>o</sup>. João Fidalgo, ao Eng<sup>o</sup>. Ivo Santos, ao Técnico Anselmo Craveiro e a todos aqueles que, de alguma forma, me ajudaram durante o decurso do estágio, agradeço por todo o envolvimento, acompanhamento, dedicação, conhecimentos técnicos e profissionais que me inculcaram e por todo tempo despendido.

Agradeço igualmente à minha orientadora de estágio no Departamento de Engenharia e Sistemas do ISEC, Prof<sup>a</sup>. Ana Rosa Borges, por toda a disponibilidade e apoio.

Expresso também o meu profundo agradecimento à minha família, em especial aos meus pais, por me terem inculcido os valores com que sempre pautaram a sua vida e pelo apoio incondicional dado ao longo de toda esta jornada.

Por último, e sem menos importância, agradeço a todos os meus amigos que estiveram e continuam a estar presentes, pelo apoio, paciência e carinho demonstrado ao longo de todo o meu percurso académico. Todos, à sua maneira, foram importantes.

Um especial agradecimento para os meus amigos Luís Vizeu, Helena Teixeira, António Sales e Andreia Nogueira, por todo o apoio demonstrado ao longo dos anos, pelo carinho e pela força entregue nas horas de maior aperto.



# Resumo

Nos dias de hoje nos órgãos deliberativos, como câmaras municipais e assembleias municipais, existe a necessidade de realizar reuniões onde são tomadas decisões. Essas decisões cumprem uma série de exigências legais.

Visto que a Associação de Informática da Região Centro (AIRC) procura satisfazer dos seus clientes, apoiando a inovação e os novos modelos de trabalho, utilizando soluções tecnologicamente avançadas e que permitam uma melhor gestão da administração pública, surge a necessidade de implementação de soluções de *software* que satisfaçam as exigências pretendidas

Este estágio teve como objetivo a criação de uma plataforma de reuniões *online*, onde envolveu o levantamento e análise de plataformas no mercado, a descrição das funcionalidades a desenvolver, e a sua implementação.

Durante o estágio foram analisadas algumas das plataformas existentes no mercado para realizar reuniões *online*, mas nenhuma das plataformas examinadas satisfazia as necessidades destas reuniões. A partir daí foi decidido desenvolver uma plataforma nova com a finalidade de proporcionar ao cliente uma melhor gestão das suas reuniões, bem como facilitar a discussão e votação dos diversos pontos da mesma. A análise de requisitos e o planeamento do projeto foram realizados em 4 fases: a 1ª fase reflete algumas funcionalidades base da reunião, dos pontos e da votação; a 2ª fase envolve as restantes funcionalidades do controlo da reunião; a 3ª fase envolve a sincronização dos dados; e a 4ª fase envolve tudo o que seja referente a ficheiros.

A plataforma foi estruturada seguindo o padrão *Model View Controller*, onde o modelo corresponde à base de dados, onde todos os dados são guardados. Esta base de dados já existia na empresa tendo apenas sido acrescentados 2 campos novos. A vista corresponde ao *frontend* que será a interface *web* onde o utilizador irá interagir. Este *frontend* foi construído em Angular 2. E o controlo corresponde ao *backend* que irá disponibilizar serviços para o *frontend*, tendo este sido construído em *java* utilizando o *Play! Framework*.

Este estágio permitiu realizar uma solução para gerir reuniões de órgãos deliberativos de modo a disponibilizar um conjunto de mecanismos para a visualização dos pontos a serem discutidos durante a reunião, e permitir o acesso à sua votação, bem como aceder aos documentos anexos aos pontos ou à documentação de outras reuniões. Este estágio permitiu ainda fortalecer e apreender conhecimentos em *Java*, *Play! Framework*, *Angular2*, *SQL*, *JavaScript*, *TypeScript*, *css*, *Json*, *sbt*, *HTTP Method* e *WebSockets*.

Palavras Chave: Reuniões Online, AIRC, *Model View Controller*, *PlayFramework*, *Angular2*.



# Abstract

In today's deliberative organizations, such as municipal councils and municipal assemblies, there is a need to hold meetings where decisions are made. These decisions fulfill a number of legal requirements.

Since the Informatics Association of the Central Region (AIRC) seeks to satisfy its customers, supporting innovation and new work models, using technologically advanced solutions and enabling a better management of public administration, there is a need to implement Software that meets the intended requirements.

The purpose of this internship was to create an online meeting platform, which involved the analysis and analysis of platforms in the market, a description of the functionalities to be developed, and their implementation.

During the internship, some of the existing platforms in the market for online meetings were analyzed, but none of the platforms examined met the needs of these meetings, and thus, it was decided to develop a new platform with the purpose of providing the client with a better management of its meetings, as well as facilitating the discussion and voting of the various meeting matters. The requirements analysis and project planning were carried out in 4 phases: the 1st phase reflects some basic functionalities of the meeting, the points and the voting; The second phase involves the remaining functionalities of the control of the meeting; The third phase involves the synchronization of the data; And the 4th phase involves everything related to files.

The platform was structured according to the model View Controller, where the model corresponds to the database, where all data is stored. This database already existed in the company having only been added two new fields. The view corresponds to the frontend which will be the web interface where the user will interact. This frontend was built in Angular 2. And the control corresponds to the backend that will provide services to the frontend, having this one been built in Java using Play! Framework.

This stage internship for a solution to manage meetings of deliberative organizations in order to provide a set of mechanisms for the visualization of the points to be discussed during the meeting, and to allow access to their voting, as well as to access the documents attached to the points or the Documentation of other meetings. This internship has also made it possible to strengthen and apprehend knowledge in Java, Play! Framework, Angular2, SQL, JavaScript, TypeScript, css, Json, sbt, HTTP Method, and WebSockets.

**Keywords:** Online Meetings, AIRC, Model View Controller, Play Framework, Angular 2.



# Índice

1	Introdução .....	1
1.1	Âmbito.....	1
1.2	Objetivos .....	1
1.3	Metodologia .....	2
1.4	Estrutura do Relatório .....	2
2	Contextualização do Estágio.....	5
2.1	Instituto Superior de Engenharia de Coimbra .....	5
2.1.1	Departamento de Engenharia Informática e de Sistemas .....	5
2.1.2	Mestrado em Informática e Sistemas.....	5
2.2	Associação de Informática da Região Centro .....	6
2.2.1	Apresentação.....	6
2.2.2	Historial.....	7
2.2.3	Produtos e Mercado .....	8
2.3	Plano de Trabalho.....	9
3	Plataformas de Reuniões Online.....	11
3.1	Enquadramento.....	11
3.2	Stakeholders e Utilizadores .....	12
3.3	Pressupostos .....	12
3.4	Enquadramento com Outras Aplicações .....	12
3.5	Estudo das plataformas existentes.....	13
3.5.1	TeamViewer.....	13
3.5.2	MeetingBurner .....	14
3.5.3	WebEx.....	14
3.5.4	Fuze.....	15
3.6	Análise das plataformas existentes.....	15
4	Análise da Plataforma.....	17



4.1	Visão.....	17
4.2	Âmbito.....	17
4.3	Planeamento .....	17
4.4	Metodologia de desenvolvimento .....	18
4.5	Análise de requisitos .....	19
4.5.1	Atores do sistema.....	19
4.5.2	User Stories.....	20
5	Desenvolvimento da Plataforma.....	25
5.1	Arquitetura .....	25
5.2	Base de Dados .....	26
5.3	Backend.....	27
5.4	Frontend .....	35
6	Conclusões .....	45
6.1	Trabalho realizado.....	45
6.2	Trabalho futuro.....	45
6.3	Dificuldades .....	46
6.4	Benefícios.....	46
	Referências.....	49
	Anexo A – Proposta de estágio.....	51
	Anexo B – Modelo de Base de Dados .....	55



# Lista de Figuras

Figura 1- Logotipo ISEC .....	5
Figura 2 – Logotipo AIRC.....	6
Figura 3 – Logotipo Team Viewer.....	13
Figura 4 – Logotipo MeetingBurner .....	14
Figura 5 – Logotipo Webex .....	14
Figura 6 – Logotipo Fuze.....	15
Figura 7 – Workflow de Desenvolvimento.....	19
Figura 8 – Arquitetura da Solução .....	25
Figura 9 – Base de Dados .....	26
Figura 10 – Pagina Login.....	36
Figura 11 – Pagina Inicial .....	37
Figura 12 – Menu Lateral .....	38
Figura 13 – Barra Superior .....	38
Figura 14 – Menu de Notificações.....	39
Figura 15 – Pagina de Consultas.....	39
Figura 16 – Pagina de Pesquisas.....	40
Figura 17 – Pagina de Votação .....	41
Figura 18 – Pagina Início/Fecho .....	42
Figura 19 – Pagina Controlo de Pontos .....	43



# Lista de Tabelas

Tabela 1 – Fases do planeamento .....	18
Tabela 2 – Ficheiro de Rotas .....	27
Tabela 3 – Classe Anexo.....	28
Tabela 4 – Classe Entidade.....	28
Tabela 5 – Classe Entidade Membro .....	29
Tabela 6 – Classe Entidade Ocasional .....	29
Tabela 7 – Classe Orgao .....	29
Tabela 8 – Classe Ponto.....	30
Tabela 9 – Classe Reuniao.....	30
Tabela 10 – Classe Seccao.....	31
Tabela 11 – Classe User.....	31
Tabela 12 – Classe Voto .....	32
Tabela 13 – Classe Service_Anexo.....	32
Tabela 14 – Classe Service_Entidade .....	32
Tabela 15 – Classe Service_EntidadeMembro .....	33
Tabela 16 – Classe Service_EntidadeOcasional .....	33
Tabela 17 – Classe Service_Orgao .....	33
Tabela 18 – Classe Service_Ponto.....	34
Tabela 19 – Classe Service_Reuniao .....	34
Tabela 20 – Classe Service_User.....	35
Tabela 21 – Classe Service_Votacao.....	35



# Definições e Acrónimos

**AIRC** – *Associação de Informática da Região Centro*

**CPA** – *Código de Processo Administrativo*

**CVS** – *Concurrent Versioning System*

**DEIS** – *Departamento de Engenharia Informática e Sistemas*

**HTML** – *HyperText Markup Language*

**HTTP** – *HyperText Transfer Protocol*

**IPC** – *Instituto Politécnico de Coimbra*

**ISEC** – *Instituto Superior de Informática de Coimbra;*

**JDBC** – *Java DataBase Connectivity*

**JMS** – *Java Message Service*

**JSON** – *JavaScript Object Notation*

**LDAP** – *Lightweight Directory Access Protocol*

**MIS** – *Mestrado em Informática e Sistemas;*

**MVC** – *Model-View-Controller*

**ORM** – *Object Relational Mapping*

**REST** – *REpresentational State Transfer*

**STA** – *Sistema de Tratamento de Atas;*

**TCP** – *Transmission Control Protocol*



# 1 Introdução

Desde 1982 que a Associação de Informática da Região Centro (AIRC) procura acompanhar a evolução natural das exigências dos seus clientes, apoiando a inovação e os novos modelos de trabalho, suportados por soluções tecnologicamente avançadas e que permitam uma clara otimização da gestão da administração pública.

Nesse sentido, há alguns anos foi criado o Sistema de Tratamento de Atas (STA), aplicação que permite elaborar as atas, minutas, ordens de trabalho e certidões das reuniões da câmara ou da assembleia municipal.

Esta solução facilita a partilha de informação acerca dos assuntos tratados nas reuniões dos órgãos deliberativos, contribuindo para o desmaterializar por completo de todos os documentos produzidos.

Porém foi sentida a necessidade de se criar um módulo novo, o qual disponibilize um conjunto de mecanismos para a visualização dos pontos a serem discutidos durante a reunião, permita o acesso à votação a cada membro que possua direito de voto, possibilitando a suspensão ou mesmo o encerramento dos pontos já debatidos, bem como a introdução do impedimento do voto para o ponto que se encontra em discussão. O acesso aos documentos anexos ao ponto a ser votado ou à documentação de outros registos de ata também deve ser disponibilizado.

## 1.1 Âmbito

Este relatório tem como objetivo descrever o trabalho desenvolvido na empresa AIRC, no âmbito da unidade curricular de projeto ou estágio industrial do 2º ano do Mestrado em Informática e Sistemas (MIS) no ramo de desenvolvimento de software do Departamento de Engenharia Informática e de Sistemas (DEIS) do Instituto Superior de Engenharia de Coimbra (ISEC).

Este estágio teve início a 23 de novembro de 2015, em regime de tempo integral, e a sua parte prática terminou a 31 de julho de 2016.

## 1.2 Objetivos

O estágio teve, como objetivos específicos, o desenvolvimento de uma plataforma de reuniões *online*, envolvendo o levantamento e análise detalhada de algumas das plataformas existentes no mercado, a definição das principais funcionalidades a desenvolver e respetivos requisitos, e a sua implementação.

O estágio teve ainda os objetivos genéricos de familiarizar com a tecnologia Angular2 e com tecnologia Play! Framework, bem como conhecer os processos internos da AIRC.

## 1.3 Metodologia

Durante o período de estágio, foi realizado um acompanhamento regular por parte dos orientadores, através de reuniões, com a finalidade de apresentar o trabalho realizado até esse momento, efetuar o planeamento das próximas tarefas, e ultrapassar problemas ou dificuldades que iam surgindo.

No desenvolvimento da plataforma foi utilizada uma metodologia ágil. Esta metodologia será apresentada numa secção mais à frente (Secção 4.4).

Para apoio ao processo de desenvolvimento de software, a empresa utiliza as seguintes ferramentas:

- **Concurrent Versioning System (CVS):** Sistema de partilha de ficheiros. É utilizado como repositório dos documentos relativos ao projeto (análise de requisitos, *templates* de documentos, *layouts*, etc.).
- **BitBucket:** Sistema de partilha de ficheiros que permite efetuar o controlo de versões. É utilizado como repositório do projeto. Esta ferramenta possibilita ainda uma visualização de forma gráfica através de uma página *web*.
- **JIRA:** Ferramenta utilizada para gestão de tarefas dos projetos, bem como parte da gestão do BitBucket.
- **Slack:** Ferramenta utilizada para comunicação interna, de modo a aumentar a produtividade da empresa.

## 1.4 Estrutura do Relatório

Este relatório está organizado em capítulos que descrevem as diversas tarefas realizadas, como foram realizadas, decisões tomadas e conclusões tiradas.

Após esta apresentação genérica do estágio, no segundo capítulo serão apresentadas as entidades envolvidas, assim como a proposta de estágio.

No terceiro capítulo, serão descritas algumas plataformas de reuniões *online* existentes no mercado, apresentando as suas funcionalidades e analisando a sua utilidade para este projeto.

No quarto capítulo, será exposta a primeira fase do desenvolvimento do *software*, onde será analisada a plataforma a desenvolver, a visão, o âmbito e o levantamento e análise de requisitos.

No quinto capítulo, será descrito o desenvolvimento da plataforma, apresentando a arquitetura da plataforma e as estratégias da sua implementação.

Por fim, no sexto capítulo, serão referidas as expectativas para a plataforma de reuniões *online* desenvolvida, o trabalho futuro a realizar, as dificuldades encontradas durante o estágio e os benefícios obtidos com a realização do mesmo.



## 2 Contextualização do Estágio

### 2.1 Instituto Superior de Engenharia de Coimbra

O ISEC [1] é uma unidade orgânica de ensino do Instituto Politécnico de Coimbra (IPC), cujo logotipo é representado na Figura 1, e constitui um centro de criação, transmissão e difusão de cultura, ciência e tecnologia, cabendo-lhe ministrar a preparação para o exercício de atividades profissionais no domínio da engenharia, e



Figura 1- Logotipo ISEC

promover o desenvolvimento da região em que se insere. O ISEC abrange várias áreas do domínio tecnológico, oferecendo aos alunos cursos modernos, práticos e com saídas profissionais [1].

A qualidade dos cursos lecionados colocam o ISEC a par da realidade tecnológica europeia e internacional, permitindo aos alunos que frequentem este instituto de engenharia de renome nacional, alargar ainda mais os horizontes profissionais.

O esforço contínuo de obtenção de parcerias constitui uma mais-valia, pois aproxima os alunos do ISEC da realidade empresarial e preparam-nos para o futuro, focando o essencial e procurando a excelência.

#### 2.1.1 Departamento de Engenharia Informática e de Sistemas

O DEIS dedica-se à formação, investigação, desenvolvimento e prestação de serviços nas áreas da Engenharia Informática e das Tecnologias de Informação [2].

O DEIS tem seguido uma política de aproximação ao sector empresarial através de estágios curriculares realizados em muitas empresas, maioritariamente da região centro, e por protocolos de colaboração com organismos públicos e empresas privadas, de entre os quais se destaca a AIRC.

#### 2.1.2 Mestrado em Informática e Sistemas

O MIS, lecionado pelo DEIS-ISEC, tem por objetivo formar mestres em informática e sistemas, capazes de exercerem a sua atividade profissional com um elevado nível de competência técnica, científica e profissional em cada uma das duas áreas de especialização (ou ramos) propostas [3]:

- Desenvolvimento de Software;
- Tecnologias da Informação e do Conhecimento.

Estas especializações possuem um carácter assumidamente profissionalizante, pretendendo-se formar especialistas com competências em domínios de aplicação muito concretos. A escolha do elenco apresentado resulta da experiência acumulada nos últimos anos na interação estreita que o DEIS do ISEC manteve com o tecido empresarial, não só regional, mas também nacional e europeu.

O estágio final (unidade curricular "estágio ou projeto industrial") decorre na maior parte do segundo ano letivo do mestrado, sendo desenvolvido no ambiente das empresas acolhedoras e acompanhado pelos docentes do mestrado.

## 2.2 Associação de Informática da Região Centro

### 2.2.1 Apresentação

O estágio foi realizado na Associação de Informática da Região Centro, também conhecida por AIRC, e que tem o logotipo representado na Figura 2 [4].

A AIRC tem, como missão, a prestação de serviços e o fornecimento de soluções inovadoras de qualidade superior e com excelente relação custo benefício, que os diferenciem face à concorrência e que satisfaçam os requisitos e superem as expectativas dos seus clientes.



Figura 2 – Logotipo AIRC

A empresa tem como visão ser encarada pelos seus clientes como parceira estratégica, com o objetivo comum de criar valor para ambas as partes, e de ser reconhecida como principal referência em termos de soluções informáticas para a administração pública local e como líder incontestada deste setor de mercado.

Os valores são o suporte moral e ético da empresa, e definem as regras básicas que regulam os comportamentos e atitudes de todos os colaboradores, cuja observância é pressuposto fundamental para executar a sua missão e alcançar a sua visão.

São eles a qualidade, inovação, satisfação do cliente, integridade, espírito de equipa e relações humanas no trabalho, respeito, sucesso e responsabilidade individual.

A direção da AIRC é constituída por um presidente e quatro vogais, e por uma assembleia intermunicipal que é constituída por um presidente, um vice-presidente e por um secretário. Ao nível operacional dispõe de três diretores: o diretor geral, o diretor de marketing e comercial, e o diretor de desenvolvimento.

### 2.2.2 Historial

A AIRC foi fundada em 1982, por 30 municípios da região centro, de onde advém o seu nome. Até aos dias de hoje, à exceção de um dos municípios que desistiram numa fase inicial, não houve alterações nos seus membros, sendo eles os seguintes:

- Município de Aguiar da Beira
- Município de Almeida
- Município de Arganil
- Município de Aveiro
- Município da Batalha
- Município de Cantanhede
- Município de Castro Daire
- Município de Celorico da Beira
- Município de Coimbra
- Município de Condeixa-a-Nova
- Município da Figueira da Foz
- Município da Guarda
- Município da Lousã
- Município de Mangualde
- Município de Meda
- Município de Mira
- Município de Miranda do Corvo
- Município de Mortágua
- Município de Oliveira de Frades
- Município de Pampilhosa da Serra
- Município de Penacova
- Município de Penalva do Castelo
- Município de Pinhel
- Município de Porto de Mós
- Município do Sabugal
- Município de Trancoso
- Município de Vila Nova de Paiva
- Município de Vila Nova de Poiares
- Município de Viseu.

Inicialmente, a AIRC instalou a sua sede no Estádio Municipal de Coimbra, nas antigas instalações do Centro Ordenador Municipal da Câmara Municipal de Coimbra e dos Serviços Municipalizados de Coimbra. Foram ainda criados Centros de Processamentos de Dados em Aveiro, Viseu, Figueira da Foz e Guarda. Nessa altura, o pessoal do Centro Ordenador Municipal

da Câmara Municipal de Coimbra, que pertencia ao quadro de pessoal do referido município, transitou para o quadro de pessoal da AIRC. Simultaneamente, foram contratados recursos humanos para reforçar os efetivos da sede e dos Centros de Processamento de Dados.

Nesta altura, a AIRC operava praticamente em exclusivo para os seus associados, tendo ainda 10 clientes não associados, processando faturação de água, contabilidade e vencimentos.

Em 1988, devido a insustentabilidade deste modelo, foram realizadas transformações profundas na organização, assumindo uma quase total rutura com o passado. Foram extintos os Centros de Processamento de Dados, mantendo apenas a sede em Coimbra, o que teve como consequência a redução dos efetivos.

Nesta altura, foi alterado significativamente o objetivo da AIRC, passando a assumir-se como uma empresa produtora de software e fornecedora de serviços, tendo, nos anos que se seguiram, desenvolvido novos módulos de software vocacionados para implementação de cada cliente.

Em 2002 a sede foi transferida para um novo edifício na Av. Fernão de Magalhães, passando de um espaço com 300m<sup>2</sup> para um cinco vezes maior.

A AIRC continuou a crescer ao longo dos anos e a consolidar a sua posição no mercado, mudando novamente de instalações em 2012 para a Zona Industrial de Taveiro. Nessa altura começou a construção da sua nova sede no Coimbra iParque, para onde se mudou em abril de 2016.

No presente, a AIRC possui uma estrutura com quase 100 profissionais e detém uma posição confortável de liderança neste sector de mercado, com cerca de 250 clientes pertencentes à administração pública, entre os quais 60% dos Municípios e algumas outras entidades, tais como Serviços Municipalizados, Comunidades Intermunicipais, Associações de Municípios, Empresas Municipais, Regiões de Turismo e Juntas de Freguesia.

### 2.2.3 Produtos e Mercado

Desde 1982 que a AIRC procura acompanhar a evolução natural das exigências dos seus clientes, apoiando a inovação e os novos modelos de trabalho, suportados por soluções tecnologicamente avançadas e que permitam uma clara otimização da gestão da administração pública. Nesse sentido, disponibiliza uma ampla oferta de soluções para os organismos da Administração Pública Local, que permite colocar a inovação ao serviço da capacidade e eficácia de resposta.

Aliando um serviço de suporte devidamente capacitado e especializado na quase totalidade das áreas e atribuições da administração pública, e abrangendo toda a cadeia de valor, as soluções

AIRC permitem a cada organismo adotar as melhores práticas de gestão e administração, assentes em mecanismos organizacionais altamente produtivos e dinâmicos.

Esta completa oferta permitiu consolidar a liderança de mercado da Administração Pública Local, estando presente com as suas soluções na maioria dos Municípios do País.

Na AIRC sabe-se que cada organização tem uma realidade muito própria, e com base nessa ideia avaliam-se os clientes, desenvolvendo soluções para as mais diversas áreas de atividade.

Gestão Financeira, Juntas de Freguesia, *Business Intelligence*, Portal Colaborativo da Organização, Balcão Único de Atendimento, Desmaterialização de Documentos e Processos, Gestão Comercial de Aguas, Gestão e Planeamento do Território e Gestão do Capital Humano são as soluções que a AIRC possui para dar resposta às necessidades dos seus clientes. Cada uma destas soluções possui um número diversificado de aplicações.

## 2.3 Plano de Trabalho

A proposta inicial do estágio (Anexo A) tinha como meta o desenvolvimento de uma plataforma *online* que fornecesse as seguintes funcionalidades:

- Estado da reunião, que permite identificar o início da reunião, a indicação do estado dos diversos pontos a debater (em preparação ou em votação), bem como o encerramento da reunião.
- Estado em que se encontram os diversos pontos da ordem de trabalhos, nomeadamente os pontos por votar, abertos a votação, pontos adiados ou suspensos e pontos encerrados.
- Acesso à reunião, que permite a visualização das reuniões em aberto para discussão, disponível a membros da reunião com e sem acesso ao direito de voto.
- Consulta de documentos-base para a reunião, permitindo a visualização da ata da reunião anterior, disponível a participantes com direito de voto na reunião anterior (inclui ainda a assinatura da respetiva reunião através do cartão de cidadão).
- Visualização da ordem de trabalhos da reunião para participantes com ou sem direito de voto, e respetivo acesso aos documentos anexos aos pontos disponibilizados.
- Consulta por texto livre em todas as atas aprovadas, disponível em tempo real para os participantes da reunião.
- Métodos de votação e gravação de intervenções por utilizador, com a possibilidade de transcrição de dados de voz, para membros com direito de voto.
- Seleção dos pontos a serem votados e respetiva votação, inserindo no sistema o resultado do voto que pretende declarar, com o objetivo de decretar a tomada de decisão, e selecionar o resultado dessa mesma decisão (a favor, abstenção ou contra).

- Inserção de declarações de voto, com a inserção no sistema da informação relacionada com o ponto a ser votado, com o objetivo de publicar a declaração de voto.
- Troca de mensagens entre participantes com direito a voto, possibilitando inserir no sistema informação, usando serviços *Java Message Service (JMS)* com o objetivo de trocar mensagens entre membros da reunião.
- Possibilidade de inserção de pontos extraordinários, passando os mesmos a fazer parte dos pontos extra da ordem de trabalhos, com o objetivo de colocar em discussão e votação. Pretendendo a discussão de aceitação do ponto extraordinário, ou seja, a aceitação por parte do *backoffice* e futura validação do ponto.

Para este estágio foi delineado um planeamento, como a seguir se apresenta:

- Análise do âmbito – Identificação das funcionalidades a implementar.
- Planeamento – Planeamento da arquitetura de *software*, escolha e estudo de ferramentas e tecnologias a utilizar, e negociação, com empresa externa, sobre o *layout* da aplicação.
- Desenvolvimento – Criação da aplicação.
- Testes e Validação – Realização de testes, enquadramento e aprovação do cliente.
- Deployment – Colocação em produção no cliente.

Este planeamento inicial não foi completamente cumprido devido a fatores externos ao estágio. Dois dos principais fatores prenderam-se com os constantes atrasos e enganos da empresa externa responsável pela realização dos *layout's*, e com a reestruturação de métodos de desenvolvimento e de tipo de arquitetura de *software* que se encontrava a decorrer na AIRC, onde se optou por esperar e começar já com esses novos métodos e ferramentas.

## 3 Plataformas de Reuniões Online

### 3.1 Enquadramento

Nos órgãos deliberativos, como câmaras municipais e assembleias municipais, existe a necessidade de realizar reuniões onde são tomadas decisões, as quais cumprem uma série de exigências legais.

Uma reunião possui sempre uma pessoa que controla a reunião, que normalmente é o secretário. Porém, mais elementos podem ter essa permissão, pois, por exemplo, na ausência do secretário a reunião não pode ficar bloqueada. Normalmente o presidente é o único elemento que possui essa permissão, mas essa decisão depende do órgão em questão.

Uma reunião tem sempre uma ordem de trabalhos, ou seja, um documento que contém uma lista de pontos e assuntos que vão ser discutidos na mesma. Antes das reuniões começarem, é necessário criar o documento da respetiva ordem de trabalhos. Esse documento tem de ser enviado aos elementos da reunião com 48 horas de antecedência, ao abrigo do disposto no artigo 25º do código de procedimento administrativo (CPA), aprovado pelo Decreto-Lei n.º 4/2015 [5]. O documento referido não é realizado nesta solução, pois a aplicação STA já realiza esse requisito, bem como já dispõe da informação sobre quem são os membros e quais têm direito de voto.

No dia da reunião, o secretário tem diversas funções. Uma delas é a marcação de presenças e faltas de elementos do órgão, e o registo da presença de entidades ocasionais, entidades externas ao órgão, como por exemplo um presidente de uma junta de freguesia. Está ainda incumbido de fazer a gestão dos pontos da reunião, podendo acrescentar, editar ou remover pontos extra à ordem de trabalhos, e fazer a gestão do ponto que está a ser discutido, podendo este ponto estar em discussão, em votação ou encerrado. Os pontos podem ainda ser adiados para uma próxima reunião ou ser suspensos, sendo o ponto aberto de novo ainda no decorrer dessa reunião. O secretário termina a reunião quando esta tiver todos os pontos encerrados ou adiados.

O secretário pode, durante a reunião, editar um esboço da minuta que foi criada previamente, ou criar uma nova minuta a partir de um modelo preexistente. Este, por motivo de ausência, pode passar o controlo da reunião a outro membro com permissões para tal, ficando esse novo elemento a controlar a reunião.

É possível também bloquear o voto de um elemento que se encontre ausente ou que tenha um impedimento para realizar esse voto, como por exemplo na situação de o ponto da ordem de

trabalhos estar relacionado com uma empresa na qual um vereador seja dono, o que levará ao bloqueio do seu voto.

Pode ainda colocar um ponto com votação secreta, passando a saber-se apenas o resultado no final da votação.

Os restantes elementos do órgão com direito de voto podem ver a ordem de trabalhos e toda a informação dos pontos, para que posteriormente possam votar nesses pontos, saindo dessa votação uma deliberação que constará no documento da ata e/ou da minuta.

Por último, existe também a necessidade de realizar pesquisas no conteúdo de documentos para encontrar um determinado assunto, ou consultar os documentos de uma determinada reunião.

## 3.2 Stakeholders e Utilizadores

Após um estudo das regras de negócio, encontramos os seguintes grupos de *stakeholders*:

- Membros do órgão deliberativo;
- Outras aplicações da AIRC que irão utilizar os dados coletados por esta aplicação;
- Equipa de desenvolvimento;
- Equipa de *design*.

Dentro dos membros do órgão deliberativo, encontramos dois perfis de utilizadores: o secretário e os restantes elementos do órgão. O secretário é a pessoa responsável por fazer a gestão da reunião, dos seus pontos e qual está em votação, incumbindo-lhe ainda realizar a minuta e/ou a ata da reunião. Os restantes elementos têm a possibilidade de participar na reunião, visualizando toda a informação.

## 3.3 Pressupostos

Os principais pressupostos para este projeto foram os seguintes:

- A aplicação terá de correr em *desktop's* e *tablet's*;
- Será utilizada a base de dados já existente na AIRC;
- A base de dados utilizada é feita em tecnologia Informix.

## 3.4 Enquadramento com Outras Aplicações

Esta solução irá interagir com outras aplicações já desenvolvidas pela AIRC. A principal aplicação será o STA, pois é nesta aplicação que se irá criar, agendar e preparar a reunião. Esta interação não é realizada diretamente, pois não existirá nenhuma ligação entre as aplicações [6].

Esta interação será realizada através dos dados da base de dados que será única para as duas aplicações.

## 3.5 Estudo das plataformas existentes

Nesta secção são analisadas algumas das plataformas existentes no mercado para realizar reuniões *online*. Esta análise foi feita com base em documentação existente.

Após uma pesquisa de plataformas de reuniões *online* que fosse mais de encontro com as funcionalidades pretendidas, analisaram-se as seguintes:

- TeamViewer;
- MeetingBurner;
- WebEx;
- Fuze.

### 3.5.1 TeamViewer

O TeamViewer [7] (com o logotipo da Figura 3) é um *software*, criado em 2005 para suporte remoto, acesso remoto e colaboração *online*. Atualmente é a principal aplicação utilizada para realizar suporte remoto a nível mundial.

O módulo de reuniões *online* permite agendar reuniões e conduzi-las, bem como outras que não tenham sido agendadas. Esta aplicação possibilita também partilhar a tela do dispositivo com os restantes membros da reunião, realizar videochamadas com todos os participantes da reunião, e o envio de mensagens de texto entre os participantes.



Figura 3 – Logotipo Team Viewer

Este módulo permite também desenhar como se estivesse a escrever num quadro em branco, para assim facilitar uma explicação de um determinado assunto. Este módulo possibilita ainda a partilha de arquivos com os participantes, e tem a possibilidade de gravar o som, a voz sobre ip, o vídeo e as imagens de *webcam* dos participantes das reuniões.

Esta aplicação proporciona a criação de grupos de participantes, para facilitar a organização dos participantes.

### 3.5.2 MeetingBurner

O MeetingBurner [8] (com o logotipo da Figura 4), criada em 2011, é uma plataforma relativamente recente na indústria das reuniões *online*, que permite discutir assuntos com pessoas distantes fisicamente, sem ser necessária a instalação de *software*. Esta possibilita agendar a reunião e gerar avisos automáticos via *email* e SMS para os participantes da reunião.

Proporciona, também, a transmissão em direto da *webcam* do orientador e da sua tela, apesar de, para esta última ser necessária a instalação de *software* adicional. Possui um *chat* para comunicação dos utilizadores da reunião que realiza estatísticas sobre a interação dos utilizadores no mesmo.

Esta plataforma permite igualmente a gravação de ficheiros de áudio e a sua partilha, logo após o término da reunião. Possui integração com Skype, Facebook e Weber para campanhas de *marketing*.

Por último, esta possibilita a inscrição de participantes para as reuniões através de formulários de registo personalizados, e possui interação com o *Paypal*.



Figura 4 – Logotipo MeetingBurner

### 3.5.3 WebEx

A WebEX [9], criada pela CISCO e com o logotipo da Figura 5, é uma plataforma que permite, à semelhança das anteriores, discutir assuntos com pessoas distantes fisicamente. Possibilita realizar videoconferência em alta definição, até 7 participantes em simultâneo. Proporciona, também, o envio de emails no final da reunião com a informação da reunião e ficheiros anexos, bem como ligar aos participantes, para os avisar da reunião.



Figura 5 – Logotipo Webex

Esta plataforma permite personalizar o *layout*, de maneira a que cada utilizador adapte à sua realidade. Possibilita ainda conversas em particular ou em grupo.

Esta plataforma também oferece a partilha de vários arquivos ao mesmo tempo pelos participantes, bem como a tela do orador.

Esta plataforma funciona também em dispositivos móveis nomeadamente em IOS, Android, Windows e BlackBerry e disponibiliza 1GB de espaço para armazenamento de ficheiros.

### 3.5.4 Fuze

A Fuze [10], empresa norte americana fundada em 2006, criou uma plataforma com o mesmo nome e com o logotipo da Figura 6, que combina chamadas telefônicas, videoconferência, mensagens de texto e reuniões online. Com esta plataforma é possível partilhar ficheiros e a tela do computador do orador, tudo de forma simples com apenas alguns cliques. Todos os documentos utilizados são guardados automaticamente na conta do utilizador, de modo a poder ser descarregado posteriormente. É possível ainda definir uma data limite até onde o utilizador poderá ver os arquivos



Figura 6 – Logotipo Fuze

Esta permite realizar videoconferência em alta definição até 12 participantes em simultâneo, sendo possível trocar a meio da reunião os membros que estão a transmitir o vídeo. É ainda possível gravar a reunião.

Esta plataforma não permite a troca de orador a meio da reunião, nem o envio de emails após a reunião, nem permite definir lembretes para a reunião.

## 3.6 Análise das plataformas existentes

Através de uma análise do funcionamento das plataformas, conclui-se que a maioria destas centra as suas funcionalidades na partilha de vídeo e áudio. Estas plataformas possuem outras diversas funcionalidades, como partilha de ecrã ou quadro interativo.

Todas estas plataformas partem do pressuposto que os utilizadores estão distantes fisicamente, e de uma forma geral resolvem esse problema.

Contudo, esse problema não é apresentado neste estágio, pois por obrigações legais todos os membros das reuniões têm de estar presentes fisicamente no mesmo espaço. Além disso, todas estas plataformas permitem a discussão dos assuntos, mas nenhuma permite a sua votação.

Outro aspeto a ter em conta é a necessidade da aplicação interagir com outras aplicações já existentes na empresa.

Tendo em conta a análise efetuada, e a existência de determinados aspetos muito específicos neste tipo de reuniões de órgãos deliberativos, foi decidido implementar uma solução completamente nova e com características distintas.



## 4 Análise da Plataforma

### 4.1 Visão

A plataforma de reuniões *online* a desenvolver é baseada em tecnologias *web*, e tem como finalidade proporcionar ao cliente uma melhor gestão das suas reuniões, bem como facilitar a discussão e votação dos diversos pontos da mesma. A plataforma irá ajudar os diversos utilizadores em todo este processo, através de uma interface simples e intuitiva.

### 4.2 Âmbito

A plataforma tem como principais funcionalidades:

- **Controlo da reunião:** o secretário da reunião é capaz de iniciar, encerrar, cancelar, pausar e retomar a reunião, marcar as presenças dos membros da reunião, bem como das entidades externas. Também é capaz de colocar um ponto em discussão, em votação, encerrar, adiar ou suspender. O secretário é ainda capaz de bloquear o participante num determinado ponto e justificar o seu bloqueio.
- **Discussão e votação de pontos:** os membros da reunião são capazes de ver toda a informação da reunião, os pontos que serão debatidos nessa reunião, bem como os anexos desses pontos. São também capazes de votar num determinado ponto que esteja em votação, bem como ceder o seu voto ao controlador da reunião.
- **Pesquisa e consulta de reuniões:** todos os utilizadores são capazes de pesquisar reuniões, e ver os documentos associados a essa reunião (ordem de trabalhos, ata e minuta), bem como pesquisar um determinado tema nesses documentos.

### 4.3 Planeamento

Após o estudo e levantamento das funcionalidades de alto nível, chegou-se à conclusão que o desenvolvimento da plataforma teria de ser organizado por fases. A principal razão desta decisão prendeu-se com a elevada complexidade de algumas funcionalidades que se pretendiam implementar.

Foram encontradas quatro fases, como mostra a Tabela 1, as quais são descritas de seguida:

A primeira fase reflete o controlo base da reunião, dos pontos e da votação por parte dos membros; a segunda fase envolve as restantes funcionalidades do controlo da reunião; a terceira

fase envolve a sincronização dos dados em tempo real bem como as notificações; a quarta fase reflete tudo o que envolva a manipulação de ficheiros.

Tabela 1 – Fases do planeamento

<i>Fases</i>	<i>Funcionalidades</i>
<i>1ª Fase</i>	Lista de membros da reunião Marcação de faltas dos membros Iniciar, encerrar, cancelar, suspender e retomar reunião Lista de pontos Colocar o ponto em discussão, em votação, encerrado, adiado ou suspenso Votar num ponto
<i>2ª Fase</i>	Alterar a hora de início e de fim da reunião Lista de entidades externas Adicionar e remover entidades externas Passar o controlo da reunião para outro membro Inserir, editar e remover pontos Colocar a votação como secreta Bloquear membro Ceder controlo do voto Inserir declaração de voto
<i>3ª Fase</i>	Sincronizar dados da reunião em tempo real Gerar notificações para determinadas ações
<i>4ª Fase</i>	Abrir ordem de trabalhos Abrir ata Abrir minuta Abrir anexos dos pontos

## 4.4 Metodologia de desenvolvimento

Foi seguida uma metodologia ágil, onde se optou por *software* funcional no lugar de documentação abrangente, pois a resposta a mudanças demonstrou ter uma maior importância que o seguimento do plano.

A metodologia adotada recentemente pela AIRC consiste, numa primeira fase, no levantamento de todas as funcionalidades e na realização do *layout* de um módulo da aplicação. No caso deste projeto, recorreu-se a uma empresa externa para a produção do *layout*.

Foram realizadas várias reuniões presenciais e por videoconferência com esta empresa externa, até esta entregar uns *mockups* que cumprissem todos os requisitos. Depois desses *mockups* serem aceites pelos órgãos superiores da AIRC, a empresa externa entregou o *layout* em HTML para ser mais fácil a sua reutilização.

Nesta fase, a equipa de desenvolvimento não está normalmente envolvida, estando apenas o *product owner*.

Depois desta fase estar concluída, o *product owner* inicia a escrita de *user story's*, para que a equipa de desenvolvimento saiba o que tem a fazer. Esta lista de *user story's* não deve ser muito extensa, e cada *user story* segue um *workflow* como mostra a Figura 7.

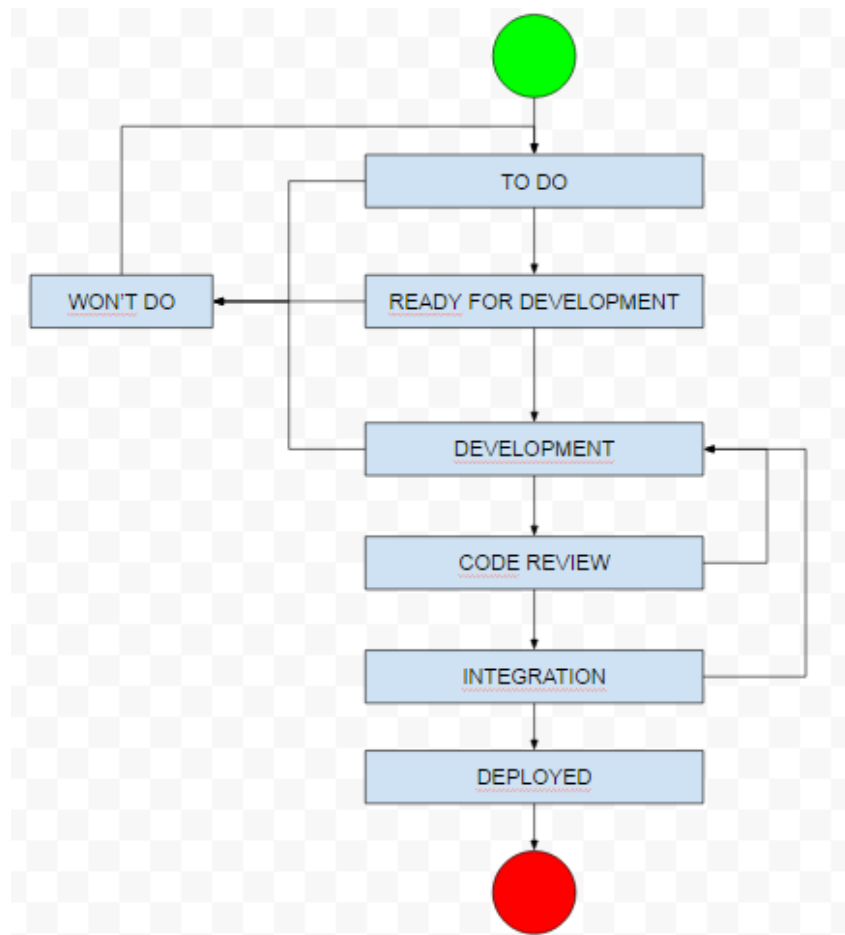


Figura 7 – Workflow de Desenvolvimento

## 4.5 Análise de requisitos

Nesta secção são apresentados os utilizadores que interagem com o sistema, e as *user stories* da plataforma.

### 4.5.1 Atores do sistema

Na plataforma existem 3 tipos de atores:

- Utilizador – Qualquer pessoa que tenha acesso a aplicação.
- Membro – Utilizador membro de órgãos deliberativos que participa na reunião.
- Controlador – Utilizador responsável pelo controlo da reunião (usualmente o secretário).

## 4.5.2 User Stories

### 4.5.2.1 UTILIZADOR

#### 4.5.2.1.1 Login

“Eu como utilizador quero aceder a informação que tenha direito conforme as minhas permissões”.

#### 4.5.2.1.2 Logout

“Eu como utilizador quero sair da aplicação”.

#### 4.5.2.1.3 Quem Sou

“Eu como utilizador quero saber sempre quem eu sou”.

#### 4.5.2.1.4 Onde estou

“Eu como utilizador quero saber sempre onde estou”.

#### 4.5.2.1.5 Reunião a decorrer

“Eu como utilizador quero saber sempre qual a reunião a decorrer”.

#### 4.5.2.1.6 Controlador da reunião a decorrer

“Eu como utilizador quero saber sempre quem está a controlar a reunião que esta a decorrer”.

#### 4.5.2.1.7 Pesquisar Documentos

“Eu como utilizador quero, em qualquer parte, poder pesquisar documentos de reuniões (caixa de texto com lupa)”.

#### 4.5.2.1.8 Critérios de pesquisa

“Eu como utilizador quero poder filtrar a pesquisa por texto, por datas, por órgão, por tipo de reuniões (ordinária, extraordinária, ordinária pública e extra pública)”.

#### 4.5.2.1.9 Resultados da pesquisa

“Eu como utilizador quero ver uma lista de documentos que cumpram os critérios de pesquisa e poder abrir esses documentos e ver a lista de parágrafos de um desses documentos e poder abrir o documento localizado nesse parágrafo”.

#### 4.5.2.1.10 Área de Notificações

“Eu como utilizador quero ter uma área de notificações (notificações de sucesso de operações, notificação de ponto em votação, etc.)”.

#### 4.5.2.1.11 Pagina inicial

“Eu como membro ou controlador quero poder ‘Entrar na Reunião a Decorrer’, ‘Abrir Reunião’ e ‘Consultar Reuniões’”.

#### 4.5.2.1.12 Entrar na Reunião

“Eu como membro ou controlador quero poder entrar na reunião que esteja a decorrer, vendo os vários detalhes da reunião”.

#### 4.5.2.1.13 Abrir Reunião

“Eu como membro ou controlador quero poder abrir uma reunião, vendo a lista de reuniões”.

#### 4.5.2.1.14 Lista de Reuniões

“Eu como membro ou controlador quero ver, inicialmente, a lista de todas as reuniões futuras, com a possibilidade de ver todas as reuniões já realizadas e ainda poder filtrar por datas”.

#### 4.5.2.1.15 Consultar Reuniões

“Eu como membro ou controlador quero poder consultar reuniões”.

#### 4.5.2.1.16 Critérios de consulta

“Eu como membro ou controlador quero poder filtrar a consulta por datas, por órgão, por tipo de reuniões (ordinária, extraordinária, ordinária pública e extra pública)”.

#### 4.5.2.1.17 Resultados da consulta

“Eu como membro ou controlador quero ver uma lista de reuniões que cumpram os critérios de consulta, e poder abrir os três documentos relacionados com essa reunião (‘Ordem de trabalhos’, ‘Minuta’ e ‘Ata’) se existirem”.

### 4.5.2.2 MEMBRO

#### 4.5.2.2.1 Lista de Pontos

“Eu como membro quero ver a lista de todos os pontos, com a informação dos seus estados (‘Em Discussão’, ‘Em Votação’, ‘Encerrado’, ‘Adiado’, ‘Suspensão’) e o resumo da votação se já ocorreu votação (total de votos ‘A favor’, ‘Contra’, ‘Abstenção’)”.

#### 4.5.2.2.2 Ponto a decorrer

“Eu como membro quero ver o ponto que esta a decorrer (‘Em Discussão’, ‘Em Votação’), destacado dos restantes”.

#### 4.5.2.2.3 Secções e Pontos Extra

“Eu como membro quero ver as secções e os pontos extra destacados dos restantes”.

#### 4.5.2.2.4 Lista de Anexos

“Eu como membro quero ver a lista de anexos do ponto seleccionado anteriormente, e poder abrir esses anexos”.

#### 4.5.2.2.5 Votar num ponto

“Eu como membro quero votar no ponto que está em votação (A favor’, ‘Contra’, ‘Abstenção’)”.

#### 4.5.2.2.6 Inserir uma declaração de voto

“Eu como membro, durante o voto, quero poder escrever uma declaração de voto ou anexar um ficheiro com a mesma”.

#### 4.5.2.2.7 Ceder Voto ao Controlador

“Eu como membro quero ceder o meu voto ao controlador”.

#### 4.5.2.2.8 Lista de resultados da votação

“Eu como membro quero ver o resultado da votação dos diferentes membros (‘A favor’, ‘Contra’, ‘Abstenção’, ‘Bloqueado’), se este ponto já foi votado”.

#### 4.5.2.2.9 Documento da Ordem de trabalhos

“Eu como membro quero poder ver o documento da ordem de trabalhos da reunião”.

### 4.5.2.3 CONTROLADOR

#### 4.5.2.3.1 Controlo da reunião

“Eu como controlador quero controlar a reunião”.

#### 4.5.2.3.2 Estado da reunião

“Eu como controlador quero iniciar a reunião, encerrar a reunião, cancelar reunião, suspender reunião e retomar reunião”.

#### 4.5.2.3.3 Horas da reunião

“Eu como controlador quero alterar a hora de início e de termino da reunião”.

#### 4.5.2.3.4 Lista de Membros Participantes da reunião

“Eu como controlador quero ver a lista de membros participantes da reunião e poder marcar faltas aos membros que não se encontram presentes”.

#### 4.5.2.3.5 Lista de Entidades ocasionais da reunião

“Eu como controlador quero ver a lista de Entidades ocasionais da reunião, e poder adicionar novas entidades e/ou remover entidades ocasionais existentes”.

#### 4.5.2.3.6 Criar ou abrir Minuta

“Eu como controlador quero poder criar uma nova Minuta ou abrir uma Minuta já existente, abrindo um ficheiro dessa minuta”.

#### 4.5.2.3.7 Passar Controlo da Reunião

“Eu como controlador quero poder passar o controlo da reunião a outra pessoa com permissões para tal”.

#### 4.5.2.3.8 Controlo dos pontos da reunião

“Eu como controlador quero controlar os pontos da reunião”.

#### 4.5.2.3.9 Lista de Pontos

“Eu como membro quero ver a lista de todos os pontos, com a informação dos seus estados (‘Em Discussão’, ‘Em Votação’, ‘Encerrado’, ‘Adiado’, ‘Suspendido’) e o resumo da votação se já ocorreu votação (total de votos ‘A favor’, ‘Contra’, ‘Abstenção’)”.

#### 4.5.2.3.10 Secções e Pontos Extra

“Eu como membro quero ver as secções e os pontos extra destacados dos restantes”.

#### 4.5.2.3.11 Ordem de Secções e Pontos Extra

“Eu como controlador quero poder alterar a ordem das secções e pontos extra”.

#### 4.5.2.3.12 Estado dos Pontos

“Eu como controlador quero alterar o estado do ponto (‘Para Discussão’, ‘Para Votação’, ‘Encerrar’, ‘Adiar’, ‘Suspendido’)”.

#### 4.5.2.3.13 Inserir Secção e/ou Ponto Extra

“Eu como controlador quero inserir uma nova secção e/ou um novo ponto extra. A secção é escolhida de uma lista predefina. O ponto contém uma descrição, e pode ter vários anexos”.

#### 4.5.2.3.14 Editar Ponto

“Eu como controlador quero poder editar um ponto extra”.

#### 4.5.2.3.15 Remover Secção e/ou ponto extra

“Eu como controlador quero poder remover uma secção e/ou um ponto extra”.

#### 4.5.2.3.16 Votação Secreta

“Eu como controlador quero, enquanto um ponto está em discussão, colocar a votação desse ponto como secreto”.

#### 4.5.2.3.17 Lista de Votação

“Eu como controlador quero ver os votos dos participantes de um ponto em tempo real”.

#### 4.5.2.3.18 Bloquear Membro

“Eu como controlador quero bloquear o voto de um membro, dizendo o motivo desse bloqueio (por ausência ou impedimento), e se este motivo for impedimento poder escrever um texto de justificação”.

#### 4.5.2.3.19 Votar por alguém

“Eu como controlador quero poder votar por um membro que me tenha cedido o voto”.



# 5 Desenvolvimento da Plataforma

## 5.1 Arquitetura

A solução foi estruturada seguindo o padrão *Model View Controller* (MVC), que separa a interação do utilizador com a representação da informação. Esta arquitetura assenta então em três vetores: modelo, vista e controlo, como demonstra a Figura 8.

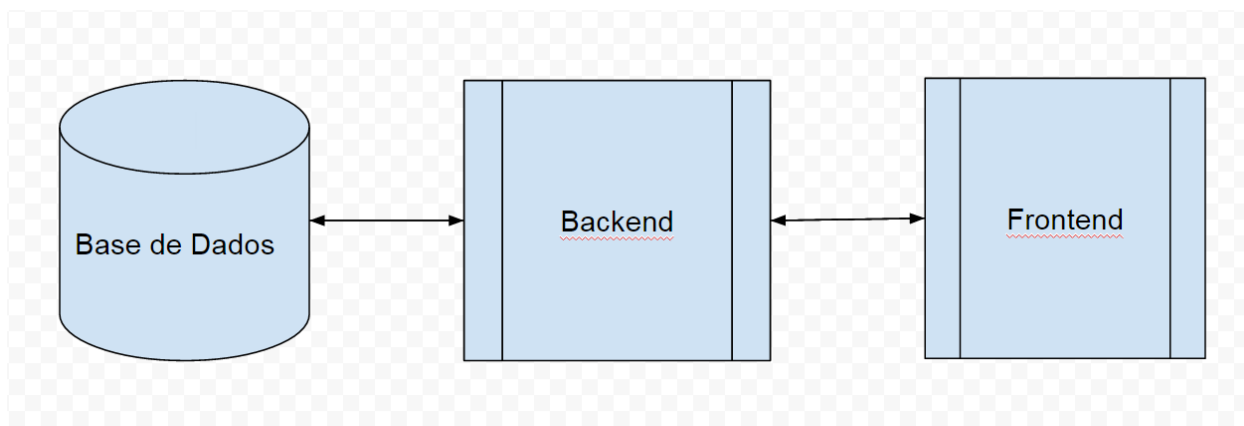


Figura 8 – Arquitetura da Solução

O modelo corresponde à base de dados, onde todos os dados são guardados. A vista corresponde ao *frontend* que será a interface *web* onde o utilizador irá interagir. O controlo corresponde ao *backend* que irá disponibilizar serviços para o *frontend*.

O *frontend* irá utilizar os serviços do *backend* através de chamadas HTTP, que utilizam JSON para a transferência de dados.

O paradigma de chamadas de HTTP em que a *web* tem sido criada, permite ao cliente carregar uma página da *web*, mas, em seguida, nada acontece até que o utilizador clique na próxima página. Neste caso isso não podia acontecer, pois as páginas dos vários utilizadores teriam de reagir de forma imediata às alterações efetuadas por outros utilizadores. Foi então decidido utilizar *websocket* para a sincronização de informação entre o *backend* e o *frontend*.

O *websocket* é uma tecnologia que permite a comunicação bidirecional, e foi desenvolvida para ser implementada em *browsers web* e servidores *web*, mas pode ser usada por qualquer cliente, aplicação ou servidor. O protocolo *websocket* é um protocolo independente baseado em TCP.



Sippl no final dos anos 70. A Informix foi numa parte da década de 90 o segundo motor de base de dados mais popular depois do Oracle, e foi a escolha por parte da AIRC há alguns anos.

## 5.3 Backend

O *backend* foi implementado em JAVA, com recurso ao Play! Framework [11]. O Play! Framework é uma *framework* MVC que facilita e torna o desenvolvimento *web* muito rápido. O programador segue convenções ditadas pela *framework*, para assim evitar a necessidade de configurações adicionais. Existe já um esqueleto da aplicação criado que é preciso seguir. Na raiz do projeto existem 4 pastas importantes:

**app** - Contém o modelo, a vista e o controlo do projeto;

**conf** – Contém os ficheiros de configuração como o *routes* ou o *application.conf*;

**project** – Contém os *build scripts* do projeto;

**public** – Contém recursos como *javascript*, *css* ou *imagens*.

O play! Framework utiliza o repositório sbt para gerir as dependências do projeto. Todavia, devido à inexistência de *drives* para o Informix nas bibliotecas do sbt, foi necessário adicionar uma nova pasta, a *lib*, onde foram colocadas manualmente os *drives* em falta.

O play! Framework segue um padrão totalmente *REpresentational State Transfer* (REST) onde todas as entradas de url's são configuradas no arquivo *routes* que contém 3 informações para cada url: o *HTTP Method*, o *Request Path* e a *Action*, como mostram os exemplos da Tabela 2.

Tabela 2 – Ficheiro de Rotas

<i>HTTP Method</i>	<i>Request Path</i>	<i>Action</i>
<i>POST</i>	/api/v1/login	airc.sta.controllers.Application.login()
<i>GET</i>	/api/v1/userinfo	airc.sta.controllers.Application.getUserInfo()

O ficheiro *application.conf* contém alguns parâmetros de configuração. Neste caso, foram guardados neste ficheiro os dados de acesso à base de dados para posteriormente serem utilizados nos serviços. O facto de este ficheiro ser apenas de texto permite que facilmente se altere ou troque esse ficheiro.

O *backend* segue o modelo MVC, e apesar de não ter uma interface com o utilizador, contém uma interface para disponibilizar os serviços. Esse foi organizado em 3 *packages* principais, os quais correspondem a cada uma das partes do MVC.

No modelo existe uma série de classes que representam o modelo de dados. Temos então 10 classes:

**Classe Anexo:** esta classe representa um anexo relativo a um ponto da ordem de trabalhos de uma reunião, Tabela 3.

Tabela 3 – Classe Anexo

<i>Nome</i>	<i>Tipo</i>	<i>Descrição</i>
<i>chave_fdocu</i>	int	Chave primária
<i>chave_fitem</i>	int	Chave forasteira de ponto da ordem de trabalhos
<i>numdoc_fdocu</i>	int	Número do documento
<i>documid_fdocu</i>	int	
<i>pathdoc_fdocu</i>	String	Path do documento
<i>dentrada_fdocu</i>	Date	Data de entrada do documento
<i>dtcria_fdocu</i>	Date	Data de criação do documento
<i>resumo_fdocu</i>	String	Resumo do documento
<i>remetente_fdocu</i>	String	Remetente do documento
<i>cod_capli</i>	String	Chave forasteira de aplicação
<i>descr_fdocu</i>	String	Descrição do documento
<i>tpreg_fdocu</i>	String	Tipo de registo do documento
<i>tpdoc_fdocu</i>	int	Tipo de documento
<i>descr_ffich</i>	int	Descrição do ficheiro

**Classe Entidade:** esta classe representa qualquer entidade de qualquer aplicação da AIRC, incluindo utilizadores ou entidades externas, Tabela 4.

Tabela 4 – Classe Entidade

<i>Nome</i>	<i>Tipo</i>	<i>Descrição</i>
<i>num_fenti</i>	int	Chave primária
<i>tpenti_fenti</i>	String	Tipo de entidade
<i>nome_fenti</i>	String	Nome completo da entidade
<i>abrevi_fenti</i>	String	Nome abreviado da entidade
<i>numcontrib_fenti</i>	String	Número de contribuinte da entidade
<i>inactiva_fenti</i>	int	Inatividade da entidade
<i>desigrua_fmora</i>	String	
<i>numpolicia_fmora</i>	String	
<i>sonumero_fmora</i>	String	

**Classe Entidade Membro:** esta classe representa uma entidade membro de uma reunião, Tabela 5.

Tabela 5 – Classe Entidade Membro

<i>Nome</i>	<i>Tipo</i>	<i>Descrição</i>
<i>chave_lpaac</i>	int	Chave primária
<i>chave_facta</i>	int	Chave forasteira
<i>natuser_lpaac</i>	int	Natureza da entidade membro na reunião
<i>presenca_lpaac</i>	int	Presença da entidade membro na reunião
<i>chave_lpart</i>	int	Chave Forasteira
<i>observ_lpart</i>	String	
<i>rsenha_lpart</i>	Short	Senha da entidade membro na reunião
<i>orgao_corga</i>	String	Chave forasteira do órgão
<i>desig_corga</i>	String	Designação do órgão
<i>chave_fuser</i>	int	Chave forasteira
<i>activo_lpart</i>	int	Inatividade da entidade membro
<i>desig_ctipo</i>	String	Designação do tipo de entidade membro
<i>cod_capli</i>	String	Chave forasteira
<i>configura_ctipo</i>	int	Gerir os votantes da reunião
<i>nome_fenti</i>	String	Nome da entidade
<i>controlador_lpaac</i>	int	Gerir os controladores da reunião

**Classe Entidade Ocasional:** esta classe representa uma entidade ocasional de uma reunião, Tabela 6.

Tabela 6 – Classe Entidade Ocasional

<i>Nome</i>	<i>Tipo</i>	<i>Descrição</i>
<i>chave_lpaac</i>	int	Chave primária
<i>chave_facta</i>	int	Chave forasteira
<i>natuser_lpaac</i>	int	Natureza da entidade ocasional na reunião
<i>presenca_lpaac</i>	int	Presença da entidade ocasional na reunião
<i>nome_fenti</i>	String	Nome da entidade
<i>num_fenti</i>	int	Número da entidade

**Classe Orgao:** esta classe representa um órgão da instituição, Tabela 7.

Tabela 7 – Classe Orgao

<i>Nome</i>	<i>Tipo</i>	<i>Descrição</i>
<i>chave_corga</i>	String	Chave primária
<i>descricao_corga</i>	String	Descrição do órgão
<i>pasta_corga</i>	String	Pasta dos ficheiros do órgão

**Classe Ponto:** esta classe representa um ponto da ordem de trabalhos de uma reunião, Tabela 8.

Tabela 8 – Classe Ponto

<i>Nome</i>	<i>Tipo</i>	<i>Descrição</i>
<i>chave_fitem</i>	int	Chave primária
<i>chave_facta</i>	int	Chave forasteira
<i>descr_fitem</i>	String	Descrição do ponto
<i>descr1_fitem</i>	String	Descrição do ponto
<i>adia_fitem</i>	int	Ponto adiado
<i>ordem_fitem</i>	int	Número na ordem de trabalhos
<i>estado_fitem</i>	int	Estado do ponto
<i>tp_fitem</i>	int	Tipo do ponto
<i>contra_fitem</i>	int	Total de votos contra
<i>extra_fitem</i>	int	Ponto extra
<i>abstencao_fitem</i>	int	Total de votos abstenção
<i>favor_fitem</i>	int	Total de votos a favor
<i>descr_estado_fitem</i>	String	Descrição do estado do ponto

**Classe Reunião:** esta classe representa um ponto da ordem de trabalhos de uma reunião, Tabela 9.

Tabela 9 – Classe Reuniao

<i>Nome</i>	<i>Tipo</i>	<i>Descrição</i>
<i>chave_facta</i>	int	Chave primária
<i>numero_facta</i>	int	Número da reunião
<i>data_facta</i>	Date	Data da reunião
<i>orgao_corga</i>	String	Chave forasteira
<i>otrab_facta</i>	String	Documento da ordem de trabalhos
<i>aprov_facta</i>	int	Reunião aprovada
<i>treuniao_facta</i>	String	Tipo de reunião
<i>inicio_facta</i>	Date	Data de início da reunião
<i>termo_facta</i>	Date	Date de fim da reunião
<i>desc_orgao_corga</i>	String	Descrição do órgão
<i>estado_facta</i>	int	Estado da reunião
<i>doc_facta</i>	String	Documento da ata
<i>minuta_facta</i>	String	Documento da minuta

**Classe Secção:** esta classe representa uma secção de pontos da ordem de trabalhos de uma reunião, Tabela 10.

Tabela 10 – Classe Seccao

<i>Nome</i>	<i>Tipo</i>	<i>Descrição</i>
<i>chave_cseco</i>	int	Chave primária
<i>chave_cdepa</i>	int	Chave forasteira
<i>num_duorg</i>	int	
<i>orgao_corga</i>	String	Chave forasteira
<i>desig_cseco</i>	String	Designação da secção
<i>ordem_cseco</i>	int	Numero de ordem

**Classe User:** esta classe representa um utilizador da aplicação, Tabela 11.

Tabela 11 – Classe User

<i>Nome</i>	<i>Tipo</i>	<i>Descrição</i>
<i>chave_fuser</i>	int	Chave primária
<i>login_fuser</i>	String	Nome do utilizador
<i>orgao_corga</i>	String	Chave forasteira
<i>activo_lpart</i>	int	Ativo
<i>rsenha_lpart</i>	int	
<i>chave_lpart</i>	int	Chave forasteira
<i>desig_ctipo</i>	String	Designação do cargo
<i>num_fenti</i>	int	Chave forasteira
<i>abrevi_fenti</i>	String	Abreviatura
<i>nome_fenti</i>	String	Nome completo

**Classe Voto:** esta classe representa um voto de um membro num ponto de uma reunião, Tabela 12.

Tabela 12 – Classe Voto

<b>Nome</b>	<b>Tipo</b>	<b>Descrição</b>
<i>chave_fvota</i>	int	Chave primária
<i>chave_fitem</i>	int	Chave forasteira
<i>chave_lpaac</i>	int	Chave forasteira
<i>chave_fdocu</i>	int	Chave forasteira
<i>dirvoto_fvota</i>	int	Tem direito a voto
<i>obsdirvoto_fvota</i>	String	Justificação do impedimento de voto
<i>tpvoto_fvota</i>	int	Tipo de voto
<i>obsvoto_fvota</i>	String	Descrição da declaração de voto
<i>declvoto_fvota</i>	int	Declaração de voto
<i>concl_fvota</i>	int	
<i>motivo_fvota</i>	int	Motivo do bloqueio de voto
<i>sistema_fvota</i>	int	Sistema de Voto
<i>pendente_fvota</i>	int	Cedência de controlo do voto
<i>controlo_fvota</i>	int	Controlo de voto
<i>descr_fitem</i>	String	Descrição do ponto
<i>descr_fenti</i>	String	Descrição da entidade

Na vista existe apenas uma classe que contém um método para cada serviço, e é responsável por retirar os parâmetros enviados pelo *frontend*, executar o serviço correspondente, e enviar os parâmetros necessários de volta para o *frontend*. Nestes métodos é verificada a autenticação dos utilizadores e se estes têm permissão para a ação determinada. A autenticação é realizada através do *Lightweight Directory Access Protocol (LDAP)*, ferramenta já implementada na AIRC. As permissões dos utilizadores, por dependerem da reunião, são efetuadas dentro do método.

No controlo existe uma série de classes que representam os serviços disponibilizados, agrupados por temas. Temos então 9 classes que estão representadas da Tabela 13 a Tabela 21.

Tabela 13 – Classe Service\_Anexo

<b>Método</b>	<b>Entradas</b>	<b>Saídas</b>	<b>Descrição</b>
<i>getAnexos</i>	int	List	Devolve a lista de anexos de um ponto

Tabela 14 – Classe Service\_Entidade

<b>Método</b>	<b>Entradas</b>	<b>Saídas</b>	<b>Descrição</b>
<i>getEntidade</i>	String	List	Devolve a lista de entidades com o nome ou número de contribuinte igual ao parâmetro passado

Tabela 15 – Classe Service\_EntidadeMembro

<b>Método</b>	<b>Entradas</b>	<b>Saídas</b>	<b>Descrição</b>
<i>getEntidadesMembrosReuniao</i>	int	List	Devolve a lista de entidades membros de uma reunião
<i>getEntidadesMembrosActaReuniao</i>	int	EntidadeMembro	Devolve a entidade membro através da chave da entidade
<i>getMembroControladorReuniao</i>	int	EntidadeMembro	Devolve a entidade membro controladora da reunião
<i>getEntidadesMembrosUser</i>	int, int	EntidadeMembro	Devolve a entidade membro através da chave do utilizador e da chave da reunião
<i>getEntidadesMembrosControladoresReuniao</i>	int	List	Devolve a lista de entidades membros que podem controlar a reunião
<i>setControladorReuniao</i>	int, int	int	Altera o controlador da reunião
<i>gravarPresencas</i>	int, boolean	int	Altera a presença de uma entidade membro numa reunião

Tabela 16 – Classe Service\_EntidadeOcasional

<b>Método</b>	<b>Entradas</b>	<b>Saídas</b>	<b>Descrição</b>
<i>getEntidadesOcasionaisReuniao</i>	int	List	Devolve a lista de entidades ocasionais de uma reunião
<i>GetEntidadeOcasionalId</i>	int, int	EntidadeOcasional	Devolve a entidade ocasional através da chave da entidade
<i>addEntidadeOcasional</i>	int, int	int	Adiciona uma entidade ocasional na reunião
<i>removeEntidadeOcasional</i>	int, int	int	Remove uma entidade ocasional na reunião

Tabela 17 – Classe Service\_Orgao

<b>Método</b>	<b>Entradas</b>	<b>Saídas</b>	<b>Descrição</b>
<i>getOrgaosLista</i>		List	Devolve a lista de órgãos
<i>getOrgaoChave</i>	String	Orgao	Devolve o órgão através da chave

Tabela 18 – Classe Service\_Ponto

<b>Método</b>	<b>Entradas</b>	<b>Saídas</b>	<b>Descrição</b>
<i>getPontosActa</i>	int	List	Devolve a lista de pontos de uma reunião
<i>getPontosExtras</i>	int	List	Devolve a lista de pontos extras de uma reunião
<i>getSeccoes</i>	String	List	Devolve a lista de secções de um órgão
<i>getPonto</i>	int	Ponto	Devolve ponto através da chave
<i>getPontoDecorrer</i>	int	Ponto	Devolve ponto a decorrer numa reunião
<i>setPosicaoPontoSubir</i>	int	int	Alterar posição ponto extra (Subir uma posição)
<i>setPosicaoPontoDescer</i>	int	int	Alterar posição ponto extra (Descer uma posição)
<i>setPosicaoPontoSubirTotal</i>	int	int	Alterar posição ponto extra (Subir todas as posições)
<i>setPosicaoPontoDescerTotal</i>	int	int	Alterar posição ponto extra ponto extra (Descer todas as posições)
<i>addPonto</i>	int, String	int	Editar ponto extra
<i>addSeccao</i>	int, String	int	Adicionar ponto extra
<i>deletePonto</i>	Ponto	boolean	Remover ponto extra
<i>editPonto</i>	Ponto, String	boolean	Editar ponto extra
<i>IniciardiscutirPonto</i>	Ponto	int	Colocar ponto em discussão
<i>IniciarvotarPonto</i>	Ponto	int	Colocar ponto em votação
<i>encerrarPonto</i>	Ponto	int	Colocar ponto como encerrado
<i>suspenderPonto</i>	Ponto	int	Colocar ponto suspenso
<i>adiarPonto</i>	Ponto	int	Colocar ponto como adiado

Tabela 19 – Classe Service\_Reuniao

<b>Método</b>	<b>Entradas</b>	<b>Saídas</b>	<b>Descrição</b>
<i>getReuniaoChave</i>	int	Reuniao	Devolve a reunião através da chave
<i>getReuniaoDecorrer</i>		Reuniao	Devolve a reunião a decorrer
<i>getReunioesLista</i>	Boolean, Date, Date	List	Devolve a lista de reuniões
<i>getReunioesConsultas</i>	Date, Date, Orgao, String	List	Devolve a lista de Reuniões
<i>setHorasReuniao</i>	int, Date, Date	Reuniao	Atualiza as horas de inicio e fim da reunião
<i>IniciarReuniao</i>	int, int	Reuniao	Dá início a reunião
<i>EncerrarReuniao</i>	int, int	Reuniao	Encerra a reunião
<i>CancelarReuniao</i>	int, int	Reuniao	Cancela a reunião
<i>pausarReuniao</i>	int, int	Reuniao	Pausa a reunião
<i>retomarReuniao</i>	int, int	Reuniao	Retoma a reunião

Tabela 20 – Classe Service\_User

<b>Método</b>	<b>Entradas</b>	<b>Saídas</b>	<b>Descrição</b>
<i>getUser</i>	String	User	Devolve o utilizador através do nome de utilizador
<i>getPermissoes</i>	int, int	int	Devolve as permissões de um utilizador numa reunião

Tabela 21 – Classe Service\_Votacao

<b>Método</b>	<b>Entradas</b>	<b>Saídas</b>	<b>Descrição</b>
<i>getVotacaoPonto</i>	int	List	Devolve os votos de um ponto
<i>getVotacaoPorVotar</i>	int	List	Devolve os votos de um ponto ainda não votados
<i>getVotacaochave</i>	int	Voto	Devolve o voto através da chave
<i>getVotacao</i>	int, int	Voto	Devolve o voto através da chave de membro e reunião
<i>Votar</i>	int, int, int, int, String, int	int	Votar
<i>BloquearVoto</i>	Voto, int, String	int	Bloquear voto de um membro
<i>VotarControlador</i>	int, int	int	Voto do controlador
<i>CederVoto</i>	int	int	Ceder o voto ao controlador
<i>setVotacaoSecreta</i>	Ponto, int	int	Alterar o sistema de voto para secreto

## 5.4 Frontend

Para a realização do *frontend* foi utilizada o *framework* AngularJS, mais especificamente a versão 2 [12]. Este é uma *framework* JavaScript *open-source*, da Google, que facilita a execução de aplicações.

O Angular 2 segue o modelo MVC que separa a vista, o modelo e o controlo.

Existe já um esqueleto da aplicação criado pelo Angular 2, o qual é preciso seguir. Na raiz do projeto existem diversos ficheiros e pastas de configurações, pois as dependências são geridas pelo próprio Angular. A pasta que irá conter o projeto em si é a pasta *src*.

Dentro desta existe uma pasta para armazenar todos os recursos, como *css*, *imagens*, etc..

Dentro da pasta *app* foi criado um esqueleto da aplicação, para facilitar o seu desenvolvimento e manutenção. Foram criadas 3 pastas principais: *models*, *components*, *services*.

Na pasta *models* existe uma serie de classes que representam o modelo de dados, à semelhança do que foi descrito no *backend*.

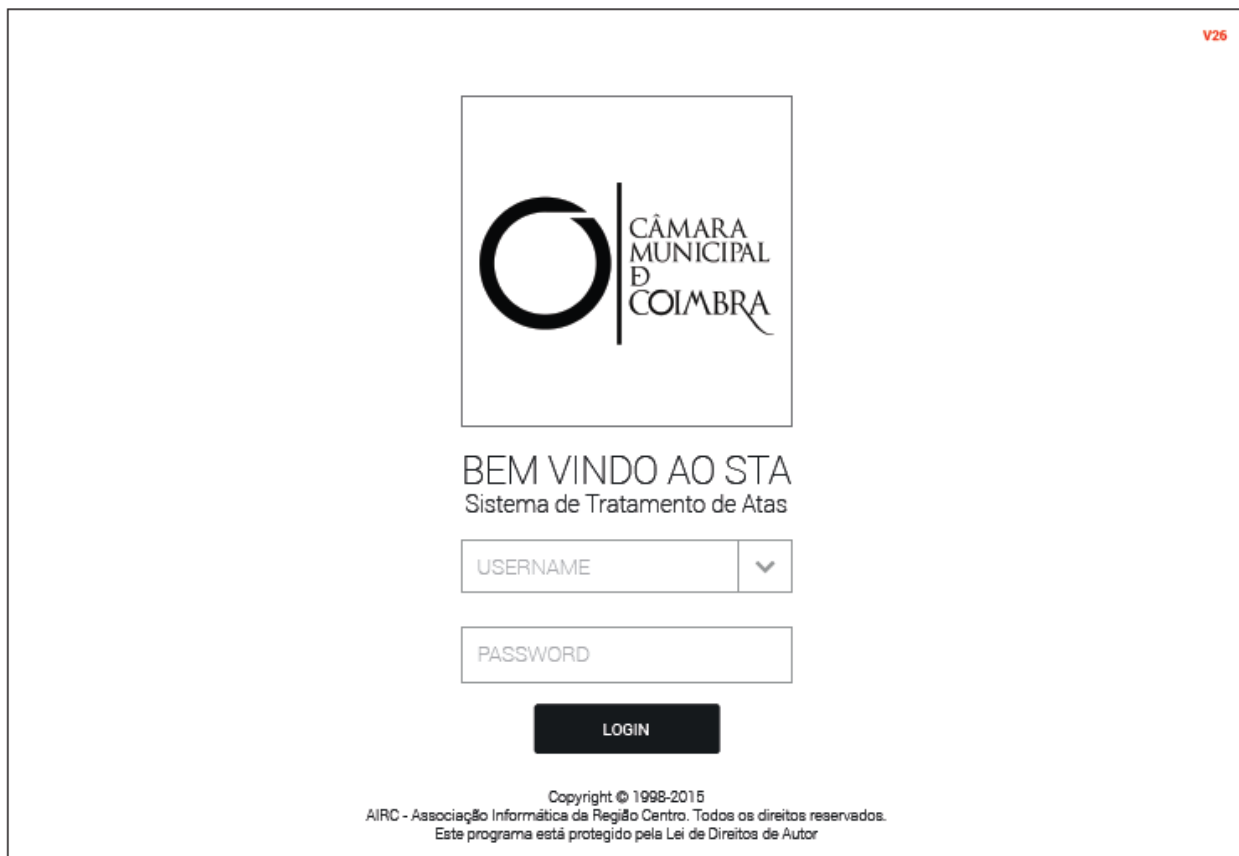
Na pasta *services* existe uma serie de classes que representam os serviços disponibilizados pelo *backend*, agrupados por temas. É aqui que é feita a ligação com o *backend*.

Por último, na pasta components existe um conjunto de pastas, onde cada uma representa um componente das várias páginas. Em regra, cada pasta destas possui 3 ficheiros: .component.css, .component.html e .component.ts.


O ficheiro component.css irá conter os estilos para essa página específica. O ficheiro component.html irá conter a vista da página. Como o nome indica, este ficheiro está escrito em html, e é aqui que se encontra descrita toda a estrutura da página. Como o Angular se baseia em componentes, para evitar trabalho desnecessário, foi utilizada uma biblioteca de componentes para Angular2, o PrimeNG. Por último, o ficheiro component.ts irá conter todo o controlo da vista, e está escrito em TypeScript.

De seguida, serão descritas as várias páginas da aplicação.

Existe uma página para efetuar o login, Figura 10, que permite introduzir o nome do utilizador e a respetiva password.



V26



BEM VINDO AO STA  
Sistema de Tratamento de Atas

USERNAME

PASSWORD

LOGIN

Copyright © 1998-2015  
AIRC - Associação Informática da Região Centro. Todos os direitos reservados.  
Este programa está protegido pela Lei de Direitos de Autor

Figura 10 – Pagina Login

Existe uma página principal, Figura 11, que permite escolher o que se pretende fazer: entrar numa reunião que já esteja a decorrer, abrir uma reunião ou consultar uma reunião.



Figura 11 – Pagina Inicial

Devido a alguns componentes serem constantes, em todas as páginas foi criado um ficheiro que contém o modelo base dessas páginas, e que contempla o menu lateral, a barra superior e o menu de notificações.

O menu lateral, Figura 12, contém a identificação do utilizador, bem como os menus necessários para a navegação entre as diversas páginas.

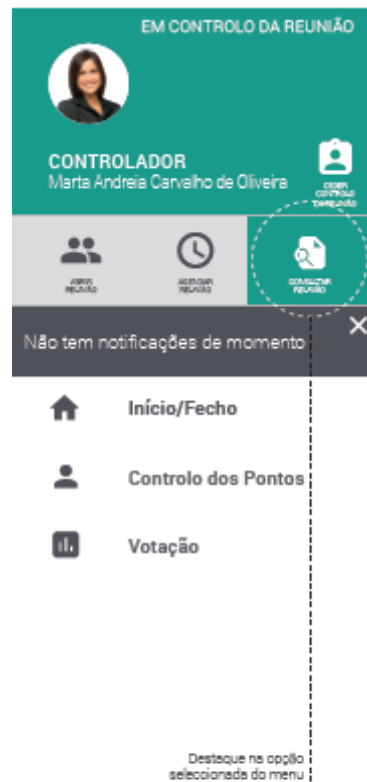


Figura 12 – Menu Lateral

A barra superior, Figura 13, possui um botão para efetuar *logout*, um texto que identifica a reunião onde o utilizador se encontra, e um campo de texto para fazer a pesquisa de reuniões mais facilmente.



Figura 13 – Barra Superior

Esta barra ainda possui um ícone que abre um menu de notificações, Figura 14, onde irão estar todas as notificações de que esse utilizador foi alvo.

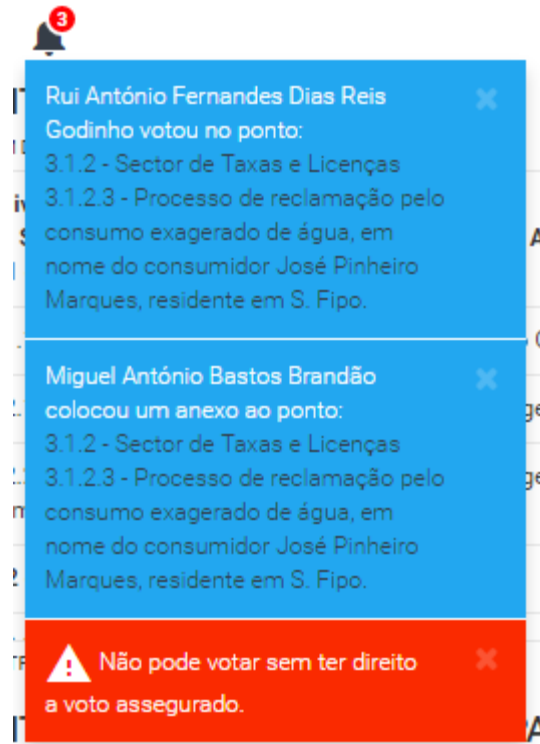


Figura 14 – Menu de Notificações

Existem duas páginas que são comuns a todos os utilizadores: as consultas de reuniões, Figura 15, e a pesquisa de reuniões, Figura 16.

**CONSULTAR REUNIÕES**

ENTRE DATAS DE 00-00-0000 A 00-00-0000

ORGÃO DA ATA: Câmara Municipal

TIPO REUNIÃO:  ORDINÁRIA  EXTRAORDINÁRIA  ORDINÁRIA PÚBLICA  EXTRA PÚBLICA  TODAS

**REUNIÕES ENCONTRADAS**

NÚMERO DA ATA	DATA DA REUNIÃO	ORGÃO DA REUNIÃO	TIPO DA REUNIÃO	ESTADO DA ATA
4	20-02-2015	Câmara Municipal	Ordinária	Aprovada
5	05-03-2015	Câmara Municipal	Ordinária	Aprovada
6	20-03-2015	Câmara Municipal	Ordinária	Aprovada
7	10-04-2015	Câmara Municipal	Ordinária	Aprovada
8	24-04-2015	Câmara Municipal	Ordinária	Aprovada
9	08-05-2015	Câmara Municipal	Ordinária	Aprovada
10	22-05-2015	Câmara Municipal	Ordinária	Aprovada
11	05-06-2015	Câmara Municipal	Ordinária	Aprovada

80 ata(s) encontrada(s)

ABRIR ATA ABRIR MINUTA ABRIR ORDEM DE TRABALHOS FECHAR AJUDA

Figura 15 – Página de Consultas



Figura 16 – Página de Pesquisas

A consulta de reuniões consiste na procura de uma reunião através de determinados parâmetros, como o órgão, o tipo de reunião e a data. Após esta consulta, é possível abrir o ficheiro da ordem de trabalhos, da minuta ou da ata, caso os mesmos existam.

A pesquisa de reuniões consiste na procura de um determinado texto no ficheiro da ordem de trabalhos, da minuta e da ata, de uma reunião. Poderá também ser efetuada através de determinados parâmetros como o órgão, o tipo de reunião e a data, filtrando ainda mais essa pesquisa. Após esta pesquisa, é possível abrir os ficheiros que cumpriram todos os critérios ou escolher o parágrafo onde o texto se encontra e abrir o ficheiro posicionado nesse parágrafo.

A página de votação, Figura 17, centraliza numa única página todas as ações que um membro necessita de realizar. Permite ao utilizador ver todos os pontos da ordem de trabalhos dessa reunião, bem como o seu estado, os seus anexos e o resultado da votação, se esta já tiver decorrido. Permite ainda ver qual o ponto que está em discussão ou em votação nesse momento e se for caso disso, votar num ponto, permitindo inserir uma declaração de voto. Além disso, permite ceder o controlo do seu voto ao controlador.

**PONTOS PARA VOTAÇÃO**

LISTA DE PONTOS

ESTADO DO PONTO	👍	👎	🗳️	
8.1.1.1 - Apreciação e aprovação do Regulamento do Conselho Municipal da Juventude do Município de Condeixa-a-Nova	ENDERRADO	2	2	2
8.1.2.1 - Processo de reclamação pelo consumo exagerado de água, em nome do consumidor José Gonçalves Nuno, residente em Eira - Pedrinha	ABERTO A VOTAÇÃO	-	-	-
8.1.2.2 - Apreciação e aprovação do Regulamento do Conselho Municipal da Juventude do Município de Pedrinha	ENDERRADO	0	0	1
8.1.2.3 - Pedido de subsídio por parte dos Serviços Sociais dos Trabalhadores do Município de Condeixa-a-Nova	ENDERRADO	3	2	1
8.1.2.3 - Processo de reclamação pelo consumo exagerado de água, em nome do consumidor Maria da Conceição Duro, residente em Coimbra	NÃO DISCUTIDO	-	-	-

DOCUMENTOS ANEXOS AO PONTO SELECIONADO

- Declarção EPAL - Consumos Agosto de 2018
- Declarção EPAL - Consumos Junho de 2018
- Declarção EPAL - Consumos Julho de 2018

**VOTAÇÃO**

A FAVOR | CONTRA | ABSTENÇÃO

DECLARAÇÃO DE VOTO

FICHEIRO DA DECLARAÇÃO DE VOTO

RESULTADOS DA VOTAÇÃO DE CADA PARTICIPANTE

VOTOS
António Miguel Costa Baptista
Marta Andreia Carvalho de Oliveira
Miguel António Bastos Brandão
Bérgio Luis Rodrigues Secco
Ana Maria dos Santos Marreco e Rosa de Gouveia
Rui António Fernandes Dias Reis Godinho
Célia Maria da Conceição Mateus
Ricardo Antunes Ferreira Castilho

Figura 17 – Página de Votação

Esta página apenas aparece ao utilizador que possua permissões para votar nessa reunião.

A página de início/fecho, Figura 18, permite marcar a presença dos membros da reunião e adicionar ou remover entidades ocasionais. Permite ainda controlar o estado da reunião, bem como as suas horas de início e fim.

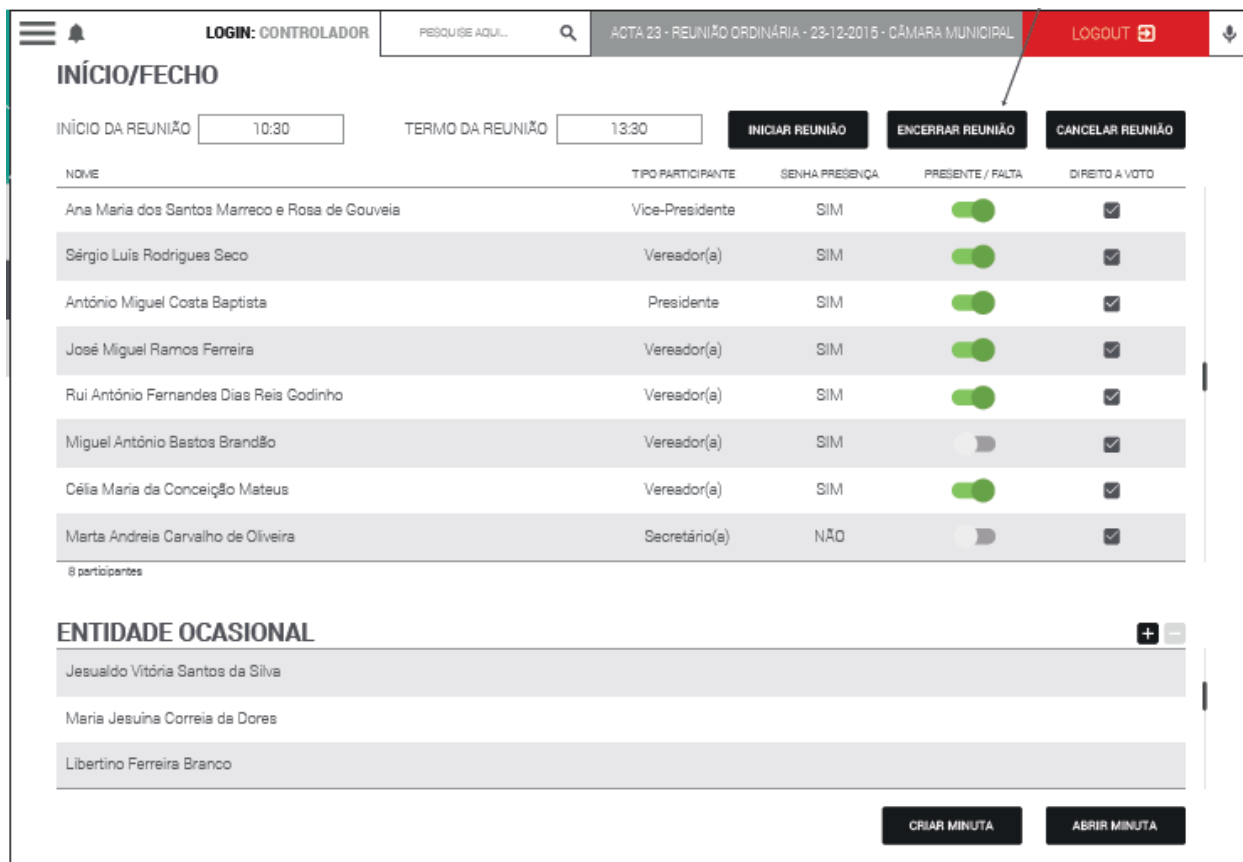


Figura 18 – Página Início/Fecho

Esta página apenas aparece ao utilizador que possua permissões para controlar essa reunião.

A página de controlo de pontos, Figura 19, permite gerir os pontos da ordem de trabalhos, podendo adicionar, editar e remover novos pontos, alterar o seu estado e votar por aqueles que lhe cederam o controlo de voto. Permite, ainda, colocar a votação como secreta e bloquear o voto de um membro por este se encontrar ausente ou por existir um impedimento.

**LOGIN: CONTROLADOR** PESQUISE AQUI... ACTA 23 - REUNIÃO ORDINÁRIA - 23-12-2016 - CÂMARA MUNICIPAL LOGOUT

### CONTROLO DOS PONTOS

ORDÉM DO DIA	ESTADO	ESTADO DO PONTO
<b>3 - Divisão Administrativa e Financeira</b>		
<b>3.1 - Secção de Expediente, Arquivo, Modernização, Apoio aos Órgãos, Taxas e Licenças.</b>		
<b>3.1.1 - Expediente, Arquivo e Modernização</b>		
3.1.1.1 - Apreciação e Aprovação do Regulamento do Conselho Municipal da Juventude do Município de Condeixa-a-Nova.	ENCERRADO	PARA DISCUSSÃO <b>PARA VOTAÇÃO</b> ENCERRA PONTO SUSPENDE PONTO ADIA PONTO
3.1.2.1 - Processo de reclamação pelo consumo exagerado de água, em nome do consumidor José Gonçalves Nujo, residente em Eira Pedrinha.	ENCERRADO	
3.1.2.2 - Processo de reclamação pelo consumo exagerado de água, em nome do consumidor Luis Filipe Parreira Duarte, residente em Urbanização Nova Conímbriga.	ENCERRADO	
<b>3.1.2 - Sector de Taxas e Licenças</b>		
3.1.2.3 - Processo de reclamação pelo consumo exagerado de água, em nome do consumidor José Pinheiro Marques, residente em S. Fipo.	ABERTO A DISCUSSÃO	REAGENDAMENTO DA ORDEM DO DIA <b>NOVO PONTO</b> <b>EDITAR</b> <b>APAGAR</b>
Deliberou por unanimidade aprovar a referida reclamação dispondo uma indemnização ao proprietário.		
3.1.2.4 - Apreciação e Aprovação do Regulamento do Conselho Municipal da Juventude do Município de Lordela.		<b>ATIVAR VOTO SECRETO</b>

### CONTROLO DE VOTO POR PARTICIPANTE

PARTICIPANTES

António Miguel Sousa Baptista	🔒
Sérgio Luis Rodrigues Seco	🔒
Marta Andreia Carvalho de Oliveira	🔒
Célia Maria da Conceição Mateus	🔒

### IMPEDIMENTO DE VOTO

MOTIVO

Ausente  Impedimento de voto

DESCRIPÇÃO DO IMPEDIMENTO DE VOTO

GUARDAR

Figura 19 – Página Controlo de Pontos

Esta página apenas aparece ao utilizador que possua permissões para controlar essa reunião.



# 6 Conclusões

## 6.1 Trabalho realizado

Este relatório permitiu descrever todo o trabalho realizado na unidade curricular de estágio ou projeto industrial e dar a conhecer ao leitor as potencialidades e estratégias que a plataforma de reuniões online implementam, de modo a facilitar a gestão de reuniões de órgãos deliberativos.

Apesar de já existirem algumas plataformas de reuniões online, estas não estão vocacionadas para reuniões de órgãos deliberativos. No estudo realizado, nenhuma das plataformas existentes incluía na íntegra as funcionalidades pretendidas.

Na sua maioria, os objetivos planeados para este estágio foram cumpridos, desde a análise das plataformas existentes até à implementação da 3ª fase de requisitos. Foi, no entanto, excluída do âmbito do estágio a fase formal de testes e validação, que será apenas executada após o término do estágio.

Apesar de alguns desvios na calendarização das tarefas, podemos considerar muito positivos os resultados finais obtidos, tendo em conta que no final da 3ª fase já será possível testar a plataforma num cliente, faltando apenas toda a interação com ficheiros.

Em resumo, nesta fase, o cliente consegue já ter um ambiente onde pode iniciar uma reunião, fazer a gestão da mesma, bem como votar nos diversos pontos da ordem de trabalhos.

## 6.2 Trabalho futuro

As sugestões para a continuidade deste trabalho apontam, essencialmente, no aperfeiçoamento do código para diminuir a dificuldade de alterações de funcionalidades da aplicação. Uma das melhorias seria a utilização de ORM em vez de JDBC, utilizando o *Hibernate*. Outra melhoria seria a utilização de melhores práticas, no que se refere ao padrão RESTfull.

Foi ainda sugerido, por um cliente, algumas novas funcionalidades:

- Votação em bloco, de modo a poder discutir e votar vários pontos ao mesmo tempo;
- Pontos sem votação, permitindo encerrar um ponto sem haver votação do mesmo;
- Notificações do browser, de modo a mostrar as notificações mesmo que o utilizador esteja noutro website.

O trabalho futuro passará, também, pela implementação de testes de qualidade, bem como a validação pelo cliente. Os testes propostos foram testes e2e, tanto do lado do *backend* como do *frontend*.

## 6.3 Dificuldades

No estágio surgiram algumas dificuldades e desafios:

- A existência de uma aplicação anterior dentro da AIRC limitou olhar para esta solução como um projeto novo, o que dificultou na criação do *layout*. Essa dificuldade verificou-se com a demora na criação do *layout* final por parte da empresa externa responsável por esta parte;
- A existência de um modelo de dados já criado para outras aplicações, limitou a criação de uma solução mais otimizada;
- A fase de reestruturação em que a AIRC se encontrava no momento do início deste estágio, fez com que o projeto se atrasasse bastante;
- A utilização de novas tecnologias de desenvolvimento, muitas delas ainda em versões experimentais, aliada à falta de documentação destas mesmas tornou mais difícil a sua aprendizagem.

## 6.4 Benefícios

Terminada a etapa da experiência pré-profissionalizante, o estagiário concluiu que esta experiência foi um marco de muita relevância no percurso acadêmico e uma aprendizagem crucial para a futura prática profissional.

Todas as dificuldades foram sempre encaradas como desafios a serem ultrapassados, e como aprendizagens adquiridas.

Este estágio permitiu ao estagiário ser autodidata na aprendizagem de novas tecnologias, de novas ferramentas, de novos conceitos, de mapear o tempo necessário para a execução de cada tarefa, tendo-lhe ainda proporcionado a integração no mercado de trabalho.

De um ponto de vista mais técnico, foi possível através deste estágio fortalecer conhecimentos em Java. A linguagem Java, que se encontra em grande utilização neste momento, constitui o padrão global para o desenvolvimento e distribuição de aplicações móveis e incorporadas, jogos, conteúdo baseado em *web* e *softwares* corporativos. Dentro da linguagem Java foi possível adquirir conhecimentos em diversos aspetos. Foi possível adquirir conhecimentos na utilização de Play! Framework, uma *framework* bastante poderosa na criação de aplicações *web*, e na utilização de JDBC, um API utilizado para interagir com a base de dados.

Para além de todo o conhecimento adquirido na área de Java, foi também possível ao estagiário melhorar e apreender conceitos de SQL, JavaScript, TypeScript, css, Json, sbt, WebSockets, HTTP Method e Angular2, não podendo deixar de parte todo o reforço de experiência e conhecimento que também foi adquirido no contexto de gestão de projeto de software, assim como a consciencialização da sua importância para o sucesso do seu desenvolvimento.

Foi sem dúvida, uma mais valia poder aprimorar conhecimentos técnicos nestas áreas.

O estagiário empenhou-se ao longo de todo o período de estágio, realizando todas as tarefas solicitadas pelo seu orientador e questionando todos os aspetos que considerava relevantes para poder apreender o máximo de conhecimentos possíveis.

Esta experiência de estágio foi, sem dúvida, enriquecedora tendo proporcionado a descoberta de novos saberes que, só com a prática se conseguem adquirir.



# Referências

- [1] ISEC, [www.isec.pt](http://www.isec.pt), acessido em outubro de 2016
- [2] DEIS, <http://www.deis.isec.pt/>, acessido em outubro de 2016
- [3] MIS, [http://www.deis.isec.pt/curso\\_mei.aspx](http://www.deis.isec.pt/curso_mei.aspx), acessido em outubro de 2016
- [4] AIRC, <http://www.airc.pt>, acessido em outubro de 2016
- [5] Código do Procedimento Administrativo, [http://www.pgdlisboa.pt/leis/lei\\_mostra\\_articulado.php?nid=2248&tabela=leis&so\\_miolo=](http://www.pgdlisboa.pt/leis/lei_mostra_articulado.php?nid=2248&tabela=leis&so_miolo=), acessido em outubro de 2016
- [6] STA, <http://www.airc.pt/produtos/erp-airc/sta-sistema-de-tratamento-de-atas>, acessido em outubro de 2016
- [7] TeamViewer, <https://www.teamviewer.com/PT/use-cases/meetings-and-collaboration/>, acessido em outubro de 2016
- [8] MeetingBurner, <https://www.meetingburner.com/index?page=features>, acessido em outubro de 2016
- [9] WebEx, <https://www.webex.com/why-webex/overview.html#why-webex>, acessido em outubro de 2016
- [10] Fuze, <https://www.fuze.com/unified-communications/collaboration#hd-audio>, acessido em outubro de 2016
- [11] Play Framework, <https://www.playframework.com/documentation/2.5.x/Home>, acessido em Novembro de 2016
- [12] Angular 2, <https://angular.io/docs/ts/latest/guide/>, acessido em Novembro de 2016



# Anexo A - Proposta de estágio

## **Proposta de Projeto / Estágio**

Ano Letivo de 2015 / 2016

### **TEMA** **Reuniões Online**

#### **SUMÁRIO**

Controlo e gestão da reunião de órgãos deliberativos e respetivas tomadas de decisão.

Este projeto destina-se a um estagiário.

Desenvolvimento de uma área de trabalho para controlo da reunião.

#### **ÂMBITO**

Com o avanço das tecnologias e a necessidade de disponibilizar a informação resultante de reuniões dos órgãos deliberativos, surge a necessidade do controlo e gestão da reunião através do módulo de Reuniões Online.

Pretende-se disponibilizar aos membros de reuniões de órgãos deliberativos, o acompanhamento e a realização da votação dos pontos em discussão em tempo real, onde possam, através do sistema, visualizar toda a informação e efetuar as ações necessárias para a realização da reunião.

Este processo baseia-se sobre um conjunto de estados condicionantes de acesso, impostas pelos vários níveis de permissão, que influenciam na interface e no modo como é efetuada a sincronização entre os diversos postos de trabalho.

#### **OBJETIVOS**

O presente projeto pretende atingir um conjunto de objetivos genéricos, que passam por diversos requisitos:

- Estado da reunião, que permite identificar a iniciação da reunião e a respetiva reunião com a indicação do estado dos diversos pontos a debater, se encontram em preparação ou em votação, bem como o encerramento da reunião.
- Estado em que se encontram os diversos pontos da ordem de trabalhos, nomeadamente os pontos por votar, abertos a votação, pontos adiados ou suspensos e pontos encerrados.
- Acesso à reunião, que permite a visualização das reuniões em aberto para discussão, disponível a membros da reunião com e sem acesso ao direito de voto.

- Consulta de documentos base para a reunião, permitindo a visualização da ata da reunião anterior, disponível a participantes com direito de voto na reunião anterior. Incluindo ainda a assinatura da respetiva reunião via cartão de cidadão.
- Visualização da ordem de trabalhos da reunião para participantes com ou sem direito de voto e respetivo acesso aos documentos anexos aos pontos disponibilizados.
- Consulta por texto livre em todas as atas aprovadas, disponível a tempo real a participantes da reunião.
- Métodos de votação e gravação de intervenções por utilizador com a possibilidade de transcrição de dados de voz, para membros com direito de voto.
- Seleção dos pontos a serem votados e respetiva votação inserindo no sistema o resultado do voto que pretende declarar, com o objetivo de decretar a tomada de decisão, e selecionar o resultado dessa mesma decisão (a favor, abstenção ou contra).
- Inserção de declarações de voto, que pretende inserir no sistema a informação relacionada com o ponto a ser votado, com o objetivo de publicar a declaração de voto.
- Troca de mensagens entre participantes com direito a voto, possibilitando inserir no sistema informação, usando serviços JMS com o objetivo de trocar mensagens entre membros da reunião.
- Possibilidade de inserção de pontos extraordinários, passando os mesmos a fazer parte dos pontos extra da ordem de trabalhos, com o objetivo de colocar em discussão e votação. Pretendendo a discussão de aceitação do ponto extraordinário, ou seja, a aceitação por parte do *BackOffice* e futura validação do ponto.

## **PROGRAMA DE TRABALHOS**

O estágio consistirá nas seguintes atividades e respetivas tarefas:

- **T1** – Formação e Análise – Aquisição de conhecimentos nas ferramentas de desenvolvimento identificadas para a elaboração do projeto e identificação das funcionalidades a implementar.
- **T2** – Desenho – Apresentação de soluções de acordo com as necessidades recolhidas.
- **T3** – Implementação – Construção dos âmbitos identificados, considerando a definição e criação das diferentes fases de produção.
- **T4** – Testes e Validação – Execução de testes para validação das tarefas desenvolvidas.
- **T5** – Deployment – Disponibilização dos resultados.

## **CALENDARIZAÇÃO DAS TAREFAS**

O plano de escalonamento dos trabalhos é apresentado em seguida:

Tarefas	Meses					
	N	N+1	N+2	N+3	N+4	N+5
T1	■	■				
T2			■	■		
T3				■	■	
T4					■	■
T5						■
Metas	INI	M1	M2	M3		M4 M5

INI		Início dos trabalhos
M1	(INI + 6 Semanas)	Tarefa T1 terminada
M2	(INI + 10 Semanas)	Tarefa T2 terminada
M3	(INI + 14 Semanas)	Tarefa T3 terminada
M4	(INI + 22 Semanas)	Tarefa T4 terminada
M5	(INI + 24 Semanas)	Tarefa T5 terminada

## LOCAL E HORÁRIO DE TRABALHO

Local de trabalho:

**AIRC** - Associação de Informática da Região Centro  
 Parque Industrial de Taveiro, Lote 48, Apartado 2  
 3045-503 Coimbra

Horário: Segunda a Sexta das 9h às 13h00min e das 14h às 18h00min.

## METODOLOGIA

Deverá, atempadamente, ser elaborado um dossier de projeto onde constarão os diversos artefactos e tarefas executados na elaboração do trabalho, bem como das decisões tomadas nas diversas reuniões de acompanhamento.



