

## Sistema de Votação Eletrónica para Eleições e Aprovação de Assuntos

### Proposta de Projeto

**Orientador:** Pedro R. M. Inácio(inacio@di.ubi.pt)

**Coorientadora:** Paula Prata(pprata@di.ubi.pt)

## Objetivos

Pretende-se neste projeto desenhar e desenvolver um sistema composto por 2 peças de software distintas, ainda que potencialmente interligadas no modelo de dados: (i) uma peça de software de votação eletrónica para eleições simples e (ii) uma peça de software para votação em assuntos e produção de texto para ata. A ideia é que todo o software desenvolvido execute num sistema que possa funcionar em modo *stand-alone* (i.e., se necessário, o próprio sistema gera um *hotspot* onde os utilizadores se possam ligar para utilizar o programa). Pode considerar-se um computador Raspberry Pi 3 como o hardware alvo, embora a maior parte da implementação e testes possa ser feita num computador *desktop*. Alguns dos *outputs* do projeto serão, por isso, as implementações das duas peças, seus *scripts* de instalação, e a imagem do sistema operativo onde estas estão totalmente funcionais.

Os requisitos das 2 peças devem ser detalhadas na parte inicial do projeto, mas tecem-se de seguida algumas considerações base para cada uma delas. Para o software de votação eletrónica para eleições:

- Devem ser suportados 3 tipos de utilizadores (*super admin*; *membro da comissão eleitoral*; e *eleitor*). A única restrição é que um *super admin* não pode ser *membro da comissão eleitoral*;
- O *super admin* existe por defeito no software, e pode definir a sua palavra-passe ou outros *super admins*;
- O *super admin* pode criar um ato eleitoral via a definição de um titulo para esse ato;
- Os *membros da comissão eleitoral* registam-se via portal deste software e a sua inclusão como membros de determinada comissão fica pendente da aceitação de um *super admin*. Quando um membro se tenta registar, deve indicar o ato para o qual será membro da comissão;
- Os *membros da comissão eleitoral* ajustam todos os detalhes de um determinado ato eleitoral, nomeadamente nomes e *e-mails* dos eleitores, período do ato eleitoral (data de abertura e data de fecho), nome dos candidatos, e se haverá hipótese de haverem votos em branco (disponível por defeito) ou nulos (disponível por defeito);
- Assim que abra o período para o ato eleitoral, o sistema deve enviar *e-mails* para todos os eleitores com *links* de voto únicos para cada um, gerados aleatoriamente;

- O sistema deve garantir que cada eleitor só vota uma vez;
- O sistema deve guardar a informação de quem votou e o total de votos, mas não deve guardar a intenção de voto dos eleitores;
- Os detalhes de determinado ato eleitoral só podem ser ajustados na presença de pelo menos dois membros da comissão eleitoral;
- Durante o período do ato eleitoral, nenhum *super admin* ou *membro da comissão eleitoral* deve poder aceder ao sistema;
- Os resultados do ato eleitoral só devem ser mostrados se dois membros da comissão eleitoral estiverem presentes. Depois dos resultados serem vistos e aprovados pela comissão eleitoral, estes devem ficar disponíveis para consulta num *link* gerado aleatoriamente durante um período de tempo estipulado pela comissão;
- Após o período de tempo referido no ponto anterior, o sistema deve automaticamente colecionar todos os detalhes e artefactos da votação num ficheiro zip, apagando-os do sistema logo de seguida. Uma nova votação deve começar do estado *clean slate*;
- (opcional) No final de um ato eleitoral, deve produzir-se um relatório final automaticamente que é assinado digitalmente por pelo menos dois membros da comissão eleitoral;
- (opcional) As autenticações são feitas através do cartão do cidadão;
- (opcional) Devem ser pensadas formas do sistema atuar autonomamente de dentro de um cofre durante o período do ato eleitoral (Raspberry Pi com Bateria + especificação do cofre).

Para o software de votação eletrónica para aprovação de assuntos:

- Devem ser suportados 4 tipos de utilizadores (*presidente da reunião*, *secretario da reunião*; *preparador*; e *membro da comissão*);
- O utilizador *presidente da reunião* existe por defeito no software e pode definir a sua palavra-passe;
- Todos os outros utilizadores fazem o registo na plataforma através do portal deste software. Aquando do registo, todos os utilizadores ficam com o perfil de *membro da comissão*;
- O *presidente da reunião* pode mudar o perfil de um utilizador registado para *secretario da reunião* ou para *preparador*;
- Só o *presidente da reunião* ou o *preparador* é que podem agendar uma reunião no sistema, indicando o título da reunião, a data e o local;
- Só o *secretario da reunião* ou o *preparador* é que podem introduzir assuntos (título e descrição) a serem analisados na reunião;
- Apenas o *presidente da reunião*, o *secretário da reunião* e os *membros da comissão* é que podem votar;

- Durante a reunião, o *secretário da reunião* pode adicionar texto a constar na ata para determinado assunto;
- Durante a reunião, todos os assuntos a aprovar devem ser mostrados separadamente (em *links* gerados para o efeito);
- Para votar, o utilizador tem de estar autenticado. Juntamente com o voto, um utilizador pode ainda inserir um comentário ou uma intenção de voto (caso o voto seja contra). O campo do comentário é diferente do da intenção de voto (sendo que este último só deve aparecer em caso de voto contra);
- No final de cada votação, deve ser produzido o texto a colocar na ata, referindo o assunto, a breve descrição do assunto, o texto adicionado pelo *secretário da reunião* durante a reunião e se este foi aprovado ou não. Caso haja votos contra e declarações de voto, estas devem ser incluídas nesse texto também, para serem incluídas em anexo;
- (opcional) As autenticações são feitas através do cartão do cidadão.

Apesar dos objetivos deste projeto enunciarem 2 peças de software, poder-se-ão fazer ajustes ao longo do semestre de acordo com o progresso possível. Se alguma peça tiver de ficar de fora do desenvolvimento por falta de tempo, que seja a de votação em assuntos e produção de texto para ata. O sistema deve ser *web based*. Um dos requisitos gerais para o sistema é que, sendo software baseado na *web*, todas as comunicações devem ser seguras e corretas através do uso forçado do *Transport Layer Security* (TLS). Como a segurança é um fator determinante, o código deve seguir um princípio de minimização. Devem utilizar-se o menor número possível de tecnologias (sugere-se Apache, SQLite3, HTML5, CSS3 e PHP).

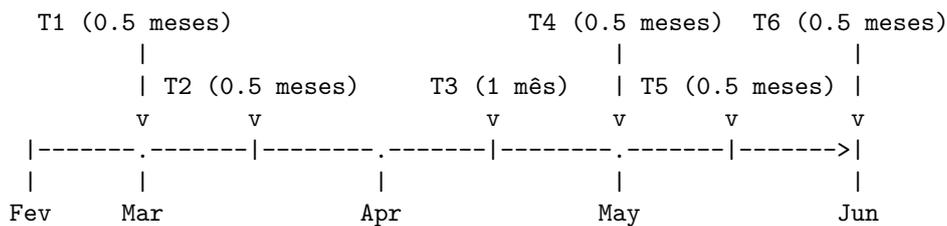
O projeto inclui todo o processo de engenharia de software, começando pelo levantamento detalhado dos requisitos, passando pelo planeamento, implementação e testes. Inclui também a elaboração da documentação e arquitetura do sistema, bem como a definição do(s) modelo(s) de dados de suporte. A documentação deve ser rica e uniforme e devem ser usados repositórios de código para manutenção e gestão de versões (e.g., github). Deve apontar-se para fluxos de utilização muito intuitivos e inclusão de mensagens simples sobre como funcionam perspectivas do utilizador. Sendo o sistema *web based*, este deve ser totalmente adaptável a diferentes tamanhos de ecrã.

Dada a sua natureza, este projeto requer conhecimentos sólidos em Programação, Segurança Informática, Engenharia de Software e Arquitetura de Computadores, bem como em Sistemas Operativos. O(a) aluno(a) terá uma oportunidade de solidificar o seu conhecimento nas várias áreas abrangidas por este projeto.

## Tarefas a Realizar e Cronologia

- T1** Contextualização com os objetivos propostos e preparação do ambiente de trabalho (0,5 meses);
- T2** Análise detalhada dos requisitos para cada uma das peças de software, investigação e definição dos limites do sistema, bem como da sua arquitetura (0,5 meses);

- T3** Implementação da primeira peça de software (sugere-se começar pela peça de votação eletrónica para eleições) (1 mês);
- T4** Implementação da segunda peça de software (peça de software para votação eletrónica de assuntos) (0,5 meses);
- T5** Testes e aprimoramento (0,5 meses);
- T6** Escrita do relatório de projeto (0,5 meses) [1].



## Requisitos Técnicos / Académicos

Ter boas classificações e conhecimentos em em Programação, Segurança Informática, Engenharia de Software e Arquitetura de Computadores e em Sistemas Operativos.

## Elementos de Avaliação a Entregar

Para além do relatório, o(a) aluno(a) deverá entregar um protótipo totalmente funcional do sistema desenvolvido, uma imagem do respetivo sistema operativo, todo o código fonte desenvolvido, bem como *scripts* ou instruções detalhadas de instalação e execução.

## Resultados Esperados

- \* O sistema operativo configurado, vários *scripts* e programas que cristalizem o sistema pretendida;
- \* A arquitetura do sistema e definição do(s) modelo(s) de dados;
- \* Um protótipo do sistema totalmente funcional;
- \* 1 relatório de projeto [1].

## Referências Bibliográficas

[1] C. Collberg and S. Kobourov, Self-plagiarism in Computer Science, Communications of the ACM, 48(4): 88 - 94, 2005.