

# Desenvolvimento e implementação em contexto realista do sistema CLASSQUIZ

Relatório preliminar de dissertação de mestrado

João Moreira

*Universidade de Aveiro*

27 de Fevereiro de 2019

**Resumo** - Este documento procura caracterizar as tarefas inerentes ao desenvolvimento e implementação, de um sistema remoto de realização de inquéritos/questionários de escolha múltipla em ambiente de sala de aula. O principal objectivo deste relatório preliminar é definir o plano imediato, e de médio prazo, de trabalho, com vista à discussão do mesmo na reunião com o orientador.

## Introdução

Os sistemas de avaliação pedagógica modernos preconizam um envolvimento contínuo dos estudantes. Nessa linha, uma solução em certas unidades curriculares será a de promover momentos de participação/avaliação de todos os alunos presentes nas aulas ou sessões de formação. Contudo, em aulas com um número elevado de alunos, a realização de pequenos momentos de avaliação pode-se revelar um desafio. O sistema aqui proposto, designado CLASSQUIZ, procura permitir a realização de inquéritos/questionários com resultados imediatos e automáticos. O sistema é baseado em três componentes:

- Terminais individuais, portáteis e sem fios, onde cada aluno presente na sala de aula poderá responder a questões postas globalmente, usando um sistema de escolha múltipla;
- Uma estação-base receptora, presente na sala, que valida, autentica e estabelece as trocas de comunicações com os terminais individuais;

- Um *software* dedicado, a correr num computador ligado ao sistema base, que mantém e gere uma base de dados de alunos, as suas respostas, bem como os tempos de pergunta/resposta.

A aplicação desenvolvida deverá correr num *browser*, de forma a ser agnóstico ao sistema operativo, e deve ser capaz de aceder a bases de dados. Esta aplicação terá ainda as funções de gerir as questões (texto/imagem) e de as colocar, por projecção, numa tela, bem como as temporizações e toda a dinâmica de registo de respostas na base de dados.

Para além da aplicação, será necessário desenvolver uma solução final para o terminal individual. Isto inclui uma reformulação da parte electrónica do mesmo, bem como o projecto e construção de uma estrutura de suporte e protecção dos elementos eléctricos do terminal.

Nas secções que se seguem, procura-se explicar, de uma forma resumida, em que consistem as tarefas definidas para a realização deste projecto, bem como algumas das actuais ideias e soluções que poderão ser adoptadas para as atingir.

## Trabalhos relacionados

Nos últimos anos, têm vindo a ser realizados trabalhos que visam contribuir para o desenvolvimento do sistema CLASSQUIZ. Em 2017, António Teixeira deu início ao desenvolvimento deste sistema, tendo procurado desenvolver um protótipo do mesmo. À data, o sistema era composto por um terminal individual, constituído por um microcontrolador Arduino Uno e um leitor de cartões RFID Mifare SL301, e uma estação-base receptora, constituída por um microcontrolador Arduino Mega 2560, tendo optado por fazer a comunicação *wireless* por meio de um módulo de Radiofrequência.

Em 2018, Manuel Mamede pegou no trabalho já desenvolvido e, iterando sobre o mesmo, criou um novo protótipo do sistema. Este novo protótipo era baseado na mesma estrutura do anterior, isto é, era composto por um terminal individual, uma estação-base receptora, e um software dedicado a correr na estação-base. Apesar de ser baseado no trabalho anterior, estes dois protótipos eram em tudo diferentes. Este novo sistema, designado CLASSQUIZ v1, era composto por um terminal individual, constituído por um microcontrolador NodeMCU v3 (LoLin) e um leitor de cartões RFID Mifare RFC522. O *software* da estação-base desta nova versão do sistema foi desenvolvido em ambiente VisualBasic, e a comunicação entre o terminal individual e a estação-base era feita por via WiFi, seguindo o protocolo UDP.

Já em 2019, no âmbito da unidade curricular Projecto em Automação e Robótica Industrial, eu próprio dei continuidade ao projecto. Fazendo uso do terminal construído por Manuel, desenvolvi uma aplicação em ambiente GTK, que permitia ao utilizador da estação-base (no caso, um computador) criar e projectar/exibir questionários de escolha múltipla. Durante a realização desse trabalho, alterei o protocolo de comunicação de UDP para TCP/IP, tendo sido essa a única alteração efectuada ao trabalho desenvolvido por Manuel, uma vez que o restante trabalho desenvolvido procurava complementar o já existente.

Infelizmente, os trabalhos referidos carecem de uma documentação mais aprofundada sobre outros sistemas similares que existam no mercado. Assim, antes de me aprofundar no desenvolvimento do sistema, será necessário proceder ao estudo do “Estado da arte”, procurando inspiração em estudos de sistemas parecidos ao que se pretende desenvolver, como, por exemplo, sistemas de votação electrónica para eleições. Estes sistemas podem ser particularmente interessantes, devido ao elevado grau de segurança necessária à implementação dos mesmos.

## Tarefa A

A primeira tarefa deste trabalho consiste em actualizar o *firmware* do dispositivo existente, de forma a que a resposta aos inquéritos apenas seja permitida enquanto o cartão de estudante permanecer no dispositivo. Isto implica uma validação em contínuo da presença em sala de aula.

Devido ao elevado número de dispositivos, será necessário que o seu endereço IP seja atribuído, de forma automática, pelo *router* ao qual se liga. Isto é conseguido através do protocolo DHCP, e pode ser definido através da página de configurações do *router*.

Por fim, de forma a que a identificação dos cartões de estudante válidos não esteja codificada de forma rígida no terminal individual, é necessário criar uma base de dados, que visa simular a base de dados na Universidade, onde estão contidas as UID's de todos os cartões de estudante, bem como a identificação do seu proprietário. Desta forma, a validação do cartão de estudante será feita por comparação com as entradas da base de dados já referida.

Durante a realização das restantes tarefas deste trabalho, será sempre necessário fazer algumas alterações ao *firmware* do terminal, de forma a incorporar no mesmo as novas funcionalidades desenvolvidas.

## Tarefa B

A etapa seguinte é a criação de uma base de dados, responsável pela gestão da informação e das respostas. Esta tarefa será realizada, em grande parte, a par com a tarefa seguinte, devido à conexão existente entre ambas.

Para cada questão, será criada uma nova entrada na tabela de questões. Os alunos, ao responderem às questões através dos seus terminais individuais, estarão a escrever nesta tabela as suas respostas. A cada entrada, ficará associada a UID (*Unique ID*) do cartão de estudante, a data e hora do registo da entrada, bem como a resposta dada pelo aluno.

Um dos principais sistemas de gestão de bancos de dados utilizados actualmente é o MySQL, sendo que a sua popularidade pode vir a ser útil na resolução de problemas que possam surgir durante a sua implementação. Ainda assim, será equacionada a utilização de sistemas alternativos, como o PostgreSQL ou o Firebase (desenvolvido

pela Google), de forma a averiguar se existem vantagens significativas na utilização de um em relação ao outro.

Ainda no que diz respeito às bases de dados, será também necessária a criação de contas de utilizador (para os professores), de forma a permitir que estes guardem aqui os seus inquéritos criados.

## Tarefa C

A terceira tarefa desta dissertação passa pelo desenvolvimento do corpo principal da aplicação na estação-base. Uma vez que a aplicação deverá correr no *browser*, podemos descrever esta etapa como sendo o desenvolvimento do *back-end* da aplicação, que irá fazer a ligação entre uma interface gráfica e a base de dados, e que permite a criação e armazenamento de inquéritos na mesma.

Existem várias tecnologias que visam facilitar o desenvolvimento desta interface entre a aplicação e a base de dados. Algumas das linguagens mais utilizadas para este fim são PHP, Python, e Ruby, mas a lista não se fica por aqui. Começando pela linguagem PHP, esta é talvez a mais utilizada no lado do servidor, mas que tem vindo a perder popularidade com o surgimento de novas *frameworks* nos últimos anos, tais como a Ruby on Rails ou a Django. Esta última é desenvolvida em Python, e tem visto a sua popularidade aumentar consideravelmente nos últimos anos. Como os próprios autores afirmam, Django é “a *framework* da *web* para perfeccionistas com prazos”. Não é a única desenvolvida em Python (uma alternativa interessante poderia ser a Flask) mas é talvez a mais indicada para quem não tem grande experiência sobre o assunto, uma vez que tem uma maneira própria e pré-definida de fazer as coisas, servindo assim de guia ao desenvolvedor. Neste momento, a intenção de escolha de uma *framework* para o *back-end* passa pela utilização da Django pelos seguintes factores: é desenvolvida em Python, que não é uma linguagem completamente nova para mim; possui uma grande comunidade e vasta documentação, que é extremamente importante para o esclarecimento de dúvidas que surjam durante o trabalho; o tutorial disponível na documentação oficial ensina a criar uma aplicação precisamente para a criação de questionários, enquadrando-se directamente no âmbito desta dissertação.

Por último, a Ruby on Rails, que é uma *framework* desenvolvida em Ruby, também se apresenta como uma alternativa interessante, mas neste momento não parece apresentar vantagens significativas em relação à Django.

## Tarefa D

A quarta tarefa é a concepção e implementação, de uma interface gráfica para a aplicação da estação-base, através da qual o utilizador da estação base pode interagir com a aplicação. Por outras palavras, esta tarefa corresponde ao *front-end* da aplicação.

Uma das linguagens de programação mais utilizadas para a criação de interfaces gráficas de aplicações *web* dá-se pelo nome de JavaScript. Existe uma enormidade de

*frameworks*, desenvolvidas com recurso a esta linguagem, que visam facilitar a vida ao desenvolvedor. Algumas das mais utilizadas actualmente são: AngularJS, Vue.js e React. Uma das vantagens da utilização de *frameworks* populares é a facilidade de encontrar soluções, tanto na documentação das mesmas, como nas suas respectivas comunidades. Contudo, para a criação de uma página *web* atraente, não basta utilizar uma linguagem de programação. Regra geral, são utilizadas linguagens de marcação (HTML) e de folhas de estilo (CSS), em conjunto com JavaScript, na criação das mesmas. Assim, *frameworks* como a Bootstrap, que fazem uso das três ferramentas (HTML, CSS e JavaScript) para permitir o desenvolvimento de componentes de interface e *front-end* para aplicações *web*, podem ser também soluções atraentes. Uma das vantagens da Bootstrap é a sua filosofia *mobile first*, garantindo uma apresentação e funcionalidade, optimizadas para dispositivos móveis. Esta característica poder-se-á revelar bastante importante no desenvolvimento da aplicação, uma vez que são cada vez mais os professores a trocar o computador pelo *tablet*.

A interface gráfica desenvolvida em ambiente GTK, para a unidade curricular Projecto em Automação e Robótica Industrial, poderá servir de referência ao desenvolvimento da aplicação *web*, tanto no seu *layout* como nas funcionalidades disponíveis.

A aplicação a desenvolver deverá permitir ao utilizador a criação de questionários, ou a importação dos mesmo a partir de um ficheiro. Cada questionário será composto por uma pergunta, um número fixo de respostas, bem como a sua duração. Deverá ainda permitir incorporar imagens, de forma opcional. Estes dados serão então guardados numa bases da dados, permitindo ao utilizador realizar o inquérito guardado a partir de outro dispositivo.

## Tarefa E

Uma vez desenvolvida a aplicação, podemos-nos virar finalmente para a parte física do problema. A posição desta tarefa sofreu alterações em relação ao plano de trabalhos proposto inicialmente, onde surgia logo após a actualização do *firmware* do terminal individual. Devido à sua complexidade, considero ser melhor guardar esta etapa para a segunda metade do projecto. A ideia é que, ao chegar a esta fase do trabalho, já possua uma ideia consolidada de como o terminal individual deve operar, e de como deve ser construído.

Podemos dividir esta tarefa em duas partes: uma parte electrónica e uma parte mecânica. No que diz respeito à parte electrónica desta tarefa, será necessário desenvolver uma nova placa PCB, de forma a tentar reduzir as dimensões do dispositivo. Será também necessário abordar o problema da alimentação, cuja abordagem levanta algumas questões. Serão utilizadas pilhas/baterias recarregáveis ou descartáveis? Como será convertida a tensão da fonte de alimentação para os 3.3V a que o microcontrolador opera? Existirá um botão ON/OFF no dispositivo? Todas estas questões irão influenciar a solução construtiva para a caixa de acomodação e protecção do dispositivo.

Ainda no que respeita à parte electrónica do terminal individual, será estudada a possibilidade de trocar de microcontrolador. O NodeMCU v2 (Amica) parece ser idêntico

ao utilizado actualmente, mas tem dimensões mais reduzidas. Contudo, será necessário averiguar se possui limitações significativas para a função que vai desempenhar. Nesta fase, será também tomada uma decisão final sobre outros elementos do dispositivo, como o número de botões presentes e o modo de funcionamento dos LED's. O modo como os LED's operam é extremamente importante, pois será, possivelmente, o único *feedback* que o aluno terá em assuntos como a validação/rejeição do seu cartão de estudante, recepção da resposta seleccionada por parte do servidor, ou até se o terminal individual está devidamente conectado ao mesmo.

A par com a parte electrónica está a parte mecânica do terminal. Será necessário construir uma caixa para acomodar e proteger o dispositivo, tendo especial atenção à dimensão final do mesmo, uma vez que se pretende que as dimensões finais sejam semelhantes à de um Raspberry Pi. Durante a reformulação da estrutura mecânica do dispositivo, será ainda necessário reformular o suporte para os botões e LED's, bem como projectar e desenvolver um sistema de guiamento e fixação do cartão de estudante, durante a operação do terminal.

Será também importante definir, atempadamente, qual será o processo de fabrico da caixa (impressão 3D, injeção de plástico, *etc.*), pois este poderá condicionar a solução final.

## Tarefa F

Esta tarefa consiste na construção de cinco terminais idênticos, para que se possam realizar os testes de escalabilidade. A construção dos terminais pode demorar algum tempo, uma vez que será, possivelmente, necessário encomendar peças e elementos de lojas *online*, algumas situadas fora do espaço comunitário. Assim, será necessário iniciar a etapa de construção dos terminais com tempo suficiente para salvaguardar eventuais demoras na recepção, bem como permitir a correcção de eventuais problemas que possam surgir na construção dos mesmos.

## Tarefa G

Esta é a etapa que precede a escrita da dissertação e consiste em testar profundamente o sistema criado. Será nesta fase que serão realizados os testes de escalabilidade do sistema, onde serão vistas questões como, por exemplo, a capacidade de gestão de tráfego da estação-base. Serão ainda convidados professores e alunos a utilizar o sistema, de forma a recolher *feedback* sobre melhorias futuras, bem como detectar e corrigir problemas que não tenham sido detectados até à data.

## Plano de trabalhos imediato, e de médio prazo

Uma vez que a primeira tarefa já se encontra praticamente concluída, o plano de trabalhos, no imediato, passa por criar uma base de dados, cuja função é a de simular a base de dados da Universidade. Esta base de dados será bastante simples e irá conter informações sobre a identificação dos alunos, bem como dos seus cartões de estudante. Será ainda necessário actualizar o *firmware* do terminal individual, para que a autenticação dos cartões de estudante seja feita por comparação com a base de dados criada.

De seguida, proceder-se-á à revisão bibliográfica, tanto dos trabalhos já realizados no âmbito desta dissertação, como de outros da mesma natureza. Com a informação recolhida, serão tomadas decisões sobre as *frameworks* a utilizar nas várias fases de desenvolvimento da aplicação da estação-base. Será ainda iniciada a escrita do “Estado da arte”. Uma vez concluída a escrita do mesmo, será dado início ao desenvolvimento da aplicação.

De forma a criar uma base de dados eficiente, que será um pilares da aplicação, é necessária a realização de tutoriais e, possivelmente, cursos sobre a criação e gestão de bases de dados. Esta revisão servirá para consolidar os conhecimentos adquiridos sobre o assunto em outras unidades curriculares. Serão ainda realizados tutoriais/cursos sobre a *framework* adoptada para a criação do *back-end* da aplicação. Desta forma, a criação da base de dados será feita tendo já uma ideia de como será utilizada pela aplicação da estação-base.

A expectativa é que a parte funcional da aplicação se encontre completamente desenvolvida até meados/fim de Abril, altura em que a evolução do projecto será reavaliada. Será nesta altura que serão tomadas decisões relativamente ao futuro do projecto, ou seja, se valerá a pena despender muito tempo na criação de uma interface gráfica mais atraente, ou se é preferível avançar para as questões relacionadas com o *hardware* do sistema.

## Calendário de execução

A figura seguinte procura exemplificar o que é, à data de escrita deste documento, o calendário de previsão de execução de cada tarefa. É importante referir que no mesmo, não estão discriminadas as etapas de revisão bibliográfica, de realização de cursos/tutoriais sobre as ferramentas a utilizar, nem da escrita da dissertação.

	Fevereiro				Março				Abril				Maio				Junho			
Tarefa A																				
Tarefa B																				
Tarefa C																				
Tarefa D																				
Tarefa E																				
Tarefa F																				
Tarefa G																				

Figura 1: Calendário de execução das tarefas previstas.

## Conclusões

No final do presente semestre lectivo, pretende-se ter desenvolvido, com sucesso, um sistema de inquérito e avaliação para ambiente de sala de aula, que seja passível de ser utilizado por todo o tipo de utilizadores, desde os mais calejados na utilização de tecnologias, aos mais inexperientes. Pretende-se ainda que o sistema desenvolvido seja robusto, dada a natureza da sua utilização, e que tenha um elevado grau de segurança, quer no acesso e transmissão de dados, devido à natureza privada e pessoal de alguns dos dados utilizados, quer na componente de armazenamento de respostas e avaliação dos alunos, de forma a evitar possíveis fraudes, ou mesmo erros que resultem do próprio sistema.

Desta forma, ainda que não se encontre num estado comercialmente viável, o objectivo é que, no final desta dissertação, o sistema CLASSQUIZ se encontre pronto a ser testado em grande escala. Ou seja, pretende-se que o sistema possa ser utilizado recorrentemente, já no próximo ano lectivo, para a realização de inquéritos/questionários, ainda que os mesmos não venham a ser utilizados para fins de avaliação.

## Referências

- Mamede, Manuel (2018). *ClassQuiz Plus - Sistema de Inquérito e Avaliação Rápida em Sala de Aula*.
- Teixeira, António (2017). *Class Quiz*.
- DB-Engines - Knowledge Base of Relational and NoSQL Database Management Systems*. URL: <https://db-engines.com/en/> (acedido em 17/02/2019).
- Django documentation*. URL: <https://docs.djangoproject.com/en/2.1/> (acedido em 13/02/2019).
- HotFrameworks - Web framework rankings*. URL: <https://hotframeworks.com/> (acedido em 17/02/2019).