

Aplicação Quiz – Programação de aplicação ao lado do Cliente

Relatório do trabalho

Elaborado por:

Ângelo Dias – 24288

Martim Dias - 24290

Docentes:

Luís Garcia

Carlos Sancho

Índice

1.	Resumo.....	iv
2.	Introdução	7
3.	Revisão da aplicação desktop	8
3.1.	Tema do trabalho	8
3.2.	Trabalho em equipa	8
3.3.	Desenvolvimento do projeto	9
3.4.	Funcionalidades desenvolvidas	9
3.4.1.	Aplicação baseada em contas	10
3.4.2.	Diversidade de temas	10
3.4.3.	Menu do administrador	10
3.4.4.	TextArt com temporizadores.....	10
3.5.	Dificuldades sentidas	10
4.	Organização do trabalho e ferramentas usadas na aplicação android	12
5.	Análise do problema	13
5.1.	Análise de sistemas semelhantes	13
5.1.1.	Perguntados 2.....	14
5.1.2.	Super Quiz	16
5.2.	Caracterização dos Utilizadores	17
5.2.1.	Persona 1	18
5.2.2.	Persona 2	18
5.3.	Cenários de Utilização.....	18
5.3.1.	Cenário 1	18
5.3.2.	Cenário 2	19
6.	Funcionalidades da aplicação	20
7.	Desenho da interface da aplicação.....	21
7.1.	Interface da aplicação.....	21
7.1.1.	Menu Terms	21
7.1.2.	Menu Login.....	22
7.1.3.	Menu Register	22
7.1.4.	Menu Home.....	23
7.1.5.	Menu Profile	23

7.1.6.	Menu Scores	24
7.1.7.	Menu Shop	24
7.1.8.	Menu Settings.....	25
7.1.9.	Menu pergunta	26
7.1.10.	Menu Final Score	26
8.	Desenho da base de dados	27
9.	Conceção da base de dados.....	28
9.1.	Demonstração da base de dados e exemplos de <i>queries</i>	28
9.1.1.	Demonstração da base de dados	28
9.1.2.	Exemplos de <i>queries</i>	28
9.2.	Tabelas desenvolvidas	30
9.2.1.	Tabela User.....	30
9.2.2.	Tabela Theme	30
9.2.3.	Tabela Questions	31
9.2.4.	Tabela Score	31
9.2.5.	Tabela Shop	31
9.2.6.	Tabela UserCurrency	32
9.2.7.	Tabela Achievements	32
9.2.8.	Tabela AchievementsUser.....	32
10.	Programação da lógica da aplicação	34
10.1.	Lógica do método para voltar á pergunta anterior	34
10.2.	Lógica para verificar conquistas	35
10.3.	Lógica para verificar correspondência dos dados do login.....	35
11.	Testes com a aplicação.....	37
11.1.	Primeiro teste.....	37
11.1.1.	Conclusão do teste 1	38
11.2.	Segundo teste	39
11.2.1.	Conclusão do teste 2	40
12.	Página web da aplicação e screencasts	41
12.1.	Demonstração do website	41
12.2.	Screencast do website.....	44
12.3.	Screencast do aplicativo	44
13.	Conclusão.....	45

1. Resumo

O projeto desenvolvido possui como objetivo o desenvolvimento de uma aplicação de quiz, que oferece entretenimento e possibilidades de testar os conhecimentos dos utilizadores do mesmo, com perguntas desafiadoras e vários temas diferentes. A app conta com moedas chamadas “pootis coins”, que são recompensadas a responder testes quiz com sucesso, e que possibilitam aos utilizadores na aquisição de *merch* na vida real, como por exemplo T-shirts.

O aplicativo foi desenvolvido na linguagem de programação JAVA com recurso ao IDE Android Studio, que foi a linguagem e o programa lecionados, respetivamente. Adicionalmente, foi desenvolvido um website com fins de publicitar este aplicativo, através das linguagens HTML, CSS e Javascript lecionadas na disciplina de Tecnologias Web.

Palavras-chave: Quiz, Testes, Moedas, Java, Android Studio, Website

Abstract

The project developed aims to develop a quiz application, which offers entertainment and possibilities to test our users' knowledge, with challenging questions and several different topics. The app has coins called “pootis coins”, which are rewarded for successfully answering quiz tests, and which enable users to purchase merch in real life, such as T-shirts.

The application was developed in the JAVA programming language using the Android Studio IDE, which was the language and program taught, respectively. Additionally, a website was developed with the purpose of advertising this application, using the HTML, CSS and Javascript languages taught in the Web Technologies discipline.

Keywords: Quiz, Tests, Coins, Java, Android Studio, Website

Agradecimentos

Como um grupo, gostaríamos de agradecer aos docentes Carlos Sancho e Luís Garcia, pelos conhecimentos transmitidos e pela atenção ao esclarecimento de dúvidas que surgiram em forma de obstáculos para o desenvolvimento deste projeto. Também oferecemos os nossos agradecimentos ao docente Luís Rosário por nos transmitir o conhecimento necessário para desenvolver o website necessário. Adicionalmente, também exprimimos agradecimentos ao colega João Silva, por alguma assistência oferecida em algumas etapas do desenvolvimento do aplicativo.

2. Introdução

O objetivo do desenvolvimento deste projeto e relatório é melhorar as competências dos alunos responsáveis para desenvolver aplicações num ambiente móvel e melhorar a organização de relatórios futuros. O objetivo do aplicativo desenvolvido é a criação de um aplicativo quiz com finalidade de os utilizadores do mesmo poderem ter um certo entretenimento a resolver os testes quiz disponíveis para poderem testar os seus conhecimentos sobre um tema existente á sua escolha, ou até para obter *merch* oficial do jogo. Um teste quiz é composto por 7 perguntas escolhidas de forma aleatória sobre o tema escolhido, em que podem ter 4 dificuldades: fácil, intermédia, difícil e extrema.

A aplicação foi desenvolvida em IDE Android Studio com recurso á linguagem de programação Java, e para um ambiente web foi adicionalmente utilizado as linguagens HTML, CSS E JavaScript, que tiveram como objetivo o desenvolvimento um website para publicitar a aplicação.

3. Revisão da aplicação desktop

A primeira etapa desta aplicação foi revisar e analisar a aplicação desktop desenvolvida previamente através da aplicação BlueJ no semestre passado.

3.1. Tema do trabalho

O tema do trabalho não foi alterado, uma vez que também foi uma aplicação de quiz, com o mesmo objetivo da desenvolvida em Android Studio, dar entretenimento e possibilitar aos utilizadores testarem os seus conhecimentos de uma forma divertida. Depois de uma avaliação intensa ao aplicativo desktop, foi-se decidido fazer algumas adições para melhorar a jogabilidade e melhorar rigorosamente a estética do aplicativo (já que era tudo em consola sem embelezamento).

Para isso foi decidido como melhoria para o projeto em ambiente móvel decidiu-se adicionar um sistema de moedas para comprar *merch* temática na vida real, adicionar mais temas e mais perguntas para dar uma maior possibilidade de testes, foi adicionado um perfil mais bem formulado, foi mantida a opção do histórico mas foi embelezada de acordo com o restante do aplicativo e foram criadas definições para melhorar a experiência dos utilizadores, para apagar os dados que o mesmo desejar (todas estas adições para o desenvolvimento android).

3.2. Trabalho em equipa

No desenvolvimento da aplicação passada, foi tomada em consideração a eficiência que o grupo organizou cada uma nas necessidades que cada membro deveria realizar, havendo assim uma dinâmica muito efetiva, que foi alcançada novamente no ambiente android.

A atribuição de tarefas foi realizada efetivamente devido ao aplicativo “discord”, que foi importante para assistir numa comunicação clara entre os membros do grupo, tanto para a organização das tarefas, comunicação de problemas ou dúvidas e envio de arquivos.

Adicionalmente, o “discord” não teve um uso muito grande sem ser o envio de arquivos, uma vez que os membros do grupo moram juntos, pelo que comunicaram diretamente sobre os problemas e organização das tarefas, sendo este o fator que melhor ajudou na efetividade do grupo.

3.3. Desenvolvimento do projeto

Como foi previamente referido, o desenvolvimento do aplicativo foi bastante efetivo. O desenvolvimento do projeto ficou dividido na seguinte forma:

- 1- Desenho manual da navegação – Primeiramente, foi realizado o desenho do protótipo dos menus e da navegação do aplicativo a desenvolver, com as opções que cada menu iria possuir. Também foi pensado nesta etapa as funcionalidades que o aplicativo iria possuir, como ver as pontuações passadas e as opções e administrador.
- 2- Desenho da base de dados – De seguida foi esquematizado todos os elementos que a base de dados desenvolvida em SQL iria possuir, para garantir que não existiria erros críticos na inserção futura dos dados e para ter melhor noção das funcionalidades a adicionar.
- 3- Criação dos elementos da base de dados – Depois de esquematizada, foi então criada a base de dados concretamente em SQL através do aplicativo XAMPP, com todos os elementos esquematizados.
- 4- Implementação da lógica do aplicativo – Com a base de dados feita, foi então criada toda a lógica da navegação e das funcionalidades de todo o aplicativo, tanto do menu do utilizador e o menu exclusivo do administrador.
- 5- Criação e implementação das perguntas – Sendo um aplicativo quiz, obviamente foi necessário criar várias perguntas e inserir as mesmas divididas pelos diferentes temas criados antes da criação e da pesquisa das perguntas inseridas.
- 6- Testes dos membros do grupo e com utilizadores – Para finalizar, foram feitos testes para descobrir qualquer bug existente e ainda foram alguns feitos com utilizadores, para obtermos feedback no que melhorar no futuro. Algumas sugestões dadas pelos utilizadores foram implementadas na aplicação android.

3.4. Funcionalidades desenvolvidas

Este aplicativo quiz possui várias funcionalidades que de certa forma permite o mesmo ser distinguido de outros projetos semelhantes, sendo que todas estas funcionalidades têm como objetivo tornar a aplicação mais simples e com uma navegação clara para todos os possíveis utilizadores, tentando oferecer a melhor experiência possível.

3.4.1. Aplicação baseada em contas

Este aplicativo possui um sistema de contas exclusivo do administrador. Quando a app é executada, se o utilizador escolher a opção “administrador” e colocar o nome de utilizador e a senha do mesmo poderá aceder às várias opções que o mesmo possui, como editar perguntas e consultar pontuações.

3.4.2. Diversidade de temas

O aplicativo disponibiliza diversidade de perguntas, pelo que possui 3 temas: tecnologia, espaço e aleatório. Assim oferece alguma diversidade do que o utilizador poderá querer jogar.

3.4.3. Menu do administrador

No login do aplicativo, se forem inseridos os dados da conta do administrador, irá abrir as opções do administrador. Aqui o mesmo pode observar o histórico de todos os utilizadores e pode editar, apagar ou inserir perguntas.

3.4.4. TextArt com temporizadores

Existem secções do aplicativo em que o sistema utiliza TextArt com símbolos ASCII como uma tentativa de embelezar o mesmo, e adicionalmente utiliza timers nos mesmos para dar um sentimento de animação na execução do aplicativo.

3.5. Dificuldades sentidas

No desenvolvimento do aplicativo, em regra geral não houveram muitos obstáculos, embora que existissem 2 prominentes:

- 1- Respostas aleatórias – A maior dificuldade foi quando é iniciado um teste quiz, fazer com que as respostas da pergunta não aparecessem no mesmo sítio sempre que essa

pergunta for jogada, pelo que seria um pouco repetitivo demais. Embora que tenha existido por um bocadinho, este problema foi ultrapassado com sucesso.

- 2- Perguntas aleatórias – Embora mais fácil que o anterior, também ofereceu alguma dificuldade. Isto possuía como objetivo as perguntas serem sorteadas e não mostradas seguindo o seu ID, o que depois de concluídas iria as tornar também muito repetidas. Este problema também foi ultrapassado.

4. Organização do trabalho e ferramentas usadas na aplicação android

Na realização da aplicação android tentou-se sempre possuir uma organização efetiva eficiente, o que se pode concluir que foi eficiente como no projeto desenvolvido em desktop.

A comunicação manteve-se semelhante à aplicação de desktop, utilizando a aplicação “discord” mais para o envio de arquivos e para algumas dúvidas à distância, mas mais uma vez como os membros do grupo moram juntos, a comunicação foi bastante direta e efetiva.

De seguida estão enumeradas as ferramentas utilizadas:

Discord – Como já foi referido, este software foi utilizado para envio de arquivos e para a comunicação à distância.

Github – Esta ferramenta foi fundamental para permitir uma boa organização dos ficheiros do projeto e ajudou no controlo de versões diferentes do projeto.

Microsoft Word – Esta ferramenta somente foi utilizada para o desenvolvimento do presente relatório.

Illustrator – O Illustrator foi utilizado no esboço de um protótipo de baixa-fidelidade para o grupo observar o que deveria ser adicionado na aplicação em desenvolvimento. Também foi utilizado para o desenvolvimento de alguns ícones do aplicativo.

Figma – Esta ferramenta foi usada para fazer um protótipo de alta-fidelidade com toda a navegação que o aplicativo iria possuir. Serviu de grande ajuda para observar o funcionamento efetivo dos caminhos da navegação do aplicativo, e para garantir que a navegação é simples.

OBS Studio – Esta ferramenta somente serviu para a gravação do screencast.

Dbdiagram.io – Esta ferramenta foi para o desenho do modelo conceptual da base de dados.

5. Análise do problema

Depois de uma revisão rigorosa da aplicação desenvolvida num ambiente desktop, foi feita uma análise do problema, para identificar problemas ou obstáculos que poderiam vir a surgir no desenvolvimento do aplicativo. De seguida serão enumeradas as funcionalidades que se pretendeu adicionar após a análise:

- 1- Sistema de moedas – Esta funcionalidade não foi possível implementar no projeto passado, pelo que o grupo possui este sistema de moedas como uma das suas prioridades, com a finalidade de oferecer uma certa recompensa ao utilizador por jogar. Com essas moedas o utilizador poderá adquirir *merch* oficial do aplicativo.
- 2- Aplicação baseada em contas – Também uma das prioridades principais foi possuir um aplicativo com um sistema de contas, em que o utilizador terá de se registar para poder usufruir das funcionalidades do aplicativo. Depois do registro, a conta ficará guardada na base de dados e o utilizador poderá fazer login com a mesma.
- 3- Sistema de conquistas – Semelhantemente ao sistema de moedas, esta funcionalidade serve para oferecer um certo sentimento recompensador ao utilizador por concluir algum dos objetivos que o mesmo possui no aplicativo.

Estas funcionalidades foram a nossa maior prioridade no desenvolvimento do aplicativo, para poder dar diversidade á experiência do aplicativo.

5.1. Análise de sistemas semelhantes

Foram analisados 2 aplicativos diferentes sobre o mesmo tema, o “quiz”. Estes aplicativos foram adicionalmente avaliados sobre os seus aspetos positivos e negativos com base dos princípios de usabilidade já nos lecionados. Através de uma pesquisa mais eficiente na internet, foram-se obtidas 2 aplicações excelentes para análise.

5.1.1. Perguntados 2

Esta app de quiz foca-se mais no aspeto de jogador contra jogador, onde o que responder corretamente a mais perguntas em menos tempo ganha, existindo 6 temas diferentes: História, Desporto, Arte, Geografia, Ciência e Cinema. Adicionalmente é bastante atraente para os utilizadores, uma vez que possui vários aspetos inovadores, como um quiz com uma loja de moedas e até as mascotes da aplicação possuem uma certa consciência e falam com os utilizadores.

Aspetos positivos:

Familiaridade – Embora possua 2 ícones diferentes do normal, são na mesma facilmente reconhecidos, sendo eles o “Roleta” e “Equipes”, pelo que irão abrir as diferentes skins das roletas e o outro as equipes/clãs. Todos os outros ícones da aplicação (por exemplo loja, versus, definições, início...) são facilmente reconhecidos ou organizados bem para facilitar a sua leitura.



Figura 1 - Elementos que aplicam familiaridade no aplicativo

Capacidade de resposta – Nas operações mais demoradas da aplicação, é mostrada uma barra de carregamento que indica o progresso da operação com algumas dicas sobre a jogabilidade.



Figura 2 - Barra de carregamento

Aspetos negativos:

Personalização – Esta aplicação embora possua aspetos bastante atraentes visualmente, o seu menu inicial é bastante confuso, uma vez que possui mais de 15 opções diferentes, em que pode tornar-se complicado fazer a leitura de todas eficientemente. Com isso, a aplicação não oferece nenhuma alternativa para organizar estas opções confusas, o que se torna uma violação clara a este princípio de usabilidade.

Múltiplas tarefas – Esta aplicação também não possibilita em circunstância alguma fazer mais de uma tarefa ao mesmo tempo, pelo que até em circunstâncias como a roleta estar a girar ou numa pergunta o utilizador não pode pausar o jogo e todas as opções ficam indisponíveis mesmo que apareçam como disponíveis, como o botão de voltar atrás, sendo assim uma clara violação ao princípio das múltiplas tarefas.

5.1.2. Super Quiz

Embora o nome, esta app é muito semelhante ao programa “Quem quer ser milionário?”, tendo a paleta de cores, as ajudas, o desenho, as recompensas e até o esquema de recompensas e de perguntas muito semelhante.

Por ser muito genérica, e já existirem muitas semelhantes, julgamos que esta app poderia ter melhorias, como adicionar um modo multijogador fácil, inovações no sistema de perguntas para se afastar do esquema repetitivo e ainda talvez dar outro tipo de recompensas, já que as moedas só servem para comprar ajudas. Por isso, não há grande vontade de jogar o jogo por muito tempo, sendo a aplicação muito repetitiva.

Aspetos positivos:

Familiaridade – Com a sua interface simples, todas as opções são facilmente compreendidas, incluindo os ícones da mesma, que todos indicam com facilidade aos utilizadores as suas funções.



Figura 3 - Exemplos de familiaridade na interface

3ª regra de Shneiderman (oferecer feedback informativo) – Depois de se responder a uma pergunta incorretamente, irá ser gerado um texto através de uma A.I. que irá explicar porquê a resposta correta da pergunta é de facto a certa, oferecendo grande feedback informativo ao utilizador.

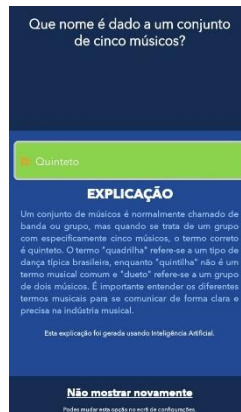


Figura 4 - Exemplo de uma resposta gerada automaticamente

Aspetos negativos:

Personalização – Semelhante á aplicação passada, esta também não possui qualquer possibilidade de personalizar a nossa experiência, sendo uma clara violação a este princípio.

Capacidade de resposta – Este aplicativo até em alturas de operações mais demoradas nunca é dado ao utilizador alguma informação sobre o progresso da operação onde somente é mostrada um ecrã a dar carregar.

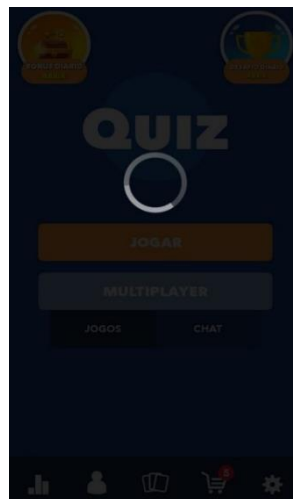


Figura 5 - Má capacidade de resposta

5.2. Caracterização dos Utilizadores

Foram criadas duas personas com fins de facilitar a compreensão de que tarefas e rotas de escolha podem ser escolhidas no aplicativo. As personas foram escolhidas com atenção ao público-alvo mais provável para este aplicativo, com cenários prováveis para cada uma das mesmas, com o objetivo de tornar estas personas o mais próximas de uma interação real.

5.2.1. Persona 1

O Jorge Amaral tem 10 anos, é de Óbidos, está no 5º ano escolar com as melhores notas da turma dele e acabou de chegar a casa da escola e para passar o tempo foi para o telemóvel da mãe jogar “Brawl Stars”. Depois de perder vários jogos, ele fartou-se do jogo e para se animar e acalmar foi jogar “Pootis Quis”, e aqui irá ser iniciada a sua interação com o aplicativo.

5.2.2. Persona 2

A Joaquina Almeida tem 23 anos e é formada em Gestão de Empresas. Trabalha no escritório de contabilidade da empresa Tabela Simétrica localizado no Barreiro, que é a mesma localidade dela. Na hora de almoço do seu trabalho depois de já ter comido a sandes que tinha feito, decidiu ir ao telemóvel testar o aplicativo “Pootis Quiz” para esperar os restantes 2 minutos que sobra na sua hora de almoço, pelo que será o início da sua interação com o aplicativo.

5.3. Cenários de Utilização

Foi desenvolvido 1 cenário para cada persona para caracterizar uma possível interação do mesmo com o sistema.

5.3.1. Cenário 1

Quando o Jorge Almeida iniciou o quiz, ele primeiramente aceitou os termos de uso preguiça e de seguida passou pela criação de uma conta através do registo na aplicação, inserindo o seu nome de utilizador, email e a palavra-passe. Depois de registado o Jorge foi tentar dar login para aceder ao quiz mas acabou por errar a senha que tinha acabado de criar. Depois de o sistema o avisar que a senha estava incorreta, ele finalmente a colocou corretamente e fez o login com sucesso.

Ele vê que existem duas opções, singleplayer e multiplayer, pelo que ele como é competitivo escolhe o multiplayer. Como o multiplayer está indisponível, ele então escolhe o singleplayer.

Depois de ser escolhido o tema “CS:GO”, o Asdrubalino irá ser redirecionado para o início do teste quiz. Ele tenta o seu melhor mas somente consegue acertar 3 perguntas, pelo que ele se irrita e fecha o aplicativo, concluindo a interação.

5.3.2. Cenário 2

A Joaquina abre a aplicação e nem lê os termos de uso, sendo rápida a aceitá-los, uma vez que não possui muito tempo. Ela é rápida a inserir a informação para se registrar, e quando se regista e vai fazer o login, ao contrário do Asdrubalino ela não erra e inicia a sessão com sucesso.

Pela falta de tempo que a mesma tem, ela decide que irá somente navegar no aplicativo. Ela navega primeiramente para o menu do perfil onde observa as conquistas, depois vai para o menu das pontuações e vê que não existem porque ela ainda não jogou, depois para a loja e fica intrigada com a t-shirt que poderá comprar no futuro com moedas e por fim acede às definições. Depois de ver as definições, observa que não tem mais tempo e fecha o aplicativo.

6. Funcionalidades da aplicação

O aplicativo desenvolvido conta com várias funcionalidades que foram implementadas com sucesso para oferecer uma melhor experiência aos utilizadores. As mesmas irão ser enumeradas:

- 1- Aplicação com sistema de contas – Este aplicativo possui um sistema de contas, graças ao registo que possui para criar e guardar contas. Depois de criada, o utilizador irá precisar de fazer login no aplicativo com o email e palavra-passe que criou para jogar. O utilizador só poderá entrar nas opções do aplicativo depois de fazer o login. Ele poderá dar log-out a qualquer momento da sua conta, e se não o fizer, a próxima vez que abrir o aplicativo já estará com a sessão iniciada na sua conta.
- 2- Diversidade de temas – O aplicativo disponibiliza uma grande diversidade de modos de jogo, pelo que irá possuir um modo multijogador e outro para um só jogador, em que o utilizador poderá escolher um dos temas disponíveis para jogar.
- 3- Conquistas – O utilizador, á medida que completa testes quiz, poderá receber uma das várias conquistas presentes do aplicativo, seja completar 10 testes quiz, fazer todas as perguntas de um tema ou até completar 5 testes quiz sem nenhuma resposta errada. O utilizador recebe logo uma conquista ao criar a conta.
- 4- Histórico – Depois de completar algum teste ou testes quiz, o utilizador poderá consultar o histórico de testes quiz que ele já jogou, em que é mostrado o tema jogado, a data que foi jogado e a pontuação que o utilizador obteve no teste.
- 5- Sistema de moedas – Sempre que o utilizador completa algum teste quiz, irá receber moedas baseado na sua performance. Com essas moedas, o mesmo irá ter a opção de comprar *merch* oficial do jogo na vida real.
- 6- Limpeza de dados - Se o utilizador desejar, o mesmo poderá apagar os dados que tem no aplicativo, seja os testes já jogados, as conquistas, as estatísticas e até toda a informação do mesmo, com exceção á exclusão da sua conta.

Todas estas funcionalidades mantêm o objetivo de deixar a aplicação com alguns aspetos diferentes das existentes do mercado, uma vez que existem várias aplicações de quiz de altos níveis que não possuem conquistas ou até limpeza de dados, por exemplo.

7. Desenho da interface da aplicação

Para um desenho inicial da aplicação, foi utilizada a ferramenta “Illustrator”, que serviu de grande ajuda no desenho do primeiro protótipo do aplicativo e para o desenho de ícones para a mesma. Adicionalmente, num protótipo melhorado foi utilizado o “Figma”, um website que permite criar protótipos de alta-fidelidade com todos os caminhos entre ecrãs, pelo que foram tomados em conta os princípios de usabilidade lecionados previamente. Ambas as ferramentas foram de grande ajuda para o desenho da interface da aplicação e foram lecionadas na disciplina de Design Gráfico.

7.1. Interface da aplicação

De seguida irão ser mostradas os esboços relevantes realizados tanto no protótipo de alta fidelidade tanto os esboços finais. A primeira imagem é o desenho do protótipo e a segunda é o desenho na aplicação concluída:

7.1.1. Menu Terms

Este primeiro menu ficou relativamente semelhante, com o objetivo de dar a informar ao utilizador da informação que o mesmo deve aceitar para iniciar o aplicativo. Somente foi alterado o esquema de cores.

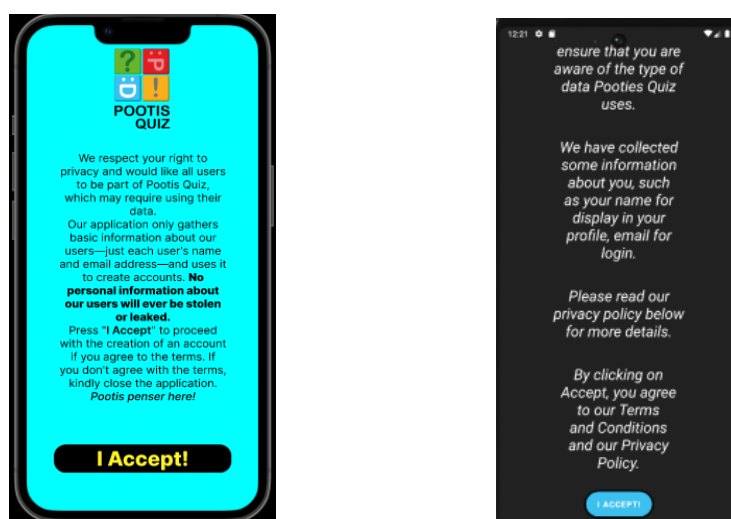


Figura 6 - Menu termos de uso

7.1.2. Menu Login

Este menu recebeu poucas alterações, sendo somente colocada uma borda amarela e alteradas as cores de alguns elementos para dar uma certa personalidade á estética do login. De resto, ficou bastante semelhante ao protótipo desenvolvido em figma. Serve para dar login no aplicativo depois de inseridos os valores de uma conta criada.

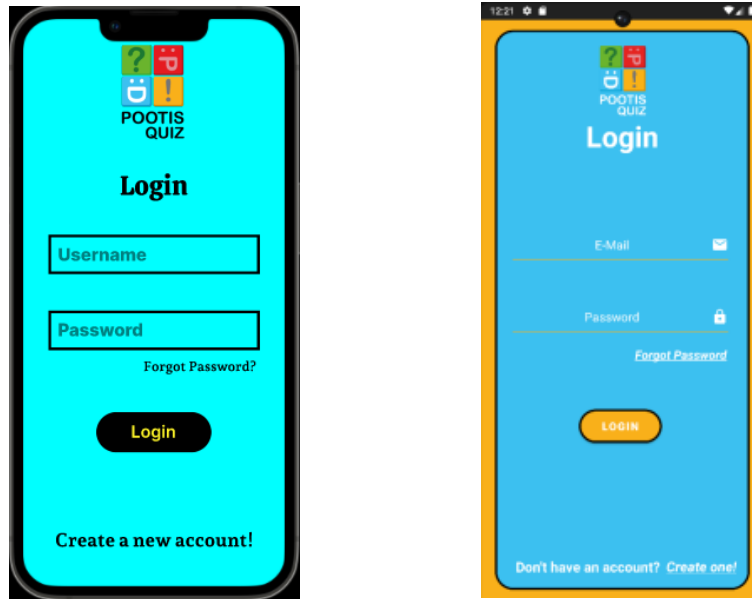


Figura 7 - Ecrã de login

7.1.3. Menu Register

Semelhante ao menu de login, o menu de register somente obteve a mesma borda amarela e algumas alterações de cores, ficando semelhante ao protótipo. Serve para criar a conta do utilizador para depois fazer o login.

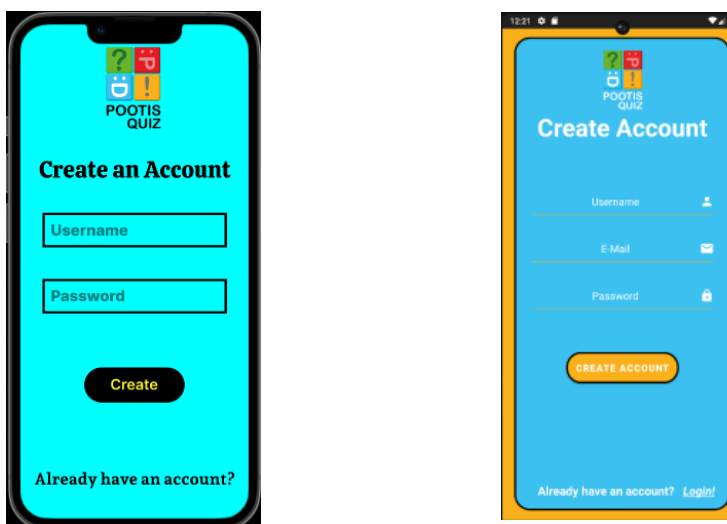


Figura 8 - Ecrã de registo

7.1.4. Menu Home

Este menu recebeu uma mudança relativamente significativa. As cores foram quase todas alteradas, foi removido o botão de “play” porque basta clicar no tema e mudou-se a posição do nome de utilizador. A estética dos temas também foi alterada drasticamente. O mesmo serve para escolher os temas e jogar o quiz.

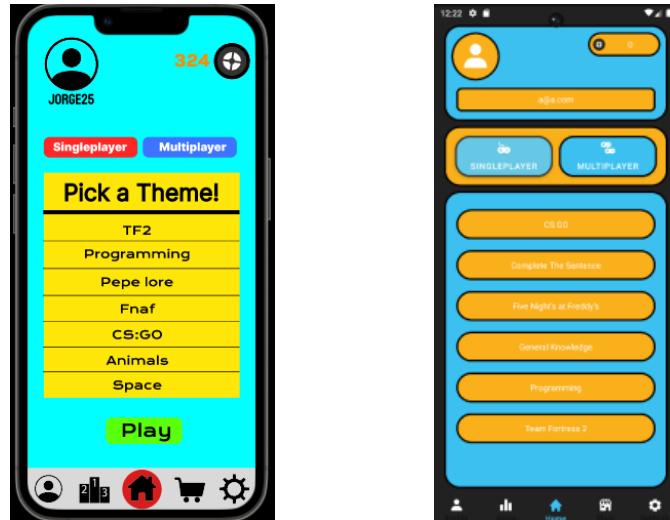


Figura 9 - Ecrã principal

7.1.5. Menu Profile

Este menu recebeu algumas mudanças também, sendo removida a imagem de fundo do utilizador e a frase do mesmo, pelo que ambas estas mudanças foram feitas porque o ecrã ficaria muito “poluído”. Também foram alteradas as cores de fundo e as das conquistas.

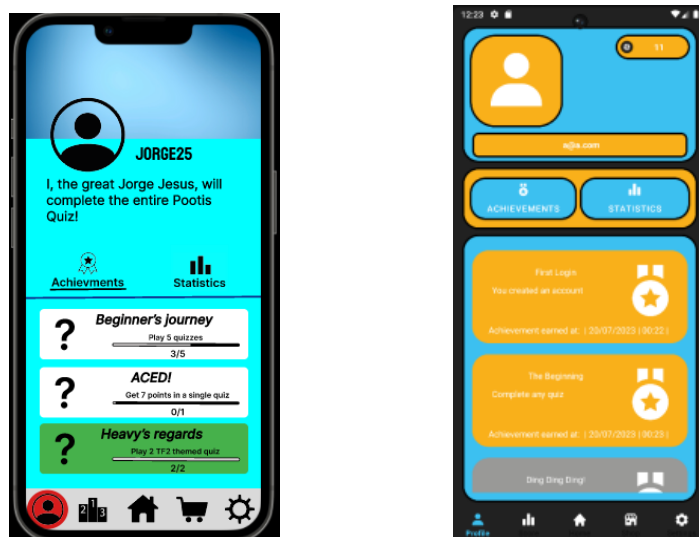


Figura 10 - Ecrã perfil

7.1.6. Menu Scores

Este menu foi bastante embelezado, pelo como se pode observar nas imagens. As pontuações já não estão em forma de tabela, e sim em forma de cadeia, pelo que dá uma estética mais positiva ao aplicativo.



Figura 11 - Ecrã pontuações

7.1.7. Menu Shop

Neste menu foi removida a opção para obter “pootis coins”, pois o grupo acredita que micro transações neste aplicativo não seriam a melhor ideia. Neste menu agora o utilizador poderá ver o ícone do mesmo e nome juntamente com as moedas no menu, e poderá continuar a ver as opções de compra e as suas moedas.

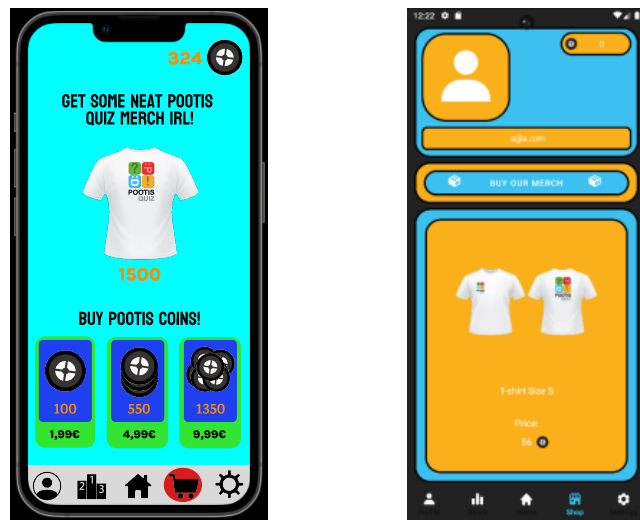


Figura 13 - Ecrã loja

7.1.8. Menu Settings

Este menu de longe foi o mais alterado. Ele foi repensado na sua totalidade, pois algumas opções foram discutidas em meio do desenvolvimento e foi se decidido que deveriam ser removidas (por exemplo, um teste quiz com imagens diferentes por pergunta seria adiões “desnecessárias”, uma vez que embora melhore a estética não contribui para o melhor funcionamento do aplicativo, que foi uma das prioridades. Do protótipo desenvolvido, só a opção de logout foi reaplicada num novo local, sendo ele alterado para opções de apagar as informações que o utilizador deixou no aplicativo, como apagar as conquistas, as pontuações ou toda a informação com exceção á sua conta.

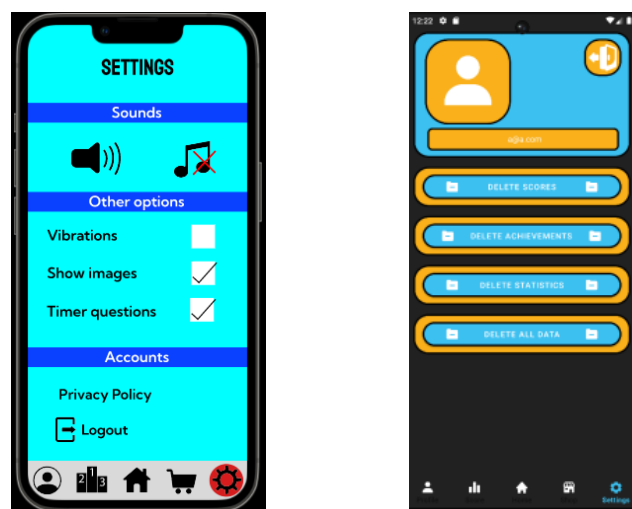


Figura 14 - Ecrã definições

7.1.9. Menu pergunta

Este menu também recebeu alterações significantes, pelo que o temporizador foi removido, tal como o tema que está a ser jogado, por ser uma ideia descartada e uma informação com pouco utilidade, respetivamente. Somente ficará aplicado o ícone do aplicativo em cima da pergunta e as respostas ficarão todas dentro de um retângulo, sendo simplificado bastante do desenho final comparado com o do protótipo desenvolvido.

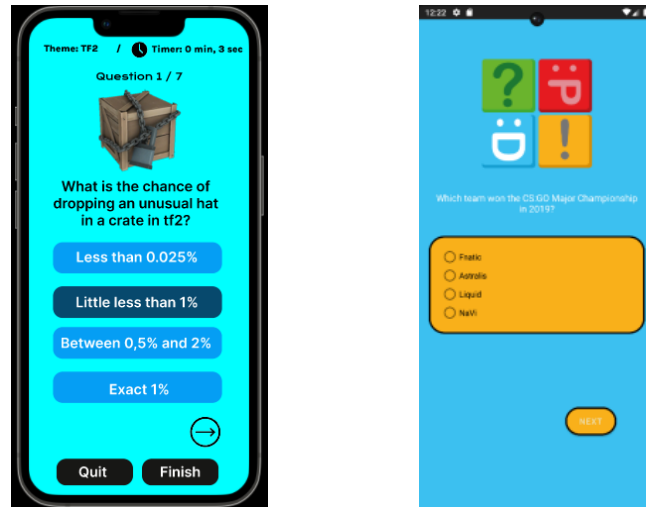


Figura 15 - Ecrã pergunta

7.1.10. Menu Final Score

Este menu recebeu poucas alterações significantes, sendo que a imagem de parabéns foi alterada pelo ícone do aplicativo e o tempo do teste foi removido pela razão já previamente demonstrada.

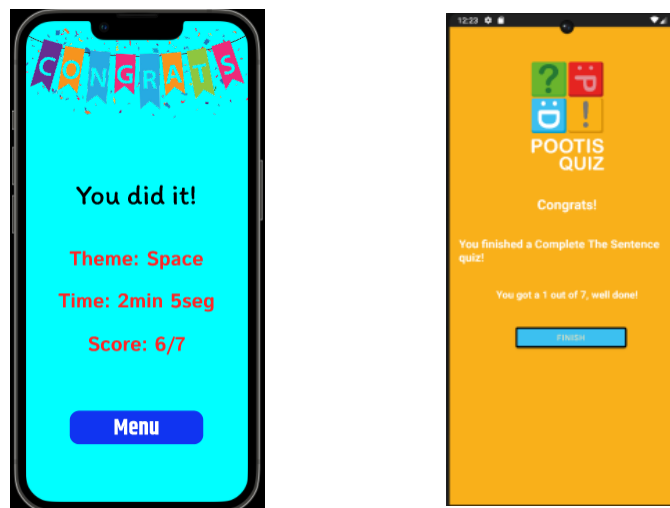


Figura 16 - Ecrã pontuação final

8. Desenho da base de dados

A base de dados obviamente teve uma grande utilidade, pois ficariam na mesma armazenados diversos elementos importantes, como as perguntas, as respostas e as conquistas, por exemplo. A mesma primeiramente foi desenhada em papel, com todos as tabelas que o grupo julgou que a mesma iria ter de possuir para assim obter para proporcionar um bom funcionamento da aplicação. De seguida irá ser mostrado o desenho da base de dados desenvolvido na ferramenta dbdiagram.io, com todas as ligações necessárias:

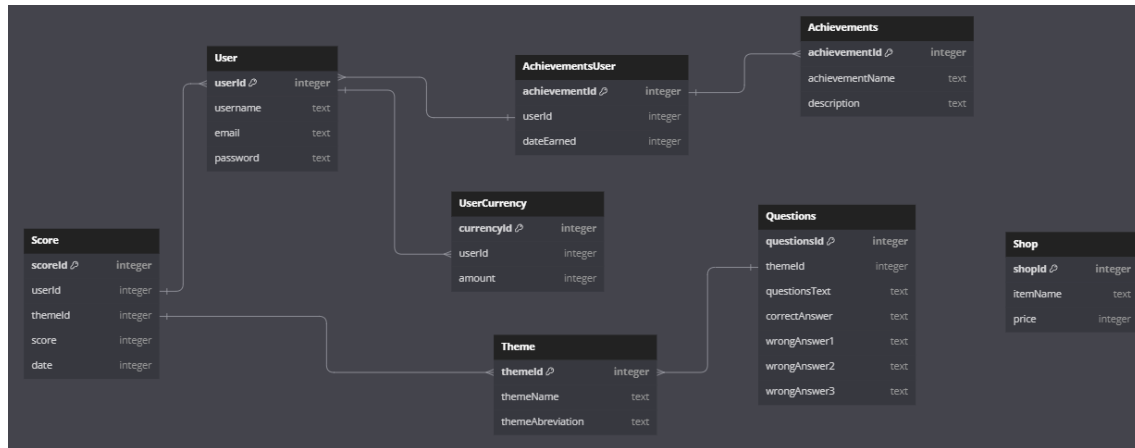


Figura 17 - Modelo conceitual da base de dados

9. Conceção da base de dados

Para realizar a conceção da base de dados, primeiramente foi necessário criar todas as tabelas que a mesma iria necessitar para o seu funcionamento efetivo.

Como já foi referido, primeiramente foi necessário fazer o desenho primitivo da base de dados em papel, para mais tarde ajudar na implementação da mesma no Android Studio, para continuar com o desenvolvimento do aplicativo. A base de dados utilizada foi a biblioteca “Room”, que cria as tabelas no formato SQL Lite.

9.1. Demonstração da base de dados e exemplos de *queries*

9.1.1. Demonstração da base de dados

De seguida será mostrada como está formada a base de dados desenvolvida para esta aplicação. No tópico 9.2. serão explicadas cada objetivo das tabelas como também os seus elementos.

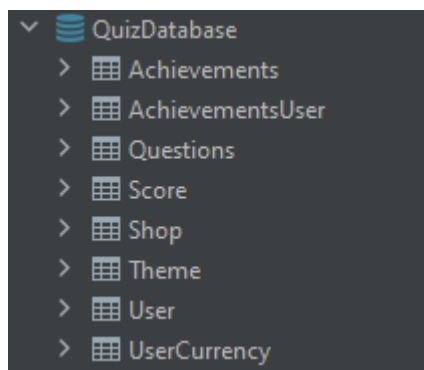


Figura 18 - Demonstração das tabelas desenvolvidas

Como se pode observar, a base de dados possui 8 tabelas, cada uma com os seus elementos necessários para desempenharem a sua função efetivamente no aplicativo.

9.1.2. Exemplos de *queries*

Para adicionar, editar ou apagar informação da base de dados “Room” foi necessária a criação de várias *queries* para fazerem a tarefa proposta. Serão agora apresentadas algumas das mais relevantes para o funcionamento efetivo da base de dados.

Primeiramente serão apresentadas algumas *queries* relevantes para fazer consultas:

QUERY que recupera todos os campos da tabela User onde o email e a senha correspondem aos valores especificados:

```
@Query("SELECT * FROM User WHERE email = :email AND password = :password")
```

QUERY que recupera todos os campos da tabela Score onde o ID do user e o ID do tema correspondam aos valores fornecidos:

```
@Query("SELECT * FROM Score WHERE userId = :userId and themeId = :themeId")
```

QUERY que recebe o ID da pergunta anterior baseado no ID da pergunta atual, pelo que irá procurar o maior ID de pergunta que seja menor que o ID da pergunta atual, na tabela Questions:

```
@Query("SELECT QuestionsId FROM questions WHERE QuestionsId < :currentQuestionId ORDER BY QuestionsId DESC LIMIT 1")
```

QUERY que recupera os campos da tabela AchievementsUser onde o ID do user e o ID da conquista correspondem aos valores dados.

```
@Query("SELECT * FROM AchievementsUser WHERE userId = :userId and achievementId = :achievementId")
```

Agora serão mostradas as *queries* relevantes para inserir a informação mais importante para o bom funcionamento do quiz (entre parênteses estarão os locais onde os dados estão a ser inseridos, por ordem):

QUERY para inserir um tema na tabela Theme (themeID, themeName, themeAbreviation):

```
db.execSQL("INSERT INTO Theme VALUES (2,'Team Fortress 2','TF2')");
```

QUERY para inserir uma conquista na tabela Achievements (achievementsID, achievementsName, description):

```
db.execSQL("Insert into Achievements values (2,'The Beginning', 'Complete any quiz')");
```

QUERY para inserir uma peça de merch na tabela Shop (shopID, itemName, price):

```
db.execSQL("Insert into Shop values (1,'T-shirt Size S', 56)");
```

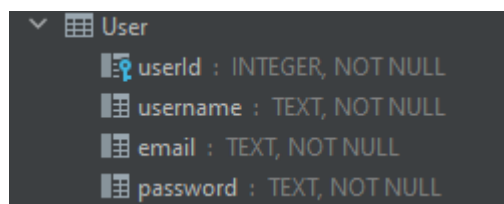
QUERY para inserir uma pergunta na tabela Questions (questionsText, themeId, correctAnswer, wrongAnswer1, wrongAnswer2, wrongAnswer3):

```
db.execSQL("INSERT INTO Questions (questionsText, themeId, correctAnswer, wrongAnswer1, wrongAnswer2, wrongAnswer3) VALUES " +  
"('What is the most played CS:GO map?', 4, 'Dust 2', 'Mirage', 'Inferno', 'Nuke')");
```

9.2. Tabelas desenvolvidas

9.2.1. Tabela User

A tabela User foi a primeira a ser desenvolvida, pela sua importância nas restantes tabelas por ser implementadas. Nela fica guardado o ID do utilizador automaticamente atribuído, o nome de utilizador, o email e a palavra-passe, pelo que estes 3 elementos são feitos pelo mesmo no registo do aplicativo.

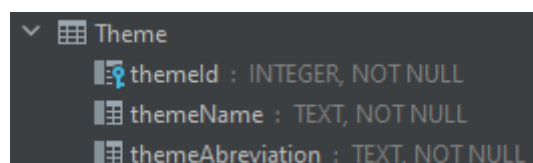


User	
primaryKey	userId : INTEGER, NOT NULL
	username : TEXT, NOT NULL
	email : TEXT, NOT NULL
	password : TEXT, NOT NULL

Figura 19 - Elementos da tabela User

9.2.2. Tabela Theme

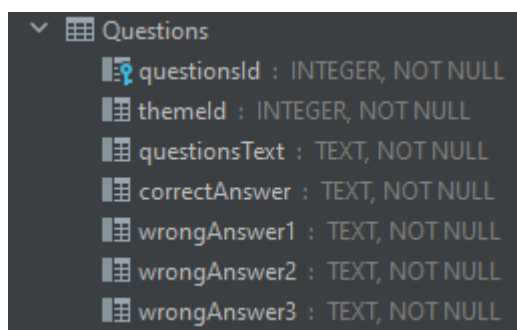
A tabela Theme foi a segunda a ser desenvolvida, para poder depois ser criada a tabela para armazenar as perguntas com os respetivos temas. Nela fica guardado o ID do tema que é atribuído, o nome do tema para mostrar no menu inicial e quando um quiz é completado e ainda um elemento com uma abreviação do tema, para aparecer nas pontuações passadas de forma abreviada, para o design ter uma melhor utilização de espaço.



Theme	
primaryKey	themeld : INTEGER, NOT NULL
	themeName : TEXT, NOT NULL
	themeAbreviation : TEXT, NOT NULL

9.2.3. Tabela Questions

Para o desenvolvimento desta tabela foi necessária a criação da tabela Themes primeiro, pois esta teria de ser chave estrangeira para a pergunta ter uma ligação com o tema correspondente. Nesta tabela existe o ID da pergunta, o id do tema que a pergunta pertence, que é a chave estrangeira, existe o texto da pergunta e de seguida exista a resposta correta e as 3 restantes incorretas, já que as perguntas são de escolha múltipla.



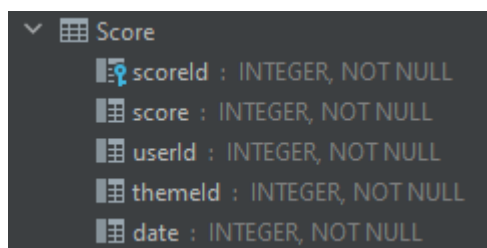
A screenshot of a database tool showing the structure of the 'Questions' table. The table has the following columns: 'questionsId' (INTEGER, NOT NULL, primary key), 'themeld' (INTEGER, NOT NULL, foreign key), 'questionsText' (TEXT, NOT NULL), 'correctAnswer' (TEXT, NOT NULL), 'wrongAnswer1' (TEXT, NOT NULL), 'wrongAnswer2' (TEXT, NOT NULL), and 'wrongAnswer3' (TEXT, NOT NULL).

Column Name	Data Type	Constraints
questionsId	INTEGER	NOT NULL, Primary Key
themeld	INTEGER	NOT NULL, Foreign Key
questionsText	TEXT	NOT NULL
correctAnswer	TEXT	NOT NULL
wrongAnswer1	TEXT	NOT NULL
wrongAnswer2	TEXT	NOT NULL
wrongAnswer3	TEXT	NOT NULL

Figura 21 - Elementos da tabela Questions

9.2.4. Tabela Score

Para esta tabela funcionar da forma desejada, foi necessário fazer ligação com a tabela do User para obter o ID do mesmo e com a tabela Theme para obter o ID também, para conseguir mostrar o histórico somente desse utilizador e o tema que o mesmo respondeu. Também precisa do ID da pontuação, da pontuação em questão e da data que o teste quiz foi jogado.



A screenshot of a database tool showing the structure of the 'Score' table. The table has the following columns: 'scoreId' (INTEGER, NOT NULL, primary key), 'score' (INTEGER, NOT NULL), 'userId' (INTEGER, NOT NULL, foreign key), 'themeld' (INTEGER, NOT NULL, foreign key), and 'date' (INTEGER, NOT NULL).

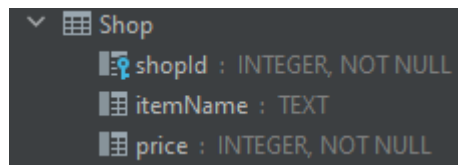
Column Name	Data Type	Constraints
scoreId	INTEGER	NOT NULL, Primary Key
score	INTEGER	NOT NULL
userId	INTEGER	NOT NULL, Foreign Key
themeld	INTEGER	NOT NULL, Foreign Key
date	INTEGER	NOT NULL

Figura 22 - Elementos da tabela Score

9.2.5. Tabela Shop

A tabela Shop somente possui o ID da loja, o nome do item e o preço do mesmo. Não é precisa mais informação pois o item da loja ainda não pode ser comprado para os utilizadores

usarem. O único objetivo desta tabela é mostrar os itens disponíveis como opção de compra futura.



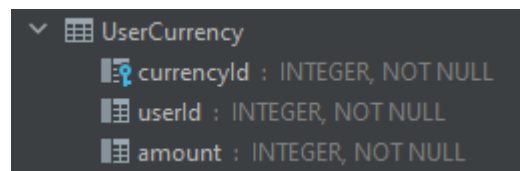
A screenshot of a database management tool showing the structure of the 'Shop' table. The table has three columns: 'shopId' (INTEGER, NOT NULL, primary key), 'itemName' (TEXT), and 'price' (INTEGER, NOT NULL).

shopId	itemName	price
--------	----------	-------

Figura 23 - Elementos da tabela Shop

9.2.6. Tabela UserCurrency

Esta tabela serve para mostrar a quantidade de moedas que o utilizador possui. Para isso foi criado o ID da currency (moeda), foi relacionado com o ID do utilizador para garantir que as moedas do mesmo estão corretas, e adicionalmente a quantidade de moedas que o mesmo possui.



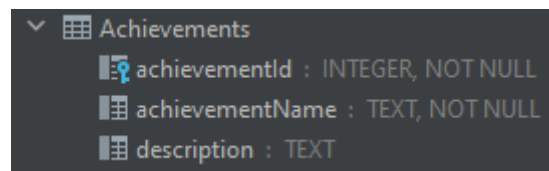
A screenshot of a database management tool showing the structure of the 'UserCurrency' table. The table has three columns: 'currencyId' (INTEGER, NOT NULL, primary key), 'userId' (INTEGER, NOT NULL), and 'amount' (INTEGER, NOT NULL).

currencyId	userId	amount
------------	--------	--------

Figura 24 - Elementos da tabela UserCurrency

9.2.7. Tabela Achievements

A tabela Achievements tem como objetivo mostrar as conquistas que o utilizador possui e as que ele ainda não concluiu. Esta tabela possui ID da conquista, o nome da mesma e a descrição de como ela é obtida.



A screenshot of a database management tool showing the structure of the 'Achievements' table. The table has three columns: 'achievementId' (INTEGER, NOT NULL, primary key), 'achievementName' (TEXT, NOT NULL), and 'description' (TEXT).

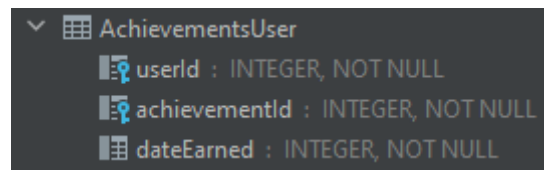
achievementId	achievementName	description
---------------	-----------------	-------------

Figura 25 - Elementos da tabela Achievements

9.2.8. Tabela AchievementsUser

A última tabela criada foi a AchievementsUser, que serve para garantir que cada utilizador tem as conquistas correspondentes somente ao seu id (á sua conta), e não a todas obtidas no aplicativo por todas as contas. Para isso foi usado como chave estrangeira o ID do utilizador,

foi usado também a chave estrangeira do ID da conquista e ainda a data que a conquista foi obtida.



A screenshot of a database schema viewer showing the structure of the 'AchievementsUser' table. The table name is at the top, followed by three columns: 'userId', 'achievementId', and 'dateEarned'. Each column is marked with a foreign key icon (a blue circle with a white 'f') and is defined as 'INTEGER, NOT NULL'.

Column Name	Data Type	Constraints
userId	INTEGER	NOT NULL, Foreign Key
achievementId	INTEGER	NOT NULL, Foreign Key
dateEarned	INTEGER	NOT NULL

Figura 26 - Elementos da tabela AchievementsUser

10. Programação da lógica da aplicação

Durante a totalidade do desenvolvimento da aplicação quiz, a linguagem utilizada foi sempre o JAVA, que foi previamente lecionada. Como a aplicação possui diversas funcionalidades muito relevantes para o seu funcionamento, alguma da lógica será explicada nesta secção, embora que se tente afastar de colocar o relatório demasiado “complexo” para os leitores. Serão apresentados os 3 maiores obstáculos no desenvolvimento da lógica da aplicação.

10.1. Lógica do método para voltar á pergunta anterior

Quando um teste quiz é iniciado, o utilizador poderá responder ás perguntas apresentadas. Se o mesmo julga que colocou a resposta errada, ele poderá voltar atrás para conferir, através deste código:

```
private void onButtonPreviousClicked() {
    int previousQuestionId = getPreviousQuestionId(questionId);
    if (previousQuestionId != -1) {
        saveSelectedAnswer();
        replaceWithQuestion(previousQuestionId);
    }
}

private int getPreviousQuestionId(int currentQuestionId) {
    int previousQuestionId = -1;
    int currentQuestionIndex = -1;

    for (int i = 0; i < questionsList.size(); i++) {
        Questions question = questionsList.get(i);
        if (question.getQuestionsId() == currentQuestionId) {
            currentQuestionIndex = i;
            break;
        }
    }

    if (currentQuestionIndex > 0) {
        Questions previousQuestion = questionsList.get(currentQuestionIndex - 1);
        previousQuestionId = previousQuestion.getQuestionsId();
    }

    return previousQuestionId;
}

private void saveSelectedAnswer() {
    int selectedRadioButtonId = radioGroupAnswers.getCheckedRadioButtonId();
    if (selectedRadioButtonId != -1) {
        RadioButton selectedRadioButton = getView().findViewById(selectedRadioButtonId);
        question.setSelectedAnswer(selectedRadioButton.getText().toString());
    }
}
```

Este código fará com que quando o utilizador clique no botão para voltar atrás para a pergunta anterior, o código vai obter o ID da pergunta anterior, salva a resposta já seleccionada

na pergunta atual e vai mostrar a pergunta anterior com o ID obtido, com a resposta que já lá estava. Isto oferece recuperabilidade ao utilizador, em caso de engano.

10.2. Lógica para verificar conquistas

Para verificar se o utilizador completou uma conquista exclusiva, chamada “Ding Ding Ding!”, que é obtida quando o utilizador não erra em nenhuma pergunta de o teste quiz, o seguinte código será executado:

```
// Check if the user already has the achievement "Ding Ding Ding!"
    if (hasPerfectScore(quizDatabase, userId, themeId)) {
        if (!hasAchievement(quizDatabase, userId, 3)) {
            quizDatabase.getAchievementUserDao().insertAchievementUser(new
            AchievementUser(userId, 3, System.currentTimeMillis()));
        }
    }
private boolean hasPerfectScore(QuizDatabase quizDatabase, int userId, int themeId) {
    return quizDatabase.getScoreDao().hasPerfectScore(userId, themeId);
}
@Query("SELECT COUNT(*) FROM Score WHERE userId = :userId AND themeId = :themeId AND score = 7")
boolean hasPerfectScore(int userId, int themeId);
```

Como se pode observar, este código serve para verificar se o utilizador teve um teste quiz sem nenhuma resposta errada, e se o mesmo possui um teste desses sem ainda possuir esta conquista, a mesma irá lhe ser atribuída.

10.3. Lógica para verificar correspondência dos dados do login

Para verificar que tanto o email e tanto a password inseridos pelo o utilizador correspondem, o seguinte bloco de código fará a verificação:

```
String hashedPassword = hashPassword(password);
User user = userDao.getUserByEmailAndPassword(email, hashedPassword);
if (user != null){
    userIdValue = user.getUserId();
    if(userIdValue > 0) {
        UserCurrency existingUserCurrency =
userCurrencyDao.getUserCurrencyById(userIdValue);
        if (existingUserCurrency == null) {
            UserCurrency newUserCurrency = new UserCurrency(userIdValue, 0);
            userCurrencyDao.insertCurrency(newUserCurrency);
```

```
        }
        AchievementUser existingUserAchievement =
achievementUserDao.getUserAchievementByUserId(userIdValue);
        if (existingUserAchievement == null) {
            AchievementUser newAchievementUser = new
AchievementUser(userIdValue, 1, currentTimeMillis);
            achievementUserDao.insertAchievementUser(newAchievementUser);
        }
    }
}
```

Este código irá verificar se tanto o email e a palavra-passe fornecidos existem na base de dados. Se existirem na base de dados, irá verificar se ele possui um registo de moedas e de conquistas associados ao utilizador. Se por vez não possuir, irá lhe criar novos registos na base de dados. Este código tem como objetivo fazer uma autenticação das informações do utilizador e dos recursos (moedas e conquistas) que pode ou pode ainda não possuir.

11. Testes com a aplicação

Para obter uma melhor noção de como o aplicativo poderia funcionar na visão do público, foi feita uma análise com utilizadores em que cada um realizou um teste com várias tarefas orientadas pelo grupo de trabalho, com o objetivo de avaliar a dificuldade que poderá surgir a navegar o aplicativo.

Como não foi possível estarmos com os utilizadores presencialmente, foi necessário usar um software de chamadas em tempo real chamado “discord”, e a interação foi capturada com um gravador de ecrã e áudio.

Devido a certos problemas de internet e gravação, a gravação dos áudios dos utilizadores e os vídeos demonstrativos das suas ações ficaram bastante insatisfatórios, uma vez que metade de ambos não foi captado, ou somente não se ouvia ou estaria tudo desfocado.

A pedido do docente, iremos mostrar os resultados obtidos noutra sessão com os utilizadores, embora sem vídeo demonstrativo

Aqui está disponibilizada a lista de tarefas que foi instruída aos utilizadores:

1. Faz o teu registo na aplicação (totalmente funcional)
2. Faz o login com a tua conta (totalmente funcional)
3. Visita o teu perfil de utilizador (totalmente funcional)
4. Consulta as estatísticas e as conquistas (só as conquistas estão funcionais)
5. Vai ver as pontuações passadas (totalmente funcional)
6. Vai á loja ver as opções disponíveis de compra (totalmente funcional)
7. Vai ás definições e apaga as tuas conquistas (totalmente funcional)
8. Vai ao menu principal e faz as etapas para iniciar um teste quiz sozinho
(parcialmente funcional, já que o multiplayer não está disponível ainda)
9. Para concluir, faz o logout da tua conta (totalmente funcional)

11.1. Primeiro teste

O primeiro utilizador foi o colega Rafael Picareta, que se localiza no grupo junto com o Pedro Sarramita e o Gonçalo Rosário, a desenvolver uma aplicação de self service.

Depois de oferecido o consentimento do mesmo, realizamos o teste com ele com as tarefas já referidas, e estes foram os resultados:

1. Faz o teu registo na aplicação
Clicks – 6 / Tempo – 15 segundos
2. Faz o login com a tua conta
Clicks – 3 / Tempo – 6 segundos
3. Visita o teu perfil de utilizador
Clicks – 1 / Tempo – 2 segundos
4. Consulta as estatísticas e as conquistas
Clicks – 2 / Tempo – 4 segundos
5. Vai ver as pontuações passadas
Clicks – 1 / Tempo – 2 segundos
6. Vai á loja ver as opções disponíveis de compra
Clicks – 2 / Tempo – 4 segundos
7. Vai ás definições e apaga as tuas conquistas
Clicks – 3 / Tempo – 6 segundos
8. Vai ao menu principal e faz as etapas para iniciar um teste quiz sozinho
Clicks – 3 / Tempo – 8 segundos
9. Para concluir, faz o logout da tua conta
Clicks – 2 / Tempo – 4 segundos

11.1.1. Conclusão do teste 1

Como se pode observar, o utilizador concluiu as suas tarefas sem nenhuma grande dificuldade, sendo que a tarefa que mais demorou foi o registo da sua conta. O seu trajeto foi o que já estava previsto pelo grupo e julgamos que este teste foi satisfatório, havendo somente uma dúvida na tarefa 6, que bastava só fazer um click, mas o utilizador enganou-se.

11.2. Segundo teste

O primeiro utilizador foi o colega Rafael Picareta, que se localiza no grupo junto com os colegas Pedro Sarramita e Gonçalo Rosário, a desenvolver uma aplicação de self-service.

Depois de oferecido o consentimento do mesmo, realizamos o teste com ele com as tarefas já referidas, e estes foram os resultados:

1. Faz o teu registo na aplicação

Clicks – 6 / Tempo – 12 segundos

2. Faz o login com a tua conta

Clicks – 3 / Tempo – 5 segundos

3. Visita o teu perfil de utilizador

Clicks – 1 / Tempo – 2 segundos

4. Consulta as estatísticas e as conquistas

Clicks – 2 / Tempo – 4 segundos

5. Vai ver as pontuações passadas

Clicks – 1 / Tempo – 2 segundos

6. Vai á loja ver as opções disponíveis de compra

Clicks – 1 / Tempo – 3 segundos

7. Vai às definições e apaga as tuas conquistas

Clicks – 3 / Tempo – 6 segundos

8. Vai ao menu principal e faz as etapas para iniciar um teste quiz sozinho

Clicks – 3 / Tempo – 7 segundos

9. Para concluir, faz o logout da tua conta

Clicks – 2 / Tempo – 5 segundos

11.2.1. Conclusão do teste 2

Como se pode observar, o utilizador concluiu as suas tarefas sem nenhuma grande dificuldade semelhante ao primeiro utilizador, e novamente a tarefa mais demorada foi o registo. O trajeto foi também o previsto, embora que este utilizador tenha sido mais eficaz do que o utilizador 1. Mais uma vez, os resultados foram bastante satisfatórios, e com ambos testes deram a entender que a navegação do aplicativo está clara, sendo esse o nosso objetivo.

12. Página web da aplicação e screencasts

Para o desenvolvimento de um site publicitário da aplicação a ser desenvolvida em android, foi utilizado a ferramenta “Visual Studio Code”, com recurso às linguagens “HTML” e “CSS”. Foi tentado fazer um website com uma estrutura e cor semelhante á do aplicativo, para os utilizadores que usarem o website para saber da existência do aplicativo não baixarem a app e verificarem que é totalmente diferente do website.

De seguida serão mostrados cada secção presente no website desenvolvido, que serve somente como um site publicitário:

12.1. Demonstração do website

Primeiramente, a primeira coisa que se observa quando se abre o site é a seguinte:

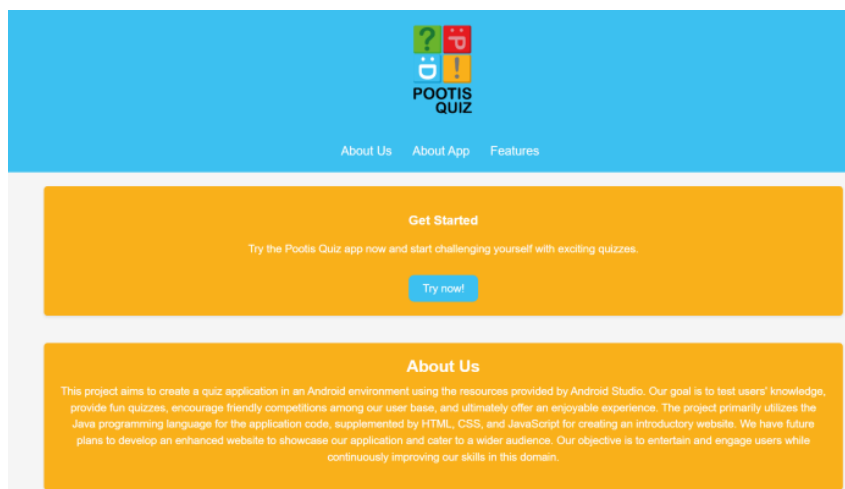


Figura 27 - 1ª parte do website (Get started/About us)

Nesta primeira parte do website o utilizador poderá na teoria premir o botão para baixar o aplicativo num futuro próximo. Também pode ler a secção “About us”, onde se revela o objetivo da criação do quiz e como foi feito. Adicionalmente, o utilizador poderá utilizar a barra de navegação para descer para os respetivos locais na página.

De seguida, é mostrada a segunda secção do website:

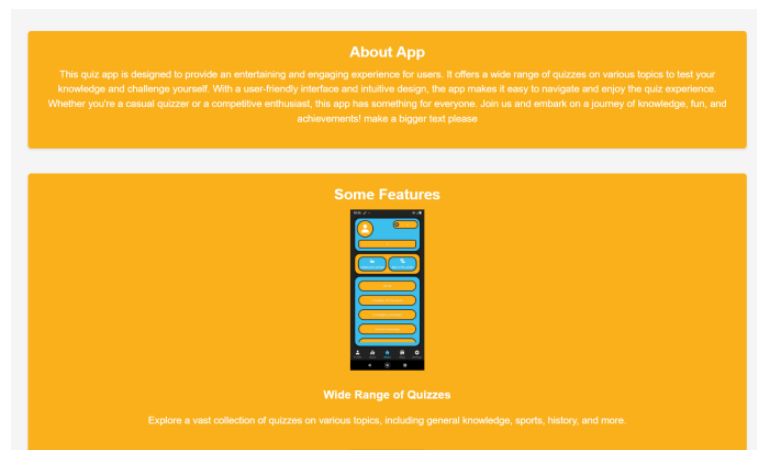


Figura 28 - 2ª parte do website (About App/Some Features)

Na segunda parte do website o utilizador pode verificar que possui informação sobre a app e o início da secção onde estão as *features* do aplicativo, onde é mostradas algumas imagens correspondentes às *features* apresentadas com o texto depois de cada uma.

Agora pode-se observar a 3ª parte do website, que é a continuação da secção das “Some Features”:

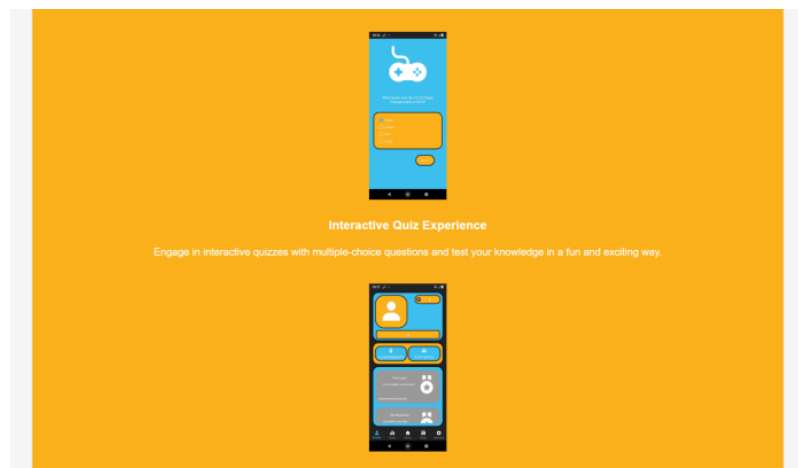


Figura 29 - 3ª parte do website (Some Features)

Como já referido, esta parte somente possui a continuação da secção passada, com as imagens de suporte também referidas.

De seguida pode-se observar a 4ª parte do website:

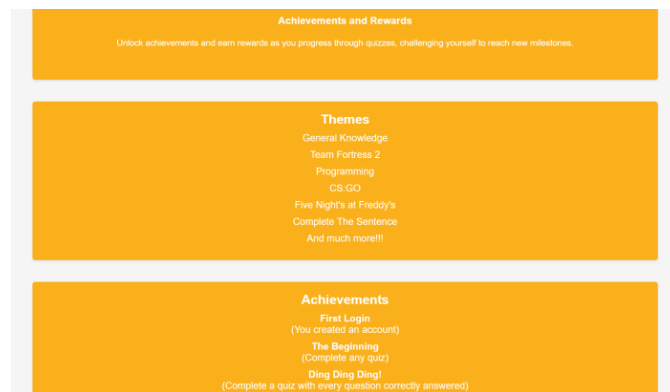


Figura 30 - 4ª parte do website (Themes/Achievements)

A penúltima parte do website somente possui como objetivo informar os utilizadores sobre os temas e algumas conquistas presentes no aplicativo.

Para concluir, será mostrada a última parte do website, com pouca informação:

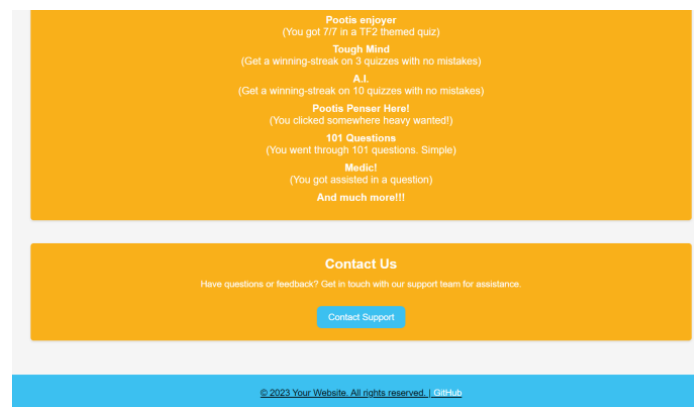


Figura 31 - 5ª parte do website (Achievements/Contact Us/Copyright)

Esta última parte revela o restante das conquistas e um local para poder depois contactar com os criadores do aplicativo. Também possui uma mensagem a referenciar os direitos reservados.

12.2. Screencast do website

De seguida será apresentado o *screencast* do website desenvolvido, onde estão presentes todos os tópicos já referidos:



Screencast website
Pootis Quiz.mkv

12.3. Screencast do aplicativo

Agora será também apresentado o screencast do aplicativo desenvolvido em ambiente android, que foi mostrado no aplicativo:



Screencast
aplicação android Pi

13. Conclusão

Tomando em conta todas as etapas seguidas, o trabalho em equipa eficiente e os obstáculos ultrapassados, acredita-se que a aplicação quiz desenvolvida é bastante satisfatória, uma vez que ultrapassou algumas das nossas expectativas, tanto no desenvolvimento, implementações e o próprio comportamento que foi obtido nos testes com os utilizadores. Adicionalmente, recebemos as poucas críticas que os utilizadores nos proporcionaram e ambas foram aplicadas com sucesso.

O desenvolvimento do website também foi satisfatório, uma vez que não existiram quaisquer obstáculos no seu desenvolvimento.

Sobre o presente relatório, foi sempre tomado em consideração oferecer uma explicação clara e fluente através de todos os tópicos abordados, com a adição de imagens relevantes a esses mesmos tópicos. Não foi fornecido muitos exemplos de código uma vez que o mesmo é bastante longo, e o objetivo deste relatório não é ser extenso, mas sim explicar de maneira eficiente e clara aos possíveis avaliadores do projeto.

Com tudo isto em conta, o grupo ficou bastante satisfeito com o resultado final do aplicativo, porque novamente foi tudo ultrapassado com sucesso e com eficiência.