

## Arquitecturas Móveis

### Trabalho Prático 2022/2023

**NOTA:** O jogo descrito neste trabalho tem direitos de propriedade intelectual associados, não sendo autorizada a sua utilização fora do contexto deste trabalho de avaliação, nomeadamente para criação de jogos e sua publicação sem a autorização prévia dos autores.

No contexto deste trabalho pretende-se desenvolver uma versão de um jogo *Android* que permite desenvolver a agilidade no cálculo matemático dos jogadores. O jogo desenrola-se num tabuleiro 5 x 5 similar ao apresentado na Figura 1, no qual são apresentadas 6 expressões matemáticas: 3 na horizontal e 3 na vertical. Os valores e operações matemáticas são sorteados de forma aleatória. O jogador terá de escolher a linha ou coluna correspondente à expressão matemática com maior valor, tendo em consideração a prioridade das operações (as operações 'x' e '÷' têm prioridade maior do que as operações '+' e '-'). Para indicar a expressão que pretende indicar como sendo a de maior valor, o jogador deverá realizar uma ação *swipe* vertical ou horizontal sobre a coluna ou linha, respetivamente.

15	+	34	-	23
x		÷		+
54	x	45	x	23
÷		x		+
90	+	87	x	10

Figura 1 Exemplo de tabuleiro de jogo

O jogo vai evoluindo em níveis de dificuldade crescente, controlados por um tempo máximo para acertar no número de expressões previsto em cada nível (por exemplo, acertar 5 expressões com valor máximo em 60 segundos). Quando o jogador falha avança para o próximo quadro dentro do mesmo nível. Se o jogador não conseguir atingir o número de respostas previsto para um determinado nível dentro do tempo disponível, o jogador perde o jogo. Por cada resposta certa é dado um bónus de tempo ao jogador, por exemplo 5 segundos, mas o valor acumulado nunca poderá exceder o limite de tempo dado no início de cada nível, nem será acumulado ao tempo atribuído para os níveis seguintes.

A evolução entre níveis é realizada através da variação dos parâmetros do jogo, como, por exemplo:

- gama de valores usados: 1..9, 1..99, 1..999
- conjunto de operadores: ['+'], ['+', '-'], ['+', '-', 'x'], ['+', '-', 'x', '÷']
- quantidade de expressões a responder corretamente por nível: 10, 15, ...

- tempo disponível para cada nível: 90s, 60s, ...
- tempo adicional a atribuir no caso de resposta certa
- ...

Neste trabalho é dada liberdade para que cada grupo possa definir as variações a serem aplicadas nível-a-nível. Como exemplo, o grupo pode definir que no primeiro nível terá uma gama de valores de 1 a 9, apenas será usado apenas o operador '+' e terá que responder a 5 questões em 60 segundos, com bônus de 5 segundos por cada resposta certa. Para o segundo nível pode, por exemplo, ser reduzido o tempo para 55s e manter todos os outros parâmetros. No entanto cabe aos elementos do grupo definir as alterações nível-a-nível que parecerem mais razoáveis e/ou desafiantes.

Este jogo, na sua versão base, é jogado apenas por um jogador, mas no contexto deste trabalho pretende-se também desenvolver uma versão multijogador.

### **Modo 1 Jogador**

Neste modo de jogo solitário o jogador irá tentando avançar nível-a-nível até que não consiga responder a todas as questões no tempo limite para o nível em curso.

O jogador vai acumulando pontos por cada nível que conclui, os quais serão contabilizados da seguinte forma:

- 2 pontos por cada expressão assinalada que corresponda ao valor máximo entre as expressões presentes no quadro
- 1 ponto por cada expressão assinalada que corresponda ao segundo maior valor

Por estratégia de jogo, um jogador pode optar por escolher propositadamente a segunda expressão com valor mais alto para obter mais pontos antes de atingir o tempo limite e o número de respostas previstas para o nível.

Quando o jogador termina um nível, o próximo nível iniciar-se-á automaticamente após 5 segundos. Durante esta passagem entre níveis deverá ser possível colocar o jogo em modo pausa. Se o jogador tentar terminar o jogo (através da tecla "Back") durante o jogo, deverá aparecer uma janela de diálogo para confirmação (o relógio de controlo do nível não deverá ser parado durante a visualização desta caixa de diálogo).

Deverá ser mantido um Top 5 na *Firestore* que permita guardar os jogos solitários com pontuações mais elevadas e tempos totais de jogo (tempo efetivo de jogo, não incluindo as pausas entre níveis).

### **Modo Multijogador**

Neste modo toda a gestão do jogo (jogadores, criação dos tabuleiros de jogo/expressões, verificação da escolha correta da expressão, contabilização dos pontos, deteção de final de jogo,...) deverá ser realizada pelo dispositivo designado por servidor. O servidor é o dispositivo do primeiro jogador que inicia o jogo neste modo. Este jogador comunicará aos restantes o IP do seu dispositivo, ao qual os restantes jogadores (clientes) se ligarão. Podem ser disponibilizadas outras formas de partilha facilitada do endereço IP, por exemplo, através de mensagem SMS, ou usar serviços de rede para a descoberta de outros jogadores na rede local (neste caso já não será necessário transmitir o endereço IP). Durante o jogo não se pretende qualquer utilização de servidores externos para realizar a comunicação. Todas as mensagens trocadas entre os dispositivos deverão respeitar o formato JSON.

O jogador servidor fica à espera da ligação dos restantes jogadores e decide quando é que o jogo se iniciará. No mínimo terá de se ligar um jogador para que este tipo de jogo possa ser iniciado. Não

se deve impôr limite ao número de jogadores (em termos de avaliação será verificado o suporte da aplicação a 1, 2 ou mais clientes, a que corresponderão notas diferentes).

As expressões de cada quadro serão criadas no servidor, o qual terá de assegurar que são usadas as mesmas expressões em todos os dispositivos, exatamente pela mesma ordem. Para garantir que todos os utilizadores respondem às mesmas questões deverá ser mantida uma lista de expressões, uma vez que poderão existir jogadores mais rápidos do que outros a responder a cada questão.

A verificação da jogada e atribuição de pontos (2 ou 1, de acordo com as regras do jogo solitário) será realizada pelo servidor, assim como, a gestão do tempo restante e bónus de tempo atribuído. Os dispositivos cliente transmitem a jogada ao servidor e, depois de verificada a situação de jogo, a mesma é transmitida a todos os clientes (inclusivamente para o que originou a jogada), para que estes possam apresentar a informação adequada no ecrã.

A deteção de final de jogo também deverá ser detetada pelo servidor e transmitida ao cliente. Caso exista um problema de comunicação durante o jogo, todos os jogadores transitarão para o Modo 1 Jogador, devendo o jogador ser informado através de mensagem adequada (em caso de interrupção, o jogo não irá integrar qualquer Top 5).

Quando um jogador não consegue responder ao número de questões previsto é eliminado do jogo continuando os restantes para o nível seguinte. Quando restar apenas um jogador, o jogo deverá continuar até este perder, o que corresponderá ao final do jogo.

Para além da contabilização de pontos já referida, neste modo serão atribuídos 5 pontos para o primeiro jogador a finalizar cada nível.

Durante o jogo deverá ser possível ir acompanhando o estado de jogo dos restantes jogadores (por exemplo: número de quadros, pontos, tempo restante). Um jogador que já tenha sido eliminado pode continuar a acompanhar a evolução dos restantes jogadores. Deverá ser tido em consideração que o espaço disponível no ecrã é reduzido, pelo que serão valorizadas formas eficazes para visualizar esta informação. A avaliação desta componente será realizada por bónus na classificação final do trabalho.

Quando um jogador termina um nível (com ou sem sucesso o nível), passa para um ecrã onde é mostrada informação sobre os jogadores ativos, os que já perderam e pontos acumulados por cada jogador. Esta informação deve ser dinâmica – a informação sobre os jogadores que ainda se encontra a jogar o nível deve ir sendo atualizada. Quando todos os jogadores terminarem o nível, passará automaticamente para o nível seguinte após uma pausa de 5 segundos.

Deverão ser mantidos dois Top 5 Multijogador em *Firestore*. Um Top 5 será para os jogos com mais pontos e o outro para os que demoraram mais tempo. Ao consultar pormenores de um jogo deverá ser possível consultar informação sobre os jogadores, pontos, níveis atingidos, tempos totais de jogo (tempo efetivo de jogo, não incluindo as pausas entre níveis), ...

### **Outras funcionalidades e informações**

A aplicação deve permitir a gestão do perfil do jogador (único no dispositivo), ao qual poderá ser associada uma fotografia. No modo multijogador deverá ser possível ver o nome e a fotografia de todos os jogadores enquanto se está em espera antes do início de cada nível. A fotografia deverá ser transmitida através do canal usado para gestão de jogo, não sendo obrigatório para este caso a utilização de JSON.

O formato de armazenamento dos dados dos Top 5 em *Firestore* é livre, podendo o grupo optar pelo formato que mais se adequar ao modelo de dados do jogo.

A atividade Android na qual se desenrolará o jogo deverá suportar os modos *portrait* e *landscape*, com *layouts* adequados a ambas. As restantes atividades não têm esta obrigatoriedade.

Deverá ainda possuir suporte para diferentes línguas, pelo menos, português e inglês. A aplicação deverá usar a língua configurada no sistema. Caso a aplicação não suporte a língua configurada, deverá ser usada a língua inglesa por omissão.

Todo o desenvolvimento deverá ser feito usando a linguagem de programação Kotlin.

### Cotações:

Interface e interação com o utilizador .....	10,0%
Modo 1 Jogador .....	20,0%
Modo Multijogador .....	25,0%
Evolução do jogo com níveis diferentes, respetiva geração e verificação .....	10,0%
Definição de perfil de jogador (com fotografia) e partilha do mesmo .....	5,0%
Gestão e apresentação dos Top 5 .....	10,0%
Suporte para diferentes línguas e ecrã de créditos (alunos, disciplina, ano letivo, ...) .....	2,5%
Suporte para diferentes orientações de ecrã (com <i>layouts</i> independentes) .....	2,5%
Robustez e qualidade do código (inclui, padrões, tratamento de erros e exceções) .....	10,0%
<i>Bónus: Visualização inteligente de informação de outros jogadores</i> .....	+0,0-5,0%
<b>Relatório Técnico</b> .....	<b>2,5%</b>
<b>Manual do Utilizador</b> .....	<b>2,5%</b>

**Realização do trabalho:** Grupos de 3 alunos

**Data de entrega:** 08:00 do dia 09.01.2023

### Forma de entrega:

Entrega de um único ficheiro em formato **ZIP** através do *Nónio* com o seguinte nome:

AMOV.2022.2023.<nr\_aluno1>.<nr\_aluno2>.<nr\_aluno3>.zip

- **Ficheiros noutros formatos serão alvo de penalização até 5% na nota final**

Este ficheiro deverá incluir:

- todo o código (pastas com os projetos) com todos os recursos essenciais para a compilação e execução. Nos projetos do *Android Studio* deverão ser removidos previamente os diretórios: <proj>/gradle e <proj>/app/build
  - **Caso os ficheiros referidos não sejam removidos será aplicada uma penalização de 5% na nota final**
- relatório técnico (PDF)
- manual do utilizador (PDF)