



Universidade da Beira Interior

Faculdade de Engenharia  
Departamento de Informática

© Pedro R. M. Inácio (inacio@di.ubi.pt), 2018/19

---

Propostas para Trabalhos Individuais  
Individual Work Proposals

**Programação de Dispositivos Móveis**  
**Programming of Mobile Devices**

Departamento de Informática  
Department of Computer Science  
Universidade da Beira Interior  
University of Beira Interior

---

Pedro R. M. Inácio  
inacio@di.ubi.pt  
2018/19

## Conteúdo

Conteúdo	2
Introdução	4
T1 – StandingSick	5
T2 – Debuxo do Tecido [a.k.a. TEBUXO]	5
T3 – <i>Peddy Paper</i> para o Museu dos Lanifícios [a.k.a. PeddyMuseum]	6
T4 – Dotcomma [a.k.a. <i>É a pontuação (,) idiota!</i> ]	6
T5 – onceupona [a.k.a. <i>O escritor Surrealista</i> ]	6
T6 – Receitas Médicas	7
T7 – Gestão de Finanças	7
T8 – Editor de Texto com Memória	8
T9 – Árvore Genealógica	8
T10 – Receitas Q.B.	9
T11 – BodyMoving	9
T12 – Pediduchos	9
T13 – Cardápio de um Restaurante	10
T14 – Quem Quer ser um Armário	10
T15 – Acumulador de Emoções	10
T16 – Gestão de Nomes de Utilizador e Palavras-Passe	11
T17 – <i>SELFLASS</i> : uma Aplicação Móvel para Auto-Avaliação da Predisposição para Voluntariado	11
T18 – <i>CONTapelosDEDOSdosPÉS</i> : Aprender a Tabuada Interati-	

**vamente** **12**

**T19 – *PrunusApp*: Aplicação para Gestão de Pessegueiros e Contagens de Pêssegos** **13**

**T20 – *Words-Dont-Come-Easy*: Aprender a Soletrar** **13**

## Introdução

### *Introduction*

Neste documento estão 16 propostas de trabalho individuais para a unidade curricular de *Programação de Dispositivos Móveis* da licenciatura em Engenharia Informática da *Universidade da Beira Interior*. Cada proposta pressupõe o desenho e implementação de uma aplicação móvel relativamente simples. A partir do momento em que são publicadas, o(a) aluno(a) tem aproximadamente 2,5 semanas para desenvolver o projeto que lhe for atribuído.

Para a execução do projeto, o(a) aluno(a) deve escolher as tecnologias e plataformas que mais lhe convirem. Contudo, em caso de dúvida, sugere-se o uso da plataforma Android, da linguagem Java e do motor de bases de dados SQLite para o desenvolvimento do projeto associado a cada proposta. Cada proposta tem explicitamente discriminadas uma série de funcionalidades mínimas que a aplicação móvel a desenvolver deve suportar, bem como uma (ou mais) funcionalidades adicionais, cujo objetivo é o de fortalecer o trabalho desenvolvido. **Contudo, valoriza-se a iniciativa do(a) aluno(a) que apresente propostas que aqui não estão contidas ou implemente funcionalidades interessantes não descritas**, desde que devidamente discutidas com o docente (reserva-se a este o direito de aceitar novas propostas ou funcionalidades como trabalho válido para avaliação).

Os trabalhos devem ser submetidos na plataforma *moodle* até às 23:55 do dia de entrega do trabalho e os nomes dos ficheiros deve seguir a **nomenclatura** especificada na área da unidade curricular na plataforma *moodle*. A submissão eletrónica deve ser composta por 2 ficheiros: (i) o arquivo *.apk* e (ii) um arquivo *.zip* contendo o projeto da aplicação. Note que a ferramenta *moodle* **bloqueia** automaticamente a submissão de trabalhos às 23:55 do dia da entrega. É absolutamente necessário que a *apk* instale num dispositivo Android, nomeadamente com Android 5.0. O código do trabalho será avaliado em algumas ocasiões, pelo que deve estar devidamente documentado (e.g., com comentários entrelinhas, mas não abusar).

Como nota final, apenas referir que se irá considerar a submissão dos melhores trabalhos na *Google Play*, caso se tratem de aplicações Android™.

## **T1 – StandingSick**

### ***T1 – StandingSick***

Quando um utente vai às urgências ou a uma consulta, tem normalmente de esperar pela sua vez bastantes minutos. A ideia deste projeto é o de construir uma aplicação que permita ir preenchendo um relatório simples sobre as dores e sintomas que o apoquentam enquanto espera. Na sua versão mais simples, a aplicação deve mostrar apenas uma série de questões predefinidas (ver a seguir) e gerar um relatório para mostrar ao médico. O conjunto de questões a considerar são:

- Tem febre?
- Tem dor de cabeça?
- Tem dores no abdómen?
- Tem dores nos braços ou nas pernas?

Todas as questões devem ser oferecidas aos olhos dos utilizadores de forma muito intuitiva (uma questão por ecrã e uso de letras grandes). Todas as questões têm respostas de escolha múltipla. Eventualmente, pode ser adicionada uma questão com resposta em texto a introduzir pelo utilizador. A aplicação deve gerar o relatório em formato TXT e permitir mostrar esse relatório ao médico, no próprio dispositivo móvel do dono, logo que se entre no consultório. Deve ser possível aceder a um histórico das consultas. A aplicação deve preencher automaticamente um relatório para apresentação ao médico.

Uma versão mais elaborada da aplicação deve permitir que se mudem as perguntas e as respostas (de escolha múltipla) possíveis, nomeadamente através do acesso a uma parte de administração da aplicação guardada por uma palavra-passe (pode ser `teste123`). Opcionalmente, deve ser dada a possibilidade de enviar o relatório via e-mail para o endereço do doutor depois de responder às questões.

## **T2 – Debuxo do Tecido [a.k.a. TEBUXO]**

### ***T2 – TEBUXO***

O Debuxo de um tecido define o modo como se efetua o entrelaçamento dos fios para fazer o tecido. Existem vários debuxos, alguns simples, outros bem mais complexos. Alguns são conhecidos por nomes específicos. Este projeto visa desenvolver uma aplicação informativa e um jogo para dispositivos móveis que aborde o tema dos debuxos. A parte informativa é dedicada à apresentação de fotos e esquemas com uma breve descrição sobre cada um dos debuxos. Esta parte constitui a versão mais simples da aplicação. Deve ser incluída alguma informação na própria aplicação (i.e., estaticamente, como recursos), mas deve também ser fornecida uma interface de administração que permita adicionar novo conteúdo à aplicação, guardada por uma palavra-passe de administrador (`teste123`).

A versão mais elaborada da aplicação é a que inclui um jogo para elaboração de Debuxos (e.g., a aplicação sugere um nome, dá uma dica, e permite que o utilizador desenhe o padrão num quadriculado). O jogo deve ir pedindo Debuxos cada vez mais complicados e apresentar uma classificação no final. Adicionalmente, pode ser dada a possibilidade de partilhar as classificações e o facto de jogar em redes sociais. Eventualmente, a aplicação pode conter um *Easter Egg* que permita fazer e exportar um debuxo para foto de perfil no *facebook*.

**Proponente:** Rita Salvado (Professora do Departamento de Ciência e Tecnologia Têxtil)

**Contacto:** `lrbss@ubi.pt`

### **T3 – *Peddy Paper* para o Museu dos Lanifícios [a.k.a. PeddyMuseum]**

#### ***T3 – PeddyMuseum***

O objetivo deste trabalho é o de projetar e desenvolver uma aplicação para o Museu dos Lanifícios da UBI (ver <http://www.museu.ubi.pt/>) que ofereça ao utilizador uma aventura ao estilo *peddy paper* no contexto do Museu. Neste tipo de aventura, o jogador é normalmente orientado por um percurso, resolve charadas e visita locais de interesse. A versão mais simples da aplicação pode ser concretizada por um conjunto de ecrãs que mostrem esse percurso com recurso a ficheiros estáticos (fotografia e informações) e fornecidos logo com a aplicação aquando da instalação. Deve ser codificado pelo menos um percurso na aplicação (pode conversar com a Professora proponente da proposta – ver em baixo). Quando o utilizador abre a aplicação, esta deve mostrar um ecrã de escolha de percursos (se houver mais do que 1). Ao clicar num dos percursos disponíveis, ao utilizador deve ser mostrado um ecrã com o ponto de início e indicações precisas de como lá chegar. Através de botões de `avancar` e `retroceder`, o utilizador deve depois poder navegar pelas etapas do percurso. Algumas dessas etapas podem ser compostas por charadas e perguntas, às quais o utilizador deve poder responder na própria aplicação.

Uma versão mais elaborada desta proposta passa por: (i) disponibilizar forma de se partilhar factos (e.g., a chegada a um ponto do percurso, a resolução de uma charada, etc.) da aventura *Peddy Paper* em redes sociais; (ii) adicionar vários percursos alternativos; (iii) os vários percursos alternativos serem escolhidos aleatoriamente no início da aventura; e (iv) poder adicionar fotografias ou comentários em pontos críticos da aventura.

**Proponente:** Rita Salvado (Professora do Departamento de Ciência e Tecnologia Têxtil)

**Contacto:** lrbss@ubi.pt

### **T4 – Dotcomma [a.k.a. *É a pontuação (,) idiota!*]**

#### ***T4 – Dotcomma***

Pretende-se desenvolver uma aplicação que permita efetuar algumas alterações simples sobre textos curtos (aproximadamente 2000 caracteres, incluindo espaços). A ideia é fazer jogos literários simples em que se mostre que retirar espaços, vogais ou pontuação tem diferentes efeitos na forma como o humano entende o texto. Numa versão mais simples o utilizador poderá decidir retirar ao texto todos os sinais de pontuação ou uma das vogais (a,e,i,o,u). O texto deverá ser re-apresentado não contendo um espaço em branco no local do carácter ou símbolo retirado, mas antes fazendo recuar o carácter que lhe é posterior. Os textos sobre os quais se intervém deverão ser selecionados de um corpus pré-existente.

Na versão mais elaborada, a aplicação deverá permitir que se adicionem novos textos aos pré-existentes, mantendo um limite máximo quanto à dimensão dos mesmos. Deverá ser dada a hipótese ao utilizador de retirar do texto: toda a pontuação ou apenas os símbolos de pontuação que o utilizador selecionar (de 1 até todos-1) e/ou qualquer um dos caracteres de a a z. Os textos truncados deverão poder ser guardados e indicar em cabeçalho os símbolos ou caracteres que lhe foram retirados.

Nota: Esta aplicação poderá ser usada com alunos do básico no sentido de enfatizar a importância da pontuação na compreensão do sentido de um texto demonstrando que a falta de uma letra é mais facilmente resolvida do que a falta de uma vírgula.

**Proponente:** Jorge Pedro Ferreira (Professor de Português do ensino secundário)

**Contacto:** asqualidadessemhomem@gmail.com

### **T5 – onceupona [a.k.a. *O escritor Surrealista*]**

#### ***T5 – onceupona***

Esta aplicação pretende ser uma versão virtual do clássico jogo do papel dobrado cuja origem, em França, se deve ao movimento surrealista. O jogo consiste em compor uma frase, texto ou desenho através da colaboração de várias pessoas. Por exemplo, cada pessoa escreve uma frase no papel e, antes de passar o papel para outra pessoa, dobra-o de forma a deixar apenas as últimas duas palavras visíveis. A próxima participante continua a construir a frase ou texto partindo daquilo que entender apenas daquelas duas palavras. O exemplo, tornado clássico, que deu nome ao jogo, está contido na primeira frase obtida desse modo: *O cadáver esquisito beberá o vinho novo.* por Mário Cesariny.

A versão mais simples da aplicação deverá permitir configurar o número de utilizadores e de rondas (e.g., 2 utilizadores, 20 rondas). Um utilizador deverá escrever um texto com um máximo de 140 caracteres, tendo o utilizador seguinte apenas acesso às duas últimas palavras do texto do utilizador anterior, a partir do qual deverá iniciar a escrita do seu texto. O jogo deverá funcionar num só dispositivo móvel (i.e., apesar de ser multi-utilizador, apenas presume que o dispositivo móvel vá trocando de mãos para as várias rondas). No final o texto deverá ser apresentado integralmente, para regozijo dos jogadores.

Uma versão mais elaborada deste jogo deve diversificar o leque de parâmetros que se podem ajustar, desde o número máximo de caracteres por utilizador/ronda até ao número de palavras a apresentar ao utilizador seguinte. Deverá ser ainda dada a hipótese de, a cada ronda, alterar a ordem dos utilizadores, desde que todos participem em cada ronda. A possibilidade de o texto obtido ser reproduzido no final do jogo recorrendo ao motor de voz pode ser bastante interessante.

Nota: A aplicação para além do seu potencial enquanto passatempo social poderá ser eficazmente utilizada em oficinas de escrita criativa. Se não houver jogo semelhante na *Google Play*, pode ser interessante publicá-lo.

**Proponente:** Jorge Pedro Ferreira (Professor de Português do ensino secundário)

**Contacto:** [asqualidadessemhomem@gmail.com](mailto:asqualidadessemhomem@gmail.com)

## **T6 – Receitas Médicas**

### ***T6 – Prescriptions***

O objetivo principal deste trabalho é implementar uma aplicação para gerir a tomada de medicamentos. Na sua versão mais simples, a aplicação deve fornecer meios para definir a forma como, pelo menos, um medicamento deve ser tomado. Depois de definido um medicamento, e por cada vez que a aplicação é acedida, esta deve mostrar quanto tempo falta para as próximas tomas, permitindo adicionalmente que se assinala determinada toma como efetuada. As tomas mais antigas ou as já efetuadas devem ser retiradas da vista do utilizador.

Na sua iteração mais elaborada, devem ser pensadas formas de suportar mais do que um medicamento e eventualmente disparar alarmes ou notificações às horas das tomas. Devem ser adicionadas perspetivas que permitam ver as tomas em determinado dia, e pensada uma funcionalidade que permita adicionar

## **T7 – Gestão de Finanças**

### ***T7 – iMBroke***

O(a) aluno(a) que elaborar nesta proposta deve desenvolver uma aplicação móvel que permita ao utilizador gerir os seus gastos mensais. Na sua versão mais simples, a aplicação deve providenciar apenas uma forma de rapidamente adicionar, editar ou eliminar gastos (concretizados por uma breve descrição, local onde foi gasto e quantidade monetária) e uma perspetiva integrada sobre todos os gastos do mês corrente, nomeadamente através de totais semanais e mensais.

Uma versão mais elaborada pode adicionar várias funcionalidades, nomeadamente: (i) permitir definir o período temporal anterior para o qual se quer obter a perspetiva integrada dos gastos; (ii) facilitar o agendamento de gastos persistentes ou repetitivos, que são verificados e processados a cada acesso à aplicação (e.g., se for agendado um gasto para o 1.º dia de cada mês, a aplicação verifica se esse dia já passou e se esse gasto já foi adicionado este mês); (iii) permitir estruturar os gastos por categoria (e.g., *Gastos com Restauração*, *Gastos em hipermercado*, *Gastos com Viatura*, etc.). Caso seja implementada esta última funcionalidade, deve também ser providenciada forma de adicionar, editar e eliminar categorias.

## **T8 – Editor de Texto com Memória**

### ***T8 – Text Editor with Memory***

O objetivo deste projeto é implementar um pequeno editor de texto capaz de propor a próxima palavra a ser escrita com base na palavra ou palavras já escritas. Para atingir a sua principal função, o programa deve ser capaz de popular uma base de dados com o que o é escrito pelo utilizador à medida que este usa o programa ou, opcionalmente, ter uma função específica para ler textos, por exemplo fornecidos em ficheiros com extensão .txt. Na sua versão mais simples, a aplicação pode simplesmente popular a base de dados a partir das palavras inseridas num campo de texto da aplicação, assumindo que as palavras são delimitadas por espaço, pontuação ou mudança de capitalização. As sugestões podem ser dadas em caixas de texto adicionais, que são atualizadas a cada letra digitada na caixa de texto principal. Juntamente com as caixas de texto com as sugestões podem estar botões que, ao serem clicados, inserem a sugestão a seguir à palavra que o utilizador está a escrever. O programa deve permitir guardar os textos em ficheiros do armazenamento externo ou partilhá-los via *e-mail*.

Uma versão mais elaborada deve permitir a leitura de textos fornecidos em ficheiros de texto só com o intuito de popular a base de dados. Adicionalmente, o editor pode fazer sugestões com base na combinação / sequência das últimas duas palavras, embora se deixe ao critério do(a) aluno(a) a possibilidade de estudar sequências ou cadeias maiores. Finalmente, devem ser adicionadas funcionalidades de eliminação de palavras da base de dados.

## **T9 – Árvore Genealógica**

### ***T9 – The Tree of Life***

Esta proposta visa o desenvolvimento de uma aplicação para construção da árvore genealógica de um utilizador. As funcionalidades mínimas incluem a inserção, atualização ou remoção de familiares e respetivas relações. A aplicação deve fornecer uma perspetiva minimamente intuitiva da árvore do utilizador, e permitir a edição dos familiares após toque no nome dos mesmos na árvore. Note que as funcionalidades desta aplicação parecem sugerir fortemente a utilização de uma base de dados de suporte.

Como funcionalidades adicionais, aponta-se a possibilidade de fornecer a possibilidade de importar a árvore de outros utilizadores (dependendo do formato do ficheiro, que deve ser especificado no âmbito do projeto), sendo da responsabilidade do processo de importação a junção das duas árvores com sentido (e.g., se um primo facultar a árvore dele, a aplicação do utilizador deve ser capaz de fundir as duas árvores nesse ramo). A base de dados da versão mais elaborada deve também guardar mais informação, nomeadamente breves descrições dos familiares, a data de nascimento e da morte (caso aplicável). Eventualmente, pode ser dada a hipótese de tirar ou associar uma fotografia a determinado familiar.



## **T10 – Receitas Q.B.**

### ***T10 – Q.B. Receipts***

Este projeto tem como principal objetivo desenvolver uma aplicação móvel que permita popular uma base de dados de receitas e depois consultá-las de uma forma intuitiva. As receitas devem ser divididas em refeições leves e completas. A lista de ingredientes e utensílios necessários à preparação devem poder ser consultados num ecrã isolado, de forma a facilitar a sua compra numa visita ao supermercado. Na sua versão mais simples, a aplicação deve apenas suportar as duas funcionalidades descritas antes, sendo que cada receita deve ser mostrada através de uma série de ecrãs simples que mostram o texto de cada passo específico da receita e apresentam 2 botões: um para indicar que aquele passo da receita já está feito (`done`), permitindo evoluir para o próximo passo, e outro para voltar para trás (`back`).

Uma versão mais elaborada da aplicação pode, por exemplo, conter temporizadores para os passos da receita que deles precisem, bem como disponibilizar meios de exportar (partilhar) ou importar receitas (sendo que este último estará dependente do formato específico do ficheiro a importar, que deve ser definido no âmbito do projeto). Podem também ser adicionadas pesquisas personalizadas que permitem, entre outras, encontrar receitas que possam ser preparadas num determinado período de tempo. Também seria interessante *gamificar* estas receitas, pedindo ao utilizador que faça os vários passos da receita em contra-relógio, dando-lhe pontos em caso de acabar cada passo da receita dentro do tempo.

## **T11 – BodyMoving**

### ***T11 – BodyMoving***

Esta proposta visa o desenvolvimento de uma aplicação para aqueles utilizadores que gostam de fazer exercício físico (embora possa ser usada por qualquer pessoa) e que queiram saber com frequência a sua massa corporal. A principal funcionalidade da aplicação é, portanto, calcular automaticamente essa métrica sempre que seja inserido o peso. A altura deve ser um parâmetro configurável (mas fixo). A aplicação deve guardar os pesos e índices de massa corporal obtidos ao longo do tempo numa base de dados e fornecer a funcionalidade de ver o histórico para o último mês.

Uma versão mais elaborada da aplicação deve permitir escolher o período temporal para o qual se mostra o histórico e incluir lembretes para fazer exercício físico. Adicionalmente, pode incluir meios de gabanço em redes sociais (i.e., partilha do peso ou índice de massa corporal após inserção ou cálculo na aplicação). Talvez seja interessante também incorporar formas de mostrar os valores em gráficos ou metas a alcançar.

## **T12 – Pediduchos**

### ***T12 – Pediduchos***

O objetivo deste trabalho é construir uma aplicação para aceitar pedidos em cafés, pastelarias e restaurantes. Na sua forma mais simples, a aplicação deve disponibilizar um menu simples, ao empregado de mesa, com as seguintes opções: (i) criar um novo pedido; editar um pedido existente; fechar um determinado pedido. Na sua versão mais simples, a aplicação deve apenas mostrar caixas de texto para introdução manual dos pedidos. Os pedidos devem ficar guardados com a data e hora do sistema (introduzidos automaticamente) e com o nome do cliente e do empregado de mesa. Deve também ter um campo para indicar o número da mesa relativo ao pedido. Todos os pedidos (mesmo os fechados) devem ficar guardados no dispositivo móvel em forma de ficheiro ou base de dados SQLite.

Uma versão mais elaborada da aplicação deve adicionar funcionalidades. Por exemplo,

deve conter uma interface de administração, protegida por palavras-passe, que permita editar (i.e., consultar, adicionar, atualizar ou remover) uma lista de produtos (e respetivos preços) que podem ser pedidos pelos clientes. Nesse caso, a aplicação deve substituir as caixas de texto para introdução dos pedidos por *droplists* onde se podem selecionar os produtos. Neste caso, a aplicação deve também ser capaz de calcular o total da conta. Opcionalmente, a aplicação deve permitir enviar o pedido (e.g., por e-mail) para um endereço predefinido quando o empregado desejar ou ao fechar a conta.

## **T13 – Cardápio de um Restaurante**

### ***T13 – Carte d'un Restaurant***

O objetivo principal deste trabalho é implementar uma pequena aplicação para dispositivos móveis que mostre o cardápio (i.e., lista de pratos, descrições e preços) de um ou mais restaurantes. Note que a índole desta aplicação parece sugerir fortemente a necessidade de utilizar de uma base de dados de suporte. Na sua versão mais simples, pode ser assumido que a aplicação será apenas utilizada para um único restaurante. Para além da interface de utilizador que permite ver o nome dos pratos e os preços, a aplicação deverá conter uma parte para editar as informações do restaurante e os preços, mediante dupla confirmação do utilizador, e providenciar a descrição mais detalhada de cada prato apenas após se selecionar esse prato.

Na sua versão mais elaborada, a aplicação deve suportar vários restaurantes e facultar formas de importar mais informação relativo a listas de pratos e preços, e.g., a partir de ficheiros disponibilizados no armazenamento externo. Adicionalmente, podem ser pensadas formas de um utilizador votar nos pratos que mais gosta ou de partilhar informação acerca dos mesmos nas redes sociais. A interface de utilizador deve cristalizar uma navegação intuitiva.

## **T14 – Quem Quer ser um Armário**

### ***T14 – Who Wants to be an Armoire***

O objetivo principal deste trabalho é implementar um jogo para dispositivos móveis parecido com o *Quem Quer Ser Milionário*. Este jogo terá de ser suportado por uma base de dados, devidamente pensada para o efeito. Na sua versão mais simples, deve pelo menos permitir jogar o jogo e inserir, eliminar ou editar perguntas. A gestão das perguntas e respostas deve ser feita numa parte protegida por autenticação (com nome de utilizador `admin` e palavra passe `teste123`). A aplicação deve maximizar a longevidade do jogo, trocando a ordem das perguntas ou das respostas a cada instanciação do mesmo, recorrendo a números pseudo-aleatórios.

Na sua versão mais elaborada, o jogo deve suportar níveis de dificuldade, associados às perguntas, e implementar várias ajudas, nomeadamente daquelas conhecidas por *ajuda 50/50* e *ajuda do público*. Podem ser pensadas outras ajudas mais avançadas, e.g., a possibilidade de trocar de pergunta por outra com dificuldade diferente, isto após atingir determinada pontuação ou ter jogado um determinado período de tempo. O modo como estas funcionalidades são concretizadas é deixado ao critério do(a) programador(a). O jogo deve ainda permitir múltiplos jogadores e mostrar um *ranking* das melhores classificações.

## **T15 – Acumulador de Emoções**

### ***T15 – Emotions***

A ideia deste projeto é desenvolver uma aplicação que permita ir escolhendo emoções sentidas ao longo do dia. Ao abrir a aplicação, o utilizador tem oportunidade de escolher

uma emoção (e.g., contente, triste, em êxtase, doente, indiferente, etc.) e gravá-la num registo associado àquela hora. Na sua versão mais simples, o utilizador pode apenas escolher de um conjunto de emoções pré-existentes (i.e., *hardcoded*, mas guardadas numa base de dados). Esta versão deve também mostrar resumos diários dessas emoções (por exemplo, uma listagem com as horas e a emoções registadas). Cada emoção deve ter um valor associado entre 0 e 100 (absoluta felicidade deve ter o valor 100, absoluta tristeza deve ter um valor de 0). A aplicação deve produzir valores médios para o grau de felicidade tendo em conta esses valores. Estes valores médios devem ser dados para cada dia, para cada semana, e para cada mês.

Uma versão mais elaborada da aplicação deve permitir ao utilizador inserir emoções que ainda não estejam na base de dados. Deve ser permitido ao utilizador definir os valores que estão associados às emoções introduzidas. Seria interessante a aplicação ter uma funcionalidade para fazer algumas previsões acerca das emoções com base em histórico (e.g., ao introduzir que está triste às 10h00, o utilizador recebe uma mensagem a dizer que costuma ficar mais alegre depois de almoço, com base no histórico à mesma hora de dias passados).

## **T16 – Gestão de Nomes de Utilizador e Palavras-Passe**

### ***T16 – Username and Passwords Manager***

Com este trabalho pretende-se que o(a) aluno(a) desenvolva uma aplicação para gerir os nomes de utilizadores e as palavras-chave que uma pessoa possa acumular ao longo da sua vida virtual. A aplicação deve validar a identidade do utilizador antes de permitir editar ou aceder a quaisquer dados geridos pela aplicação (ver, e.g., aplicação *KeePass*). Na sua versão mais simples, os dados podem ser guardados numa base de dados em texto-limpo, mas devem ao menos ser fornecidas as funcionalidades de procurar por determinada combinação, quer pelo nome de utilizador, quer por palavra-passe.

Na sua versão mais elaborada a aplicação deve providenciar uma forma de gerar palavras-passe aleatórias. Para a funcionalidade referida em último, a aplicação pode, por exemplo, pedir quais caracteres queremos usar, bem como a quantidade. Esta versão deve guardar mais informação que a anterior, nomeadamente o *Uniform Resource Locator* (URL) dos sites a que pertencem as combinações nome de utilizador/palavra-passe, descrições e datas de expiração das palavra-passe. Caso se tente aceder a uma palavra-passe expirada, a aplicação pode sugerir a sua renovação. Como funcionalidade adicional, a informação na base de dados deve estar cifrada por uma chave mestra.

## **T17 – SELFCLASS: uma Aplicação Móvel para Auto-Avaliação da Predisposição para Voluntariado**

### ***T17 – SELFCLASS: a Mobile App for the Self-Assessment of the Predisposition to be a Volunteer***

Os principais objetivos deste projeto são desenhar e prototipar uma aplicação móvel para a disponibilização de um inquérito e realização de uma auto-avaliação da predisposição de uma pessoa para o voluntariado. A aplicação móvel deve ser implementada recorrendo a uma base de dados contendo uma série de perguntas para várias dimensões que precisam ser avaliadas neste contexto (como por exemplo Capacitação, Atitudes Agressivas, etc.). As questões e parâmetros de avaliação foram previamente preparados para cada uma destas dimensões, e serão disponibilizados a quem for implementar este projeto. A ideia é que, após o início da aplicação, um ecrã com as diversas dimensões que precisam de avaliação seja mostrado ao utilizador. O utilizador pode então escolher uma dessas dimensões, sendo depois guiado para o conjunto de perguntas relacionadas com a respetiva dimensão. A aplicação deve guardar o progresso e a pontuação à medida que o inquérito vai sendo feito, devolvendo a avaliação para cada uma das dimensões no final, bem como

uma pontuação geral ponderada, caso todos os questionários tenham sido respondidos. Junto das questões devem ser incluídas as referências bibliográficas para as mesmas (as referências são fornecidas – esta informação é estática).

Uma versão mais elaborada da aplicação deve armazenar os resultados numa base de dados (uma melhoria da base de dados indicada em cima) local. Com esta funcionalidade, deve ser disponibilizada, ao utilizador, uma interface de gestão desses dados, para o caso em que se quer ver ou apagar resultados antigos.

A aplicação deve ser preferencialmente desenvolvida para Android. O(a) autor(a) da melhor aplicação poderá ser convidado(a) a disponibilizar o aplicação na Google Play, a melhorar a sua documentação e a testar a aplicação em ambiente real no contexto de um projeto em curso.

### **Proponente**

Esta proposta foi submetida pela Doutora Marta Alves, do Departamento de Psicologia e Educação, contacto de referência para esta proposta (e a quem o grupo de trabalho poderá recorrer educadamente). O seu e-mail é [mpalves@ubi.pt](mailto:mpalves@ubi.pt).

**Nota:** pode ser preciso (e porreiro) apresentar o projeto a 28 de fevereiro. A(as) melhor(es) aplicação(ões) final(is) deve(m) chamar-se *Volunteer Self-Assessment*.

## **T18 – *CONTApelosDEDOSdosPÉS: Aprender a Tabuada Interativamente***

### ***T18 – CONTApelosDEDOSdosPÉS: Learning to Multiply Interactively***

O objetivo principal deste projeto é desenvolver uma aplicação educativa que fomente a aprendizagem da tabuada. A versão básica da aplicação deve apenas gerar dois números aleatórios inteiros e mostrá-los no ecrã, pedindo ao utilizador que diga qual é o produto dos dois. I.e., a aplicação mostra algo semelhante a

$$5 \times 3 = \boxed{\phantom{00}}$$

ao utilizador e, após este responder, a aplicação diz-lhe se estava certo ou errado, evoluindo outra vez para nova pergunta. A versão básica deve suportar vários níveis de dificuldade:

- Nível fácil – números gerados apenas entre o 0 e o 5 (tabuadas do 0 ao 5);
- Nível médio – números gerados apenas entre o 1 e o 10 (tabuadas do 1 ao 10);
- Nível especialista – números gerados apenas entre o 3 e o 12 (tabuadas do 3 ao 12);
- Nível épico – números gerados entre o 3 e o 100;
- Nível lendário – números gerados entre o 10 e o 1000.

A mudança de níveis de dificuldade deve ser simples de fazer, bastando clicar em botões que estão na mesma atividade onde aparece o desafio. Deve ainda ser possível guardar nomes de utilizadores da aplicação (deve ser assumido que a aplicação pode ser usada por mais do que um utilizador). A aplicação deve guardar o total de acertos para cada nível e utilizador.

Uma versão mais elaborada da aplicação deve ter um modo especial que ajusta o nível de forma automática de acordo com as seguintes regras: se um utilizador acertar 10 vezes seguidas em determinado nível, a aplicação muda automaticamente (mas avisando o utilizador) para o próximo; se o utilizador falhar 5 vezes seguidas, a aplicação muda automaticamente (mas avisando o utilizador) para o nível de dificuldade anterior. Deve também guardar os resultados de cada sessão (acertos vs. falhanços) numa base de dados e permitir visualizar esse histórico. Uma outra funcionalidade que pode ser adicionada é a possibilidade de partilhar um determinado resultado para as redes sociais ou por e-mail.

## **T19 – *PrunusApp*: Aplicação para Gestão de Pessegueiros e Contagens de Pêssegos**

### ***T19 – PrunusApp: App to Manage Peach-Trees and Number of Peaches***

O objetivo deste trabalho é desenhar e implementar uma pequena aplicação para gestão de um pomar de Pessegueiros (ou parte desse pomar) e para armazenamento de contagens de pêssegos. A aplicação deve, por isso, oferecer (duas) interfaces simples para introdução (e armazenamento numa base de dados local) de dois tipos de dados principais:

1. Pessegueiros – para os quais se devem guardar um número (identificador, incrementado automaticamente), o nome do pomar (se o nome não existir, deve ser adicionado à base de dados), e notas/detalhes (um campo livre para dados diversos);
2. Contagens – as quais devem estar associadas a Pessegueiros e a uma determinada data.

A versão básica da plataforma deve mostrar uma atividade para listagem e introdução de Pessegueiros. Depois de selecionado um pessegueiro, deve permitir evoluir para uma segunda atividade onde se podem colocar as contagens para esse pessegueiro, e eventualmente ajustar a data (que deve estar ao menos predefinida para o dia atual).

Uma versão mais elaborada da aplicação deve permitir gerir os pomares, nomeadamente através da definição das suas áreas e localização geográfica. Adicionalmente, deve permitir guardar a posição GPS de todos os pessegueiros, sendo que o código de identificação do pessegueiros deve ser um valor MD5 das duas coordenadas concatenadas. Deve também permitir guardar o estado do tempo aquando das contagens, e oferecer forma de visualização intuitiva das contagens feitas para cada pessegueiro ou para a totalidade dos pessegueiros de um pomar.

Este projeto tem potencial para vir a ter continuação como projeto de licenciatura no segundo semestre e insere-se num projeto de investigação e desenvolvimento atualmente em desenvolvimento.

## **T20 – *Words-Dont-Come-Easy*: Aprender a Soletrar**

### ***T20 – Words-Dont-Come-Easy: Learning to Spell***

O objetivo deste trabalho é implementar uma aplicação educativa que fomente a aprendizagem da escrita. A versão básica da aplicação deve apenas conter uma única atividade onde são mostradas palavras aleatórias incompletas (uma palavra de cada vez). Estas palavras devem ter origem num dicionário de palavras (descarregar um dicionário de uma fonte *online*, e.g., de <http://natura.di.uminho.pt/download/sources/Dictionaries/wordlists/>). Nesta versão básica, o jogador tem tantas tentativas quanto queira para acertar na palavra, mas a pontuação que recebe deve ser 100 a dividir pelo número dessas tentativas; e deve haver um botão para desistir de determinada palavra que só aparece após 5 tentativas falhadas. A versão básica deve permitir guardar o nome do utilizador (pedido da primeira vez que a aplicação é executada) e a sua melhor pontuação. A aplicação deve guardar o nível de um utilizador (deve subir-se de nível a cada 1000 pontos adquiridos).

A versão mais elaborada da aplicação deve ter uma base de dados de utilizadores que podem usar a aplicação à vez. Essa base de dados deve guardar as melhores classificações por sessão de 20 jogadas. Podem ser pensadas formas da complexidade do jogo evoluir com o nível do jogador (e.g., nos níveis iniciais, as palavras são mais pequenas; a cada 10 níveis o tamanho das palavras aumenta até 2 letras). Podem ser adicionados botões de *share* para quando são atingidas elevadas pontuações, bem como uma forma de verificar quais os vencedores de jogos anteriores.