



NOVA SCHOOL OF  
SCIENCE & TECHNOLOGY

**Interação Pessoa-Máquina**

**2024/2025**

# **Taskable**

---

## Stage 4 - Computational Prototype



**Authors:**

69951, João Norberto  
69970, Afonso Leote  
70199, Manuel Luz  
71798, João Fernandes

**Lab class Nº 1**

**Group Nº 5**

**Professor:**  
Teresa Romão

November 22, 2024

## O nosso protótipo:

Ao longo de três semanas, desenvolvemos um protótipo computacional funcional da nossa aplicação Taskable, para dispositivos Android, recorrendo à framework Flutter, que utiliza a linguagem de programação Dart. Para fazer download do respetivo apk, é pedido aos avaliadores que utilizem o seguinte link:

<https://afonsote.github.io/ipm/taskable.apk>

Caso seja necessário obter mais informação relativa ao nosso projeto, como objetivo, inspiração para desenvolvimento da app, os stages anteriores, ou até mesmo fotografias da aplicação é possível utilizar o seguinte link (também podem fazer download a partir do website):

<https://afonsote.github.io/ipm/>

## Instruções:

A aplicação foi desenvolvida para dispositivos móveis que utilizam o sistema operativo android, com o SDK da versão 35 (API 35 x86\_64, com target Android API 35 (Google Play)), utilizando o IDE Android Studio. O dispositivo de referência utilizado durante o desenvolvimento foi um telemóvel de 6.4 polegadas, que corresponde ao 'Medium Phone' disponibilizado pelo emulador do Android Studio. Apesar disto, o grupo tentou desenvolver a aplicação de modo a que fosse possível utilizar em qualquer telemóvel Android, independentemente do tamanho do ecrã, mas recomendamos utilizar um telemóvel de 6.4 polegadas para obter a experiência completa da nossa aplicação.

Para instalar a aplicação, deve-se utilizar o link disponibilizado acima para fazer download do apk, sendo também possível fazer através do nosso website. Se algum dos avaliadores possuir um dispositivo android, com 6.4 polegadas, recomendamos a instalarem o apk no respetivo dispositivo. De qualquer maneira, recomenda-se a utilização do Android Studio, recorrendo ao emulador de um 'Medium Phone' com o SDK referido anteriormente, para diminuir a probabilidade de ocorrência de erros ou problemas de segurança. O seguinte vídeo mostra como é que é possível correr um apk no IDE Android Studio:

<https://www.youtube.com/watch?v=qa0bUQv-n8Q>

## Briefing

A gestão de uma casa partilhada por várias pessoas envolve a divisão de diversas tarefas como limpeza, recolha de lixo, compras de bens comuns, entre outras. Estabelecer uma divisão de tarefas justa, acessível e rotineira não é simples, seja pela quantidade de pessoas, pela variedade de tarefas ou pelos imprevistos do dia a dia, especialmente para estudantes deslocados, que começam as suas primeiras experiências de vida independente.

Sendo assim, a nossa aplicação tem como objetivo ser uma ferramenta intuitiva e de fácil acesso que permita lidar com estas dificuldades, de modo permitir uma gestão eficiente da casa que inclui divisão de tarefas e contas, listas de compras e um sistema de pontos para recompensar quem fez mais tarefas.

# Cenários

## 1 - Criar e distribuir tarefas

“Sou a Maria, estudante de Eng. Informática do primeiro ano na FCT e recentemente partilho casa com a Paula e com o Passos. Nestes primeiros dias a casa está uma confusão, existe imensa loiça por lavar, a sala e a casa estão sujas e o lixo está cheio. Para distribuir essas tarefas de forma justa, instalo a aplicação “Taskable”, crio uma conta e uma casa com o Passos e a Paula, que já têm conta (usernames Passos e Paula), , e crio as tarefas “Lavar a loiça” que se repete diariamente, “Limpar a sala” e “Levar o lixo” que se repetem semanalmente no sábado. Como nenhum de nós se chega à frente, vamos atribuir as tarefas aleatoriamente e agora já temos uma distribuição organizada.”

## 2 - Trocar uma tarefa entre pessoas

“O meu nome é Maria e partilho casa com a Paula, já estabelecemos na aplicação “Taskable” que alternadamente no domingo uma de nós tem a tarefa “Limpar a cozinha”. Como sou estudante deslocada, este fim de semana vou para Freixo de Espada à Cinta e não vou poder “Limpar a cozinha”. Sendo assim, seleciono a tarefa de “Limpar a cozinha” da Paula da semana seguinte e peço para trocar com a minha “Limpar a cozinha” do próximo domingo, assim nenhuma de nós fica com trabalho a mais, nem com a cozinha suja!”

## 3 - Adicionar item à lista de compras e juntar listas

“Chamo-me Maria e partilho casa com o Passos e com a Paula. Como hoje tenho algum tempo livre, vou às compras. Com o “Taskable” adiciono “Bananas” e “Iogurtes” à minha lista de compras. Como quero receber pontos bónus, observo as listas dos meus colegas, crio uma lista conjunta e faço as compras por eles. Registo na aplicação os 4.62€ que o Passos me deve e os 12.35€ da Paula. Assim todos temos os nossos produtos, as contas facilmente divididas e eu verifico na Leaderboard que recebi os pontos bónus!”

## Features:

A nossa aplicação apresenta as seguintes funcionalidades:

1. Login e Sign Up:

O utilizador pode fazer login caso tenha conta, ou criar uma conta, caso ainda não tenha criado, fornecendo o seu nome de utilizador e a sua respetiva palavra-passe.

2. Criar Casa:

Após fazer login, na página inicial “My Homes”, o utilizador pode criar uma casa, indicando o nome e password do mesmo. Há também a possibilidade de adicionar pessoas à casa que vai ser criada, ou de criar o grupo sem adicionar pessoas. Caso a casa seja criada sem utilizadores, é possível adicionar novos utilizadores à casa, através do menu “Settings > Manage Users”, ou partilhando a palavra passe e nome do respectivo grupo, para que os outros utilizadores se consigam juntar.

3. Juntar a uma casa:

Também na página inicial, o utilizador pode-se juntar a uma casa à sua escolha, desde que insira o nome e a palavra passe corretamente da casa a que se quer juntar, ou desde que seja convidado. Tanto após esta funcionalidade, como na funcionalidade 2, as casas ficam sempre possíveis de ser acedidas pelo utilizador na página principal “My Homes”.

4. Criar tarefas

Após entrar na página da casa, na janela “Tasks”, existe um botão com um símbolo de “+” que permite criar tarefas que sejam necessárias de realizar. Quando se clica no botão para adicionar novas tarefas aparece uma janela onde é possível definir o nome da tarefa e a respetiva descrição (opcional). É também possível definir a priori o utilizador que a vai realizar mas não é estritamente necessário (é útil não escolher nenhum utilizador para depois dividir aleatoriamente por todos os utilizadores). Depois, é obrigatório escolher a frequência com a mesma se repete, podendo ser diariamente, semanalmente (obrigatório escolher os dias da semana) ou pontualmente, caso a tarefa seja apenas para realizar num único dia.

5. Distribuir tarefas:

Na janela “Tasks”, especificamente na subjanela “Home Tasks”, existem dois tipos de tarefas, as que têm utilizadores associados (“Assigned”) e as que ainda não têm (“Unassigned”). É possível distribuir tarefas de diferentes formas. Nesta janela, na aba “Unassigned”, é possível atribuir tarefas de forma aleatória, clicando no botão “Randomize” e selecionando as tarefas que se pretendem distribuir aleatoriamente, pelos vários elementos da casa. Outra das opções é selecionar qual o utilizador que vai fazer a tarefa no momento em que a criamos (ficando a task imediatamente na aba “Assigned”), ou o próprio utilizador escolher que vai fazer certa tarefa, clicando numa tarefa que ainda não esteja atribuída e confirmando que fica responsável pela mesma (botão “I’ll take it”).

6. Troca de tarefas:

Na aba das tarefas que estão atribuídas (em “Assigned”), caso um utilizador veja que não consegue fazer uma tarefa em determinado dia, devido a algum imprevisto, é possível fazer um pedido de troca com uma tarefa pertencente a outra pessoa. Para isso, seleciona a tarefa que quer ficar responsável, de outro elemento da casa, e pode fazer um pedido de troca selecionando uma das suas tarefas (neste caso, a que não irá conseguir fazer).

7. Display organizada de Tarefas pessoais:

Na janela “Tasks”, especificamente na subjanela “My Tasks”, são mostradas por ordem ascendente de data, as tarefas que foram atribuídas ao utilizador que fez login. São destacadas o número e as tarefas que foram atribuídas para serem realizadas no dia de “hoje” automaticamente, permitindo uma mais fácil visualização das tarefas que foram atribuídas ao utilizador da aplicação.

8. Criar listas de compras:

É possível criar várias listas de compras, sendo que todos os utilizadores de uma casa são capazes de ver as listas criadas. É então possível adicionar um item a uma qualquer lista do utilizador, ou à lista comum da casa, ou remover itens que fazem parte das mesmas. O utilizador tem também a possibilidade de ver as listas das outras pessoas, mas não tem a possibilidade de as editar.

9. Juntar listas de compras:

O utilizador pode juntar (dar *merge*) da sua própria lista, com a de outros utilizadores. Antes de juntar as várias, o utilizador tem a hipótese de escolher uma cor para cada uma destas, através de um *colorpicker*, sendo que se não escolher, são selecionadas aleatoriamente. Por fim, é criada uma lista conjunta com as diferentes cores escolhidas para cada utilizador. É sempre possível eliminar esta lista, caso o utilizador faça *swipe* da esquerda para a direita, ou fazer *check* aos itens já comprados, ao clicar duas vezes nos mesmos.

10. Criar dívidas entre utilizadores:

O utilizador pode criar dívidas entre dois utilizadores, inserindo a quantia que um destes deve ao outro, e uma pequena frase que descreva o porquê desta dívida. Todos os utilizadores de uma casa são capazes de ver todas as dívidas da mesma.

11. Leaderboard automático:

Cada casa tem um *leaderboard*, que mostra o número de pontos de cada utilizador. Este número é atualizado quando um utilizador faz uma tarefa, ganhando um número fixo de pontos, que é adicionado ao seu score atual no *leaderboard*. O utilizador com menos pontos numa casa, é o utilizador com menor número total de tarefas feitas, e vice-versa.

## 12. Definições da Casa:

No janela “Home Settings” existem 3 botões que permitem três ações: “Manage Users” que permite visualizar que utilizadores estão presentes no grupo e adicionar outros utilizadores se for necessário; “Change Home Name”, que muda o nome da casa, efetivamente atualizando o nome na lista de todas as casas em que o utilizador está presente; “Leave Home” que ao clicar, faz com que o utilizador saia da casa em questão.

As ferramentas utilizadas para implementar estas funcionalidades foram:

- **Flutter** - a framework utilizada para produzir frontend e backend
- **Dart** - linguagem de programação usada pelo Flutter
- **Paint.NET** - para design do logótipo e outras features visuais
- **Git e Github** - para controlo de versões do projeto
- **Android Studio** - Integrated development environment (IDE) para:
  - Escrever o código Flutter;
  - Emulador de dispositivo android.
- Utilizamos também as seguintes bibliotecas do Flutter:
  - *cupertino\_icons* - para utilizar icons do iOS;
  - *animations* - para animações pré-definidas no Flutter;
  - *flutter\_colorpicker* - para seleção de cores personalizadas;
  - *flutter\_test* - para criar e executar testes no projeto;
  - *flutter\_launcher\_icons* - para personalização dos icons na app;
  - *flutter\_lints* - para boas práticas de código Flutter;
  - *provider* - para manter o estado da sessão.

## Incompletudes:

Durante a implementação do protótipo computacional, alguns aspectos da interface foram utilizados, apesar de não estarem implementados, e algumas funcionalidades do sistema também não foram implementadas. Em seguida podemos ver uma lista com os tais aspectos e funcionalidades:

- Estado do sistema entre sessões:  
O nosso sistema não guarda estado entre diferentes sessões, ou seja, se o utilizador fechar a aplicação e voltar a abrir, vai encontrar a aplicação no estado predefinido durante a implementação. Como esta funcionalidade não impacta a interface, envolvendo apenas a integração do nosso sistema com uma base de dados, decidimos apenas manter o estado dentro de uma sessão.
- Login:  
O nosso sistema não obriga o utilizador a criar conta, ou a inserir um nome e palavra passe já existentes, visto que, esta funcionalidade não impacta a interface, envolvendo apenas a integração do nosso sistema com uma base de dados. A autenticação com serviços externos, como uma conta Google, também não foi implementada, pela mesma razão.
- Definições do utilizador:  
O nosso sistema não possui uma página para as definições do utilizador, já que esta seria extremamente semelhante à página de definições do grupo. Como a adição desta página não impacta a interface do sistema, decidimos não incluí-la.
- Notificação dos utilizador:  
Os utilizadores seriam notificados caso fosse criada uma dívida, caso fosse atribuída uma nova tarefa, ou caso uma das suas tarefas estivesse perto da data limite, ou atrasada. Como esta faceta do nosso sistema não impacta a interface do mesmo, decidimos não a implementar.
- Dias da semana:  
A nossa aplicação supõe que estamos sempre no dia 1 de dezembro, e as tasks são geradas apenas até ao final do mês, uma vez que seria apenas uma tarefa de backend, não influenciando a interface.
- Tarefas não realizadas:  
A nossa aplicação não lida com tarefas que não foram realizadas por utilizadores, atribuindo a responsabilidade ao utilizador de realizar as suas tarefas, sendo apenas a nossa aplicação uma ferramenta de apoio. Decidimos não implementar este aspeto da nossa aplicação, visto que não tem qualquer impacto na interface, tratando-se apenas de uma feature de backend.