

# Trabalho prático Flutter



## 1. Tema

Pretende-se criar uma aplicação simples de apresentação de previsões meteorológicas, tirando partido de uma API de acesso público a dados meteorológicos (ex: *OpenWeatherMap*, *WeatherAPI*, *AccuWeatherAPI*, ...).

## 2. Funcionalidades

Pretende-se que a aplicação apresente o estado meteorológico atual para uma determinada localidade, assim como a previsão para os dias seguintes. Deste modo, deverá disponibilizar os seguintes ecrãs:

1. **Ecrã principal:** onde o utilizador deverá entrar assim que a *app* estiver carregada e que deverá conter uma previsão sucinta do dia atual (e, idealmente, da hora atual), assim como dos dias seguintes que estejam disponíveis (deverá explicar porque optou por N dias). O utilizador deverá carregar num botão sempre que pretender atualizar os dados. Neste ecrã, o utilizador deverá poder selecionar um dia e navegar para o segundo ecrã, onde será visível os detalhes desse dia;
2. **Ecrã de detalhe:** onde será apresentada toda a informação relativamente à previsão para o dia selecionado. Deverá ser apresentada a condição, temperatura mínima/máxima assim como a imagem correspondente à condição;

Algumas sugestões de design podem ser encontradas [aqui](#), no entanto, o aluno tem liberdade total na criação da UI. No relatório final deverá explicar a razão da sua escolha e as opções tomadas relativamente à organização do layout e respetivo design.

Características adicionais da aplicação que serão avaliadas:

- Utilização da localização atual do utilizador em vez de uma localização fixa (deverá ser usado o **plugin de location**);
- Persistência da última atualização através do **plugin shared preferences** e deverá mostrar um texto com a informação de quando foi a última atualização;
- Suporte para internacionalização (deverá ser usado o *plugin* oficial **intl**);
- Ícone da *app* personalizado;
- *Splash screen* (deverá ser feito no lado nativo da *app*);
- Uso de animações implícitas para apresentação de conteúdos;
- Funções extra no ecrã principal (por exemplo, organização por semanas, etc.);
- Estrutura e organização geral do projeto.

O projeto da *app* deverá ser criado utilizando o comando:

```
flutter create --org pt.isec.<número de aluno> weather
```

**Nota:** caso o aluno não tenha acesso a equipamento adequado para compilar a versão iOS, poderá optar logo pela exclusão dessa componente durante o processo de criação, acrescentando o argumento `--platforms android` a seguir ao `create`.

### 3. Restrições

Para a implementação do projeto deverão ser usadas apenas as bibliotecas do Flutter e as referidas anteriormente. A gestão de estado pode ser toda feita através do recurso ao `setState()` como realizado nas aulas.

### 4.Submissão

Tendo em consideração que nem todos os alunos poderão ter acesso a equipamento adequado para realização do trabalho para funcionamento em iOS, para uniformização dos critérios de avaliação apenas será avaliada a componente Android. Assim, antes de submeter, deverá remover/eliminar as pastas `/ios` do projeto (caso a tenha) e executar o comando `flutter clean` (ou remover manualmente a pasta `<proj>/build` do projeto e `<proj>/dart_tool/flutter_build`).

O projeto deverá ser submetido no Nónio até às 13:00h do dia 23 de Janeiro de 2022. Após esta hora, as entregas terão uma penalização de 5% por cada hora de atraso.