

Enunciado

Pretende-se que seja implementada a app `WTest`, com as funcionalidades descritas neste documento.

Se for utilizado um gestor de dependências, deverá ser um dos seguintes:

- Carthage
- CocoaPods
- Swift Package Manager

Também deverá ser entregue um ficheiro README, com instruções, passo-a-passo, para garantir que as dependências são incluídas e o projecto compila.

Com o código, deverá ser entregue explicação do que foi feito e notas relevantes; seja por comentários ao código ou seja no ficheiro README.

Notas:

Cabe ao candidato gerir, dentro do tempo que tem, o balanço entre qualidade e completude. Ou seja, é compreensível que o candidato sacrifique a qualidade para implementar todos os requerimentos ou não implemente todos os requerimentos para garantir uma melhor qualidade; também é compreensível que o candidato tire proveito de bibliotecas de domínio público e explique somente o que lhe pareça necessário à compreensão do que foi elaborado.

Cabe também ao candidato decidir qual a melhor forma de navegar entre ecrãs, sendo que os *entry points* aos exercícios deverão todos de estar ao mesmo nível.

As *mock-ups* apresentadas não exemplificam todo o ecrã, servindo só como esclarecimento ao comportamento da GUI. É aceite que o candidato implemente um *layout* diferente do apresentado, desde que apresente o mesmo conteúdo.

Funcionalidade 1

A app deverá recolher a lista de códigos postais existente num ficheiro CSV publicado em `https://github.com/centraldedados/codigos_postais`, fazendo o *download* desse ficheiro.

A recolha dos códigos postais deverá ser feita no primeiro, e somente no primeiro, arranque da aplicação. Ou seja, a lista deverá ser gravada localmente para evitar nova recolha; exceptuando quando a app termina antes do *download* estar completo.

Enquanto o *download* ocorre, a app deverá apresentar um indicador de actividade.

Cada entrada deverá apresentar o código postal no formato , em conjunto com a designação postal.

- O teclado não deverá sobrepor-se a nenhuma entrada na lista; ou seja, se for feito *scroll* até ao fundo do ecrã, a última entrada não deverá ficar atrás do teclado.

- "2695"
- "2695-650"
- "2695 650"
- "São João"
- "sAo joA da TaLH"
- "sao talha"
- "talh joa"
- "joao talha"
- "talha 650 joao"

[illegible]

Funcionalidade 3

A app deverá apresentar um ecrã que liste uma colecção de artigos, adquirida remotamente.

- Chamando o método GET de

`https://5bb1cd166418d70014071c8e.mockapi.io/mobile/1-1/articles`.

Cada entrada na lista deverá apresentar o título, o autor e o sumário.

A lista de artigos deverá ser recolhida de forma paginada, com páginas de 10 entras.

- Usando os parâmetros numéricos, em *query string*, `page` e `limit`.

A recolha de dados deverá ser assíncrona.



Aquando a selecção de uma entrada, a app deverá navegar para um ecrã onde apresenta toda a informação do artigo, excepto o sumário.

O `hero` do artigo, deverá:

- ter o topo ancorado ao topo do ecrã
- ter o fundo ancorado ao topo da lista
- ter uma altura mínima
- ignorar a ancora do topo aquando necessário



Título

Autor

1 de Outubro de 2018

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque aliquam elit odio. Etiam vulputate tortor id ullamcorper semper. Praesent imperdiet magna sit amet lacus sollicitudin, laoreet molestie diam interdum. Cras posuere purus interdum libero feugiat vulputate. Aenean ultrices vel nisi eget pulvinar. Vestibulum mattis ac lectus id sodales. Phasellus sapien eros, interdum sit amet pharetra eu, molestie sed urna. Cras leo felis, sodales ac sollicitudin at,



Título

Autor

1 de Outubro de 2018

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque aliquam elit odio. Etiam vulputate tortor id ullamcorper semper. Praesent imperdiet magna sit amet lacus sollicitudin, laoreet molestie diam interdum. Cras posuere purus interdum libero feugiat vulputate. Aenean ultrices vel nisi eget pulvinar. Vestibulum mattis ac



Título

Autor

1 de Outubro de 2018

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque aliquam elit odio. Etiam vulputate tortor id ullamcorper semper. Praesent imperdiet magna sit amet lacus sollicitudin, laoreet molestie diam interdum. Cras posuere purus interdum libero feugiat vulputate. Aenean ultrices vel nisi eget pulvinar. Vestibulum mattis ac lectus id sodales. Phasellus sapien eros, interdum sit amet pharetra eu, molestie sed urna. Cras leo felis, sodales ac sollicitudin at, fringilla ut erat. Phasellus volutpat quam magna, non pulvinar ante cursus faucibus. Donec et quam at arcu feugiat fermentum. Etiam mattis accumsan turpis. Aliquam

Ambos os ecrãs deverão ser compatíveis com o *endpoint*

<https://5badefb9a65be000146763a4.mockapi.io/mobile/1-0/articles> .

Funcionalidade 4

A app deverá ter um formulário com para introduzir os seguintes dados:

- texto livre
- e-mail
- números
- somente versais e hífen
- uma data
- uma das seguintes opções:
 - Mau
 - Satisfatório
 - Bom
 - Muito Bom
 - Excelente

O teclado, ou qualquer outro controlo de inserção de dados, não deverá suprimir o acesso aos vários *inputs*.

Se **Funcionalidade 2** encontra-se implementada, o formulário também deverá permitir a introdução de um código postal; reutilizando o ecrã implementado para **Funcionalidade 2**.

O formulário deverá validar os dados inseridos.

- nenhum campo vazio deverá ser considerado válido

- o e-mail deverá ser validado por *regex*
- a data seleccionada:
 - não pode ser uma segunda-feira
 - não pode ser no futuro
- o campo que contém somente versais e hífens deverá:
 - ter no mínimo três caracteres
 - ter no máximo sete caracteres
 - não ter diacríticos

Se necessário, incluir botão para validar os dados.

Cabe ao candidato decidir como indicar a não validade dos campos.

Nota: Na perspective do utilizador, todos os dados são introduzidos por *text fields*. Cada *text field* deverá ter um *placeholder* alusivo ao conteúdo esperado.

Texto livre ...	Sed tempor pretium sapien et ullamcorp...
email@example.com	email@example.com
0000...	1234507
AAA-AA...	ABC-XY
Domingo, 4 de Novembro de 2018	Domingo, 4 de Novembro de 2018
Qualidade ▾	Muito Bom ▾
####-###, Designação Postal >	3780-425, Avelãs de Cima >

Funcionalidade 5

Esta funcionalidade depende de **Funcionalidade 3**.

Deverá ser criado um novo *target* da app, onde os ecrãs de **Funcionalidade 3** terão novos comportamentos.

Deverá ser usado, em alternativa, o *endpoint*

`https://5bb1d1e66418d70014071c9c.mockapi.io/mobile/2-0/articles` .

Cada entrada na lista deverá conter, em adição, a data de publicação.

O ecrã de detalhe deverá conter, em adição, o *avatar* do autor e a lista de comentários.


- O *avatar* deverá ser contido num círculo.
- Os comentários de um artigo são retornados pelo método GET de `https://5bb1d1e66418d70014071c9c.mockapi.io/mobile/2-0/articles/{id}/comments`.
- Os comentários deverão ser recolhida de forma paginada, com páginas de 15 entras.

Título 1 de Outubro de 2018 >
Autor
Aenean metus turpis, imperdiet et lacus ac, luctus pharetra quam. Nullam nec auctor quam, sed mattis enim.

Título 1 de Outubro de 2018 >
Autor
Aenean metus turpis, imperdiet et lacus ac, luctus pharetra quam. Nullam nec auctor quam, sed mattis enim.

Título 1 de Outubro de 2018 >
Autor
Aenean metus turpis, imperdiet et lacus ac, luctus pharetra quam. Nullam nec auctor quam, sed mattis enim.

Título 1 de Outubro de 2018 >
Autor
Aenean metus turpis, imperdiet et lacus ac, luctus pharetra quam. Nullam nec auctor quam, sed mattis enim.



Título 1 de Outubro de 2018
Autor

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque aliquam elit odio. Etiam vulputate tortor id ullamcorper semper. Praesent imperdiet magna sit amet lacus sollicitudin, laoreet molestie diam interdum. Cras posuere purus interdum libero feugiat vulputate. Aenean ultrices vel nisi eget pulvinar. Vestibulum mattis ac lectus id sodales. Phasellus sapien eros, interdum sit amet pharetra eu, molestie sed urna. Cras leo felis, sodales ac sollicitudin at,

integer placerat tincidunt augue et sollicitudin. Donec blandit felis sed est dictum, ut dignissim libero dictum. Etiam lacinia orci a feugiat egestas. Sed faucibus ligula sit amet nisi porta sodales. Aliquam mattis, velit vitae lobortis ultrices, sapien augue scelerisque velit, at dignissim odio nibh vitae erat. Maecenas nec vehicula arcu. Proin consectetur metus eu sagittis tincidunt.

Comentários

Autor 1 de Outubro de 2018
Cras posuere purus interdum libero feugiat vulputate.

Autor 1 de Outubro de 2018
Cras posuere purus interdum libero feugiat vulputate.

Autor 1 de Outubro de 2018
Cras posuere purus interdum libero feugiat vulputate.

Nota: Ambos os *targets* deverão ser compiláveis e executáveis. Ou seja, espera-se que o mesmo projecto permita compilar:

- um *target*, que utilize os dados indicados em **Funcionalidade 3**.
- um *target*, que utilize os dados indicados em **Funcionalidade 5**.