

## Como utilizar este Template

1. Faça uma cópia [ File → Make a copy... ]
2. Renomeie este arquivo para: **“Capstone\_Stage1”**
3. Substitua todos os textos **em verde**

## Instruções para Envio

1. Assim que completar todas as partes, faça o download deste documento como um PDF [ File → Download as PDF ]
2. Crie um repositório novo no Github para o capstone. Dê o nome **“Capstone Project”**
3. Adicione este documento no seu repositório. Tenha certeza que está nomeado como **“Capstone\_Stage1.pdf”**

---

### Descrição

### Público-Alvo/Intended User

### Funcionalidades/Features

### Protótipo de Interfaces do Usuário

#### Tela 1

#### Tela 2

### Considerações Chave/Key Considerations

Como seu app vai tratar a persistência de dados?

Descreva qualquer caso de uso específico (“corner case”) da experiência do Usuário (UX).

Descreva quais bibliotecas você utilizará e compartilhe a razão de incluí-las.

Descreva como você implementará o Google Play Services.

### Próximos Passos: Tarefas Necessárias

Tarefa 1: Configuração do Projeto/Project Setup

Tarefa 2: Implement UI for Each Activity and Fragment

Tarefa 3: Your Next Task

Tarefa 4: Your Next Task

Tarefa 5: Your Next Task

**Usuário do GitHub:** **adrian971029**

# InfoBook

## Descrição

Este aplicativo es para todos os amantes dos livros. Ele nos brinda informações sobre vários livros, seu autor, a capa e as principais informações sobre eles.

## Público-Alvo/Intended User

O publico alvo são amantes da leitura em geral. Pessoas que procuram conhecer novos livros para saber quais livros ler.

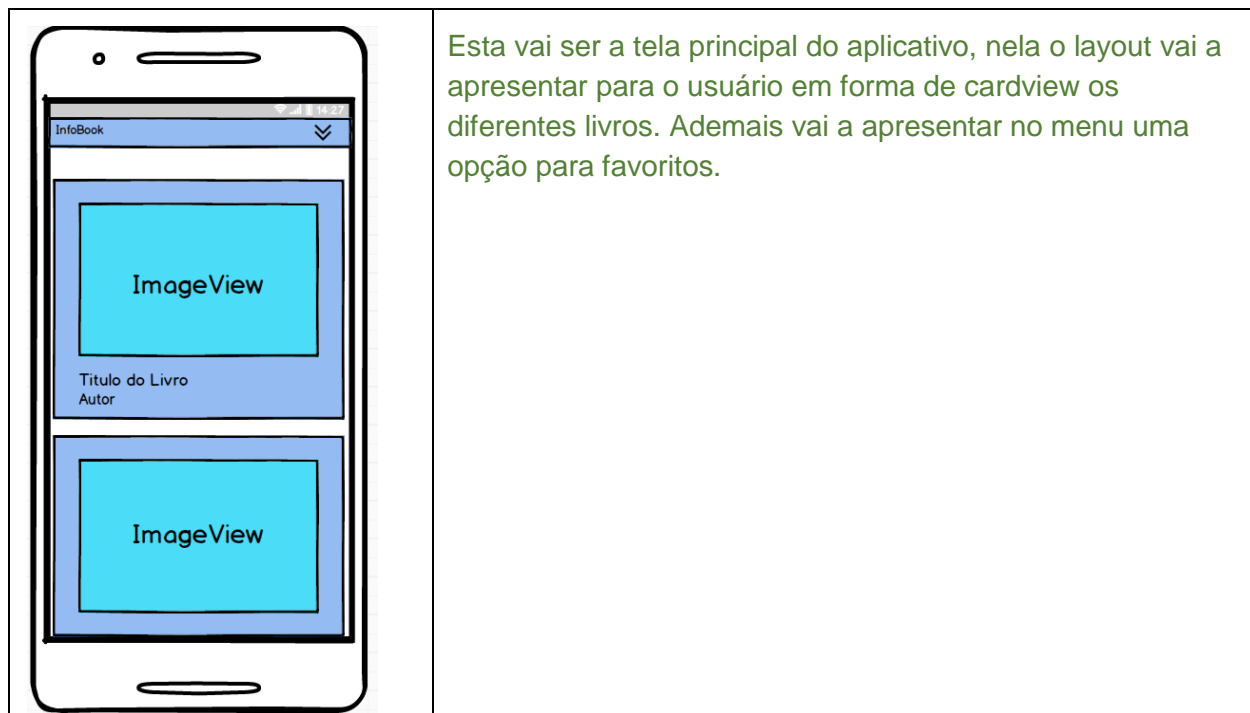
## Funcionalidades/Features

- Salva informações dos livros
- Mostra todo tipo de informações dos livros.

## Protótipo de Interfaces do Usuário

Elas podem ser feitas a mão (tire uma foto dos seus desenhos e os insira neste fluxo), ou usando um programa como o Photoshop ou Balsamiq.

### Tela 1



## Tela 2



### Tela 3



Adicione quantas telas achar necessário para demonstrar o fluxo de UI do seu app.

### Considerações Chave/Key Considerations

Como seu app vai tratar a persistência de dados?

Meu aplicativo vai tratar a persistência de dados utilizando um ContentProvider.

Descreva qualquer caso de uso específico (“corner case”) da experiência do Usuário (UX).

Tenho pensado criar um widget, sua utilidade vai ser apresentar para o usuário uma sugestão de leitura, resumindo cada certo 30 minutos vai apresentar um livro diferente para o usuário. Os dados vão ser recebidos de uma AP REST de livros gratuita, utilizando Retrofit para receber eles, em caso do aplicativo não ter internet vai informar para o usuário e vai mostrar um botão de atualizar para em caso de que a internet volte possa recarregar a ui, ademais o usuário vai ter no menu a opção da tela de favoritos onde vai poder procurar os livros que tenha salvado previamente. O aplicativo vai implementar duas versões uma gratuita e outra paga, na versão gratuita vai ser apresentados anúncios de testes para o usuário. O usuário também vai ter uma

opção no menu que al ser selecionada vai apresentar num mapa a biblioteca mais perto dele. Tudo o aplicativo vai ser implementado utilizando a linguagem Java.

**Descreva quais bibliotecas você utilizará e compartilhe a razão de incluí-las.**

Vou utilizar Picasso para o tratamento de imagens dos livros, o Butter Knife para melhorar meu código al tratar as view e Retrofit para consumir a API da internet.

Versões que vai ser utilizadas:

```
Butter Knife: com.jakewharton:butterknife:8.8.1
```

```
Retrofit: com.squareup.retrofit2:retrofit:2.4.0
```

```
Picasso: com.squareup.picasso:picasso:2.71828
```

**Descreva como você implementará o Google Play Services.**

- Google Mobile Ads vai ser utilizado para mostrar anúncios na versão gratuita
- Google Maps vai ser utilizado para apresentar ao usuário a biblioteca mais perto dele.

Versões que vai ser utilizadas:

```
com.google.android.gms:play-services-ads:15.0.1
```

```
com.google.android.gms:play-services-maps:15.0.1
```

## Próximos Passos: Tarefas Necessárias

Esta é a parte onde você falará sobre as principais funcionalidades do seu app (mencionadas acima) e as dividirá em tarefas técnicas tangíveis que você pode completar de forma incremental até finalizar o app.

### Tarefa 1: Configuração do Projeto/Project Setup

Ordem de configuração do projeto:

- Importar bibliotecas necesarias (Picasso, Butter Knife, Retrofit)
- Configurar no manifest permições para acesso a internet

## **Tarefa 2: Implementar a Interface de Usuário (UI) para cada Activity e Fragment**

- Construir a UI da MainActivity
- Construir a ToolBar
- Construir a UI da DetailsActivity
- Construir o layout do fragment utilizado na DetailsActivity DetailsIFragment

## **Tarefa 3: Implementar lógica de cada Activity para consumir API e mostrar os Livros**

Esta tarefa vai ser implementar a lógica de todas as activity para começar a mostrar os livros na tela principal e na tela de detalhes.

- Implementar lógica da MainActivity
- Implementar lógica da DetailsActivity
- Implementar lógica do FragmentActivity

## **Tarefa 4: Criar ContentProvider e toda a lógica para salvar e mostrar livros**

Esta tarefa vai ser criar o ContentProvider e toda a lógica para salvar e mostrar os livros na tela de favoritos.

- Criar ContentProvider
- Criar toda a lógica do ContentProvider para inserir e procurar Livros
- Implementar lógica do menu para alternar para opção Favoritos mostrar os livros salvados.
- Utilizar um Loader para carregar os livros salvados no ContentProvider.

## **Tarefa 5: Criar e implementar Widget**

Esta tarefa vai ser criar o widget com todos os passos necessários e o serviço que vai se encarregar de mostrar cada 30 minutos um livro diferente de sugestão para o usuário.

- Criar widget
- Criar e implementar serviço para que cada 30 min mostre um livro de sugestão para o usuário

## Tarefa 6: Criar layout para diferentes tamanhos de tela

Esta tarefa vai ser criar telas responsivas para diferentes tamanhos de tela

- Criar layout para sw-600dp

Adicione quantas tarefas que achar necessárias para concluir seu app.

---

### Instruções para Envio

4. Assim que completar todas as partes, faça o download deste documento como um PDF [ File → Download as PDF ]
5. Crie um repositório novo no Github para o capstone. Dê o nome “**Capstone Project**”
6. Adicione este documento no seu repositório. Tenha certeza que está nomeado como “**Capstone\_Stage1.pdf**”