

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: [rodrigogivisiez](#)

Musica clássica

Descrição

Este aplicativo traz uma lista das melhores músicas clássicas existente selecionadas exclusivamente para os fãs de carteirinha desta categoria de músicas que sem dúvida é a melhor de todas.

Um aplicativo super leve para você não se preocupar com travamentos e nem com excesso de espaço desnecessário ocupado por ele, você pode simplesmente clicar e ouvir as melhores músicas clássicas apresentadas de forma simples sem ter que se preocupar em procurar em toda web.

Você também pode contar com um sistema de ranking para ter um acesso rápido as canções mais acessadas, bem como pode favoritar suas canções favoritas para ter uma listagem com o seu perfil.

O aplicativo de músicas clássicas já trará um player junto para ter uma ferramenta totalmente voltada para melhor atender em termos de performance e customização..

Você pode contar ainda com uma funcionalidade para você publicar suas próprias composições, afinal de contas o melhor ainda está por vir e todos querem conhecer o seu talento, esta é uma ótima oportunidade de você mostrar para o mundo uma versão moderna de de uma canção clássica com sua própria interpretação ou uma composição de sua própria autoria.

O melhor de tudo é que você ainda pode baixar as canções preferidas para você poder ouvir em qualquer lugar independente de estar com acesso a internet ou não.

Se gostar do aplicativo deixe-nos um comentário para sabermos se estamos fazendo um bom trabalho e qualquer sugestão envie-nos para que possamos melhor a cada versão e transformar este no melhor aplicativo de música clássica do google play.

Público-Alvo/Intended User

O público alvo é para amantes de música clássica, que desejam ter acesso simplificado e organizado destas, podendo ser um amante amador e expor seu trabalho.

Funcionalidades/Features

Principais funcionalidades:

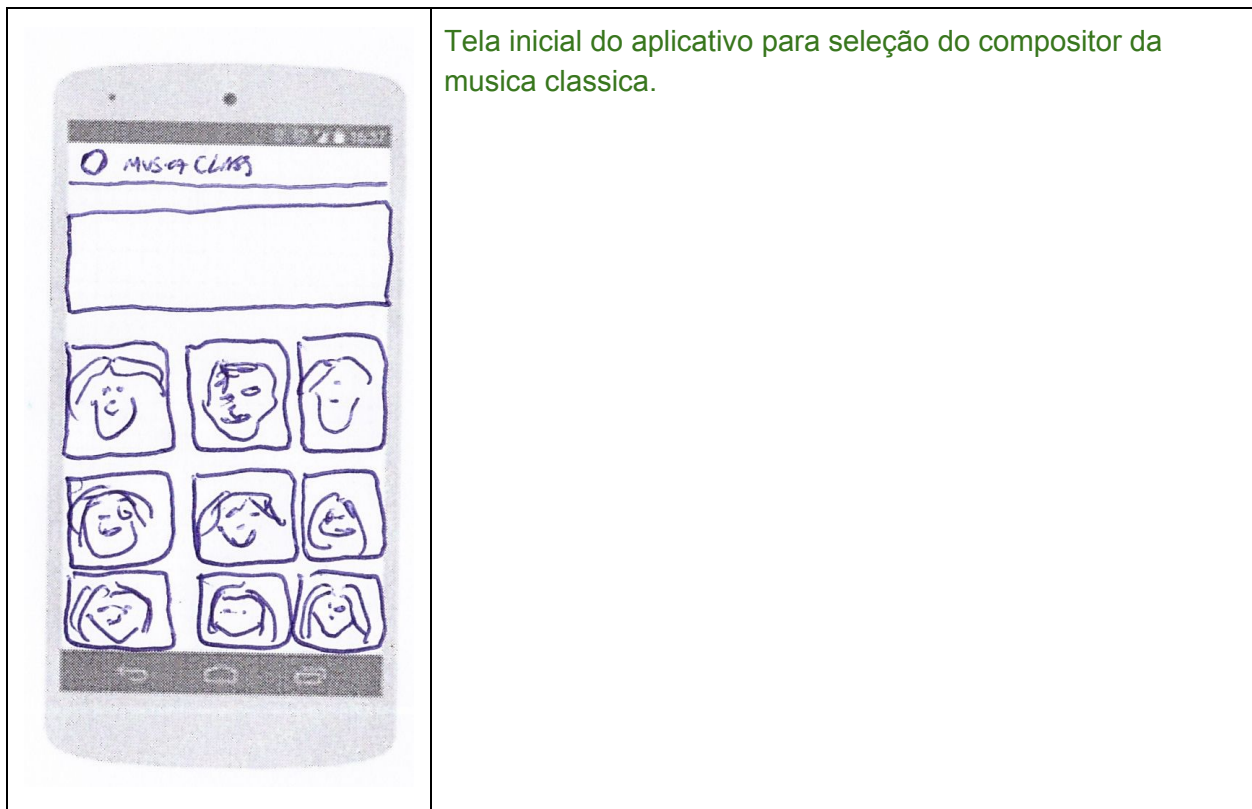
- Salva informações
- Envia áudio
- Ranking

- Roda audio em 2º plano
- Consulta em um servidor as músicas
- Envia para o servidor as músicas
- Favoritos
- Configurações

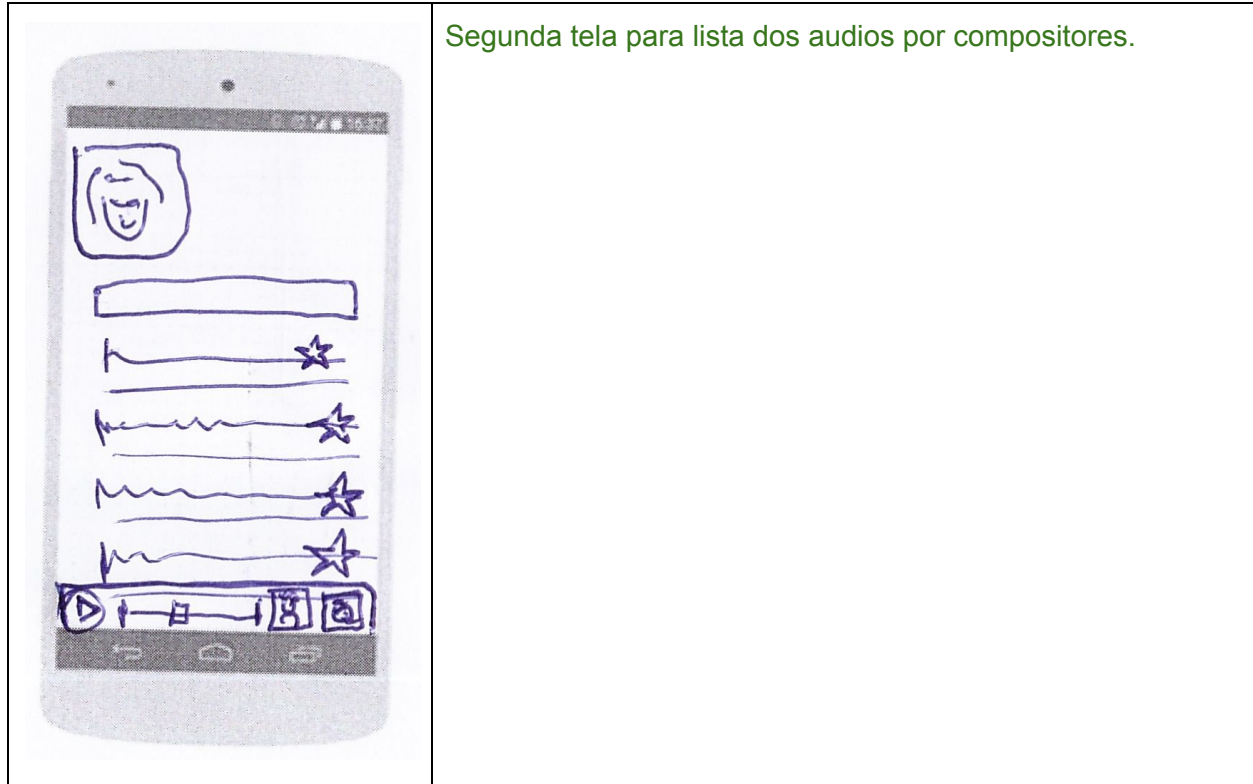
Protótipo de Interfaces do Usuário

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

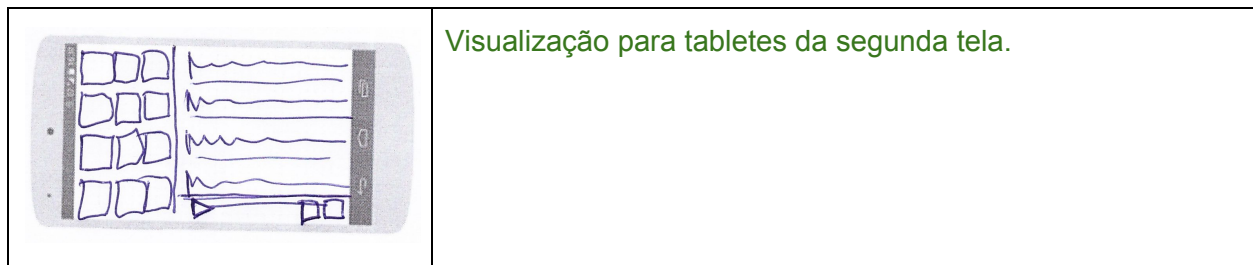


Tela 2

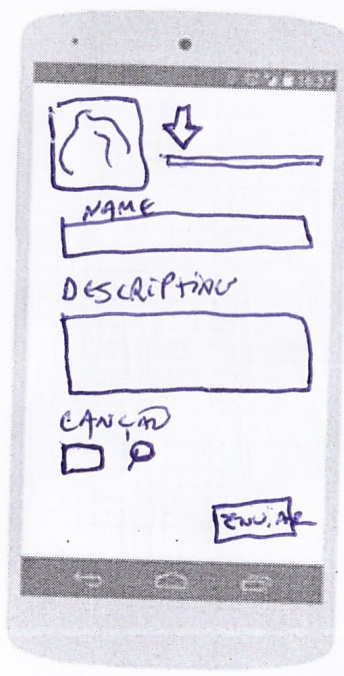


Adicione quantas telas achar necessário para demonstrar o fluxo de UI do seu app.

Tela 2




Tela 2

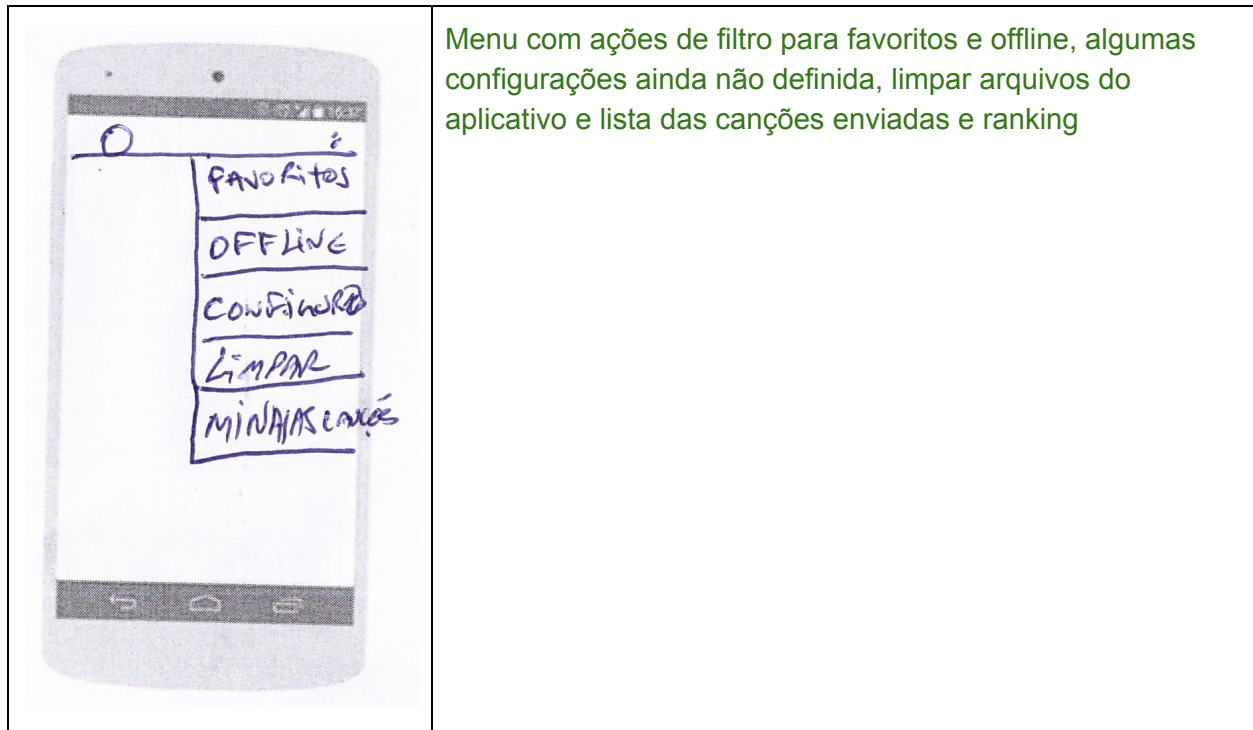
	<p>Activity para envio de audio. Com seleção de foto, nome , descrição e áudio.</p>
---	---

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

Tela 2

	<p>Widget para controle do audio</p>
---	--------------------------------------

Tela 2



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?

A persistência de dados será feita de forma nativa, bem como o content provider..

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

O aplicativo terá como tela principal uma lista contendo alguns compositores de música clássica.

Ao clicar em algum compositor listará suas composições que podem ser ouvidas online ou uma opção para ficar disponível off-line.

Poderá retornar a tela anterior com o botão padrão do android para navegação entre Activits

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

Picasso ou Glide para tratar o carregamento e cache das imagens.

Butterknife para não precisar ficar definindo findViewById

Exoplayer será utilizado para implementar o player

Descreva como você implementará o Google Play Services.

Firebase auth para autenticação

Firebase para banco de dados online.

Storage para servidor de arquivos.

Admob para anuncios.

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

Vou configurar o projeto para api mínima de 19 e estou adicionando as bibliotecas para cada função.

Vou configurar o firebase e o storage para ter certeza que esta de acordo.

Tarefa 2: Configuração do Projeto/Project Setup

Vou desenhar a estrutura de campos dos dados.

Criando uma prévia das classes que representarão meus dados.

Tarefa 3: Implementar a Interface de Usuário (UI) para cada Activity e Fragment

Vou neste momento tentar construir basicamente toda a infraestrutura de UI do aplicativo de forma que seja possível navegar por entre as activitys e poder ter uma visão geral de todo o aplicativo.

Tarefa 4: Criar Feature de api para comunicação com o servidor

Agora vou, “Implementar Google Play Services”, e “Tratar Casos de Erro”, para ter certeza que tenho acesso aos dados do servidor e do storage.

Configurar chaves key de acesso no firebase.

Logo após vou implementar a persistencia de dados.

Tarefa 5: Criar Feature de widget para controle do player

Agora estou pronto para implementar o widget, bem como as configurações necessárias no AndroidManifest.xml para que funcione o widget e o content provider

Tarefa 6: Criar Feature para implementação de favoritos

Tendo a persistência funcionando implementarei a função de favoritar e desfavoritar.

Agora vou implementar a função de download e upload de arquivos pelo firebase e storage.

Tarefa 7: Criar Testes Automatizados

Criar testes automatizados utilizando expresso junit e Mockito não somente aqui no final , mas durante os passos mais importantes do projeto.

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**”