

How to Use this Template

1. Make a copy [File → Make a copy...]
2. Rename this file: **"Capstone_Stage1"**
3. Replace the text in green

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it **"Capstone Project"**
3. Add this document to your repo. Make sure it's named **"Capstone_Stage1.pdf"**

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any corner cases in the UX.](#)

[Describe any libraries you'll be using and share your reasoning for including them.](#)

[Describe how you will implement Google Play Services.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

GitHub Username: [ahmedalimahmoud](#)

Music Player

Description

Quick search all music and audio files, Easy to support all music & audio file formats,

One of the most gorgeous and powerful music player for Android!

You can manage your musics easily, Music Player will guide you easily to find all the music in your phone.

plus you can back up your favourite Music

Intended User

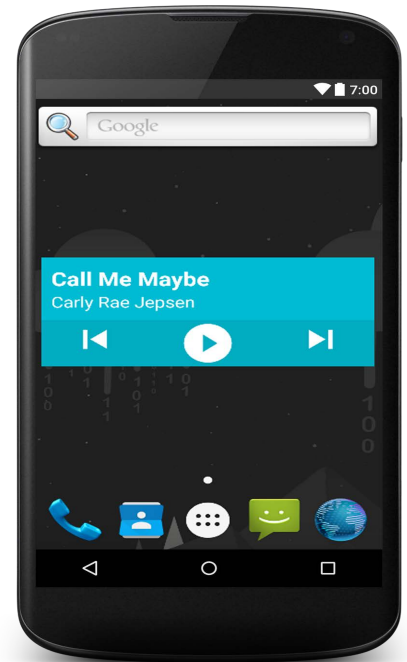
All Users Who want to play Music

Features

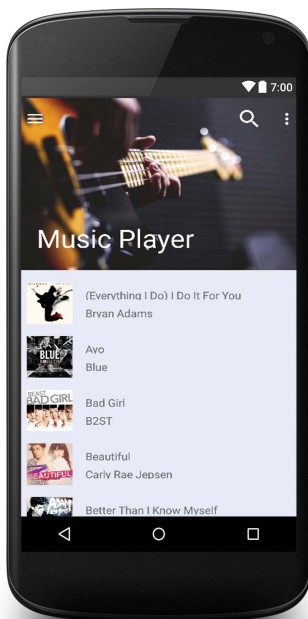
- Easy to Use
- Material Design Interface
- backup your list and retrieve it any time

User Interface Mocks

this is the Music
Player Widget

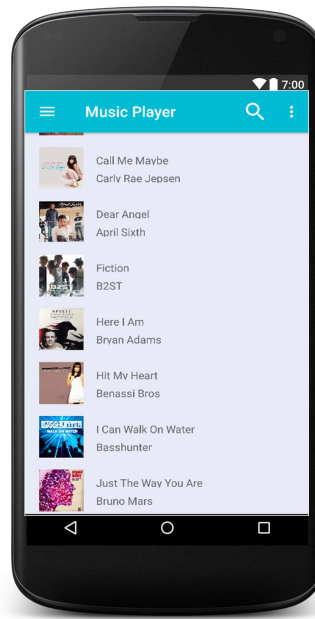


Screen 1



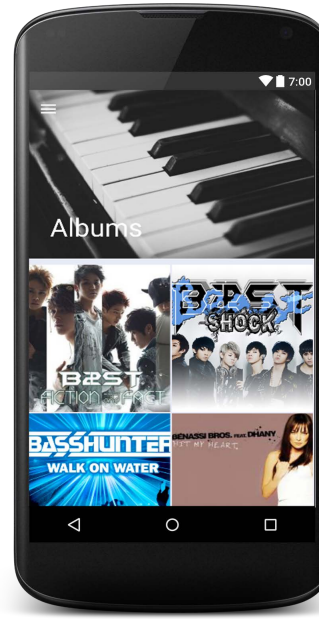
this is the home screen of
the app which view all
music

Screen2



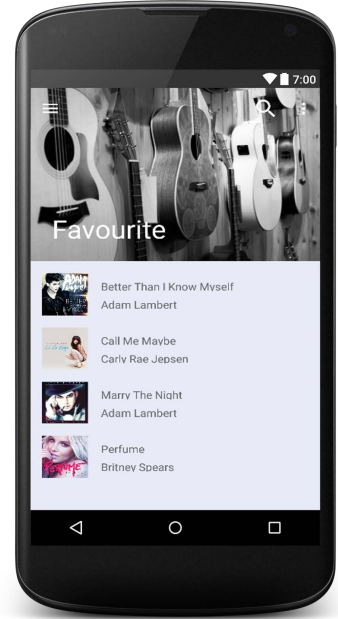
this is the home screen of
the app which view all
music

Screen3



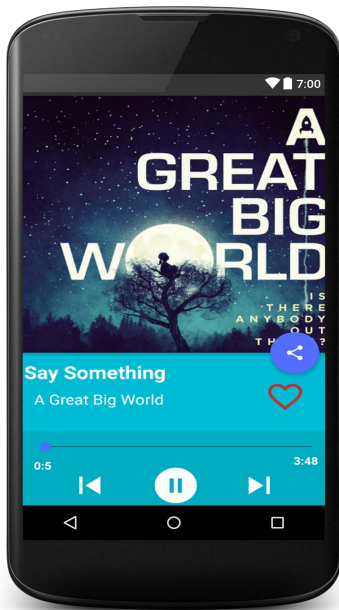
this is the album screen
which view all albums on
the phone

Screen4



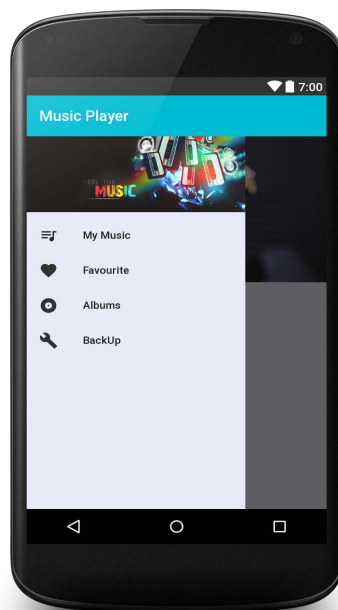
this is the favourite
Screen which view your
favourite music

Screen 5



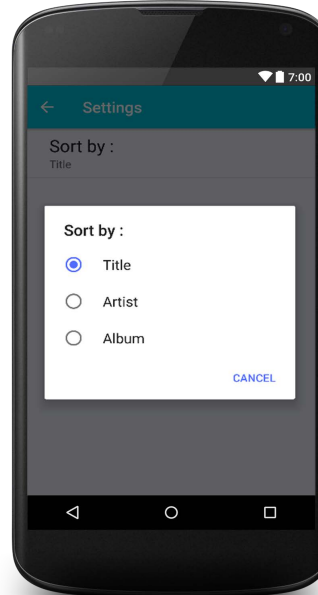
this is the MusicPlayer Screen where the song play

Screen6



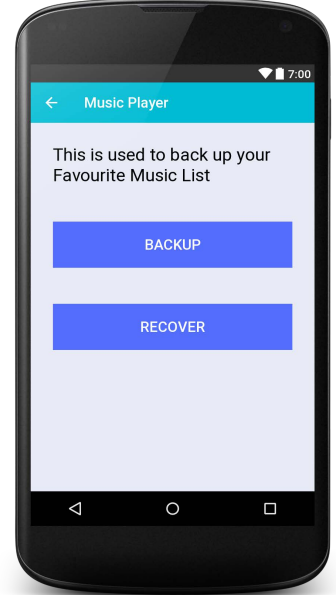
this is the Navigation List Screen To switch between the other screens

Screen7



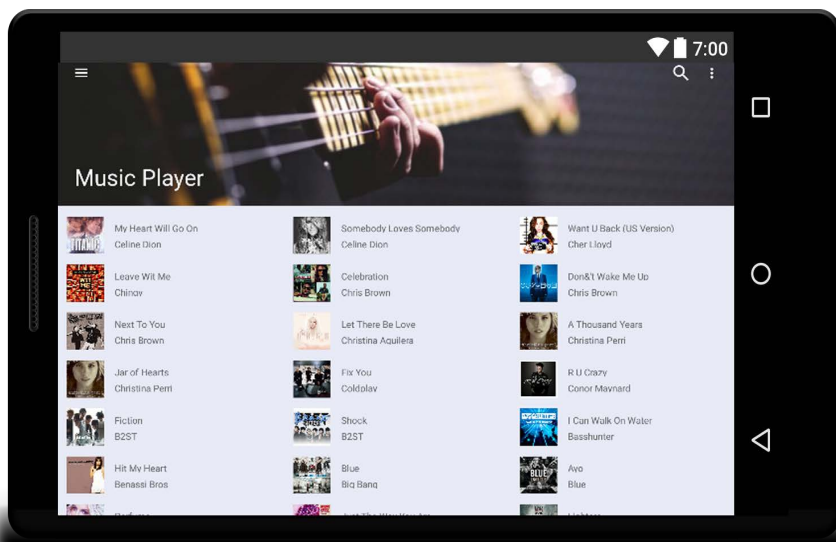
this is Settings Screen Where you select how you want to order Songs

Screen8



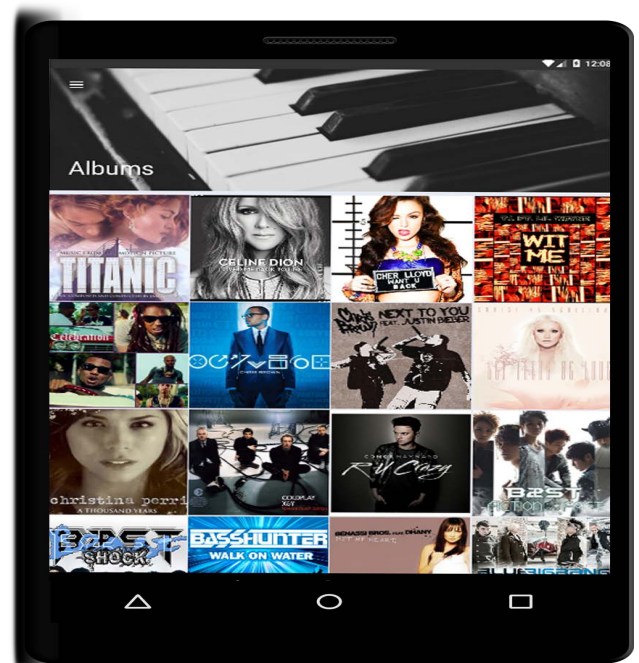
this is Backup Screen Where You select to backup or recover songs

Screen 9



this is the home Screen on the tablet

Screen10



this is Album Screen on the Tablet

Key Considerations

How will your app handle data persistence?

Will Use Content Provider To store The Favourite Music List ,and Preference For the Setting List and SavedInstanceState For MusicPlayer Screen

Describe any corner cases in the UX.

I used Preference To store Now Playing Song ,also Used Empty View To Appear When no music Found Or an Error or No Network for Backup

Describe any libraries you'll be using and share your reasoning for including them.

butterknife:used to call the view

picasso: used to view Song Cover

design:used to design a material design Views

Describe how you will implement Google Play Services.

1-Google Sign in Authentication:used to sign in for the backup

2-admob:used to add ads to the app

3-firebase:used for analysis

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Project Setup

Create Anew Project With API 21 and Navigation Drawable Activity

- Configure libraries e.g:picasso,butterknife,design,preference
- Add internet and storage access permission to the manifest

Task 2: Implement UI for Each Activity and Fragment

Design The UI:

- Build UI for MainActivity
- Build UI for Media Player Activity
- Build UI for Favourite Activity
- Build UI for Album Activity
- Build UI for Back up Activity
- Build UI for Setting Activity and Frament
- Build UI for Widget

Task 3: Implement Functions

implement Function for:

- Media Player
- Loaders
- Async Task
- Adapters
- Back up

Task 4: Data persistence

Work on Data persistence:

- create savedInstanceState for Media Player to Save data when Rotate
- Create Preference for the Settings Activity
- Create Content Provider To Store Favourite List Data and all music List

Task 5: Implement Widget

Implement Widget steps:

- Design Widget Layout
- implement widget functionality
- use Content Provider To retrieve and play Music

Task 6: Implement BackUp Functionality

Steps :

- Connect to DB
- Insert User In dB and Check if User Exsist Or not
- Insert Favourite List Information In the DB
- Retrieve Favourite List Information from the DB

Task 7: Implement Google Services

add the following Google Services :

- AdMob
- Google Sign In Authentication
- Firebase

Task 8: Finishing

Steps :

- Add Empty View For All Activities Which need Empty View when no Data Shown
- Move All Strings and Integers Into String.xml ,Integer.xml
- Translate all Strings to Arabic
- Make Sure That the app accessible for blind and disabled people
- Make Sure That the app Support RTL and LTR
- add High Quality Images and use Vectors
- Handel All Errors and Log Messages
- Inspect and Clean The App

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
3. Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"