[Description](#_sm4ra97uwo11)

[Intended User](#_aws88pzfmqca)

[Features](#_zheq5430xrpq)

[User Interface Mocks](#_giquerrw6g46)

[Screen 1](#_a4jdupabry3k)

[Screen 2](#_dpcbbkx5yry)

[Key Considerations](#_gvcvmae8jn8u)

[How will your app handle data persistence?](#_v8my7nhtvz0m)

[Describe any corner cases in the UX.](#_gw69vjn1ico0)

[Describe any libraries you’ll be using and share your reasoning for including them.](#_6yqqubmw5bs)

[Describe how you will implement Google Play Services.](#_qrxg682nywe6)

[Next Steps: Required Tasks](#_v518bncmggeg)

[Task 1: Project Setup](#_8oe8zpk3qsmp)

[Task 2: Implement UI for Each Activity and Fragment](#_rzllsk6uqztx)

[Task 3: Your Next Task](#_fdmohs7hes)

[Task 4: Your Next Task](#_umfwsvmx7tpn)

[Task 5: Your Next Task](#_kjidlkq4xm3u)

**GitHub Username**: [DomainFlag](https://github.com/DomainFlag)

Jiggles

# Description

Interactive Digital Light-Weight media service where you can discover new music, albums and playlists you will love. Listen to music, share your music and discuss the latest news and releases through forum with people like you that do enjoy listening to music…

# Intended User

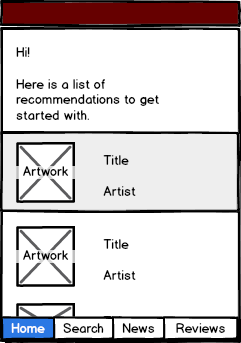
The target group are people that are casual listeners that do like listening to music and are willing to go a step further by interacting with others on forum, share their music and more...

# Features

* Discover songs / albums / artists
* Recommendations (songs / albums / artists)
* New releases (with critic’s review) / news
* Song playback (notification, widget)
* Create own collection
* CRUD threads / comments on forum

# User Interface Mocks

## Screen 1

Landing Home page when app is first started, user is exposed with a list of recommendations that the user can chose which he likes the most.

## Screen 2

# Key Considerations

### How will your app handle data persistence?

As you can see in the repository there are 2 jiggles folders, where there is a jiggles\_end folder which will be used as a remote back-end server for the native app. Thus, the forum(threads) / releases(critics) / news / collections will be preserved on remote MongoDB server, and a Content Provider will be used to cache the latest data. Preferences will be used too, as an example, persisting the type of content which the user wants to see first when opening the app (songs / albums / artist collection or default one the latest releases with critic’s reviews / news).

### Describe any edge or corner cases in the UX.

* Rotations changes, saving the current state of activity, because the user can have bad UX if all the activity done won’t be saved.
* Properly managing the ExoPlayer audio playback state during the activity lifecycle and audio focus for cases like navigation or calling.
* Keeping the user logged in the next time he opens the app by storing the token on local storage.

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

* Picasso will be used to handle the loading and caching of images.
* Use of Spotify Android Playback SDK for audio playback
* Use of Spotify Android Auth SDK for logging to Spotify (obtain the credentials)
* Use of okhttp for managing HTTP responses/requests
* ExoPlayer will be used for loading local audio files and its playback

### Describe how you will implement Google Play Services or other external services.

Application will be using ads (AdMob) which depends on google play services.

# Next Steps: Required Tasks

## Task 1: Project Setup

First, is to configure the libraries used for requesting the web server(jiggles\_end) to authenticate and get the access token for future interactions. From here, with the token provided, I will be able to set up the necessary URIs for fetching data (especially the Spotify content URI) which from there I will be able to use with the Spotify API for audio playback and start building the app.

## Task 2: Implement UI for Each Activity and Fragment

The app will be consisted of MainActivity which will contain

## Task 3: Your Next Task

Describe the next task. For example, “Implement Google Play Services,” or “Handle Error Cases,” or “Create Build Variant.”

Describe the next task. List the subtasks. For example:

* Create layout
* Something else