GitHub Username: joaobiriba

Motivetto

Description

Motivetto is a game for casual-gamers combining music to sliding puzzle game.
Sort the music pieces in the right order and be fast to earn more points.
Share your points in the leaderboard, show your friends how great you are!

Intended User

This is an app for Casual Gamers. People want to play a game with an Android device in small chunks of spare time.

Features

Motivetto main features. For example:

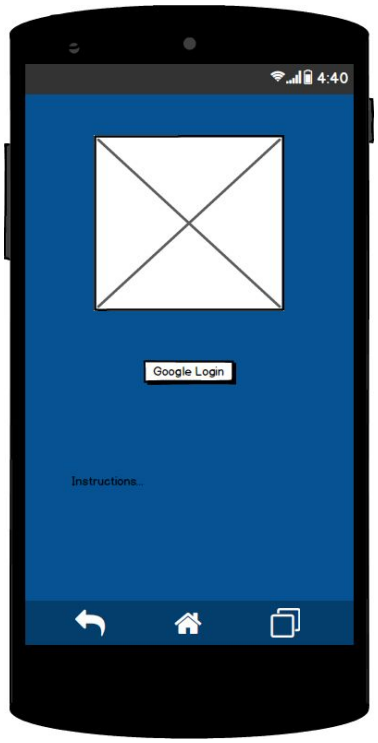
- Log in with your google account
- Retrieve random 10 seconds of music using Spotify API.

- Divide the music preview in 9 parts random distributed.
- Sort the pieces dragging like sliding number games.
- Use google play games api to store points and leaderboard.

User Interface Mocks

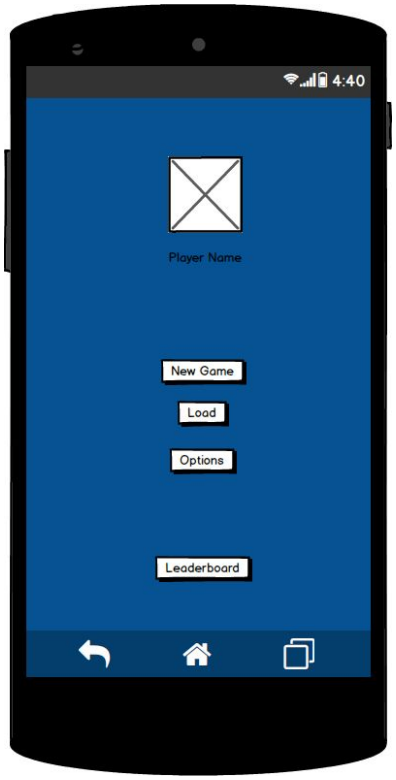
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1



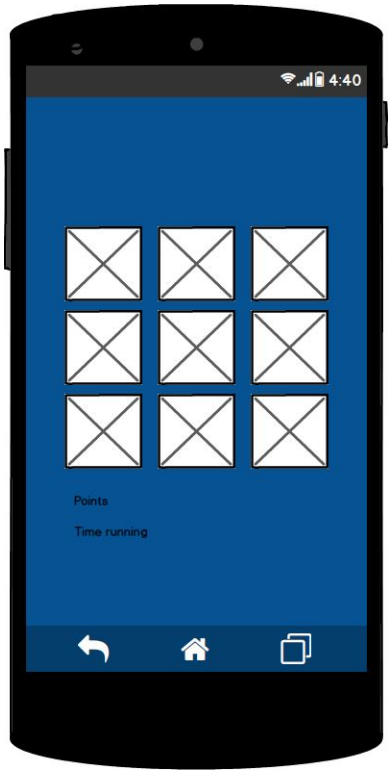
Splash Screen with Login button and instructions

Screen 2



Main menu screen with buttons to start a new game, load a saved game, options and show leaderboard.

Screen 3



Board game screen with 3x3 matrix with “pieces of music” random puzzled. A player have to slide a piece in the empty space to rearrange the right order. When the order is ok the music is played and the game goes to another music. Tap on a piece to play that piece of song.

Key Considerations

How will your app handle data persistence?

Settings are stored in shared preferences.
A content provider to store user profile synced with google play games services backend

Describe any corner cases in the UX.

If the player hit back during a running game the state board and current points are not saved. Points are updated and stored only upon winning a game, sorting right a music.

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

Spotify Web Api Wrapper to query and retrieve random music from spotify.
Google Play Games Services to save points and handle leaderboard.
Glide to load images from spotify albums and others resources.

Next Steps: Required Tasks

Task 1: Project Setup

- Configure google play games services backend and library (https://developers.google.com/games/services/)
- Configure Spotify Web Api Wrapper
- Build a dummy screen to login an user and retrieve a random music from spotify.

Task 2: Implement UI for Each Activity and Fragment

- Build UI for Splash Login Screen
- Build UI for Main Menu Screen
- Build UI for Board Game Screen

Task 3: Implement Main Game Flow

- Develop login of an user and retrieving info.
- Store info of an user
- Develop starting of a game (retrieve a random music and divide it in pieces randomly distributed in the 3x3 matrix)
- Start a timer
- Develop main user gaming interaction (sliding and sorting)
- Develop winning case (next music)
- Develop timer expired case (game over)

Task 4: Options And logout

- Define and implement options to be saved and reused for a player
- Handle a logout from user
- Show leaderboard button.

Task 5: Support all screens

- Support various screens (Tablet UI)