

Beatster



Beschreibung: Beaster ist eine Anwendung, die eine Suchmaschine für Musik ist. Innerhalb der Anwendung erstellt der Benutzer ein Konto, mit dem er auf das Streaming zugreifen kann.

Ähnliche Anwendungen:

Spotify - <https://www.spotify.com/>

Youtube Music - <https://music.youtube.com>

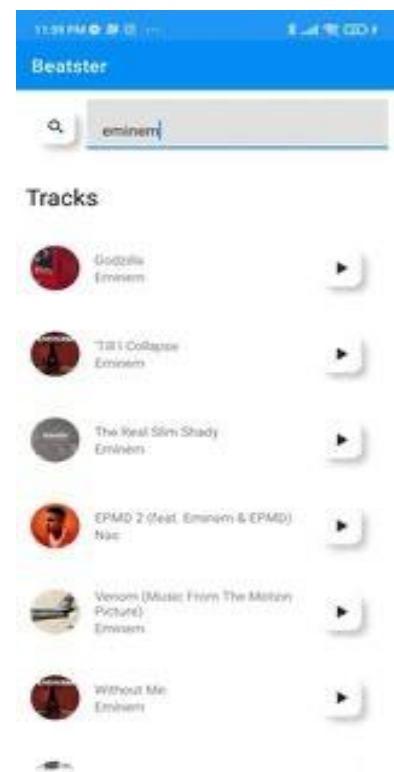
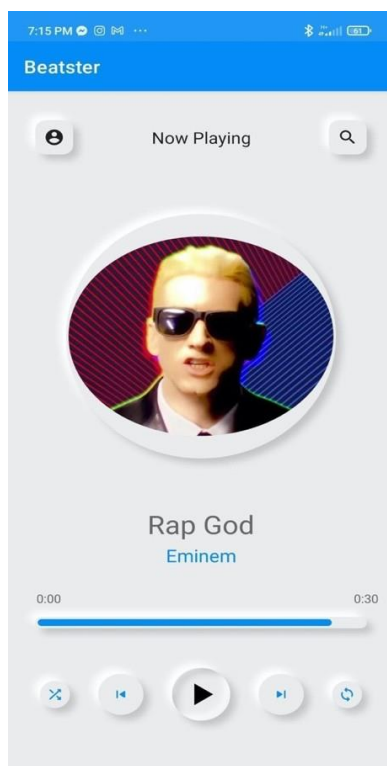
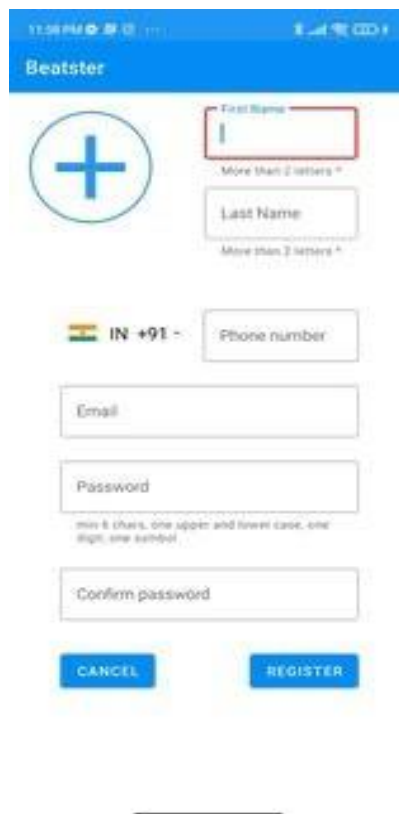
Napster - <https://us.napster.com>

Apple Music - <https://music.apple.com/>

Features :

- Register
- Edit info that will save in shared preferences our data to ease the login procedure
- Login with encrypted data
- Music player
- Read only screen for account data
- Search screen for tracks
- Foreground music notification

GUI:



Zeitplan:

Labor 3-4

- Login screen logic
- Login screen ui flow
- Register screen layout
- Database integration
- CountryPicker implementation
- Main screen layout

Labor 5

- Manage profile image storage
- Password encryption
- View profile functionality
- Search screen layout
- Search functionality

Labor 6

- Music player functionality
- Foreground service for music player
- Use Databinding for Ui elements
- Photo app chooser
- Handle bugs

Labor 7

- Presentation



Extra after Review:

Register screen - at first run (FTUE):

camera permission is requested

read external storage permission is requested

after the permissions are granted, the Register screen is displayed

UI:

Photo holder - when you tap on it a dialog will be displayed asking for two options:

"Take a photo"/ "Choose a photo from gallery"

Take a photo -> will open the camera -> once the photo is taken the camera will close and the register screen will be displayed with the photo set on the photo holder

Choose a photo from gallery -> will open the photo gallery and, after the photo is selected, the register screen will be displayed with the photo set on the photo holder (an Image Picker can be used)

First Name - validate to be greater than 2 letters (will appear as hint)

Last Name - validate to be greater than 2 letters and not have 3 times the same letter (will appear as hint)

Fields with short explanation of these rules below each input field

Country - select the country from a list -> if selected, will autocomplete the phone number code

Phone number - select code by country (a picker for the code from a list with countries) - selecting the code once again will override the autocompleted code

Email - validation to contain the "@" symbol and a "." after the "@" symbol.

Password - password validation rules: minimum length - 6, at least one capital letter, at least one lowercase letter, at least one symbol and at least one digit

Short explanation text below password field with the rules

Confirm password field

Register button - will validate data first and, if all the rules are followed correctly, will save the data and will open the next screen (Search screen)

Cancel button - will clear all the fields

Back button - will clear all the fields and will close the app

LOGIC:

- mandatory input data: first name, last name, email, password, confirm password -> if one of these is not set, the missing data field/fields will be highlighted in a different color and the toast message "Please insert missing data in order to register" will be displayed

- cannot enter the Search screen without registration

- until saving a user in shared preferences the app will always open the register screen and will not go further until registration

- tapping on any field (except pickers) will open the keyboard
- after tapping on Register button a User object, holding all the inserted data, will be created and saved in the shared preferences (it has to persist for the login process)
- when tapping on the Register button, if any of the validations fails, a toast with the message "Please insert data correctly" will be displayed and the wrong data field will be highlighted in a different color (e.g. red)
- registration will not be allowed until all the data passes the validations