

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Main screen](#)

[Detail screen](#)

[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 0: Build web service](#)

[Task 1: Project Setup](#)

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

[Task 3: Create data classes](#)

[Task 4: Change design of the app](#)

[Task 5: Implement Google Play Services](#)

GitHub Username: TomMuehlegger

Fitacity

Description

Use the Fitacity app to train hard in a gym or simply everywhere. Fitacity helps to improve the execution of the different exercises. Depending on the region of your body you can select from different categories and subcategories to choose the best exercise for your needs.

Each exercise detail view contains a description, a how-to video, the difficulty level and the equipment you need. Furthermore you can store your favorite exercises on your phone.

Intended User

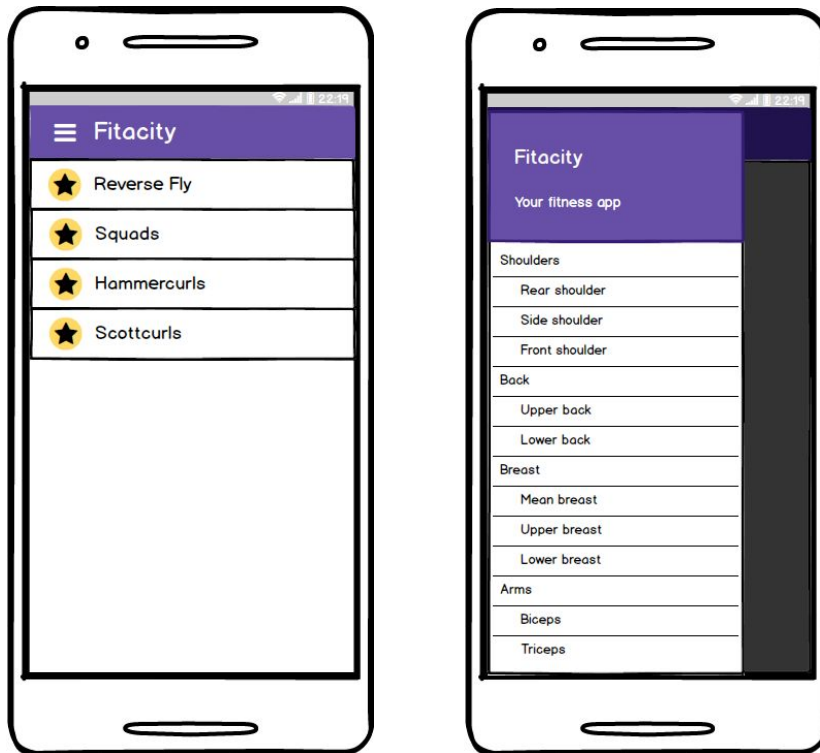
Fitness-conscious people, Health-conscious people, Gym-Freaks, Athletes

Features

- Saves favorite exercises
- Get categories and exercises from remote services
- Plays how-to videos

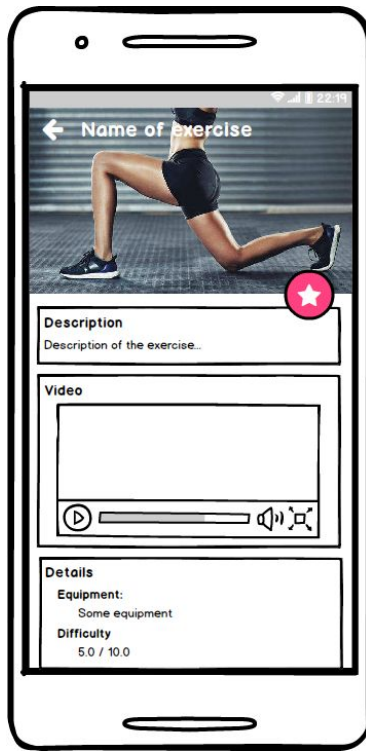
User Interface Mocks

Main screen



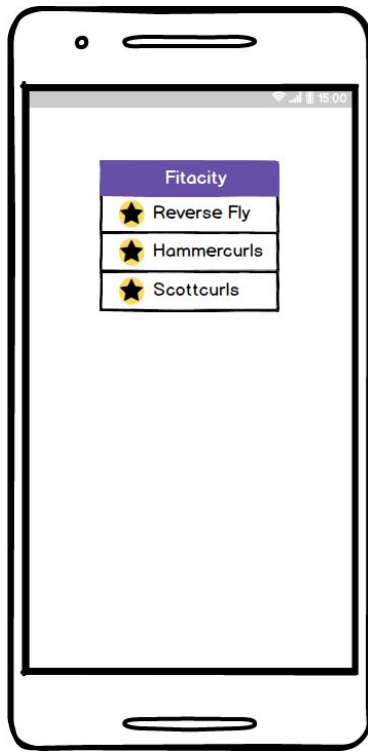
The main screen gives the user an overview of the exercises marked as favorite. Furthermore a navigation menu (with the categories and their subcategories) is shown, when the user clicks on the menu symbol on left upper corner. By selecting a category or a subcategory, all the exercises of the selected item are shown in the list. On selecting a category, all exercises of it's subcategory are displayed.

Detail screen



The detail screen provides several informations about an exercise. It contains a description, a how-to video, the difficulty level and the equipment you need. The FAB on the right lower corner of the toolbar allows the user to add (remove) the exercise to (from) the favorites.

Widget



The widget provides an overview of the user's favorite exercises. If there isn't at least one favorite exercise, "No favorite exercises" is displayed. When clicking a favorite exercise in the widget, this exercise is shown in the app.

Key Considerations

How will your app handle data persistence?

There is a Content Provider to store the user's favorite exercises. Furthermore there is a server side implementation to load the category and exercise data from.

Describe any corner cases in the UX.

The how-to video is displayed directly inside the app (not with an intent) within the YouTubePlayerView of the YouTubeAndroidPlayerApi's.

The app uses a Loader to retrieve the data from the web server and move it to the views. When starting the app, all categories (and subcategories) are loaded from the server. On selecting a category, the exercises of this category are loaded and displayed on the list view.

Another feature of the app is the widget, which provides access to the favorite exercises.

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

The **Glide** library should be used to load images (from URL) to a Collapsing Toolbar of the detail view for exercises.

The **YoutubeAndroidPlayerAPI** library should be used to play the how-to videos on the detail screen for exercises directly in the app.

Describe how you will implement Google Play Services.

Play-services-ads: To display a test ad on the main screen

Play-services-analytics: To analyze the popularity of the exercises (track adding and removing them to/from the favorites)

Next Steps: Required Tasks

Task 0: Build web service

- Setup MySQL database
- Insert data → categories and exercises
- Create PHP code to retrieve data from DB and response as JSON

Task 1: Project Setup

- Configure libraries (Download, Import, Gradle)
- Set the latest gradle versions

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity (layout file and java class)
- Build UI for ExerciseDetailActivity (layout file and java class)
- Build class for the ExerciseRecyclerViewAdapter
- Build UI for the list item of the ExerciseRecyclerView
- Build UI for the main menu navigation header

Task 3: Create data classes

- Create (parcelable) class for category
- Create (parcelable) class for exercise
- Create classes for the Content Provider
 - Contract class - specifies the columns and URIs of the ContentProvider
 - DBHelper class - to create and update the SQLite database
 - ContentProvider class - to retrieve and store data
- Create utils class
 - JSON converter class - to convert JSON strings to data classes
 - Network class - to handle API calls

Task 4: Change design of the app

- Add dimensions
- Add styles
- Change colors

Task 5: Implement Google Play Services

- Create an AdView to display a test ad
- Add Google Analytics to the app to analyze the popularity of the exercises