

**GitHub Username:** RazanRajab

# Female Fitness App

## Description

Sometimes you be so lazy and don't want to go to the gym, or there is no gym in your region. But now there is no Gym required, my app helps you to keep fit and weight loss at home, it provides a women workouts for female fitness.

## Intended User

Any woman who wants to keep her body fit.

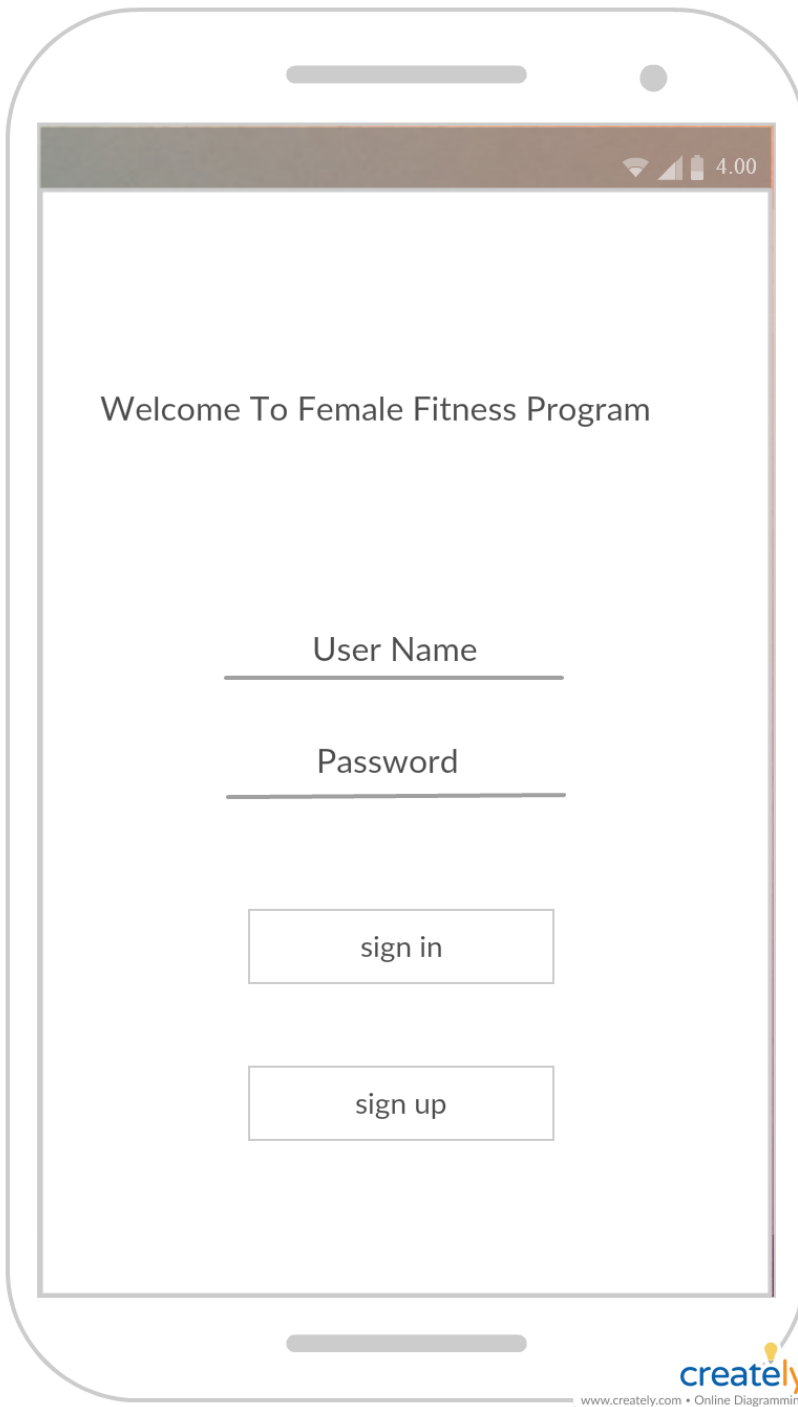
## Features

- Saves user information.
- Displays List of exercises.
- Displays video for each exercise.
- Lets you save your favorite exercises.
- Displays favorite exercise.
- Has a Widget on homescreen to display your favorite exercises list.

## 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 Google Drawings, [www.ninjamock.com](http://www.ninjamock.com), Paper by 53, Photoshop or Balsamiq.

## Sign in or Sign up activity



A mobile application interface for a fitness program. The screen is titled "Welcome To Female Fitness Program". It features two input fields: "User Name" and "Password", each with a horizontal line below the text. Below the password field are two buttons: "sign in" and "sign up". The interface is enclosed in a rounded rectangle with a grey border. At the top, there is a status bar with a brown background, showing a Wi-Fi icon, a signal strength icon, a battery icon, and the number "4.00". At the bottom, there is a grey bar with a small "createely" logo and the text "www.createely.com • Online Diagramming".

Welcome To Female Fitness Program

User Name

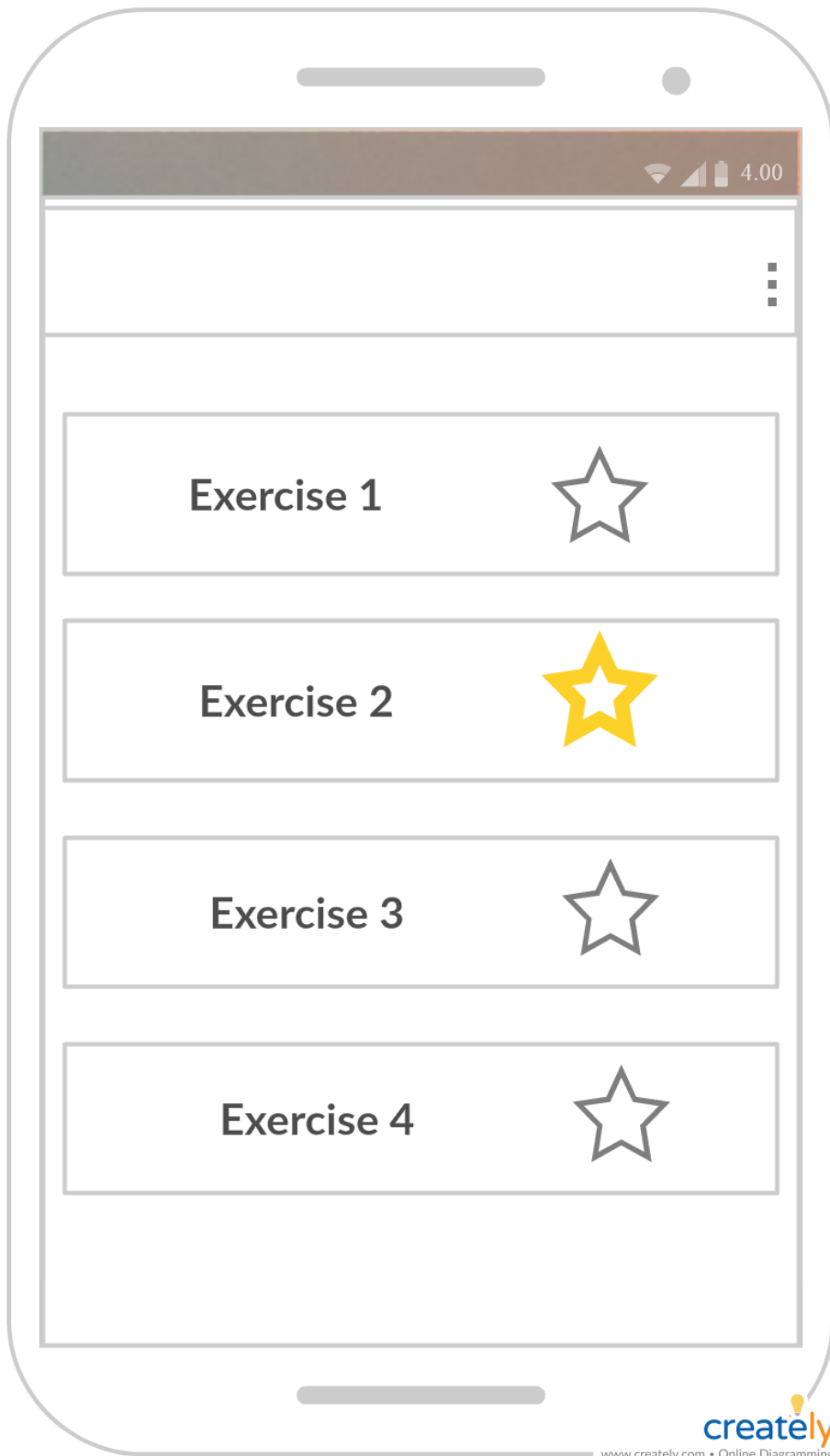
Password

sign in

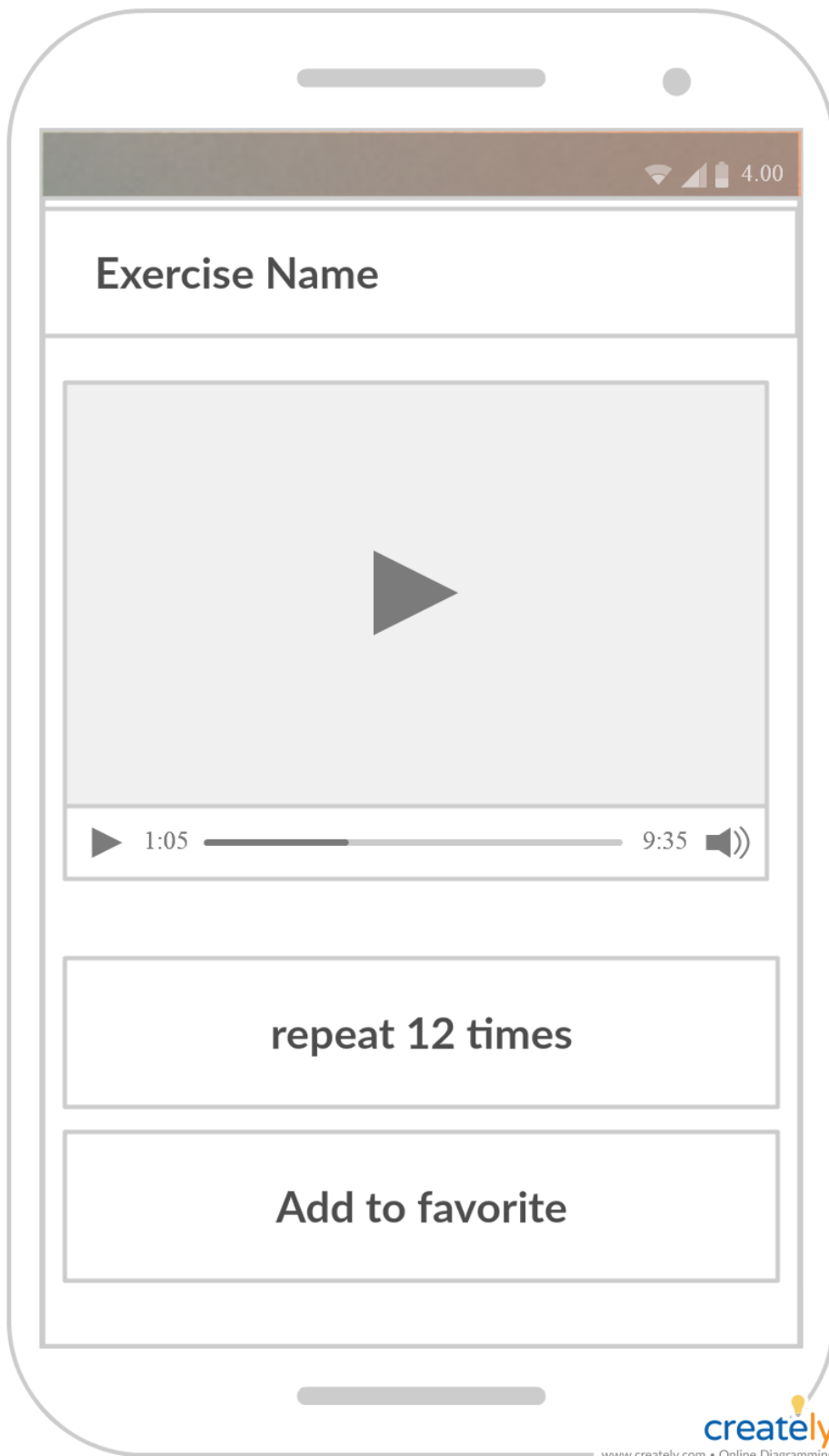
sign up

createely  
www.createely.com • Online Diagramming

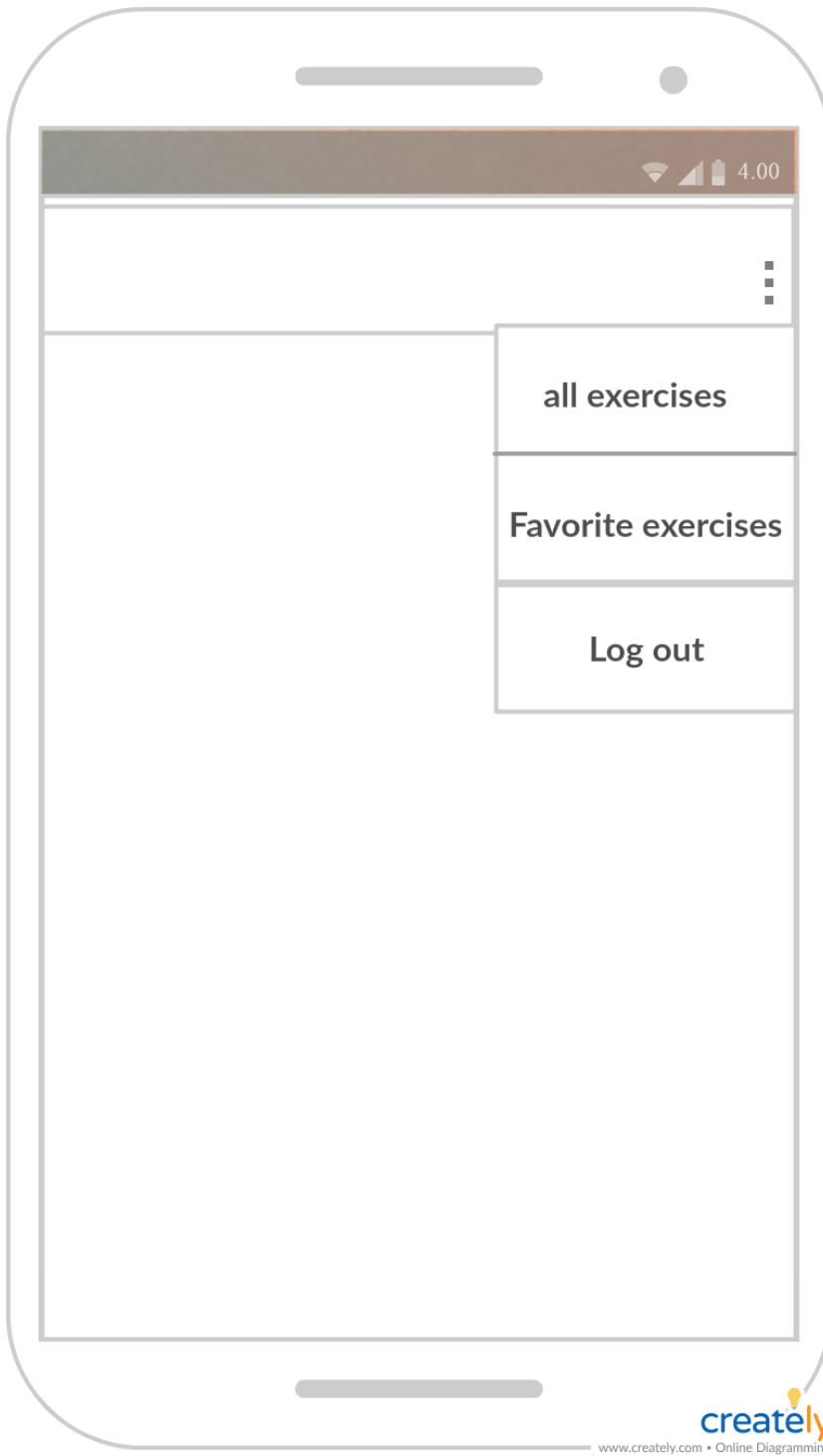
## Main Activity



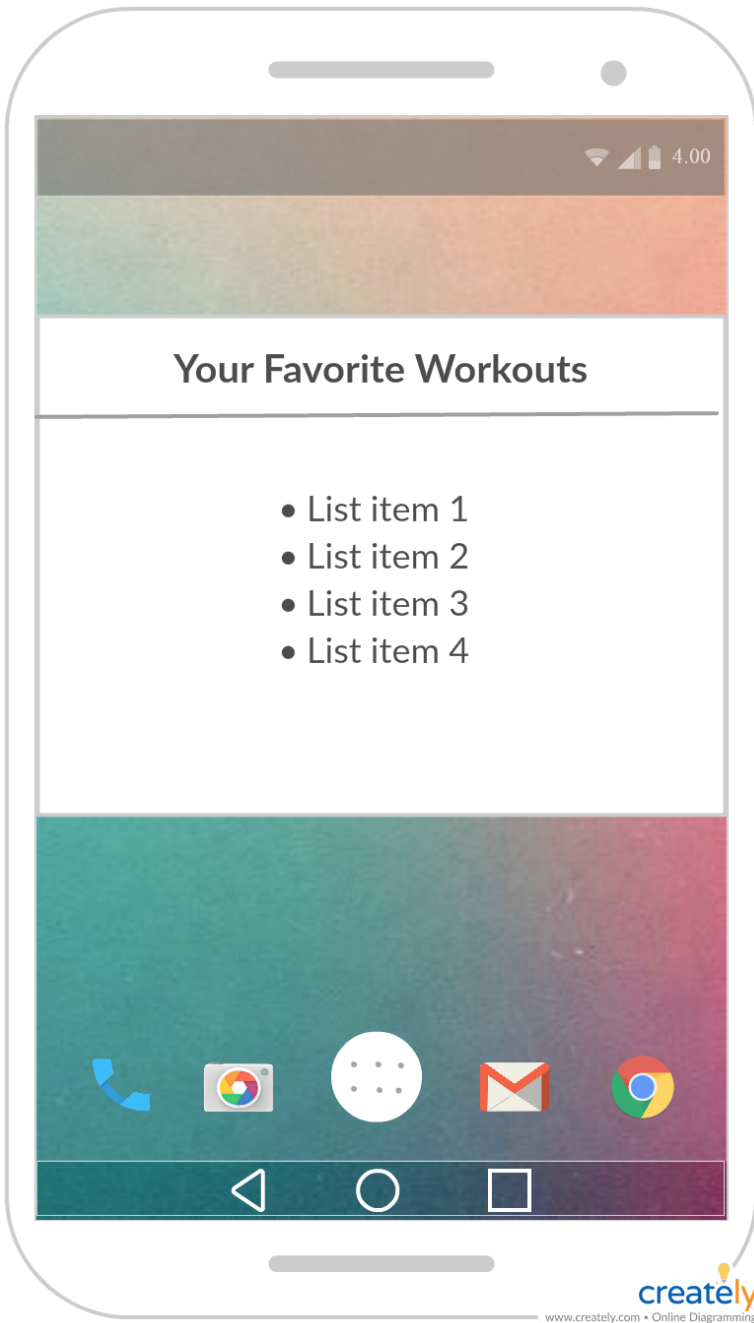
## Exercise Activity



## Menu and menu items



## Widget on homescreen



## Key Considerations

### How will your app handle data persistence?

- I will use the Firebase Cloud Firestore to save the static data of the exercises manually then make the app read this data to display the exercises.
- I will use the Firebase Auth to handle the user information.
- I will use the Room, Live Data and View Model to handle the favorite exercises data.

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

When the mainActivity is launched it will display all the exercises, and if the user want to display his favorite exercises he can do it by pressing on the menu item “favorite exercises”, And he can go back to all the exercises by pressing on the menu item “all exercises”.

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

ButterKnife library : I used this third-party library to annotate fields with @BindView and a view ID for Butter Knife to find and automatically cast the corresponding view in the layout.

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

I will read the exercises data from cloud Firestore  
I will use Firebase Auth to store user information

## Next Steps: Required Tasks

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

### Task 1: Project Setup

- Use Android studio 3.5 and gradle plugin 3.5.0
- Create new project
- Configure libraries and dependencies
- Configure permissions in manifest
- Configure values of colors in colors.xml
- Connect the app with Firebase

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

- Build UI for SignInActivity
- Build UI for the MainActivity
- Build UI for the ExerciseActivity
- Build UI for the menu
- Build UI for the widget

### Task 3: Implement the sign in and sign up functionality

- Implement the sign up method.
- Implement the sign in method.

### Task 4: Implement the MainActivity functionalities

- Create the model class(Exercise).
- Implement class to fetch the exercises data from firebase with help of AsyncTask.
- Implement displaying the exercises functionality in Recycler View.



### **Task 5: Implement the ExerciseActivity functionalities**

- Initialize the exoplayer
- Implement displaying the exercise data
- Prepare the database using Room(Entity and Dao)
- Implement add to favorite functionality

### **Task 6: Implement the menu items functionalities**

- Implement displaying the favorite exercises with help of Live data and View model
- Implement the functionality of logout menu item.
- Implement the functionality of all Exercises menu item.

### **Task 7: Implement the widget functionalities**

- Implement the Favorite Exercises Widget Provider
- Implement displaying the exercises with help of Remote view Service

### **Task 8: Implement some test cases using espresso.**

### **Task 9: check if the app meet all the specifications include the following:**

- App is written solely in the Java Programming Language
- App utilizes stable release versions of all libraries, Gradle, and Android Studio.
- App validates all input from servers and users. If data does not exist or is in the wrong format, the app logs this fact and does not crash.
- App includes support for accessibility. That includes content descriptions, navigation using a D-pad, and, if applicable, non-audio versions of audio cues.
- App keeps all strings in a strings.xml file and enables RTL layout switching on all layouts.