

## How to Use this Template

1. Create a new document, and copy and paste the text from this template into your new document [ Select All → Copy → Paste into new document ]
  2. Name your document file: “**Capstone\_Stage1**”
  3. Replace the text in green
- 

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[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 1: Project Setup](#)

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

[Task 3: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

**GitHub Username:** Mukuljangir372

# Home Workout

## Description

Home Workout is an ad free fitness app that keeps you motivated and inspired all the day by giving you workout videos and fitness pro tips. It helps you in Home workout at home with no equipment, all exercise performed by body weight.

This app provides Full body workout with lower and upper body workout. It also provides you Abs, legs, arms workout without any equipment. Just follow videos and you are done.

This app provides professional and homemade videos that helps you in home workout with no equipment with full inspiration from pro tips. Stick with home workout and you must notice a change in your body.

You can also access your favorite videos in your favorite section. Just click on favorite button to move your favorite video in your favorite section. Always keep updated with workout videos and keep working hard to get fit.

## Intended User

This app is for every age group who is passionate about fitness or workout goals.

## Features

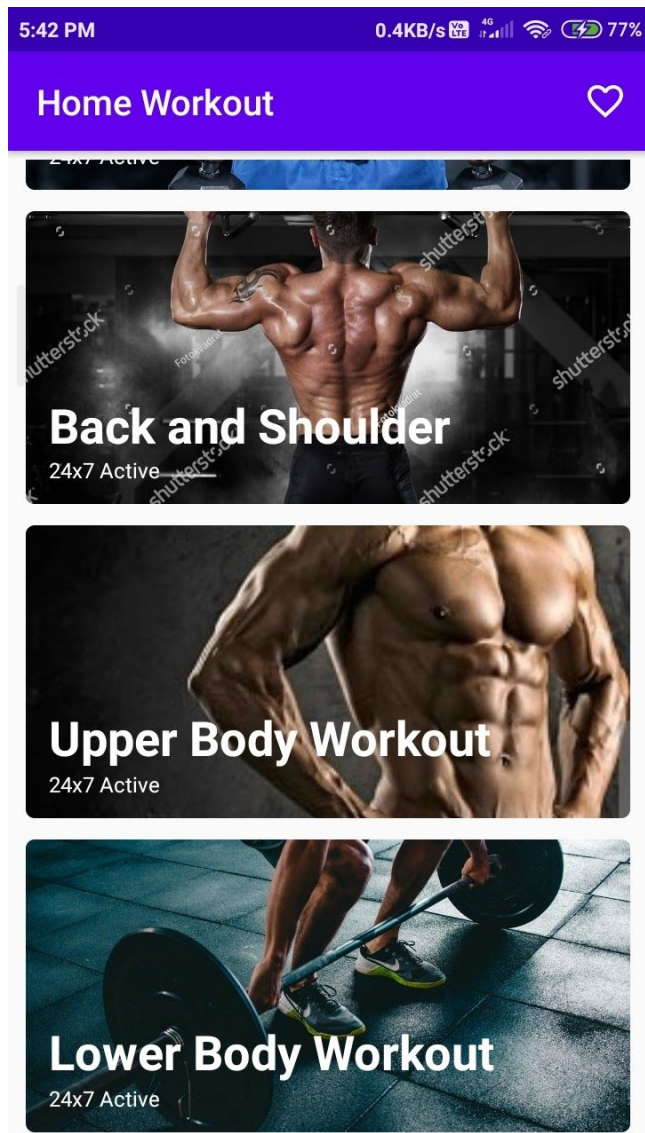
Main features of app:

- Workout videos
- Favorite workout videos
- Always keep updated with workout videos

## User Interface Mocks

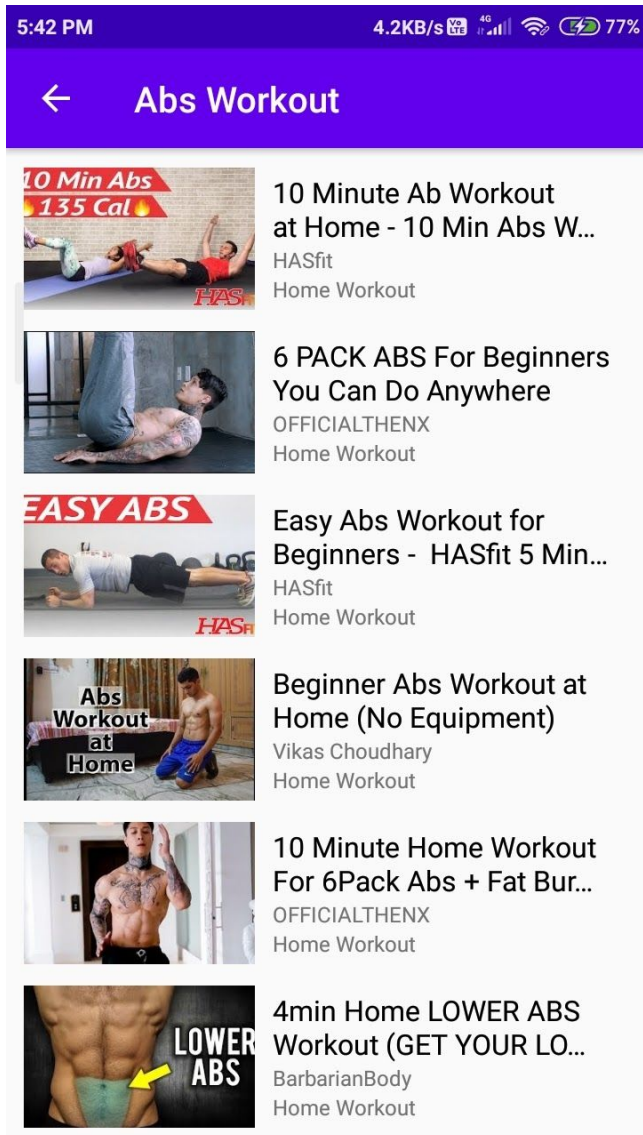
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Google Drawings, [www.ninjamock.com](http://www.ninjamock.com), Paper by 53, Photoshop or Balsamiq.

## Screen 1



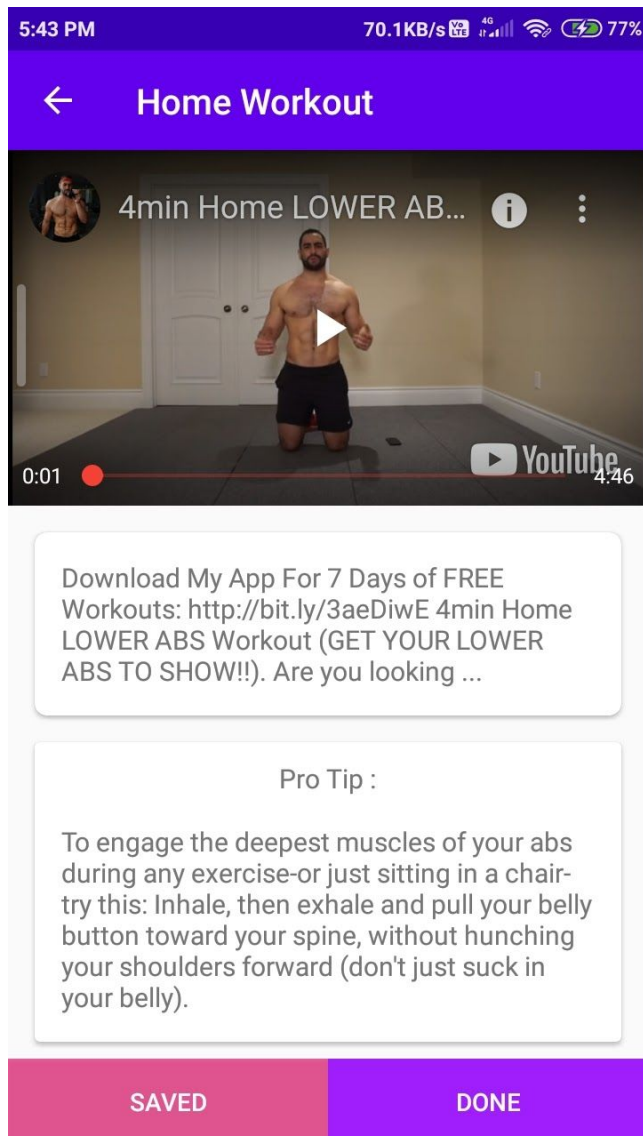
This is the main screen of the app which provides you different home workout options like abs workout, legs and upper body workout and so on.

## Screen 2



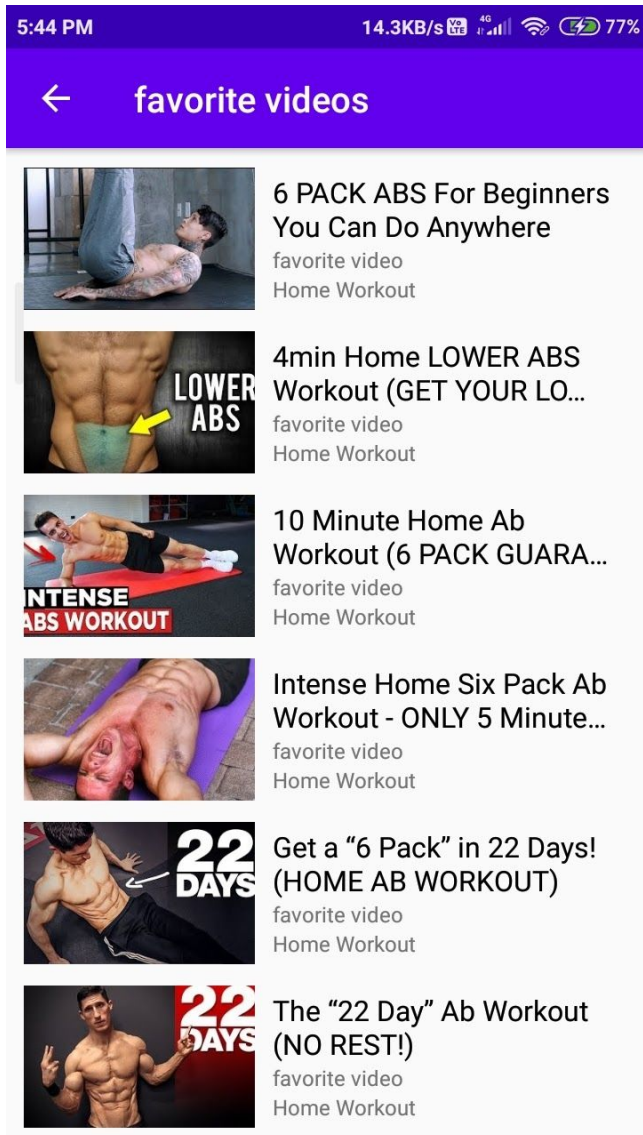
This is the second screen of the app which provides you daily updated professional workout videos or home made workout videos.

### Screen 3 :



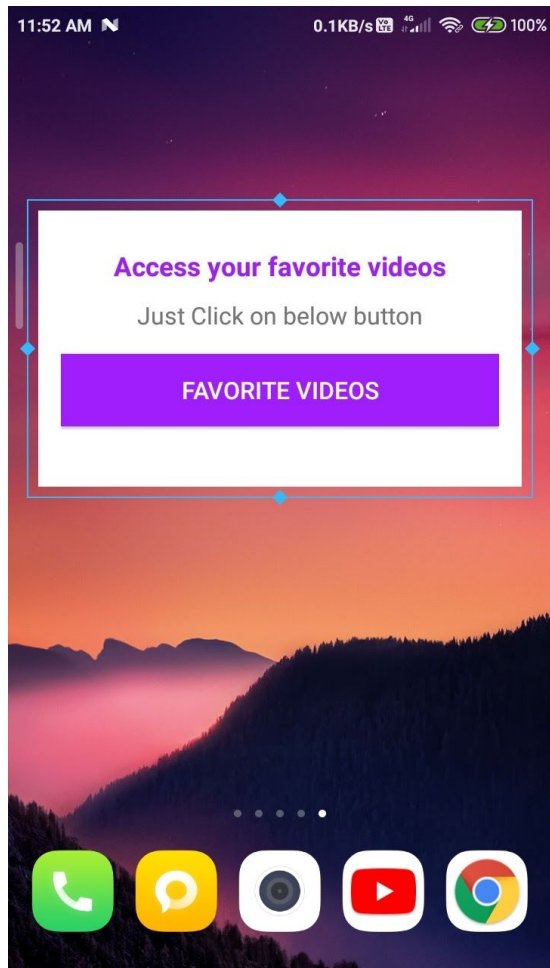
This is the third screen of the app which helps you in workout with playing workout videos with different pro tips. Just click on the done button when your workout is done and click on the save button to save your favorite video and watch later anywhere as you want.

## Screen 4



Access your favorite workout video anywhere anytime as you want in your favorite video section.

## Screen 5



This is the Last Screen of the app. As you know, this screen provided a home screen widget for users to interact with their favorite workout videos.

## Key Considerations

1. Using Android studio with all updated gradle files and dependencies
2. Written in java and xml only
3. App using youtube api and youtube player view for playing videos
4. App works fine in different devices (Test by firebase Test lab)

### How will your app handle data persistence?

For data persistence in my app, I'm using Room, live data and view model to save and retrieve favorite workout videos for user easily and faster.

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

This app uses a youtube player for playing videos using a youtube video url. Also provides a beautiful and simple ui.

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

Using a lots of libraries and tool that are mentioned below : (Note : Android studio and application related information also mentioned below)

Names :	Versions:
Compile SDK Version	29
Build Tools Version	29.0.2
Min SDK Version	23
Target SDK Version	29
Gradle Version	5.6.4
Gradle Plugin Version	3.6.1
Android Studio	3.6.2
Glide	4.11.0



Dagger 2	2.20
Room	2.2.5
RxAndroid	2.0.1
Event Bus	3.2.0
Retrofit	2.1.0
Youtube player view	10.0.5
ButterKnife	10.2.1

Libraries and its usage :

1. Glide for faster loading images
2. EventBus for pusing events and receiving data using container
3. Retrofit + Rxjava for api calls
4. Dagger 2 for DI means dependency injection
5. YoutubePlayerView for playing videos
6. ButterKnife for views binding

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

To monetize my app, I'm using ads in my app that uses google play services.

## 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

Simply using Android studio created a new project and updated all gradle files and libraries. Then Import all libraries are used in this project and start building with android studio in java and xml.

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

- Implement Ui for Main Activity :
  1. Use RecyclerView to load workout list using adapter
  2. Made all required layouts using xml
- Implement Ui for WorkoutListActivity:
  1. Use youtube api for retrieve all videos and load into recyclerview using adapter
  2. Made all required layouts using xml
- Implement Ui for WorkoutDetailedActivity:
  1. Use Youtube player view for playing youtube videos using video id
  2. Use Done button view for when workout is done by user
  3. Use Save button view for saving favorite videos in favorite video section

Use a lot of layouts for loading videos or lists that are used by adapters of recyclerView.

## Task 3: Time to Create Room Database

- Now, It's time for the Room database : Just simply make a database and access your favorite workout videos anywhere anytime as you want.
  1. Make a database class that extends RoomDatabase
  2. Make Entity for the room database in a class and annotate with @Entity. Entity helps you to arrange data in tabular form.
  3. Make Dao for the database and annotate with @Dao. This provides you methods with annotations that help you to save, delete,update and retrieve data for your app.
  4. Just Init your Activity with this database and you are ready for arranging your data in the room database.

- Complete database

## **Task 4: Network calls**

- Making Api calls like getting video using youtube api with the help of retrofit and Rxandroid
- Make an interface that handle api calls
- Implement interface with retrofit
- Get result from youtube api using Rxandroid
- Successfully Make a list and load to recycler View using an adapter

## **Task 5: Adding Home Screen Widget**

Now It's time for making a home screen widget for user that can easily find favorite workout videos.

1. Make Xml file for home screen widget provider
2. Add a beautiful Layout to home screen widget
3. Also make a class that extends AppWidgetProvider that provided you to create, update and delete widget on homescreen
4. Meantime, you must define receiver and meta data for home screen widget in Manifest file
5. Completed

## **Task 6: Debugging and Testing**

Now It's time to finding some bugs using debugging and testing app

- Testing application using firebase test lab
- Use debugging for finding bugs and fix them

Debugging :

1. Use debugging Breakpoints to check java code lines in android studio
2. Debug and start building gradle with all files
3. Install apk and it's time to checking all breakpoints
4. Hopefully, Find it and fix it

Testing :

1. Test apk using android studio logcat
2. Test apk on different devices using firebase test lab

---

### Submission Instructions

- After you've completed all the sections, download this document as a PDF [ File → Download as PDF ]
  - Make sure the PDF is named "**Capstone\_Stage1.pdf**"
- Submit the PDF as a zip or in a GitHub project repo using the project submission portal

If using GitHub:

- Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
- Add this document to your repo. Make sure it's named "**Capstone\_Stage1.pdf**"