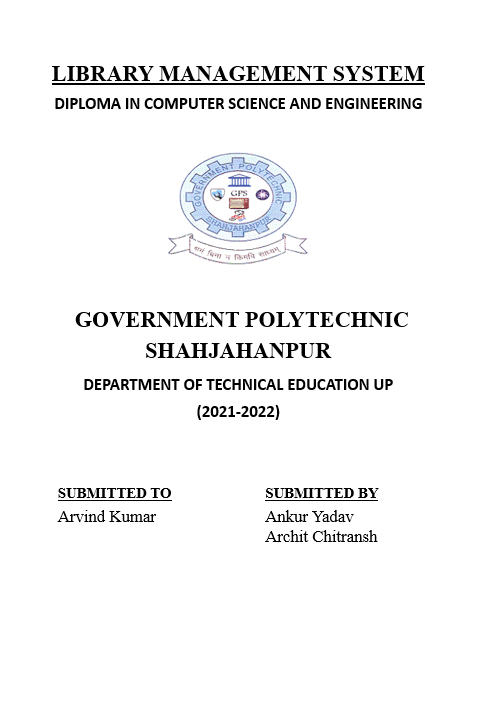
**FIT FREAK**

**PROJECT REPORT**

**FOR**

**DIPLOMA**

****

**COMPUTER SCIENCE AND ENGINEERING**

**GOVERNMENT POLYTECHNIC**

**SHAHJAHANPUR**

**DEPARTMENT OF TECHNICAL EDUCATION UP**

**(2021-2022)**

**Submitted To Submitted By**

**Mr. Arvind Kumar Ankur Yadav**

**(E20222038900001)**

**Archit Chitransh**

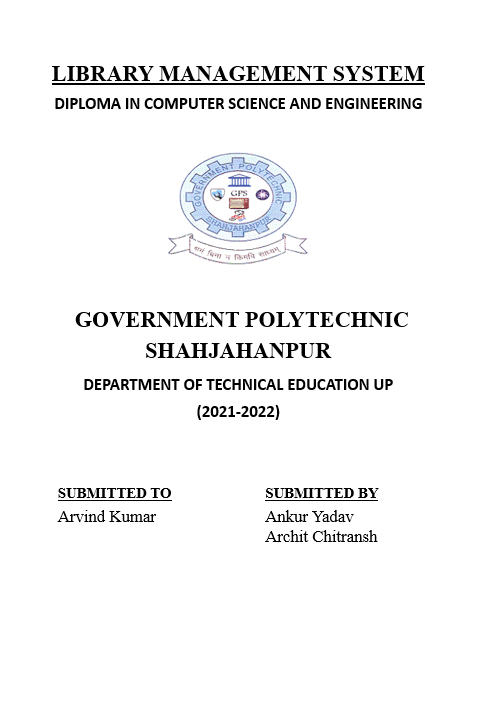
**(E20222038900002)**

**FIT FREAK**

**PROJECT REPORT**

**FOR**

**DIPLOMA**

****

**COMPUTER SCIENCE AND ENGINEERING**

**GOVERNMENT POLYTECHNIC**

**SHAHJAHANPUR**

**DEPARTMENT OF TECHNICAL EDUCATION**

**U.P.**

**(2021-2022)**

PROJECT INCHARGE HEAD OF DEPARTMENT

**DECLARATION**

We hereby declare that the project Report **“Fit freak”** is our own work and effort and that it has not been submitted anywhere for any award. The text embodied in this report has not been submitted to any other university or Institute for the award of any degree or diploma.

**Date: ANKUR YADAV**

**ARCHIT CHITRANSH**

**CERTIFICATE**

This is to certify that the project report entitled **“FIT FREAK”** that is being submitted by **Ankur Yadav** and **Archit Chitransh** in fulfilment for the award of Diploma in Computer Science & Engineering Department of **Government Polytechnic, Shahjahanpur** is a record of bonafide work carried out by them under my guidance and supervision.

The text embodied in this report has not been submitted to any other university or Institute for the award of any degree or diploma.

**Date:**

**Project Guide**

**Mr. Arvind Kumar**

**ACKNOWLEDGMENT**

We give our special thanks to **Mr. Arvind Kumar** CSE Department without his guidance this project work would not able to complete successfully. We are indebted for his continuous help and supports.

We are thankful to management and administration.

Last but not the least we thank our parents and God almighty.

Ankur Yadav(E20222038900001)

Archit Chitransh(E20222038900002)

**LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| **TERMS** | **MEANING** |
| API | Application Programming  Interface |
| XML | Extensible Markup Language |
| HTTP | Hyper Text Transfer Protocol |
| JSON | JavaScript Object Notations |
| IDE | Integrated Development Environment |
| GUI | Graphical User Interface |
| URL | Uniform Resource Locator |
| FTP | File Transfer Protocol |
| DFD | Data Flow Diagram |
| ER | Entity Relationship |
| FRS | Functional Requirement Specification |
| SRS | System Requirement Specification |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Figure Name** | **Page No.** |
| 1. | Level 0 DFD | 12 |
| 2. | Level 1 DFD | 12 |
| 3. | JAVA Diagram | 3 |
| 4. | XML Diagram | 4 |
| 5. | Gradle Diagram | 4 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Table Name** | **Page No.** |
| 1. | HARDWARE REQUIREMENTS | 7 |
| 2. | SOFTWARE REQUIREMENTS | 8 |

**LIST OF SNAPSHOTS**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Snapshot Name** | **Page No.** |
|  | **SCREENSHOTS** | 16 |
| 1. | OPENING PAGE | 16 |
| 2. | HOMEPAGE | 16 |
| 3. | EXERCISE LIST | 17 |
| 4. | EXERCISE BRIEF WITH TIMER | 17 |
| 5. | OTHER OPTIONS | 18 |
|  | **DESIGNS WITH THEIR CODE** | 19 |
| 6. | OPENING PAGE | 19 |
| 7. | HOMEPAGE | 20 |
| 8. | EXERCISE LIST | 21 |
| 9. | EXERCISE WITH STOPWATCH | 22 |
| 10. | OTHER OPTIONS | 23 |

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| **CHAPTERS** | **PAGE NO.** |
| **DECLARATION** | **iii** |
| **CERTIFICATE** | **iv** |
| **ACKNOWLEDGEMENT** | **v** |
| **LIST OF ABBREVATIONS** | **vi** |
| **LIST OF FIGURES** | **vii** |
| **LIST OF TABLES** | **viii** |
| **LIST OF SNAPSHOTS** | **ix** |
| **ABSTRACT** | **0** |
| **CHAPTER 1: INTRODUCTION** | **1** |
| 1.1 FUNCTIONALITIES | 1 |
| **CHAPTER 2: LITERATURE REVIEW** | **2** |
| **CHAPTER 3: TECHNOLOGY SPECIFICATION** | **3-6** |
| 3.1 FRONT END | 3 |
| 3.1.1 JAVA | 3 |
| 3.1.2 XML | 3 |
| 3.2.3 GRADLE | 4 |
| 3.2 TOOLS USED | 5 |
| 3.2.1 ANDROID STUDIO | 5 |
| 3.2.2 FIREBASE | 5 |
| 3.2.3 IntelliJ IDEA | 6 |
|  |  |

|  |  |
| --- | --- |
|  |  |
| **CHAPTER 4: SRS** | **7-8** |
| 4.1 FUNCTIONAL REQUIREMENT | 7 |
| 4.2 NON-FUNCTIONAL REQUIREMENTS | 7 |
| 4.3 HARDWARE REQUIREMENTS | 7 |
| 4.4 SOFTWARE REQUIREMENTS | 8 |
| 4.5 SRS AND METHODOLOGY | 8 |
| 4.5.1 MODEL USED | 8 |
| **CHAPTER 5: FEASIBILITY STUDY** | **9-10** |
| 5.1 OPERATIONAL FEASIBILITY | 9 |
| 5.2 ECONOMIC FEASIBILITY | 10 |
| 5.3 TECHNICAL FEASIBILITY | 10 |

|  |  |
| --- | --- |
| **CHAPTER 6: MODULE DESCRIPTION** | **11** |
| 6.1 TEST MODULE | 11 |
| 6.2 GRADLE MODULE | 11 |
| 6.3 TIME MODULE | 11 |
| **CHAPTER 7: SYSTEM DESIGN** | **12** |
| 7.1 DFD | 12 |
| 7.1.1 LEVEL 0 DFD | 12 |
| 7.1.2 LEVEL1 DFD | 12 |
| **CHAPTER 8: TESTING** | **13-14** |
| 8.1 INTRODUCTION TO TESTING | 13 |
| 8.2 LEVEL OF SOFTWARE TESTING | 13 |
| 8.2.1 UNIT TESTING | 13 |
| 8.2.2 SYSTEM TESTING | 13 |
| 8.2.3 INTEGRATION TESTING | 14 |
| 8.3 TESTING PROCESS | 14 |
|  |  |
| **CHAPTER 9: SYSTEM ANALYSIS** | **15** |
| 9.1 OVERVIEW | 15 |
| 9.2 IMPLEMENTATION | 15 |
| 9.3 OPERATION MAINTENANCE | 15 |
| **CHAPTER 10: SNAPSHOTS** | **16-18** |
| 10.1 SCREENSHOTS | 16 |
| 10.1.1 OPENING PAGE | 16 |
| 10.1.2 HOMEPAGE | 16 |
| 10.1.3 EXERCISE LIST | 17 |
| 10.1.4 EXERCISE BRIEF WITH TIMER | 17 |
| 10.1.5 OTHER OPTIONS | 18 |
| 10.2 DESIGNS WITH THEIR CODE | 19 |
| 10.2.1 OPENING PAGE | 19 |
| 10.2.2 HOMEPAGE | 20 |
| 10.2.3 EXERCISE LIST | 21 |
| 10.2.4 EXERCISE WITH STOPWATCH | 22 |
| 10.2.5 OTHER OPTIONS | 23 |
| **CHAPTER 11: CONCLUSION** | **24** |
| **CHAPTER 12: FUTURE SCOPE** | **25** |
| **REFERNCES** | **26** |

**ABSTRACT**

The fitness application system intend is to automate the existing manual system by the help of computerized apparatus and full-fledged computer software, fulfilling their requirement, so that the user can easily configure what exercise to perform under his age group. This application creates an individual efficient fitness plan based on your age group. This app is easy to use and ensures the best possible result.

The user has been wasting his / her time and the money to search for gym trainer.

But by using this application the user can be his own trainer.

Fitness application, as report above can lead to error free, secure, well founded and fast executives’ system. It can help the user to concentrate on their other activities preferably to concentrate on the better exercises for them. Thus it will help organization in better utilization of resources. That means that one need not to be distracted by information that is not relevant, while being able to reach the information.

Chapter 1

INTRODUCTION

The main objective of the Fitness Application is developed to override the problems prevailing in the practicing manual system and searching for fitness trainers. This application is providing a system which is use to maintaining their health care. It take care of all their fitness and also provides different kinds of exercises for all the age groups. The purpose of the project is to build an android application to reduce the time and money. We can easily get the list of all exercise and also the time of completion of each exercise and how to do it. It can assist the user to concentrate on their fitness.

The project is completely built at administrative end and thus only the administrator is guaranteed to access. The user can only look up for the content he/she needs under their age group.

To understanding the requirements and new generation capabilities we have chosen “Android Studio” as front –end and “Visual Studio Code” for the coding part. Android Studio is the official combined development environment for Google’s Android operating system, built on JetBrains IntelliJ IDEA software and designed exactly for Android development. It is available for download on Windows, macOS and Linux based operating systems.

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity)

* 1. **FUNCTIONALITIES:**

**A Personalized experience:** - The android application provides a friendly and easy user interface in which user can easily identify different types of exercises he needs to perform for his/her age group.

**Categorized Exercises: -** Exercises are categorized under different age group.

**Time Period for each exercise: -** There is a certain time defined for each exercise so that user don’t over perform any exercise and mess up their routine.

**Chapter 2**

**LITERATURE REVIEW**

When it comes to exercise equipment, the most valuable piece of gear might be right in your back pocket. Today, there a ton of applications devoted to all kinds of workouts, from yoga and Pilates to HIIT and strength training sessions. Whether you're new to fitness and are looking for something to help get you started, or are someone who just needs a little more accountability, these workout apps will help you meet all of your health and fitness goals this year.

A fitness app is an application that can be downloaded on any mobile device and used anywhere to get fit. As of 2015, the number of health-related apps released on the two leading platforms, iPhone operating system (iOS) and Android, had reached more than 165,000. Apps can perform various functions to facilitate healthy behaviour change. They can be used as a platform to promote healthy behaviour change with personalized workouts, fitness advice and nutrition plans.

One of the most important benefits of using a fitness app is 'motivation'. Notification and reminders from fitness apps keep reminding you about your health goals, thus keeping you motivated.

You may also come across your fitness app various times in a day while using your smartphone. Fitness apps have made our lives easier and enable you to track your activities on a daily basis. Thus, making you stay focused on your activities and overall fitness.

This application has some great built-in features like two categories, before age 18 and After age 18 which makes searching exercise for your age group easier. The timer indicates you for how long you have to perform a single exercise. There are full steps available under each exercise for how to perform that particular exercise. You can easily share this app with your friends and relatives.

The best part is that you can do all of these workouts from home—whenever is best for you! Whatever your fitness goals are for the new year. Download one right to your phone to tone up, stretch it out, or slim down—no gym membership required!

**Chapter 3**

**TECHNOLOGY SPECIFICATION**

**3.1 FRONT END**

**3.1.1 JAVA**

* Java is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible.
* Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture.
* Java was originally developed by James Gosling at Sun Microsystems and released in May 1995 as a core component of Sun Microsystems' Java platform.



**3.1.2 XML**

* Extensible Markup Language (XML) is a markup language and file format for storing, transmitting, and reconstructing arbitrary data.
* It defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The World Wide Web Consortium's XML 1.0 Specification of 1998 and several other related specifications—all of them free open standards—define XML.
* Several schema systems exist to aid in the definition of XML-based languages, while programmers have developed many applications programming interfaces (APIs) to aid the processing of XML data.



**3.1.3 GRADLE**

* Gradle is a build automation tool for multi-language software development.
* It controls the development process in the tasks of compilation and packaging to testing, deployment, and publishing. Supported languages include Java (as well as Kotlin, Groovy, Scala), C/C++, and JavaScript.
* It also collects statistical data about the usage of software libraries around the globe.
* Gradle is distributed as open-source software under the Apache License 2.0, and was first released in 2008.



**3.2 TOOLS USED**

**3.2.1 ANDROID STUDIO**

The Android Studio IDE is able to download and use. It has a rich UI development environment with templates to give new creator a opening pad into Android development. Developers will find that Studio gives them the tools to build phone and tablet mixture as well as arrival technology solution for Android.

Android Studio is purposeful to be used by development teams as small as one person or as big as global teams. The Android Studio IDE can be connected to big teams with GIT or same version control services for larger teams. The concept of repeatedly checking code provides actionable feedback to the developers with the goal of releasing versions of a mobile solution faster to the Google Play App Store. To finish this, there is rigorous support for LINT tools, Pro-Guard and App Signing tools.

Android Studio was stated on May 16, 2013 at the Google I/O conference. It was in early approach preview stage starting from version 0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. The first stable build was launched in December 2014, starting from version 1.0. The current stable version is 3.2.1, which was released in October 2018.

**3.2.2 FIREBASE**

Firebase is a mobile and web application development platform developed by Firebase, Inc. in 2011, then acquired by Google in 2014.As of October 2018, the Firebase platform has 18 products, which are used by 1.5 million apps.

Firebase evolved from Envolve, a prior startup founded by James Tamplin and Andrew Lee in 2011. Envolve provided developers an API that enables the integration of online chat functionality into their websites. After releasing the chat service, They founded Firebase as a separate company in September 2011 and it launched to the public in April 2012.

Firebase's first product was the Firebase Realtime Database, an API that synchronizes application data across iOS, Android, and Web devices, and stores it on Firebase's cloud. The product assists software developers in building real-time, collaborative applications.

**3.2.3 IntelliJ IDEA**

IntelliJ IDEA is an integrated development environment (IDE) written in Java for developing computer software. It is developed by JetBrains (formerly known as IntelliJ), and is available as an Apache 2 Licensed community edition, and in a proprietary commercial edition. Both can be used for commercial development.

The first version of IntelliJ IDEA was released in January 2001, and was one of the first available Java IDEs with advanced code navigation and code refactoring capabilities integrated.

In a 2010 InfoWorld report, IntelliJ received the highest test center score out of the four top Java programming tools: Eclipse, IntelliJ IDEA, NetBeans and JDeveloper.

In December 2014, Google announced version 1.0 of Android Studio, an open-source IDE for Android apps, based on the open-source community edition of IntelliJ IDEA. Other development environments based on IntelliJ's framework include AppCode, CLion, Data Grip, GoL and, Php Storm, PyCharm, Rider, Ruby Mine, WebStorm, and MPS.

**Chapter 4**

**SRS (Software Requirement Specifications)**

**4.1 FUNCTIONAL REQUIREMENT**

* Duration (Stopwatch)
* Start Activity
* Stop Activity

**4.2 NON-FUNCTIONAL REQUIREMENTS**

* **Performance Requirements**

1. The load time for the user interface screen should take no longer than 5 seconds.
2. Time should be displayed on screen.

* **Design Constraints**

The application should be able to run on any Android device having a minimum Android version 8.0(Oreo)

The application should be available at all times.

**4.3 HARDWARE REQUIREMENTS**

|  |  |
| --- | --- |
| **Name of component** | **Specification** |
| Processor | Oreo, Snapdragon 400, MediaTek P20 and above |
| RAM | 1GB or more |
| Storage | 2GB or more |
| Display | Touch Screen |

**4.4 SOFTWARE REQUIREMENTS**

The software requirement specification is produced at the the culmination of the analysis task. The functions and performance allocated to software as part of engineering system are refined by establishing a complete information description, a detailed functional and behavioral description, an indication of performance requirements and design constraints, appropriate validation criteria, and the other data pertinent to requirements.

|  |  |
| --- | --- |
| **Name Of Component** | **Specification** |
| Operating System | Android Version 8.0 and above |
| Language | Java |

**4.5 SRS AND METHODOLOGY**

**4.5.1 Model Used**

**Waterfall Model**

The waterfall is a universally accepted SDLC model. In this method, the whole process of software development is divided into various phases.

The waterfall model is a continuous software development model in which development is seen as flowing steadily downwards (like a waterfall) through the steps of requirements analysis, design, implementation, testing (validation), integration, and maintenance.

**Iterative Model**

It is a particular implementation of a software development life cycle that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete. In short, iterative development is a way of breaking down the software development of a large application into smaller pieces.

**Note:** We have used waterfall model in the development of this project but iterative model can also be considered.

**Chapter 5**

**FEASIBILITY STUDY**

A feasibility study assesses the operational, technical and economic merits of the proposed project. From the systems analyst perspective, the feasibility analysis is the primary tool for recommending whether to proceed to the next phase or to discontinue the project.

The feasibility study is a management-oriented activity. The objective of a feasibility study is to find out if an information system project can be done and

to suggest possible alternative solutions.

The System to be act as worth-while it should pass through some test that examine that it should proceed further or not. This series of test is commonly known as feasibility study on the system and it plays a very vital role for every system projects. Feasibility studies undergo three major analyses to predict the system to be success and they are as follows: -

* Operational Feasibility
* Technical Feasibility
* Economic Feasibility

**5.1 Operational Feasibility: -**

Operational feasibility is a measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development. Operational feasibility reviews the willingness of the organization to support the proposed system. It measures the viability of a system in terms of the **PIECES** framework.

* The fitness app works properly in all the Android version(Android 8.0) which is the major difference between the mobile application and web applications software I.e. efficiency of working any operating system.
* The infrastructure of this fitness app is very simple and easy to use .
* The app doesn’t interrupt users security and privacy. The user data is fully protected and secured as our app doesn’t require to much permission for successful installation the app.

**5.2 Technical Feasibility: -**

A large part of determining resources has to do with assessing technical feasibility. It considers the technical requirements of the proposed project. The technical requirements are then compared to the technical capability of the organization. The systems project is considered technically feasible if the internal technical capability is sufficient to support the project requirements.   
The analyst must find out whether current technical resources can be upgraded or added to in a manner that fulfils the request under consideration.  This is where the expertise of system analysts is beneficial, since using their own experience and their contact with vendors they will be able to answer the following questions.

The essential questions that help in testing the operational feasibility of a system include the following:

* The fitness app is technically feasible as it is developed within the limits of current technologies (using Java, XML and Gradle). A lot of fitness apps are also available but this app doesn’t require much specification compare to others.
* The development of this project required to people’s manpower who worked as programmer, testers and debuggers and also took some external help and guidance from the internet and guardians.
* The project doesn’t require much high software and hardware requirements as it is a very noncomplex, easy to use and lightweight application program.

**5.3 Economic Feasibility: -**

Economic analysis could also be referred to as cost/benefit analysis. It is the most frequently used method for evaluating the effectiveness of a new system. In economic analysis the procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs. If benefits outweigh costs, then the decision is made to design and implement the system. An entrepreneur must accurately weigh the cost versus benefits before taking an action.

* The proposed application program is pretty much cost effective as it doesn’t require higher hardware and software needs.
* The estimated cost for the development of this project is approximately 8000/-. (Which includes hardware, software, manpower and other resources).
* The desired price of this project is 12000/-

**Chapter 6**

**MODULE DESCRIPTION**

**6.1 TEST MODULE**

Test Modules are containers for tests of user stories or software requirements. Every test module has a clear and well-differentiated scope from every other test module to avoid redundant activities and checks. This makes tests less fragile and easier to maintain. Giving each Test Module a descriptive name makes it easy to identify what the tests in the module cover. We used test module in the build/ in XML file.

**6.2 GRADLE MODULE**

Gradle Module Metadata is a format used to serialize the Gradle component model. It is similar to Apache Maven ™'s POM file or Apache Ivy™ ivy.xml files. The goal of metadata files is to provide to consumers a reasonable model of what is published on a repository.

Gradle Module Metadata is a unique format aimed at improving dependency resolution by making it multi-platform and variant-aware. We used gradle module in the build/ in XML file.

**6.3 TIME MODULE**

Time module provides many ways of representing time in code, such as objects, numbers, and strings. It also provides functionality other than representing time, like waiting during code execution and measuring the efficiency of your code.

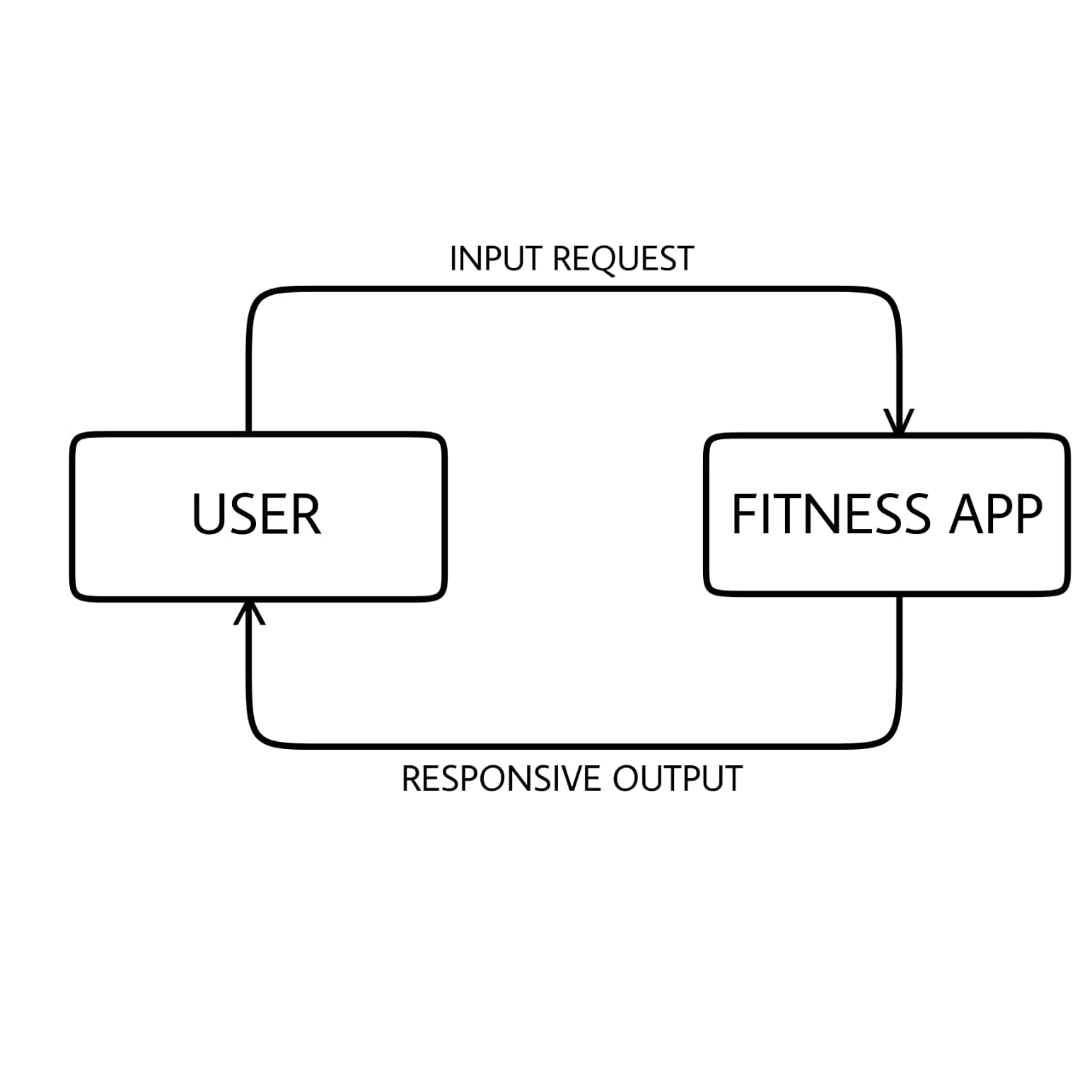
Time module is used in the XML but not together with Gradle and test which works as the stopwatch.

**Chapter 7**

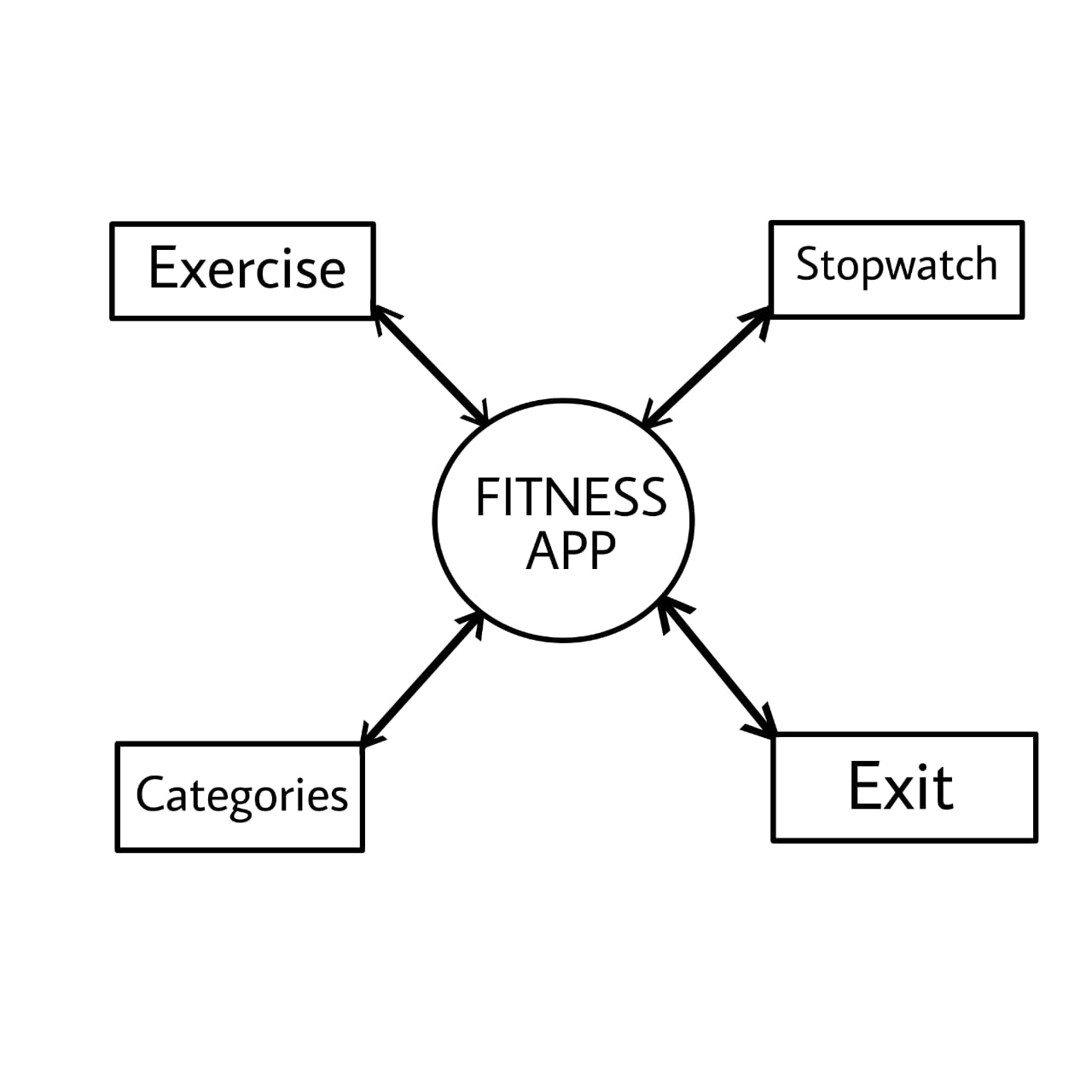
**SYSTEM DESIGN**

**7.1 DFD (DATA FLOW DIAGRAMS)**

**7.1.1 Level 0 DFD**

****

**7.1.2 Level 1 DFD**

****

**Chapter 8**

**TESTING**

**8.1 Introduction to Testing**

Testing is a method to check whether the actual software product matches expected requirements and to ensure that software product is Defect free. It involves execution of software/system components using manual or automated tools to evaluate one or more properties of interest. The purpose of software testing is to identify errors, gaps or missing requirements in contrast to actual requirements.

**8.2 Level of Software Testing**

Tests are grouped together based on where they are added in SDLC or the by the level of detailing they contain. In general, there are four levels of testing: unit testing, integration testing, system testing, and acceptance testing. The purpose of Levels of testing is to make software testing systematic and easily identify all possible test cases at a particular level.

**8.2.1 Unit Testing**

In this type of testing, errors are detected individually from every component or unit by individually testing the components or units of software to ensure that if they are fit for use by the developers. It is the smallest testable part of the software. It helped us in testing tabs separately in project and different activities one by one separately. Hence, we fixed all bugs.

**8.2.2 System Testing**

In system testing, complete and integrated softwares are tested i.e all the system elements forming the system is tested as a whole to meet the requirements of the system**.** This testing was done in the project at the end after the integration to check whole project.

**8.2.3 Integration Testing**

Integration testing is the process of testing the interface between two software units or module. The purpose of the integration testing is to expose faults in the interaction between integrated units. Once all the modules have been unit tested, integration testing is performed.

It tests each module at a time and refers to test by system testing. Integration is combining, it means it combines some unit testing and then tests.

**8.3 Testing Process**

These are 11 steps software testing process is an experience based practical approach for solution to test assignment. These are explained as following below.

1.Assess Development Plan and Status

2. Develop the Test Plan

3. Test Software Requirements

4. Test Software Design

5. Build Phase Testing

6. Execute and Record Result

7. Acceptance Test

8. Report Test Results

9. The Software Installation

10. Test Software Changes

11. Evaluate Test Effectiveness

**Chapter 9**

**SYSTEM ANALYSIS**

**9.1 Overview**

It is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components.

To provide an overview of fitness apps including the key players like Equinox who may have turned to digital in order to remain relevant during COVID-19, key end-user/consumer needs for fitness app (fitness and fitness related), key current and future market drivers (what is the market expecting, what technology will impact fitness, what other services), and the shortcomings of current fitness apps.

**9.2 Implementation**

Implementation is a process of ensuring that the information system is operational. It involves -

Constructing a new system from scratch. We have used Java, XML and Json for the implementation of this app based on the Android studio. It helps the user to stay fit and workout able for long term.

Implementation allows the users to take over its operation for use and evaluation. It involves training the users to handle the system and plan for a smooth conversion.

**9.3 System Maintenance**

Maintenance means restoring something to its original conditions. Enhancement means adding, modifying the code to support the changes in the user specification. System maintenance conforms the system to its original requirements and enhancement adds to system capability by incorporating new requirements.

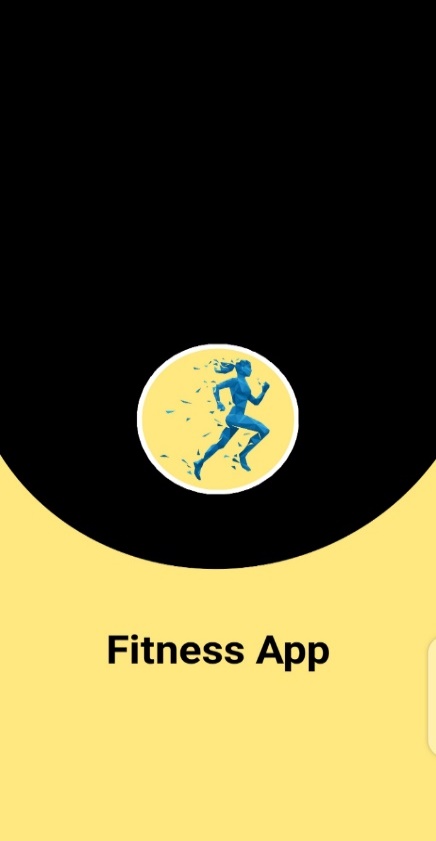
There were some bugs at first. Like, some tabs were not responding which starts responding and enhanced the user interface too in comparison to last time.

**Chapter 10**

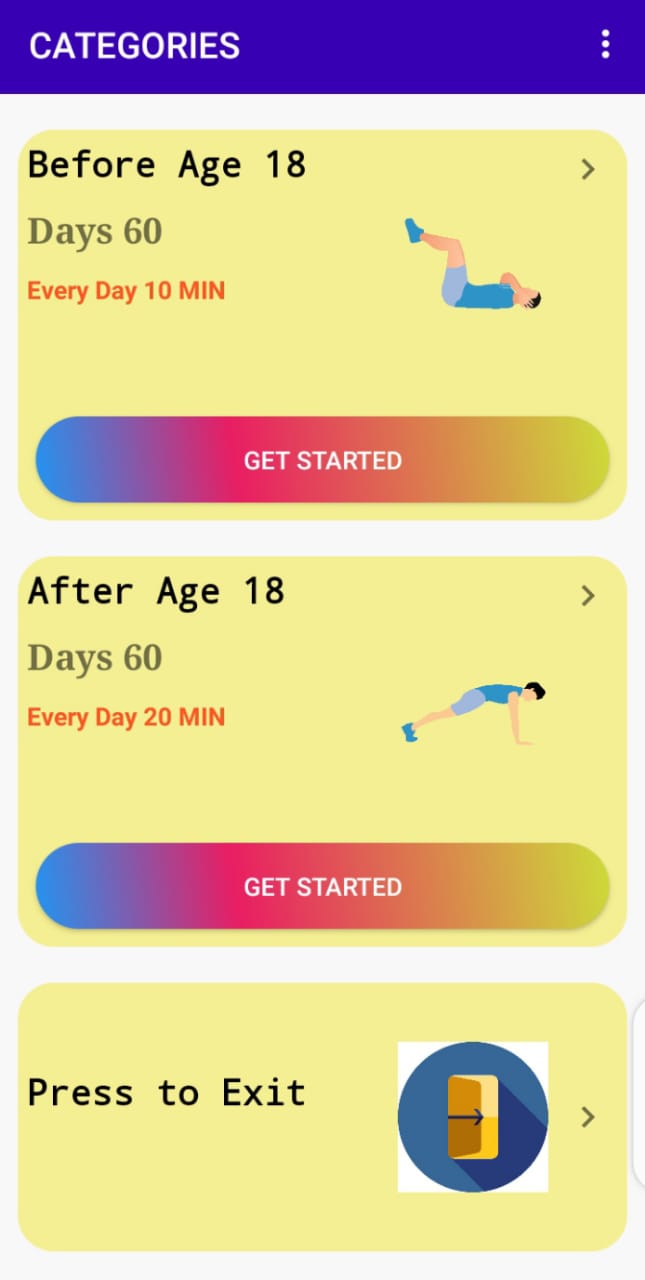
**SNAPSHOTS**

**10.1 Screenshots**

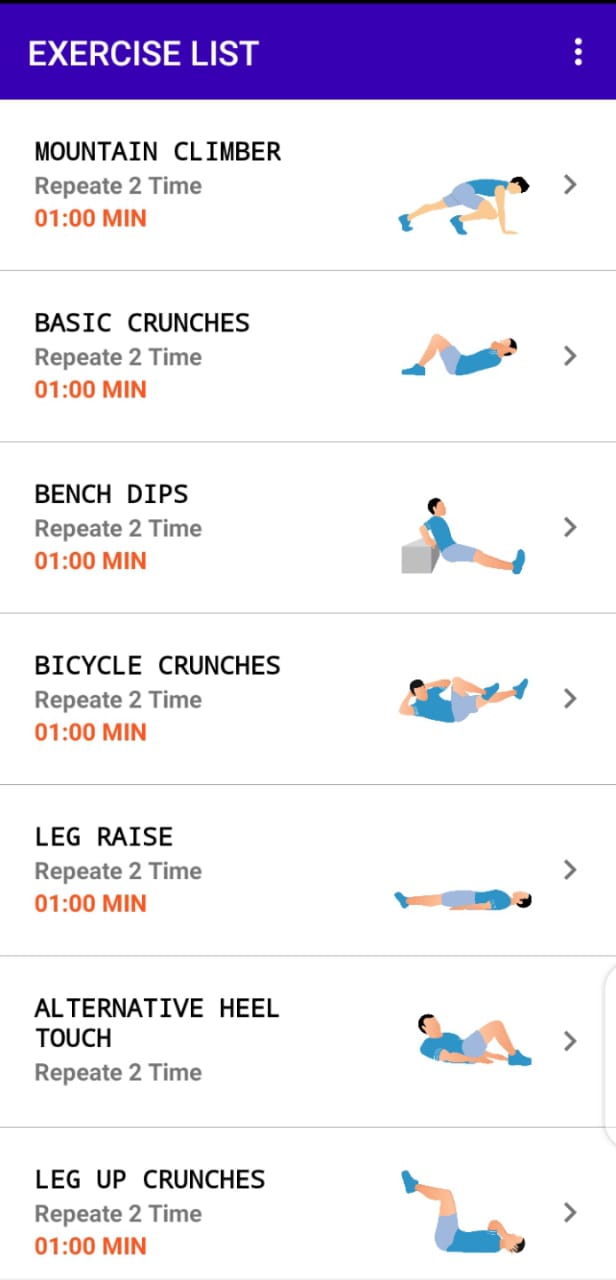
**10.1.1 Opening App**

****

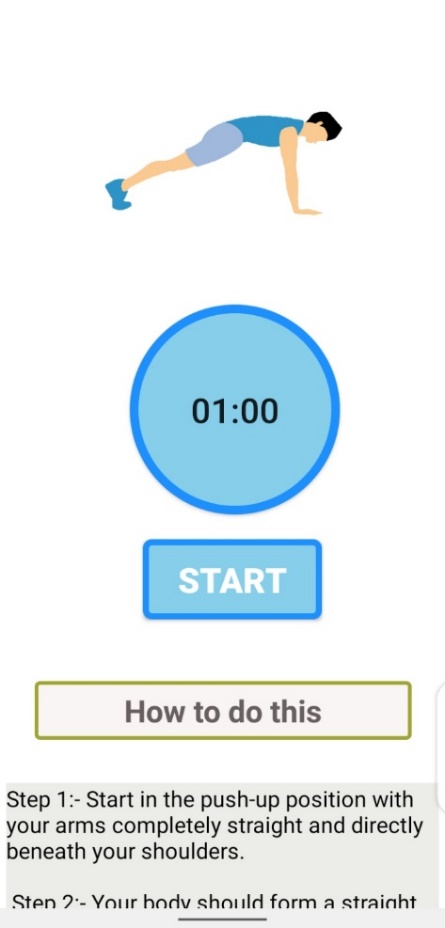
**10.1.2 Homepage**

****

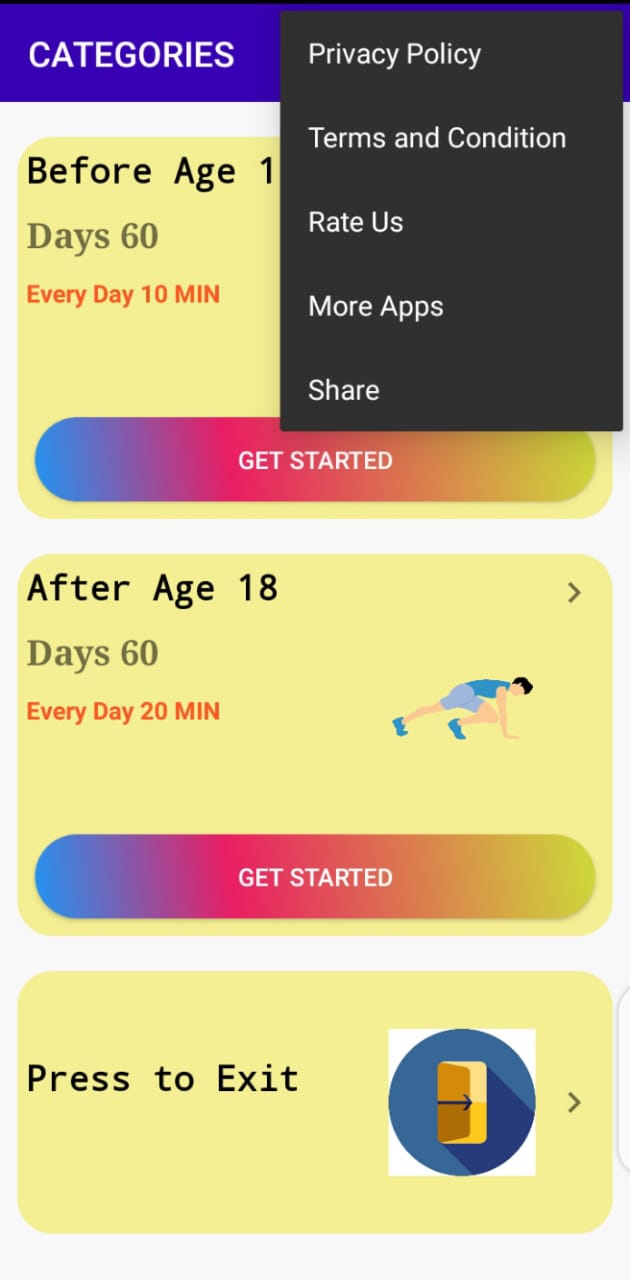
**10.1.3 Exercises List**

****

**10.1.4 Exercise brief with timer**

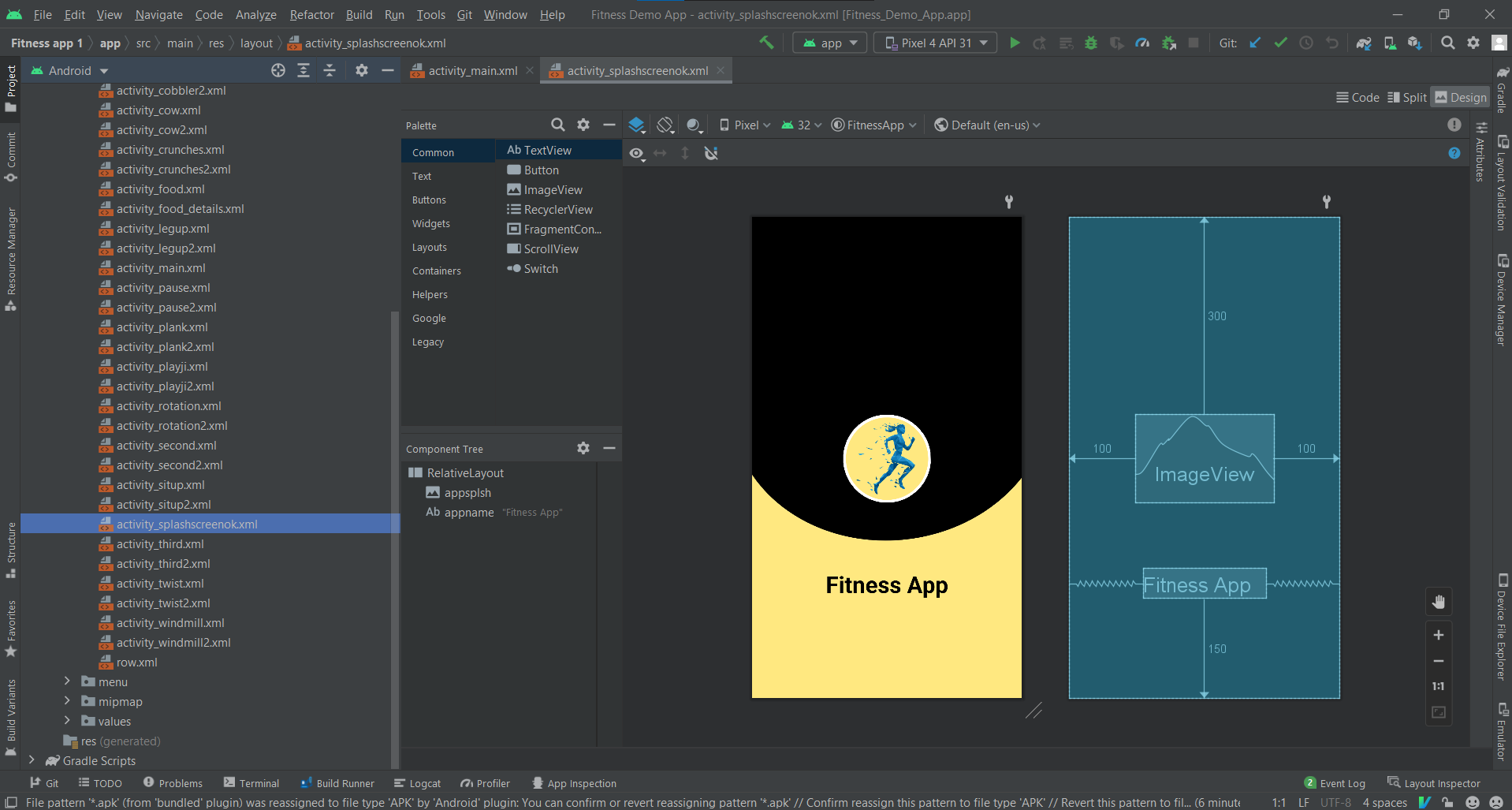
****

**10.1.5 Other Options available**

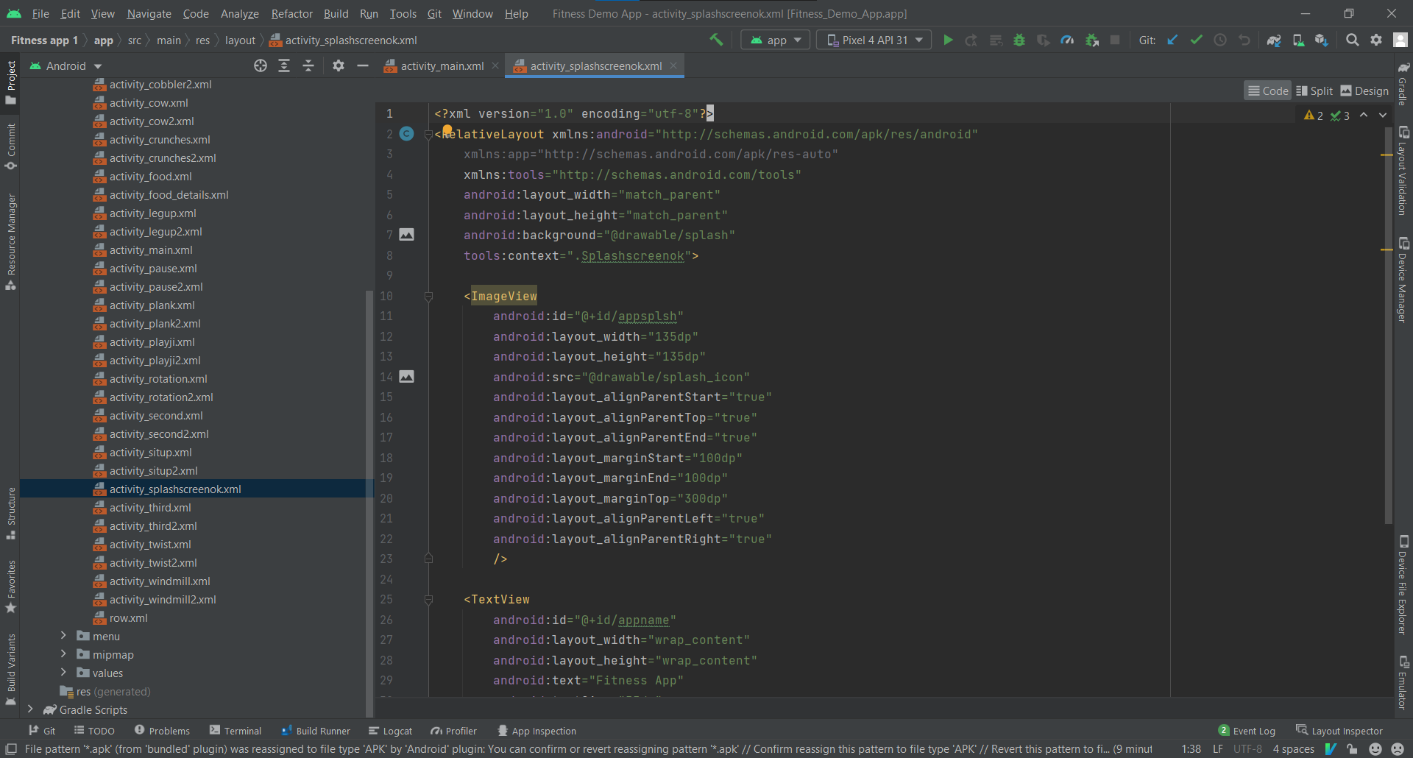
****

**10.2 Designs and their code**

**10.2.1 Opening Page**

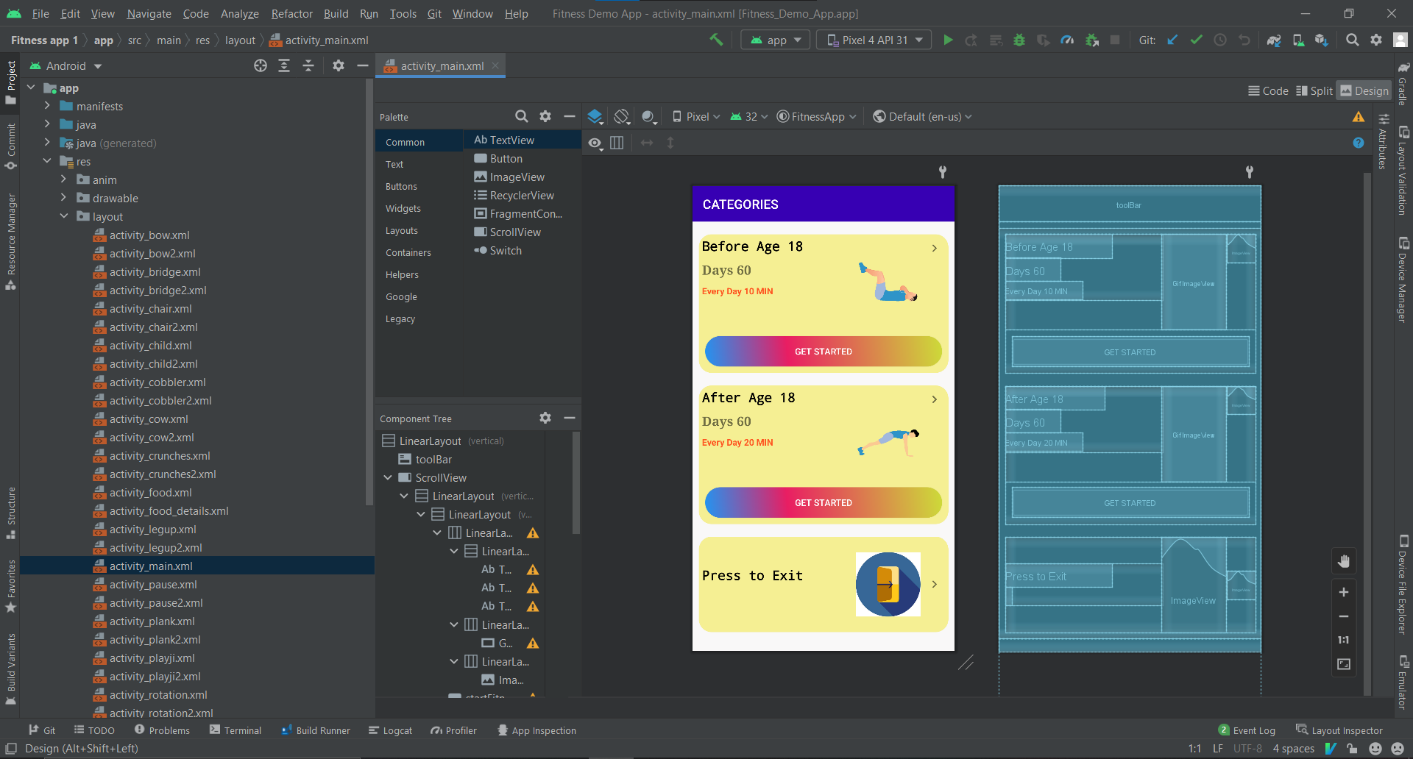
****

Design

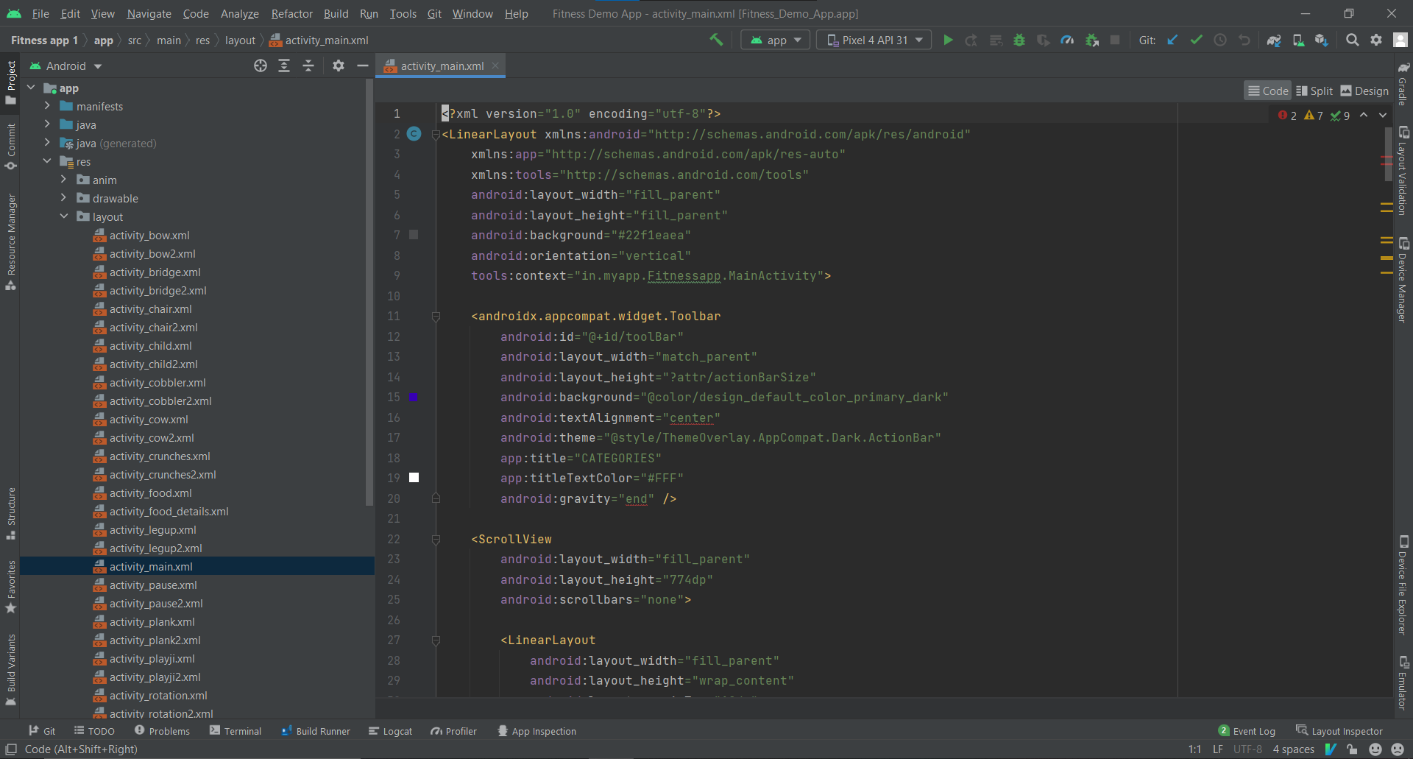


Code

**10.2.2 Homepage**

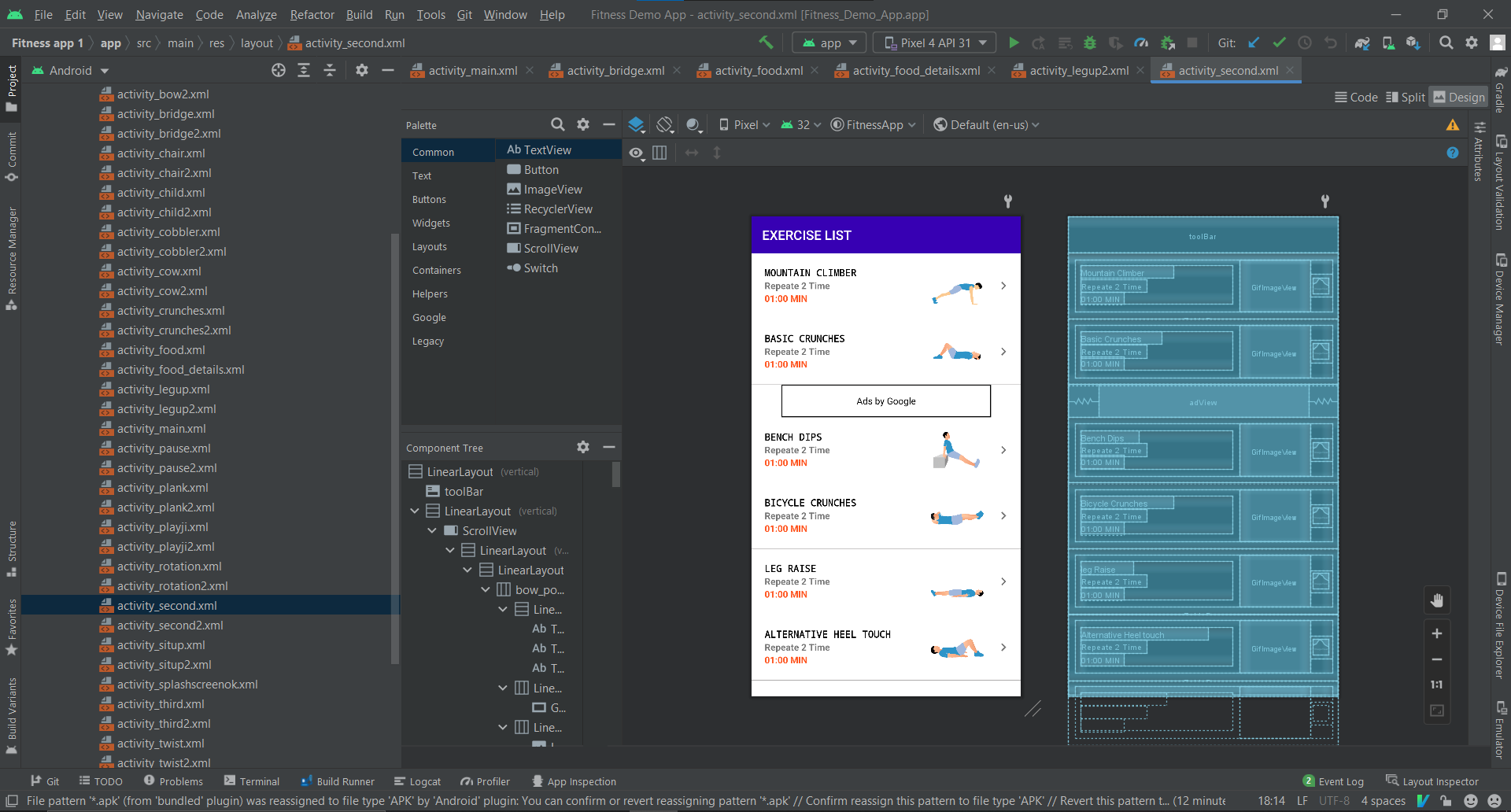
****

**Design**

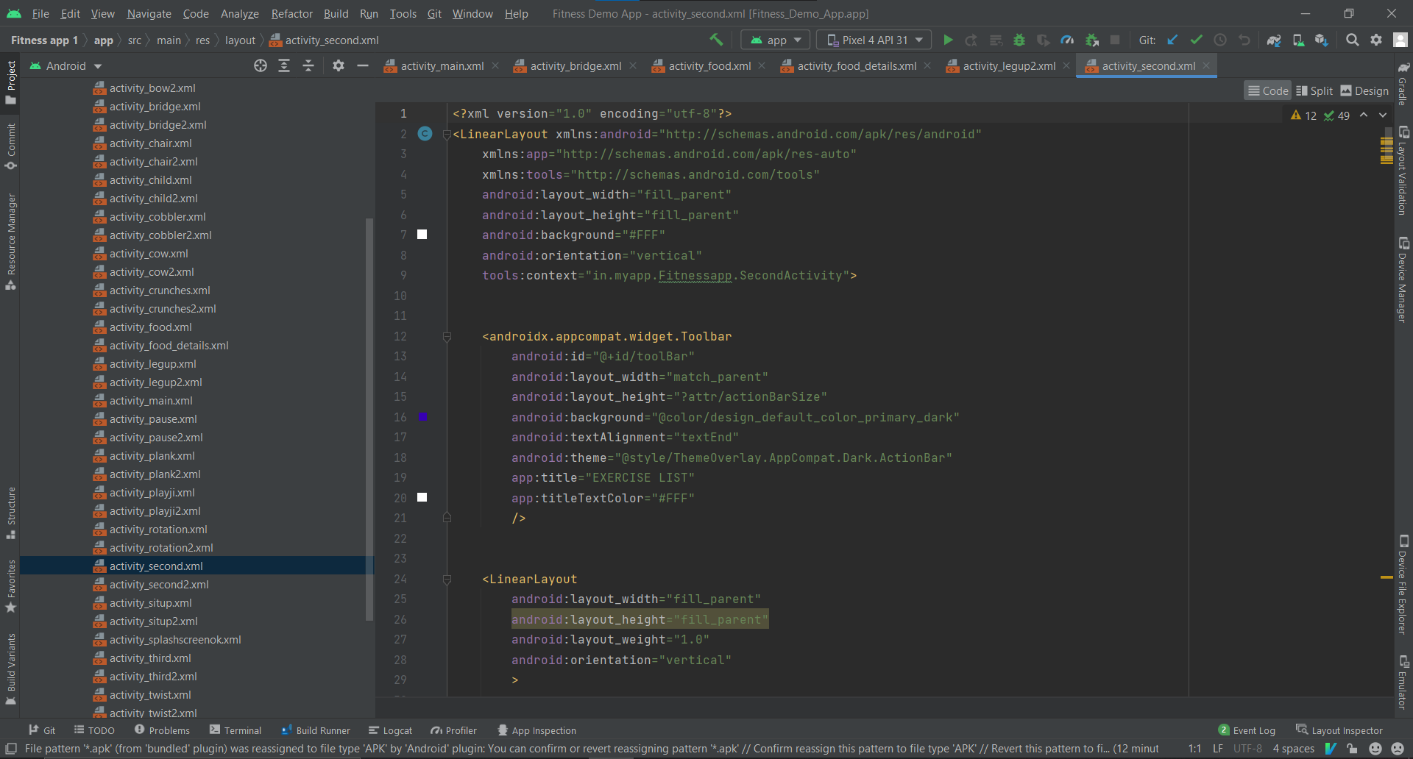


**Code**

**10.2.3 Exercise List**

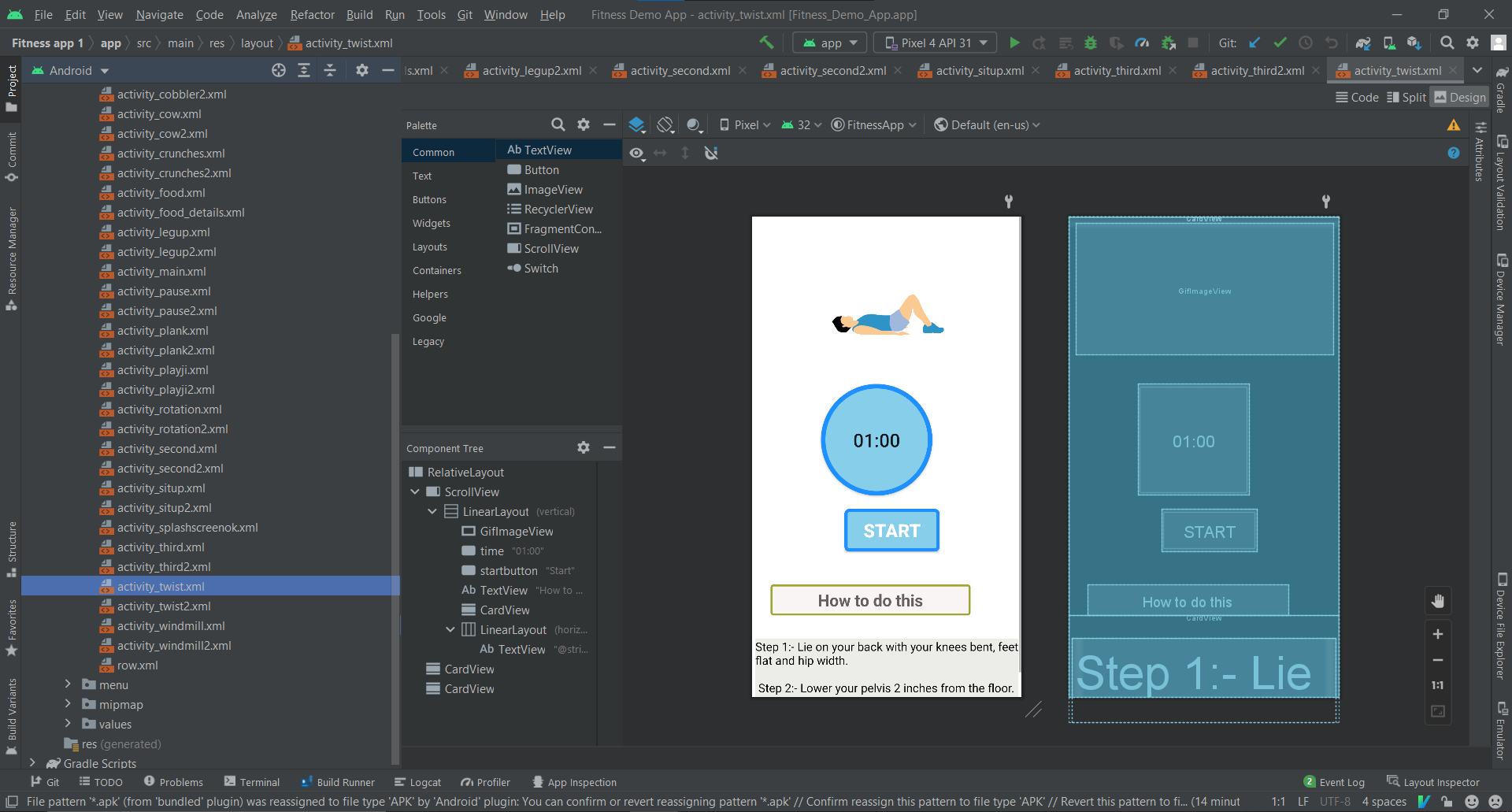
****

**Design**

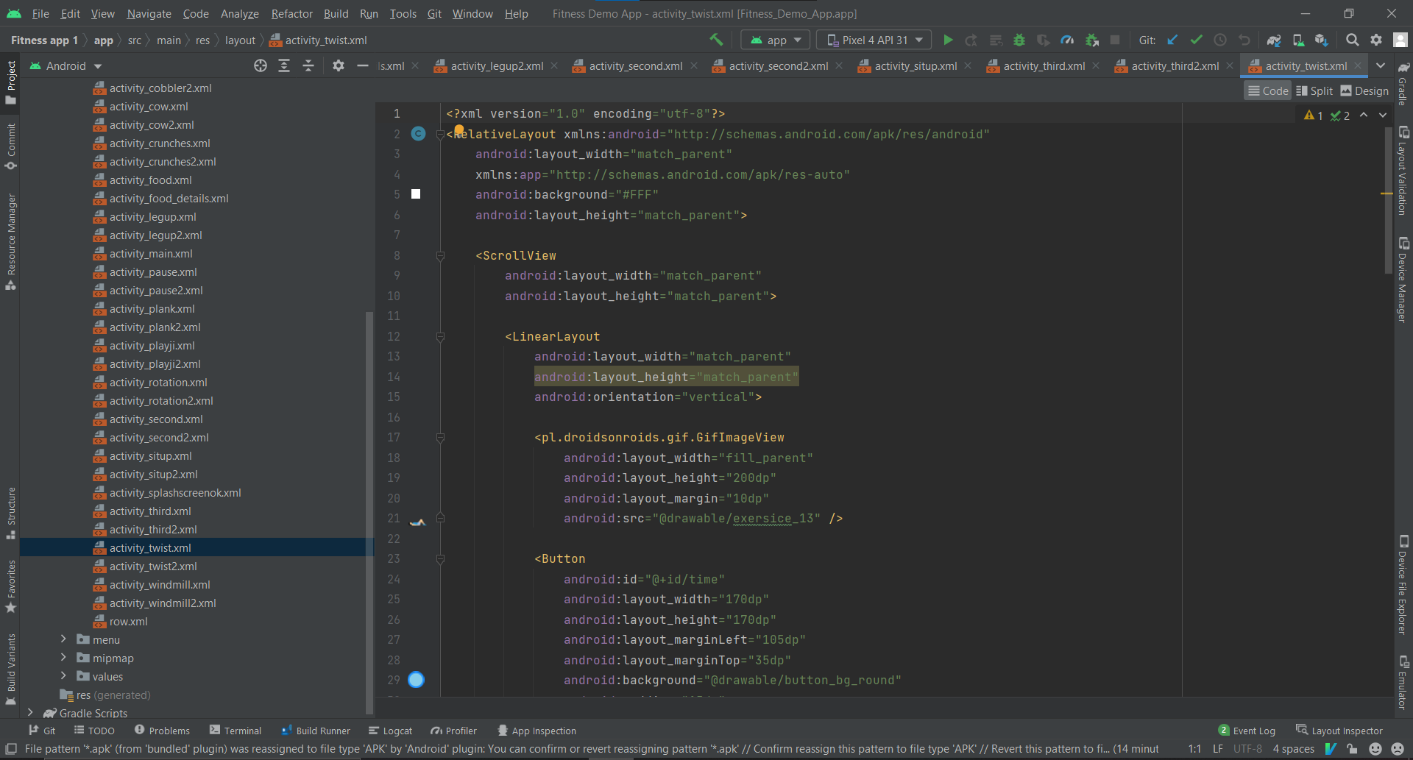
****

**Code**

**10.2.4 Exercise with stopwatch**

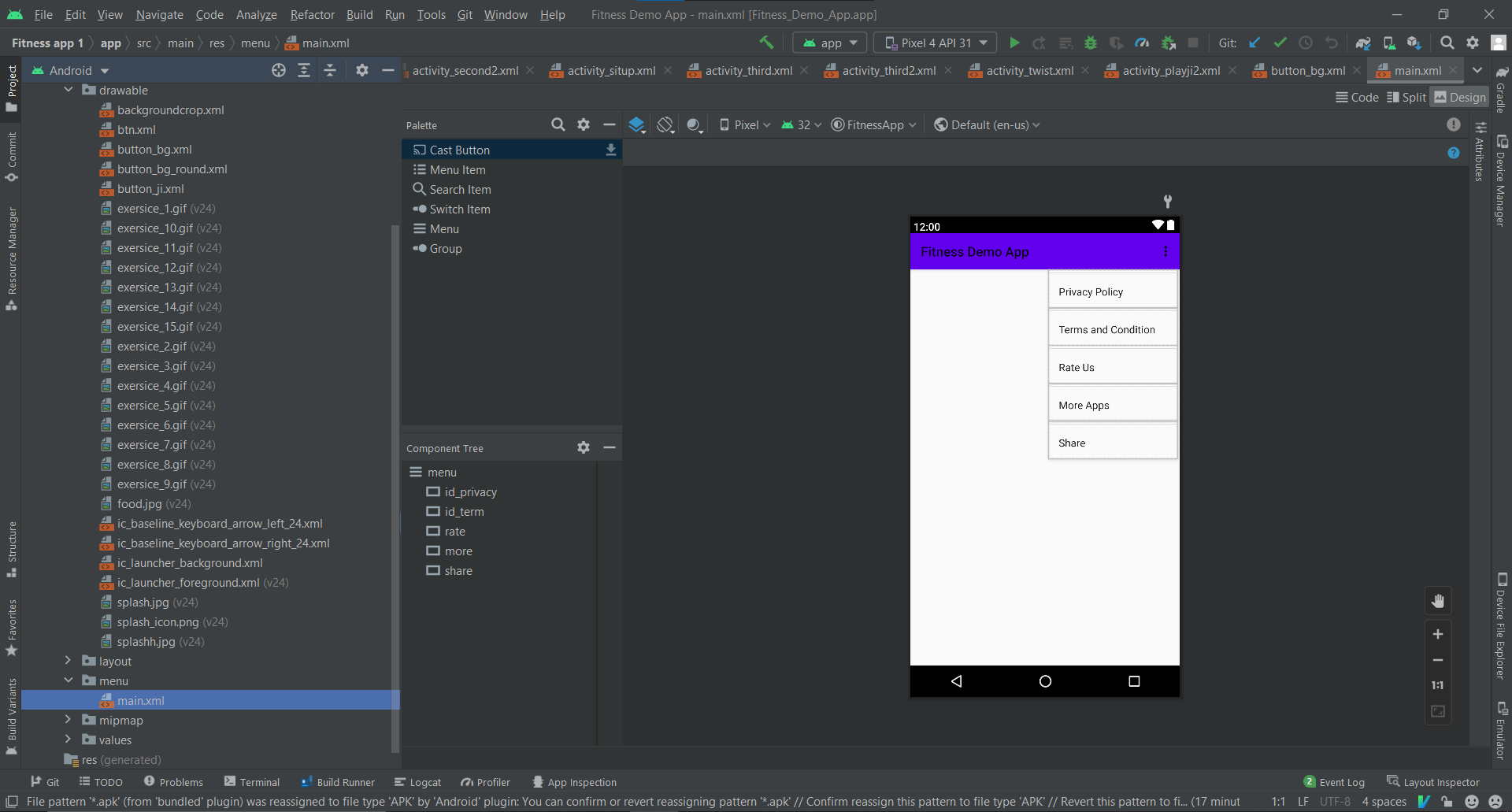
****

**Design**

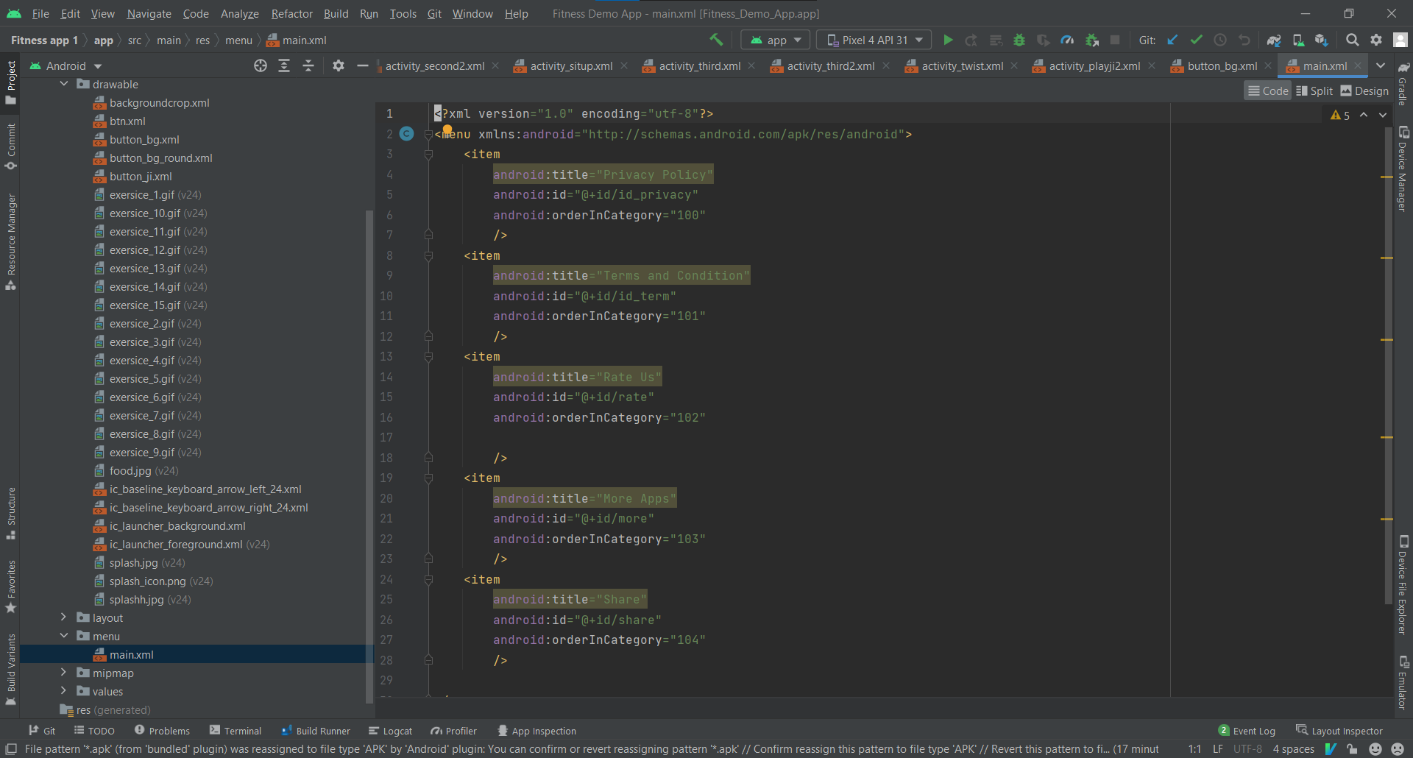
****

**Code**

**10.2.5 Other Options**

****

**Design**

****

**Code**

**Chapter 11**

**Conclusion**

The project titled **Fit Freak App** is an android-based application that enables the user to keep an eye on their fitness regime. The project has been completed successfully with maximum satisfaction. The constraints are met and overcome successfully. The system is designed like it was decided in the design phase. The project gives a good idea on developing a user-friendly application satisfying the user. The system is very flexible and versatile. This application has a user-friendly screen that enables the user to use it without any inconvenience.

I hope you have enjoyed learning about how to stay healthy and fit. There are many ways to exercise along with many healthy foods you can eat to help stay fit. When exercising it is good to make sure your pulse rate is good, and that you warm up your muscles before exercising and to make sure you do a cool down to help your muscles cool down.

To create the level of enthusiasm among the users this app has a practical approach. Daily exercising would keep them healthy and fit.

Here is the application which helps the user in finding the right exercise for his/her age group and performing them for in a order of time.

**Chapter 12**

**FUTURE SCOPE**

The future work of this application is to raise your awareness, as a fitness professional in his personal Life, as to how people may select fitness exercise from the many activities that are out there. If we know what people are looking for from a fitness app then we know what we are competing with from a gym or online trainers’ point of view.

In Future we will Import: -

1. Progress Bar – which will show the progress of user.
2. Navigation Drawer – for better performance.
3. New Stages – new exercise stages will be added.
4. Calendar – which will show on which date the exercise is done by the user.

**REFERENCES**

**Websites:**

* <https://www.youtube.com/>
* <https://github.com/>
* <https://www.google.co.in/>
* <https://en.wikipedia.org/wiki/Main_Page>