Step Counter

1. Introduction.

A StepCounter is a device, usually portable and electronic or electromechanical, that counts each step a person takes by detecting the motion of the person's hands or hips. Because the distance of each person's step varies, an informal calibration, performed by the user, is required if presentation of the distance covered in a unit of length (such as in kilometers or miles) is desired, though there are now pedometers that use electronics and software to automatically determine how a person's step varies.
A Step Tracker/Counter generates an event each time a step is taken by the user.

About StepCounter.

This app is very simple to use and interact as its UI is as easy as it looks, and does exactly what you need. Stats are clear and easily accessible from the menu on the right, and it will automatically set you a daily goal and monitor your progress. In addition to your counted steps, the information will display the distance in km.
Also, you will able to view and manage the goals you set to follow and step size you take or you personally set to follow the application as it is recommended set.
Also, here you can store or save the data file in your android device as it collects and shows the graphical statistics.
A notifications setting is also an option to keep track on your current steps taken.

How does it work?

These applications work with the phones which have the Accelerometer sensor built in it. While running, this application will monitor the change in accelerometer and based on it will conclude if the step is taken. It can provide the information about the angle at which you are holding the device, direction, speed at which it is moved and the gravity. Accelerometer provides the values whenever they are changed along with the Time at which the event occurred. Application stores these values in the database and look for a pattern.

An accelerometer will provide three values x-axis value, y-axis value and z-axis value. As the person starts taking the first step, x-axis value will increase at a rate depending upon the force at which the person starts to move, y-axis which will tell the relative change in the height of the device along with the force will show an increase in value as while taking the step person will rise from the ground and then it will decrease till the step touches the ground. Based on this kind of pattern match, Step event is detected.

Step Counter

Step Counter

2. Requirement Specifications.

Tools and Technology Used.

Hardware: Android Device with Android version 6.0+

Software: Android Studio version 3.3.3.0.0 Google LLC.

Modularity: Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems. It is a replacement for the Eclipse Android Development Tools (ADT) as the primary IDE for native Android application development.

Language use in Android Studio:

- **Java** – Java is the official language for Android development and is supported by Android Studio. It has a steep learning curve, one of the reasons being that it's simply the official language of Android app development, which means it is one of the most supported languages by Google and the one that most apps in the Play Store are built with. Java is an object-oriented programming language with confusing topics like constructors, null pointer exceptions, checked exceptions, and a lot more. It's not terribly readable and you'll use a lot of code for simple things.

Also it contains other language to create your android app by Kotlin, C/C++, C#, BASIC, Corona/LUA, PhoneGap (HTML, CSS, JavaScript).

3. System Design Details.
3.1   Methodology.

Waterfall model is a sequential flow which is used in developing particular software.

This helps us to guide through the making of software/application.

It consists of **six phases** which are as follows:

1) Requirement Analysis.
2) Designing of software.
3) Implementation of software.
4) Testing/Working of the software.
5) Deployment of the software.
6) Maintaining of a software.

Benefits:

➤ The main purpose of this model is that it is one time process.
➤ When you proceed to next phase you can't go back to the previous phase so once you have completed one phase checking thoroughly and moving to next phase is a smart move.
➤ The use of this model allows to early design of the software in the planning phase then in modeling phase which it is only implemented.
➤ Tasks of this model to various group of people is easy to manage and co-ordinate.
➤ Once you have all the requirements needed it is easy to make the software/application.
➤ As the model is sequential and linear the working of well working software can be divided into group as per the phases.
➤ The final process or results or output are well log maintained which means it is well documented.

Why I choose this waterfall model for my software/application?

➤ As my software/application has all the requirements analyze at the start of making the software it is easy for this model to proceed.
➤ The root of making this software is clear and stable from beginning which means no sudden changes will be done.
➤ Customer/Client using the final software will not be involved in making of the software.

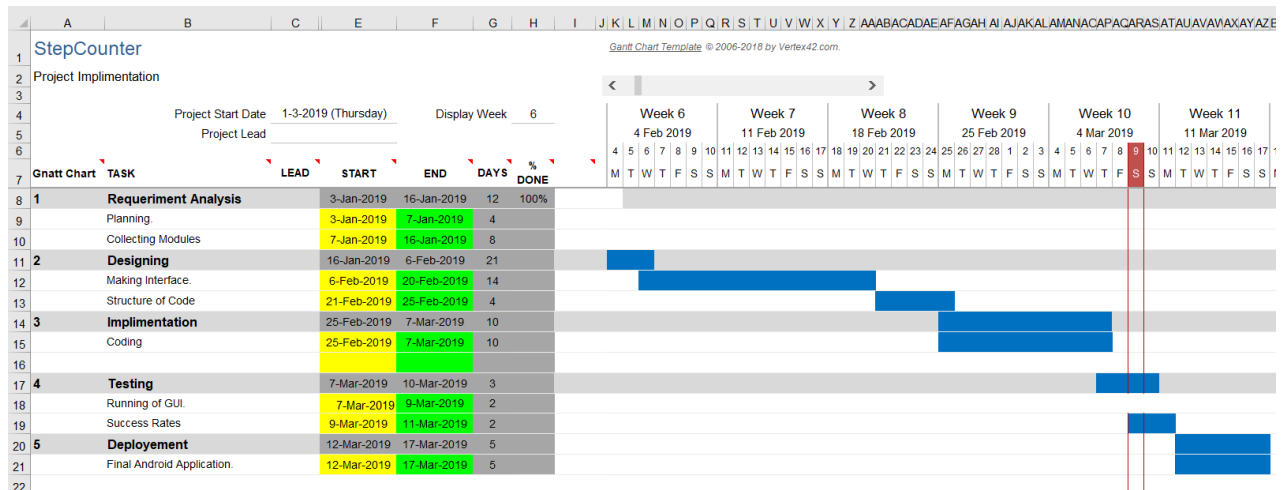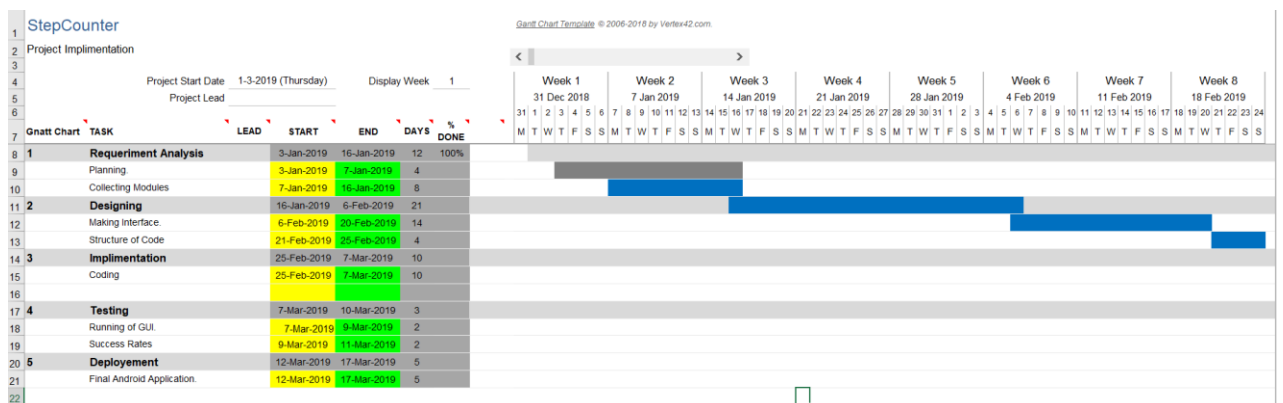# Step Counter

Diagram Representation.

TYBSc (CS) Sem VI   Roll No: 21
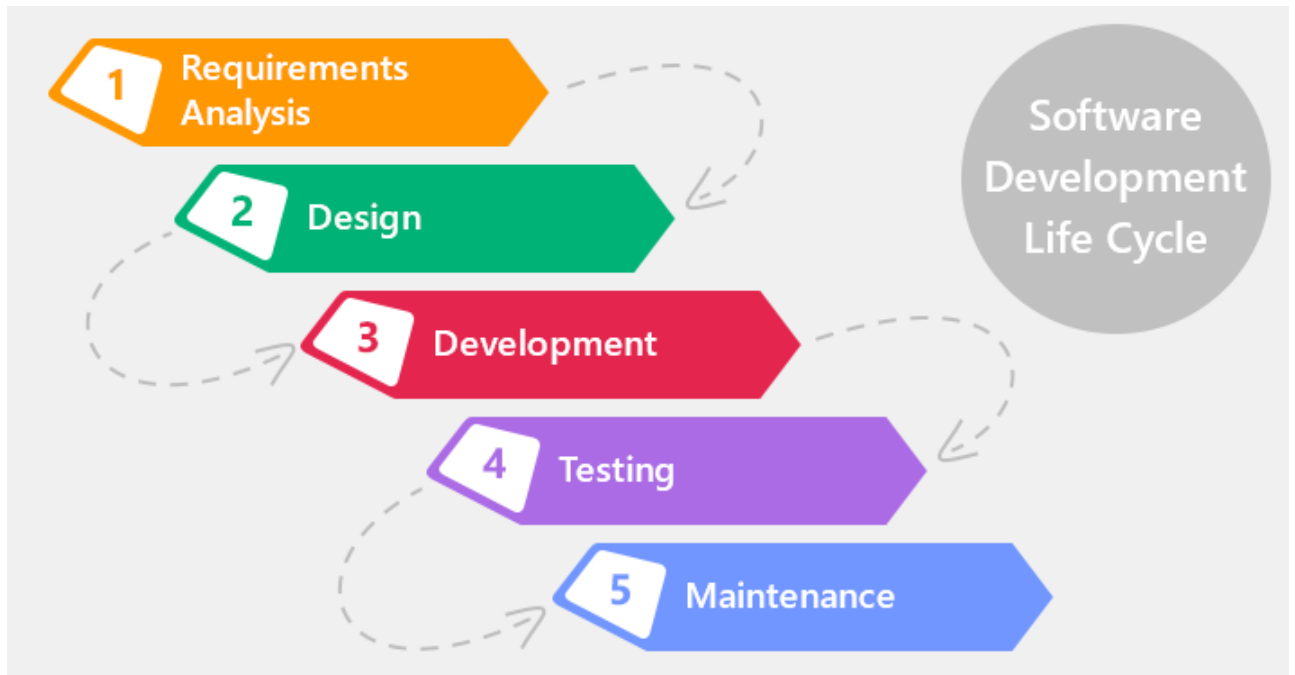
Step Counter

3.2 Architecture.
1) Gantt chart.

A Gantt chart is a horizontal bar chart developed as a production control tool. Frequently used in project management, a Gantt chart provides a graphical illustration of a schedule that helps to plan, coordinate, and track specific tasks in project.

Gantt chart give a clear illustration of project status, but one problem with them is that they don't indicate task dependencies – you cannot tell how one task is falling behind schedule affects other tasks.

TYBSc (CS) Sem VI   Roll No: 21
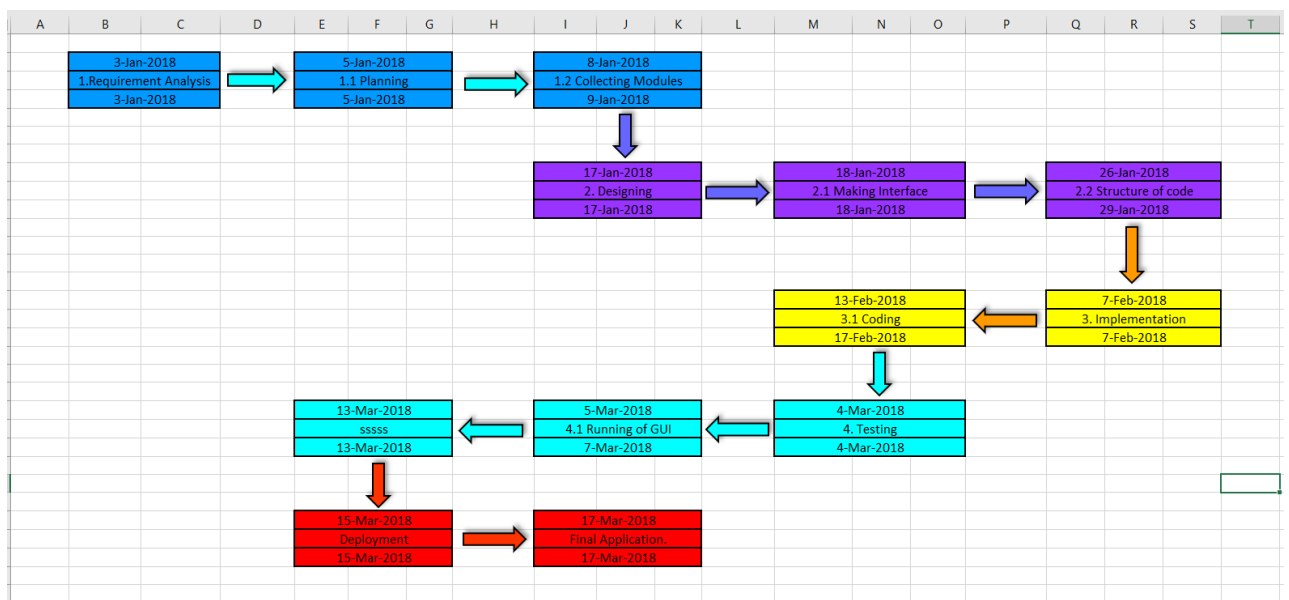
Step Counter

2) Pert Chart.

(Program Evaluation Review Technique)

A PERT chart is a project management tool used to schedule, organize, and coordinate tasks within a project.

Pert represents a graphical illustration of a project as a network diagram consisting of numbered nodes representing events, or milestones in the project linked by labelled vectors representing tasks in the project.

The direction of the arrows on the lines indicates the sequence of tasks.

TYBSc (CS) Sem VI   Roll No: 21

Step Counter

## 3.3 UML (Unified Modeling Language).
3.3.1 Activity Diagram.

Activity diagram presents a number of benefits to users. They describe what must happen in the system being modeled.
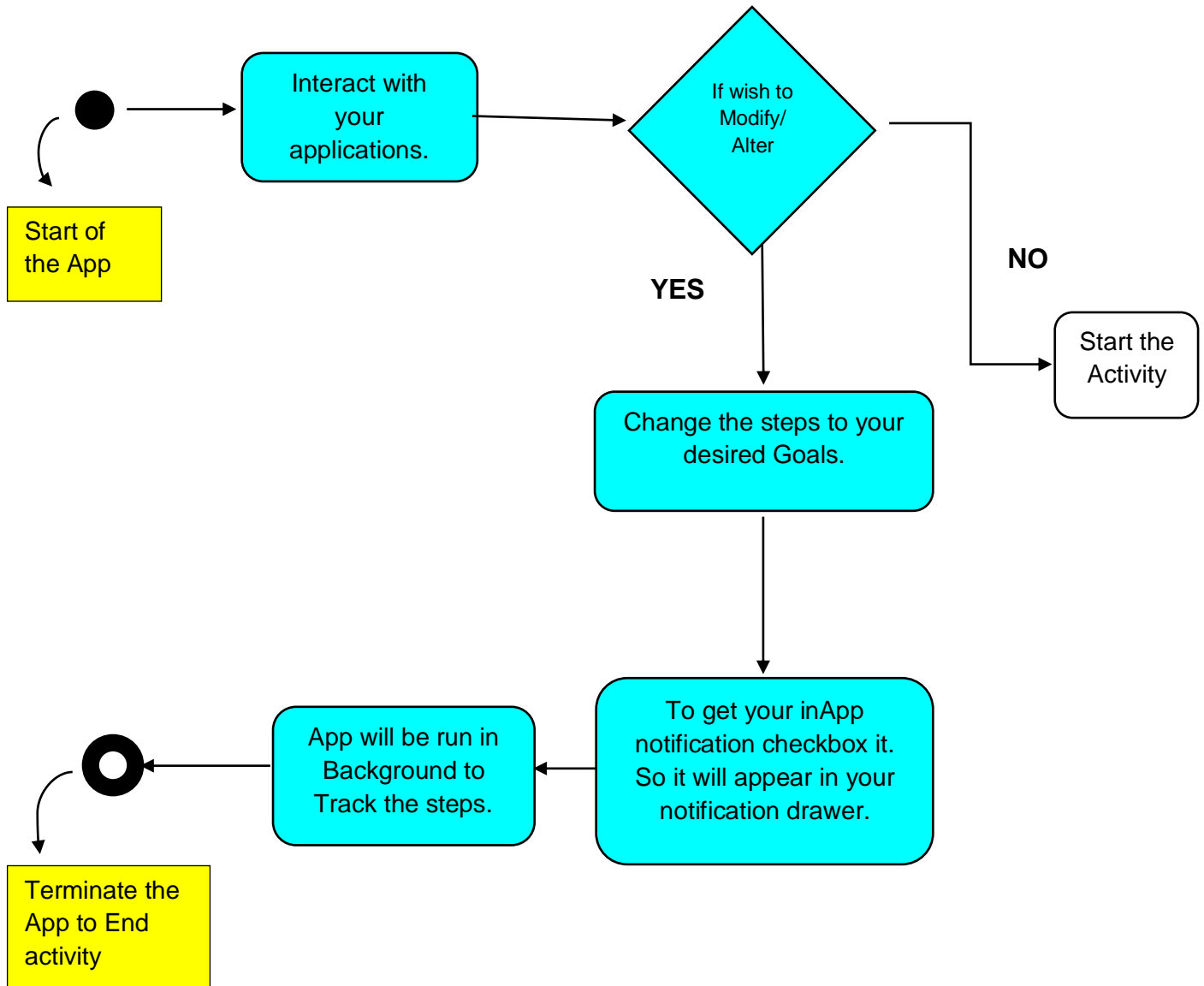
It is used for starting, ending, merging, or receiving steps in the flow.

The components of an activity diagram includes:

Action, Decision mode, Control flows, Start node, and end node.

Symbol	Name	Description
●	Start symbol	Represents the beginning of a process or workflow in an activity diagram. It can be used by itself or with a note symbol that explains the starting point.
Activity	Activity symbol	Indicates the activities that make up a modeled process. These symbols, which include short descriptions within the shape, are the main building blocks of an activity diagram.
▬▬▶	Connector symbol	Shows the directional flow, or control flow, of the activity. An incoming arrow starts a step of an activity; once the step is completed, the flow continues with the outgoing arrow.
[Condition]	Condition text	Placed next to a decision marker to let you know under what condition an activity flow should split off in that direction.
◉	End symbol	Marks the end state of an activity and represents the completion of all flows of a process.

Step Counter

Start of the App

Interact with your applications.

If wish to Modify/Alter

YES

NO

Start the Activity

Change the steps to your desired Goals.

To get your inApp notification checkbox it. So it will appear in your notification drawer.

App will be run in Background to Track the steps.

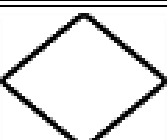**Terminate the App to End activity**

3.3.2 Flow Chart

A flowchart is a diagram that depicts a process, system or computer algorithm. They are widely used in multiple fields to document, study, and plan improve and communicate often complex processes in clear, easy-to-understand diagrams. Flowcharts use rectangles, ovals, diamonds and potentially numerous other shapes to define the type of step, along with connecting arrows to define flow and sequence.
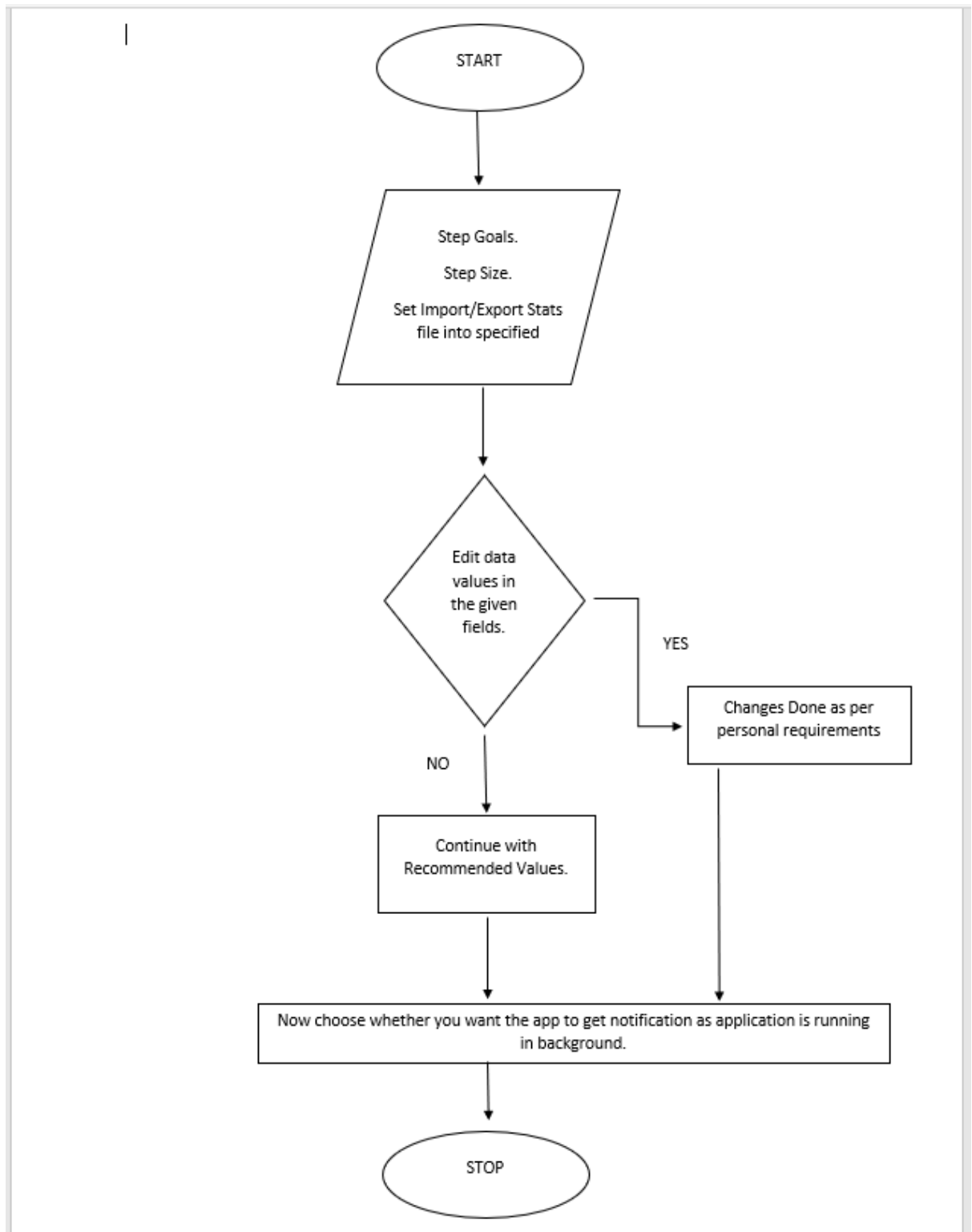
As a visual representation of data flow, flowcharts are useful in writing a program or algorithm and explaining it to others or collaborating with them on it. You can use a flowchart to spell out the logic behind a program before ever starting to code the automated process.

Flowchart symbols

Here are some of the common flowchart symbols:

Flowchart Symbol	Name	Description
	Process symbol	Also known as an "Action Symbol," this shape represents a process, action, or function. It's the most widely-used symbol in flowcharting.
	Start/End symbol	Also known as the "Terminator Symbol," this symbol represents the start points, end points, and potential outcomes of a path. Often contains "Start" or "End" within the shape.
	Decision symbol	Indicates a question to be answered — usually yes/no or true/false. The flowchart path may then split off into different branches depending on the answer or consequences thereafter.
	Input/output symbol	Also referred to as the "Data Symbol," this shape represents data that is available for input or output as well as representing resources used or generated. While the paper tape symbol also represents input/output, it is outdated and no longer in common use for flowchart diagramming.

Step Counter

START

Step Goals.

Step Size.

Set Import/Export Stats file into specified

Edit data values in the given fields.

YES

NO

Changes Done as per personal requirements

Continue with Recommended Values.

Now choose whether you want the app to get notification as application is running in background.

STOP

Step Counter

3.3.3 Sequence Diagram.

Sequence diagrams are a popular dynamic modeling solution in UML because they specifically focus on *lifelines*, or the processes and objects that live simultaneously, and the messages exchanged between them to perform a function before the lifeline ends.

A sequence diagram is a type of interaction diagram because it describes how—and in what order—a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as event diagrams or event scenarios.

Classes
Class roles describe the way an object will behave in context. Use the UML object symbol to illustrate class roles, but don't list object attributes.


Activation & Execution: Activation boxes represent the time an object needs to complete a task. When an object is busy executing a process or waiting for a reply message, use a thin gray rectangle placed vertically on its lifeline.
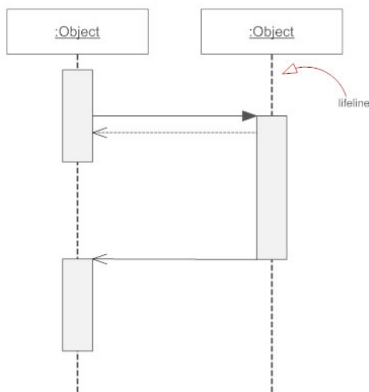

Messages
Messages are arrows that represent communication between objects. Use half-arrowed lines to represent asynchronous messages. Asynchronous messages are sent from an object that will not wait for a response from the receiver before continuing its tasks. For message types, see below.

Step Counter
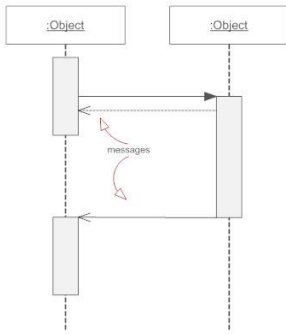
Lifelines
Lifelines are vertical dashed lines that indicate the object's presence over time.
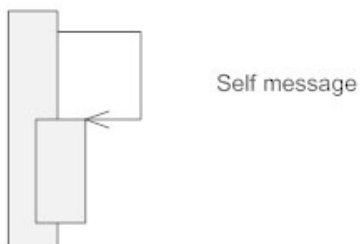

Synchronous-Messages
A synchronous message requires a response before the interaction can continue. It's usually drawn using a line with a solid arrowhead pointing from one object to another.
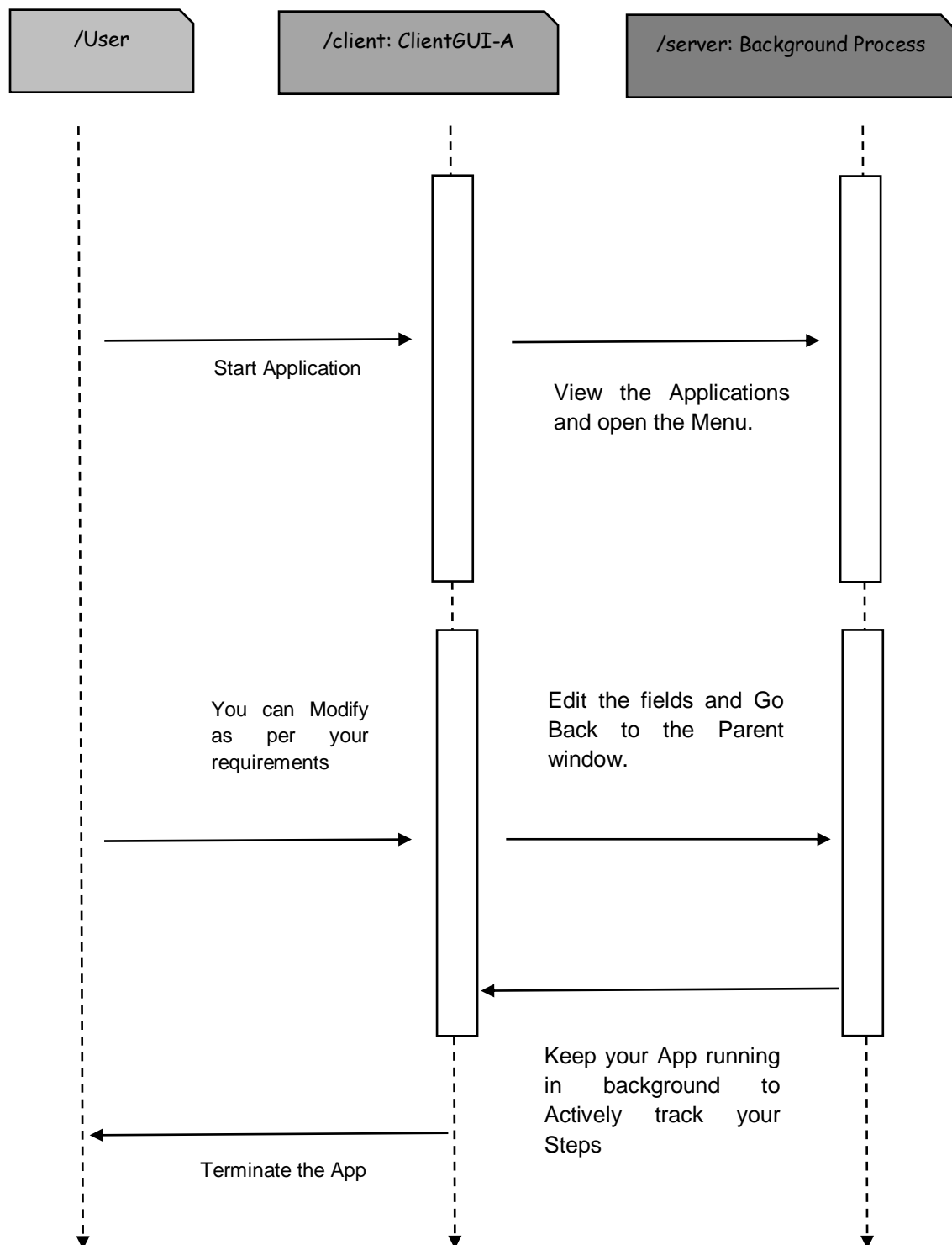

Self-Message
A message an object sends to itself, usually shown as a U shaped arrow pointing back to itself.

Step Counter

/User	/client: ClientGUI-A	/server: Background Process

Start Application

View the Applications and open the Menu.

You can Modify as per your requirements

Edit the fields and Go Back to the Parent window.

Keep your App running in background to Actively track your Steps

Terminate the App

3.3.4 Use – Case Diagram.

A use case is a written description of how users will perform tasks on your website. It outlines, from a user's point of view, a system's behavior as it responds to a request. Each use case is represented as a sequence of simple steps, beginning with a user's goal and ending when that goal is fulfilled.

Benefits of Use Cases

Use cases add value because they help explain how the system should behave and in the process, they also help brainstorm what could go wrong. They provide a list of goals and this list can be used to establish the cost and complexity of the system. Project teams can then negotiate which functions become requirements and are built.

Elements of a Use Case

Depending on how in depth and complex you want or need to get, use cases describe a combination of the following elements:

Actor – anyone or anything that performs a behavior (who is using the system)

Stakeholder – someone or something with vested interests in the behavior of the system under discussion (SUD)

Primary Actor – stakeholder who initiates an interaction with the system to achieve a goal

Preconditions – what must be true or happen before and after the use case runs.

Triggers – this is the event that causes the use case to be initiated.

Main success scenarios [Basic Flow] – use case in which nothing goes wrong.

Alternative paths [Alternative Flow] – these paths are a variation on the main theme. These exceptions are what happen when things go wrong at the system level.
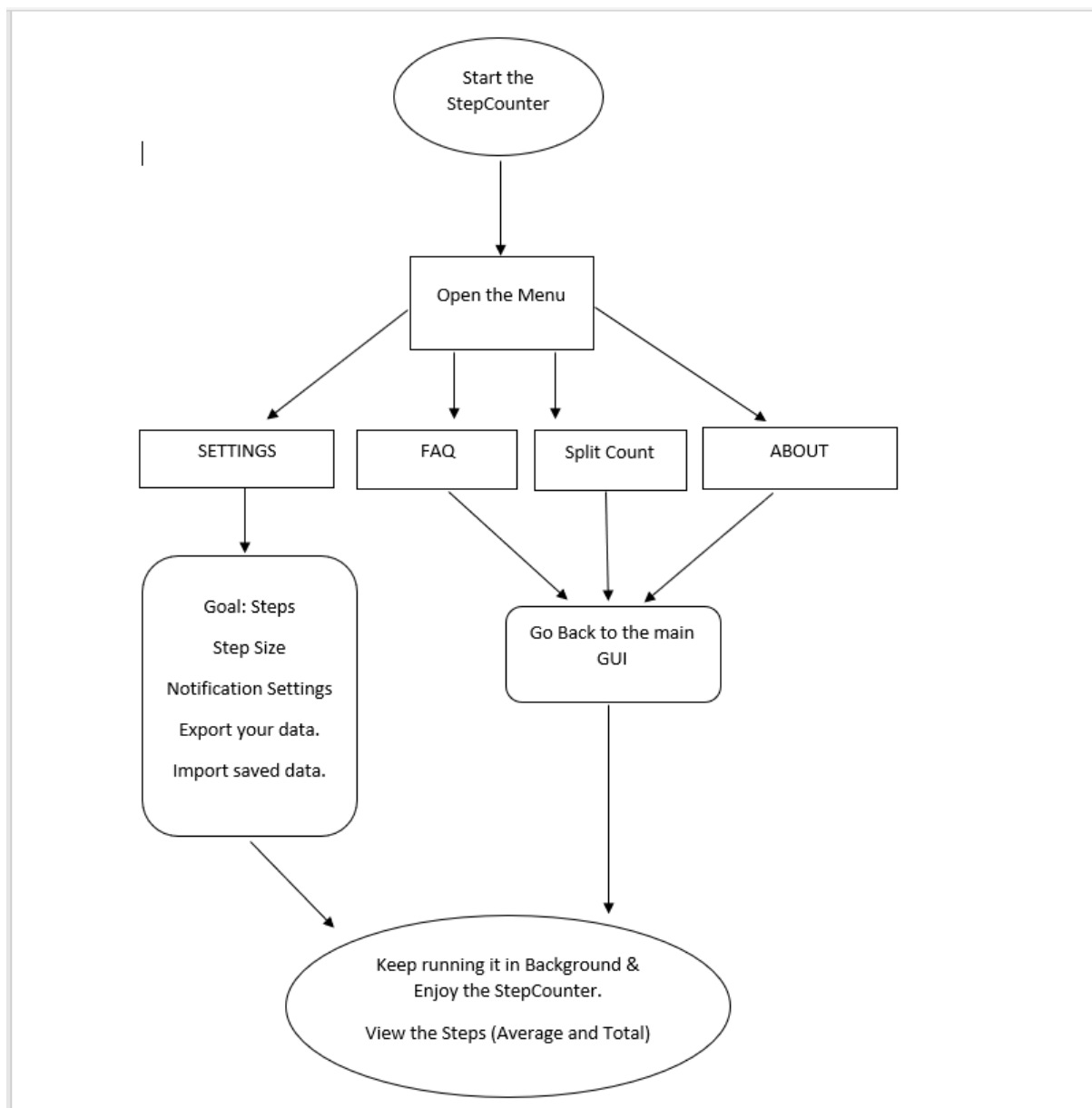
Step Counter
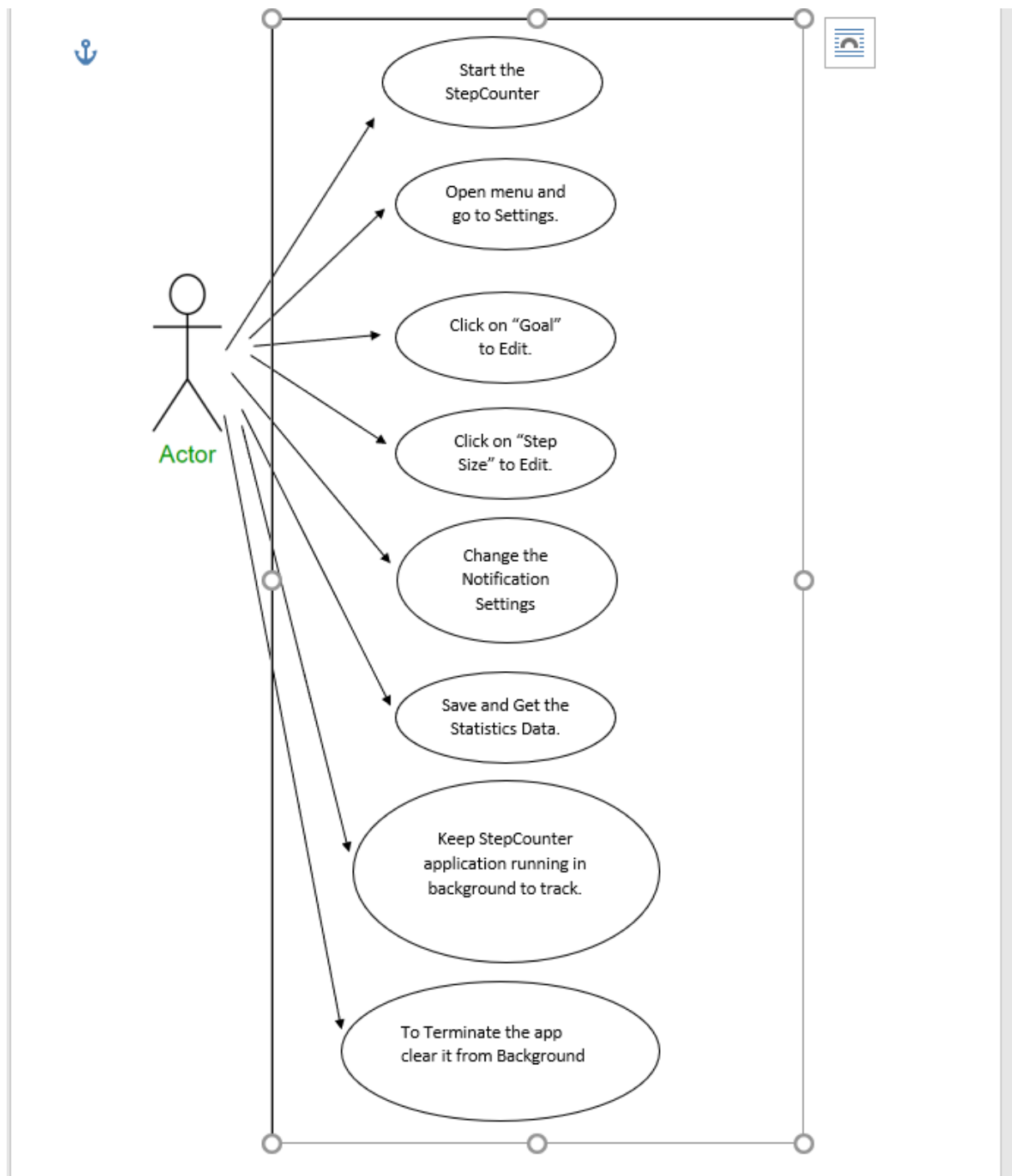
Step Counter

3.4 DFD (Data Flow Diagram).

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination.

DFD Level 0 is also called a Context Diagram. It's a basic overview of the whole system or process being analyzed or modeled. It's designed to be an at-a-glance view, showing the system as a single high-level process, with its relationship to external entities.

Step Counter

3.5 Algorithm.

Main Steps to make the Android applications.

1. START.
2. Design the basic User Interface for your Android application.
3. Drag and Drop the Components/items from the inbuild Palette.
4. Next is assigning them to specify the task and link to the Android Manifest file.
5. Collect the different modules to integrate it with the desired feature for App.
6. Following are the Modules and packages designated for the app which are as follows Dialog_Split, and Statistics, Fragment_Overview and Settings, Logger, BootReciever, Database, SensorListener, and ShutdownReciever.
7. Now we will add the various Icon for compatible version of your android device into Drawable folder.
8. Layout is next which is main part of our App which keeps it as much interactive as possible for user which contains all xml-based files.
9. Now we designed in app MENU in sequence as we designed to build on.
10. Now we extract all the String values on to the one file under strings.xml file which are bridge between Activity_Main and Android Manifest.
11. Now we designed Settings and its action which user will invoke by clicking and make all the details available.
12. There are few auto-generated files which creates in android studio as we develop the app such as all the Gradle scripts, few RES files, Java buildConf files.
13. Now finally we add the icon packs for our android application which is viewed in the app drawer of user's android device.
14. Open the application interact, set some goals, customize accordingly and enjoy the app as you will now able to know how many steps have you walked in a day as you eave the app in background running.
15. Easily you can stop or terminate the app by simply clearing it for recent task manager after use.
16. STOP.

4. System / Mobile Implementation.

The implementation of StepCounter is easily implemented on any android device using specified android version.

By using JAVA language within the Android Studio has greatly improved the performance and efficient of creating the apps and allocating specific packages, easily access to Gradle scripts and also to structure the files and to provide sequence-based files in android studio as we go on creating in the building of application

As the Graphical User Interface is the main key component for developing the app very eloquently that the interaction is easier to get it as per our needs and also if you are familiar to java-based Android studio it is easier to quickly fetch and perform tasks without spending time onto UI features.

As we have put the FAQ and About information into the app you will be easily able to use and understand the features and use it by any user without deep knowledge of the Programming language.

By using Java and the Android studio's virtual device development is quite easy to do and execute the files and any error might occur it is easily solved by showing where exactly the line(s) are and suggestions are given to help to recognize it and fix it by developers' permission.

Also, the source code in the Android studio is quite reliable and structured based it forms a tree which includes the files as a node and root directory the main head file which consists the surface value of the existing file. As user does not need the access of source code it is safely maintained in the app src files and it is non-editable by developers to avoid any changes to or overwrite anything to it.

Once the App is created it is easily downloaded to your Android device with selecting appropriate version of your android to run it so there are many options to choose from and also it is easily sharable and run into numerous devices to enjoy its features and applications on to your own needs.
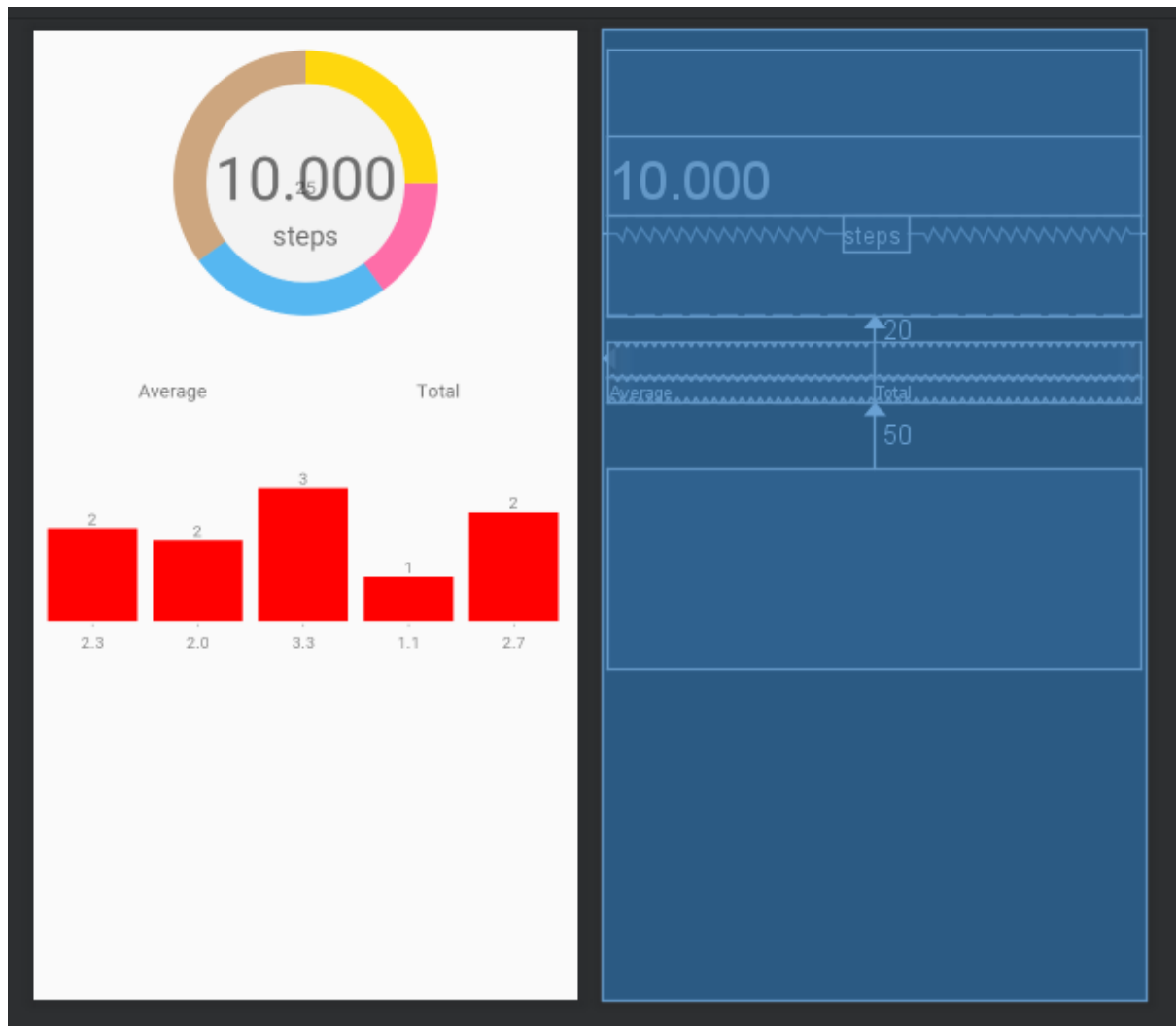
5. Results.
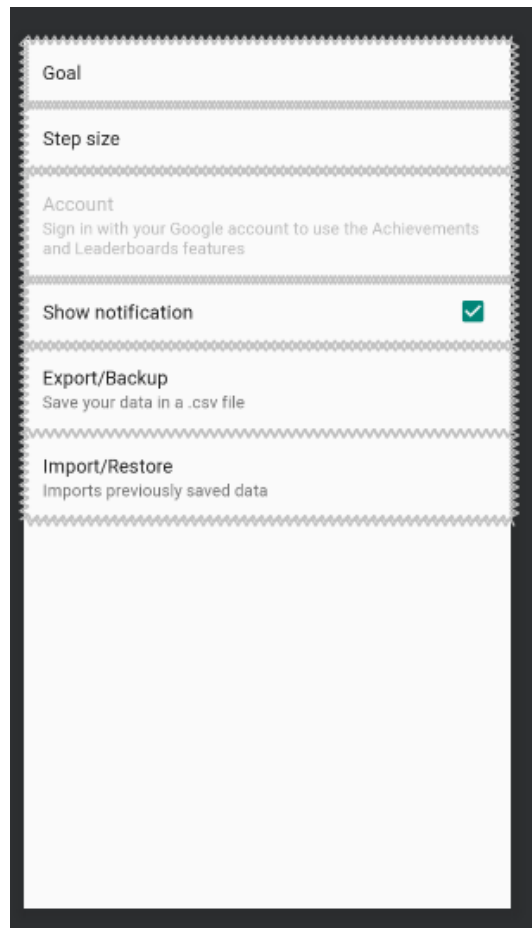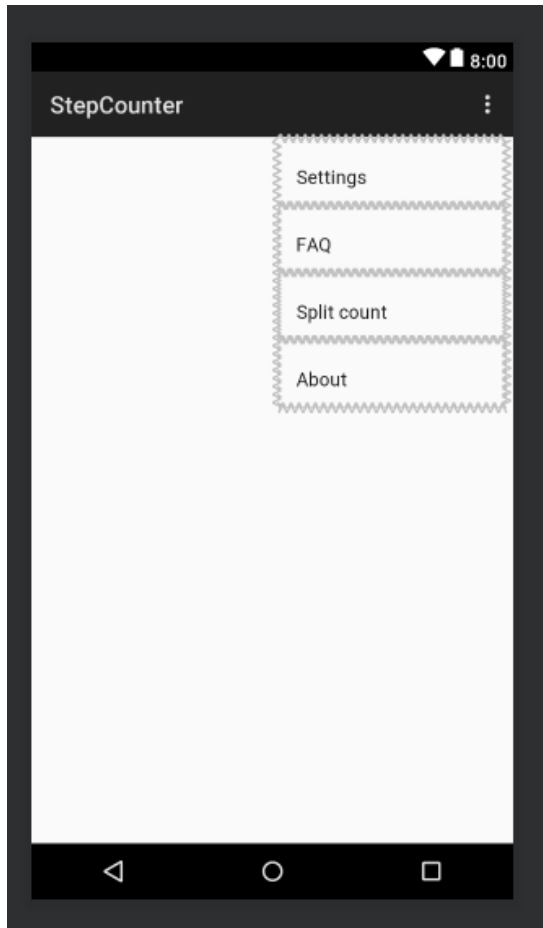## 5.1 Preview/Screenshots.

It is a blueprint of what the GUI will look like.

A) Following are the Interfaces you see when you run in the Android Device App. In Android Studio Developer Software.
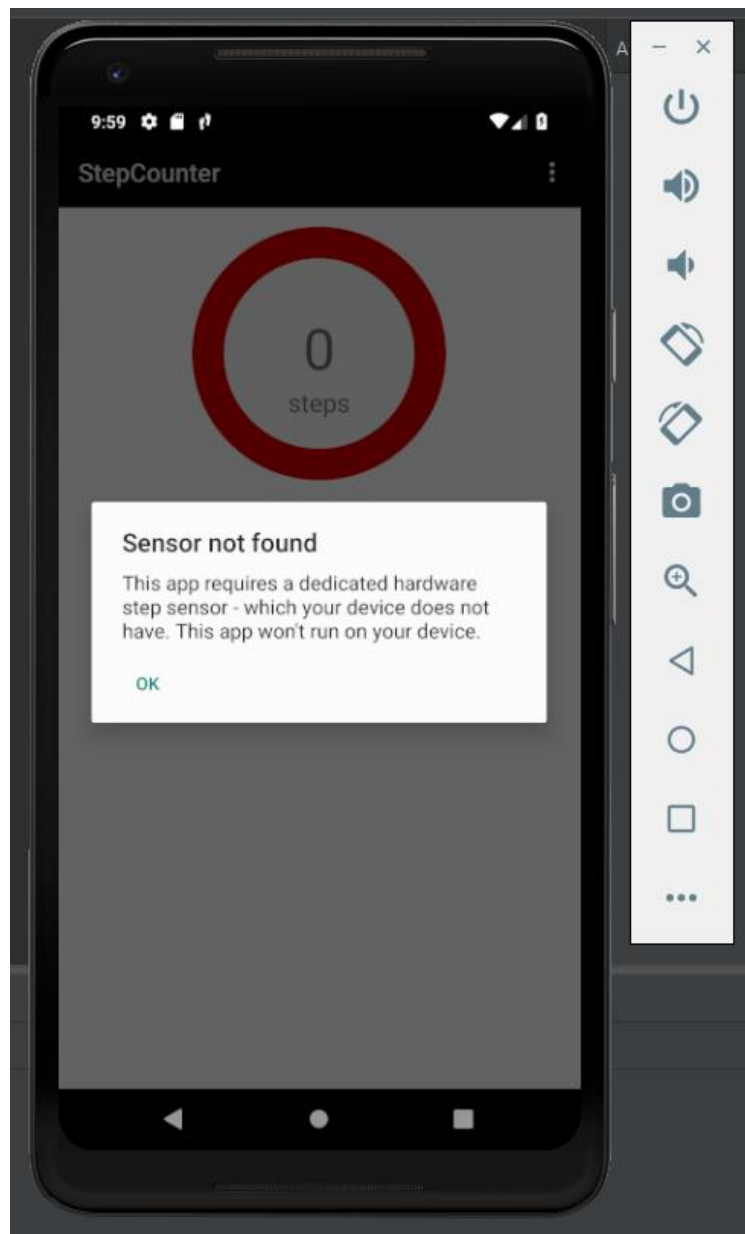
Step Counter

B) Design of the GUI in the Android Studio to set the components.

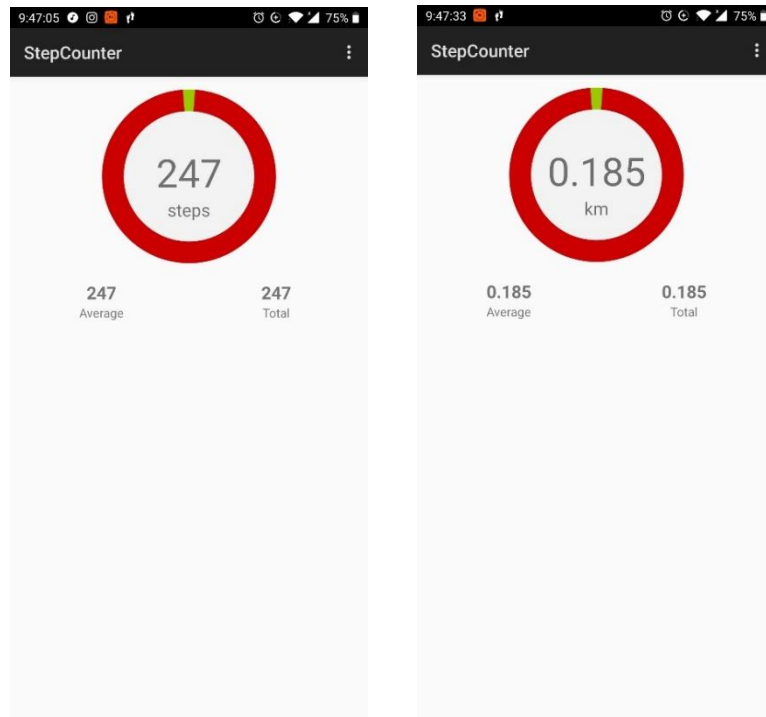TYBSc (CS) Sem VI   Roll No: 21

Step Counter

C) When It is open into Android Studio as it has built in virtual device which does not contain any sensor it won't work and will show following message.
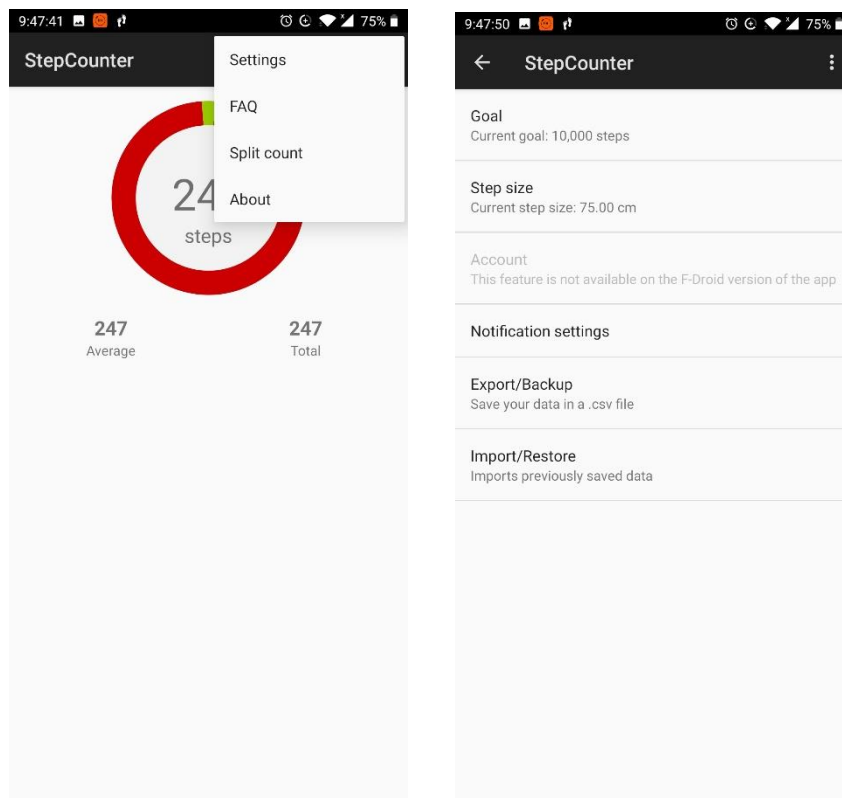
Step Counter

D) Following are the Snippets from StepCounter Application onto your Android Device which user can experience. User Interface of the 1st Screen.
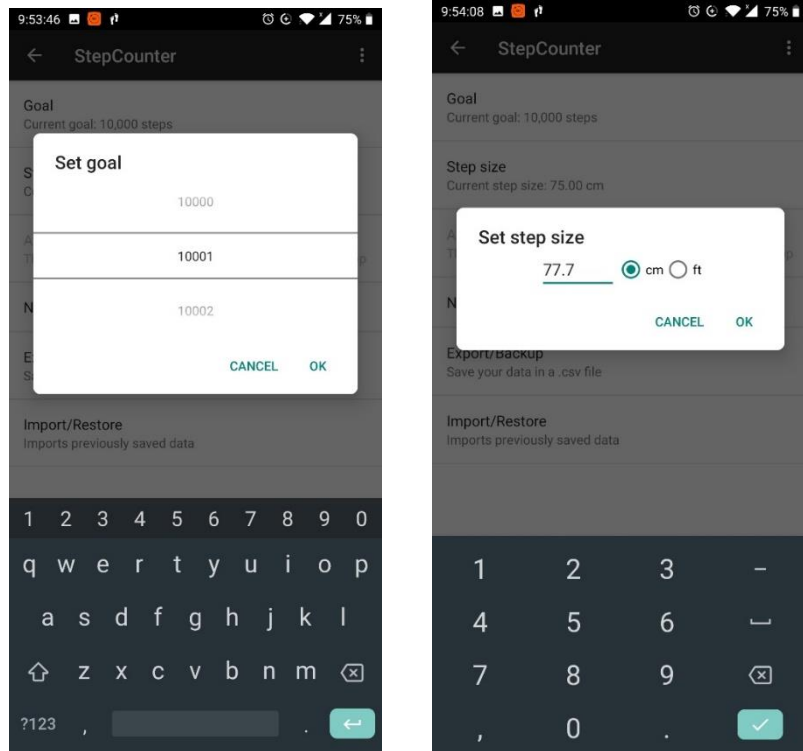

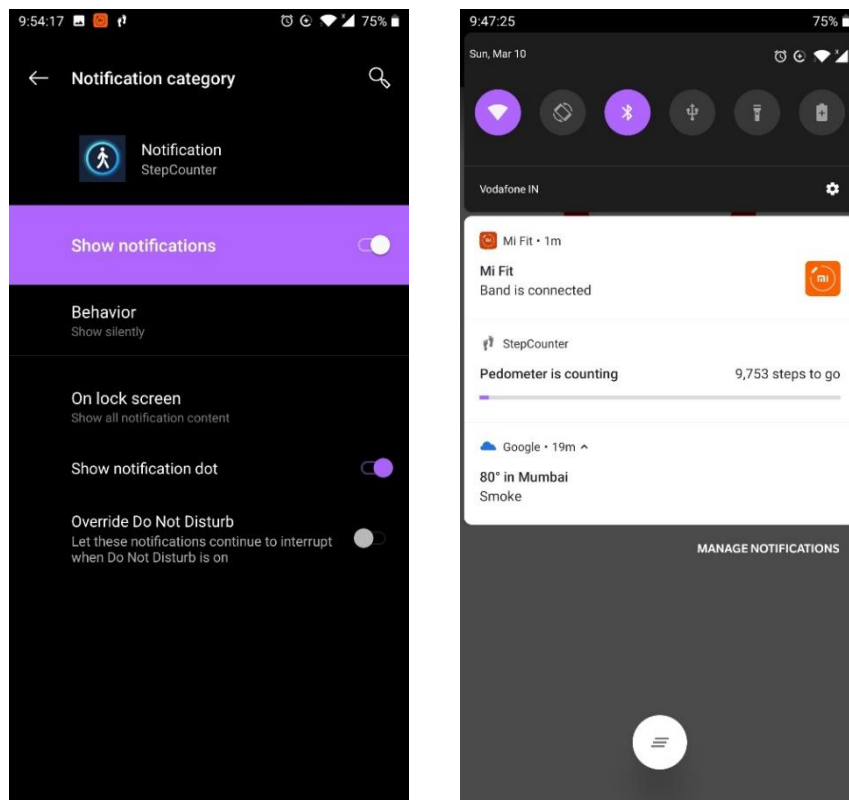E) Now, you may click the Menu --- and choose appropriate option.

Step Counter

F) You can able to change the Settings and set it as per your needs/goals.
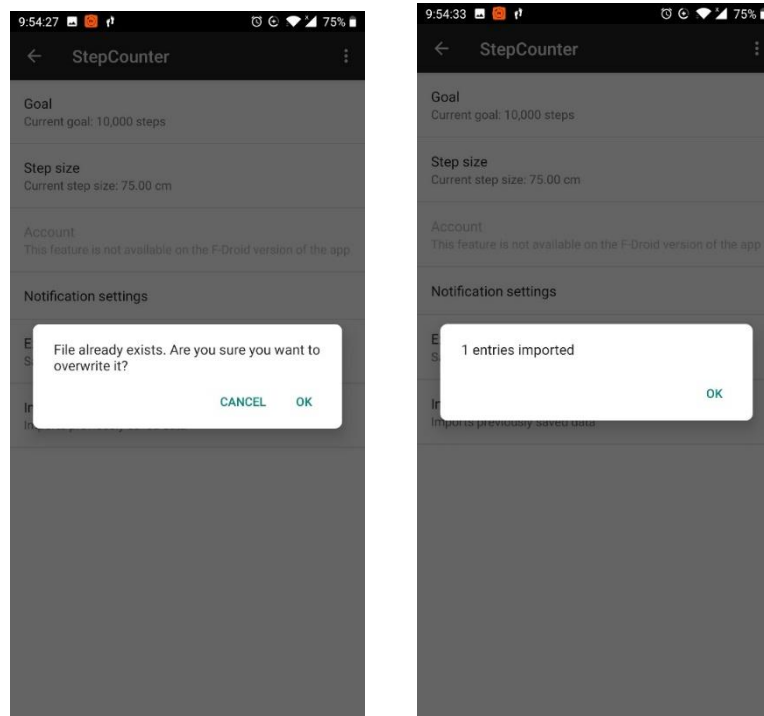

G) If you want Notifications you may check it or uncheck to stay the app innotifcation in the Notification Tray.
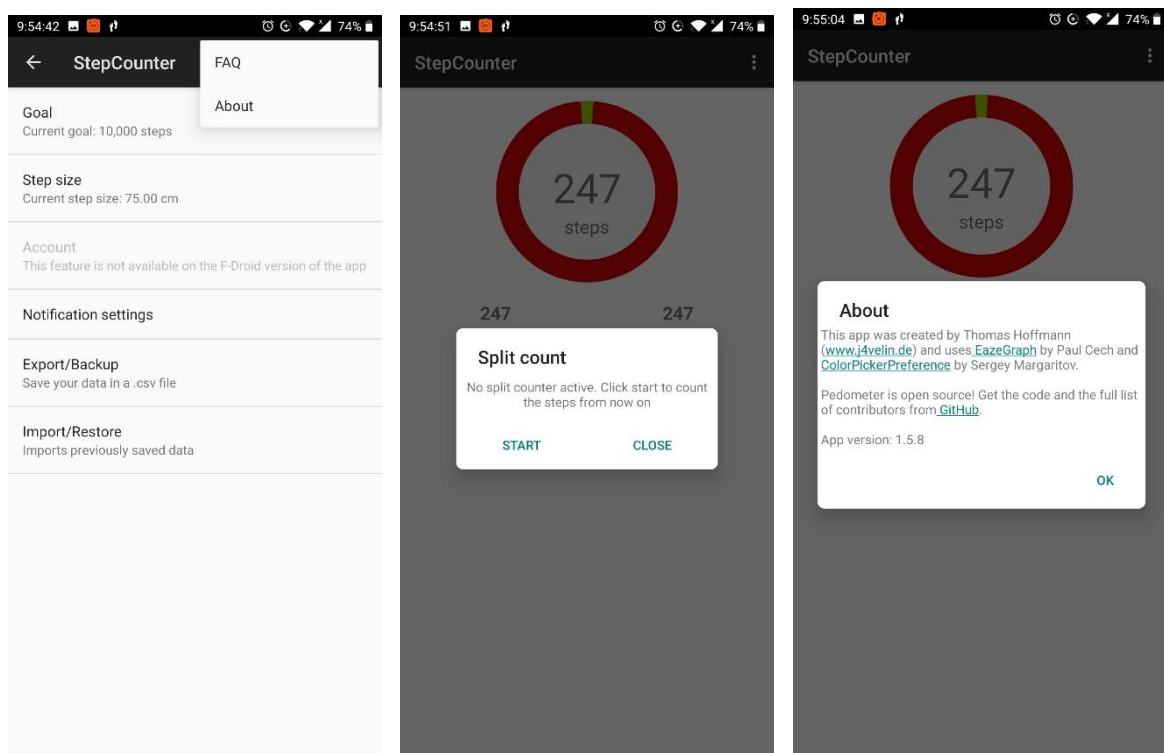
Step Counter

H) To Export or Import the saved file you may select it.
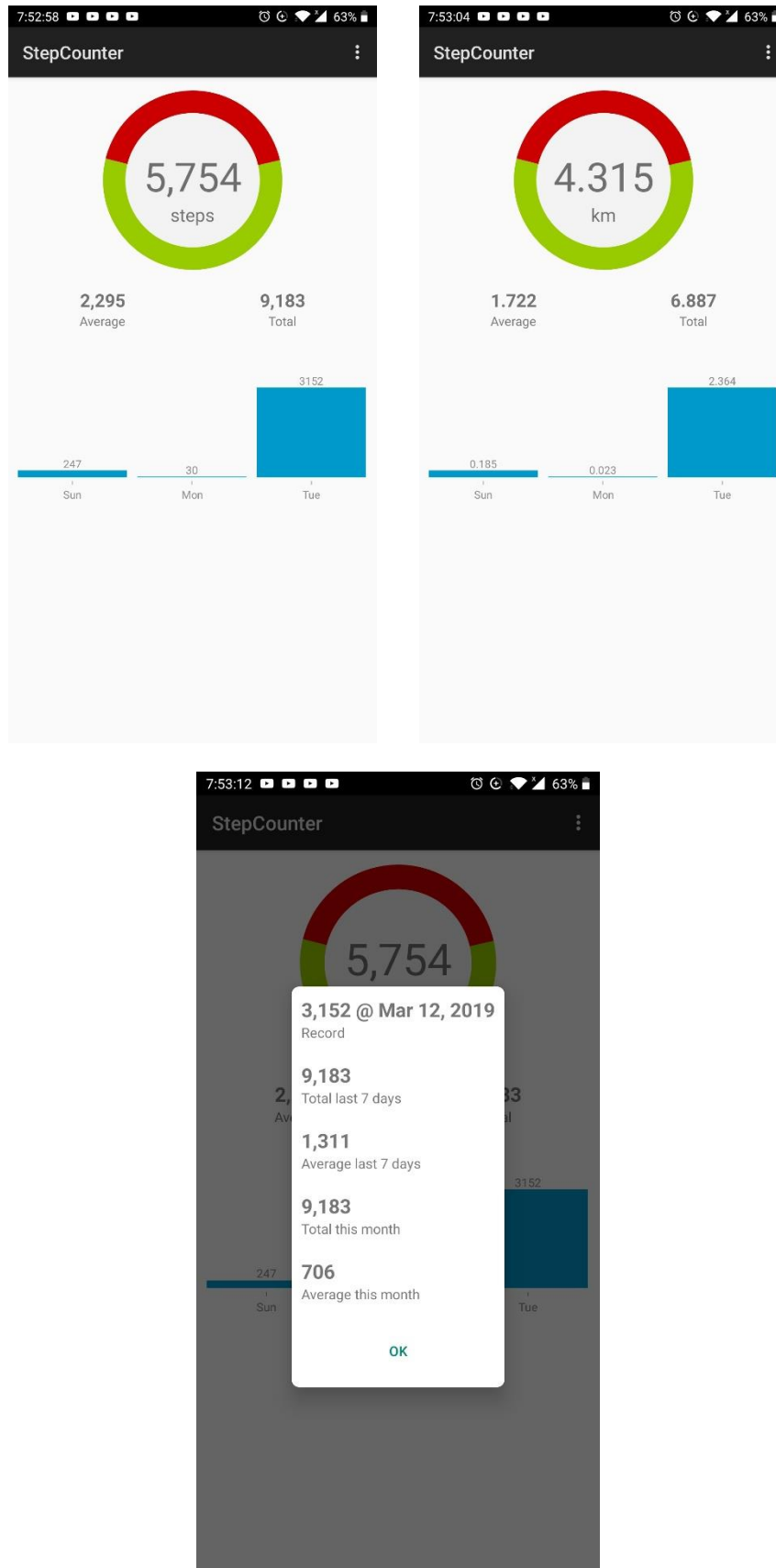

I) If you want to know more about the StepCounter App and FAQ too can be viewed here.

Step Counter

Live Examples of StepCounter App in Android 9.0

TYBSc (CS) Sem VI   Roll No: 21

Step Counter

5.2 Reports

- A StepCounter has 3 major goals/objectives which are as follows:
- A pedometer tells you how many steps you take. This can help motivate you to walk.
- A pedometer can help you set specific goals for walking. For example, you can set goals for walking a certain number of steps during an activity or throughout the day. You can easily track your goals and adjust them.
- A pedometer can remind you to walk more. A quick check may show that you need more steps to meet your goal for the day. You may even find it fun to add more steps to such things as grocery shopping or returning a book to the library.
- Find your activity level.
- For the first week, follow your usual routine. Don't change how active you are. Write down your steps each day in a step diary. This will give you a starting record of how active you are.
- Look at this record for the first week to see where you can add steps to your daily routine.
- You may easily reach your goals as you start and get into the fitness and active life for your health benefits and to motivate yourself and don't be lazy.
- Set a goal for the second week. For example, try to add 300 to 500 steps each day or every other day. Or add 10% to how many steps you walked the first week. Do what works best for you.

Step Counter

5.3 References.

For source code and GUI

- ➢ Java tutorials point based on Android Activity.
- ➢ GitHub
- ➢ Android Studio.
- ➢ Components and user interface models and packages from various other website to complete the App development.

Information about StepCounter and Sensors

- ➢ Wikipedia.
- ➢ https://www.cardiosmart.org/~/media/Documents/Fact%20Sheets/en/abk5253.ashx
- ➢ Understanding the fundamentals of Pedometer.
- ➢ https://developer.android.com/guide/topics/sensors

Use the step counter sensor

The step counter sensor provides the number of steps taken by the user since the last reboot while the sensor was activated. The step counter has more latency (up to 10 seconds) but more accuracy than the step detector sensor. The following code shows you how to get an instance of the default step counter sensor:

KOTLIN   JAVA

```java
private SensorManager sensorManager;
private Sensor sensor;
...
sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
sensor = sensorManager.getDefaultSensor(Sensor.TYPE_STEP_COUNTER);
```

To preserve the battery on devices running your app, you should use the `JobScheduler` class to retrieve the current value from the step counter sensor at a specific interval. Although different types of apps require different sensor-reading intervals, you should make this interval as long as possible unless your app requires real-time data from the sensor.

TYBSc (CS) Sem VI   Roll No: 21

Step Counter

Use the step detector sensor

The step detector sensor triggers an event each time the user takes a step. The latency is expected to be below 2 seconds. The following code shows you how to get an instance of the default step detector sensor:

```
KOTLIN    JAVA

private SensorManager sensorManager;
private Sensor sensor;
...
sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
sensor = sensorManager.getDefaultSensor(Sensor.TYPE_STEP_DETECTOR);
```

Work with raw data

The following sensors provide your app with raw data about the linear and rotational forces being applied to the device. In order to use the values from these sensors effectively, you need to filter out factors from the environment, such as gravity. You might also need to apply a smoothing algorithm to the trend of values to reduce noise.

Making documentation and Diagram presentation

➢ Lucid chart.com
➢ Notes on creating Gantt chart and PERT chart, & UML Diagrams.
➢ Tutorials Point

6. Conclusions and future scope.

There is no in-app update which is safe to say it is a one-time installation process and which contains about 10-12 mb maximum of your storage in your device. Also, you are able to share and use it in any device compatible which now-a-days are.

StepCounter is easy to use, someone without any programming knowledge will able to use. Everything you need to know about the StepCounter is inbuild in the app itself under the "About" menu which gives you brief description about the application.

You should keep your application running in your background to work and track the steps via Sensors are actively tracked in the app and database is logged in simultaneously. It is fun to see how much you walk each day/ step taken each day to show how active you are it is also visually well represented in the most aesthetic way to user.

All the data is collected and can be saved in an Xcel sheet which is graphically view in your mobile device.

This application plays important role to your fitness and general health which can be used to track the steps of the day and eat accordingly and when the steps are less you may not eat that much which balanced the Calories and you'll be able to maintain your weight.