Project 2 Report

*Yaohan Xing  
Student of COMP 2430  
Lakehead University  
Thunder Bay, ON, CANADA*[*yxing1@lakeheadu.ca*](mailto:yxing1@lakeheadu.ca)

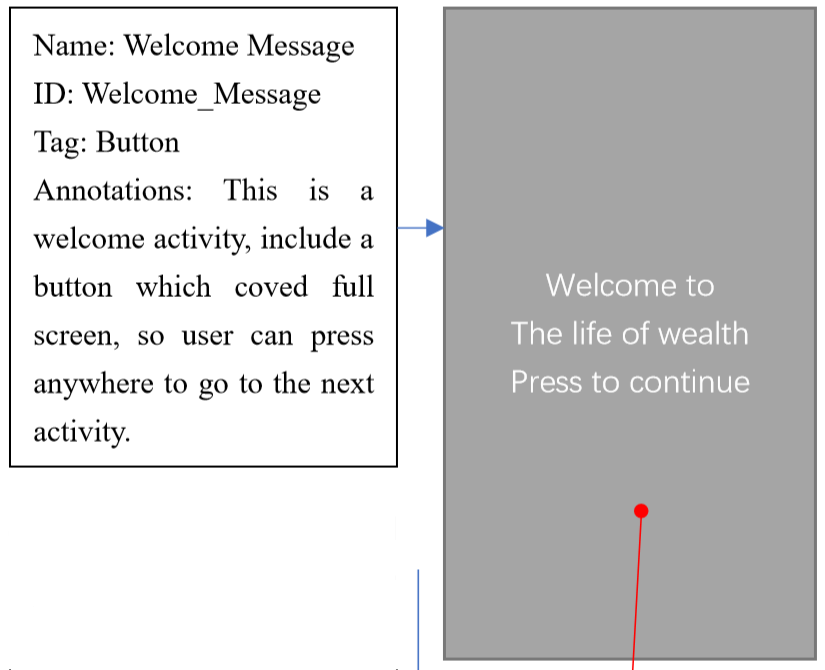
This document is the report for the project II of COMP 2430.

# Create A Wireframe

The wireframe will show the flow chart of application. There are the layout of application and what will happen when press the buttons in the application.

## Welcome Activity

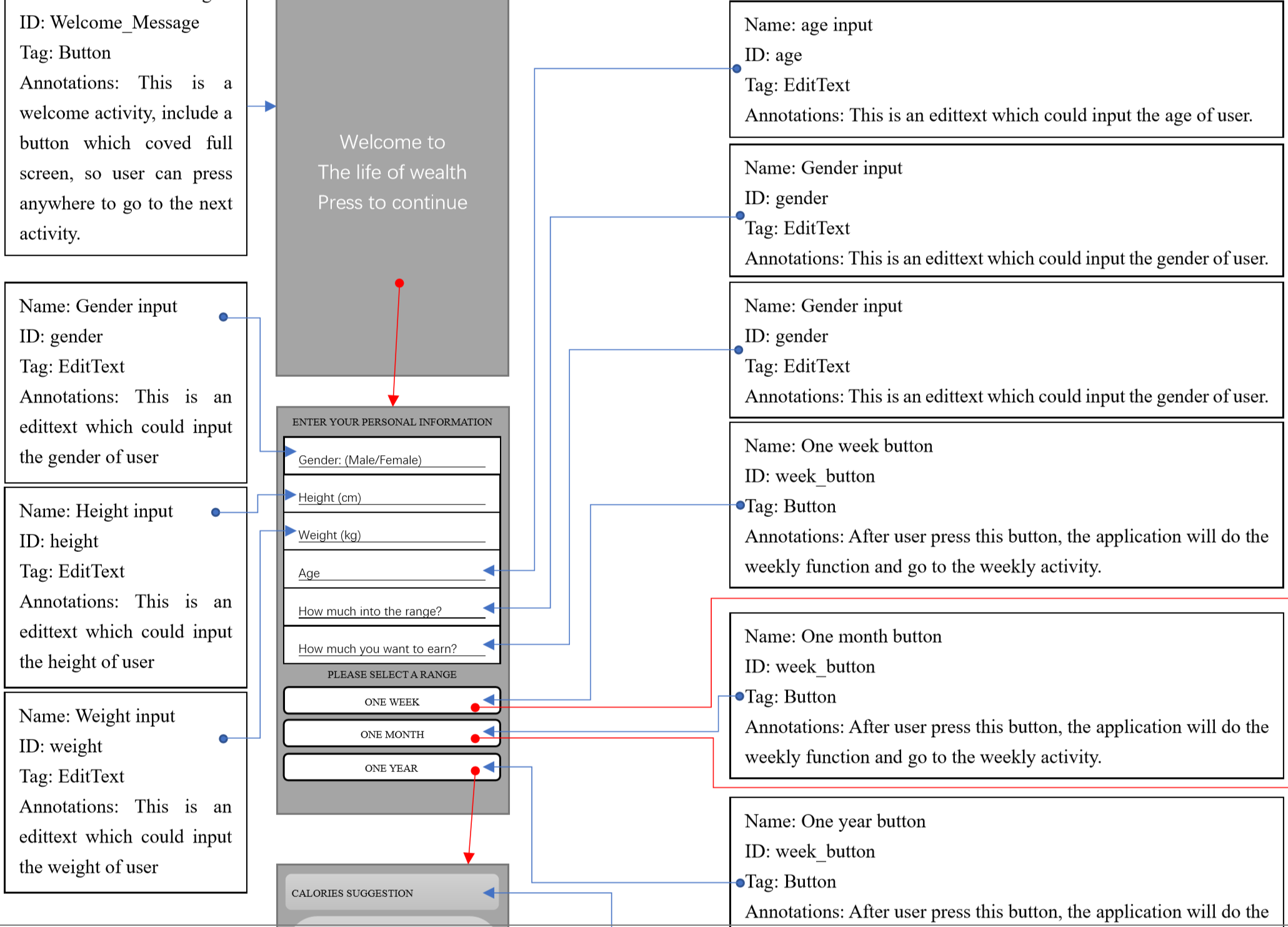
The welcome page will show the welcome message like “Welcome to the life of wealth, Press to continue.” There is a button which cover the full screen so user could touch anywhere to continue.



## Information Activity

The second activity will let user input their personal information such as gender, weight, height, age and how much money for the money plan. There are three buttons that named week, month, and year. They will start different function to calculate the calories for daily use and make the money plan.

These three buttons will start the three different activity which is weekly activity, monthly activity and yearly activity.

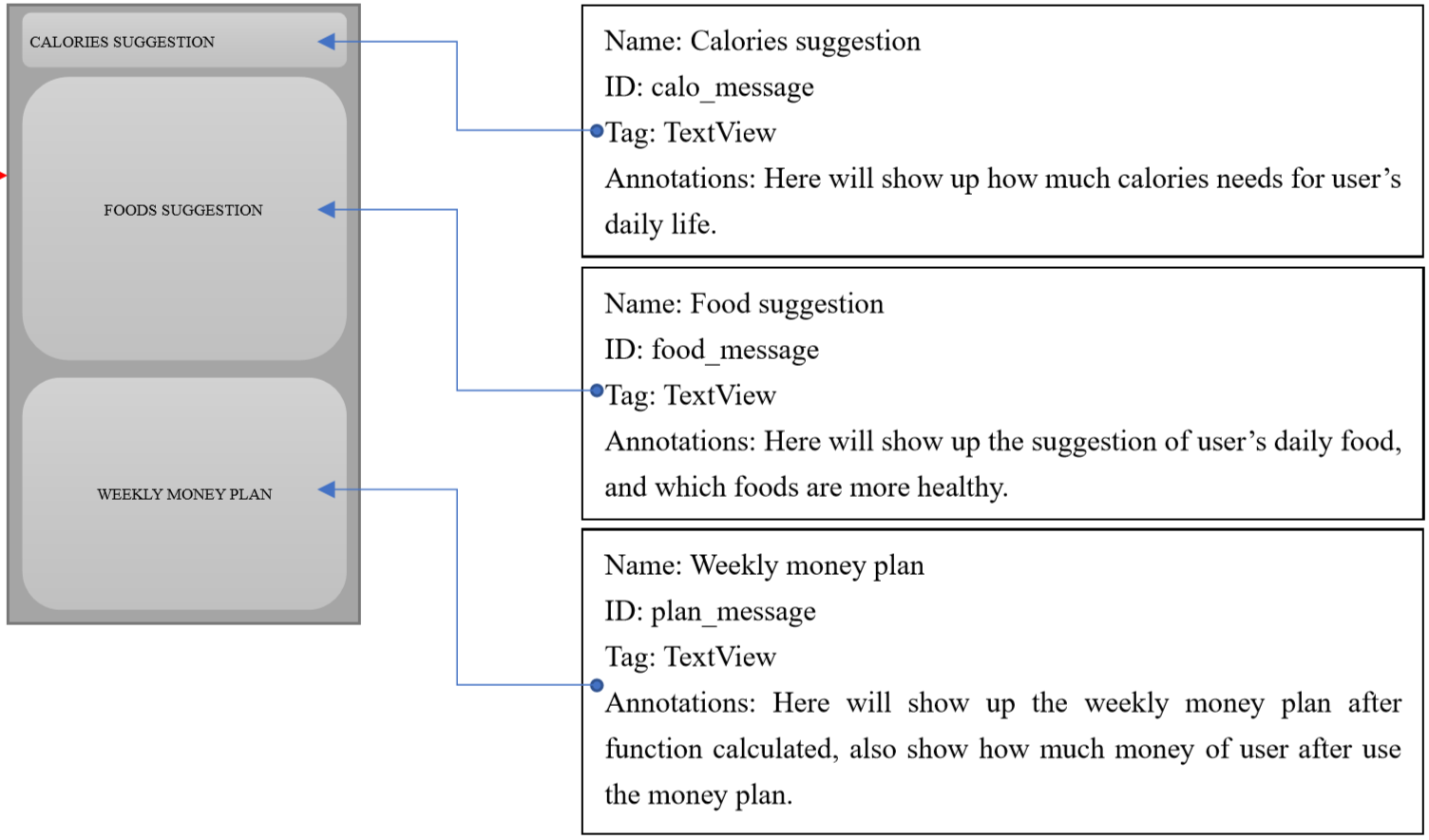


## Money plan Activity

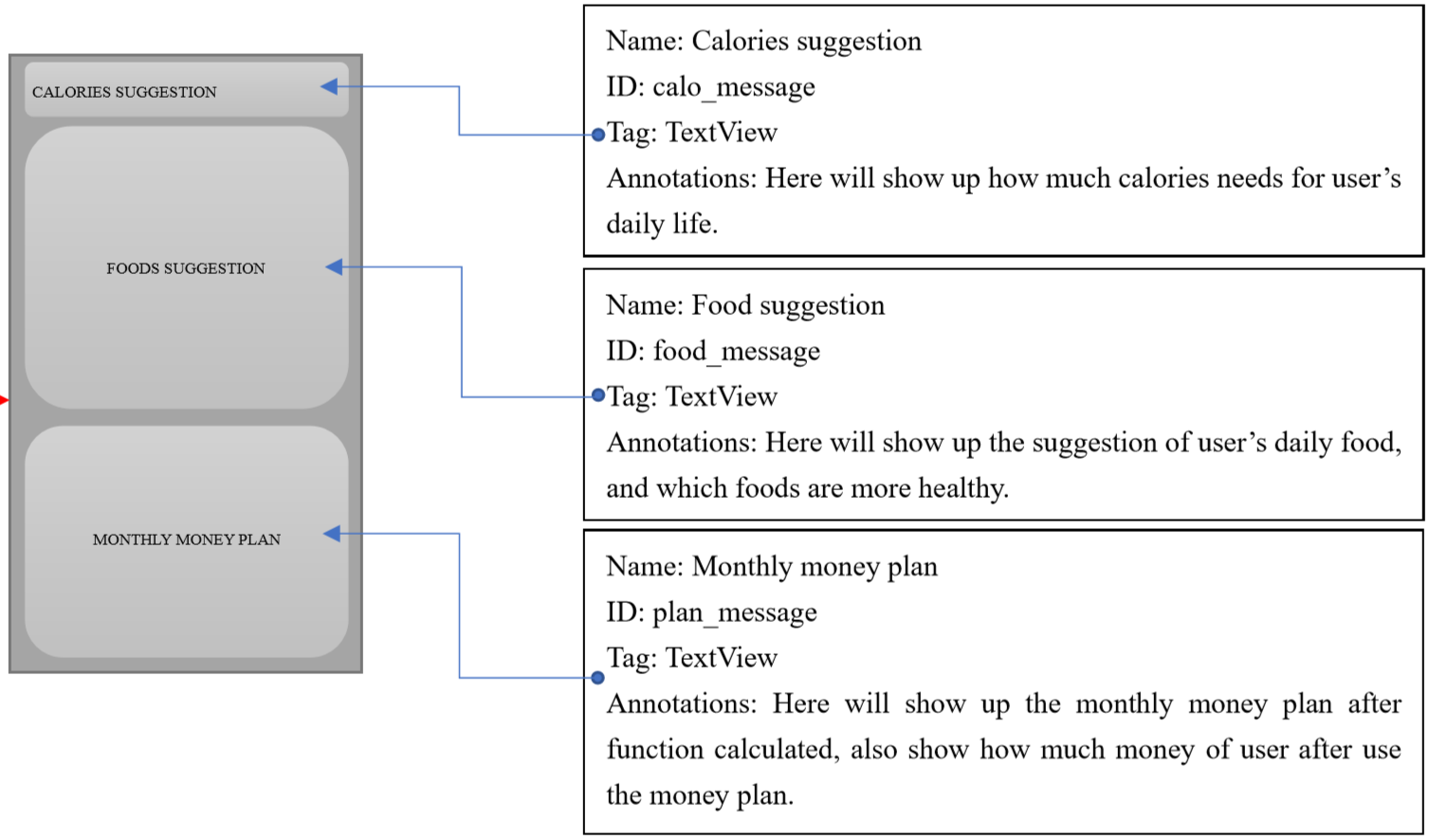
Three buttons from the information activity will start three different activity that for weekly, monthly and yearly money plan.

There are three text box that will show the calories for daily use, food suggestion and money plan. They will get the intent from the information activity that user inputted and calculate the results and show.

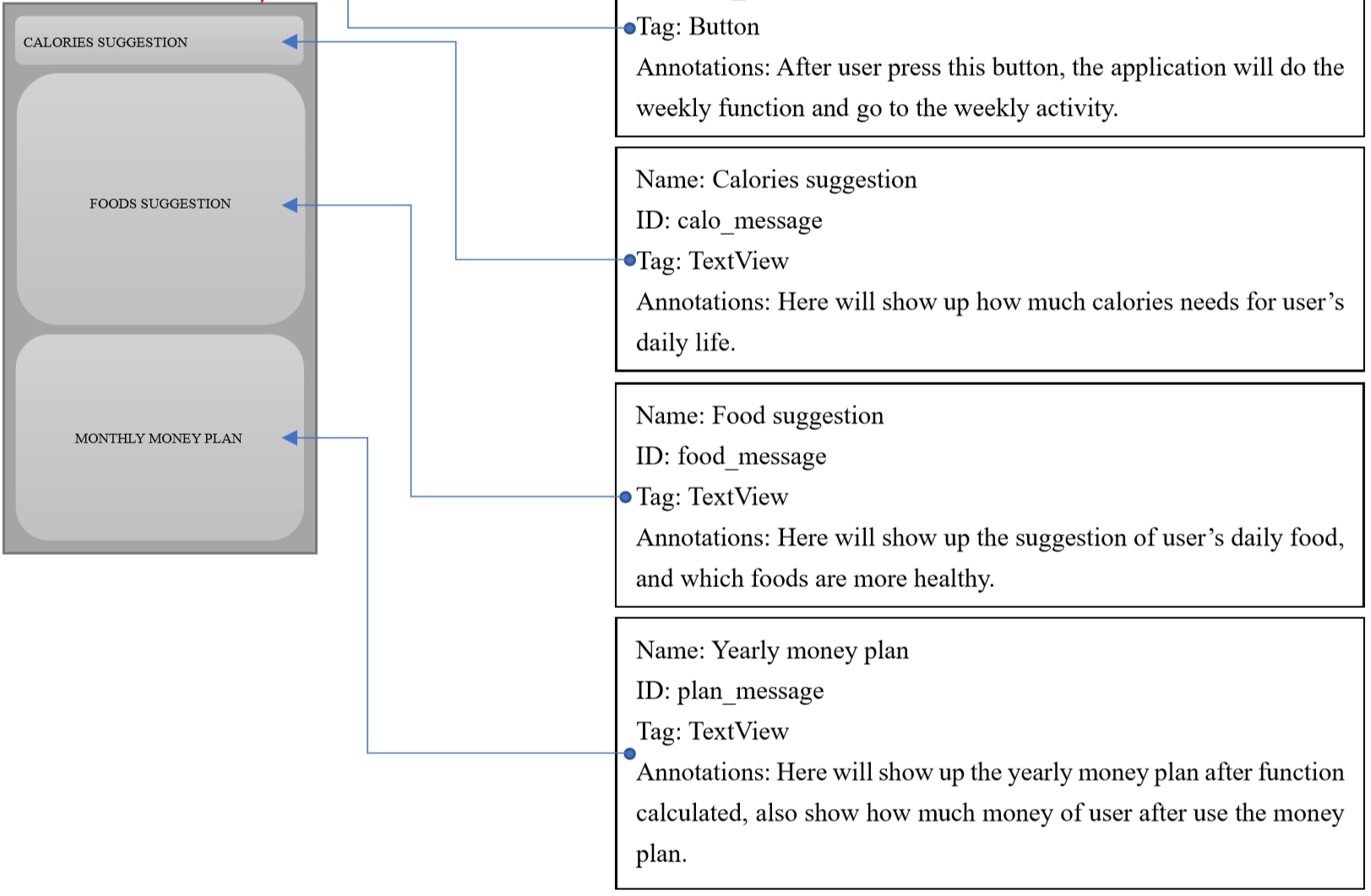
Weekly plan activity.



Monthly plan activity.



Yearly plan activity.



# Creat The Tappable Prototype

Tippable prototype should allow the users to navigate through the application’s activity and it is expected to have clickable buttons, typeable edit views, dummy displays and images. However, it won’t have a matured or full-fledged functionality.

1. <Button
2. android:id="@+id/First\_BTN"
3. android:layout\_width="match\_parent"
4. android:layout\_height="match\_parent"
5. android:text="@string/Welcome\_Message"
6. android:textColor="@color/white"
7. android:textSize="30sp"
8. android:background="@color/dark\_red"></Button>

# Create A New Project

## Choose my project

Open the Android Studio and create a new project. Select "Basic Activity" under the "Phone and Tablet" tab, then press the "Next" button.

## Configure my Project

On the "Configure your project" tab, I named my project to Currency Conversion. Then, select "API 15: Android 4.0.3(IceCreamSandwich)" for "Minimum API level", then press the "Finish" button.

## Set the Color.xml

Open the color.xml in the values folder. Then named color dark red (#D00020), light blue (#00BCD4), white (#FFFFFF) and gray (#757575).

1. <?xml version="1.0" encoding="utf-8"?>
2. <resources>
3. <color name="colorPrimary">#008577</color>
4. <color name="colorPrimaryDark">#00574B</color>
5. <color name="colorAccent">#D81B60</color>
6. <color name="light\_blue">#00BCD4</color>
7. <color name="dark\_red">#D00020</color>
8. <color name="white">#FFFFFF</color>
9. <color name="gray">#757575</color>
10. </resources>

# Design The Welcome Activity

The first step is designing the welcome activity, which have the welcome message and the button to the next activity.

## Set up the layout file

Firstly, change the layout file to a linearlayout file, and set arrangement to vertical by “android:orientation="vertical".”

## Add an Button

Inset a button and set it cove the full screen so user could press anywhere to continue to the next activity.

Named the button “WELCOME TO THE LIFE OF WEALTH PRESS TO CONTINUE” as welcome message to user.

# Create Information Activity

Information activity will ask user to enter personal information like gender, height, weight, age, and how much money for the money plan.

## Add an message TextView

The TextView will show the tip message “Please enter your personal information” to ask user to input their information.

1. <TextView
2. android:id="@+id/shown\_message"
3. android:layout\_width="match\_parent"
4. android:layout\_height="100dp"
5. android:background="@color/light\_blue"
6. android:gravity="center"
7. android:text="@string/ShownMessage"
8. android:textColor="@color/white"
9. android:textSize="30sp"/>

## Add a gender input EditText

This EditText could ask user to input their gender of the application. However, for the tappable prototype, it will not have the functionality.

1. <EditText
2. android:id="@+id/gender"
3. android:layout\_width="match\_parent"
4. android:layout\_height="wrap\_content"
5. android:hint="@string/gender\_male\_female"
6. android:inputType="text"
7. android:textSize="20sp"
8. android:importantForAutofill="no"
9. android:textColor="@color/gray"/>

## Add a height input EditText

This EditText could ask user to input their height of the application. However, for the tappable prototype, it will not have the functionality.

1. <EditText
2. android:id="@+id/height"
3. android:layout\_width="match\_parent"
4. android:layout\_height="wrap\_content"
5. android:hint="@string/height\_cm"
6. android:inputType="number"
7. android:textSize="20sp"
8. android:importantForAutofill="no"
9. android:textColor="@color/gray"/>

## Add a weight input EditText

This EditText could ask user to input their weight of the application. However, for the tappable prototype, it will not have the functionality.

1. <EditText
2. android:id="@+id/weight"
3. android:layout\_width="match\_parent"
4. android:layout\_height="wrap\_content"
5. android:hint="@string/weight\_kg"
6. android:inputType="number"
7. android:textSize="20sp"
8. android:importantForAutofill="no"
9. android:textColor="@color/gray"/>

## Add a age input EditText

This EditText could ask user input their age of the application. However, for the tappable prototype, it will not have the functionality.

1. <EditText
2. android:id="@+id/age"
3. android:layout\_width="match\_parent"
4. android:layout\_height="wrap\_content"
5. android:hint="@string/age"
6. android:inputType="number"
7. android:textSize="20sp"
8. android:importantForAutofill="no"
9. android:textColor="@color/gray"/>

## Add a intial funding input EditText

This EditText could ask user to input their initial money of the application. However, for the tappable prototype, it will not have the functionality.

1. <EditText
2. android:layout\_width="match\_parent"
3. android:layout\_height="wrap\_content"
4. android:id="@+id/investmentNumber"
5. android:hint="@string/investmentMessage"
6. android:inputType="number"
7. android:textSize="20sp"
8. android:importantForAutofill="no"
9. android:textColor="@color/gray"/>

## Add a goal input EditText

This EditText could ask user to input their dream number about money of the application. However, for the tappable prototype, it will not have the functionality.

1. <EditText
2. android:layout\_width="match\_parent"
3. android:layout\_height="wrap\_content"
4. android:id="@+id/goalNumber"
5. android:hint="@string/goalMessage"
6. android:inputType="number"
7. android:textSize="20sp"
8. android:importantForAutofill="no"
9. android:textColor="@color/gray"/>

## Add a tip message textview

This text will asp user to select an option for ranges from one week, one month and one year.

1. <TextView
2. android:layout\_width="match\_parent"
3. android:layout\_height="wrap\_content"
4. android:id="@+id/optionMessage"
5. android:text="@string/optionMessage"
6. android:gravity="center"
7. android:background="#00BCD4"
8. android:textColor="@color/white"
9. android:textSize="30sp"/>

## Add one week button

This button will start the activity of weekly plan.

1. <Button
2. android:id="@+id/week\_button"
3. android:layout\_width="match\_parent"
4. android:layout\_height="wrap\_content"
5. android:text="@string/oneWeek"
6. android:textColor="@color/gray"/>

This button will be set on click function which start weekly plan activity.

1. week\_button.setOnClickListener {
2. weeklyFunction()
3. }
4. fun weeklyFunction(){
5. val intent = Intent(**this**,weeklyActivity::**class**.java)
6. startActivity(intent)
7. }

## Add one year button

This button will start the activity of yearly plan.

1. <Button
2. android:id="@+id/year\_button"
3. android:layout\_width="match\_parent"
4. android:layout\_height="wrap\_content"
5. android:text="@string/oneYear"
6. android:textColor="@color/gray"/>

This button will be set on click function which start yearly plan activity.

1. year\_button.setOnClickListener {
2. yearlyFunction()
3. }
4. fun yearlyFunction(){
5. val intent = Intent(**this**,yearlyActivity::**class**.java)
6. startActivity(intent)
7. }

## Add one month button

This button will start the activity of monthly plan.

This button will be set on click function which start monthly plan activity.

1. <Button
2. android:id="@+id/month\_button"
3. android:layout\_width="match\_parent"
4. android:layout\_height="wrap\_content"
5. android:text="@string/oneMonth"
6. android:textColor="@color/gray"/>
7. month\_button.setOnClickListener {
8. monthlyFunction()
9. }
10. fun monthlyFunction(){
11. val intent = Intent(**this**,monthlyActivity::**class**.java)
12. startActivity(intent)
13. }

