

VISVESVARAYA TECHNOLOGICAL UNIVERSITY
JNANA SANGAMA, BELAGAVI-590018, KARNATAKA



A Mini Project Report
on

BMI CALCULATOR

*Submitted in partial fulfillment of the requirements for the VI Semester of degree of
Bachelor of Engineering in Information Science and Engineering of Visvesvaraya
Technological University, Belagavi*

by

JAFER SADIQ	(1RN19IS065)
KARTHIK RAJ R	(1RN19IS068)
M BALAKRISHNA KAMATH	(1RN19IS074)

Under the Guidance of

Ms. Priyanka M R
Assistant Professor
Department of ISE



Department of Information Science and Engineering

RNS Institute of Technology

**Dr. Vishnuvaradhana Road, Rajarajeshwari Nagar post,
Channasandra, Bengaluru-560098**

2021-2022

RNS INSTITUTE OF TECHNOLOGY

Dr. Vishnuvaradhana Road, Rajarajeshwari Nagar post,
Channasandra, Bengaluru - 560098

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING



CERTIFICATE

Certified that the project work entitled “**BMI CALCULATOR**” has been successfully completed by **JAFER SADIQ (1RN19IS065)**, **KARTHIK RAJ R (1RN19IS068)** and **M BALAKRISHNA KAMATH (1RN19IS074)**, bonafide students of **RNS Institute of Technology, Bengaluru** in partial fulfillment of the requirements for the award of degree in **Bachelor of Engineering in Information Science and Engineering of Visvesvaraya Technological University, Belagavi** during academic year **2021- 2022**. The mini project report has been approved as it satisfies the academic requirements with respect to the Mobile Applications Development laboratory.

Mrs. Priyanka M R
Mini Project Guide
Assistant Professor
Department of ISE

Dr. Suresh L
Professor and HOD
Department of ISE
RNSIT

Dr. M K Venkatesha
Principal
RNSIT

External Viva

Name of the Examiners

Signature with Date

1. _____

1. _____

2. _____

2. _____

DECLARATION

We, **JAFER SADIQ [USN: 1RN19IS065]**, **KARTHIK RAJ R [USN: 1RN19IS068]**, **M BALAKRISHNA KAMATH [USN: 1RN19IS074]**, students of VI Semester BE, in Information Science and Engineering, RNS Institute of Technology hereby declare that the Project entitled “**BMI CALCULATOR**” has been carried out by us and submitted in partial fulfillment of the requirements for the *VI Semester of degree of **Bachelor of Engineering in Information Science and Engineering** of Visvesvaraya Technological University, Belagavi* during academic year 2021-2022.

Place: Bengaluru

Date:

JAFER SADIQ(1RN19IS068)

KARTHIK RAJ R(1RN19IS065)

M BALAKRISHNA KAMATH(1RN19IS074)

ABSTRACT

The body mass index (BMI) is the metric currently in use for defining anthropometric height/weight characteristics in adults and for classifying (categorizing) them into groups. The common interpretation is that it represents an index of an individual's fatness. It also is widely used as a risk factor for the development of or the prevalence of several health issues. In addition, it is widely used in determining public health policies. The BMI has been useful in population-based studies by virtue of its wide acceptance in defining specific categories of body mass as a health issue. However, it is increasingly clear that BMI is a rather poor indicator of percent of body fat. Importantly, the BMI also does not capture information on the mass of fat in different body sites. The latter is related not only to untoward health issues but to social issues as well. Lastly, current evidence indicates there is a wide range of BMIs over which mortality risk is modest, and this is age related. Body fatness has been an important psychosocial issue among humans.

ACKNOWLEDGMENT

The fulfillment and rapture that go with the fruitful finishing of any assignment would be inadequate without the specifying the people who made it conceivable, whose steady direction and support delegated the endeavors with success.

We would like to profoundly thank **Management** of **RNS Institute of Technology** for providing such a healthy and vibrant environment to carry out the project work.

We would like to express our sincere thanks to our beloved Principal **Dr. M K Venkatesha** for his support and inspired me towards the attainment of knowledge.

We wish to place on record our words of gratitude to **Dr. Suresh L**, Professor and Head of the Department, Information Science and Engineering, for being the enzyme and master mind behind our project work.

We would also like to thank our Mini Project guide **Ms. Priyanka M R**, Assistant Professor, Department of ISE, RNSIT, Bengaluru, for his valuable inputs, suggestions, time and guidance.

We would like to thank all other teaching and non-teaching staff of Information Science & Engineering who have directly or indirectly helped me to carry out the project work.

Place: Bengaluru

Date:

JAFER SADIQ
KARTHIK RAJ R
M BALAKRISHNA KAMATH

TABLE OF CONTENTS

CERTIFICATE	
ABSTRACT	ii
ACKNOWLEDGMENT	iii
TABLE OF CONTENTS	iv
LIST OF FIGURES	vi
LIST OF TABLES	vii
ABBREVIATIONS	viii
1. INTRODUCTION TO ANDROID	1
1.1 History	1
1.2 Android Version	1
1.3 Architecture of Android	2
1.4 Android Studio Installation	5
2. INTRODUCTION TO PROJECT	7
2.1 Overview of the Project	7
2.2 Aim of the Project	8
3. SYSTEM DESIGN	12
3.1 System Requirements	12
3.2 User Interface	12
4. IMPLEMENTATION	15
4.1 BMI Formulation	15
4.2 BMI Range for different Categories	15
5. TESTING	18
5.1 Unit Testing	18
5.2 Integration Testing	20
5.3 System Testing	21

6. RESULTS	22
6.1 Main Screen Interface	22
6.2 Results for different BMI Categories	23
7. CONCLUSION AND FUTURE ENHANCEMENTS	27
7.1 Conclusion	27
7.2 Future Enhancements	27
REFERENCES	28

LIST OF FIGURES

Figure No.	Description	Page No.
Figure 1.1	Android Architecture diagram	3
Figure 2.1	BMI calculator	7
Figure 2.2	BMI table for adults	8
Figure 2.3	BMI table for children	8
Figure 2.4	BMI Chart for adults	9
Figure 2.5	BMI formula	11
Figure 3.1	Initial Screen	12
Figure 3.2	Main Screen	13
Figure 3.3	Result Screen	14
Figure 4.1	BMI Category	16
Figure 4.2	BMI Obesity categories	16
Figure 6.1	Main Screen	22
Figure 6.2	Severe Thinness	23
Figure 6.3	Moderate Thinness	23
Figure 6.4	Mild Thinness	24
Figure 6.5	Normal	24
Figure 6.6	Overweight	25
Figure 6.7	Obese Class I	25
Figure 6.8	Obese Class II	26

LIST OF TABLES

Table No.	Description	Page No.
Table 1.1	Android Versions	2
Table 5.1	Main Screen Unit Testing	18
Table 5.2	Integration Testing	20
Table 5.3	System Testing	21

ABBREVIATIONS

iOS - iPhone Operating System

Inc. - Incorporated

OS - Operating System

SDK - Software Development Kit

HTML - Hyper Text Markup Language

SQL - Structured Query Language

API - Application Programming Interface

DVM - Dalvik Virtual Machine

JVM - Java Virtual Machine

IDE - Integrated Development Environment

RAM - Random Access Memory

UI - User Interface

DDMS - Dalvik Debug Monitor Server

Chapter 1

INTRODUCTION TO ANDRIOD

1.1 History

In past mobile phones were used only to make calls but with the introduction of smartphone the mobile phone has evolved to a low powered hand held processing system. This evolution was caused by the operating system for the mobile phones making them smart that have processing and storage of their own. Now the mobile provides numerous functionalities from calling to texting, multimedia sharing, emails, socializing applications, word processor, excel sheets to various multiplayer games and much more.

The operating system for these hand held devices are iOS by Apple Inc., Windows by Windows Inc. and Android by Google. Among the competitors in smartphone operating system industry Android holds the largest market share in terms of units shipped worldwide and number of users.

Android is an open source operating system based on Linux kernel on which applications run on an application framework that controls the activities supported by the libraries and Dalvik virtual machine which compiles and converts all java class files into a single file. There can be number of virtual machines running simultaneously on a single device handling different applications or instances of an application.

Android operating system provides memory management, process management to the applications and services running. Each release of android improved user experience and brought enhanced features. In 2012 Android became the most popular operating system for mobile devices, surpassing Apple's iOS, and, as of 2020, about 75 percent of mobile devices run Android.

1.2 Android Versions

The development of the Android operating system was started in 2003 by Android, Inc. Later on, it was purchased by Google in 2005. The beta version of Android OS was released on November 5, 2007, while the software development kit (SDK) was released on November 12, 2007.

The first Android mobile was publicly released with Android 1.0 of the T-Mobile G1 (aka HTC Dream) in October 2008. The first Android version which was released under the numerical order format was Android 10.

Code name	Version numbers	API level	Release date
No codename	1.0	1	September 23, 2008
No codename	1.1	2	February 9, 2009
Cupcake	1.5	3	April 27, 2009
Donut	1.6	4	September 15, 2009
Eclair	2.0 - 2.1	5 - 7	October 26, 2009
Froyo	2.2 - 2.2.3	8	May 20, 2010
Gingerbread	2.3 - 2.3.7	9 - 10	December 6, 2010
Honeycomb	3.0 - 3.2.6	11-13	February 22, 2011
Ice Cream Sandwich	4.0 - 4.0.4	14-15	October 18, 2011
Jelly Bean	4.1 - 4.3.1	16-18	July 9, 2012
KitKat	4.4 - 4.4.4	19-20	October 31, 2013
Lollipop	5.0 - 5.1.1	21- 22	November 12, 2014
Marshmallow	6.0 - 6.0.1	23	October 5, 2015
Nougat	7.0	24	August 22, 2016
Nougat	7.1.0 - 7.1.2	25	October 4, 2016
Oreo	8.0	26	August 21, 2017
Oreo	8.1	27	December 5, 2017
Pie	9.0	28	August 6, 2018
Android 10	10.0	29	September 3, 2019
Android 11	11	30	September 8, 2020

Table 1.1 Android Versions

1.3 Android Architecture

Android operating system is a stack of software components which is roughly divided into five sections and four main layers as shown below in the architecture diagram.

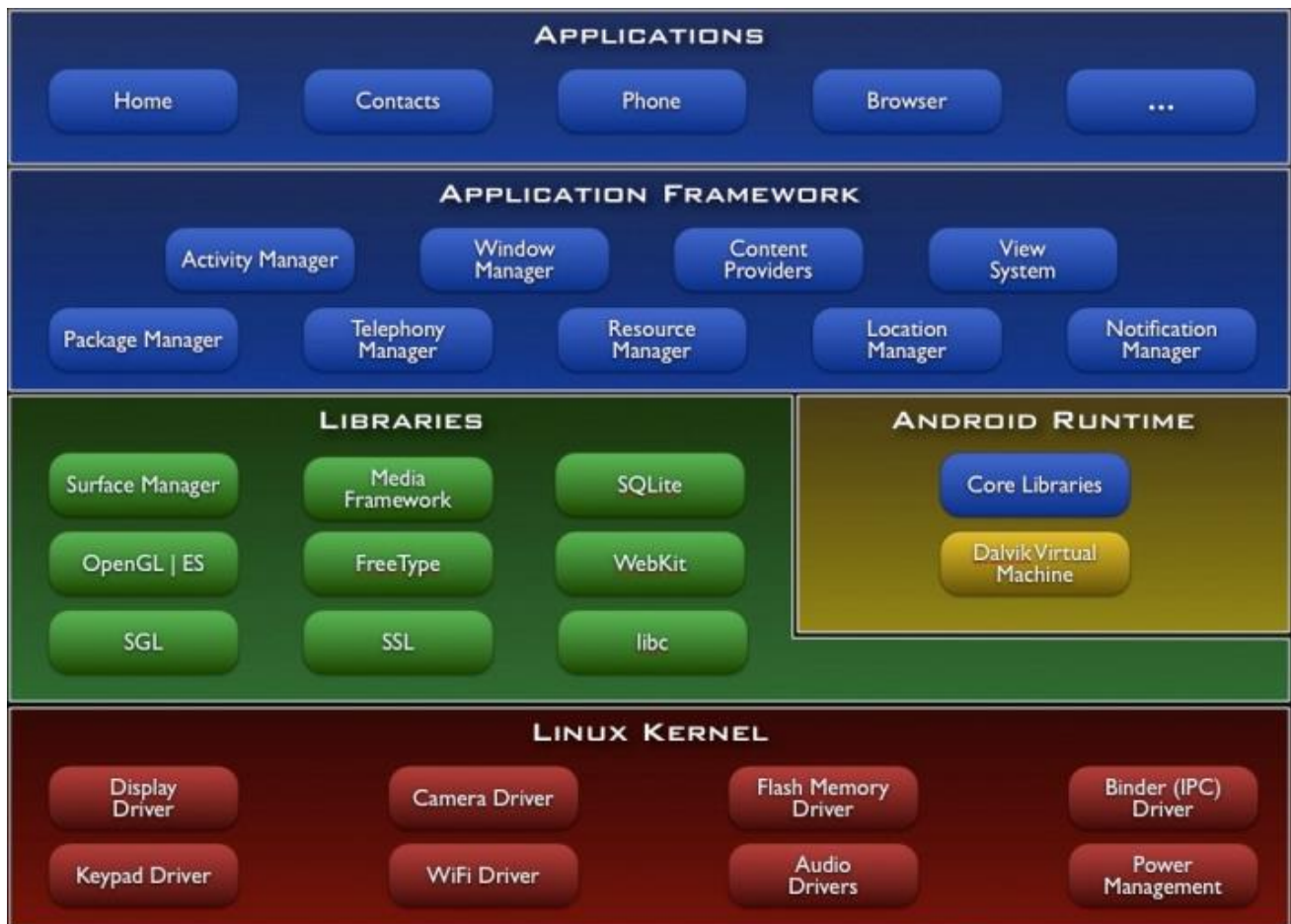


Figure 1.1 Android Architecture diagram

- ❖ **Linux Kernel:** This is the layer at the very bottom of the Android architecture. All other layers run on top of the Linux kernel and rely on this kernel to interact with the hardware. This layer contains all the essential hardware drivers which help to control and communicate with the hardware. It provides the basic functionality like Process Management, Memory Management and Device Management like Camera, Display, Flash etc.
- ❖ **Libraries:** This is a set of common functions of the application framework that enables the device to handle different types of data. Some of the most important set of libraries that are included are – Web kit which is the browser engine to display HTML, OpenGL used to render 2- D or 3-D graphics on to the screen, SQLite which is a useful repository for storing and sharing of application data.

Android libraries available to the Android developer is as follows

- android.app – Provides access to the application model and is the cornerstone of all Android applications.
- android.content – Facilitates content access, publishing and messaging between applications and application components.
- android.database – Used to access data published by content providers and includes SQLite database management classes.
- android.opengl – A Java interface to the OpenGL ES 3D graphics rendering API.
- android.os – Provides applications with access to standard operating system services including messages, system services and inter-process communication.
- android.text – Used to render and manipulate text on a device display.
- android.view – The fundamental building blocks of application user interfaces.
- android.widget – A rich collection of pre-built user interface components such as buttons, labels, list views, layout managers, radio buttons etc.
- android.webkit – A set of classes intended to allow web-browsing capabilities to be built into applications.

❖ **Android Runtime :** The Android runtime mainly consist of the Dalvik Virtual Machine (DVM). DVM is very much like the standard Java Virtual Machine (JVM) except that it is optimized for mobile devices that have low processing power and low memory. DVM generates a .dex file from the .class file at compile time and provides higher efficiency in low resources devices. Each application has its own process and an instance of DVM. Android runtime also provides core libraries that enable the Android developers to create applications using the Java language.

❖ **Application Framework :** The Android runtime mainly consist of the Dalvik Virtual Machine (DVM). DVM is very much like the standard Java Virtual Machine (JVM) except that it is optimized for mobile devices that have low processing power and low memory. DVM generates a .dex file from the .class file at compile time and provides higher efficiency in low resources devices. Each application has its own process and an instance of DVM. Android runtime also provides core libraries that enable the Android developers to create applications using the Java language.

- ❖ **Applications:** This is the topmost layer in the architecture and the layer where the application that we develop fits in. This layer provides several pre-installed applications that are default for certain things like Contacts Books, Browser etc.

1.4 Android Studio Installation

Android Studio is the official integrated development environment (IDE) for Android application development. It is based on the IntelliJ IDEA, a Java integrated development environment for software, and incorporates its code editing and developer tools.

To support application development within the Android operating system, Android Studio uses a Gradle-based build system, emulator, code templates, and GitHub integration. Every project in Android Studio has one or more modalities with source code and resource files.

PROCEDURE TO BE FOLLOWED TO DOWNLOAD AND INSTALL ANDROID STUDIO:

STEP 1: Android Studio and the Software Development Kit can be downloaded directly from any web browser using the below link.

<https://developer.android.com/studio>

STEP 2: Android Studio is available for Mac, Windows, and Linux desktop platforms.

Windows

To install Android Studio on Windows, proceed as follows:

- i. If you downloaded an .exe file (recommended), double-click to launch it. If you downloaded a .zip file, unpack the ZIP, copy the android-studio folder into your Program Files folder, and then open the android-studio > bin folder and launch studio64.exe (for 64-bit machines) or studio.exe (for 32-bit machines).
- ii. Follow the setup wizard in Android Studio and install any SDK packages that it recommends.

Mac

To install Android Studio on your Mac, proceed as follows:

- i. Launch the Android Studio DMG file.
- ii. Drag and drop Android Studio into the Applications folder, then launch it.
- iii. Select if you want to import previous Android Studio settings, then press OK.
- iv. The Android Studio Setup Wizard guides you through the rest of the setup, which includes downloading Android SDK components that are required for development.

Linux

To install Android Studio on Linux, proceed as follows:

- i. Unpack the .zip file you downloaded to an appropriate location for your applications, such as within /usr/local/ for your user profile, or /opt/ for shared users. If you're using a 64-bit version of Linux, make sure you first install the required libraries for 64-bit machines.
- ii. To launch Android Studio, open a terminal, navigate to the android-studio/bin/ directory and execute studio.sh.
- iii. Select whether you want to import previous Android Studio settings or not, then click OK.
- iv. The Android Studio Setup Wizard guides you through the rest of the setup, which includes downloading Android SDK components that are required for development.

Chapter 2

INTRODUCTION TO PROJECT

2.1 Overview of the Project

The Body Mass Index (BMI) Calculator can be used to calculate BMI value and corresponding weight status while taking age into consideration. Use the "Metric Units" tab for the International System of Units or the "Other Units" tab to convert units into either US or metric units. Note that the calculator also computes the Ponderal Index in addition to BMI, both of which are discussed below in detail.

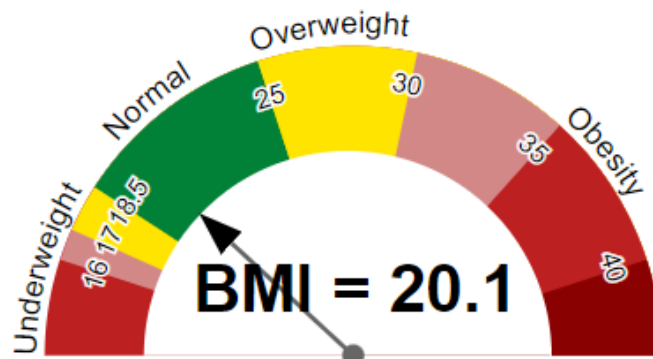


Figure 2.1 BMI Calculator

Figure 2.1 describes the value ranges of the body mass index.

- Healthy BMI range: 18.5 kg/m^2 - 25 kg/m^2
- Healthy weight for the height: 59.9 kgs - 81.0 kgs
- Ponderal Index: 11.1 kg/m^3

2.2 BMI INTRODUCTION

BMI is a measurement of a person's leanness or corpulence based on their height and weight, and is intended to quantify tissue mass. It is widely used as a general indicator of whether a person has a healthy body weight for their height

Specifically, the value obtained from the calculation of BMI is used to categorize whether a person is underweight, normal weight, overweight, or obese depending on what range the value falls between.

These ranges of BMI vary based on factors such as region and age, and are sometimes further divided into subcategories such as severely underweight or very severely obese. Being overweight or underweight can have significant health effects, so while BMI is an imperfect measure of healthy body weight, it is a useful indicator of whether any additional testing or action is required. Refer to the table below to see the different categories based on BMI that are used by the calculator.

2.2 AIM OF THE PROJECT

BMI Table for adults

This is the World Health Organization's (WHO) recommended body weight based on BMI values for adults. It is used for both men and women, age 18 or older.

Category	BMI range - kg/m ²
Severe Thinness	< 16
Moderate Thinness	16 - 17
Mild Thinness	17 - 18.5
Normal	18.5 - 25
Overweight	25 - 30
Obese Class I	30 - 35
Obese Class II	35 - 40
Obese Class III	> 40

Figure 2.2 BMI table for adults

Figure 2.2 describes about the BMI value ranges for adults.

BMI table for children and teens, age 2-20

The Centers for Disease Control and Prevention (CDC) recommends BMI categorization for children and teens between age 2 and 20.

Category	Percentile Range
Underweight	<5%
Healthy weight	5% - 85%
At risk of overweight	85% - 95%
Overweight	>95%

Figure 2.3 BMI table for children

Figure 2.3 describes about the BMI value ranges for children.

BMI chart for adults

This is a graph of BMI categories based on the World Health Organization data. The dashed lines represent subdivisions within a major categorization.

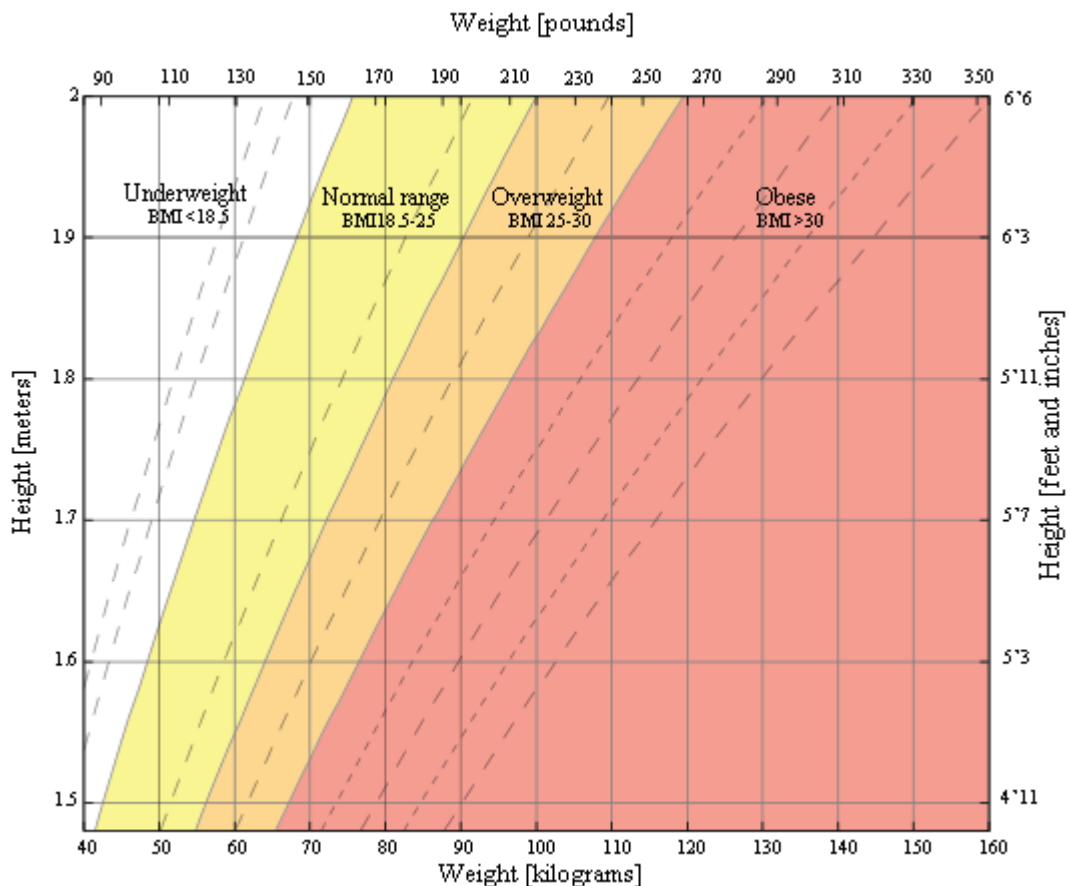


Figure 2.4 BMI chart for adults

Figure 2.4 describes of BMI chart of Height v/s Weight for adults.

Risks associated with being overweight

Being overweight increases the risk of a number of serious diseases and health conditions.

Below is a list of said risks, according to the Centers for Disease Control and Prevention (CDC):

- High blood pressure
- Higher levels of LDL cholesterol, which is widely considered "bad cholesterol," lower levels of HDL cholesterol, considered to be good cholesterol in moderation, and high levels of triglycerides.

- Osteoarthritis, a type of joint disease caused by breakdown of joint cartilage
- Sleep apnea and breathing problems
- Certain cancers (endometrial, breast, colon, kidney, gallbladder, liver)
- Mental illnesses such as clinical depression, anxiety, and others, Body pains and difficulty with certain physical functions .Generally, an increased risk of mortality compared to those with a healthy BMI

As can be seen from the list above, there are numerous negative, in some cases fatal, outcomes that may result from being overweight. Generally, a person should try to maintain a BMI below 25 kg/m^2 , but ideally should consult their doctor to determine whether or not they need to make any changes to their lifestyle in order to be healthier.

Risks associated with being underweight

Being underweight has its own associated risks, listed below:

- Malnutrition, vitamin deficiencies, anemia (lowered ability to carry blood vessels)
- Osteoporosis, a disease that causes bone weakness, increasing the risk of breaking a bone.
- A decrease in immune function
- Growth and development issues, particularly in children and teenagers
- Possible reproductive issues for women due to hormonal imbalances that can disrupt the menstrual cycle. Underweight women also have a higher chance of miscarriage in the first trimester.
- Potential complications as a result of surgery
- Generally, an increased risk of mortality compared to those with a healthy BMI

In some cases, being underweight can be a sign of some underlying condition or disease such as anorexia nervosa, which has its own risks. Consult your doctor if you think you or someone you know is underweight, particularly if the reason for being underweight does not seem obvious.

BMI Formula

Below are the equations used for calculating BMI in the International System of Units (SI) and the US customary system (USC) using a 5'10", 160-pound individual as an example:

USC Units:

$$\text{BMI} = 703 \times \frac{\text{mass (lbs)}}{\text{height}^2 (\text{in})} = 703 \times \frac{160}{70^2} = 22.96 \frac{\text{kg}}{\text{m}^2}$$

SI, Metric Units:

$$\text{BMI} = \frac{\text{mass (kg)}}{\text{height}^2 (\text{m})} = \frac{72.57}{1.78^2} = 22.90 \frac{\text{kg}}{\text{m}^2}$$

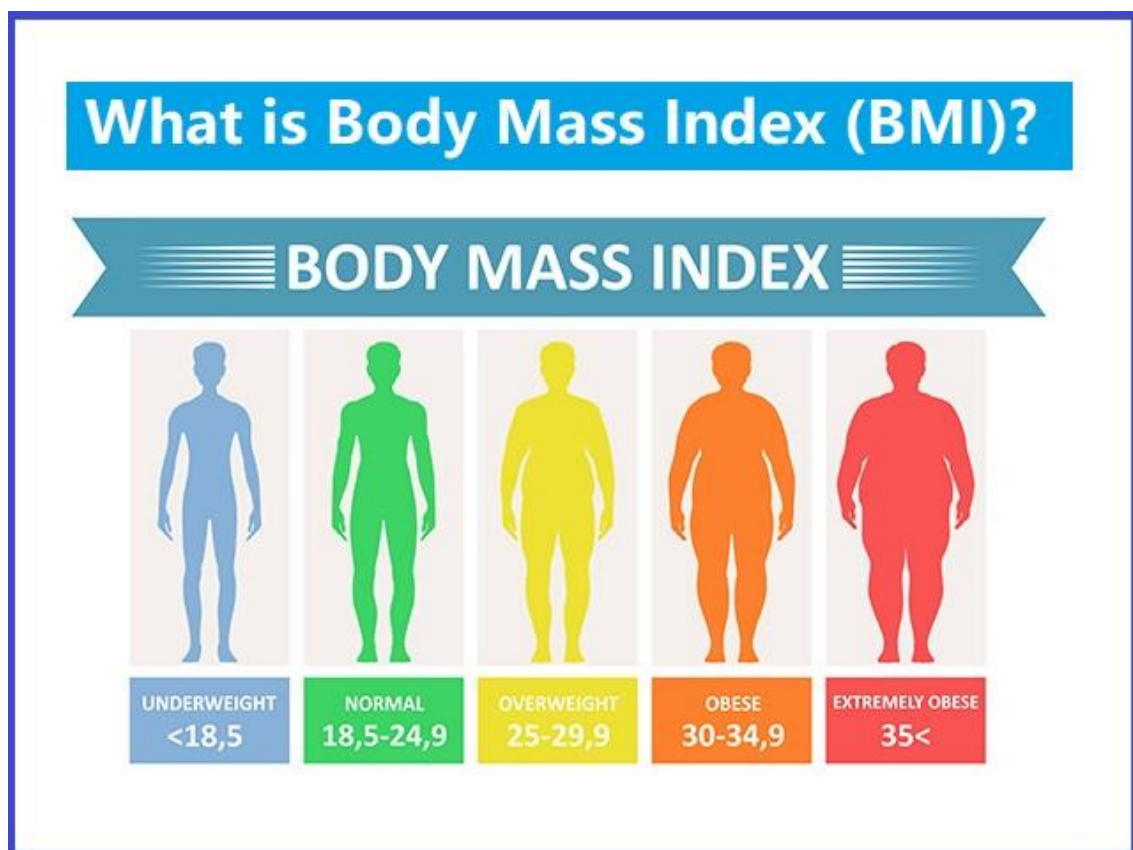


Figure 2.5 BMI formula

Figure 2.5 describes the value ranges for underweight, normal, overweight and obese.

Chapter 3

SYSTEM DESIGN

3.1 System Requirements

3.1.1 Hardware Requirements:

- Processor: Pentium IV or above
- RAM: 8GB or more
- Hard Disk: 2GB or more

3.1.2 Software Requirements:

- Operating System: Windows 7 or above
- IDE: Android Studio
- API Level: 19 or above

3.2 User Interface

➤ INITIAL SCREEN:

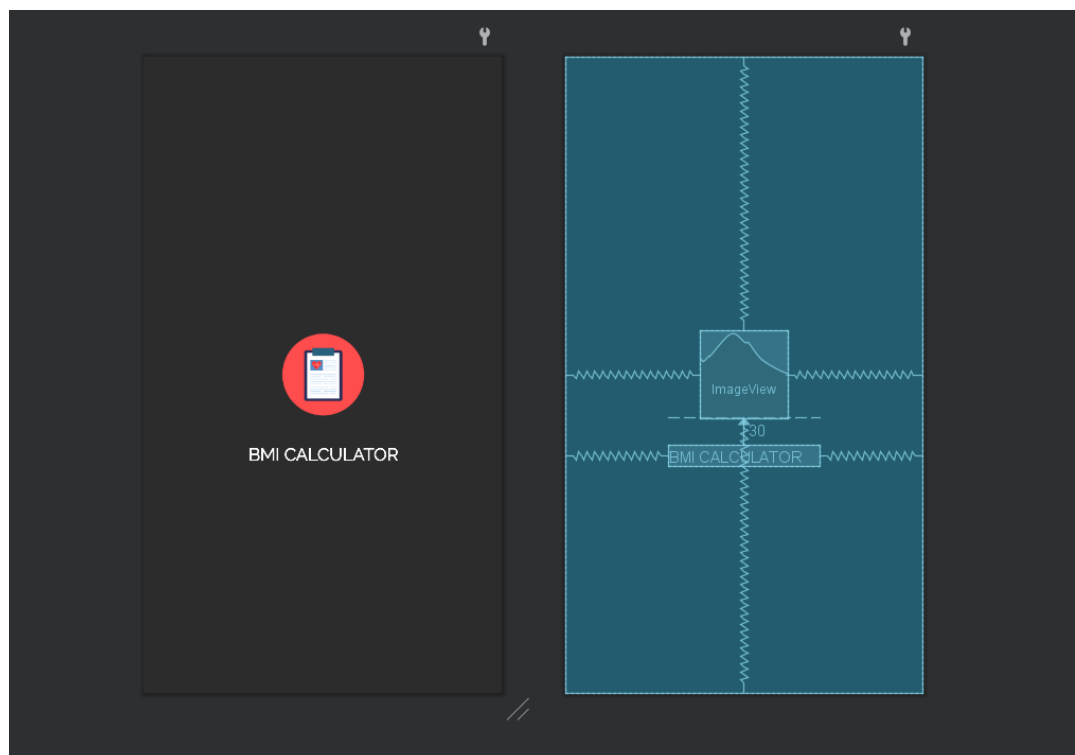


Figure 3.1 Initial Screen

Figure 3.1 is the splash screen which appears for few seconds when the app is launched.

➤ MAIN SCREEN:

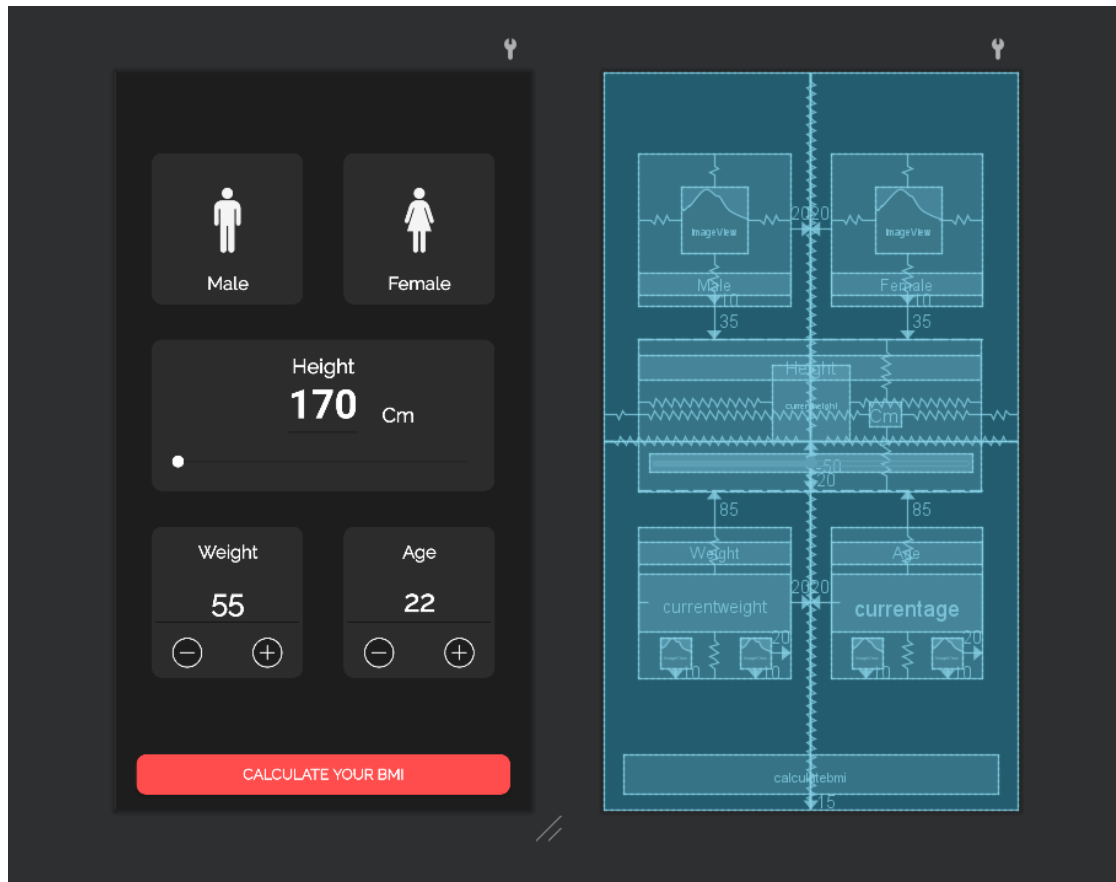


Figure 3.2 Main Screen

Figure 3.2 is the main activity screen in which we have various EditText, ImageView, TextView and Buttons.

We can either enter the values through editText or use seekbar or buttons to enter the required values.

There are also validations for the editText so there are no out of improper values that are being entered.

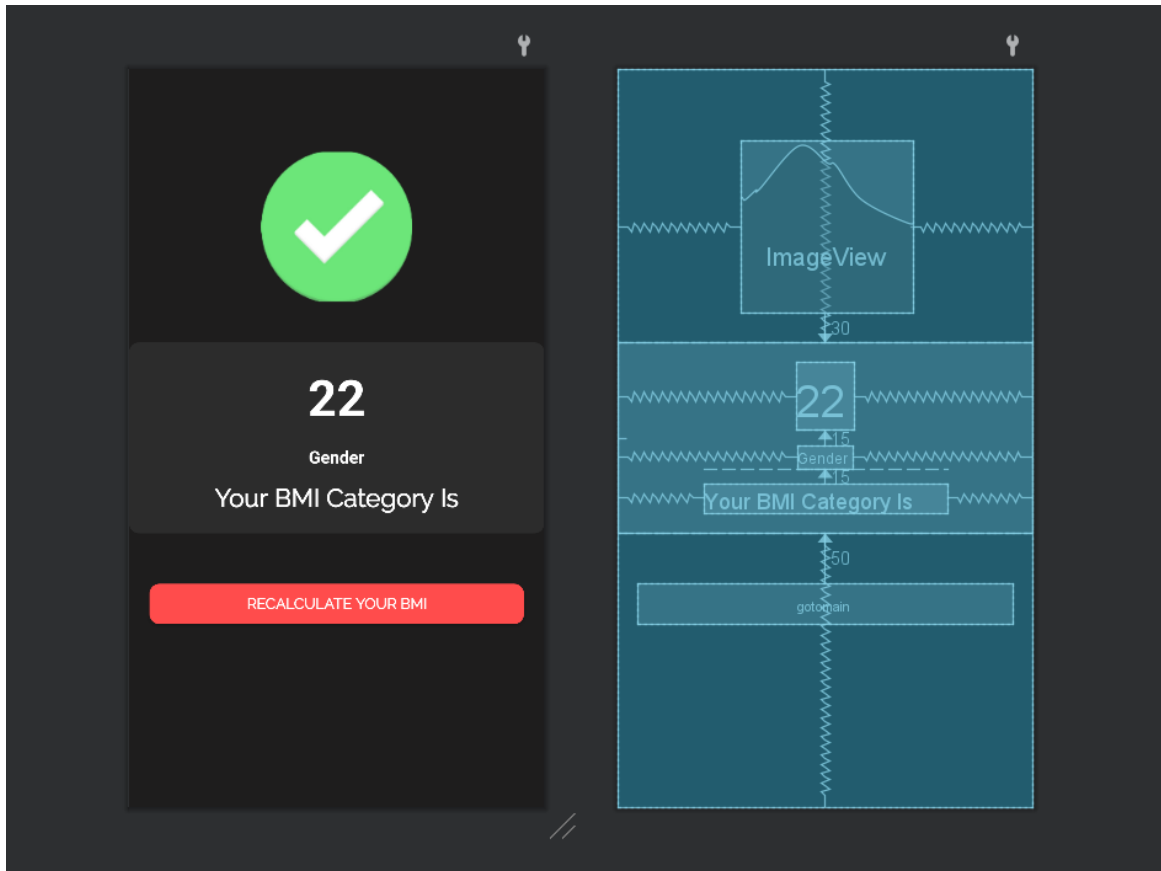
➤ RESULT SCREEN:

Figure 3.3 Result Screen

Figure 3.3 is the output screen of the calculator app in which according to the entered values particular output category values are displayed in it.

Chapter 4

IMPLEMENTATION

4.1 BMI FORMULATION

- BMI is a reliable indicator of body fatness for most people. It is used to screen for weight categories that may lead to health problems.
- BMI is a person's weight in kilograms divided by the square of height in meters. A high BMI can indicate high body fatness.

USC Units:

$$\text{BMI} = 703 \times \frac{\text{mass (lbs)}}{\text{height}^2 \text{ (in)}} = 703 \times \frac{160}{70^2} = 22.96 \frac{\text{kg}}{\text{m}^2}$$

SI, Metric Units:

$$\text{BMI} = \frac{\text{mass (kg)}}{\text{height}^2 \text{ (m)}} = \frac{72.57}{1.78^2} = 22.90 \frac{\text{kg}}{\text{m}^2}$$

intheight=Float.parseFloat(height);

intweight=Float.parseFloat(weight);

intheight=intheight/100;

intbmi=intweight/(intheight*intheight);

4.2 BMI RANGE FOR DIFFERENT CATEGORIES

The CDC and WHO use the BMI range charts below to evaluate the weight of adults aged 20 and over given his or her height. These ranges typically apply to North American and European populations.

- If your BMI is less than 18.5, it falls within the underweight range.
- If your BMI is 18.5 to <25, it falls within the healthy weight range.
- If your BMI is 25.0 to <30, it falls within the overweight range.
- If your BMI is 30.0 or higher, it falls within the obesity range.

BMI	Status
≤ 18.4	Underweight
18.5 - 24.9	Normal
25.0 - 39.9	Overweight
≥ 40.0	Obese

Figure 4.1 BMI Category

Figure 4.1 shows the different BMI categories.

Obesity is frequently subdivided into categories:

- Class 1: BMI of 30 to < 35
- Class 2: BMI of 35 to < 40
- Class 3: BMI of 40 or higher. Class 3 obesity is sometimes categorized as “severe” obesity.

BMI	Category
< 16.0	Severely Underweight
16.0 - 18.4	Underweight
18.5 - 24.9	Normal
25.0 - 29.9	Overweight
30.0 - 34.9	Moderately Obese
35.0 - 39.9	Severely Obese
> 40.0	Morbidly Obese

Figure 4.2 BMI Obesity categories

Figure 4.2 shows the different BMI obesity categories.

CODE IMPLEMENTATION:

```

if(intbmi<16){
    mbmicategory.setText("Severe Thinness");
    mbackground.setBackgroundColor(Color.RED);
    mimageView.setImageResource(R.drawable.crosss);
}

```

```
else if(intbmi<16.9 && intbmi>16)
{
    mbmicategory.setText("Moderate Thinness");
    mbackground.setBackgroundColor(R.color.halfwarn);
    mimageview.setImageResource(R.drawable.warning);
}
else if(intbmi<18.4 && intbmi>17)
{
    mbmicategory.setText("Mild Thinness");
    mbackground.setBackgroundColor(R.color.halfwarn);
    mimageview.setImageResource(R.drawable.warning);
}
else if(intbmi<24.9 && intbmi>18.5 )
{
    mbmicategory.setText("Normal");
    mimageview.setImageResource(R.drawable.ok);
}
else if(intbmi <29.9 && intbmi>25)
{
    mbmicategory.setText("Overweight");
    mbackground.setBackgroundColor(R.color.halfwarn);
    mimageview.setImageResource(R.drawable.warning);
}
else if(intbmi<34.9 && intbmi>30)
{
    mbmicategory.setText("Obese Class I");
    mbackground.setBackgroundColor(R.color.halfwarn);
    mimageview.setImageResource(R.drawable.warning);
}
else
{
    mbmicategory.setText("Obese Class II");
    mbackground.setBackgroundColor(R.color.warn);
    mimageview.setImageResource(R.drawable.crosss);
}
```

Chapter 5

TESTING

Software testing is an essential phase in the development life cycle of an application. Testing ensures that the developed system meets its functional and non-functional requirements. Two important terms in software testing are Verification and Validation. Verification is the process of evaluating work-products like requirement specs, design specs and test cases etc. of different development phases to make sure that they meet the requirements for that phase. It ensures that the system is built in the right way. Whereas Validation is the process of evaluating the software at the end of the development phase to make sure that it meets the business requirements. It is used to make sure that the product fulfils its intended use and that the end product is built right.

One of the most important tools to test and debug an Android app is the Dalvik debug monitor server (DDMS) that is part of the Android framework. DDMS helps you to debug your code as it prints errors, warning and other information from your code. It also provides stack traces for exceptions on the Logcat output.

Various other testing strategies have been adopted to make sure the correctness of the BMI calculator app. They are discussed in this chapter.

5.1 Unit Testing

Unit testing is a strategy in software testing where individual components in a software are tested for correctness.

Following are the unit testing test cases for the BMI app:

Table 5.1 Main Screen Unit Testing

Sl.No	Test Case	Action	Result
1	On changing the seekbar value of height	The EditText value of height changes accordingly.	Pass
2	On clicking the + icon of weight	The EditText value of weight increases by 1	Pass

Sl.No	Test Case	Action	Result
3	On clicking the - icon of weight	The EditText value of weight decreases by 1	Pass
4	On clicking the + icon of age	The EditText value of age increases by 1	Pass
5	On clicking the - icon of age	The EditText value of age decreases by 1	Pass
6	On selecting male image	Typetext value changes from default 0 to male.	Pass
7	On selecting female image	Typetext value changes from default 0 to female.	Pass

5.2 Integration Testing

Integration testing is a strategy in software testing where different modules are combined and test to make sure they work together correctly. It is done when the components are unit test and the main objective is to test the interfaces between different components.

Following are the integration testing test cases for the BMI calculator app:

Table 5.2 Integration Testing

Sl.No	Test Case	Action	Result
1	On clicking the calculate BMI button in the main screen	User must be redirected to the result screen where the BMI value, gender and the category to which the value falls into.	Pass
2	On clicking the recalculate button	User must be redirected to main screen where he/she can enter the required inputs needed to calculate the bmi value.	Pass

5.3 SYSTEM TESTING

System Testing is carried out on the whole system in the context of either system requirement specifications or functional requirement specifications or in the context of both. System testing tests the design and behavior of the system and also the expectations of the customer. It is performed to test the system beyond the bounds mentioned in the software requirements specification (SRS).

Following are the system testing test cases for the BMI calculator app:

Table 5.3 System Testing

Sl.No	Test Case	Action	Result
1	On leaving the gender selection without selecting	A Toast message gets displayed saying "Select your Gender first"	Pass
2	If no value is entered for height in EditText.	A Toast message gets displayed saying "Select your Height first"	Pass
3	If no value is entered for weight in EditText.	A Toast message gets displayed saying "Select your weight first"	Pass
4	If no value is entered for age in EditText.	A Toast message gets displayed saying "Age is incorrect"	Pass

Chapter 6

RESULTS

All the Activities provided in the application and its operations have been presented in as snapshots. A detailed view of all the snapshots of the application is given in this section.

6.1 Main Screen Interface

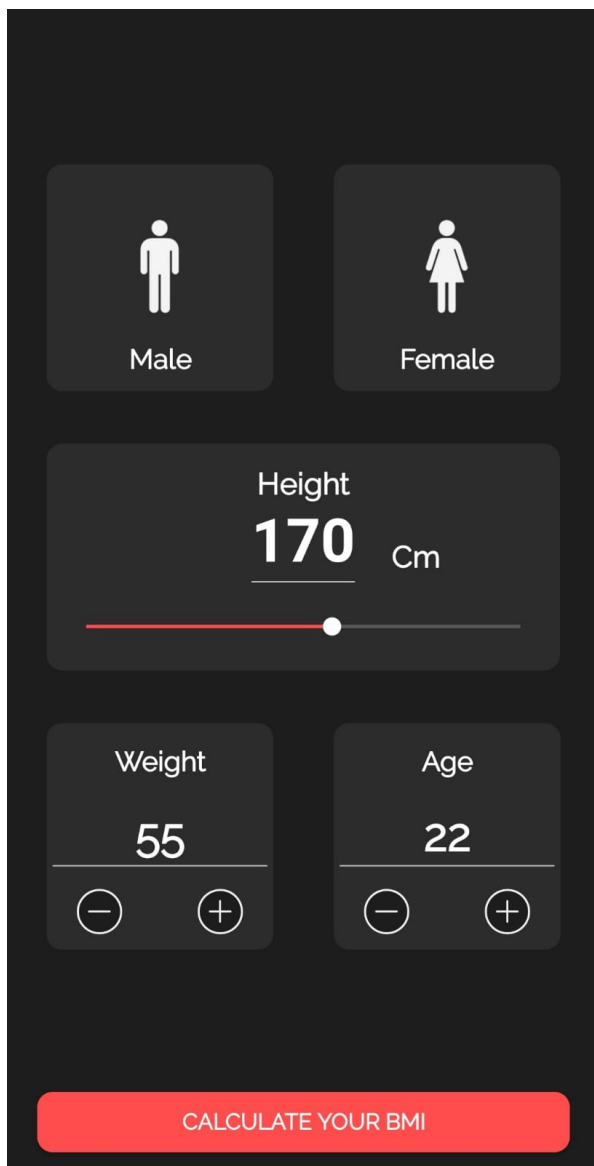


Figure 6.1 is the initial UI interface consists of inputs for Height, Weight and age.

The input can be given through either EditText or seekbar or buttons.

The user can select gender by clicking the required image.

The user can click on the “CALCULATE YOUR BMI ” button to get the result.

Figure 6.1 Main Screen

6.2 RESULTS FOR DIFFERENT BMI CATEGORIES

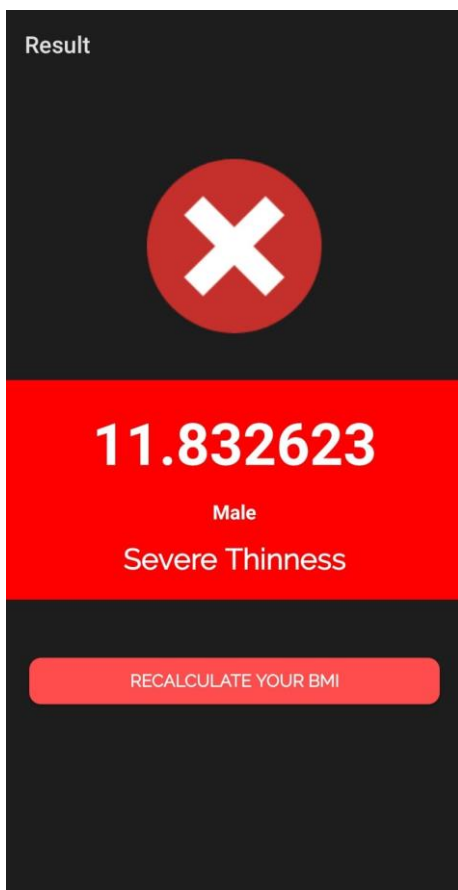


Figure 6.2 shows the BMI value displayed in the result falls under the category of **Severe Thinness**.

This BMI value is below **16**.

Figure 6.2 Severe Thinness

Figure 6.3 shows the BMI value displayed in the result falls under the category of **Moderate Thinness**.

This BMI value is within range **16 - 16.9**.

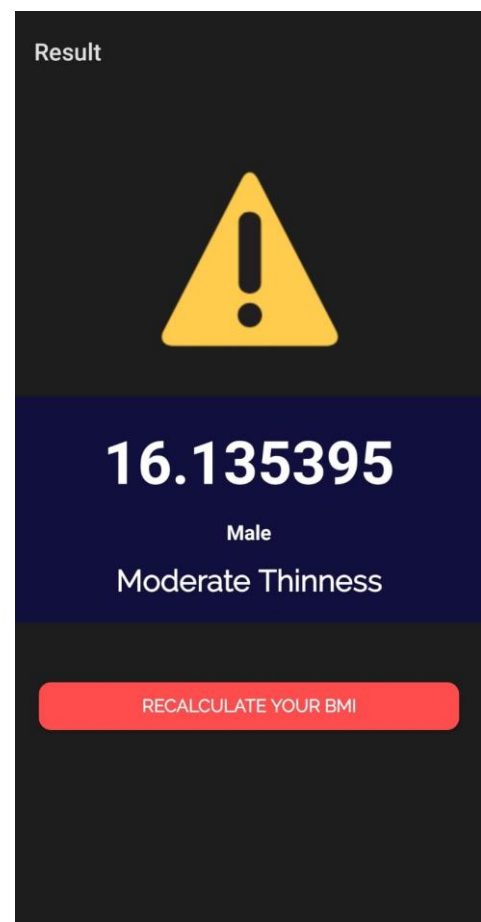


Figure 6.3 Moderate Thinness

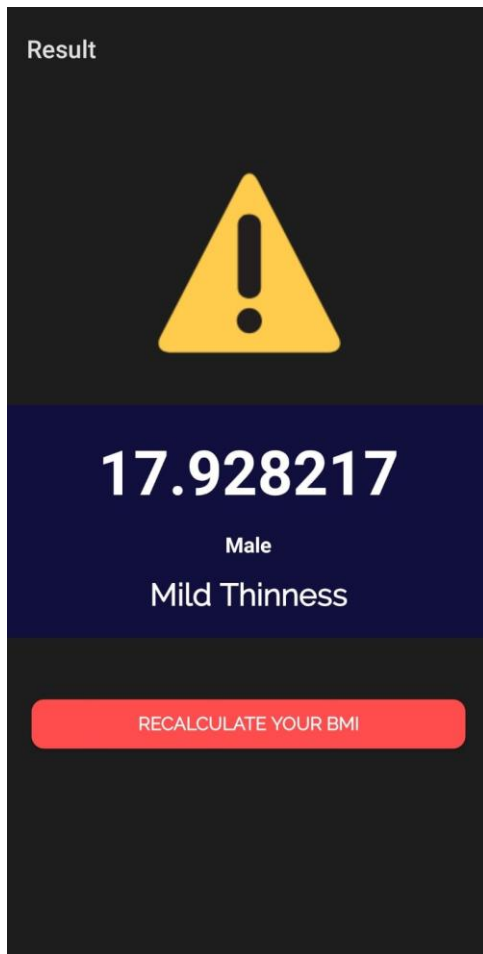


Figure 6.4 shows BMI value displayed in the result falls under the category of **Mild Thinness**.

This BMI value is within range **17 - 18.4**.

Figure 6.4 Mild Thinness

Figure 6.5 shows the BMI value displayed in the result falls under the category of **Normal**.

This BMI value is within range **18.5 – 24.9**.

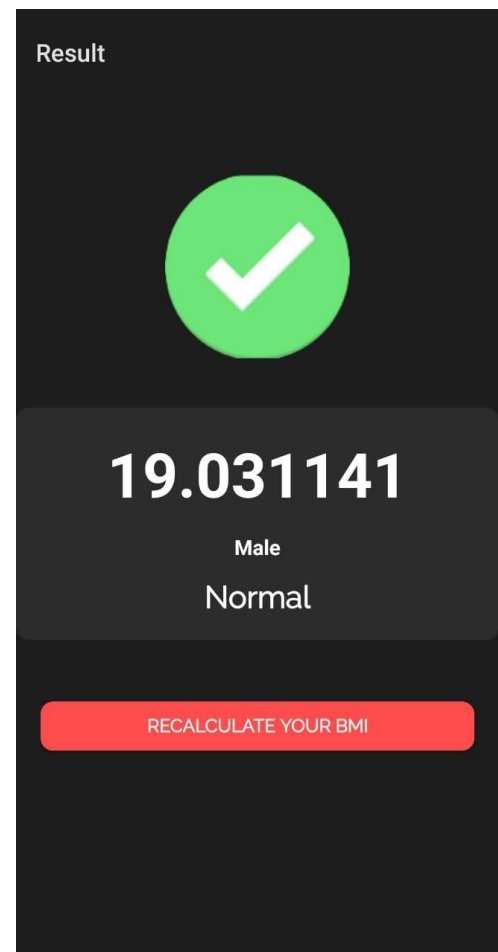


Figure 6.5 Normal

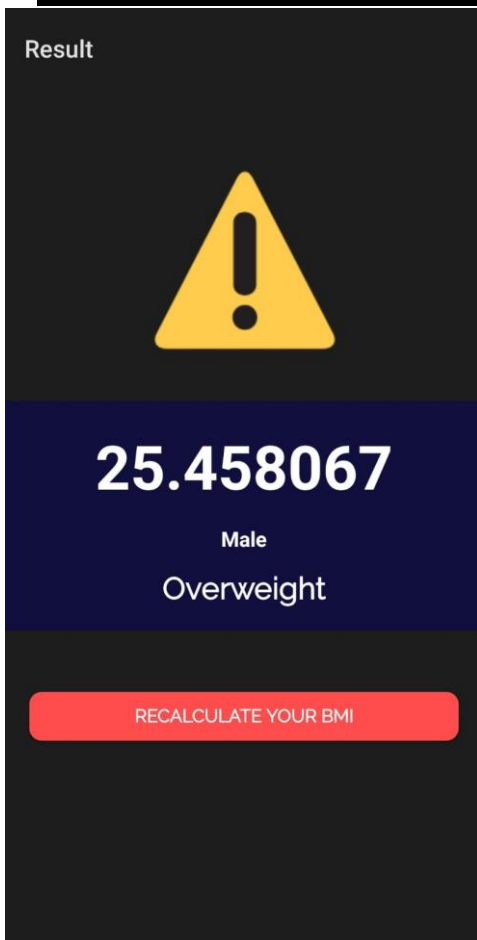


Figure 6.6 Overweight

Figure 6.6 shows the BMI value displayed in the result falls under the category of **Overweight**.

This BMI value is within range **25 – 29.9**.

Figure 6.7 shows the BMI value displayed in the result falls under the category of **Obese Class I**.

This BMI value is within range **30 – 34.9**.

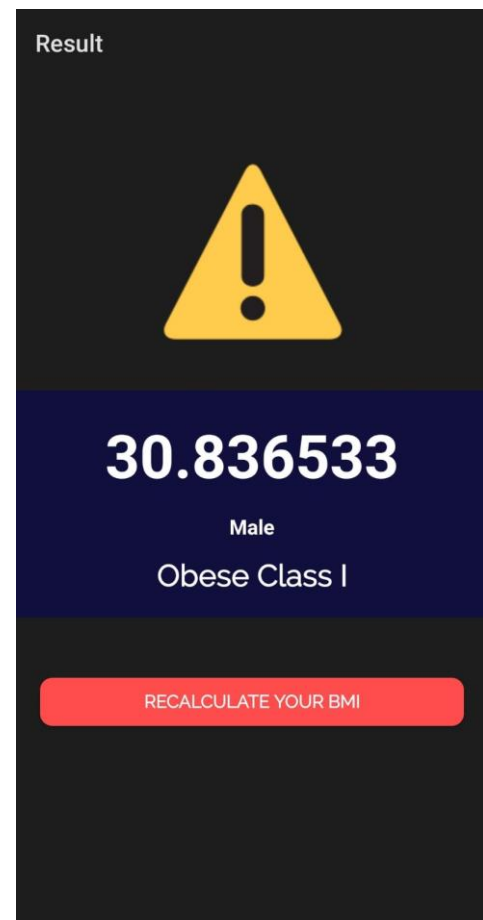


Figure 6.7 Obese Class I

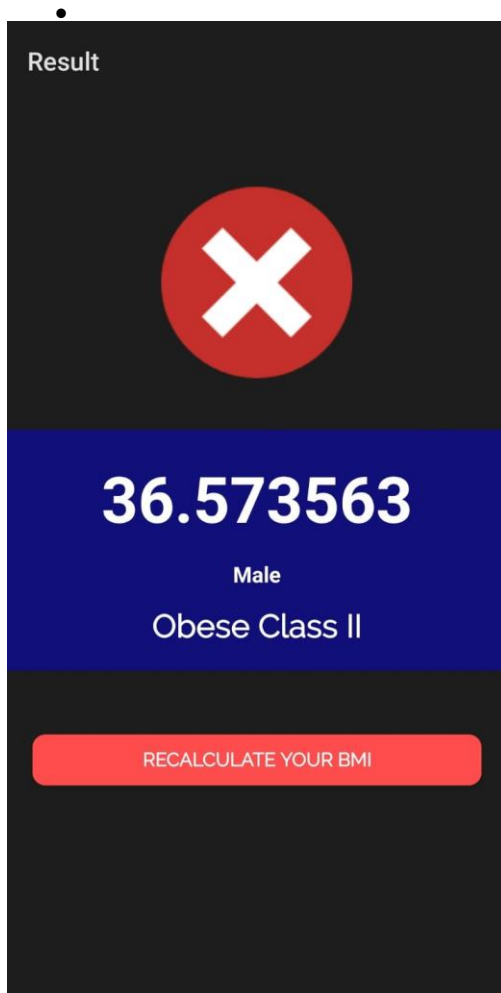


Figure 6.8 shows the BMI value displayed in the result falls under the category of **Obese Class II**.

This BMI value is above **34.9**.

Figure 6.8 Obese class II

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENT

7.1 Conclusion

BMI is a useful measure of overweight and obesity. It's a good way to gauge whether your weight is in healthy proportion to your height. BMI is an estimate of body fat and a good gauge of your risk for diseases that can occur with more body fat.

In fact, knowing your BMI can help you – and your GP – determine any health risks you may face if it's outside of the healthy range. Being overweight can lead to a range of chronic conditions including: Type 2 Diabetes.

This BMI Calculator application provides a quick way for a user to measure his/her Body Mass Index. Although it provides a moderate precision for BMI values, it can instantly show signs of mass index exceeding or subceeding the normal range of 18.5-24.9 and warn them to be cautious when it comes to their diet and nutritional intake. It provides a user friendly UI which can be used by basically any age group of users.

7.2 Future Enhancement

The project deals with the usage of BMI calculation formula that is the ratio of height to weight. This project can be uplifted by considering following measures:

- A more sophisticated approach can be implemented by using age as an additional parameter to calculate the body mass index for better overall precision.
- An extension can be implemented for the application which suggests the user to intake particular food types to bring their diet back to a normal flow and in turn improve their BMI.

REFERENCES

- 1) Google Developer Training, "Android Developer Fundamentals Course – Concept Reference", Google Developer Training Team, 2017.
- 2) Dawn Griffiths and David Griffiths, "Head First Android Development", 1st Edition, O'Reilly SPD Publishers, 2015.
- 3) <https://www.javatpoint.com/android-versions>
- 4) https://www.tutorialspoint.com/android/android_architecture.html
- 5) <https://www.nhs.uk/live-well/healthy-weight/bmi-calculator/>
- 6) <https://patient.info/doctor/bmi-calculator-calculator>