**Chapter 1**

**INTRODUCTION**

**1.1 Problem Definition:**

When we have to send data or information to someone then we send it in the form of files. But when the data is to large i. e large number of different files, then it is not possible to send all data in single time.so here in this project we are trying to combine all the file into single file and then at receiver side we again extracting all the individual file from that single file. Same type of systems are also available but the use compression techniques, so in case of data corruption we are unable to recover original data, so here in this system we deal with the original data so that in this case it is easy to recover the data.

**1.2 Existing System:**

We have observed many limitations of the existing system as follows.

1. It may not be possible to extract all of the files from the Zip file correctly. Damaged data can affect the entire Zip file, multiple files, or just one file.

### Following errors are encountered when ZIP is corrupt:

* "Unexpected end of archive" error while opening ZIP file.
* WinZip Self-Extractor Header Corrupt. Possible Cause: Bad Disk or File Transfer Error.
* **Zip file corrupt**-- Cannot open file: it does not appear to be a valid zip archive.
* CRC (cyclic redundancy check) error.
* **Compressed (ZIP folders error)** -- Compressed (ZIP) folder is invalid or corrupted.

1. The uncompressing process not only uses memory but also processor time. The process is slow and is a disadvantage when you are trying to access a file quickly. The processing time varies due to file size and the method used to compress the file. Also, running multiple applications on your system during the uncompressing process makes the task run longer.
2. When uncompressing a file for reading, your computer uses more memory to complete the task. During the uncompressing process, your computer will pause and allocate any free memory to complete the task. You may encounter one or more "Low Resources" or "Out of Memory" errors if your computer doesn't have enough memory. The errors also occur if you have multiple applications running at the same time while your computer is processing the compressed file.
3. Your computer's anti-virus program may not be able to scan a compressed file for viruses and other malware, resulting in your computer being exposed to files that may contain viruses, Trojans, spyware and other harmful programs. These types of programs can destroy your computer. Compressed files from unknown sources should be uncompressed only after a complete scan with an anti-virus program that is capable of scanning the compact files.
4. If you send a compressed file to someone else via the Internet, email or on a computer network, that person may not be able to open the file if they don't have a program to uncompress it. This disadvantage is easily resolved by including a link to download a free or shareware program to uncompress the file.
5. An increase in file size is another disadvantage of compressing files. In some situations including compressing video and audio files, the file you are compressing cannot be made smaller, resulting in a compressed file that is larger in size than the original file.

**1.3 Proposed System:**

We thought of designing a project where we are going to design a project named PACKER-UNPACKER where we are going to combine data from multiple files into single file and then extracting individual files from that single file at receiver side. Here we deals with the original data without any compression technique so that it is easier and faster to combine data and extract that data.

In order to achieve our target of developing the application PACKER-UNPACKER we have to take into consideration various requirements to have a complete application such as: -

1. The System will combine data from different files with different extension like .txt, .pdf, .jpg, .mp3, .png etc..
2. Our system also combine multiple zip files.
3. Providing security.
4. We also have to use some encryption techniques.
5. We can send it through different applications.
6. Avoiding data losses.

**Chapter 2**

**LITERATURE SURVEY**

We have been gathering information about how to integrate all the information a user needs to use and what requirements it will generate to do so. As we are thinking of developing both the Desktop Application it gives rise to many things from language to be used, Operating System.

# 2.1 What is Android Studio ?

**Android Studio** is the official [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) for [Google](https://en.wikipedia.org/wiki/Google)'s [Android](https://en.wikipedia.org/wiki/Android_(operating_system)) [operating system](https://en.wikipedia.org/wiki/Operating_system), built on [JetBrains](https://en.wikipedia.org/wiki/JetBrains)' [IntelliJ IDEA](https://en.wikipedia.org/wiki/IntelliJ_IDEA) software and designed specifically for [Android development](https://en.wikipedia.org/wiki/Android_software_development).[[8]](https://en.wikipedia.org/wiki/Android_Studio#cite_note-8) It is available for download on [Windows](https://en.wikipedia.org/wiki/Windows), [macOS](https://en.wikipedia.org/wiki/MacOS) and [Linux](https://en.wikipedia.org/wiki/Linux) based operating systems or as a subscription-based service in 2020.[[9]](https://en.wikipedia.org/wiki/Android_Studio#cite_note-9)[[10]](https://en.wikipedia.org/wiki/Android_Studio#cite_note-10) It is a replacement for the [Eclipse Android Development Tools](https://en.wikipedia.org/wiki/Eclipse_(software)#Android_Development_Tools) (E-ADT) as the primary IDE for native Android application development.

Android Studio was announced on May 16, 2013 at the [Google I/O](https://en.wikipedia.org/wiki/Google_I/O) conference. It was in early access 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.[[11]](https://en.wikipedia.org/wiki/Android_Studio#cite_note-11) The first stable build was released in December 2014, starting from version 1.0.[[12]](https://en.wikipedia.org/wiki/Android_Studio#cite_note-12)

On May 7, 2019, [Kotlin](https://en.wikipedia.org/wiki/Kotlin_(programming_language)) replaced [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) as Google's preferred language for Android app development. Java is still supported, as is [C++](https://en.wikipedia.org/wiki/C%2B%2B)

Android Studio supports all the same programming languages of [IntelliJ](https://en.wikipedia.org/wiki/IntelliJ) (and [CLion](https://en.wikipedia.org/wiki/CLion)) e.g. [Java](https://en.wikipedia.org/wiki/Java_(programming_language)), [C++](https://en.wikipedia.org/wiki/C%2B%2B), and more with extensions, such as [Go](https://en.wikipedia.org/wiki/Go_(programming_language));[[20]](https://en.wikipedia.org/wiki/Android_Studio#cite_note-20) and Android Studio 3.0 or later supports [Kotlin](https://en.wikipedia.org/wiki/Kotlin_(programming_language))[]](https://en.wikipedia.org/wiki/Android_Studio#cite_note-21) and "all Java 7 language features and a subset of Java 8 language features that vary by platform version."[[22]](https://en.wikipedia.org/wiki/Android_Studio#cite_note-22) External projects [backport](https://en.wikipedia.org/wiki/Backporting) some Java 9 features.[[23]](https://en.wikipedia.org/wiki/Android_Studio#cite_note-23) While IntelliJ states that Android Studio supports all released Java versions, and Java 12, it's not clear to what level Android Studio supports Java versions up to Java 12 (the documentation mentions partial Java 8 support). At least some new language features up to Java 12 are usable in Android.[[24]](https://en.wikipedia.org/wiki/Android_Studio#cite_note-24)

Once an app has been compiled with Android Studio, it can be published on the [Google Play Store](https://en.wikipedia.org/wiki/Google_Play_Store). The application has to be in line with the Google Play Store [developer content policy](https://play.google.com/about/developer-content-policy/)

# 2.2 What is JAVA?

Java is a programming language and computing platform first released by Sun Microsystems in 1995. There are lots of applications and websites that will not work unless you have Java installed, and more are created every day. Java is fast, secure, and reliable. From laptops to datacenters, game consoles to scientific supercomputers, cell phones to the Internet, Java is everywhere!

**Chapter 3**

**REQUIREMENT ANALYSIS**

**3.1 Operating Environment:**

**3.1.1. Hardware Requirement:**

* + 1. Intel core i5 or higher
    2. 8 GB RAM or higher
    3. Keyboard
    4. Mouse

**3.1.2. Software Requirement:**

* + 1. Windows O.S
    2. ANDROID STUDIO

**3.2 Design and Implementation Constraints:**

There are a few constraints of the system:-

1. The design should be user friendly.
2. There should be a separate logger for each of two applications.
3. The desktop applications should be installed properly.

**3.3 Assumption and Dependencies:**

We assume that the all the devices attached to the system will work properly without failure. All the devices will have a proper configuration for the application to perform smoothly.

There are a few dependencies of the system:

1. JDK 1.7 or higher should be present on the devices for the system to function properly.
2. The service account should be properly designed properly with all the permissions in place.

**3.4 System Features:**

**3.4.1 System Feature 1(function requirements):**

The system will have authentication system for the applications.

**3.4.2 System Feature 2(function requirements):**

The system will have to select operation which is to be perform.

**3.4.3 System Feature 3(function requirements):**

The system will have to select files.

**3.4.4 System Feature 4(function requirements):**

The system will maintain information of every individual file.

**3.5 External Interface Requirements:**

**3.5.1 User interfaces:**

In our system we provide a GUI on both Packer and Unpacker: -

3.5.1.1 Packer will have GUI for:

1. Choosing files
2. Packing Files

3.5.1.2 Unpacker will have GUI for:

1. Choosing files
2. Unpacking Files

**3.6 Non Functional Requirements:**

**3.6.1 Performance Requirement:**

1. Authentication should be known in order to use the applications.

**3.6.2 Safety Requirements:**

1. To keep this system safe care should be taken to avoid the theft of components of the system.
2. The input voltage for **devices** should not be more than the standards applied to them.

**3.6.3 Security Requirements:**

1. To make the system secure we are providing security to output file.

**Chapter 4**

**DESIGN**

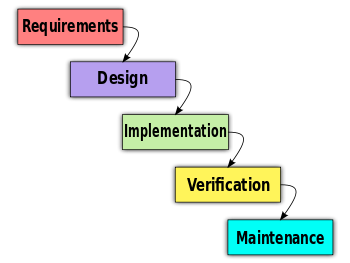
**4.1 Analysis Model**:

The waterfall model is a [sequential](http://en.wikipedia.org/wiki/Sequence) [design](http://en.wikipedia.org/wiki/Design) process, often used in [software development processes](http://en.wikipedia.org/wiki/Software_development_process), in which progress is seen as flowing steadily downwards (like a [waterfall](http://en.wikipedia.org/wiki/Waterfall)) through the phases of Conception, Initiation, [Analysis](http://en.wikipedia.org/wiki/Analysis), [Design](http://en.wikipedia.org/wiki/Software_design), Construction, [Testing](http://en.wikipedia.org/wiki/Software_testing), [Production/Implementation](http://en.wikipedia.org/wiki/Implementation), and [Maintenance](http://en.wikipedia.org/wiki/Software_maintenance).

The waterfall development model originates in the [manufacturing](http://en.wikipedia.org/wiki/Manufacturing) and [construction](http://en.wikipedia.org/wiki/Construction) industries: highly structured physical environments in which after-the-fact changes are prohibitively costly, if not impossible. Since no formal software development methodologies existed at the time, this hardware-oriented model was simply adapted for software development.

The first known presentation describing use of similar phases in software engineering was held by Herbert D. Benington at Symposium on advanced programming methods for digital computers on 29 June 1956. This presentation was about the development of software for [SAGE](http://en.wikipedia.org/wiki/Semi_Automatic_Ground_Environment). In 1983 the paper was republished with a foreword by Benington pointing out that the process was not in fact performed in strict top-down, but depended on a prototype.

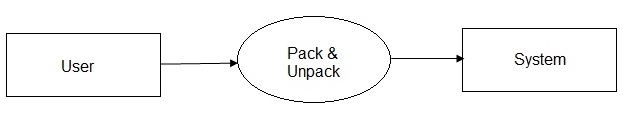
The first formal description of the waterfall model is often cited as a 1970 article by [Winston W. Royce](http://en.wikipedia.org/wiki/Winston_W._Royce), although Royce did not use the term "waterfall" in this article. Royce presented this model as an example of a flawed, non-working model. This, in fact, is how the term is generally used in writing about software development—to describe a critical view of a commonly used software development practice.



**Fig 4.1: Waterfall Model**

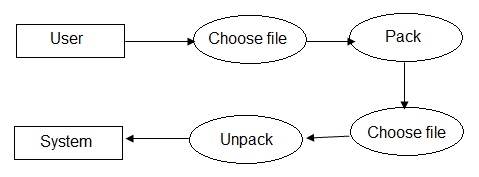
**4.2 DFD (Data Flow Diagram)**:

**4.2.1 DFD Level 0**:

****

**Fig 4.2: DFD Level 0**

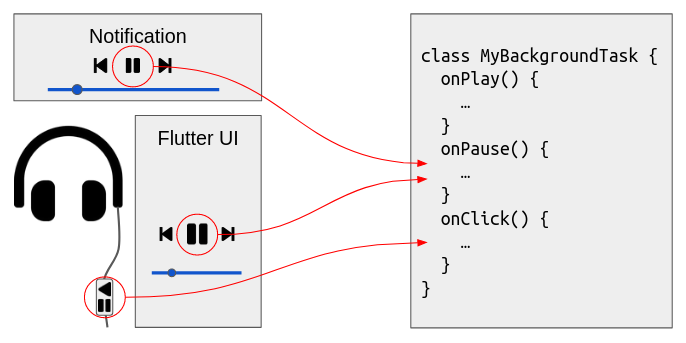
**4.2.2 DFD Level 1**:



**Fig 4.3: DFD Level 1**

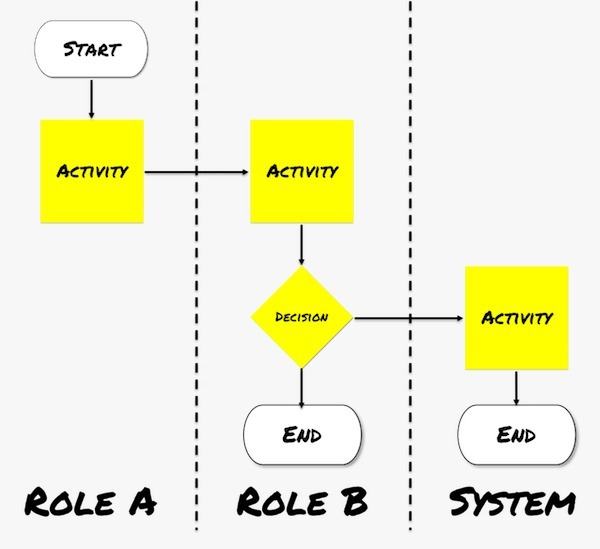
**4.3 UML Diagram**:

**4.3.1 Use-Case Diagram**:



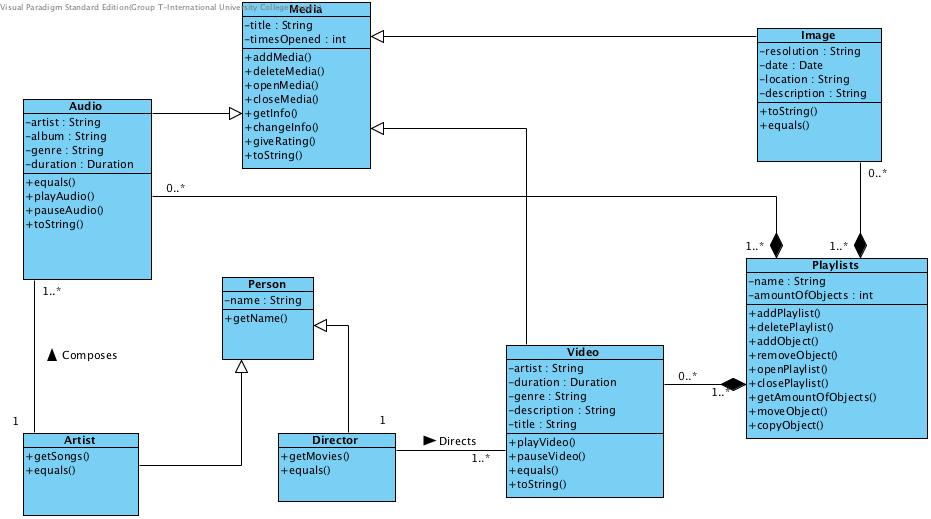
**Fig 4.4: Use Case Diagram**

**4.3.2 Sequence Diagram**:



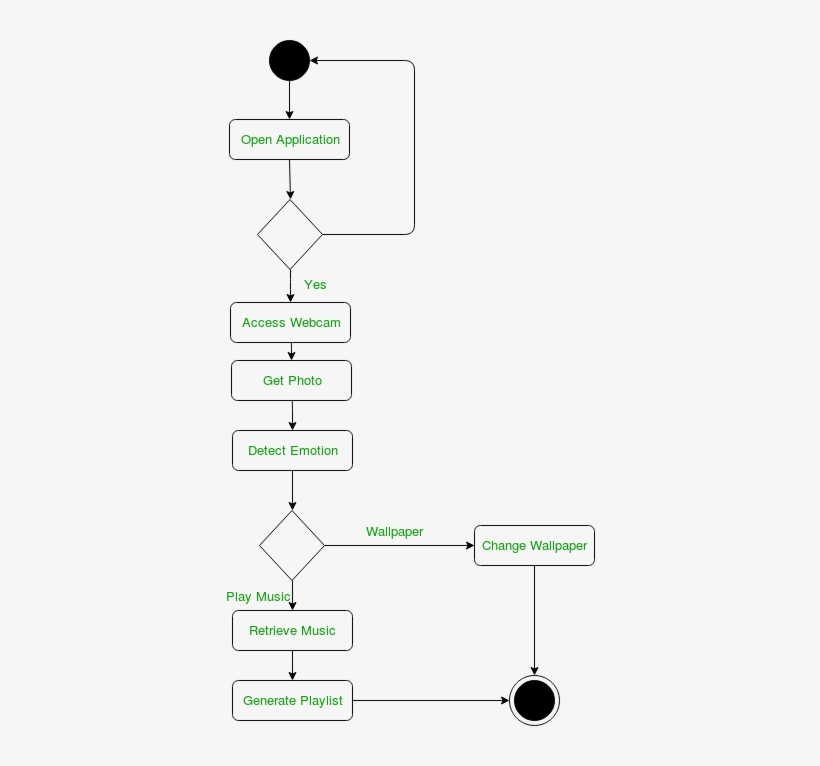
**Fig 4.5: Sequence Diagram**

**4.3.3 Class Diagram**:



**Fig 4.6: Class Diagram**

**4.3.4 Activity Diagram**:



**Fig 4.7: Activity Diagram**

**Chapter 5**

**CODING**

* 1. **Android Manifest :**

<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

package="com.example.metalhealthandselfimprovement">

<uses-permission android:name="android.permission.INTERNET" />

<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />

<application

android:allowBackup="true"

android:icon="@mipmap/ic\_launcher"

android:label="@string/app\_name"

android:networkSecurityConfig="@xml/network\_security\_config"

android:roundIcon="@mipmap/ic\_launcher\_round"

android:supportsRtl="true"

android:theme="@style/Theme.MetalHealthAndSelfImprovement">

<activity android:name=".Q13Answer"></activity>

<activity android:name=".Q12Answer" />

<activity android:name=".Q11Answer" />

<activity android:name=".Q10Answer" />

<activity android:name=".Q9Answer" />

<activity android:name=".Q8Answer" />

<activity android:name=".Q7Answer" />

<activity android:name=".Q6Answer" />

<activity android:name=".Q5Answer" />

<activity android:name=".Q4Answer" />

<activity android:name=".Q3Answer" />

<activity android:name=".Q2Answer" />

<activity android:name=".Q1Answer" />

<activity android:name=".QuestionsActivity" />

<activity android:name=".Zen\_Music\_Player" />

<activity android:name=".AbousUs" />

<activity android:name=".UserActivity" />

<activity android:name=".Dashboard" />

<activity android:name=".MainActivity">

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

</activity>

</application>

</manifest>

* 1. **Andorid Xml Main :**

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity"

android:background="#fece2f">

<ImageView

android:id="@+id/imageView"

android:layout\_width="303dp"

android:layout\_height="231dp"

android:layout\_marginStart="10dp"

android:layout\_marginLeft="10dp"

android:layout\_marginTop="56dp"

android:layout\_marginEnd="10dp"

android:layout\_marginRight="10dp"

android:foregroundGravity="center"

android:src="@drawable/logo"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.42"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<ImageView

android:id="@+id/imageView2"

android:layout\_width="292dp"

android:layout\_height="319dp"

android:layout\_marginStart="10dp"

android:layout\_marginLeft="10dp"

android:layout\_marginEnd="10dp"

android:layout\_marginRight="10dp"

android:foregroundGravity="center"

android:src="@drawable/flash"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.373"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.618" />

<TextView

android:id="@+id/textView"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:fontFamily="sans-serif-black"

android:gravity="center"

android:text=" Be The Better Version of Yourself"

android:textSize="18sp"

android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/imageView"

app:layout\_constraintVertical\_bias="0.666" />

</androidx.constraintlayout.widget.ConstraintLayout>

* 1. **Java main :**

package com.example.metalhealthandselfimprovement;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.media.MediaPlayer;

import android.os.Bundle;

import android.os.Handler;

import android.provider.MediaStore;

import android.view.Window;

import android.view.WindowManager;

import android.view.animation.Animation;

import android.view.animation.AnimationUtils;

import android.widget.ImageView;

import android.widget.TextView;

import static java.lang.Thread.sleep;

public class MainActivity extends AppCompatActivity {

private static int SPLASH\_SCREEN=5000;

Animation topAnim,bottomAnim,fadeAnim;

ImageView img\_logo,img\_flash;

TextView slogan;

@Override

protected void onCreate(Bundle savedInstanceState) {

requestWindowFeature(Window.FEATURE\_NO\_TITLE);

super.onCreate(savedInstanceState);

getWindow().setFlags(WindowManager.LayoutParams.FLAG\_FULLSCREEN,WindowManager.LayoutParams.FLAG\_FULLSCREEN);

getSupportActionBar().hide();

setContentView(R.layout.activity\_main);

topAnim= AnimationUtils.loadAnimation(this,R.anim.top\_animation);

bottomAnim= AnimationUtils.loadAnimation(this,R.anim.bottom\_animation);

fadeAnim=AnimationUtils.loadAnimation(this,R.anim.fade);

img\_logo=findViewById(R.id.imageView);

img\_flash=findViewById(R.id.imageView2);

slogan=findViewById(R.id.textView);

img\_logo.setAnimation(topAnim);

img\_flash.setAnimation(bottomAnim);

slogan.setAnimation(fadeAnim);

final MediaPlayer bg\_intro =MediaPlayer.create(this,R.raw.intro);

new Handler().postDelayed(new Runnable() {

@Override

public void run() {

bg\_intro.start();

try {

sleep(2000);

} catch (InterruptedException e) {

e.printStackTrace();

}

Intent intent=new Intent(MainActivity.this,Dashboard.class);

startActivity(intent);

finish();

}

},SPLASH\_SCREEN);

}

}

* 1. **Android Xml Dashboard -**

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="#fece2f"

tools:context=".Dashboard">

<ImageView

android:id="@+id/img\_new\_path\_sound"

android:layout\_width="37dp"

android:layout\_height="38dp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.991"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.687"

app:srcCompat="@drawable/ic\_sound" />

<ImageView

android:id="@+id/img\_presents"

android:layout\_width="357dp"

android:layout\_height="557dp"

android:layout\_marginBottom="312dp"

android:contentDescription="@string/todo"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.601"

app:srcCompat="@drawable/img\_presents" />

<Button

android:id="@+id/btn\_begin\_journey"

android:layout\_width="250dp"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:layout\_centerVertical="true"

android:background="@drawable/rounded\_button"

android:text="@string/begin\_journey"

android:textColor="#FFFFFFFF"

android:textSize="23sp"

android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.496"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.557" />

<Button

android:id="@+id/btn\_new\_path"

android:layout\_width="250dp"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:layout\_centerVertical="true"

android:background="@drawable/rounded\_button"

android:text="@string/new\_path"

android:textColor="#FFFFFFFF"

android:textSize="23sp"

android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.496"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.698" />

<TextView

android:id="@+id/textView3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:gravity="center"

android:text="@string/thanks\_for\_using\_our\_app"

android:textSize="24sp"

android:textStyle="bold|italic"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.496"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.909" />

<ImageView

android:id="@+id/img\_img\_sound"

android:layout\_width="37dp"

android:layout\_height="38dp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="1.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.367"

app:srcCompat="@drawable/ic\_sound" />

<ImageView

android:id="@+id/img\_begin\_jouney\_sound"

android:layout\_width="37dp"

android:layout\_height="38dp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.986"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.559"

app:srcCompat="@drawable/ic\_sound" />

</androidx.constraintlayout.widget.ConstraintLayout>

* 1. **Android Java Dashboard –**

package com.example.metalhealthandselfimprovement;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.media.MediaPlayer;

import android.os.Bundle;

import android.view.View;

import android.view.Window;

import android.view.WindowManager;

import android.widget.Button;

import android.widget.ImageView;

public class Dashboard extends AppCompatActivity {

Button button\_begin,button\_new\_path;

ImageView image\_presents,img\_img\_sound,img\_begin\_jouney\_sound,img\_new\_path\_sound;

@Override

protected void onCreate(Bundle savedInstanceState) {

requestWindowFeature(Window.FEATURE\_NO\_TITLE);

super.onCreate(savedInstanceState);

getWindow().setFlags(WindowManager.LayoutParams.FLAG\_FULLSCREEN,WindowManager.LayoutParams.FLAG\_FULLSCREEN);

getSupportActionBar().hide();

setContentView(R.layout.activity\_dashboard);

MediaPlayer aboutUS=MediaPlayer.create(this,R.raw.about\_us);

MediaPlayer journey1=MediaPlayer.create(this,R.raw.journey);

MediaPlayer journey2=MediaPlayer.create(this,R.raw.journey2);

button\_begin=findViewById(R.id.btn\_begin\_journey);

button\_new\_path=findViewById(R.id.btn\_new\_path);

image\_presents=findViewById(R.id.img\_presents);

img\_img\_sound=findViewById(R.id.img\_img\_sound);

img\_begin\_jouney\_sound=findViewById(R.id.img\_begin\_jouney\_sound);

img\_new\_path\_sound=findViewById(R.id.img\_new\_path\_sound);

img\_img\_sound.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if(journey1.isPlaying()){

journey1.stop();

}

if (journey2.isPlaying()){

journey2.stop();

}

aboutUS.start();

}

});

img\_begin\_jouney\_sound.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if(aboutUS.isPlaying()){

aboutUS.stop();

}

if(journey2.isPlaying()){

journey2.stop();

}

journey1.start();

}

});

img\_new\_path\_sound.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if(aboutUS.isPlaying()){

aboutUS.stop();

}

if(journey1.isPlaying()){

journey1.stop();

}

journey2.start();

}

});

button\_begin.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent intent\_userActivity=new Intent(Dashboard.this,QuestionsActivity.class);

startActivity(intent\_userActivity);

}

});

button\_new\_path.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent intent\_Zen\_Music=new Intent(Dashboard.this,Zen\_Music\_Player.class);

startActivity(intent\_Zen\_Music);

}

});

image\_presents.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent intent\_about\_us=new Intent(Dashboard.this,AbousUs.class);

startActivity(intent\_about\_us);

}

});

}

}

* 1. **Android About us Xml –**

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".AbousUs">

<ImageView

android:id="@+id/imageView3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:contentDescription="TODO"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:srcCompat="@drawable/aboutusinfo" />

</androidx.constraintlayout.widget.ConstraintLayout>

* 1. **Android About us Java –**

package com.example.metalhealthandselfimprovement;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.Window;

import android.view.WindowManager;

public class AbousUs extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

requestWindowFeature(Window.FEATURE\_NO\_TITLE);

getWindow().setFlags(WindowManager.LayoutParams.FLAG\_FULLSCREEN,WindowManager.LayoutParams.FLAG\_FULLSCREEN);

getSupportActionBar().hide();

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_abous\_us);

}

}

* 1. **Android user xml –**

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".UserActivity">

<WebView

android:id="@+id/webView"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent" />

</LinearLayout>

* 1. **Android user Java –**

package com.example.metalhealthandselfimprovement;

import androidx.appcompat.app.AppCompatActivity;

import android.annotation.SuppressLint;

import android.os.Bundle;

import android.view.KeyEvent;

import android.view.Window;

import android.view.WindowManager;

import android.webkit.WebSettings;

import android.webkit.WebView;

import android.webkit.WebViewClient;

public class UserActivity extends AppCompatActivity {

WebView web;

@SuppressLint("SetJavaScriptEnabled")

@Override

protected void onCreate(Bundle savedInstanceState) {

requestWindowFeature(Window.FEATURE\_NO\_TITLE);

super.onCreate(savedInstanceState);

getWindow().setFlags(WindowManager.LayoutParams.FLAG\_FULLSCREEN,WindowManager.LayoutParams.FLAG\_FULLSCREEN);

getSupportActionBar().hide();

setContentView(R.layout.activity\_user);

web=findViewById(R.id.webView);

WebSettings webSettings=web.getSettings();

webSettings.setJavaScriptEnabled(true);

web.setWebViewClient(new Callback());

web.loadUrl("https://www.intellect.co/");

}

private class Callback extends WebViewClient {

@Override

public boolean shouldOverrideKeyEvent(WebView view, KeyEvent event){

return false;

}

}

}

* 1. **Android music xml –**

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="#263238"

tools:context=".Zen\_Music\_Player">

<Button

android:id="@+id/btn\_zen1"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_marginTop="112dp"

android:shadowColor="#9A8F8F"

android:text="ZEN 1"

android:textColor="#133FE3"

app:backgroundTint="#FFFFFF"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:rippleColor="#C3AFAF"

app:strokeColor="#D5A2A2" />

<Button

android:id="@+id/btn\_zen2"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:text="ZEN 2"

android:textColor="#133FE3"

app:backgroundTint="#FFFFFF"

app:iconTint="#FFFFFF"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/btn\_zen1" />

<Button

android:id="@+id/btn\_zen3"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:text="ZEN 3"

android:textColor="#133FE3"

app:backgroundTint="#FFFFFF"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/btn\_zen2" />

<Button

android:id="@+id/btn\_zen4"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:text="ZEN 4"

android:textColor="#133FE3"

app:backgroundTint="#FFFFFF"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/btn\_zen3" />

<Button

android:id="@+id/btn\_zen10"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="163dp"

android:text="ZEN 10"

android:textColor="#133FE3"

app:backgroundTint="#FFFFFF"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/btn\_zen9"

app:layout\_constraintVertical\_bias="0.0" />

<Button

android:id="@+id/btn\_zen5"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:text="ZEN 5"

android:textColor="#133FE3"

app:backgroundTint="#FFFFFF"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/btn\_zen4" />

<Button

android:id="@+id/btn\_zen6"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:text="ZEN 6"

android:textColor="#133FE3"

app:backgroundTint="#FFFFFF"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/btn\_zen5" />

<Button

android:id="@+id/btn\_zen7"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:text="ZEN 7"

android:textColor="#133FE3"

app:backgroundTint="#FFFFFF"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/btn\_zen6" />

<Button

android:id="@+id/btn\_zen8"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:text="ZEN 8"

android:textColor="#133FE3"

app:backgroundTint="#FFFFFF"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/btn\_zen7" />

<Button

android:id="@+id/btn\_zen9"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:text="ZEN 9"

android:textColor="#133FE3"

app:backgroundTint="#FFFFFF"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/btn\_zen8" />

<TextView

android:id="@+id/textView2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text=" ZEN LIBRARY"

android:textSize="30sp"

android:textStyle="bold|italic"

app:layout\_constraintBottom\_toTopOf="@+id/btn\_zen1"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<TextView

android:id="@+id/textView4"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Be the beter version of yourself"

android:textStyle="bold|italic"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.497"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/btn\_zen10"

app:layout\_constraintVertical\_bias="0.641" />

</androidx.constraintlayout.widget.ConstraintLayout>

* 1. Android music java –

package com.example.metalhealthandselfimprovement;

import androidx.appcompat.app.AppCompatActivity;

import android.media.MediaPlayer;

import android.os.Bundle;

import android.view.View;

import android.view.Window;

import android.view.WindowManager;

import android.widget.Button;

import android.widget.TextView;

public class Zen\_Music\_Player extends AppCompatActivity {

Button btn\_zen1,btn\_zen2,btn\_zen3,btn\_zen4,btn\_zen5,btn\_zen6,btn\_zen7,btn\_zen8,btn\_zen9,btn\_zen10;

MediaPlayer mediaPlayer;

TextView tv\_ZEN1,tv\_ZEN2,tv\_ZEN3,tv\_ZEN4,tv\_ZEN5,tv\_ZEN6,tv\_ZEN7,tv\_ZEN8,tv\_ZEN9,tv\_ZEN10;

@Override

protected void onCreate(Bundle savedInstanceState) {

requestWindowFeature(Window.FEATURE\_NO\_TITLE);

super.onCreate(savedInstanceState);

getWindow().setFlags(WindowManager.LayoutParams.FLAG\_FULLSCREEN,WindowManager.LayoutParams.FLAG\_FULLSCREEN);

getSupportActionBar().hide();

setContentView(R.layout.activity\_zen\_\_music\_\_player);

MediaPlayer ZEN1=MediaPlayer.create(this,R.raw.zen1);

MediaPlayer ZEN2=MediaPlayer.create(this,R.raw.zen2);

MediaPlayer ZEN3=MediaPlayer.create(this,R.raw.zen3);

MediaPlayer ZEN4=MediaPlayer.create(this,R.raw.zen4);

MediaPlayer ZEN5=MediaPlayer.create(this,R.raw.zen5);

MediaPlayer ZEN6=MediaPlayer.create(this,R.raw.zen6);

MediaPlayer ZEN7=MediaPlayer.create(this,R.raw.zen7);

MediaPlayer ZEN8=MediaPlayer.create(this,R.raw.zen8);

MediaPlayer ZEN9=MediaPlayer.create(this,R.raw.zen9);

MediaPlayer ZEN10=MediaPlayer.create(this,R.raw.zen10);

btn\_zen1=findViewById(R.id.btn\_zen1);

btn\_zen2=findViewById(R.id.btn\_zen2);

btn\_zen3=findViewById(R.id.btn\_zen3);

btn\_zen4=findViewById(R.id.btn\_zen4);

btn\_zen5=findViewById(R.id.btn\_zen5);

btn\_zen6=findViewById(R.id.btn\_zen6);

btn\_zen7=findViewById(R.id.btn\_zen7);

btn\_zen8=findViewById(R.id.btn\_zen8);

btn\_zen9=findViewById(R.id.btn\_zen9);

btn\_zen10=findViewById(R.id.btn\_zen10);

btn\_zen1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if (ZEN1.isPlaying()){

ZEN1.pause();

}else{

if(ZEN2.isPlaying()){

ZEN2.pause();

}

if(ZEN3.isPlaying()){

ZEN3.pause();

}

if(ZEN4.isPlaying()){

ZEN4.pause();

}

if(ZEN5.isPlaying()){

ZEN5.pause();

}

if(ZEN6.isPlaying()){

ZEN6.pause();

}

if(ZEN7.isPlaying()){

ZEN7.pause();

}

if(ZEN8.isPlaying()){

ZEN8.pause();

}

if(ZEN9.isPlaying()){

ZEN9.pause();

}

if(ZEN10.isPlaying()){

ZEN10.pause();

}

ZEN1.start();

}

if(ZEN1.isPlaying()){

btn\_zen1.setText(("ZEN1 is Playing"));

btn\_zen2.setText(("ZEN2"));

btn\_zen3.setText(("ZEN3"));

btn\_zen4.setText(("ZEN4"));

btn\_zen5.setText(("ZEN5"));

btn\_zen6.setText(("ZEN6"));

btn\_zen7.setText(("ZEN7"));

btn\_zen8.setText(("ZEN8"));

btn\_zen9.setText(("ZEN9"));

btn\_zen10.setText(("ZEN10"));

}else{

btn\_zen1.setText(("ZEN1 is Paused"));

}}

});

btn\_zen2.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if (ZEN2.isPlaying()){

ZEN2.pause();

}else{

if(ZEN1.isPlaying()){

ZEN1.pause();

}

if(ZEN3.isPlaying()){

ZEN3.pause();

}

if(ZEN4.isPlaying()){

ZEN4.pause();

}

if(ZEN5.isPlaying()){

ZEN5.pause();

}

if(ZEN6.isPlaying()){

ZEN6.pause();

}

if(ZEN7.isPlaying()){

ZEN7.pause();

}

if(ZEN8.isPlaying()){

ZEN8.pause();

}

if(ZEN9.isPlaying()){

ZEN9.pause();

}

if(ZEN10.isPlaying()){

ZEN10.pause();

}

ZEN2.start();

}if(ZEN2.isPlaying()){

btn\_zen2.setText(("ZEN2 is Playing"));

btn\_zen1.setText(("ZEN1"));

btn\_zen3.setText(("ZEN3"));

btn\_zen4.setText(("ZEN4"));

btn\_zen5.setText(("ZEN5"));

btn\_zen6.setText(("ZEN6"));

btn\_zen7.setText(("ZEN7"));

btn\_zen8.setText(("ZEN8"));

btn\_zen9.setText(("ZEN9"));

btn\_zen10.setText(("ZEN10"));

}else{

btn\_zen2.setText(("ZEN2 is Paused"));

}}

});

btn\_zen3.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if (ZEN3.isPlaying()){

ZEN3.pause();

}else{

if(ZEN2.isPlaying()){

ZEN2.pause();

}

if(ZEN1.isPlaying()){

ZEN1.pause();

}

if(ZEN4.isPlaying()){

ZEN4.pause();

}

if(ZEN5.isPlaying()){

ZEN5.pause();

}

if(ZEN6.isPlaying()){

ZEN6.pause();

}

if(ZEN7.isPlaying()){

ZEN7.pause();

}

if(ZEN8.isPlaying()){

ZEN8.pause();

}

if(ZEN9.isPlaying()){

ZEN9.pause();

}

if(ZEN10.isPlaying()){

ZEN10.pause();

}

ZEN3.start();

}

if(ZEN3.isPlaying()){

btn\_zen3.setText(("ZEN3 is Playing"));

btn\_zen2.setText(("ZEN2"));

btn\_zen1.setText(("ZEN1"));

btn\_zen4.setText(("ZEN4"));

btn\_zen5.setText(("ZEN5"));

btn\_zen6.setText(("ZEN6"));

btn\_zen7.setText(("ZEN7"));

btn\_zen8.setText(("ZEN8"));

btn\_zen9.setText(("ZEN9"));

btn\_zen10.setText(("ZEN10"));

}else{

btn\_zen3.setText(("ZEN3 is Paused"));

}

}

});

btn\_zen4.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if (ZEN4.isPlaying()){

ZEN4.pause();

}else{

if(ZEN2.isPlaying()){

ZEN2.pause();

}

if(ZEN3.isPlaying()){

ZEN3.pause();

}

if(ZEN1.isPlaying()){

ZEN1.pause();

}

if(ZEN5.isPlaying()){

ZEN5.pause();

}

if(ZEN6.isPlaying()){

ZEN6.pause();

}

if(ZEN7.isPlaying()){

ZEN7.pause();

}

if(ZEN8.isPlaying()){

ZEN8.pause();

}

if(ZEN9.isPlaying()){

ZEN9.pause();

}

if(ZEN10.isPlaying()){

ZEN10.pause();

}

ZEN4.start();

}

if(ZEN4.isPlaying()){

btn\_zen4.setText(("ZEN4 is Playing"));

btn\_zen2.setText(("ZEN2"));

btn\_zen3.setText(("ZEN3"));

btn\_zen1.setText(("ZEN1"));

btn\_zen5.setText(("ZEN5"));

btn\_zen6.setText(("ZEN6"));

btn\_zen7.setText(("ZEN7"));

btn\_zen8.setText(("ZEN8"));

btn\_zen9.setText(("ZEN9"));

btn\_zen10.setText(("ZEN10"));

}else{

btn\_zen4.setText(("ZEN4 is Paused"));

}

}

});

btn\_zen5.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if (ZEN5.isPlaying()){

ZEN5.pause();

}else{

if(ZEN2.isPlaying()){

ZEN2.pause();

}

if(ZEN3.isPlaying()){

ZEN3.pause();

}

if(ZEN4.isPlaying()){

ZEN4.pause();

}

if(ZEN1.isPlaying()){

ZEN1.pause();

}

if(ZEN6.isPlaying()){

ZEN6.pause();

}

if(ZEN7.isPlaying()){

ZEN7.pause();

}

if(ZEN8.isPlaying()){

ZEN8.pause();

}

if(ZEN9.isPlaying()){

ZEN9.pause();

}

if(ZEN10.isPlaying()){

ZEN10.pause();

}

ZEN5.start();

}

if(ZEN5.isPlaying()){

btn\_zen5.setText(("ZEN5 is Playing"));

btn\_zen2.setText(("ZEN2"));

btn\_zen3.setText(("ZEN3"));

btn\_zen4.setText(("ZEN4"));

btn\_zen1.setText(("ZEN1"));

btn\_zen6.setText(("ZEN6"));

btn\_zen7.setText(("ZEN7"));

btn\_zen8.setText(("ZEN8"));

btn\_zen9.setText(("ZEN9"));

btn\_zen10.setText(("ZEN10"));

}else{

btn\_zen5.setText(("ZEN5 is Paused"));

}

}

});

btn\_zen6.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if (ZEN6.isPlaying()){

ZEN6.pause();

}else{

if(ZEN2.isPlaying()){

ZEN2.pause();

}

if(ZEN3.isPlaying()){

ZEN3.pause();

}

if(ZEN4.isPlaying()){

ZEN4.pause();

}

if(ZEN5.isPlaying()){

ZEN5.pause();

}

if(ZEN1.isPlaying()){

ZEN1.pause();

}

if(ZEN7.isPlaying()){

ZEN7.pause();

}

if(ZEN8.isPlaying()){

ZEN8.pause();

}

if(ZEN9.isPlaying()){

ZEN9.pause();

}

if(ZEN10.isPlaying()){

ZEN10.pause();

}

ZEN6.start();

}

if(ZEN6.isPlaying()){

btn\_zen6.setText(("ZEN6 is Playing"));

btn\_zen2.setText(("ZEN2"));

btn\_zen3.setText(("ZEN3"));

btn\_zen4.setText(("ZEN4"));

btn\_zen5.setText(("ZEN5"));

btn\_zen1.setText(("ZEN1"));

btn\_zen7.setText(("ZEN7"));

btn\_zen8.setText(("ZEN8"));

btn\_zen9.setText(("ZEN9"));

btn\_zen10.setText(("ZEN10"));

}else{

btn\_zen6.setText(("ZEN6 is Paused"));

}

});

btn\_zen7.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if (ZEN7.isPlaying()){

ZEN7.pause();

}else{

if(ZEN2.isPlaying()){

ZEN2.pause();

}

if(ZEN3.isPlaying()){

ZEN3.pause();

}

if(ZEN4.isPlaying()){

ZEN4.pause();

}

if(ZEN5.isPlaying()){

ZEN5.pause();

}

if(ZEN6.isPlaying()){

ZEN6.pause();

}

if(ZEN1.isPlaying()){

ZEN1.pause();

}

if(ZEN8.isPlaying()){

ZEN8.pause();

}

if(ZEN9.isPlaying()){

ZEN9.pause();

}

if(ZEN10.isPlaying()){

ZEN10.pause();

}

ZEN7.start();

}

if(ZEN7.isPlaying()){

btn\_zen7.setText(("ZEN7 is Playing"));

btn\_zen2.setText(("ZEN2"));

btn\_zen3.setText(("ZEN3"));

btn\_zen4.setText(("ZEN4"));

btn\_zen5.setText(("ZEN5"));

btn\_zen6.setText(("ZEN6"));

btn\_zen1.setText(("ZEN1"));

btn\_zen8.setText(("ZEN8"));

btn\_zen9.setText(("ZEN9"));

btn\_zen10.setText(("ZEN10"));

}else{

btn\_zen7.setText(("ZEN7 is Paused"));

}

}

});

btn\_zen8.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if (ZEN8.isPlaying()){

ZEN8.pause();

}else{

if(ZEN2.isPlaying()){

ZEN2.pause();

}

if(ZEN3.isPlaying()){

ZEN3.pause();

}

if(ZEN4.isPlaying()){

ZEN4.pause();

}

if(ZEN5.isPlaying()){

ZEN5.pause();

}

if(ZEN6.isPlaying()){

ZEN6.pause();

}

if(ZEN7.isPlaying()){

ZEN7.pause();

}

if(ZEN1.isPlaying()){

ZEN1.pause();

}

if(ZEN9.isPlaying()){

ZEN9.pause();

}

if(ZEN10.isPlaying()){

ZEN10.pause();

}

ZEN8.start();

}

if(ZEN8.isPlaying()){

btn\_zen8.setText(("ZEN8 is Playing"));

btn\_zen2.setText(("ZEN2"));

btn\_zen3.setText(("ZEN3"));

btn\_zen4.setText(("ZEN4"));

btn\_zen5.setText(("ZEN5"));

btn\_zen6.setText(("ZEN6"));

btn\_zen7.setText(("ZEN7"));

btn\_zen1.setText(("ZEN1"));

btn\_zen9.setText(("ZEN9"));

btn\_zen10.setText(("ZEN10"));

}else{

btn\_zen8.setText(("ZEN8 is Paused"));

} }

})

btn\_zen9.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if (ZEN9.isPlaying()){

ZEN9.pause();

}else{

if(ZEN2.isPlaying()){

ZEN2.pause();

}

if(ZEN3.isPlaying()){

ZEN3.pause();

}

if(ZEN4.isPlaying()){

ZEN4.pause();

}

if(ZEN5.isPlaying()){

ZEN5.pause();

}

if(ZEN6.isPlaying()){

ZEN6.pause();

}

if(ZEN7.isPlaying()){

ZEN7.pause();

}

if(ZEN8.isPlaying()){

ZEN8.pause();

}

if(ZEN1.isPlaying()){

ZEN1.pause();

}

if(ZEN10.isPlaying()){

ZEN10.pause();

}

ZEN9.start();

}

if(ZEN9.isPlaying()){

btn\_zen9.setText(("ZEN9 is Playing"));

btn\_zen2.setText(("ZEN2"));

btn\_zen3.setText(("ZEN3"));

btn\_zen4.setText(("ZEN4"));

btn\_zen5.setText(("ZEN5"));

btn\_zen1.setText(("ZEN1"));

btn\_zen7.setText(("ZEN7"));

btn\_zen8.setText(("ZEN8"));

btn\_zen1.setText(("ZEN1"));

btn\_zen10.setText(("ZEN10"));

}else{

btn\_zen9.setText(("ZEN9 is Paused"));

}

}

});

btn\_zen10.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if (ZEN10.isPlaying()){

ZEN10.pause();

}else{

if(ZEN2.isPlaying()){

ZEN2.pause();

}

if(ZEN3.isPlaying()){

ZEN3.pause();

}

if(ZEN4.isPlaying()){

ZEN4.pause();

}

if(ZEN5.isPlaying()){

ZEN5.pause();

}

if(ZEN6.isPlaying()){

ZEN6.pause();

}

if(ZEN7.isPlaying()){

ZEN7.pause();

}

if(ZEN8.isPlaying()){

ZEN8.pause();

}

if(ZEN9.isPlaying()){

ZEN9.pause();

}

if(ZEN1.isPlaying()){

ZEN1.pause();

}

ZEN10.start();

}

if(ZEN10.isPlaying()){

btn\_zen10.setText(("ZEN10 is Playing"));

btn\_zen2.setText(("ZEN2"));

btn\_zen3.setText(("ZEN3"));

btn\_zen4.setText(("ZEN4"));

btn\_zen5.setText(("ZEN5"));

btn\_zen1.setText(("ZEN1"));

btn\_zen7.setText(("ZEN7"));

btn\_zen8.setText(("ZEN8"));

btn\_zen9.setText(("ZEN9"));

btn\_zen1.setText(("ZEN1"));

}else{

btn\_zen10.setText(("ZEN10 is Paused"));

}

});

}

}

**Chapter 6**

**TESTING**

**6.1 Software Testing:**

**6.1.1. Black Box Testing:**

The term black box refers to the software which is treated as a black box. By treating it as a black box, we mean that the system or the source codes not checked at all. It is done from customer’s view point. The test engineer in black box testing only knows the set of inputs and the expected outputs and is unaware how those inputs are transformed into outputs by the software.

Various Black Box testing techniques are:

1) Boundary Value Analysis (BVA)

2) Robustness testing.

3) Worst case testing.

**6.1.2 White Box Testing:**

White box testing is a way of testing the external functionality of the code by examining and testing the program code that realizes the external functionality. It is a methodology to design test cases that uses the control structure of the application to design the test. White box testing is used to test the program code, code structure and internal design flow.

White box testing types:

1) Static white box testing

2) Dynamic white box testing.

**6.1.3 Gray Box Testing:**

Gray box testing consists of methods and tools derived from the knowledge of the application internals and the environment with which it interacts, that can be applied in black box testing to enhance testing productivity, bug finding and bug analyzing efficiency. It incorporates the elements of both black box as well as white box testing. It considers the outcome on the user end, system specific knowledge and the operating system.

**Table 6.1: Test Cases for Pack and unpack Module**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Description** | **Expected Result** | **Actual Result** | **Remark** |
| Animation | Click on App | Animation will roll up on screen | Animation Successful | Pass |
| About | Click on Imaage | About Page should be open. | About Us page is successfully Open | Pass |
| Jr button | Click on Start your Journey button | Next Screen should be open | Screen opened successfully | Pass |
| Music Btn | Click on Zen Music button | Music screen should be open | Music screen successfully displayed | Pass |
| Play Music | Click on Music on your screen | Music Should be played | Music played successfully | Pass |
| Question | Click on Question | Answer screen should be directed | Answer screen directed successfully | Pass |
| Back | Click on Back Icon | Screen should go back to previous screen | Task Successfully | Pass |

**Chapter 7**

**COST ESTIMATION**

**7.1 Phase Description:**

**Table 7.1: Phase Description**

|  |  |  |
| --- | --- | --- |
| **Phase** | **Task** | **Description** |
| Phase 1 | Topic Selection | Search the project title as per the market need. |
| Phase 2 | Information Gathering | Collect raw data according the topic selected. |
| Phase 3 | Synopsis Preparation | Preparation of synopsis and describe requirements as per topic selected |
| Phase 4 | Design | Preparation of UML diagrams. |
| Phase 5 | System Architecture | Prepare System Architecture describing the project workflow. |
| Phase 6 | Coding | Implement the code for all the modules and integrate all the modules. |
| Phase 7 | Testing Of Project | Test the code and overall process weather the process works properly. |
| Phase 8 | Deployment | Finalize the project by correcting the suggestions given by Guide. |
| Phase 9 | Report Writing | Prepare the document for this project with conclusion and future enhancement. |

**7.2 Project Plan:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Task | 2021 |  |  | Start | End |
|  | April | May | June |  |  |
| Topic Selection |  |  |  | 1/4/2021 | 5/4/2021 |
| Information gathering |  |  |  | 6/4/2021 | 12/4/2021 |
| Synopsis Prep |  |  | 13/4/21 | 20/4/21 |
| Design |  | |  | 21/4/21 | 15/5/21 |
| System Architecture |  |  |  | 16/5/21 | 21/5/21 |
| Coding |  |  | | 22/5/21 | 9/6/2021 |
| Testing |  |  |  | 9/6/2021 | 11/6/2021 |
| Deployment |  |  |  | 12/6/2021 | 15/6/21 |
| Report Writing |  |  |  | 16/6/21 | 20/6/21 |

**Fig 7.1: chart for Project Planning**

**7.3 Estimation of KLOC:**

The number of lines required for implementation of various modules can be estimated as follows:

**Table 7.2: Estimation of KLOC**

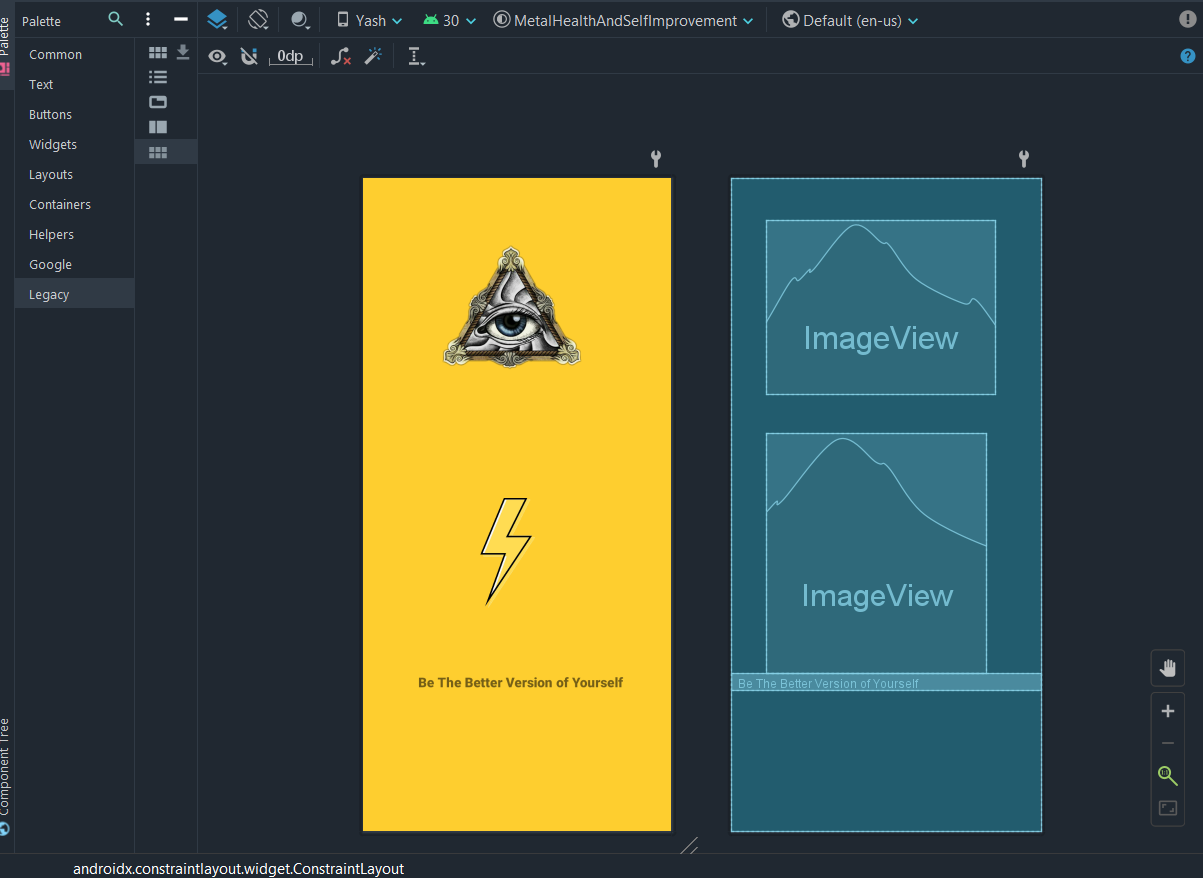
|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Modules** | **KLOC** |
| 1. | Graphical User Interface | 0.45 |
| 2. | User authentication Code | 0.35 |
| 3. | Java Code | 0.50 |
| 4. | Xml Code | 0.55 |
| 5. | Device Drivers | 0.35 |
| 6. | Interfacing Code | 0.40 |

Thus the total number of lines required is approximately 2.60 KLOC

**Chapter 8**

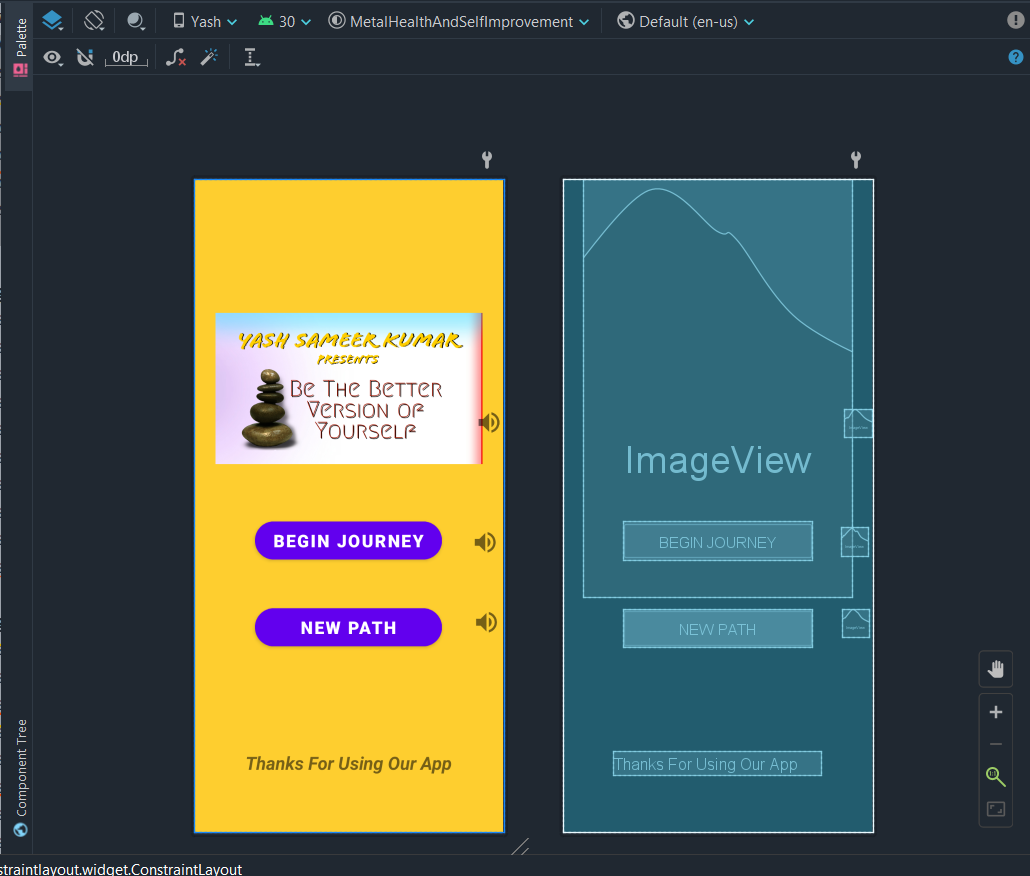
**RESULT**

**8.1 Start Screen:**

****

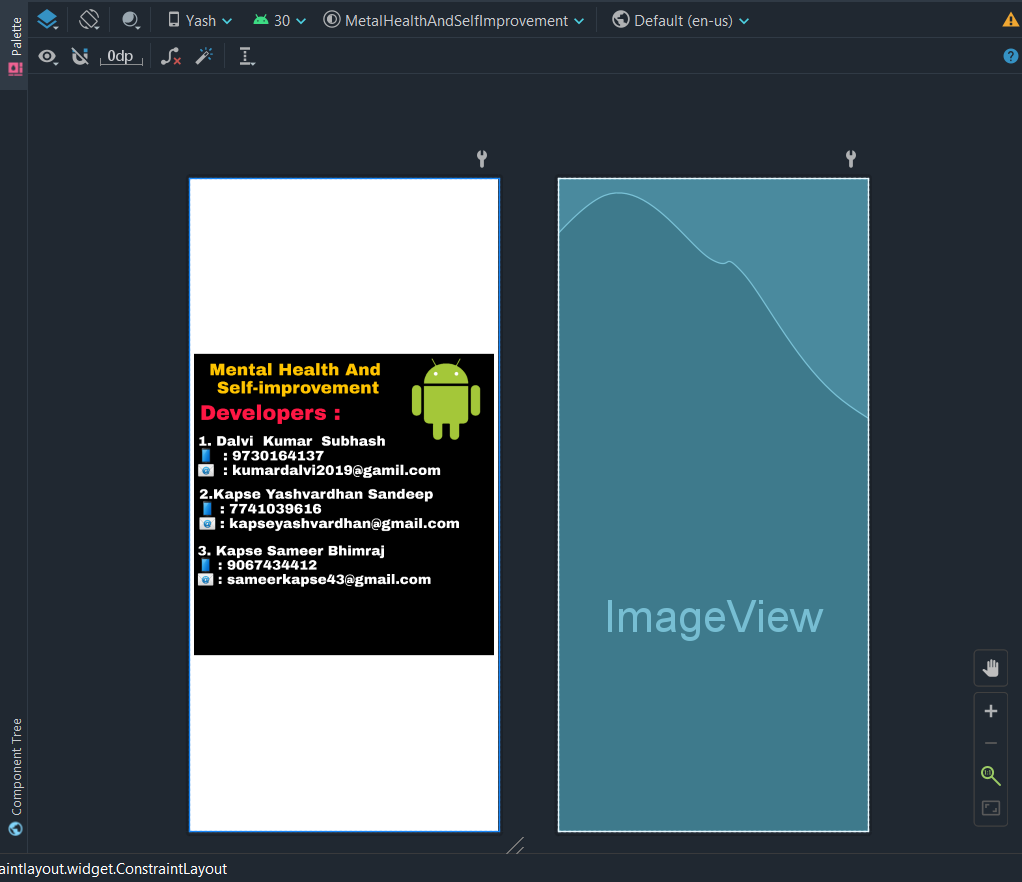
**Fig 8.1: Animation (Start Screen)**

**8.2 Home Screen / Dashboard :**



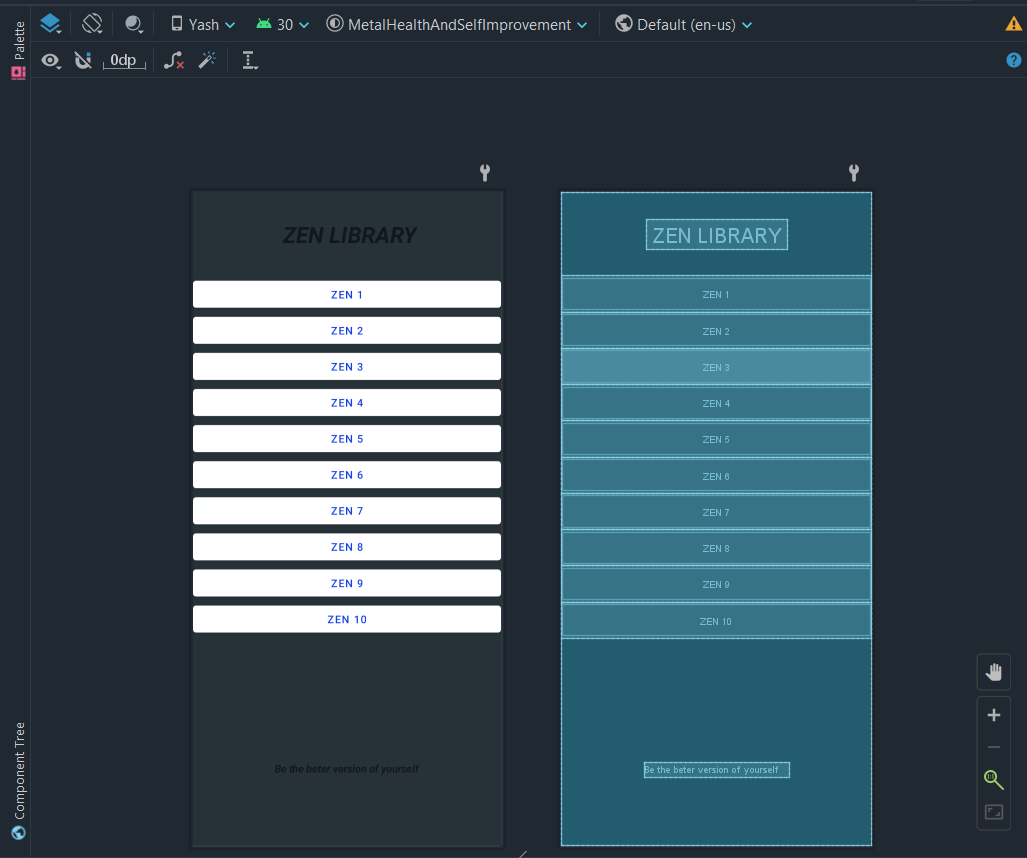
**Fig 8.2: Dashboard**

**8.3 About Us :**



**Fig 8.3: Choose file to pack**

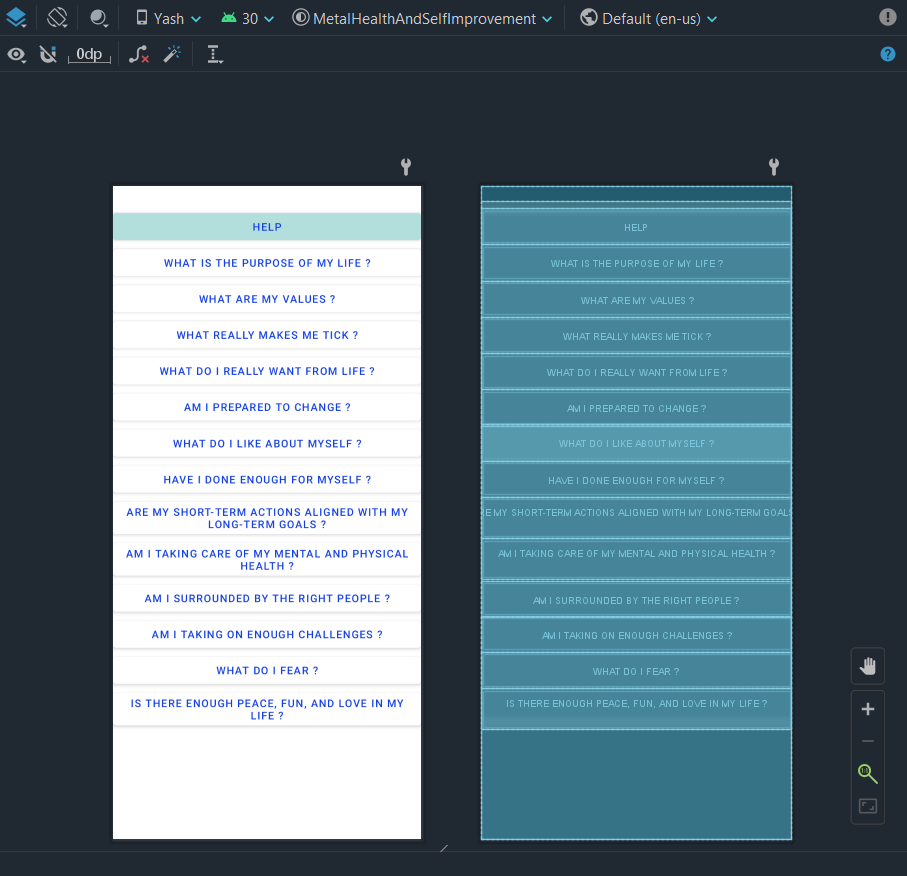
**8.4 Zen Music Screen -**

****

**Fig 8.4: Zen Music**

**\**

**8.5 Question Set Screen**

****

**Fig 8.5: Self Finding Screen**

**CONCLUSION**

This thesis has shown what Android is and how it works with XML in combination with Java. It has described how to setup a development environment and the emulator AVD.

We were successful in making an app, which can understand our question and give us a defining answer to overcome that question and connecting all the strings and making an Music section for people who needs an calmness into their life. It was a great journey from the start to now.

So henceforth, we have finally joined all the dots to make this possible.

**FUTURE SCOPE**

Our project technologies can be extended in to following future scope:

* 1. In the world of today, Mental health app is must and this is just an small initiative and there is a lot more than this.
  2. Music has a lot potential in todays world and has a big future in store.
  3. Infact, the [Android Architecture](https://data-flair.training/blogs/android-architecture/) by Google is a lot helpful in defining the best practices. It makes Android App development much easier and understandable to handle.
  4. There’s a drastic rise in the number of Android Applications today. Infact, real-time applications are more in trend that helps in making payment, or shopping online

**REFERENCES**

1. <https://developer.android.com/>
2. [https://developers.google.com](https://developers.google.com/drive/v3/web/about-sdk)
3. <https://www.tutorialspoint.com/>
4. https://www.guru99.com
5. <https://www.codewithharry.com/>
6. <https://www.youtube.com/watch?v=mXjZQX3UzOs>
7. <https://www.youtube.com/watch?v=JLIFqqnSNmg>
8. [https://developers.google.com](https://developers.google.com/sheets/api/quickstart/dotnet)
9. [https://stackoverflow.com](https://stackoverflow.com/questions/725627/accessing-google-spreadsheets-with-c-sharp-using-google-data-api)
10. Prof. Sachin Bhosale Sir (sachin.bhosale@pravara.in)