

1. INTRODUCTION

1.1 Overview of the project:

The project “DEMONSTRATION OF ASYNCHRONOUS TASK”, is a single web-page implementing the functionality of a simple moving banner. The "STOP" and "START" button lets the user start the movement of the banner.

We have the facility to start the movement of the banner if we want. If we wish, we can stop the movement of the banner with a single "STOP" button. It is a very user-friendly and user-interactive project.

Demonstration of asynchronous task project is an application specially built to implement the functionality of a simple moving banner. On pressing the Start button, the banner message should scroll from right to left. On pressing the Stop button, the banner message movement should stop. The banner message displayed is “Demonstration of Asynchronous Task”.

1.2 Purpose

The project has been implemented in modules so as to make it user friendly i.e., if a code viewer is finding trouble with a particular function it can go to a particular module and make corresponding changes in it instead of searching in the program.

2.REQUIREMENT SPECIFICATIONS

A software requirement definition is an abstract description of the services which the system should provide, and the constraints under which the system must operate. It should only specify the external behaviour of the system

2.1 Software and Hardware requirements

	Microsoft Windows	Mac	Linux
Operating System Version	Microsoft Windows 7/8/10 (32- or 64-bit) The Android Emulator	Mac OS X 10.10 (Yosemite) or higher, up	GNOME or KDE desktop Tested on Linux

	only supports 64-bit Windows.	to 10.14 (macOS Mojave)	based on Debian (4.19.67-2rodete2).
Random Access Memory (RAM)	4 GB RAM minimum; 8 GB RAM recommended.		
Free Digital Storage	2 GB of available digital storage minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image).		
Minimum Required JDK Version	Java Development Kit 8		
Minimum Screen Resolution	1280 x 800		

2.2 FUNCTIONAL REQUIREMENTS:

In software engineering, a functional requirement defines a function of a software system or its component. A function is described as a set of inputs, the behaviour, and outputs (see also software).

Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. Behavioural requirements describing all the cases where the system uses the functional requirements are captured in use cases.

The various methods used in this project are as follows: -

❖ **Emulator**

To perform and display the functionality of the project.

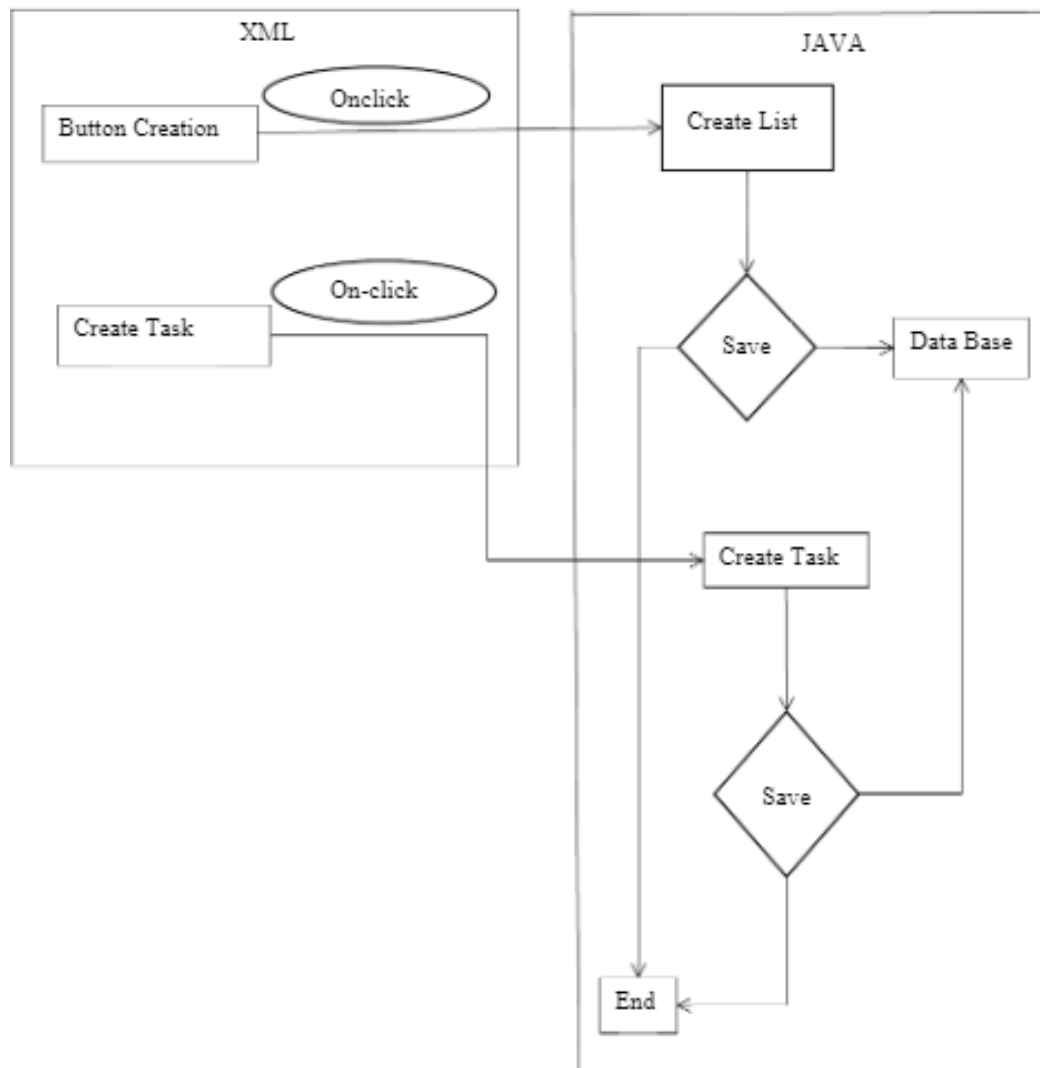
❖ **Android studio**

To create, design, test, debug and run the android project.

❖ **Mouse**

To navigate through the emulator..

3.DESIGN

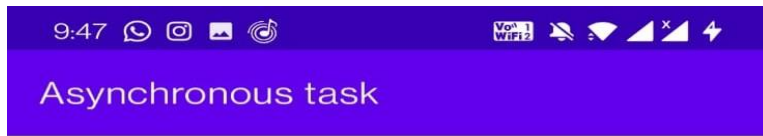


Flow chart for Representation of Xml to Java.

4.TESTING

No	Functions with parameters under test	Expected result	Actual result	Comments
1.	When the run is executed. The emulator should open with a certain application	Emulator should pop-up on screen with in a min.	The emulator appeared.	PASS
2.	When we click on + icon	Text box should appear.	Text box appeared.	PASS
3.	The written message.	Appear in list with TASKCOMPLETED button	List with button appeared.	PASS
4.	When we click on button.	The message should disappear(deleted)	Message disappeared(deleted).	PASS

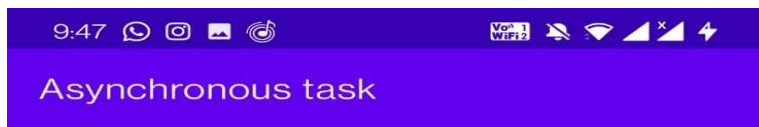
5. SNAPSHOTS



Async Task

START

STOP



Async Task

START

STOP

Asynchronous Task !!!!

Async Task Started!!!!!!!

6. CODE OF THE PROJECT

XML-CODE:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="123dp"
        android:layout_marginBottom="630dp"
        android:text="Async Task"
        android:textSize="36sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
    <Button
        android:id="@+id/buttonstart"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="167dp"
        android:layout_marginBottom="441dp"
        android:text="Start" />
```

```
<Button
    android:id="@+id/buttonstop"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="169dp"
    android:layout_marginBottom="328dp"
    android:text="Stop" />

<TextView
    android:id="@+id/marqueeText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginStart="50dp"
    android:layout_marginTop="250dp"
    android:layout_marginEnd="117dp"
    android:layout_marginBottom="207dp"
    android:ellipsize="marquee"
    android:marqueeRepeatLimit="marquee_forever"
    android:scrollHorizontally="true"
    android:singleLine="true"
    android:text="Demonstration of Asynchronous Task !!!!!"
    android:textSize="20sp"
    android:textStyle="bold"
    android:visibility="invisible" />

</RelativeLayout>
```


JAVA-CODE:

```
import androidx.appcompat.app.AppCompatActivity;

import android.os.AsyncTask;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    TextView marqtxt;
    Button btnstart, btnstop;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        marqtxt = (TextView) findViewById(R.id.marqueeText);
        btnstart = (Button) findViewById(R.id.buttonstart);
        btnstop = (Button) findViewById(R.id.buttonstop);
        btnstart.setOnClickListener(new View.OnClickListener() {
            @Override public void onClick(View v) {
                ExampleAsyncTask task = new ExampleAsyncTask();
                task.execute();
            }
        });
        btnstop.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                marqtxt.setSelected(false);
                marqtxt.setVisibility(View.INVISIBLE);
            }
        });
    }
}
```

```
}  
  
private class ExampleAsyncTask extends AsyncTask<String, String,String>{  
    @Override  
    protected void onPreExecute() {  
        super.onPreExecute();  
        Toast.makeText(getBaseContext(),"Async Task  
Started!!!!!!!!!!",Toast.LENGTH_SHORT).show();  
    }  
    @Override  
    protected String doInBackground(String... strings) {  
        try {  
            Thread.sleep(250);  
        }  
        catch (InterruptedException e){  
            e.printStackTrace();  
        }  
        return null;  
    }  
    @Override  
    protected void onPostExecute(String s) {  
        super.onPostExecute(s);  
        marqtxt.setVisibility(View.VISIBLE);  
        marqtxt.setSelected(true);  
    }  
}  
}
```

7.CONCLUSION AND ENHANCEMENTS

The development of the project is not an easy process as it involves lot of challenges indifferent stages of software analysis, design, coding and testing.

Having understood the requirements properly and implementing the solutions as per expectation as brought to the closure of the project.

We have tried our best to make this project very realistic, so that the user does not face any trouble when switching over from any real-life android project to this highly useful one

8. REFERENCES

Reference Books and E-book

1. Mobile application development by jeff McWherter, scott Gowell

Websites for Reference:

- ❖ www.Github.com
- ❖ www.wikipedia.com
- ❖ www.Youtube.com

