VISVESVARAYA TECHNOLOGICAL UNIVERSITY
JNANASANGAMA, BELAGAVI - 590018

Mobile Application Development Laboratory Mini Project Report

On

STOPWATCH

Submitted in partial fulfillment for the award of degree of

Bachelor of Engineering
in
Computer Science and Engineering

Submitted by

NAGAJYOTHI.M.S 1BG18CS066
NAMYA SHARMA 1BG18CS071
MARUTHI.K 1BG18CS064

Vidyayāmruthamashnuthe

B.N.M Institute of Technology
Approved by AICTE, Affiliated to VTU, Accredited as grade A Institution by NAAC.
All UG branches – CSE, ECE, EEE, ISE &Mech.E accredited by NBA for academic years 2018-19 to 2020-21 & valid upto 30.06.2021
Post box no. 7087, 27th cross, 12th Main, Banashankari 2nd Stage, Bengaluru- 560070,
INDIA Ph: 91-80- 26711780/81/82 Email: principal@bnmit.in, www. bnmit.org

Department of Computer Science and Engineering

2020-21

B.N.M Institute of Technology

Approved by AICTE, Affiliated to VTU, Accredited as grade A Institution by NAAC.
All UG branches – CSE, ECE, EEE, ISE &Mech.E accredited by NBA for academic years 2018-19 to 2020-21 & valid upto 30.06.2021
Post box no. 7087, 27th cross, 12th Main, Banashankari 2nd Stage, Bengaluru- 560070, INDIA
Ph: 91-80- 26711780/81/82 Email: principal@bnmit.in, www. bnmit.org

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Vidyaya Amrutham Ashnuthe

CERTIFICATE

Certified that the Mini Project entitled **Stopwatch** carried out by Ms. **Nagajyothi.M.S,** Ms. **Namya Sharma,** Mr. **Maruthi.k** USN **1BG18CS066, 1BG18CS071, 1BG18CS064,** bona-fide students of VI Semester B.E., **B.N.M Institute of Technology** in partial fulfilment for the Bachelor of Engineering in **COMPUTER SCIENCE AND ENGINEERING** of the **Visvesvaraya Technological University**, Belagavi during the year 2020-21. It is certified that all corrections/ suggestions indicated for internal Assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements in respect of Mobile Application Development Laboratory Mini Project prescribed for the said degree.

Mr. Karthik S
Assistant Professor
Department of CSE
BNMIT, Bengaluru

Mrs. Karthika N J
Assistant Professor
Department of CSE
BNMIT, Bengaluru

Dr. Sahana D. Gowda
Professor and HOD
Department of CSE
BNMIT, Bengaluru

Name & Signature

Examiner 1:

Examiner 2:

ACKNOWLEDGEMENT

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of my project.

I would like to thank **Shri. Narayan Rao R Maanay**, Secretary, BNMEI, Bengaluru for providing the excellent environment and infrastructure in the college.

I would like to sincerely thank **Prof. T J Rama Murthy**, Director, BNMIT, Bengaluru for having extended his constant support and encouragement during the course of this project.

I would like to sincerely thank **Dr. S Y Kulkarni** ,Additional Director, BNMIT, Bengaluru for having extended his constant support and encouragement during the course of this project.

I would like to express my gratitude to **Prof. Eishwar N Maanay**, Dean, BNMIT, Bengaluru for his relentless support and encouragement.

I would like to thank **Dr. Krishnamurthy G N**, Principal, BNMIT, Bengaluru for his constant encouragement.

I would like to thank, **Dr. Sahana D. Gowda**, Professor & Head of the Department of Computer Science and Engineering for the encouragement and motivation she provides.

I would also like to thank **Mr.Karthik S & Mrs.Karthika N J,** Assistant Professors, Department of Computer Science and Engineering for providing me with their valuable insight and guidance wherever required throughout the course of the project and its successful completion.

Nagajyothi M.S 1BG18CS066

Namya Sharma 1BG18CS071

Maruthi .k 1BG18CS064

ABSTRACT

The stopwatch is basically a timepiece that is designed particularly to measure the time elapsed between the activation and deactivation of the stopwatch. The users can start the Stopwatch and stop it anytime. It was originally invented by Samuel Watson. There is also a button to restart the stopwatch that will set the stopwatch to starting. So our task is to make an application for Stopwatch which can be used whenever required.

A stopwatch is a simple app that is used to record or measure time. It has mainly 5 main functions: Start, Stop, Pause ,Reset and lap. The stopwatch function is also present as an additional function of many electronic devices such as wristwatches, cell phones, portable music players, and computers.

In manual timing, the clock is started and stopped by a person pressing a button. In fully automatic time, both starting and stopping are triggered automatically, by sensors. The timing functions are traditionally controlled by two buttons on the case. Pressing the top button starts the timer running, and pressing the button a second time stops it, leaving the elapsed time displayed. A press of the second button then resets the stopwatch to zero. The second button is also used to record *split times* or *lap times*. When the split time button is pressed while the watch is running it allows the elapsed time to that point to be read, but the watch mechanism continues running to record total elapsed time. Pressing the split button a second time allows the watch to resume display of total time.

Table of Contents

List of Figures

Chapter 1

INTRODUCTION

1.1 Overview

Mobile application development is the process to making software for smart phones and digital assistants, most commonly for Android and iOS. The software can be preinstalled on the device, downloaded from a mobile app store or accessed through a mobile web browser. The programming and markup languages used for this kind of software development include Java, Swift, C# and HTML5.

Mobile app development is rapidly growing. From retail, telecommunications and e-commerce to insurance, healthcare and government, organizations across industries must meet user expectations for real-time, convenient ways to conduct transactions and access information. Today, mobile devices and the mobile applications that unlock their value are the most popular way for people and businesses to connect to the internet. To stay relevant, responsive and successful, organizations need to develop the mobile applications that their customers, partners and employees demand.Yet mobile application development might seem daunting. Once you’ve selected the OS platform or platforms, you need to overcome the limitations of mobile devices and usher your app all the way past the potential hurdles of distribution. Fortunately, by following a few basic guidelines and best practices, you can streamline your application development journey. We can start explaining mobile development, which is not about building phone apps, though it is a huge part of it.

Actually, It’s doing any reasonably development for any kind of mobile devices such as developing apps for phones, tablets, smart watches, and every form of wearable devices that run any kind of mobile operating system.

Mobile development presents a reasonably distinctive chance for a one-person development team to build an actual, usable, significant app end-to-end during a comparatively short period. However, mobile apps development represents more than just a chance for the solo-developer to create their own project as it is arguably the longer term of development, as mobile devices are getting larger and bigger parts of our lives.

1.2 Problem Statement

The aim of this application is to show a basic implementation of a stopwatch. This application will allow the users to Start the stopwatch time from the beginning Or from the pause state, and one can stop the stopwatch time and Reset the time to zero and also enable the users to Record the lapse of the Stopwatch time. User can access this application anywhere and anytime using his or her Smartphone.

1.3 Mobile Application Development Need & Importance

App development is essential today for an online business. If you want to boost your sales using technology, app development is truly recommended for everyone. Here listed 4 reasons that you need to know how much it notable is.

1. Accessibility from Variety of Platforms

You may gain access to nearly every type of online platforms by developing your apps. Develop apps will help reach into marketplaces as far reaching areas via Google Play, Blackberry, as well as Apple App Store, Symbian, and other internet marketplaces and through social media web sites just like Facebook or Myspace, Twitter, among others. Besides having the ability to mail data to clients, app growth possesses additional exclusive capabilities, which include coupon codes, evaluation of functions, and also force announcements. Your visitors within the quickest feasible valuable time, and also obtain an immediate answer, which assists you examine the advertising tool.

2. Targeted Audience

It’s simple to get obsessed with app ideas. I hear fantastic ideas daily, but generally they’re strategies that originate around a work or function, instead of a particular target audience. The audience is very close to an afterthought, merely crucial while making the advertising program. The a lot more applications I style and also release, the a lot more I’m certain it’s easier to begin with a particular audience and also produce suggestions depending on the requirements of this target audience.

1. Efficiency and Effectiveness

Effectiveness and efficiency: Business management of internet based use is conducted with effectiveness and also performance. Can decrease their functional expenses whilst at the same time improving the effectiveness of their procedures. Can decrease their producing expenditures, which make your company environment friendly. Threatened functions offer customers the ability of handling their companies and never have to get the cost of employing an additional worker for carrying this out train.

Internet based functions will not need customers to install these types of on their hard disk drives that lead to the decrease of memory space. Moreover, any specific up-to-date variations can be found immediately for the customers. The dependable and also efficiently created internet based functions are created to make sure that they are suitable for all of the different internet browsers, working devices, and also equipment.

2. To engage with customers everywhere

You can involve with your all types of customers everywhere by developing your apps system. Modern and updated apps perform multiple task in business as well as other site too. You should develop your apps system that is really user friendly and easy to access so that customer can contact anytime from anywhere they are.

1.4 Android Studio

In recent times, Android became the world's most popular operating system for various reasons. As an Android programmer, I want to share what the Android Studio is? Android Studio is an IDE for Google Android Development launched on 16th May 2013, during Google's I/O 2013 event. Android Studio contains all the Android tools to design, test, debug, and profile your application. The Android Studio uses Gradle to manage your project, a Build Automation Tool.

For developing your first app, you need to download Android Studio for your preferred platform (Windows®, Mac OS X, or Linux) from the Android developers site. Android Studio can develop and test your application on either a real device or an emulator.

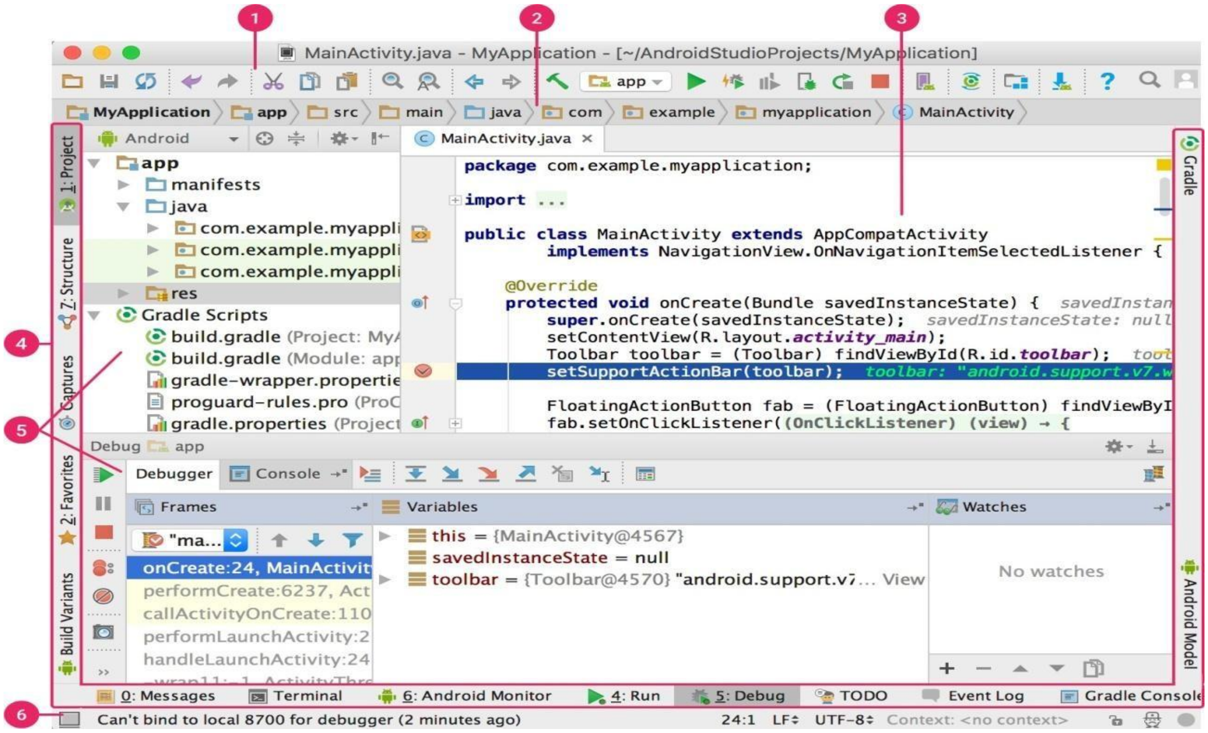
Android Studio can be installed on Windows operating systems, OSX and Linux and is recommended by Google itself that the hardware must have at least 4 GB of memory and 1GB of free hard disk space, but we recommend that you have more memory because it was noted that Android Studio is still a little slow. You must have Java installed on the machine via the JDK (Java Development Kit), not the JRE, as it is usually installed, once to develop on Android is necessary for all Java development classes to be present on the machine.

Android Studio has many exciting features that can help you to develop your Android application like:

- Powerful code editor with smart editing and code re-factoring.
- Emulator to show your code output in various resolutions, including Nexus 4, Nexus 7, Nexus 10, and many other android phones.
- Gradle based build support.
- Maven Support.
- Template-based wizards.
- Dracula Theme Environment to enjoy your coding experience.

You can experience all the awesome features by using Android Studio in-hand.

The user interface



```
package com.example.myapplication;

import ...

public class MainActivity extends AppCompatActivity
        implements NavigationView.OnNavigationItemSelectedListener {

    @Override
    protected void onCreate(Bundle savedInstanceState) {  savedInstan
        super.onCreate(savedInstanceState);  savedInstanceState: null
        setContentView(R.layout.activity_main);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);  tool
        setSupportActionBar(toolbar);  toolbar: "android.support.v7.w

        FloatingActionButton fab = (FloatingActionButton) findViewByI
        fab.setOnClickListener((OnClickListener) (view) → {
```

```
onCreate:24, MainActivit
performCreate:6237, Act
callActivityOnCreate:110
performLaunchActivity:2
handleLaunchActivity:24
```

```
this = {MainActivity@4567}
savedInstanceState = null
toolbar = {Toolbar@4570} "android.support.v7... View
```

```
Can't bind to local 8700 for debugger (2 minutes ago)    24:1  LF÷  UTF-8÷  Context: <no context>
```

Figure 1.4.1 The Android Studio main window.

- **The toolbar** lets you carry out a wide range of actions, including running your app and launching Android tools.
- **The navigation bar** helps you navigate through your project and open files for editing. It provides a more compact view of the structure visible in the Project window.
- **The editor window** is where you create and modify code. Depending on the current file type, the editor can change.
- **The tool window bar** runs around the outside of the IDE window and contains the buttons that allow you to expand or collapse individual tool windows.
- **The tool windows** give you access to specific tasks like project management, search, version control, and more.
- **The status bar** displays the status of your project and the IDE itself, as well as any warnings or messages.

Chapter 2

SYSTEM REQUIREMENTS

2.1 Software Requirements

Software requirements deal with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application.

The following are the software requirements for the application:

- Operating System: Windows 10
- Development Environment: Android Studio 4.2
- API: Java Development Kit (JDK) 7
- Core Language: Java, XML for Front-end.

2.2 Hardware Requirements

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware.

- CPU: Intel processor with support for Intel VT-x (Intel 64)
- Cores: Dual-Core (Quad-Core recommended)
- RAM: minimum 4GB (>4GB recommended)
- Secondary Storage: 250GB hard disk space plus at least 1GB for Android SDK, Emulator System images, and cashes.
- Screen resolution: 1366 x 800.

Chapter 3

SYSTEM DESIGN

3.1 XML Design

The proposed system we will be building an Android application where timing functions are traditionally controlled by two buttons on the case. Pressing the top button starts the timer running, and pressing the button a second time stops it, leaving the elapsed time displayed. A press of the second button then resets the stopwatch to zero. The second button is also used to record lap times. When the split time button is pressed while the watch is running it allows the elapsed time to that point to be read, but the watch mechanism continues running to record total elapsed time. Pressing the split button a second time allows the watch to resume display of total time.

3.1.1 State Flow Diagram (SFD):

Most existing digital devices use sequential circuits that create or track sequences of events. Many of these must take certain actions for particular inputs, and different actions for other inputs. This implies the circuit maintains an internal state, so that it can respond to inputs contextually. Digital circuits that maintain state, and create or track sequences of inputs, are known as state machines. This project introduces state machines, and present a design project (a stopwatch) that requires a state machine.

In operation, the binary number stored in the state register is updated at every clock edge, and so a sequence of constantly updated numbers (or states) cycles through the state register. The stored numbers can be used to create a meaningful sequence of output signals. This project illustrates the state machine design process with two different state machines.

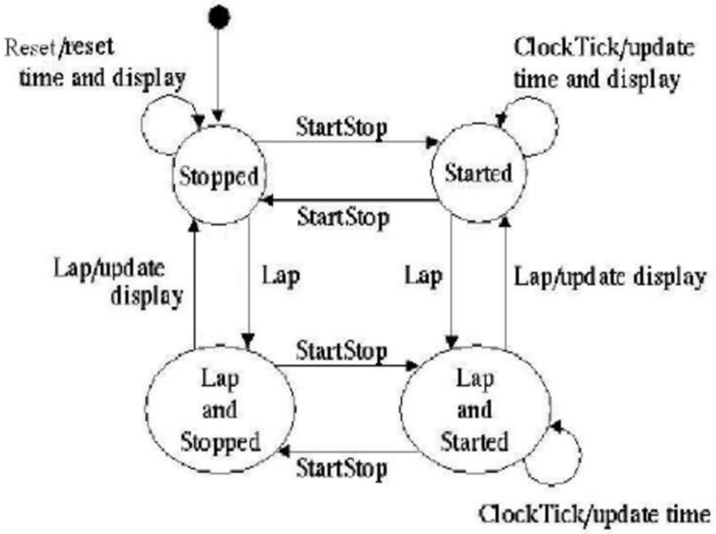


Figure 3.1 State diagram for three button stopwatch

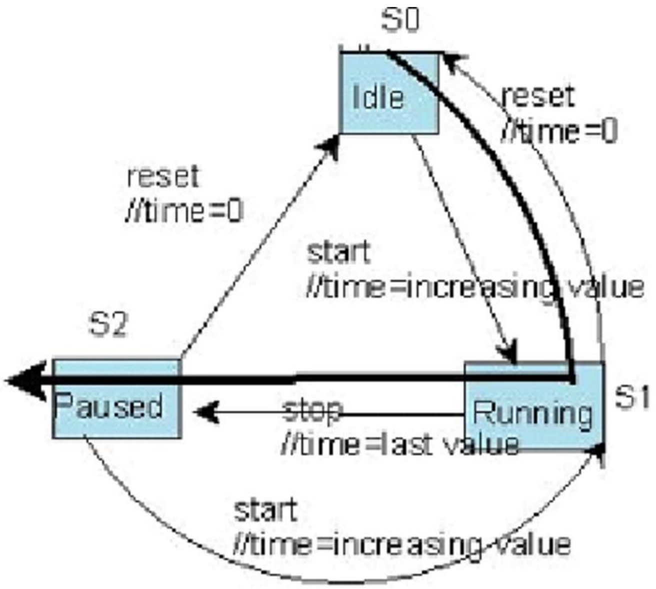


Figure 3.2 State transition diagram for Stopwatch behavior

3.2 XML Code

Activity_stopwatch.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="@drawable/app_bg"
    tools:context=".StopwatchActivity">

    <TextView
        android:id="@+id/time_view"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_marginStart="180dp"
        android:layout_marginLeft="180dp"
        android:layout_marginTop="96dp"
        android:layout_marginEnd="180dp"
        android:layout_marginRight="180dp"
        android:layout_marginBottom="568dp"
        android:background="@drawable/borders"
        android:textAppearance="@android:style/TextAppearance.Large"
        android:textColor="@color/white"
        android:textColorHint="@color/white"
        android:textSize="50dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.49"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/start_button"
        android:layout_width="80dp"
        android:layout_height="73dp"
        android:layout_gravity="center_horizontal"
        android:layout_marginStart="300dp"
        android:layout_marginLeft="300dp"
        android:layout_marginTop="206dp"
        android:layout_marginEnd="30dp"
        android:layout_marginRight="30dp"
        android:layout_marginBottom="452dp"
        android:background="@drawable/play_icon"
        android:onClick="onClickStart"
        app:backgroundTint="#0000838F"
        app:backgroundTintMode="add"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/stop_button"
        android:layout_width="81dp"
        android:layout_height="75dp"
        android:layout_gravity="center_horizontal"
        android:layout_marginStart="30dp"
        android:layout_marginLeft="30dp"
        android:layout_marginTop="206dp"
        android:layout_marginEnd="300dp"
        android:layout_marginRight="300dp"
        android:layout_marginBottom="452dp"
        android:background="@drawable/pause_icon"
        android:onClick="onClickStop"
        app:backgroundTint="#0000838F"
        app:backgroundTintMode="add"
        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" />

    <Button
        android:id="@+id/reset_button"
        android:layout_width="78dp"
        android:layout_height="74dp"
        android:layout_centerHorizontal="true"
        android:layout_gravity="center_horizontal"
        android:layout_marginStart="10dp"
        android:layout_marginLeft="10dp"
        android:layout_marginTop="206dp"
        android:layout_marginEnd="5dp"
        android:layout_marginRight="5dp"
        android:layout_marginBottom="452dp"
        android:background="@drawable/reset_icon"
        android:onClick="onClickReset"
        app:backgroundTint="#0000838F"
        app:backgroundTintMode="add"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toStartOf="@+id/lap_button"
        app:layout_constraintStart_toEndOf="@+id/stop_button"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/lap_button"
        android:layout_width="81dp"
        android:layout_height="73dp"
        android:layout_marginStart="5dp"
        android:layout_marginLeft="5dp”
        android:layout_marginTop="206dp"
        android:layout_marginEnd="10dp"
        android:layout_marginRight="10dp"
        android:layout_marginBottom="452dp"
        android:background="@drawable/lap_icon"
        android:onClick="onClickLaps"
        app:backgroundTint="#0000838F"
        app:backgroundTintMode="add"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toStartOf="@+id/start_button"
        app:layout_constraintStart_toEndOf="@+id/reset_button"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/timelapse"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="180dp"
        android:layout_marginLeft="180dp"
        android:layout_marginTop="612dp"
        android:layout_marginEnd="180dp"
        android:layout_marginRight="180dp"
        android:layout_marginBottom="100dp"
        android:textColor="@color/white"
        android:textColorHint="@color/white"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.49"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

Borders.xml

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:shape="rectangle">
    <stroke
        android:width="2dp"
        android:color="#ffffff"
        />
</shape>
```

Activity_login.xml

```
<?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"
    android:background="@drawable/app_bg"
    android:padding="40dp"
    tools:context=".LoginActivity">

    <EditText
        android:id="@+id/username1"
        android:layout_width="match_parent"
        android:layout_height="44dp"
        android:layout_marginTop="80dp"
        android:background="@drawable/input_bg"
        android:drawableLeft="@drawable/ic_user"
        android:drawablePadding="10dp"
        android:hint="Username"
        android:paddingLeft="20dp"
        android:textColor="@color/white"
        android:textColorHint="@color/white" />

    <EditText
        android:id="@+id/password1"
        android:layout_width="match_parent"
        android:layout_height="44dp"
        android:layout_below="@+id/username1"
        android:layout_marginTop="50dp"
        android:background="@drawable/input_bg"
        android:drawableLeft="@drawable/ic_baseline_security_24"
        android:drawablePadding="10dp"
        android:hint="Password"
        android:paddingLeft="20dp"
        android:textColor="@color/white"
        android:textColorHint="@color/white" />

    <Button
        android:id="@+id/bsignin1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/password1"
        android:layout_marginLeft="20dp"
        android:layout_marginTop="50dp"
        android:layout_marginRight="20dp"
        android:background="@drawable/button_bg"
        android:drawableRight="@drawable/ic_baseline_check_24"
        android:text="Sign in"
        android:textColor="#0E1159"
        android:textSize="16sp"
        android:textStyle="bold"
        app:backgroundTint="@color/white" />


    <Button
        android:id="@+id/bDelete"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/bsignin1"
        android:layout_marginLeft="20dp"
```

```
        android:layout_marginTop="50dp"
        android:layout_marginRight="20dp"
        android:background="@drawable/button_bg"
        android:drawableRight="@drawable/ic_clear"
        android:text="Delete account"
        android:textColor="#0E1159"
        android:textSize="16sp"
        android:textStyle="bold"
        app:backgroundTint="@color/white" />

    <Button
        android:id="@+id/bSignup"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/bDelete"
        android:layout_marginLeft="20dp"
        android:layout_marginTop="50dp"
        android:layout_marginRight="20dp"
        android:background="@drawable/button_bg"
        android:drawableRight="@drawable/ic_login"
        android:text="Sign up"
        android:textColor="#0E1159"
        android:textSize="16sp"
        android:textStyle="bold"
        app:backgroundTint="@color/white" />

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="21dp"
        android:drawableLeft="@drawable/ic_baseline_access_alarm_24"
        android:drawablePadding="20dp"
        android:text="STOPWATCH"
        android:textColor="@color/white"
        android:textSize="24sp" />
```

Activity_main.xml

```
<?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"
  android:background="@drawable/app_bg"
  android:padding="40dp"
  tools:context=".MainActivity">

  <EditText
      android:id="@+id/username"
      android:layout_width="389dp"
      android:layout_height="44dp"
      android:layout_marginTop="80dp"
      android:background="@drawable/input_bg"
      android:drawableLeft="@drawable/ic_user"
      android:drawablePadding="10dp"
      android:hint="Username"
      android:paddingLeft="20dp"
      android:textColor="@color/white"
      android:textColorHint="@color/white" />

  <EditText
      android:id="@+id/password"
      android:layout_width="389dp"
      android:layout_height="44dp"
      android:layout_below="@+id/username"
      android:layout_marginTop="50dp"
      android:background="@drawable/input_bg"
      android:drawableLeft="@drawable/ic_baseline_security_24"
      android:drawablePadding="10dp"
      android:hint="Password"
      android:paddingLeft="20dp"
      android:textColor="@color/white"
      android:textColorHint="@color/white" />

  <EditText
      android:id="@+id/repassword"
      android:layout_width="389dp"
      android:layout_height="44dp"
      android:layout_below="@+id/password"
      android:layout_centerHorizontal="true"
      android:layout_marginTop="48dp"
      android:background="@drawable/input_bg"
android:drawableLeft="@drawable/ic_baseline_security_24
```

```
  android:drawablePadding="10dp"
  android:hint="Confirm password"
  android:paddingLeft="20dp"
  android:textColor="@color/white"
  android:textColorHint="@color/white" />

<Button
    android:id="@+id/bsignup"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/repassword"
    android:layout_marginTop="50dp"
    android:background="@drawable/button_bg"
    android:drawableRight="@drawable/ic_baseline_check_24"
    android:text="Register"
    android:textColor="#0E1159"
    android:textColorHint="#0E1159"
    android:textSize="16sp"
    android:textStyle="bold"
    app:backgroundTint="@color/white" />

<Button
    android:id="@+id/bsignin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/bsignup"
    android:layout_marginTop="50dp"
    android:background="@drawable/button_bg"
    android:drawableRight="@drawable/ic_login"
    android:text="Log in"
    android:textColor="#0E1159"
    android:textColorHint="#0E1159"
    android:textSize="16sp"
    android:textStyle="bold"
    app:backgroundTint="@color/white" />

<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="21dp"
```

```
        android:drawableLeft="@drawable/ic_baseline_access_alarm_24"
        android:drawablePadding="20dp"
        android:text="STOPWATCH"
        android:textColor="@color/white"
        android:textSize="24sp" />
```

Button_bg.xml

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
    <solid android:color="@color/white"/>
    <corners android:radius="70dp"/>
</shape>
```

Input_bg.xml

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
<stroke android:color="@color/white" android:width="2dp"/>
    <corners android:radius="70dp"/>
</shape>
```

Strings.xml

```
<resources>
    <string name="app_name">Miniproject_stopwatch</string>
    <string name="start">start</string>
    <string name="stop">stop</string>
    <string name="reset">reset</string>
</resources>
```

Colours.xml

```
<?xml version = "1.0" encoding = "utf-8"?>
<resources>
    <color name = "colorPrimary">#1989C8</color>
    <color name = "colorPrimaryDark">#1989C8</color>
    <color name = "colorAccent">#FF0D3D</color>
    <color name = "textColor">#0B1A37</color>
    <color name = "white">#ffffff</color>
</resources>
```

Chapter 4

IMPLEMENTATION

4.1 Description

- **Void onCreate (Bundle)**

  The **entire lifetime** of an activity happens between the first call to onCreate(Bundle) and onDestroy(). An activity will do all setup of "global" state in onCreate(), and release all remaining resources in onDestroy().

- **public class Intent**

  An Intent provides a facility for performing late runtime binding between the code in different applications. Its most significant use is in the launching of activities, where it can be thought of as the glue between activities.

- **public void onClickStart**

  Creates and starts a new activity.

- **public void onClickStop**

  This stops/closes an activity.

- **public void onClickReset**

  This will restart the activity.

- **public void onClickLaps**

  This increases lap count when a function is called.

- **public void onPause**

  The activity is about to go into the background and has stopped interacting with the user. This can happen when another activity is launched in front of the current activity.

- **public void RunTimer**

  This will run the code every second to check whether the stopwatch is running and if it is,it will increment the number of seconds and display the number of seconds in text view.

4.1 Java Code

StopwatchActivity.java

```
package com.example.miniproject_stopwatch;

import android.media.MediaPlayer;
import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import java.util.Locale;

public class StopwatchActivity extends AppCompatActivity {

    Button play, pause, restart,laps;
    TextView timeLapse;
    private int sec = 0;
    private boolean is_running;
    private boolean was_running;
    int lapCount=0;
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_stopwatch);
        play= findViewById(R.id.start_button);
        pause= findViewById(R.id.stop_button);
        restart= findViewById(R.id.reset_button);
        laps= findViewById(R.id.lap_button);
        timeLapse= findViewById(R.id.timelapse);
        if (savedInstanceState != null) {
            sec = savedInstanceState.getInt("seconds");
            is_running = savedInstanceState.getBoolean("running");
            was_running = savedInstanceState .getBoolean("wasRunning");
        }
        running_Timer();
```

```
}


 @Override

public void onSaveInstanceState(
        Bundle savedInstanceState) {
    super.onSaveInstanceState(savedInstanceState);
    savedInstanceState.putInt("seconds", sec);
    savedInstanceState.putBoolean("running", is_running);
    savedInstanceState.putBoolean("wasRunning", was_running);
}
@Override
protected void onPause()
{
    super.onPause();
    was_running = is_running;
    is_running = false;
}
@Override
protected void onResume()
{
    super.onResume();
    if (was_running) {
        is_running = true;
    }
}
public void onClickStart(View view)
{   final MediaPlayer mp= MediaPlayer.create(this, R.raw.start_sound);
    mp.start();
    is_running = true;
}
public void onClickStop(View view)
{
    final MediaPlayer mp= MediaPlayer.create(this, R.raw.stop_sound);
    mp.start();
    is_running = false;
}
public void onClickReset(View view)
{   final MediaPlayer mp= MediaPlayer.create(this, R.raw.reset_sound);
    mp.start();
    is_running = false;
    timeLapse.setText("");
    lapCount=0;
    sec = 0;
    }

    public void onClickLaps(View view)
    {
        final MediaPlayer mp= MediaPlayer.create(this, R.raw.reset_sound);
        mp.start();
        is_running = true;
        timeLapseFun();
    }


    private void running_Timer()
    {

        final TextView t_View = findViewById(R.id.time_view);
        final Handler handle = new Handler();



        handle.post(new Runnable() {
            @Override

            public void run()
            {
                int hrs = sec / 3600;
                int mins = (sec % 3600) / 60;
                int secs = sec % 60;

                String time_t = String .format(Locale.getDefault(), "   %02d:%02d:%02d  ",
                     hrs,mins, secs);

                t_View.setText(time_t);


                if (is_running) {
                    sec++;
                }

                handle.postDelayed(this, 1000);
            }
        });
    }
    void timeLapseFun() {

        // increase lap count when function is called
        lapCount++;
        int hrs = sec / 3600;
```

```
            int mins = (sec % 3600) / 60;
            int secs = sec % 60;
            String laptext = String.format(Locale.getDefault(), "%02d:%02d:%02d", hrs, mins, secs);

                    laptext = "Lap " + lapCount + " ------------->        " + laptext + " \n "
                      + timeLapse.getText();

            //showing simple toast message to user
            Toast.makeText(StopwatchActivity.this, "Lap " + lapCount, Toast.LENGTH_SHORT).show();

            // showing the lap text
            timeLapse.setText(laptext);
        }
    }
```

LoginActivity.java

```
package com.example.miniproject_stopwatch;

import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import com.google.android.material.textfield.TextInputEditText;

public class LoginActivity extends AppCompatActivity {
    EditText username, password;
    Button btnlogin, bDelete,bSignup;
    DBHelper db;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
        username = findViewById(R.id.username1);
        password = findViewById(R.id.password1);
        btnlogin = findViewById(R.id.bsignin1);
        bDelete = findViewById(R.id.bDelete);
        bSignup = findViewById(R.id.bSignup);
        db = new DBHelper(this);
        btnlogin.setOnClickListener(v -> {
            String user = username.getText().toString();
            String pass = password.getText().toString();

            if (user.equals("") || pass.equals(""))
                Toast.makeText(LoginActivity.this, "Please enter your details",
Toast.LENGTH_SHORT).show();
            else {

      Boolean checkuserpass = db.checkUsernamePassword(user, pass);
                if (checkuserpass) {
                                              Toast.makeText(LoginActivity.this, "Signed in successfully!",
Toast.LENGTH_SHORT).show();
                    Intent intent = new Intent(getApplicationContext(), StopwatchActivity.class);
                    startActivity(intent);
                } else {
                    Toast.makeText(LoginActivity.this, "Incorrect username or password",
Toast.LENGTH_SHORT).show();
                }
            }
        });

        bDelete.setOnClickListener(v -> {
            String user = username.getText().toString();
            String pass = password.getText().toString();
            Boolean checkuserpass = db.checkUsernamePassword(user, pass);
            if (!checkuserpass) {
                Toast.makeText(LoginActivity.this, "Incorrect username or password",
Toast.LENGTH_SHORT).show();}
            else{
                    int deletedRows = db.deleteData(user, pass);
                    Toast.makeText(LoginActivity.this, "Data Deleted", Toast.LENGTH_LONG).show();
                }
        });
        bSignup.setOnClickListener(v -> {
            String user = username.getText().toString();
            String pass = password.getText().toString();
            Boolean checkuserpass = db.checkUsernamePassword(user, pass);
            if (!checkuserpass) {
                Toast.makeText(LoginActivity.this, "Redirecting to registration page",
Toast.LENGTH_SHORT).show();
                Intent intent = new Intent(getApplicationContext(), MainActivity.class);
                startActivity(intent);
}
```

```
            else{
                Toast.makeText(LoginActivity.this, "User already exists. Please sign in",
Toast.LENGTH_LONG).show();
            }
        });
    }
}
```

MainActivity.java

```
package com.example.miniproject_stopwatch;

import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    EditText username,password,repassword;
    Button signup,signin;
    DBHelper db;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        username= findViewById(R.id.username);
        password= findViewById(R.id.password);
        repassword= findViewById(R.id.repassword);
        signin= findViewById(R.id.bsignin);
        signup= findViewById(R.id.bsignup);
        db=new DBHelper(this);

        signup.setOnClickListener(v -> {
            String user=username.getText().toString();
            String pass=password.getText().toString();
            String repass=repassword.getText().toString();
            if(user.equals("")||pass.equals("")||repass.equals(""))
```

```
 Toast.makeText(MainActivity.this,"Please enter your details"
,Toast.LENGTH_SHORT).show();

            else
            {
                if(pass.equals(repass))
                {
                    Boolean checkuser=db.checkUsername(user);
                    if(!checkuser) {
                        Boolean insert=db.insertData(user,pass);
                        if(insert) {
                            Toast.makeText(MainActivity.this,"Registered successfully!"
,Toast.LENGTH_SHORT).show();
                            Intent intent=new Intent(getApplicationContext(),StopwatchActivity.class);
                            startActivity(intent);
                        }
                        else
                        {
                            Toast.makeText(MainActivity.this,"Registration failed!"
,Toast.LENGTH_SHORT).show();
                        }
                    }
                    else
                    {

    Toast.makeText(MainActivity.this,"User already exists. Please sign in"
,Toast.LENGTH_SHORT).show();
                    }
                }
                else
                {
                    Toast.makeText(MainActivity.this,"Passwords do not match. Please check and re-type it"
,Toast.LENGTH_SHORT).show();
                }
            }
        });

        signin.setOnClickListener(v -> {
            Intent intent=new Intent(getApplicationContext(),LoginActivity.class);
            startActivity(intent);
        });
    }
}
```

DBHelper.java

```
package com.example.miniproject_stopwatch;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class DBHelper extends SQLiteOpenHelper {

    public static final String DBNAME="Login.db";

    public DBHelper( Context context) {
        super(context, "Login.db", null, 1);
    }

    @Override
    public void onCreate(SQLiteDatabase MyDB) {
        MyDB.execSQL("create Table users(username TEXT primary key, password TEXT not null)");
    }

    @Override
    public void onUpgrade(SQLiteDatabase MyDB, int oldVersion, int newVersion) {
        MyDB.execSQL("drop table if exists users");
    }
    public boolean insertData(String username, String password)
    {
        SQLiteDatabase MyDB=this.getWritableDatabase();
        ContentValues contentValues=new ContentValues();
        contentValues.put("username",username);
        contentValues.put("password",password);
        long results=MyDB.insert("users", null, contentValues);
        return results != -1;
    }
    public boolean checkUsername(String username) {
        SQLiteDatabase MyDB=this.getWritableDatabase();
        Cursor cursor=MyDB.rawQuery("Select * from users where username=?", new String[]
{username});
```

```
        return cursor.getCount() > 0;
    }
    public boolean checkUsernamePassword(String username, String password) {
        SQLiteDatabase MyDB=this.getWritableDatabase();

 Cursor cursor=MyDB.rawQuery("Select * from users where username=? and password=?",
new String[] {username,password});

        return cursor.getCount() > 0;
    }
    public int deleteData (String username, String password) {
        SQLiteDatabase MyDB = this.getWritableDatabase();
        Cursor cursor=MyDB.rawQuery("delete from users where username=? and password=?",
new String[] {username,password});
        if(cursor.getCount()>0)
            return 1;
        else
            return 0;
    }

}
```

Chapter 5

RESULTS

- **Initial Page:**

  Fig5.1 is the output UI with the blue color background and Stopwatch highlighted using style.xml and color.xml resource file. There are buttons used to enter username, password and confirm password .After entering credentials it shows the registration and login button.

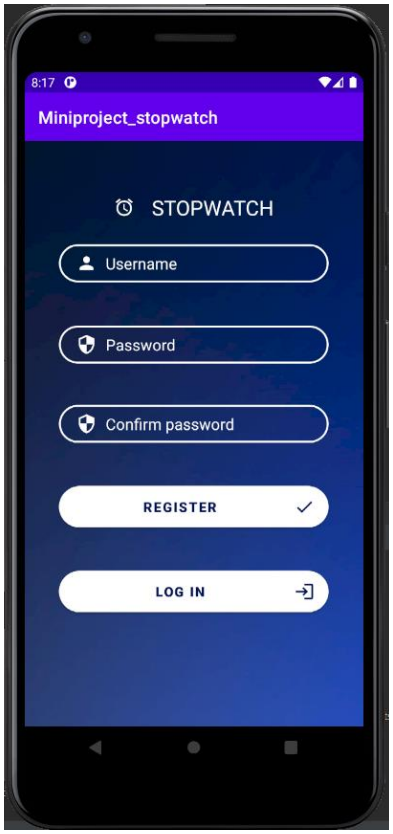
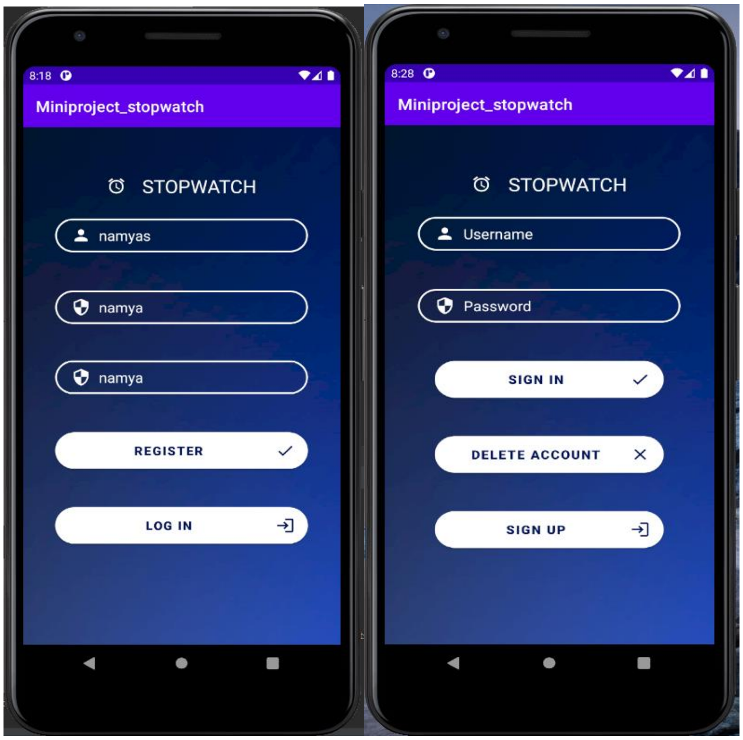


Figure 5.1 Home page of the application.

- **Sign in button after registeration:**

In shown in fig 5.2 After registeration we can click on sign in by entering the username and password.There is also a button for deleting the account and again creating a new one.



namyas

namya

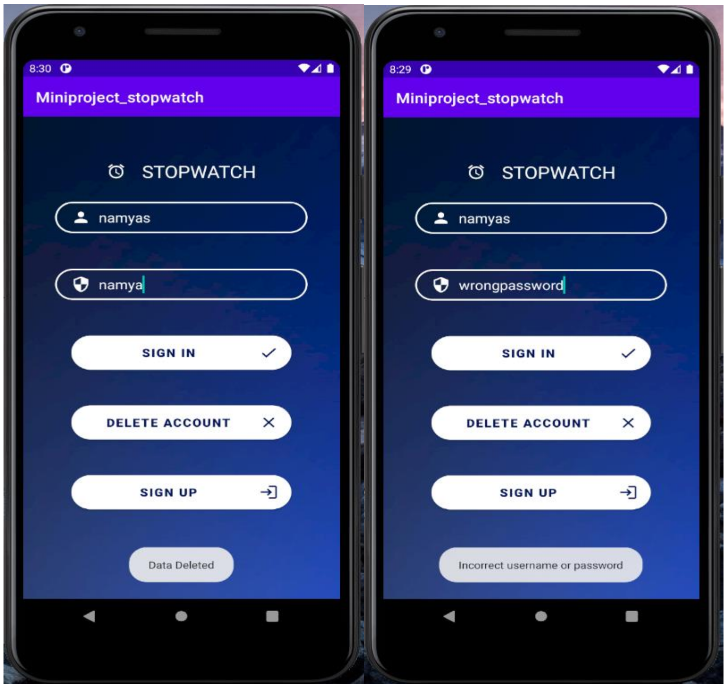
namya

Username

Password

Figure 5.2 Registeration and sign in Layout

- **Deleting account :** After deleting account if we try to sign in it will show wrong credentials.



namyas

namya

Data Deleted

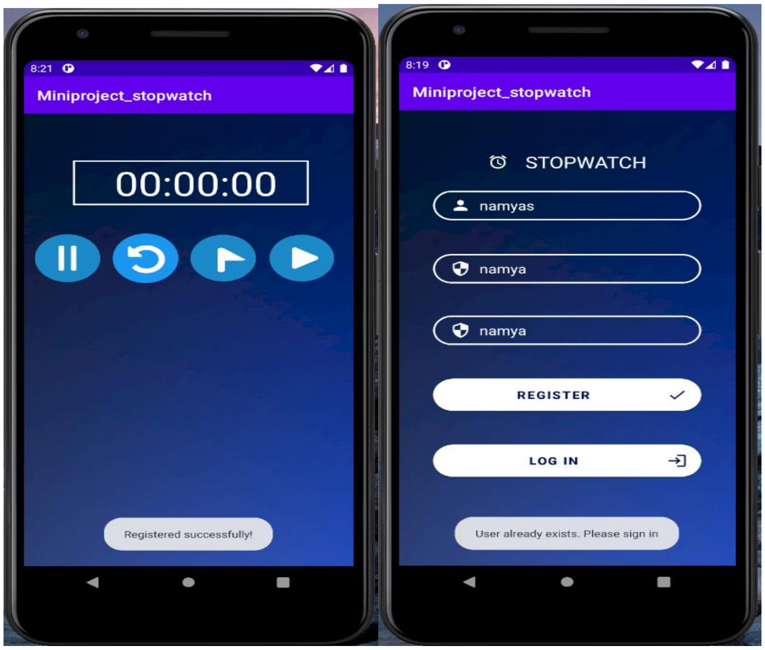
namyas

wrongpassword

Incorrect username or password

Figure 5.3 Deleting account

- **Successful registeration:** After deleting account we can register again .After registeration is successful, it shows that the user exists.



00:00:00

Registered successfully!

namyas

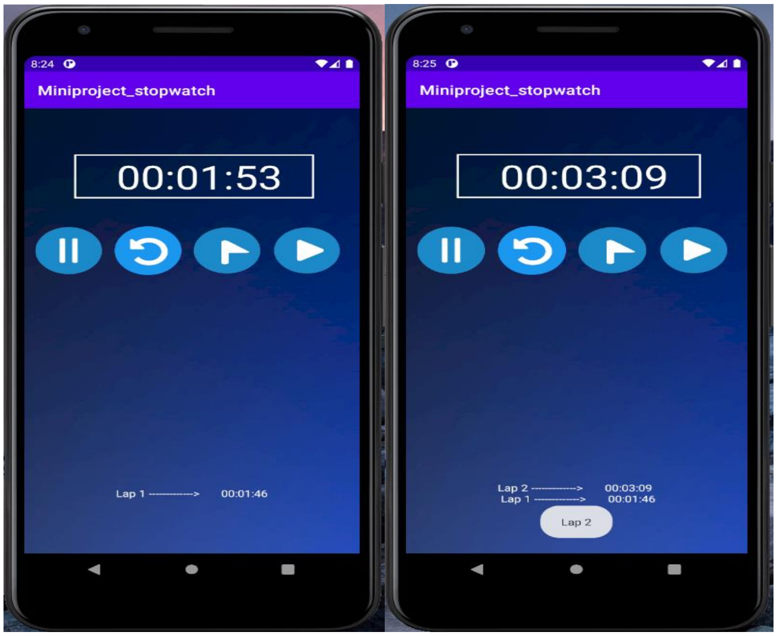
namya

namya

User already exists. Please sign in

Figure 5.4 Successful Registration after deleting account

- **Activity laps:** This shows the activity count for each lap. After 2 laps it shows a notification for the time count of each lap.



00:01:53

Lap 1 -----------> 00:01:46

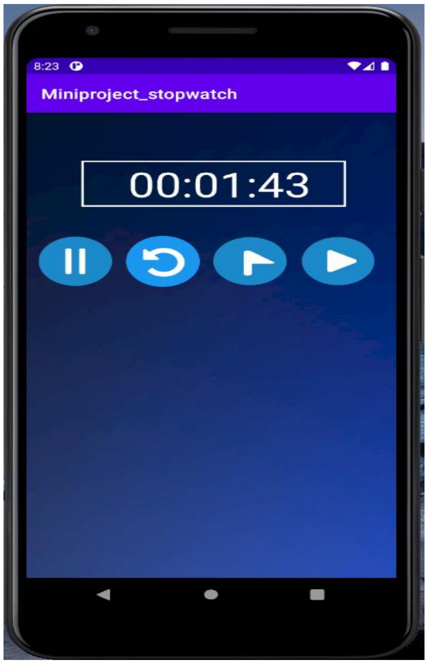
00:03:09

Lap 2 -----------> 00:03:09
Lap 1 -----------> 00:01:46

Lap 2

Figure 5.5 Activity lap counts along with notification

- **Activity stop:** After each lap the timer is stopped and reset. Example: here we have stopped the activity after first lap.



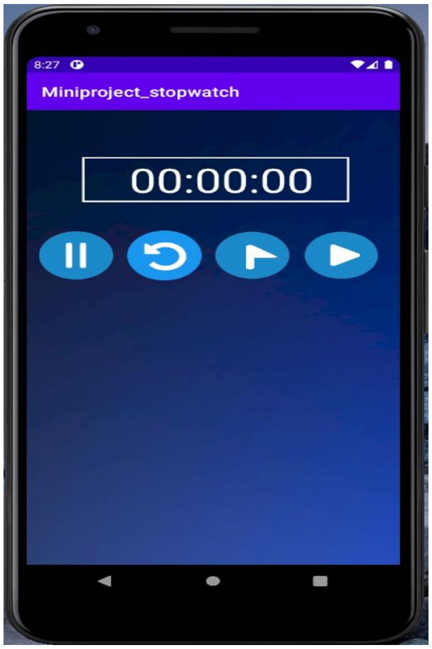
8:23

Miniproject_stopwatch

00:01:43

Figure 5.6 Stop activity

- **Activity Reset:** After the timer is stopped after each lap the stopwatch activity is reset to zero.



8:27

Miniproject_stopwatch

00:00:00

Figure 5.7 Reset activity

Chapter 6

CONCLUSION AND FUTURE ENHANCEMENTS

The working of a stopwatch demonstrates process of measuring time elapsed between activation and it's deactivation. After doing the project on stopwatch, we are able to learn and develop knowledge in designing an application by using the Android studio. We also learnt how to develop skills and knowledge in systematic design approach.

FUTURE SCOPE:

In the future, this application can be further enhanced by adding more functionality to show more operations on working of stopwatch like mentioning the specific event name during the elapsed time and also adding different sound for the specific elapsed time.

BIBLIOGRAPHY

[1] https://developer.android.com/guide/topics/ui/declaring-layout.

[2] Android Programming Tutorials by Mark L. Murphy.

[3] Akshata V.S.1, Rumana Pathan2, Poor nima Patil3 and Farjana Nadaf4 , B'Safe & B'Secure The Door to Safety Swings, Department of Computer Science Engineering, KLS's VDRIT, Haliyal, India, (IJCEM), ISSN: 2348 9510 Volume 1, Issue 7, October 2014.

[4] http://en.wikipedia.org/wiki/andorid

[5] https://www.geeksforgeeks.org/how-to-create-a-stopwatch-app-using-android-studio