

# **TAMILNADU EMPLOYMENT REGISTRATION**

## **MINI PROJECT REPORT**

Submitted in partial fulfilment of requirements for the award of the degree in

### **Master of Computer Application**

By

**J.JINOTH EDWIN – 20202102404211**

Under the guidance of

**Mrs.Pa.ANDAL M.Sc.,M.Phil.,SET.,NET.,**

Assistant Professor of Computer Science



DEPARTMENT OF COMPUTER SCIENCE (SF)

KAMARAJ COLLEGE,

(Affiliated to Manonmaniam Sundaranar University, Tirunelveli)

THOOTHUKUDI – 628 003

2020-2022

## **BONAFIDE CERTIFICATE**



**DEPARTMENT OF COMPUTER SCIENCE (SF),  
KAMARAJ COLLEGE,  
THOOTHUKUDI – 628 003.**

This is to certify the work entitled “**TAMILNADU EMPLOYEMENT REGISTRATION**” done by J.JINOTH EDWIN, Register Number : 20202102404226 in partial fulfilment of the requirement for the award of the degree in Master of Computer Application during the year 2021-2022.

**Internal Guide**

**Head of the Department**

Submitted for the viva-voce examination held at Department of Computer Science , KAMARAJ COLLEGE ,THOOTHUKUDI on 21.12.2021.

**EXAMINERS**

I.

II.

## TABLE OF CONTENT

Chapter No	Title	Page no.
1	INTRODUCTION	1
2	OVERVIEW	2
3	SYSTEM ANALYSIS 3.1 Existing System 3.2 Proposed System	3
4	SYSTEM SPECIFICATION 4.1 Hardware Specification 4.2 Software Specification 4.3 Software Description	5
5	SYSTEM DESIGN 5.1 Input Design 5.2 Output Design 5.3 Database Design	10
6	SYSTEM ARCHITECTURE 6.1 System Architecture 6.2 Flow Chart 6.3 UML Diagram	12
7	SYSTEM IMPLEMENTATION	17
8	SYSTEM TESTING	18
9	CONCLUSION	22
10	APPENDIX A. Sample source code B. Sample screen shot C. Bibliography	23

## ACKNOWLEDGEMENT

First I would like to thank my Almighty God for all the blessings he has endowed upon me. With profound sense of indebtedness, I thank my parents for their immense love and support to do this project.

I express my heartfelt thanks to our respected Principal **Dr.D.Nagarajan M.sc.,M.Phil.,Ph.D.,** for providing an opportunity to study M.C.A. course in this college.

My sincere thanks to our respected Director **Dr. A. Arunachala Rajan M.Com., M.Phil., Ph.D., M.B.A., M.A.[Eng.], B.Ed., PGDCA** for providing an opportunity to study M.C.A. course in this college.

I express my heartfelt thanks to our Head of the Department **Mr.s.Rajan MCA., M.Phil.,** for providing valuable suggestion for our project.

With immense pleasure, I register my deep sense of gratitude to my project guide **Mrs.Pa.Andal M.Sc.,M.Phil.,SET.,NET.,** for her excellent guidance which laid a path to achieve my goal. Also I extend it to the staff members and the lab assistants, for their cooperation and support in the completion of my project work. Finally , I owe to thanks to one and all who directly or indirectly helped me for the completion of my project.

## **ABSTRACT**

This is an app for the candidates to apply for the government jobs. The candidate can apply through this app for their future jobs. This app consists of login and registration page for verification purpose. Then the details of the candidates can be filled. The details will be uploaded to the database. after the registration success. The app will shows about the latest government exams based on their qualification. This android project help to find out government jobs, in this case, every person has a smartphone ("the whole world in your hands") every detail available on the internet, but they didn't use. Now a day nearly 1.5 lakh students pass out of college every year, Tamilnadu has Largest unemployment rate 7.6%, the Unemployment rate has increased by 1.4 % for the last Two years. This android application helps to increase knowledge about government jobs, it can increase the employment rate, decrease the Unemployment rate. This mobile application developed by using java in android studio, Sqlite was used to store user data for that mobile application. It can support all android lower-end devices, easy to use, more flexibility, and functionality. User register their detail, user details are stored in the database. Admin module has Four Categories UG, PG, SSLC and HSS. Admin can also view all user Registered data by these categories. User data is highly encrypted, Admin only views User data. Up to date jobs, jobs detail, and user details stored after a successful qualification in Sqlserver.

# **1.INTRODUCTION**

An employee registration application is a single form or a collection of forms where an individual seeks employment. The applicant must fill out all the required details such as personal information, skills, and qualification. In employee registration page the candidate gives the detail about their own, and in qualification page they need to give their correct qualification. Based on the details provided, the employer can offer employment. We can register our details with qualification and can refer the jobs based on the qualification. This app also can be used by admin. The admin can view the details of the candidates. This application is very useful for the employees because it doesn't need network while using this application. After the launch of the TNER application, candidates are not required to visit the TN Employment website for registration and card renewal. They will find all the information related to TN Employment Exchange online registration process, tn velaivaaiappu online renewal, login, documents required, eligibility, etc in app itself.

This application is user friendly and it is easily understood by every candidate. This application is developed by the latest version of the android and this app will support for only android OS. First install this app and then register with your E-mail id and password, and use your mail id as login name. Give the correct details of your own in upcoming modules.

Here my work is involved in HSS, Display sslc Pg jobs and Ugjobs. These 4 modules use database to retrieve the information of Candidates. Here we have 4 Qualification like SSLC, HSS, Pg, ug.

These modules will be displayed after view module. These modules can be accessed by admin only. It will display the details of the Candidates separately based on their each Qualification. Candidate of each qualification was retrieved from the database and displayed separately.

## **2.OVERVIEW**

An employee management system is a platform where all work-related as well as important personal details of an employee is stored and managed in a secure way. By using this system, you can manage admin activities in an easier and quicker way. This app is very much useful for the candidates. The jobs also available for freshers.

TNER is a Employment Registration App of Tamil Nadu State Government. TN Candidates ( Job Seekers), Who have Successfully Finished 10th Standard or 12th Standard or Any Under Graduation or Any Post Graduation Can Apply through this app for TN Govt Jobs....

→This App Interface are User Friendly.

→Everyone Can Easily use it.

→This app consist of login Page and registration page for verification purpose.

→Through the Registration Page, the Candidate Details are uploaded on the database for Future Use.

→After Successful Registration, it will Show the Latest Government Exams Which is Suitable for their Qualification.

### **3. SYSTEM ANALYSIS**

Analysis can be defined as breaking up of any whole so as to find out their nature, function etc.

It defines design as to make preliminary sketches of to sketch a pattern or outline for plan. To plan and carry out especially by artistic arrangement or in a skilful way. System analysis and design can be characterized as a set of techniques and processes, a community of interests, a culture and an intellectual orientation.

The Various tasks in the system analysis include the following:

- ❖ Understanding application.
- ❖ Planning.
- ❖ Scheduling.
- ❖ Developing candidate solution.
- ❖ Performing costs benefit analysis.
- ❖ Recommending alternative solutions.
- ❖ Selling of the system.
- ❖ Supervising, installing and maintain the system.



### **3.1 EXISTING SYSTEM**

- The user when enter their details at the same time network issues are may come.
- Lot of content in a same page so it is difficult to enter their details.
- We can only register in the website not able to search their related jobs.
- Only admin can view user's details about admin and user's information.
- The user can only register through website only.
- We need system to access this website.

### **3.2 PROPOSED SYSTEM**

To overcome the drawbacks of the existing system, the proposed system has been evolved,

- The network issues will not be a problem while entering and storing the data.
- Best user interface.
- This app contains many modules for entering data , the modules like register , login , candidate details and qualification.
- The user can easily register their data in a separate module.
- It is system independent.
- The details of jobs can be viewed in four different categories like UG, PG, SSLC and HSS and Admin can view all user Registered data by these categories.
- SQLite is the server used to store details of the candidate.

## 4.SYSTEM SPECIFICATION

### 4.1 Hardware Specification

Section	Requirement and Recommendations
CPU	x86_64 CPU architecture; 10th generation Intel Core or newer, or AMD CPU with support for a Windows Hypervisor
RAM	Minimum 4gb or Maximum 8 GB RAM
DISPLAY	1280 x 800 minimum screen resolution.
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).

## 4.2 Software Specification

Section	Requirement and Recommendations
Supported Operating Systems	64-bit Microsoft® Windows® 8/10
Software	Android Studio
Additional Software Requirement	JDK 1.8 or higher
Database	SQLite

## 4.3 Software Description

### ANDRIOD STUDIO

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ, IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems or as a subscription-based service in 2020. It is a replacement for the Eclipse 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 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. The first stable build was released in December 2014, starting from version 1.0.

On May 7, 2019, Kotlin replaced Java as Google's preferred language for Android app development. Java is still supported, as is C++.

The Android Emulator has additional requirements beyond the basic system requirements for Android Studio, which are described below:

- SDK Tools 26.1.1 or higher;
- 64-bit processor;
- Windows: CPU with UG (unrestricted guest) support;
- Intel Hardware Accelerated Execution Manager (**HAXM**) 6.2.1 or later (HAXM 7.2.0 or later recommended).

## ANDRIOD STUDIO FEATURES

A specific feature of the Android Studio is an absence of the possibility to switch auto save feature off.

The following features are provided in the current stable version:

- ❖ Gradle-based build support
- ❖ Android-specific refactoring and quick fixes
- ❖ Lint tools to catch performance, usability, version compatibility and other problems
- ❖ ProGuard integration and app-signing capabilities
- ❖ Template-based wizards to create common Android designs and components
- ❖ A rich layout\_editor that allows users to drag-and-drop UI components, option to preview\_layouts on multiple screen configurations
- ❖ Support for building Android\_Wear apps
- ❖ Built-in support for Google Cloud Platform, enabling integration with Firebase Cloud Messaging (Earlier 'Google Cloud Messaging') and Google App Engine
- ❖ Android Virtual Device (Emulator) to run and debug apps in the Android studio.

Android Studio supports all the same programming languages of IntelliJ (and CLion) e.g. Java, and more with extensions, such as Go; and Android Studio 3.0 or later supports Kotlin and "all Java 7 language features and a subset of Java 8 language features that vary by platform version." External projects backport some Java 9 features. 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. Once an app has been compiled with Android Studio, it can be published on the Google\_Play\_Store. The application has to be in line with the Google Play Store developer\_content\_policy.

## DB Browser

This program was developed originally by Mauricio Piacentini ([@piacentini](#)) from Tabuleiro Producoes, as the Arca Database Browser. The original version was used as a free companion tool to the Arca Database Xtra, a commercial product that embeds SQLite databases with some additional extensions to handle compressed and binary data.

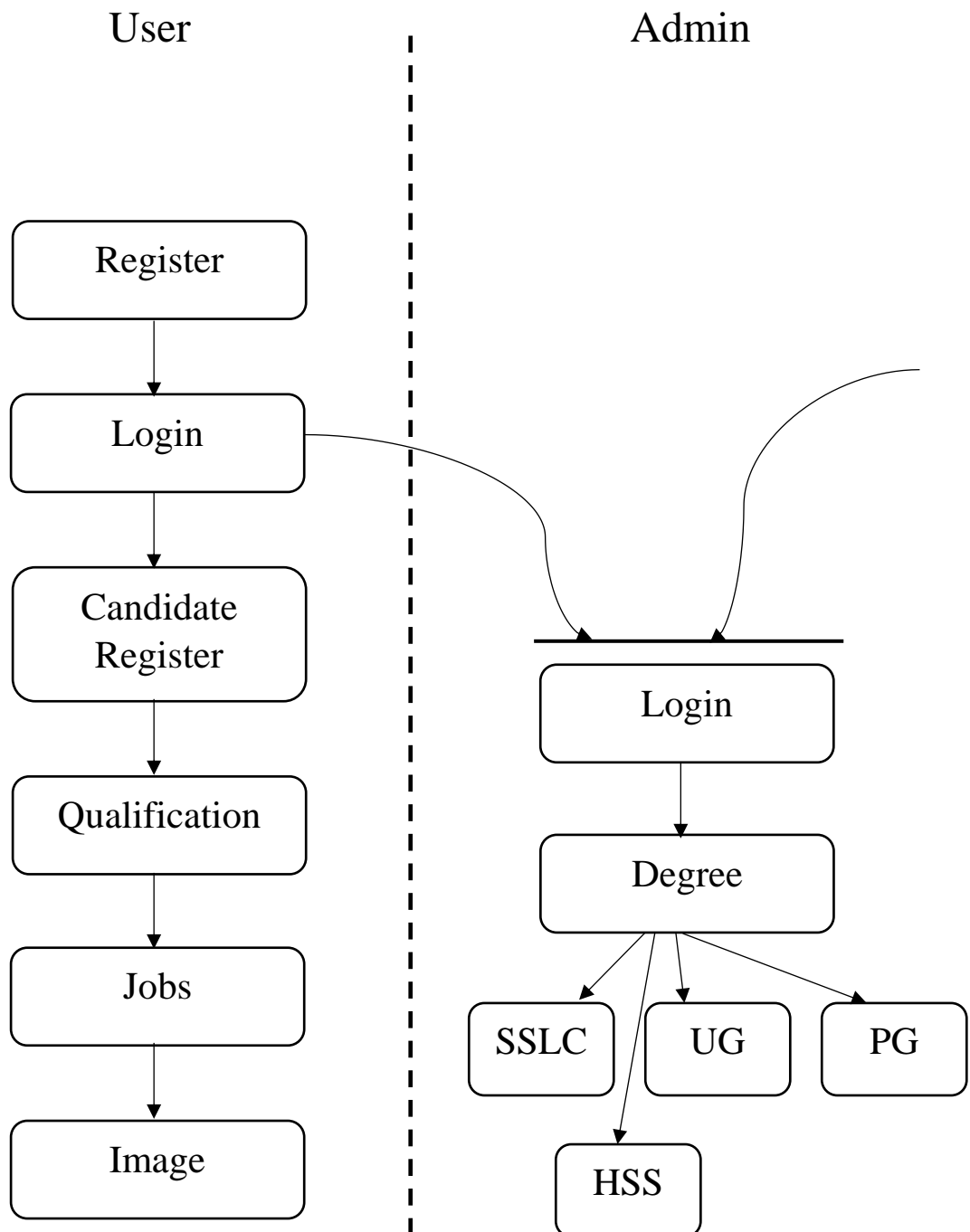
The original code was trimmed and adjusted to be compatible with standard SQLite 2.x databases. The resulting program was renamed SQLite Database Browser, and released into the Public Domain by Mauricio. Icons were contributed by [Raquel Ravanini](#), also from Tabuleiro. Jens Miltner ([@jmiltner](#)) contributed the code to support SQLite 3.x databases for the 1.2 release.

Pete Morgan ([@daffodil](#)) created an initial project on GitHub with the code in 2012, where several contributors fixed and improved pieces over the years. René Peinthor ([@rp-](#)) and Martin Kleusberg ([@MKleusberg](#)) then became involved, and were the main driving force from that point. Justin Clift ([@justinclift](#)) started the new [github.com/sqlitebrowser](#) organisation on GitHub, (roughly) co-ordinates the project, and creates our Windows and macOS nightly builds. [John T. Haller](#), of [PortableApps.com](#) fame, created the new logo. He based it on the Tango icon set (public domain).

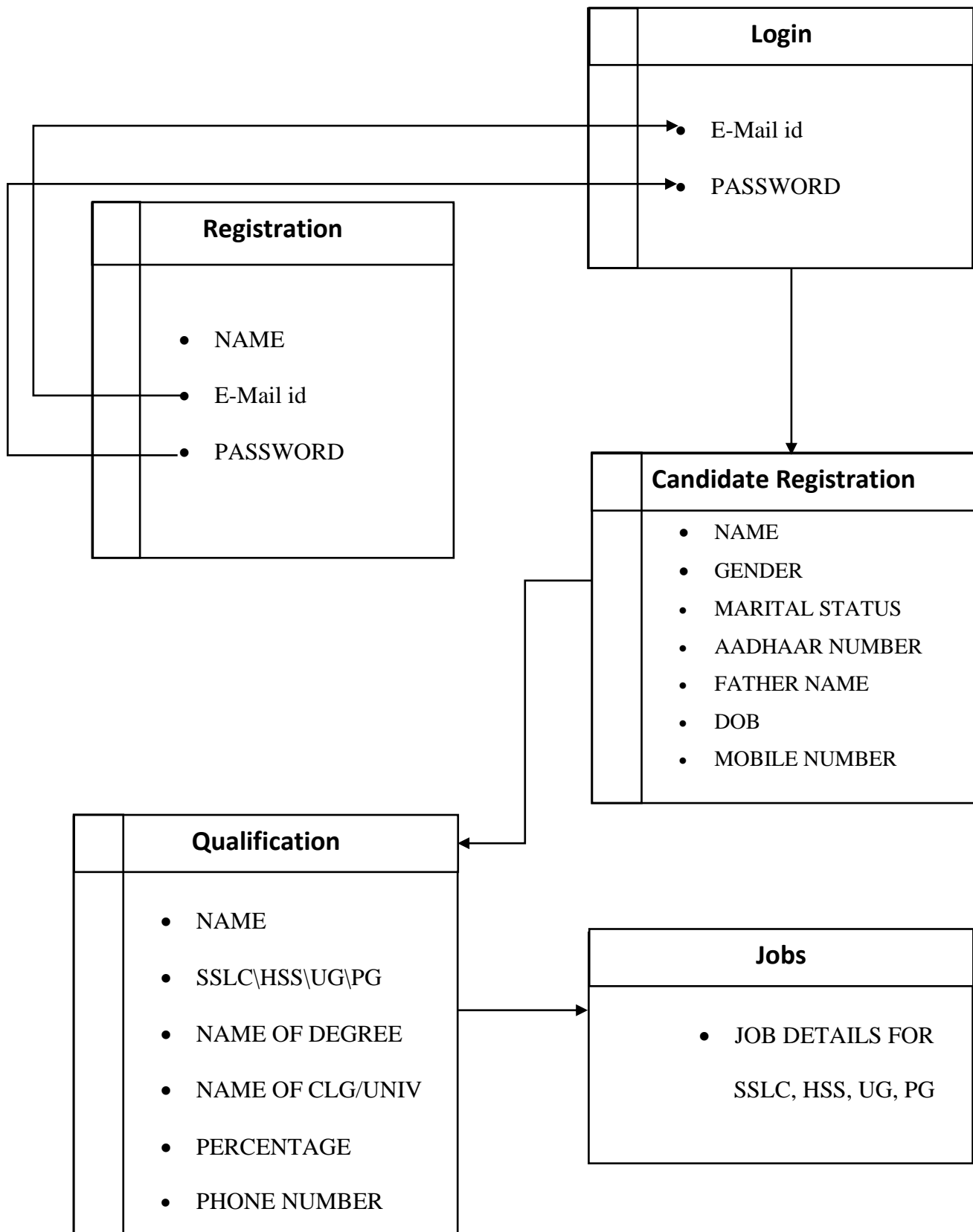
In August 2014, the project was renamed to “Database Browser for SQLite” at the request of [Richard Hipp](#) (creator of [SQLite](#)), as the previous name was creating unintended support issues. In September 2014, the project was renamed to “DB Browser for SQLite”, to avoid confusion with an existing application called “Database Browser”.

## 5. SYSTEM DESIGN

### 5.1 Input Design



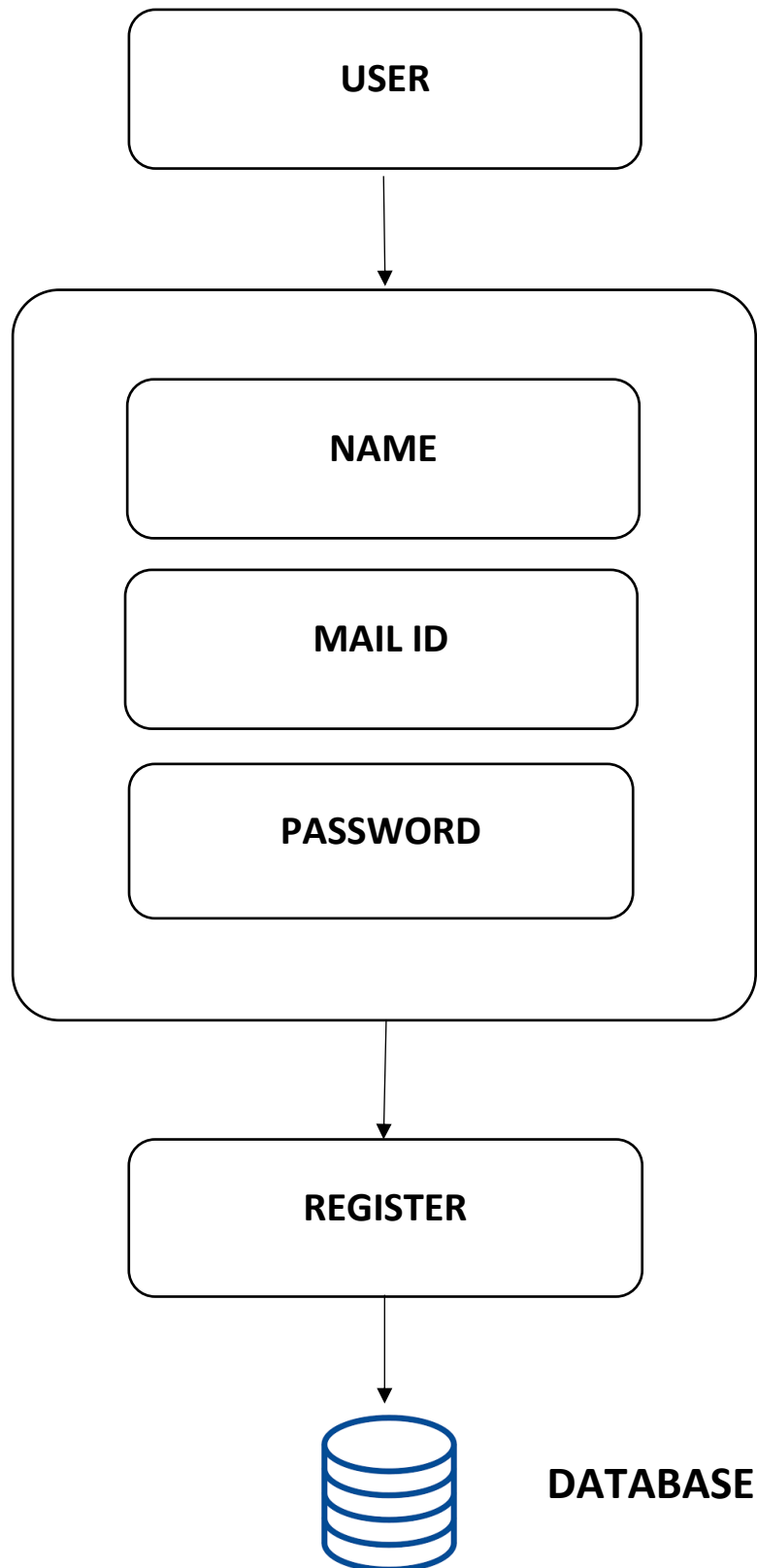
## 5.2 Database Design

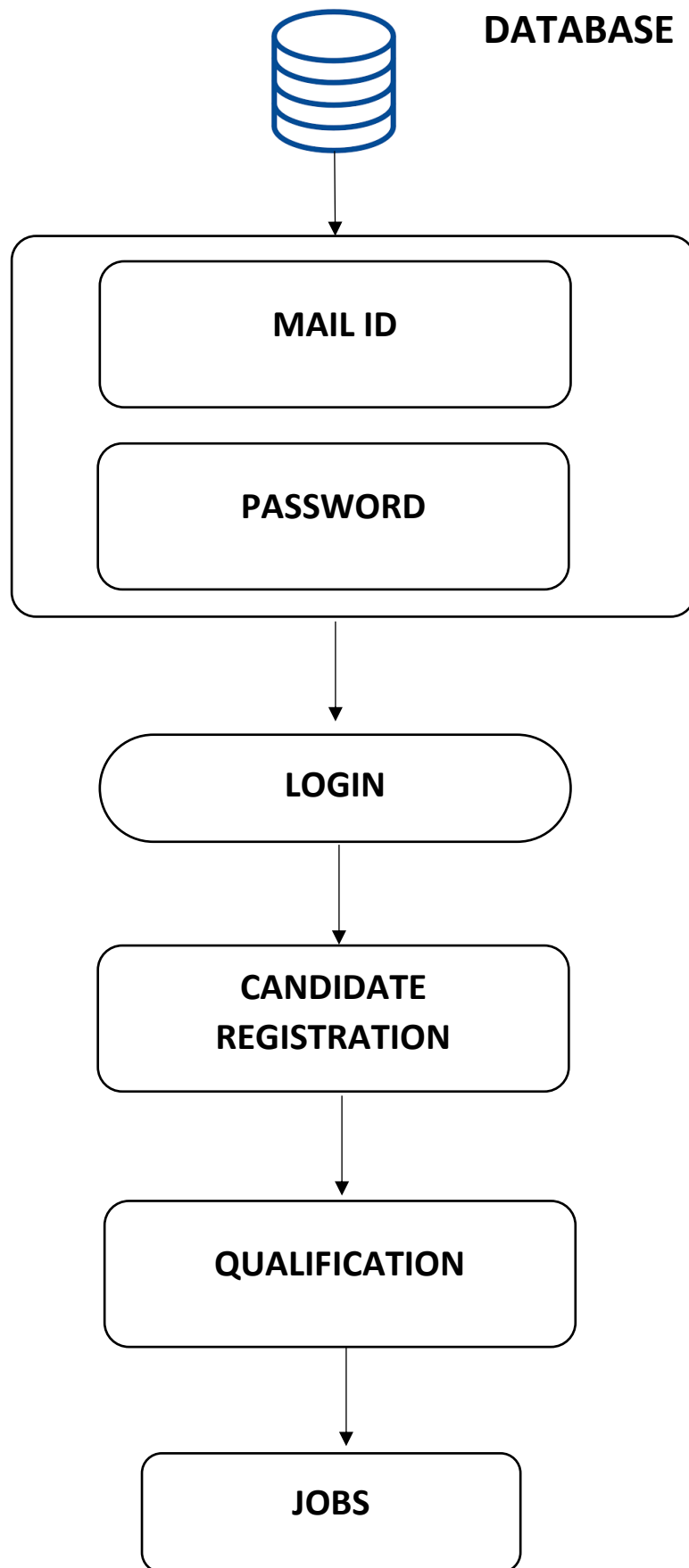


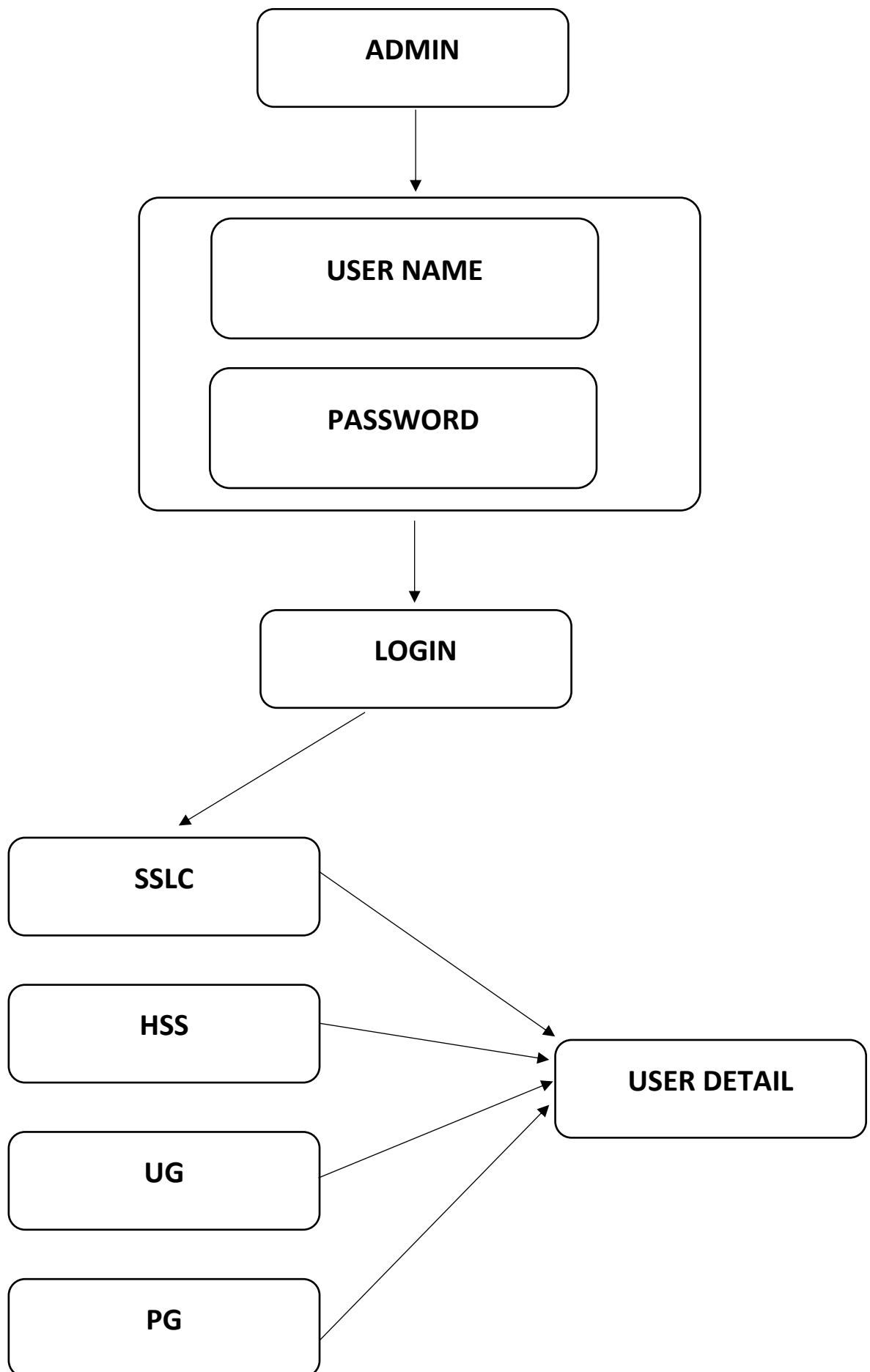


## 6. SYSTEM ARCHITECTURE

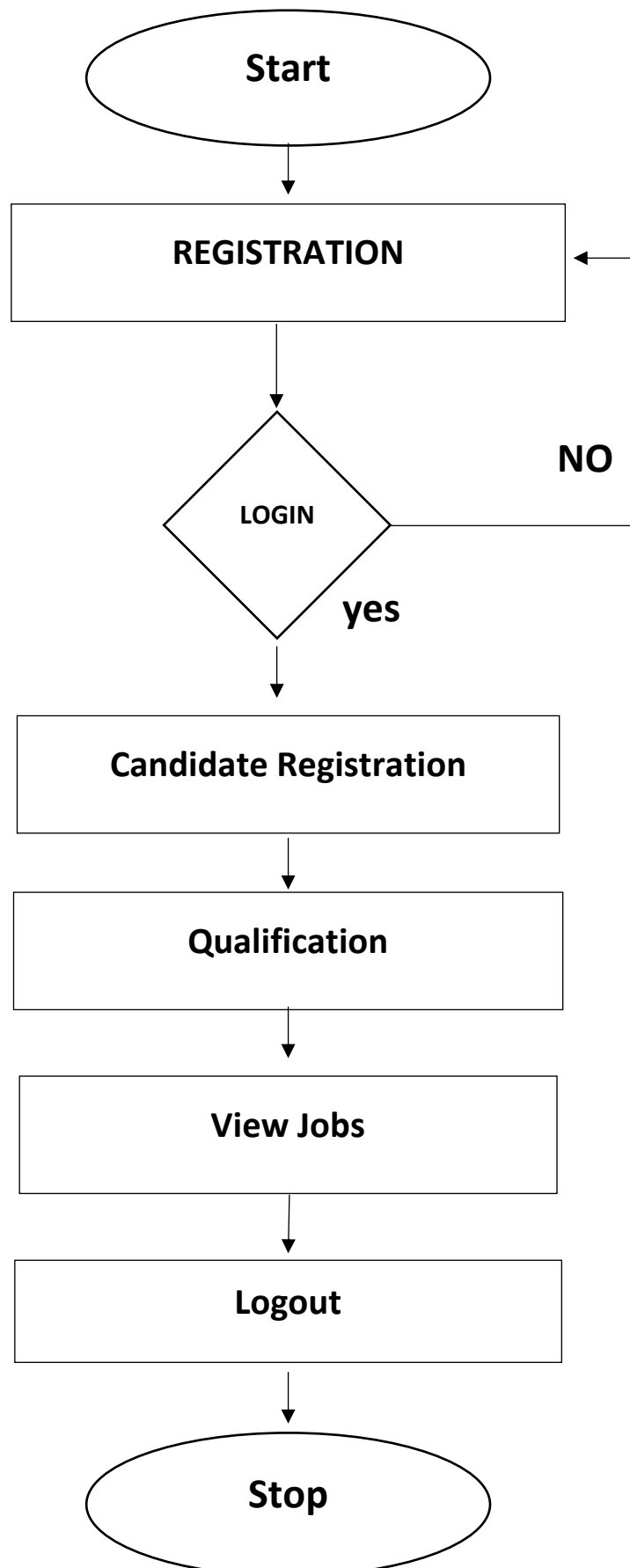
### 6.1 System Architecture





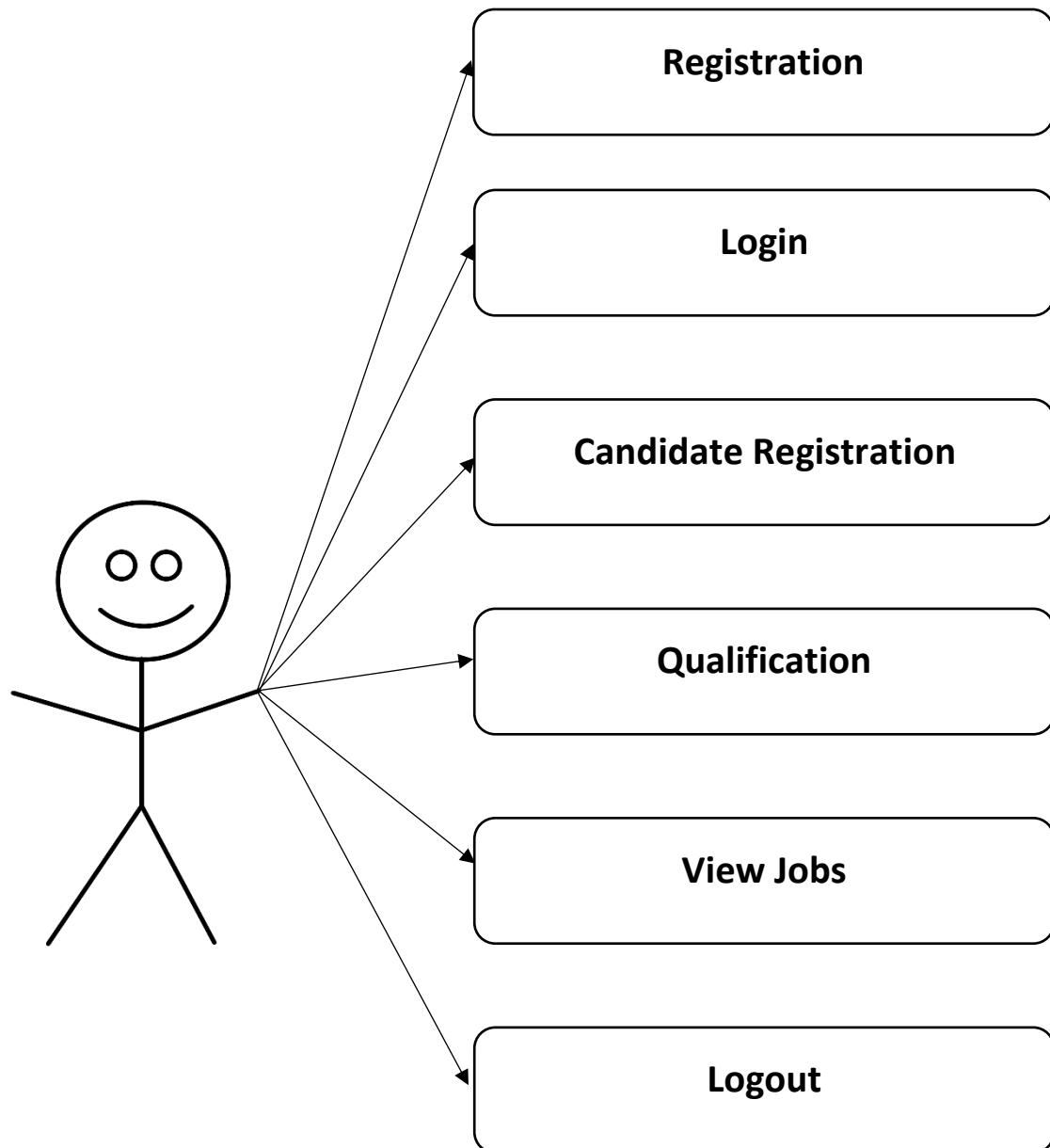


## 6.2 Flow Chart



## 6.3 UML Diagram

### USECASE



## **7. SYSTEM IMPLEMENTATION**

Implementation is the most crucial stage in achieving a successful system and giving the user's confidence that the new system is workable and effective. Implementation of a modified application to replace an existing one. This type of conversation is relatively easy to handle, provide there are no major changes in the system.

Each program is testing individually at the time of development using the data and has verified that this program linked together in the way specified in the programs specification, the computer system and its environment is tested to the satisfaction of the user. The system that has been developed is accepted and proved to be satisfactory for the user. And so, the system is going to be implemented very soon. A simple operating procedure is included so that the user can understand the different functions clearly and quickly.

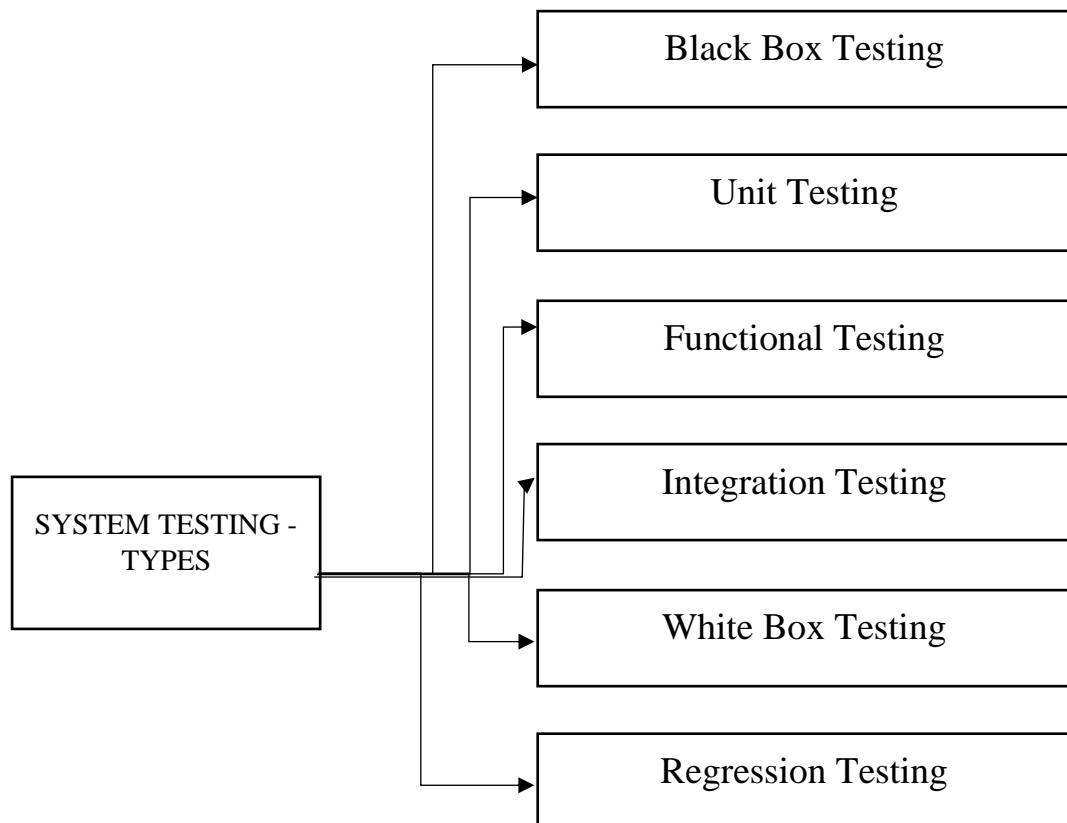
The purpose of System Implementation can be summarized as follows: making the new system available to a prepared set of users , and positioning on-going support and maintenance of the system within the Performing Organization .

We implement a project for the candidate who wants to apply for government jobs. This application provide jobs based on their qualification and also for freshers. The data are stored in database. Here, SQLite is the database which store all candidate's details.

Initially as a first step the executable form of the application is to be created and loaded in the common server machine which is accessible to the entire user and the server is to be connected to a network. The final stage is to document the entire system which provides components and the operating procedures of the system.

## 8. SYSTEM TESTING

System Testing (ST) is a black box testing technique performed to evaluate the complete system the system's compliance against specified requirements. In System testing, the functionalities of the system are tested from an end-to-end perspective. System Testing is usually carried out by a team that is independent of the development team in order to measure the quality of the system unbiased. It includes both functional and Non-Functional testing.



## 8.1 Black box Testing

Initially, the requirements and specifications of the system are examined. Tester chooses valid inputs (positive test scenario) to check whether SUT processes them correctly. Also, some invalid inputs (negative test scenario) are chosen to verify that the SUT is able to detect them. Tester determines expected outputs for all those inputs. Software tester constructs test cases with the selected inputs. The test cases are executed. Software tester compares the actual outputs with the expected outputs. Defects if any are fixed and re-tested.

First time we test this application without any idea. We give code for some module and test it ,then clear the error which was come.

## 8.2 Unit Testing

**Unit testing** is a type of software testing where individual units or components of a software are tested. The purpose is to validate that each unit of the software code performs as expected. Unit Testing is done during the development (coding phase) of an application by the developers. Unit Tests isolate a section of code and verify its correctness. A unit may be an individual function, method, procedure, module, or object. In SDLC, STLC, V Model, Unit testing is first level of testing done before integration testing. Unit testing is a White Box testing technique that is usually performed by the developer. Though, in a practical world due to time crunch or reluctance of developers to tests, QA engineers also do unit testing.

Every modules are tested after a completion of code for a module. Likewise , Every modules are tested and the errors are cleared.



## 8.3 Functional testing

**Functional testing** is a type of software testing that validates the software system against the functional requirements/specifications. The purpose of Functional tests is to test each function of the software application, by providing appropriate input, verifying the output against the Functional requirements.

Functional testing mainly involves black box testing and it is not concerned about the source code of the application. This testing checks User Interface, APIs, Database, Security, Client/Server communication and other functionality of the Application Under Test. The testing can be done either manually or using automation. This application has many functions like onClick, onCreate setOnClickListener etc., each and every functions are tested separately. These functions are used to move next page after entering the details of the candidate.

## 8.4 Integration Testing

**Integration Testing** is defined as a type of testing where software modules are integrated logically and tested as a group. A typical software project consists of multiple software modules, coded by different programmers. The purpose of this level of testing is to expose defects in the interaction between these software modules when they are integrated. Integration Testing focuses on checking data communication amongst these modules. Hence it is also termed as '**I & T**' (Integration and Testing), '**String Testing**' and sometimes '**Thread Testing**'. We tested this application by combining two modules such as register and login.

## **8.5 White Box Testing**

White box testing techniques analyse the internal structures the used data structures, internal design, code structure and the working of the software rather than just the functionality as in black box testing. It is also called glass box testing or clear box testing or structural testing. Developers use this testing method to verify the flow of inputs and outputs through the application, improving usability and design and strengthening security. We understand every module's designing and working of the module and test each and every module separately.

## **8.6 Regression Testing**

Regression Testing is defined as a type of software testing to confirm that a recent program or code change has not adversely affected existing features. Regression Testing is nothing but a full or partial selection of already executed test cases which are re-executed to ensure existing functionalities work fine. This testing is done to make sure that new code changes should not have side effects on the existing functionalities. It ensures that the old code still works once the latest code changes are done. The app was tested after making some changes in the code that will not affect the application. It's tested through many android version but it is not supported for latest versions above 11.

## **9. CONCLUSION**

In this Report, An Android framework based on jobs seekers can search for their suitable jobs based on their qualification for searching government jobs and employers can see their vacancies in this app supports the process of job seeker Registration. In this project Admin can also view the details of the candidate. But enhancement to the project can easily be made without changing the current design and programming structure.

Since this project has been designed exclusively as a project, certain complexities that do faced by any real life manual problem like total no. of employee, address redundancy etc. are considered in this project. But enhancement to the project can easily be made without changing the current design and programming structure.

## 10. APPENDIX

### 10.1 Sample Source Code

#### AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
    xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.login">

    <application
        android:icon="@drawable/tner"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/Theme.AppCompat">

        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".RegisterActivity">
            <intent-filter>
                <action android:name="com.example.login.RegisterActivity" />
                <category android:name="android.intent.category.DEFAULT" />
            </intent-filter>
        </activity>
        <activity android:name=".adminlog">
            <intent-filter>
                <action android:name="com.example.login.adminlog" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

```

        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
<activity android:name=".view">
    <intent-filter>
        <action android:name="com.example.login.view" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
<activity android:name=".DisplaySslc">
    <intent-filter>
        <action android:name="com.example.login.DisplaySslc" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
<activity android:name=".HSS">
    <intent-filter>
        <action android:name="com.example.login.HSS" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
<activity android:name=".ugjob">
    <intent-filter>
        <action android:name="com.example.login.ugjob" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
<activity android:name=".pgjobs">
    <intent-filter>
        <action android:name="com.example.login.pgjob" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
<activity android:name=".LASTPAGE">

```

```

<intent-filter>
    <action android:name="com.example.login.LASTPAGE" />
    <category android:name="android.intent.category.DEFAULT" />
</intent-filter>
</activity>
<activity android:name=".DashboardActivity">
    <intent-filter>
        <action android:name="com.example.login.DashboardActivity" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
<activity android:name=".Form">
    <intent-filter>
        <action android:name="com.example.login.Form" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
<activity android:name=".form2">
    <intent-filter>
        <action android:name="com.example.login.form2" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
<activity android:name=".job">
    <intent-filter>
        <action android:name="com.example.login.job" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
</application>
</manifest>

```

## SQLitehepler.java

```
package com.example.login;

import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteOpenHelper;
import android.database.sqlite.SQLiteDatabase;
import android.widget.TextView;

public class SQLiteHelper extends SQLiteOpenHelper
{
    public static final String DATABASE_NAME="project";
    public static final String TABLE_NAME="UserTable";
    public static final String Table_Columnn_ID="id";
    public static final String Table_Columnn_1_Name="name";
    public static final String Table_Columnn_2_Email="email";
    public static final String Table_Columnn_3_Password="password";
    public static final String TABLE_NAE1="detail";
    public static final String Table_Columnn_ID1="id";
    public static final String Table_Columnn_11_Name1="name1";
    public static final String Table_Columnn_12_GENDER="gender";
    public static final String Table_Columnn_13_Marital_status="msts";
    public static final String Table_Columnn_14_Aadhaar="acard";
    public static final String Table_Columnn_15_Fathername="fname";
    public static final String Table_Columnn_16_dob="dob";
    public static final String Table_Columnn_17_phno="phno";
    public static final String TABLE_NAME2="qualification";
    public static final String Table_Columnn_ID2="id";
    public static final String Table_Columnn_18_name="name";
    public static final String Table_Columnn_19_course="course";
    public static final String Table_Columnn_20_deg="degree";
    public static final String Table_Columnn_21_clg="college";
    public static final String Table_Columnn_22_per="per";
    public static final String Table_Columnn_23_phone="phone";
```

```

public SQLiteHelper(Context context)
{
    super(context, DATABASE_NAME, null, 1);
}

@Override
public void onCreate(SQLiteDatabase database)
{
    String CREATE_TABLE="CREATE TABLE IF NOT EXISTS "+TABLE_NAME+"
("+Table_Column_ID+" INTEGER PRIMARY KEY, "+Table_Column_1_Name+"
VARCHAR, "+Table_Column_2_Email+" VARCHAR, "+Table_Column_3_Password+"
VARCHAR)";

    database.execSQL(CREATE_TABLE);

    String CREATE_TABLE1="CREATE TABLE IF NOT EXISTS "+TABLE_NAME1+"
("+Table_Column_ID1+" INTEGER PRIMARY KEY, "+Table_Column_11_Name1+"
VARCHAR,"+Table_Column_12_GENDER+"
VARCHAR,"+Table_Column_13_Marital_status+"
VARCHAR,"+Table_Column_14_Aadhaar+"
INTEGER,"+Table_Column_15_Fathername+" VARCHAR,"+Table_Column_16_dob+"
INTEGER,"+Table_Column_17_phno+" INTEGER)";

    database.execSQL(CREATE_TABLE1);

    String CREATE_TABLE2="CREATE TABLE IF NOT EXISTS "+TABLE_NAME2+"
("+Table_Column_ID2+" INTEGER PRIMARY KEY, "+Table_Column_18_name+"
VARCHAR,"+Table_Column_19_course+" VARCHAR,"+Table_Column_20_deg+"
VARCHAR,"+Table_Column_21_clg+" VARCHAR,"+Table_Column_22_per+"
INTEGER,"+Table_Column_23_phone +" INTEGER)";

    database.execSQL(CREATE_TABLE2);
}

@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion)
{
    db.execSQL("DROP TABLE IF EXISTS "+TABLE_NAME);
    onCreate(db);
    db.execSQL("DROP TABLE IF EXISTS "+TABLE_NAME1);
}

```



```

        onCreate(db);
        db.execSQL("DROP TABLE IF EXISTS "+TABLE_NAME2);
        onCreate(db);
    }
    public Cursor getAllDetail()
    {
        SQLiteDatabase db=this.getWritableDatabase();
        Cursor res = db.rawQuery("select * from qualification WHERE course = ?",new String[]
{"ssl"});
        return res;
    }
    public Cursor AllDetail()
    {
        SQLiteDatabase db=this.getWritableDatabase();
        Cursor res = db.rawQuery("select * from qualification WHERE course = ?",new String[]
{"hss"});
        return res;
    }
    public Cursor ugDetail()
    {
        SQLiteDatabase db=this.getWritableDatabase();
        Cursor res = db.rawQuery("select * from qualification WHERE course = ?",new String[]
{"ug"});
        return res;
    }
    public Cursor pgDetail()
    {
        SQLiteDatabase db=this.getWritableDatabase();
        Cursor res = db.rawQuery("select * from qualification WHERE course = ?",new String[]
{"pg"});
        return res;
    }
}

```

## **MainActivity.java**

```
package com.example.login;

import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.text.TextUtils;
import android.text.TextWatcher;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity
{
    Button LogInButton, RegisterButton, Admin;
    EditText Email, Password;
    ImageView img;
    String EmailHolder, PasswordHolder;
    Boolean EditTextEmptyHolder;
    SQLiteDatabase sqLiteDatabaseObj;
    SQLiteHelper sqLiteHelper;
    Cursor cursor;
    String TempPassword = "NOT_FOUND";
    public static final String UserEmail = "";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

```

LoginButton = (Button) findViewById(R.id.buttonLogin);
RegisterButton = (Button) findViewById(R.id.buttonRegister);
Admin = (Button) findViewById(R.id.adbtn);
Email = (EditText) findViewById(R.id.editEmail);
Password = (EditText) findViewById(R.id.editPassword);
img=(ImageView)findViewById(R.id.imageView);
sqliteHelper = new SQLiteHelper(this);
//Adding click listener to log in button.
LoginButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        // Calling EditText is empty or no method.
        if(CheckEditTextStatus()) {
            if(validation()) {
                // Calling login method.
                LoginFunction();
                CheckFinalResult();
            }
        }
        else
        {
            Toast.makeText(MainActivity.this, "Please Enter UserName or Password.",
Toast.LENGTH_LONG).show();
        }
    }
});
// Adding click listener to register button.
RegisterButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        // Opening new user registration activity using intent on button click.
        Intent intent = new Intent(MainActivity.this, RegisterActivity.class);
        startActivity(intent);
    }
}

```

```

    }
    );
    Admin.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            // Opening new user registration activity using inteant on button click.
            Intent intent = new Intent(MainActivity.this, adminlog.class);
            startActivity(intent);
        }
    }
    );
}
// Login function starts from here.
public void LoginFunction() {
    sqLiteDatabaseObj = sqLiteHelper.getWritableDatabase();
    // Adding search email query to cursor.
    cursor = sqLiteDatabaseObj.query(SQLiteHelper.TABLE_NAME, null, " " +
    SQLiteHelper.Table_Column_2_Email + "=?", new String[]{EmailHolder}, null, null, null);
    while (cursor.moveToNext()) {
        if (cursor.isFirst()) {
            cursor.moveToFirst();
            // Storing Password associated with entered email.
            TempPassword =
cursor.getString(cursor.getColumnIndex(SQLiteHelper.Table_Column_3_Password));
            // Closing cursor.
            cursor.close();
        }
    }
}
public boolean validation() {
    String email = Email.getText().toString().trim();
    String emailPattern = "[a-zA-Z0-9._-]+@[a-z]+\\.+[a-z]+";
    if (email.matches(emailPattern))
    {

```

```

        return true;
    }
    else {
        Toast.makeText(getApplicationContext(), "Invalid email address",
Toast.LENGTH_SHORT).show();
        return false;
    }
}

// Checking EditText is empty or not.
public boolean CheckEditTextStatus()
{
    // Getting value from All EditText and storing into String Variables.
    EmailHolder = Email.getText().toString();
    PasswordHolder = Password.getText().toString();
    // Checking EditText is empty or no using TextUtils.
    if (TextUtils.isEmpty(EmailHolder) || TextUtils.isEmpty(PasswordHolder))
    {
        EditTextEmptyHolder = false;
    }
    else
    {
        EditTextEmptyHolder = true;
    }
    return EditTextEmptyHolder;
}

// Checking entered password from SQLite database email associated password.
public void CheckFinalResult () {
    if (TempPassword.equalsIgnoreCase(PasswordHolder))
    {
        Email.getText().clear();
        Password.getText().clear();
        Toast.makeText(MainActivity.this, "Login Successful",
Toast.LENGTH_LONG).show();
        // Going to Dashboard activity after login success message.

```

```

        Intent intent = new Intent(MainActivity.this, DashboardActivity.class);
        // Sending Email to Dashboard Activity using intent.
        intent.putExtra(UserEmail, EmailHolder);
        startActivity(intent);
    } else {
        Toast.makeText(MainActivity.this, "UserName or Password is Wrong, Please Try
Again.", Toast.LENGTH_LONG).show();
    }
    TempPassword = "NOT_FOUND";
}
}

```

### **RegisterActivity.java**

```

package com.example.login;

import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class RegisterActivity extends AppCompatActivity
{
    EditText Email, Password, Name ;
    Button Register,login;
    String NameHolder, EmailHolder, PasswordHolder;
    Boolean EditTextEmptyHolder;
    SQLiteDatabase sqLiteDatabaseObj;
    String SQLiteDataBaseQueryHolder ;
    SQLiteHelper sqLiteHelper;
    Cursor cursor;
    String F_Result = "Not_Found";

    @Override
    protected void onCreate(Bundle savedInstanceState)

```

```

{
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_register);
    Register = (Button)findViewById(R.id.buttonRegister);
    login=(Button) findViewById(R.id.logbtn);
    Email = (EditText)findViewById(R.id.editEmail);
    Password = (EditText)findViewById(R.id.editPassword);
    Name = (EditText)findViewById(R.id.editName);
    sqLiteHelper = new SQLiteHelper(this);

    // Adding click listener to register button.
    Register.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            // Creating SQLite database if dose n't exists
            SQLiteDataBaseBuild();
            // Creating SQLite table if dose n't exists.
            SQLiteTableBuild();
            // Checking EditText is empty or Not.
            CheckEditTextStatus();
            // Method to check Email is already exists or not.
            CheckingEmailAlreadyExistsOrNot();
            // Empty EditText After done inserting process.
            EmptyEditTextAfterDataInsert();
        }
    });
    login.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent = new Intent(RegisterActivity.this, MainActivity.class);
            startActivity(intent);
        }
    });
}

// SQLite database build method.
public void SQLiteDataBaseBuild()
{
    sqLiteDatabaseObj = openOrCreateDatabase(SQLiteHelper.DATABASE_NAME,
    Context.MODE_PRIVATE, null);
}

// SQLite table build method.
public void SQLiteTableBuild()
{
    sqLiteDatabaseObj.execSQL("CREATE TABLE IF NOT EXISTS " +
    SQLiteHelper.TABLE_NAME + "(" + SQLiteHelper.Table_Column_ID + " PRIMARY KEY
    AUTOINCREMENT NOT NULL, " + SQLiteHelper.Table_Column_1_Name + "
    VARCHAR, " + SQLiteHelper.Table_Column_2_Email + " VARCHAR, " +
    SQLiteHelper.Table_Column_3_Password + " VARCHAR);");
}

// Insert data into SQLite database method.

```

```

public void InsertDataIntoSQLiteDatabase() {
    if (validation()) {

        // If editText is not empty then this block will executed.
        if (EditTextEmptyHolder) {

            // SQLite query to insert data into table.
            SQLiteDataBaseQueryHolder = "INSERT INTO " + SQLiteHelper.TABLE_NAME
+ " (name,email,password) VALUES('" + NameHolder + "', '" + EmailHolder + "', '" +
PasswordHolder + "');"
            // Executing query.
            sqLiteDatabaseObj.execSQL(SQLiteDataBaseQueryHolder);
            // Closing SQLite database object.
            sqLiteDatabaseObj.close();
            // Printing toast message after done inserting.
            Toast.makeText(RegisterActivity.this, "User Registered Successfully",
Toast.LENGTH_LONG).show();
        }
        // This block will execute if any of the registration EditText is empty.
        else {
            // Printing toast message if any of EditText is empty.
            Toast.makeText(RegisterActivity.this, "Please Fill All The Required Fields.",
Toast.LENGTH_LONG).show();
        }
    }
}

// Empty edittext after done inserting process method.
public void EmptyEditTextAfterDataInsert(){
    Name.getText().clear();
    Email.getText().clear();
    Password.getText().clear();
}

// Method to check EditText is empty or Not.
public void CheckEditTextStatus()
{
    // Getting value from All EditText and storing into String Variables.
    NameHolder = Name.getText().toString() ;
    EmailHolder = Email.getText().toString();
    PasswordHolder = Password.getText().toString();
    if(TextUtils.isEmpty(NameHolder) || TextUtils.isEmpty(EmailHolder) ||
TextUtils.isEmpty(PasswordHolder)){
        EditTextEmptyHolder = false ;
    }
    else
    {
        EditTextEmptyHolder = true ;
    }
}

// Checking Email is already exists or not.
public void CheckingEmailAlreadyExistsOrNot(){

```



```

// Opening SQLite database write permission.
sqliteDatabaseObj = sqliteHelper.getWritableDatabase();
// Adding search email query to cursor.
cursor = sqliteDatabaseObj.query(SQLiteHelper.TABLE_NAME, null, " " +
SQLiteHelper.Table_Column_2_Email + "=?", new String[]{EmailHolder}, null, null, null);
while (cursor.moveToNext()) {
    if (cursor.isFirst()) {
        cursor.moveToFirst();
        // If Email is already exists then Result variable value set as Email Found.
        F_Result = "Email Found";
        // Closing cursor.
        cursor.close();
    }
}
// Calling method to check final result and insert data into SQLite database.
CheckFinalResult();
}
public boolean validation() {
    String email = Email.getText().toString().trim();
    String emailPattern = "[a-zA-Z0-9._-]+@[a-z]+\\.[a-z]+";
    if (email.matches(emailPattern))
    {
        return true;
    }
    else {
        Toast.makeText(getApplicationContext(), "Invalid email address",
Toast.LENGTH_SHORT).show();
        return false;
    }
}
// Checking result
public void CheckFinalResult(){
    // Checking whether email is already exists or not.
    if(F_Result.equalsIgnoreCase("Email Found"))
    {
        // If email is exists then toast msg will display.
        Toast.makeText(RegisterActivity.this,"Email Already
Exists",Toast.LENGTH_LONG).show();
    }
    else {
        // If email already dose n't exists then user registration details will entered to SQLite
        database.
        InsertDataIntoSQLiteDatabase();
    }
    F_Result = "Not_Found" ;
}
}

```

## **DashBoard.java**

```
package com.example.login;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
public class DashboardActivity extends AppCompatActivity
{
    String EmailHolder;
    TextView Email;
    Button LogOUT ;
    Button Next;
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_dashboard);
        Email = (TextView)findViewById(R.id.textView1);
        LogOUT = (Button)findViewById(R.id.button1);
        Next = (Button)findViewById(R.id.button2);
        Intent intent = getIntent();
        // Receiving User Email Send By MainActivity.
        EmailHolder = intent.getStringExtra(MainActivity.UserEmail);
        // Setting up received email to TextView.
        Email.setText(Email.getText().toString()+EmailHolder);
        // Adding click listener to Log Out button.
        LogOUT.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                //Finishing current DashBoard activity on button click.
                finish();
                Toast.makeText(DashboardActivity.this,"Log Out Successful",
                Toast.LENGTH_LONG).show();
            }
        });
        Next.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view)
            {
                Intent intent = new Intent(DashboardActivity.this, Form.class);
                startActivity(intent);
            }
        });
    }
}
```

## **Form.java**

```
package com.example.login;

import android.content.Context;
import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class Form extends AppCompatActivity
{
    EditText Name1,gender,Marital_status,Aadhaar,Fathername,dob,phno ;
    Button last;
    String NameHolder,
genderHolder,Marital_statusHolder,AadhaarHolder,FathernameHolder,dobHolder,
phnoHolder;
    Boolean EditTextEmptyHolder;
    SQLiteDatabase sqLiteDatabaseObj;
    String SQLiteDatabaseQueryHolder ;
    SQLiteHelper sqLiteHelper;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_form1);
        last = (Button)findViewById(R.id.next);
        Name1 = (EditText)findViewById(R.id.edittxt1);
        gender=(EditText)findViewById(R.id.gender);
        Marital_status = (EditText)findViewById(R.id.marri);
        Aadhaar = (EditText)findViewById(R.id.edittxt4);
        Fathername = (EditText)findViewById(R.id.edittxt5);
        dob = (EditText)findViewById(R.id.editTextDate);
        phno=(EditText)findViewById(R.id.editTextNumber);
        sqLiteHelper = new SQLiteHelper(this);
        // Adding click listener to register button.
        last.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                SQLiteDatabaseBuild();
                // Creating SQLite table if dose n't exists.
                SQLiteTableBuild();
                // Checking EditText is empty or Not.
                if(CheckEditTextStatus())
                {
```

```

        validation();
        aadhar();
        dob();
        // Empty EditText After done inserting process.
    }
    CheckEditTextStatus();
    InsertDataIntoSQLiteDatabase();
    EmptyEditTextAfterDataInsert();
}
});
}
public void SQLiteDataBaseBuild()
{
    sqliteDatabaseObj = openOrCreateDatabase(SQLiteHelper.DATABASE_NAME,
Context.MODE_PRIVATE, null);
}
// SQLite table build method.
    public void SQLiteTableBuild()
    {
        sqliteDatabaseObj.execSQL("CREATE TABLE IF NOT EXISTS " +
SQLiteHelper.TABLE_NAE1+ "(" + SQLiteHelper.Table_Column_ID1 + " PRIMARY KEY
AUTOINCREMENT NOT NULL, " + SQLiteHelper.Table_Column_11_Name1 + "
VARCHAR(60), " + SQLiteHelper.Table_Column_12_GENDER + " VARCHAR, " +
SQLiteHelper.Table_Column_13_Marital_status + " VARCHAR, " +
SQLiteHelper.Table_Column_14_Aadhaar + "
INTEGER,"+SQLiteHelper.Table_Column_15_Fathurname +
"VARCHAR,"+SQLiteHelper.Table_Column_16_dob +
"INTEGER,"+SQLiteHelper.Table_Column_17_phno+" INTEGER)");
    }
    // Insert data into SQLite database method.
    public void InsertDataIntoSQLiteDatabase()
    {
        // If editText is not empty then this block will executed.
        if(EditTextEmptyHolder)
        {
            // SQLite query to insert data into table.
            SQLiteDataBaseQueryHolder = "INSERT INTO "+SQLiteHelper.TABLE_NAE1+"
(name1,gender,msts,acard,fname,dob,phno)
VALUES('"+NameHolder+"','"+genderHolder+"', '"
+Marital_statusHolder+"','"+AadhaarHolder+"','"+FathurnameHolder+"','"+dobHolder+"','"+
phnoHolder+"')";
            // Executing query.
            sqliteDatabaseObj.execSQL(SQLiteDataBaseQueryHolder);
            // Closing SQLite database object.
            sqliteDatabaseObj.close();
            net();
        }
        else {
            Toast.makeText(Form.this, "Please Fill All The Required Fields.",
Toast.LENGTH_LONG).show();

```

```

    }
}
public boolean CheckEditTextStatus()
{
    // Getting value from All EditText and storing into String Variables.
    NameHolder = Name1.getText().toString() ;
    genderHolder=gender.getText().toString() ;
    Marital_statusHolder=Marital_status.getText().toString();
    AadhaarHolder=Aadhaar.getText().toString();
    FathernameHolder=Fathername.getText().toString();
    dobHolder=dob.getText().toString();
    phnoHolder=phno.getText().toString();
    if(TextUtils.isEmpty(NameHolder)|| TextUtils.isEmpty(Marital_statusHolder)||
    TextUtils.isEmpty(AadhaarHolder)|| TextUtils.isEmpty(FathernameHolder)||
    TextUtils.isEmpty(dobHolder)|| TextUtils.isEmpty(phnoHolder))
    {
        EditTextEmptyHolder = false ;
    }
    else {
        EditTextEmptyHolder = true ;
    }
    return EditTextEmptyHolder;
}
// Empty edittext after done inserting process method.
public void EmptyEditTextAfterDataInsert()
{
    Name1.getText().clear();
    gender.getText().clear();
    Marital_status.getText().clear();
    Aadhaar.getText().clear();
    Fathername.getText().clear();
    dob.getText().clear();
    phno.getText().clear();
}
public boolean validation() {
    String phnum= phno.getText().toString().trim();
    String phnpattern = "[0-9]+";
    if (phnum.matches(phnpattern)&& phno.length() == 10)
    {

    }
    else {
        Toast.makeText(getApplicationContext(), "Invalid Phone Number",
        Toast.LENGTH_SHORT).show();
    }
    return true;
}
public boolean aadhar() {
    String adhar= Aadhaar .getText().toString().trim();
    String adharpattern = "[0-9]+";

```

```

        if (adhar.matches(adharpattern)&& Aadhaar.length() == 12)
        {

        }
        else {
            Toast.makeText(getApplicationContext(), "Invalid Aadhaar Number ",
Toast.LENGTH_SHORT).show();
        }
        return true;
    }
    public boolean dob() { //[a-zA-Z0-9._-]+@[a-z]+\.[a-z]+ ";
        String date= dob .getText().toString().trim();
        String dobpattern = "[0-9]+[0-9]+/[0-9]+[0-9]+/[0-9]+[0-9]+";
        if (date.matches(dobpattern)&& dob.length() == 10)
        {

        }
        else {

        }
        return true;
    }
    public void net()
    {
        Intent intent=new Intent(Form.this,form2.class);
        startActivity(intent);
    }
}

```

## **form2.java**

```

package com.example.login;

import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class form2 extends AppCompatActivity {
    EditText name,course, degree, college, per,phone;
    Button next;
}

```

```

String nameHolder,courseHolder, degHolder, clgHolder, perHolder,phoneHolder;
Boolean EditTextEmptyHolder;
SQLiteDatabase sqLiteDatabaseObj;
String SQLiteDataBaseQueryHolder;
SQLiteHelper sqLiteHelper;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.content_form2);
    next = (Button) findViewById(R.id.nextbtn);
    name=(EditText) findViewById(R.id.nnnn);
    course = (EditText) findViewById(R.id.qual);
    degree = (EditText) findViewById(R.id.edittxt1);
    college = (EditText) findViewById(R.id.edittxt2);
    per = (EditText) findViewById(R.id.edittxt3);
    phone = (EditText) findViewById(R.id.phone);
    sqLiteHelper = new SQLiteHelper(this);
    // Adding click listener to register button.
    next.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View view) {
            // Creating SQLite database if dose n't exists
            SQLiteDataBaseBuild();
            // Creating SQLite table if dose n't exists.
            SQLiteTableBuild();
            // Checking EditText is empty or Not.
            CheckEditTextStatus();
            // Method to check Email is already exists or not.
            InsertDataIntoSQLiteDatabase();
            // Empty EditText After done inserting process.
            EmptyEditTextAfterDataInsert();
        }
    });
}
// SQLite database build method.
public void SQLiteDataBaseBuild() {
    sqLiteDatabaseObj = openOrCreateDatabase(SQLiteHelper.DATABASE_NAME,
Context.MODE_PRIVATE, null);
}
// SQLite table build method.
public void SQLiteTableBuild()
{
    sqLiteDatabaseObj.execSQL("CREATE TABLE IF NOT EXISTS " +
SQLiteHelper.TABLE_NAME2 + "(" + SQLiteHelper.Table_Column_ID2 + " PRIMARY
KEY AUTOINCREMENT NOT NULL," + SQLiteHelper.Table_Column_18_name + "
VARCHAR," + SQLiteHelper.Table_Column_19_course + " VARCHAR," +
SQLiteHelper.Table_Column_20_deg+ " VARCHAR," +
SQLiteHelper.Table_Column_21_clg + " VARCHAR," +

```

```

SQLiteHelper.Table_Column_22_per + "
INTEGER,"+SQLiteHelper.Table_Column_23_phone + " INTEGER)");
}
// Insert data into SQLite database method.
public void InsertDataIntoSQLiteDatabase() {
    // If editText is not empty then this block will executed.
    if (EditTextEmptyHolder) {
        // SQLite query to insert data into table.
        SQLiteDatabaseQueryHolder = "INSERT INTO " + SQLiteHelper.TABLE_NAME2
+ " (name,course,degree,college,per,phone) VALUES (" + nameHolder + "," +
courseHolder + "," + degHolder + "," + clgHolder + "," + perHolder + "," + phoneHolder
+ ")";
        // Executing query.
        sqLiteDatabaseObj.execSQL(SQLiteDatabaseQueryHolder);
        // Closing SQLite database object.
        sqLiteDatabaseObj.close();
        // Printing toast message after done inserting.
        Toast.makeText(form2.this, "Data entered Successfully",
Toast.LENGTH_LONG).show();
        next();
    }
    // This block will execute if any of the registration EditText is empty.
    else {
        // Printing toast message if any of EditText is empty.
        Toast.makeText(form2.this, "Please Fill All The Required Fields.",
Toast.LENGTH_LONG).show();
    }
}
// Empty edittext after done inserting process method.
public void EmptyEditTextAfterDataInsert() {
    name.getText().clear();
    course.getText().clear();
    degree.getText().clear();
    college.getText().clear();
    per.getText().clear();
    phone.getText().clear();
}
// Method to check EditText is empty or Not.
public void CheckEditTextStatus() {
    // Getting value from All EditText and storing into String Variables.
    nameHolder = name.getText().toString();
    courseHolder = course.getText().toString();
    degHolder = degree.getText().toString();
    clgHolder = college.getText().toString();
    perHolder = per.getText().toString();
    phoneHolder=phone.getText().toString();
    if (TextUtils.isEmpty(nameHolder)||TextUtils.isEmpty(courseHolder) ||
TextUtils.isEmpty(degHolder) || TextUtils.isEmpty(clgHolder) ||
TextUtils.isEmpty(perHolder)||TextUtils.isEmpty(phoneHolder)) {

```



```

        EditTextEmptyHolder = false;
    }
    else
    {
        EditTextEmptyHolder = true;
    }
    validation();
}
public boolean validation() {
    String phno= phone.getText().toString().trim();
    String phonepattern = "[0-9]+";
    if (phno.matches(phonepattern)&& phone.length() == 10)
    {

    }
    else {
        Toast.makeText(getApplicationContext(), "Invalid Phone Number",
Toast.LENGTH_SHORT).show();
    }
    return true;
}
public void next()
{
    Intent intent=new Intent(form2.this,job.class);
    startActivity(intent);
}
}

```

### **job.java**

```

package com.example.login;

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

import androidx.appcompat.app.AppCompatActivity;

public class job extends AppCompatActivity {
    TextView jobss;
    Button next;
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }
}

```

```

setContentView(R.layout.job);
next = (Button) findViewById(R.id.button3);
jobss=(TextView)findViewById(R.id.jobss);
jobss.setText(" \nFOR SSLC CANDIDATE \n \n Draughtsman, Computer Operator and
Programming Assistant\n" +
"Multi-Purpose Hospital Health Worker\n" +
"Office Assistant\n" +
"Assistant, Record Clerk\n" +
"Worker, Helper\n" +
"Professional Assistant - I, JRF\n" +
"Assistant, Security Guard\n" +
"Assistant, Record Clerk\n" +
"Record Clerk, Assistant\n" +
"Assistant, Record Clerk\n" +
"Assistant, Security Guard\n" +
"Peon and Driver\n" +
"Record Clerk, Assistant\n" +
"Village Assistant\n" +
"Driver, Chowkidar\n" +
"Psychologist, Social Worker\n" +
"Record Clerk, Security/ Watchman\n" +
"Medical Officer, Staff Nurse\n" +
"Record Clerk, Assistant, Security / Watchman\n" +
"Record Clerk, Assistant, Security / Watchman\n" +
"Psychologist, Security\n" +
"Record Clerk, Assistant\n" +
"Medical Officer, Attender\n" +
"Medical Officer, Staff Nurse\n" +
"Medical Officer, Staff Nurse\n" +
"Medical Officer, Staff Nurse\n" +
"Medical Officer, Staff Nurse\n" +
"Driver\n" +
"Wireman\n" +
"MHC - Driver, Physiotherapist\n" +
"Medical Officer, Staff Nurse\n" +
"Office Assistant\n" +
"Peon\n" +
"Staff Nurse, Data Entry Operator, Accountant, Social Worker & Various\n" +
"Junior Assistant, Typist\n" +
"Junior Assistant, Driver\n" +
"Clerical Assistant, Peon\n" +
"Peon\n"+" \n \n FOR HSS CANDIDATE"+" \n \n National Health Mission Tamil
Nadu \n" +
"District Health Society Sivaganga \n" +
"Tamil Nadu Civil Supplies Corporation Thanjavur \n" +
"Greater Chennai Corporation \n" +
"Tamil Nadu Civil Supplies Corporation Theni \n" +
"Greater Chennai Corporation \n" +
"Tamil Nadu Civil Supplies Corporation Thoothukudi \n" +
"Tamil Nadu Civil Supplies Corporation Tiruchirappalli \n" +

```

"Tamil Nadu Civil Supplies Corporation Perambalur \n" +  
 "Tamil Nadu Civil Supplies Corporation Dindigul \n" +  
 "District Health Society Trichy \n" +  
 "Tamil Nadu Civil Supplies Corporation Tiruvannamalai \n" +  
 "Tamil Nadu Civil Supplies Corporation Pudukkottai \n" +  
 "Tamil Nadu Civil Supplies Corporation \n" +  
 "Madurai Corporation \n" +  
 "District Health Society Virudhunagar \n" +  
 "Central University of Tamil Nadu \n" +  
 "Indian Oil Corporation Limited \n" +  
 "Chennai Corporation \n" +  
 "Sivaganga Health Department \n" +  
 "Ramanathapuram Juvenile Justice Board \n" +  
 "Tamil Nadu Civil Supplies Corporation \n" +  
 "Tidel Park Coimbatore \n" +  
 "Thanjavur District Health Society \n" +  
 "District Child Protection Unit kanniyakumari \n" +  
 "TamilNadu State Rural Livelihood Mission Karur \n" +  
 "Tamil Nadu Civil Supplies Corporation \n" +  
 "Government Mohan Kumaramangalam Medical College Hospital Salem \n" +  
 "Madras Regimental Centre \n" +  
 "Coimbatore Health Department \n" +  
 "Bharathiar University \n" +  
 "Tirunelveli Water Resources Department \n" +  
 "Indira Gandhi Centre for Atomic Research \n" +  
 "Kanniyakumari Child Welfare Committee \n" +  
 "Tamilnadu Hindu Religious and Charitable Endowments Department \n" +  
 "Andrew Yule and Company \n" +  
 "District Health Society Nagapattinam \n" +  
 "Greater Chennai Corporation \n\n Anna University\n" +  
 "IOCL\n" +  
 "Commissioner of GST & Central Excise\n" +  
 "District Health Society, Coimbatore\n" +  
 "District Health Services, Ariyalur\n" +  
 "District Court, Perambalur\n" +  
 "Principal Commissioner of Customs, Chennai\n" +  
 "District Health Society, Tiruvannamalai\n" +  
 "District Health Society, Tiruchirappalli\n" +  
 "District Health Society, Thanjavur\n" +  
 "District Health Society, Virudhunagar\n" +  
 "District Health Society, Salem\n" +  
 "District Health Society, Virudhunagar\n" +  
 "District Health Society, Ranipet\n" +  
 "District Health Society, Villupuram\n" +  
 "District Health Society, Ranipet\n" +  
 "District Health Society, Vellore\n" +  
 "District Health Society, Ramanathapuram\n" +  
 "District Health Society, Vellore\n" +  
 "District Health Society, Ramanathapuram\n" +  
 "District Health Society, Tiruvallur\n" +

"District Health Society, Pudukkottai\n" +  
 "District Health Society, Perambalur\n" +  
 "District Health Society, Tiruvallur\n" +  
 "District Health Society, Pudukkottai\n" +  
 "District Health Society, Nilgiris\n" +  
 "District Health Society, Tirunelveli\n" +  
 "District Health Society, Nilgiris\n" +  
 "District Health Society, Namakkal\n" +  
 "District Health Society, Thoothukudi\n" +  
 "District Health Society, Tiruppur\n" +  
 "District Health Society, Kanniyakumari\n" +  
 "District Health Society, Coimbatore\n" +  
 "District Health Society, Kancheepuram\n" +  
 "District Health Society, Krishnagiri\n" +  
 "District Health Society, Sivagangai\n" +  
 "District Health Society, Ariyalur\n" +  
 "District Health Society, Kallakurichi\n" +  
 "District Health Society, Karur\n" +  
 "District Health Society, Sivagangai\n" +  
 "District Health Society, Erode\n" +  
 "District Health Society, Salem\n" +  
 "District Health Society, Dindigul\n" +  
 "District Health Society, Dharmapuri\n" +  
 "District Health Society, Tiruvarur\n\n\n FOR UG CANDIDATE \n\n\n Anna  
 University Professional Assistant-I, III & Peon cum Mechanic \n" +  
 "Trade & Technician Apprentice \n" +  
 "Tax Assistant, Stenographer Grade II & Multi Tasking Staff, Havaladar\n" +  
 "District Quality Consultant, IT-Coordinator & Other \n" +  
 "Midlevel Health Care Provider & MPHWP (Male)/ Health Inspector Gr 2 \n" +  
 "Steno Typist, Typist \n" +  
 "Sukhami, Seamen, Greaser \n" +  
 "Mid Level Health Provider \n" +  
 "Multipurpose Health Worker (Male)/ Health Inspector Grade II \n" +  
 "Multipurpose Health Worker (Male)/ Health Inspector Grade II \n" +  
 "Mid Level Health Provider (MLHP) \n" +  
 "MPHW (Male) /Health Inspector Grade II, Mid Level Health Provider \n" +  
 "Multi Purpose Health Worker (Male)/ Health Inspector Gr. II \n" +  
 "Mid Level Health Care Provider (MLHP) \n" +  
 "Mid Level Health Provider \n" +  
 "Multipurpose Health Worker (Male) /Health Inspector Gr II \n" +  
 "Multi Purpose Health Worker (Male)/ Health Inspector Gr. II \n" +  
 "Mid Level Health Care Provider \n" +  
 "Multi Purpose Health Worker (Male)/ Health Inspector Gr. II \n" +  
 "Multipurpose Health Worker (Male) /Health Inspector Gr II \n" +  
 "Mid Level Health Provider \n" +  
 "Multipurpose Health Worker (Male) /Health Inspector Gr II \n" +  
 "Mid Level Health Provider \n" +  
 "Multi Purpose Health Worker (Male)/ Health Inspector Gr. II \n" +  
 "Mid Level Health Provider (MLHP) \n" +  
 "Multi Purpose Health Worker (Male)/ Health Inspector Gr. II – \n" +

"Multi Purpose Health Worker (Male)/ Health Inspector Gr. II – \n" +  
 "Multi Purpose Health Worker (Male)/ Health Inspector Gr. II – \n" +  
 "Mid Level Health Provider \n" +  
 "Mid Level Health Care Provider \n" +  
 "Multipurpose Health Worker (Male)/ Health Inspector Grade II \n" +  
 "Multi Purpose Health Worker (Male)/ Health Inspector Gr. II \n" +  
 "Mid Level Health Provider\n" +  
 "Mid Level Health Provider \n" +  
 "Mid Level Health Provider \n" +  
 "MLHP, MPHWP (Male) / Health Inspector \n" +  
 "MPHW (Male) /Health Inspector Grade II \n" +  
 "MPHW (Male) /Health Inspector Grade II \n" +  
 "MLHP, MPHWP (Male) / Health Inspector Gr II \n" +  
 "Mid Level Health Provider \n" +  
 "MPHW (Male) /Health Inspector Grade II, Mid Level Health Provider \n" +  
 "Mid Level Health Provider \n" +  
 "MLHP, MPHWP (Male) / Health Inspector \n" +  
 "Multi Purpose Health Worker (Male)/ Health Inspector Gr. II \n" +  
 "Mid Level Health Provider \n" +  
 "MPHW (Male) /Health Inspector Grade II \n" +  
 "Mid Level Health Provider \n" +  
 "Mid Level Health Provider \n" +  
 "Mid Level Health Provider \n" +  
 "MPHW (Male) /Health Inspector Grade II \n" +  
 "Mid Level Health Provider (MLHP) \n\n\n FOR PG CANDIDATE \n\n\n  
 Instructional Designer - work\n" +  
 "\n" +  
 "Manager Finance\n" +  
 "\n" +  
 "Research Assistant\n" +  
 "Project Associate II\n" +  
 "\n" +  
 "Research Assistant\n" +  
 "\n" +  
 "Senior Project Associate\n" +  
 "\n" +  
 "Senior Research Fellow\n" +  
 "\n" +  
 "\n" +  
 "Senior R & D Engineer\n" +  
 "\n" +  
 "Project Associate I\n" +  
 "\n" +  
 "Content and Administrative Manager\n" +  
 "\n" +  
 "Clinical Trial Specialist\n" +  
 "\n" +  
 "Post Doctoral Fellow\n" +  
 "\n" +  
 "Project Linked Junior Reserach Fellow\n" +

"\n" +  
 "Associate\n" +  
 "\n" +  
 "Project Assistant\n" +  
 "\n" +  
 "Programme Manager\n" +  
 "\n" +  
 "Program Officer\n" +  
 "\n" +  
 "\n" +  
 "Program Manager\n" +  
 "\n" +  
 "Program Officer\n" +  
 "\n" +  
 "Program Officer\n" +  
 "\n" +  
 "Program Manager\n" +  
 "\n" +  
 "Program Officer\n" +  
 "\n" +  
 "Program Manager\n" +  
 "\n" +  
 "Procurement Officer\n" +  
 "\n" +  
 "Technical Assistant\n" +  
 "\n" +  
 "Senior Research Fellow\n" +  
 "\n" +  
 "Scientist C\n" +  
 "\n" +  
 "Clinical Psychology Consultant\n" +  
 "\n" +  
 "Project Junior Software Developer\n" +  
 "\n" +  
 "Junior Research Fellow\n" +  
 "\n" +  
 "Junior Research Fellow/ Project Associate I\n" +  
 "\n" +  
 "Senior Research Fellow/ Project Associate II\n" +  
 "\n" +  
 "System Support Engineer Trainee\n" +  
 "\n" +  
 "Server and Network Administrator Trainee\n" +  
 "\n" +  
 "Junior Research Fellow IIT BHU\n" +  
 "\n" +  
 "Junior Research Fellow ICT\n" +  
 "\n" +  
 "Junior Research Fellow NIT Calicut\n" +  
 "\n" +

```

        "Project Technical Officer\n" +
        "\n" +
        "Senior Research Fellow\n" +
        "\n" +
        "TutorSpecialist\n" +
        "\n" +
        "Senior Resident\n" +
        "\n" +
        "Project Associate I\n" +
        "\n" +
        "Research Officer\n" +
        "\n" +
        "Research Assistant\n" +
        "\n" +
        "Marketing Officer, Admission Counsellor, More Vacancies\n" +
        "\n" +
        "Tutor/ Demonstrator\n" +
        "\n" +
        "Junior Research Fellow\n" +
        "\n" +
        "Assistant Professor\n" +
        "\n" +
        "Guest Relation Manager\n" +
        "\n" +
        "SEO Executive\n" +
        "\n" +
        "Senior Consultant\n" +
        "\n" +
        "Assistant Manager\n" +
        "\n" +
        "Video Editor\n");
String[] para=jobss.getText().toString().split("\n");
Toast.makeText(job.this,""+para.length,Toast.LENGTH_LONG).show();

next.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent(job.this, LASTPAGE.class);
        startActivity(intent);
    }
});
}
}

```

### **LAST PAGE.java**

```
package com.example.login;

import android.os.Bundle;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class LASTPAGE extends AppCompatActivity {
    TextView tq, wish;
    ImageView tku;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.lastpage);
        tq = (TextView) findViewById(R.id.tq);
        wish = (TextView) findViewById(R.id.wish);
        tku = (ImageView) findViewById(R.id.tku);
    }
}
```

### **adminlog.java**

```
package com.example.login;

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class adminlog extends AppCompatActivity {
    Button LogInButton;
    EditText Email, Password;
    String EmailHolder, PasswordHolder;
    Boolean EditTextEmptyHolder;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```



```

setContentView(R.layout.adminlog);
LogInButton = (Button) findViewById(R.id.buttonLogin);
Email = (EditText) findViewById(R.id.editEmail);
Password = (EditText) findViewById(R.id.editPassword);
LogInButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (CheckEditTextStatus()) {
            if (validation()) {
                course();
            }
        }
        else
        {
            Toast.makeText(adminlog.this, "Please Enter UserName or Password.",
Toast.LENGTH_LONG).show();
        }
    }
});
}

public boolean CheckEditTextStatus()
{
    // Getting value from All EditText and storing into String Variables.
    EmailHolder = Email.getText().toString();
    PasswordHolder = Password.getText().toString();
    // Checking EditText is empty or no using TextUtils.
    if (TextUtils.isEmpty(EmailHolder) || TextUtils.isEmpty(PasswordHolder))
    {
        EditTextEmptyHolder = false;
    }
    else
    {
        EditTextEmptyHolder = true;
    }
    return EditTextEmptyHolder;
}

public boolean validation() {
    String email = Email.getText().toString().trim();
    String emailPattern = "admin@gmail.com";
    String password = Password.getText().toString().trim();
    String passwordPattern = "admin";
    if (email.matches(emailPattern) && password.matches(passwordPattern))
    {
        Toast.makeText(getApplicationContext(), "Login Successfull",
Toast.LENGTH_SHORT).show();
        return true;
    }
    else {
        Toast.makeText(getApplicationContext(), "Please enter correct email address or
password", Toast.LENGTH_SHORT).show();
    }
}

```

```

        return false;
    }
}
public void course()
{
    Intent intent=new Intent(adminlog.this,view.class);
    startActivity(intent);
}
}

```

### **view.java**

```

package com.example.login;

import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import androidx.appcompat.app.AppCompatActivity;

public class view extends AppCompatActivity {
    Button ss,hss,ug,pg;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.view);
        ss=(Button)findViewById(R.id.sslc1);
        hss=(Button)findViewById(R.id.hss);
        ug=(Button)findViewById(R.id.ug);
        pg=(Button)findViewById(R.id.pg);
        ss.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(view.this,DisplaySslc.class);
                startActivity(intent);
            }
        });
        hss.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(view.this,HSS.class);
                startActivity(intent);
            }
        });
        ug.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(view.this,ugjob.class);
            }
        });
    }
}

```

```

        startActivity(intent);
    }
});
pg.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent=new Intent(view.this,pgjobs.class);
        startActivity(intent);
    }
});
}
}

```

### **DisplaySslc.java**

```

package com.example.login;

import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

public class DisplaySslc extends AppCompatActivity {
    Button Finish;
    TextView s,textView;
    Context context;
    SQLiteHelper myDB;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.sslcc);
        myDB =new SQLiteHelper(this);
        Finish = (Button) findViewById(R.id.btnsslcc);
        s = (TextView) findViewById(R.id.s);
        textView=(TextView)findViewById(R.id.sscltxt);
        display();
        Finish.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(DisplaySslc.this, view.class);
                startActivity(intent);
            }
        });
    }
}

```

```

    }
    });
}

private void display() {
    Cursor result = myDB.getAllDetail();
    while (result.moveToNext())
    {
        textView.append("\n\t"+result.getString(1));
        textView.append("\t\t"+result.getString(2));
        textView.append("\t\t"+result.getString(3));
        textView.append("\t\t"+result.getString(4));
        textView.append("\t\t"+result.getString(5));
        textView.append("\t\t"+result.getString(6)+"\n");
    }
}
}

```

## **HSS.java**

```

package com.example.login;

import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

public class HSS extends AppCompatActivity {
    Button Finish1;
    TextView h,textView1;
    Context context;
    SQLiteHelper myDBH;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.h);
        myDBH =new SQLiteHelper(this);
        Finish1 = (Button) findViewById(R.id.button4);
        h = (TextView) findViewById(R.id.h);
        textView1=(TextView)findViewById(R.id.hsstxt);
        display();
    }
}

```

```

        Finish1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(HSS.this, view.class);
                startActivity(intent);
            }
        });
    }
    private void display() {
        Cursor result = myDBH.AllDetail();
        while (result.moveToNext())
        {
            textView1.append("\n\t"+result.getString(1));
            textView1.append("\t\t"+result.getString(2));
            textView1.append("\t\t"+result.getString(3));
            textView1.append("\t\t"+result.getString(4));
            textView1.append("\t\t"+result.getString(5));
            textView1.append("\t\t"+result.getString(6)+"\n");
        }
    }
}

```

### **ugjob.java**

```
package com.example.login;
```

```

import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

```

```

public class ugjob extends AppCompatActivity {
    Button Finish3;
    TextView ugjob,textView2;
    Context context;
    SQLiteHelper myDB1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.ug);
        myDB1 =new SQLiteHelper(this);
        Finish3 = (Button) findViewById(R.id.but1);
        ugjob = (TextView) findViewById(R.id.ug1);
    }
}

```

```

        textView2=(TextView)findViewById(R.id.ugtxt);
        display();
        Finish3.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(ugjob.this, view.class);
                startActivity(intent);
            }
        });
    }
    private void display() {
        Cursor result = myDB1.ugDetail();
        while (result.moveToNext())
        {
            textView2.append("\n\t"+result.getString(1));
            textView2.append("\t\t"+result.getString(2));
            textView2.append("\t\t"+result.getString(3));
            textView2.append("\t\t"+result.getString(4));
            textView2.append("\t\t"+result.getString(5));
            textView2.append("\t\t"+result.getString(6)+"\n");
        }
    }
}

```

### **pgjobs.java**

```

package com.example.login;

import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ScrollView;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

public class pgjobs extends AppCompatActivity {
    Button Finishpg;
    TextView s1,textView3;
    Context context;
    SQLiteHelper myDBpg;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

```

```

super.onCreate(savedInstanceState);
setContentView(R.layout.pg);
Finishpg = (Button) findViewById(R.id.button);
s1 = (TextView) findViewById(R.id.pg1);
myDBpg = new SQLiteHelper(this);
textView3 = (TextView) findViewById(R.id.pgtxt);
display();
Finishpg.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent(pgjobs.this, view.class);
        startActivity(intent);
    }
});
}
private void display() {
    Cursor result = myDBpg.pgDetail();
    while (result.moveToNext())
    {
        textView3.append("\n\t"+result.getString(1));
        textView3.append("\t\t"+result.getString(2));
        textView3.append("\t\t"+result.getString(3));
        textView3.append("\t\t"+result.getString(4));
        textView3.append("\t\t"+result.getString(5));
        textView3.append("\t\t"+result.getString(6)+"\n");
    }
}
}

```

### **activity\_main.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:id="@+id/activity_main"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="#6C627C"
android:paddingLeft="32px"
android:paddingTop="32px"
android:paddingRight="32px"
android:paddingBottom="32px"
tools:context="com.example.login.MainActivity">

```

```
<TextView
    android:id="@+id/textView"
    android:layout_width="130dp"
    android:layout_height="48dp"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:gravity="center"
    android:text="Sign In"
    android:textColor="#03A9F4"
    android:textSize="20dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.459"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.023" />
```

```
<EditText
    android:id="@+id/editEmail"
    android:layout_width="379dp"
    android:layout_height="52dp"
    android:layout_below="@+id/textView"
    android:layout_centerHorizontal="true"
    android:ems="10"
    android:gravity="center"
    android:hint="Enter Email"
    android:inputType="textEmailAddress"
    android:textColor="@color/purple_200"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.135" />
```

```
<EditText
    android:id="@+id/editPassword"
    android:layout_width="339dp"
    android:layout_height="51dp"
    android:layout_below="@+id/editEmail"
    android:layout_centerHorizontal="true"
    android:ems="10"
    android:gravity="center"
    android:hint="Enter Password"
    android:inputType="textPassword"
    android:textColor="@color/purple_200"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.567"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
```



```

        app:layout_constraintVertical_bias="0.309" />

<Button
    android:id="@+id/buttonLogin"
    android:layout_width="138dp"
    android:layout_height="52dp"
    android:layout_below="@+id/editPassword"
    android:text="Login "
    app:backgroundTint="#1EE9C0"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<Button
    android:id="@+id/buttonRegister"
    android:layout_width="129dp"
    android:layout_height="54dp"
    android:layout_below="@+id/buttonLogin"
    android:text="Sign Up"
    app:backgroundTint="#E91E63"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.481"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.634" />

<Button
    android:id="@+id/adbtn"
    android:layout_width="132dp"
    android:layout_height="63dp"
    android:text="Admin"
    android:textSize="24sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.831"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.937" />

<ImageView
    android:id="@+id/imageView"
    android:layout_width="89dp"
    android:layout_height="73dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.289"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"

```

```

        app:layout_constraintVertical_bias="0.95"
        app:srcCompat="@drawable/user" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

### **activity\_register.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:id="@+id/activity_register"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#6A496C"
    android:paddingLeft="32dp"
    android:paddingTop="32dp"
    android:paddingRight="32dp"
    android:paddingBottom="32dp"
    tools:context="com.example.login.RegisterActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:gravity="center"
        android:text="Sign Up"
        android:textColor="#03A9F4"
        android:textSize="20sp" />

    <EditText
        android:id="@+id/editName"
        android:layout_width="358dp"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textView"
        android:layout_marginTop="27dp"
        android:ems="10"
        android:gravity="center"
        android:hint="Enter Name"
        android:inputType="textEmailAddress"
        android:paddingLeft="32dp"
        android:paddingTop="32dp"
        android:paddingRight="32dp"

```

```
android:paddingBottom="32dp"
android:textColor="#F6095B" />
```

```
<EditText
    android:id="@+id/editEmail"
    android:layout_width="337dp"
    android:layout_height="69dp"
    android:layout_below="@+id/editName"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="20dp"
    android:ems="10"
    android:gravity="center"
    android:hint="Enter Email"
    android:inputType="textEmailAddress"
    android:textColor="#F6075B" />
```

```
<EditText
    android:id="@+id/editPassword"
    android:layout_width="349dp"
    android:layout_height="65dp"
    android:layout_below="@+id/editEmail"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="20dp"
    android:ems="10"
    android:gravity="center"
    android:hint="Enter Password"
    android:inputType="textPassword"
    android:textColor="#F30659" />
```

```
<Button
    android:id="@+id/buttonRegister"
    android:layout_width="341dp"
    android:layout_height="66dp"
    android:layout_below="@+id/editPassword"
    android:layout_marginTop="64dp"
    android:text="Register Here "
    android:textSize="24sp"
    app:backgroundTint="#E91E63" />
```

```
<Button
    android:id="@+id/logbtn"
    android:layout_width="288dp"
    android:layout_height="83dp"
    android:layout_below="@id/buttonRegister"
    android:layout_alignLeft="@id/buttonRegister"
    android:layout_alignRight="@id/buttonRegister"
    android:layout_alignParentLeft="false"
    android:layout_marginLeft="15dp"
    android:layout_marginTop="28dp"
    android:layout_marginRight="38dp"
```

```
    android:background="#C51162"
    android:gravity="center"
    android:text="LOGIN"
    android:textSize="24sp" />
```

```
</RelativeLayout>
```

### **activity\_dashboard.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:id="@+id/activity_dashboard"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="32px"
    android:paddingLeft="32px"
    android:paddingRight="32px"
    android:paddingTop="32px"
    tools:context="com.example.login.DashboardActivity">
```

```
<TextView
    android:id="@+id/textView1"
    android:layout_width="370dp"
    android:layout_height="183dp"
    android:layout_alignParentTop="true"
    android:gravity="center|top"
    android:text=" Login Successful, "
    android:textColor="#39DCA8"
    android:textSize="30dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.301"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.076" />
```

```
<Button
    android:id="@+id/button1"
    android:layout_width="170dp"
    android:layout_height="61dp"
    android:layout_below="@+id/textView1"
    android:layout_alignLeft="@id/textView1"
    android:background="#121111"
    android:clickable="true"
```

```

        android:cursorVisible="true"
        android:text=" LOGOUT"
        android:textSize="34sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.481"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.677" />

```

```

<Button
    android:id="@+id/button2"
    android:layout_width="179dp"
    android:layout_height="59dp"
    android:layout_alignLeft="@id/textView1"
    android:layout_alignBottom="@id/textView1"
    android:layout_alignParentLeft="false"
    android:layout_alignParentRight="true"
    android:background="#121010"
    android:clickable="true"
    android:cursorVisible="true"
    android:text="NEXT"
    android:textSize="34sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.534" />

```

```

</androidx.constraintlayout.widget.ConstraintLayout>

```

### **activity form1.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:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#40383B"
    android:paddingLeft="32dp"
    android:paddingTop="32dp"
    android:paddingRight="32dp"
    tools:context="com.example.login.Form">

```

```

<TextView
    android:id="@+id/textView"
    android:layout_width="309dp"
    android:layout_height="39dp"
    android:layout_alignParentTop="true"
    android:gravity="left"
    android:text="Candidate Registration form"
    android:textColor="#03A9F4"
    android:textSize="22sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.604"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.0" />

```

```

<TextView
    android:id="@+id/textView3"
    android:layout_width="113dp"
    android:layout_height="53dp"
    android:layout_below="@+id/textView"
    android:text="Name:"
    android:textSize="24dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.02"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.117" />

```

```

<EditText
    android:id="@+id/edittxt1"
    android:layout_width="199dp"
    android:layout_height="46dp"
    android:layout_below="@id/textView3"
    android:layout_alignLeft="@id/textView3"
    android:layout_alignRight="@id/textView3"
    android:layout_alignParentLeft="false"
    android:layout_alignParentRight="true"
    android:textSize="24dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.908"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.116" />

```

```

<TextView
    android:id="@+id/textView5"
    android:layout_width="112dp"

```

```

android:layout_height="38dp"
android:layout_below="@+id/textView"
android:text="Gender:"
android:textSize="24dp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.016"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.248" />

```

```

<EditText
    android:id="@+id/gender"
    android:layout_width="197dp"
    android:layout_height="56dp"
    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="0.238" />

```

```

<TextView
    android:id="@+id/textView6"
    android:layout_width="128dp"
    android:layout_height="64dp"
    android:layout_below="@id/textView5"
    android:layout_alignLeft="@id/textView5"
    android:layout_alignTop="@id/textView5"
    android:layout_alignBottom="@id/textView5"
    android:layout_alignParentLeft="true"
    android:layout_alignParentBottom="false"
    android:text="Marital Status : "
    android:textSize="24dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.362" />

```

```

<EditText
    android:id="@+id/marri"
    android:layout_width="192dp"
    android:layout_height="54dp"
    android:layout_above="@id/textView7"
    android:layout_alignParentRight="true"
    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="0.361" />
```

```
<TextView
    android:id="@+id/textView12"
    android:layout_width="103dp"
    android:layout_height="60dp"
    android:layout_alignTop="@id/textView6"
    android:layout_alignRight="@id/textView6"
    android:text="Aadhaar Number : "
    android:textSize="24dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.485" />
```

```
<EditText
    android:id="@+id/edittxt4"
    android:layout_width="196dp"
    android:layout_height="57dp"
    android:layout_alignTop="@id/textView6"
    android:layout_toLeftOf="@id/textView6"
    android:digits="1234567890"
    android:ems="10"
    android:hint="xxxx xxxx xxxx"
    android:inputType="number"
    android:maxLength="12"
    android:phoneNumber="true"
    android:textSize="24dp"
    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="0.475" />
```

```
<TextView
    android:id="@+id/textView13"
    android:layout_width="93dp"
    android:layout_height="65dp"
    android:layout_alignTop="@id/textView6"
    android:layout_toStartOf="@+id/edittxt5"
    android:layout_toLeftOf="@+id/edittxt5"
    android:text="Father's Name : "
    android:textSize="24dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
```



```
app:layout_constraintHorizontal_bias="0.0"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.603" />
```

```
<EditText
    android:id="@+id/edittxt5"
    android:layout_width="197dp"
    android:layout_height="53dp"
    android:layout_above="@id/textView6"
    android:layout_toRightOf="@id/textView6"
    android:text="Mr."
    android:textSize="24dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.942"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.592" />
```

```
<TextView
    android:id="@+id/textView14"
    android:layout_width="94dp"
    android:layout_height="34dp"
    android:layout_alignTop="@id/textView6"
    android:layout_alignParentTop="true"
    android:text="DOB : "
    android:textSize="24dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.05"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.687" />
```

```
<EditText
    android:id="@+id/editTextDate"
    android:layout_width="197dp"
    android:layout_height="51dp"
    android:layout_alignRight="@id/textView6"
    android:layout_alignBottom="@id/textView6"
    android:layout_alignParentLeft="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentBottom="true"
    android:ems="10"
    android:hint="MM/DD/YY"
    android:inputType="date"
    android:maxLength="8"
    android:textSize="24dp"
    app:layout_constraintBottom_toBottomOf="parent"
```

```

app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.873"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.688" />

```

```

<TextView
    android:id="@+id/textView7"
    android:layout_width="134dp"
    android:layout_height="43dp"
    android:layout_alignBottom="@id/editTextDate"
    android:layout_toLeftOf="@id/editTextDate"
    android:text="Mobile No. : "
    android:textSize="24dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.779" />

```

```

<EditText
    android:id="@+id/editTextNumber"
    android:layout_width="176dp"
    android:layout_height="47dp"
    android:layout_alignTop="@id/textView7"
    android:layout_alignRight="@id/textView7"
    android:layout_toRightOf="@id/textView7"
    android:digits="0123456789"
    android:ems="10"
    android:inputType="number"
    android:maxLength="10"
    android:phoneNumber="true"
    android:textSize="24dp"
    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="0.785" />

```

```

<Button
    android:id="@+id/nxt"
    android:layout_width="156dp"
    android:layout_height="51dp"
    android:layout_below="@id/editTextNumber"
    android:layout_alignRight="@id/editTextNumber"
    android:background="#FFF9C4"
    android:text="Next"
    android:textColor="#1009DF"

```

```

        android:textColorHighlight="#FF5722"
        android:textColorHint="#FFEB3B"
        android:textSize="30dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.954" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

### **content form2.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="#FFF9C4">

    <TextView
        android:id="@+id/textView4"
        android:layout_width="331dp"
        android:layout_height="62dp"
        android:layout_alignParentTop="true"
        android:layout_alignParentRight="true"
        android:layout_centerInParent="false"
        android:background="#F6C3D5"
        android:gravity="center"
        android:shadowColor="#F30757"
        android:text="QUALIFICATION"
        android:textColor="#2E2121"
        android:textSize="36sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.023" />

    <EditText
        android:id="@+id/nnnn"
        android:layout_width="299dp"
        android:layout_height="59dp"
        android:gravity="center"
        android:hint=" NAME"

```

```

android:textColor="#100F0F"
android:textColorHint="#0C0C0C"
android:textSize="24sp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.163" />

```

```

<EditText
    android:id="@+id/edittxt1"
    android:layout_width="307dp"
    android:layout_height="50dp"
    android:layout_alignLeft="@id/textView4"
    android:layout_alignParentBottom="true"
    android:background="#F0F4C3"
    android:gravity="center"
    android:hint="Name of degree"
    android:textColor="#090909"
    android:textColorHint="#100F0F"
    android:textSize="24sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.384"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.411" />

```

```

<EditText
    android:id="@+id/edittxt2"
    android:layout_width="338dp"
    android:layout_height="44dp"
    android:background="#F0F4C3"
    android:gravity="center"
    android:hint="Name of college/university"
    android:textColor="#0B0A0A"
    android:textColorHint="#0C0B0B"
    android:textSize="24sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.547"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.528" />

```

```

<EditText
    android:id="@+id/edittxt3"
    android:layout_width="197dp"
    android:layout_height="45dp"
    android:background="#F0F4C3"

```

```

        android:gravity="center"
        android:hint="Percentage"
        android:maxLength="3"
        android:numeric="decimal"
        android:textColor="#131313"
        android:textColorHint="#100F0F"
        android:textSize="24sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.644" />

```

```

<EditText
    android:id="@+id/qual"
    android:layout_width="288dp"
    android:layout_height="56dp"
    android:background="#F9FBE7"
    android:gravity="center"
    android:hint=" SSLC/HSS/UG/PG"
    android:textColor="#130F0F"
    android:textColorHint="#040303"
    android:textSize="24sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.503"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.28" />

```

```

<EditText
    android:id="@+id/phone"
    android:layout_width="241dp"
    android:layout_height="53dp"
    android:gravity="center"
    android:hint="PHONE NUMBER"
    android:inputType="number"
    android:maxLength="10"
    android:textColor="#191616"
    android:textColorHint="#131313"
    android:textSize="24sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.78" />

```

```

<Button
    android:id="@+id/nextbtn"
    android:layout_width="151dp"

```

```

        android:layout_height="57dp"
        android:background="#4FC3F7"
        android:text="NEXT"
        android:textColor="#F4ECEC"
        android:textSize="30sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.921" />

```

```

</androidx.constraintlayout.widget.ConstraintLayout>

```

### **job.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:id="@+id/scrollview"
android:layout_width="match_parent"
android:layout_height="match_parent">

```

```

    <TextView
        android:id="@+id/textView11"
        android:layout_width="375dp"
        android:layout_height="66dp"
        android:gravity="center"
        android:text="TNER Welcomes You "
        android:textColor="#F40CA7"
        android:textSize="36sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.444"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.024" />

```

```

    <Button
        android:id="@+id/button3"
        android:layout_width="162dp"
        android:layout_height="62dp"
        android:background="#FFCCBC"
        android:text="Next"
        android:textColor="#1B1919"
        android:textColorHint="#E1BEE7"
        android:textSize="36sp"

```

```
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.497"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.976" />
```

```
<TextView
    android:id="@+id/avail"
    android:layout_width="421dp"
    android:layout_height="45dp"
    android:gravity="center"
    android:text="The Availables Jobs are..."
    android:textSize="24sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.444"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.138" />
```

```
<ScrollView
    android:layout_width="365dp"
    android:layout_height="433dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.565"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.64">
```

```
    <TextView
        android:id="@+id/jobss"
        android:layout_width="312dp"
        android:layout_height="412dp"
        android:scrollbarAlwaysDrawHorizontalTrack="true"
        android:scrollbarStyle="insideInset"
        android:scrollbars="horizontal"
        android:scrollHorizontally="true"
        android:textSize="18sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.494"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.617" />
```

```
</ScrollView>
```

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

## lastpage.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="#FFFFFF">

<ImageView
    android:id="@+id/tku"
    android:layout_width="345dp"
    android:layout_height="200dp"
    android:background="@color/white"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.051"
    app:srcCompat="@drawable/whatsapp_image_2021_12_13_at_9_13_11_pm" />

<TextView
    android:id="@+id/tq"
    android:layout_width="348dp"
    android:layout_height="167dp"
    android:background="#A0F68D"
    android:gravity="center"
    android:shadowColor="#4DB6AC"
    android:text="THE ONLY WAY TO DO GREAT WORK IS TO LOVE WHAT
YOU DO ...."
    android:textColor="#D50000"
    android:textColorHighlight="#AA00FF"
    android:textColorHint="#F3B2B2"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.494"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.499" />

<TextView
    android:id="@+id/wish"
    android:layout_width="341dp"
    android:layout_height="84dp"
    android:gravity="center"
    android:text="WISH YOU ALL SUCCESS !!!"
    android:textColor="#F8125B"
```



```

        android:textSize="30sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.93" />

```

```

</androidx.constraintlayout.widget.ConstraintLayout>

```

## **adminlog.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:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#6C627C"
    android:paddingLeft="32px"
    android:paddingTop="32px"
    android:paddingRight="32px"
    android:paddingBottom="32px"
    tools:context="com.example.login.MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="130dp"
        android:layout_height="48dp"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:gravity="center"
        android:text="Sign In"
        android:textColor="#03A9F4"
        android:textSize="20dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.459"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.023" />

    <EditText
        android:id="@+id/editEmail"
        android:layout_width="379dp"
        android:layout_height="52dp"
        android:layout_below="@+id/textView"
        android:layout_centerHorizontal="true"

```

```

android:ems="10"
android:gravity="center"
android:hint="Enter Email"
android:inputType="textEmailAddress"
android:textColor="@color/purple_200"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.135" />

```

<EditText

```

    android:id="@+id/editPassword"
    android:layout_width="339dp"
    android:layout_height="51dp"
    android:layout_below="@+id/editEmail"
    android:layout_centerHorizontal="true"
    android:ems="10"
    android:gravity="center"
    android:hint="Enter Password"
    android:inputType="textPassword"
    android:textColor="@color/purple_200"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.567"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.309" />

```

<Button

```

    android:id="@+id/buttonLogin"
    android:layout_width="138dp"
    android:layout_height="52dp"
    android:layout_below="@+id/editPassword"
    android:text="Login "
    app:backgroundTint="#1EE9C0"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.498"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.768" />

```

</androidx.constraintlayout.widget.ConstraintLayout>

## **view.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:id="@+id/cont"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="#E8EAF6">
```

```
<Button
    android:id="@+id/sslc1"
    android:layout_width="168dp"
    android:layout_height="87dp"
    android:text="SSLC"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.172" />
```

```
<Button
    android:id="@+id/hss"
    android:layout_width="169dp"
    android:layout_height="78dp"
    android:text="HSS"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.52"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.379" />
```

```
<Button
    android:id="@+id/ug"
    android:layout_width="170dp"
    android:layout_height="85dp"
    android:text="UG"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.506"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.589" />
```

```
<Button
    android:id="@+id/pg"
    android:layout_width="174dp"
    android:layout_height="81dp"
    android:text="PG"
```

```

        android:textSize="30sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.791" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

## **sslcc.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">

    <TextView
        android:id="@+id/s"
        android:layout_width="305dp"
        android:layout_height="46dp"
        android:background="#FFCDD2"
        android:gravity="center"
        android:text="SSLC CANDIDATE"
        android:textColor="#130303"
        android:textSize="30sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.647"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.037" />

    <HorizontalScrollView
        android:layout_width="340dp"
        android:layout_height="467dp"
        android:padding="10px"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.595">

        <ScrollView
            android:id="@+id/scrollView4"
            android:layout_width="386dp"

```

```

        android:layout_height="452dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.64"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.584">

        <TextView
            android:id="@+id/sscltxt"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:textSize="18sp" />
    </ScrollView>
</HorizontalScrollView>

<Button
    android:id="@+id/btnssl"
    android:layout_width="196dp"
    android:layout_height="44dp"
    android:background="#FFFFFF"
    android:text="FINISH"
    android:textColor="#512DA8"
    android:textColorHint="#FFFFFF"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.605"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.975" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

## **h.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">

```

```
<TextView
    android:id="@+id/h"
    android:layout_width="322dp"
    android:layout_height="53dp"
    android:background="#FFCDD2"
    android:gravity="center"
    android:text="HSS CANDIDATE"
    android:textColor="#221C1C"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.023" />
```

```
<HorizontalScrollView
    android:layout_width="343dp"
    android:layout_height="484dp"
    android:padding="10px"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.595">
```

```
<ScrollView
    android:id="@+id/scrollView2"
    android:layout_width="407dp"
    android:layout_height="475dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent">
```

```

<TextView
    android:id="@+id/hsstxt"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:textSize="18sp" />
</ScrollView>
</HorizontalScrollView>

<Button
    android:id="@+id/button4"
    android:layout_width="165dp"
    android:layout_height="52dp"
    android:background="#FFFFFF"
    android:text="FINISH"
    android:textColor="#512DA8"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.458"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.975" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

### **ug.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">

```

```

<TextView
    android:id="@+id/ug1"
    android:layout_width="264dp"
    android:layout_height="46dp"
    android:background="#FFCDD2"
    android:gravity="center"
    android:text="UG CANDIDATE"
    android:textColor="#200202"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.573"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.045" />
<HorizontalScrollView
    android:layout_width="343dp"
    android:layout_height="484dp"
    android:padding="10px"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.595">

<ScrollView
    android:id="@+id/scrollView5"
    android:layout_width="418dp"
    android:layout_height="476dp"
    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="0.498">

    <TextView
        android:id="@+id/ugtxt"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:textSize="18sp" />
</ScrollView>
</HorizontalScrollView>

<Button
    android:id="@+id/but1"
    android:layout_width="199dp"
    android:layout_height="52dp"

```



```

        android:background="#FFFFFF"
        android:text="FINISH"
        android:textColor="#512DA8"
        android:textSize="30sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.56"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.975" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

### **pg.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:id="@+id/con"
android:layout_width="match_parent"
android:layout_height="match_parent">

    <TextView
        android:id="@+id/pg1"
        android:layout_width="321dp"
        android:layout_height="57dp"
        android:background="#FFCDD2"
        android:gravity="center"
        android:text="PG CANDIDATE"
        android:textColor="#170B0B"
        android:textSize="30sp"
        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="0.023" />

    <HorizontalScrollView
        android:layout_width="343dp"
        android:layout_height="484dp"
        android:padding="10px"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"

```

```
app:layout_constraintVertical_bias="0.595">
```

```
<ScrollView
    android:id="@+id/scrollView3"
    android:layout_width="411dp"
    android:layout_height="499dp"
    android:scrollbarAlwaysDrawHorizontalTrack="true"
    android:scrollbarAlwaysDrawVerticalTrack="true"
    android:scrollbarStyle="outsideInset"
    android:scrollbars="horizontal|vertical"
    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="0.435">
```

```
    <TextView
        android:id="@+id/pgtxt"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:textSize="18sp" />
```

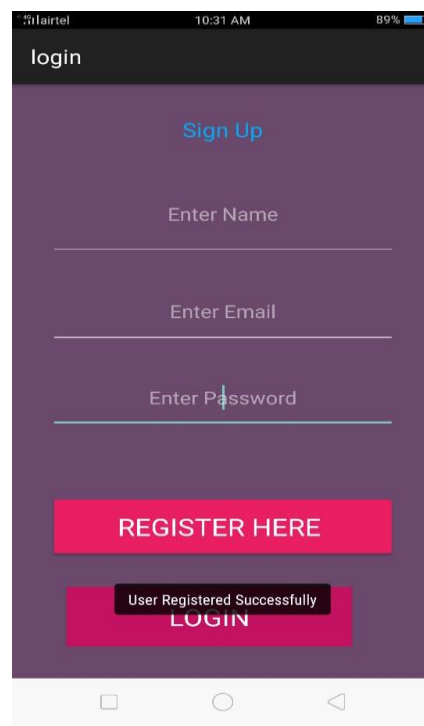
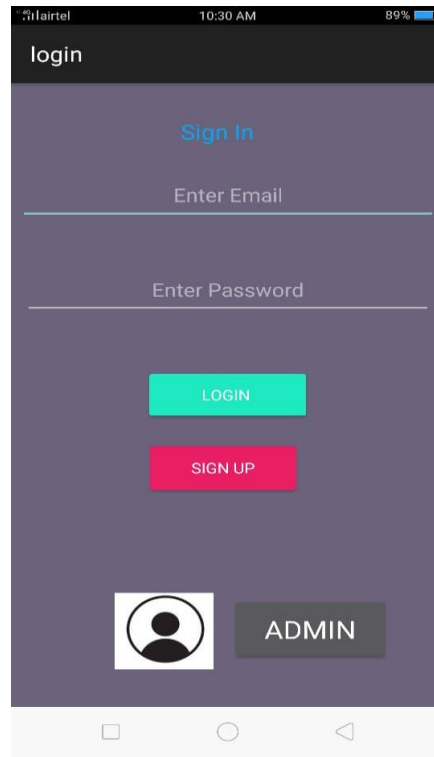
```
</ScrollView>
```

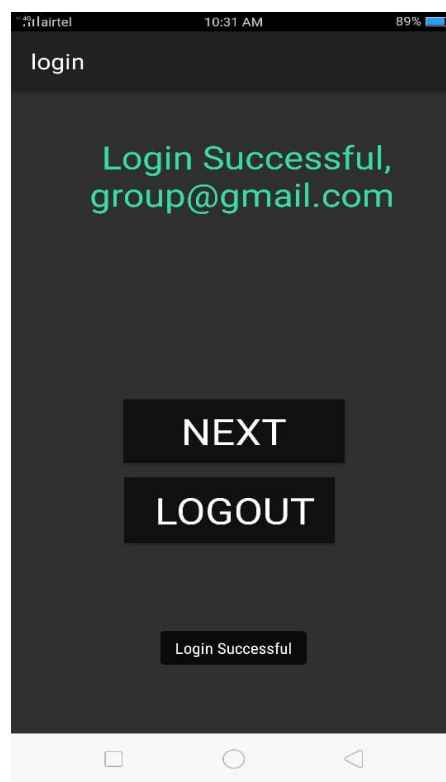
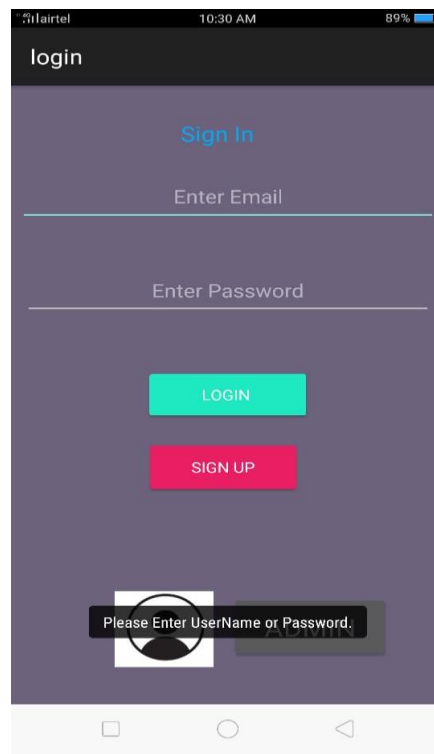
```
</HorizontalScrollView>
```

```
<Button
    android:id="@+id/button"
    android:layout_width="187dp"
    android:layout_height="44dp"
    android:background="#FBF5F5"
    android:text="FINISH"
    android:textColor="#512DA8"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.975" />
```

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

## 10.2 SAMPLE SCREENSHOTS





login

Candidate Registration form

Name: Group

Gender: female

Marital Status : unmarried

Aadhaar Number : 120876433929

Father's Name : Mr.sekar

DOB : 03/17/00

Mobile No. 9944506258

NEXT

login

QUALIFICATION

Group

PG

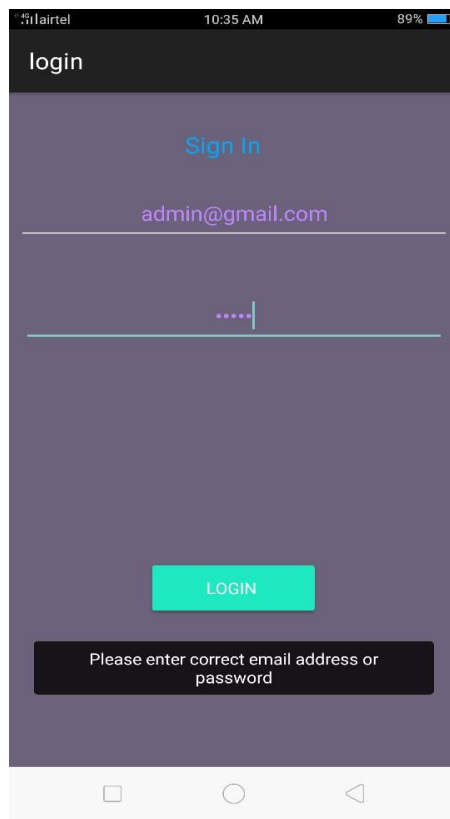
MCA

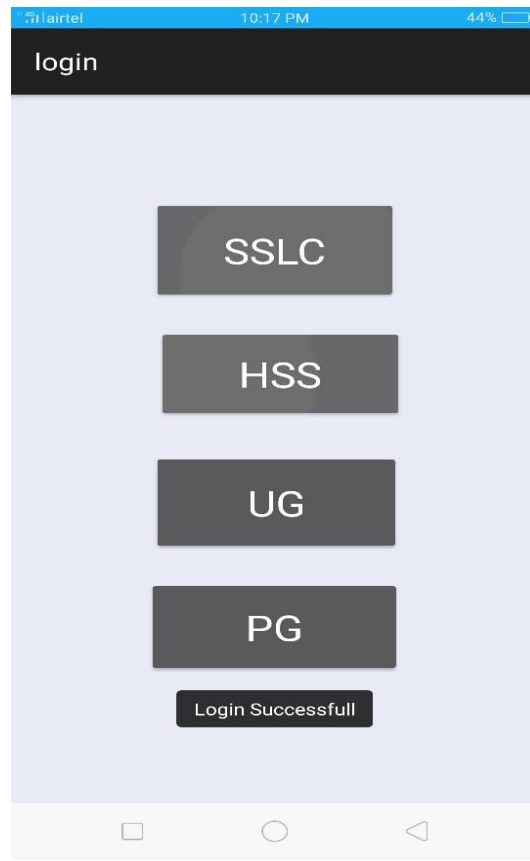
kamaraj College

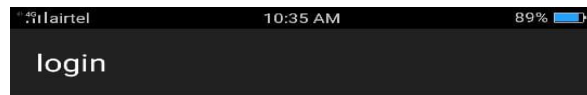
86

9944506258

NEXT







THE ONLY WAY TO DO  
GREAT WORK IS TO  
LOVE WHAT YOU DO ....

WISH YOU ALL  
SUCCESS !!!





### **10.3 BIBLIOGRAPHY**

- 1) **Tamilan guide website:** <https://www.tamilanguide.in /2016/11/latest-govt-jobs-2017-2018-182340.html>
- 2) **Employment registration website:** <https://www.tn.gov.in /service/dept/92579/1597>
- 3) **Android studio book:** Head First Android Development(dawn Griffiths) Starting with Android(Dr.M.M.Sharma).
- 4) <https://www.geeksforgeeks.org/how-to-read-data-from-sqlite-database-in-android/>