DECLARATION

We hereby declare that the project work entitled “Online School Management System” is a record of original work done by us under the guidance of Mr. Subhas Chandra Nath.

NAMES:

Sarad Bansal (32201217019) (Team Leader)

Amulya Abhinandan (32201217057)

Harsh Pathak (32201217046)

Anjali Kumari Supan (322012170055)

Date:

ACKNOWLEDGEMENT

We deem it pleasure to acknowledge our sense of gratitude to our project guide **Mr. Subhas Chandra Nath** under whom we've carried out our project work. His incisive, objective guidance and timely advice encouraged us with constant flow of energy to continue our work.

We wish to reciprocate in full measure the kindness shown by **Dr. Dibyendu Kumar Pal (HOD of Computer Application)** who inspired us with his valuable suggestions in successfully completing the project work.

We shall remain grateful to **Dr. Partha Pratim Bhattacharya (Principal of Asansol Engineering College)** for providing us a strong academic atmosphere by enforcing strict discipline to do the project work with utmost dedication and concentration.

Finally, we must say that no height is ever achieved without some sacrifices made at some end and it is here where we owe our special debt to our family and friends for helping us throughout the entire period of time.

CONTENTS

ABSTRACT

In the existing Teaching Management System, every college department follows the portal procedure in which faculty have to use desktop, which is time taking process.

This module is single **ONLINE SCHOOL MANAGEMENT SYSTEM (GuruKul)** is the application which is developed while keeping the lockdown and Covid-19 in mind. It is the app which will help the students and teachers to do their regular day to day study routine to go with the twist of not being present at the class physically but by the attending the class online from the comfort of their home and following the rules of lockdown and social distancing. There are many features of uploading and removing the video lectures and other study materials.

This application can be used by the students and teachers even after the Covid-19 is over. The project's main idea was to make a platform with the help of two different applications and which are connected with the same database, the database will be maintained by faculties.

INTRODUCTION

A school management system is a large database system which can be used for managing school's daily work. It is configurable and can be configured to meet most individual school's needs. It is a multi-user system. Generally speaking, it is platform available for running on a Local Area Network (LAN).It is used for handling all teachers and students details, also has information about student and teacher details, personal details and other details. Here all the work is computerized.

School Management Application helps overcome boring paper work in school. This Application is very easy to use. It can work accurately and very smoothly in different scenario. It reduces effort work load and increase efficiency in work. In aspects of time value, it is worthy. In this Application teacher can manage student's grade sheet, update their marks, managing online leave application, students and teacher's registration. Administrative Staff can add or remove teacher and student from database via this application. They also can register themselves.

OBJECTIVES

Main Objective :

Build an application to support the school activities in the lockdown.

Sub Objective :

- Track student schedules
- Hold records of the student's assignments
- Enrolling new students
- Help parents to follow up to his son's performance in school.
- Help the teacher to facilitate access to parents and introduce them to the extent of deficiencies at the student process.

Importance of the Project:

The school as institution to provide students, children and young person's to science and education, and is a building where students learn reading, writing, mathematics, science and various other studies, what are the features of the school and its goals and where are the various teaching centres? What are the facilities and equipment that requires availability in the school? We must therefore provide students with the electronic knowledge, because the world is with electronic features and became dependent on technology dramatically and through the presence of the Web site of the school makes it easier for students to deal with websites and opens a large field of learning and enter this world full of information and useful things that make it benefit significantly and it also

helps to understand the field of development through research and discovery of new information in the materials that can also cultures through the discovery of new methods of learning and teaching methods. In Gaza, the student must have a knowledge of technology and science, which has become the easy way to reach the outside world in order to raise the educational level of the student and also to send a message to all the world that the Palestinian student can keep up with the electronic world quickly despite the wars that are exposed and that he could be scientifically advanced in the world.

Limitation of the Project:

- Online student feedback is limited.
- E-Learning requires strong time managements skills.
- Cheating prevention during assignments is difficult.
- It lacks face to face communication.

HARDWARE AND SOFTWARE SPECIFICATIONS FOR RUNNING THE APPLICATION

Hardware Requirements :

Device to run the Application, Minimum 1GB of RAM, Internet Connectivity, 50mb of internal storage availability of storage.

Software Requirements :

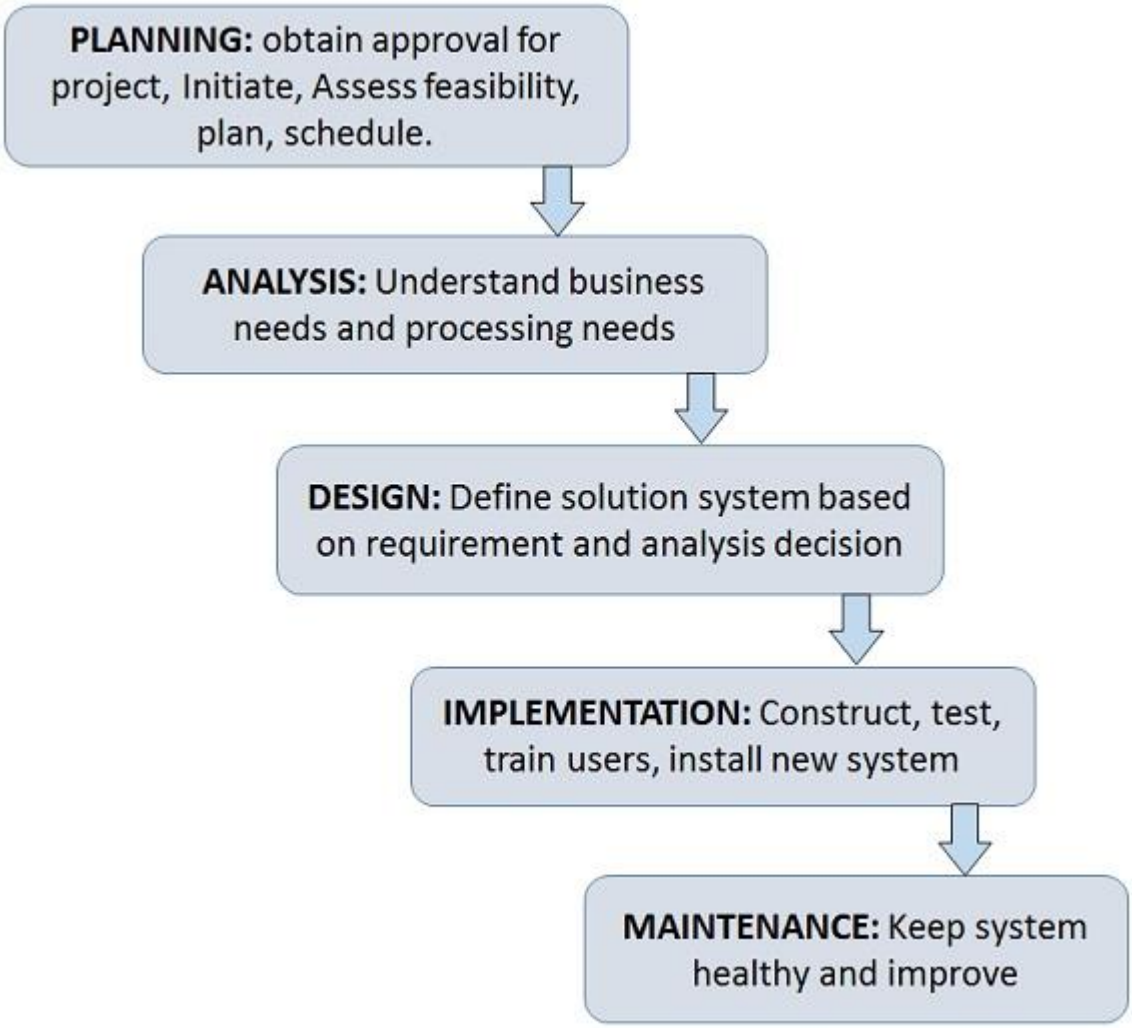
Android Device, Android Version 5.2 minimum or above.

SDLC

An effective System Development Life Cycle (SDLC) should result in a high-quality system that meets customer expectations, reaches completion within time and cost evaluations, and works effectively and efficiently in the current and planned Information Technology infrastructure.

System Development Life Cycle (SDLC) is a conceptual model which includes policies and procedures for developing or altering systems throughout their life cycles.

PHASES OF SDLC



Feasibility Study or Planning

- Define the problem and scope of existing system.
- A feasibility report for the entire project is created at the end of this phase.

Analysis and Specification

- Gather, analyse, and validate the information.
- A Software Requirement Specification (SRS) document, which specifies the software, hardware, functional, and network requirements of the system is prepared at the end of this phase.

System Design

- Includes the design of application, network, databases, user interfaces, and system interfaces.
- Finally, prepare a design document which will be used during next phases.

Implementation

- Implement the design into source code through coding.
- Integrate the information system into its environment and install the new system.

Maintenance/Support

- Include all the activities such as phone support or physical on-site support for users that is required once the system is installing.
- Maintenance and support may be needed for a longer time for large systems and for a short time.

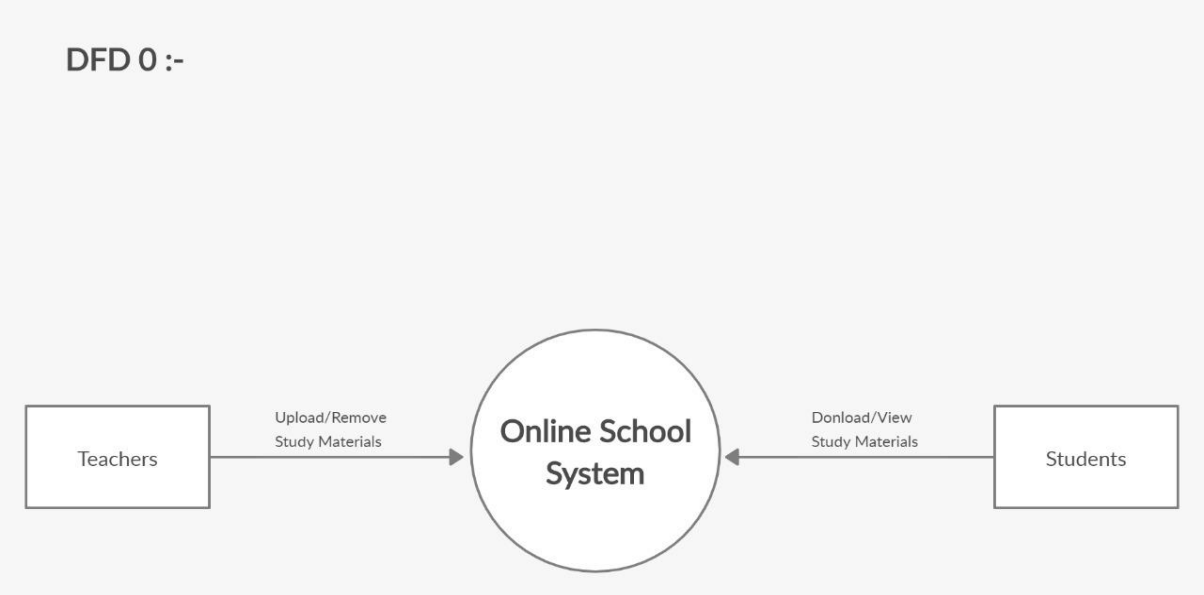
DATA FLOW DIAGRAM

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. That's why DFDs remain so popular after all these years. While they work well for data flow software and systems, they are less applicable nowadays to visualizing interactive, real-time or database-oriented software or systems.

Data flow diagram levels

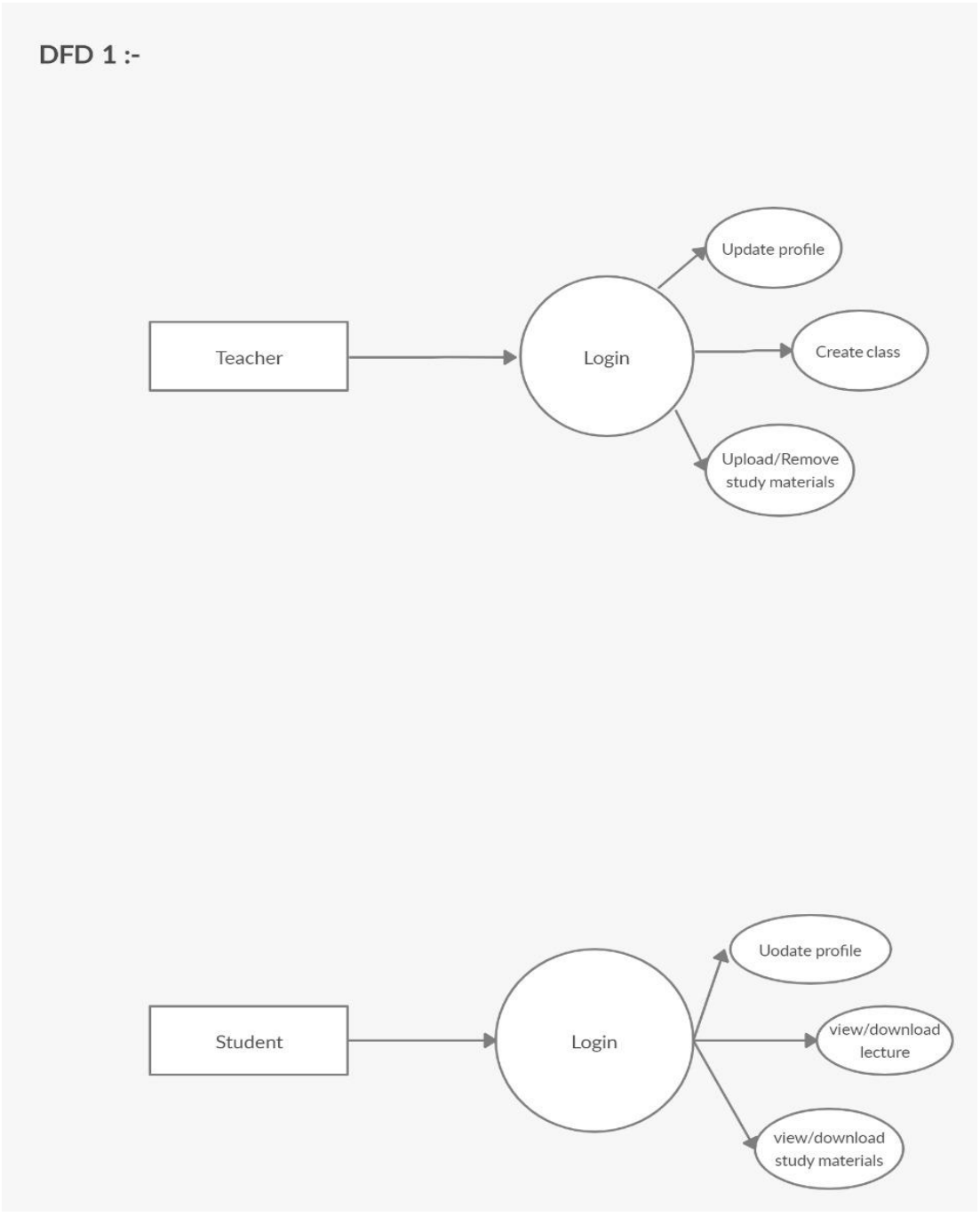
Data flow diagrams are also categorized by level. Starting with the most basic, level 0, DFDs get increasingly complex as the level increases.

Level 0 DFDs, also known as context diagrams, are the most basic data flow diagrams. They provide a broad view that is easily digestible but offers little detail. Level 0 data flow diagrams show a single process node and its connections to external entities.



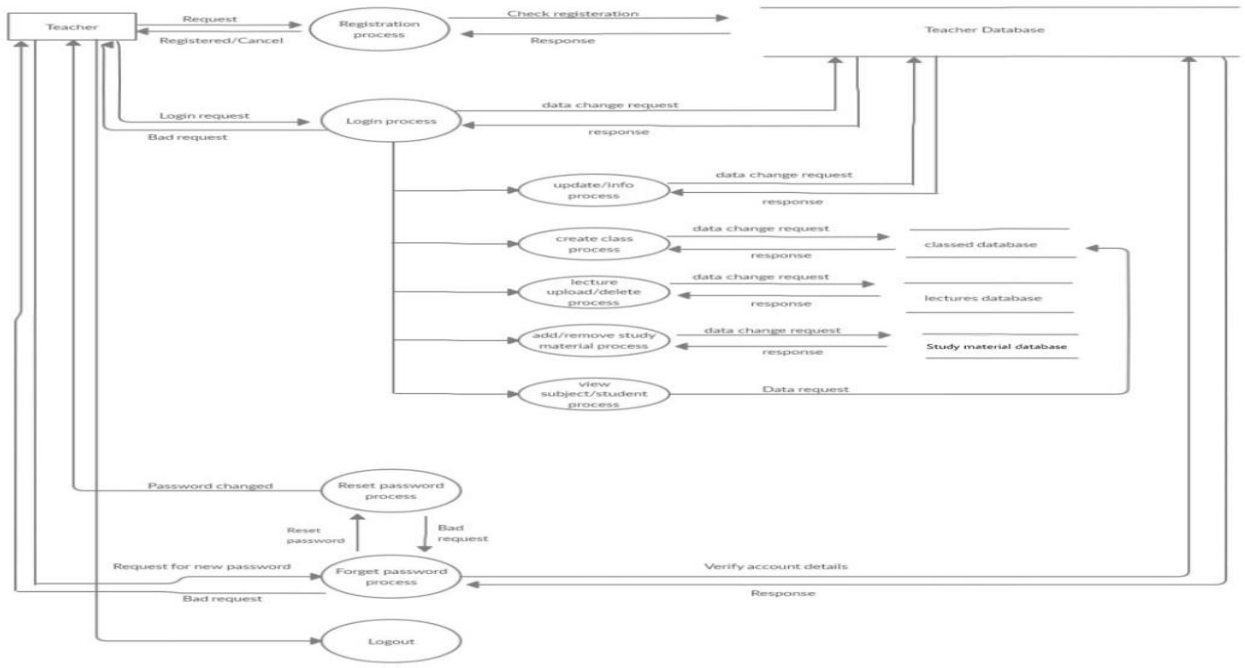
Level 1 DFDs are still a general overview, but they go into more detail than a context diagram. In a level 1 data flow diagram, the single process node from the context diagram is broken down into sub processes. As these processes are added, the diagram will need additional data flows and data stores to link them together.

DFD 1 :-

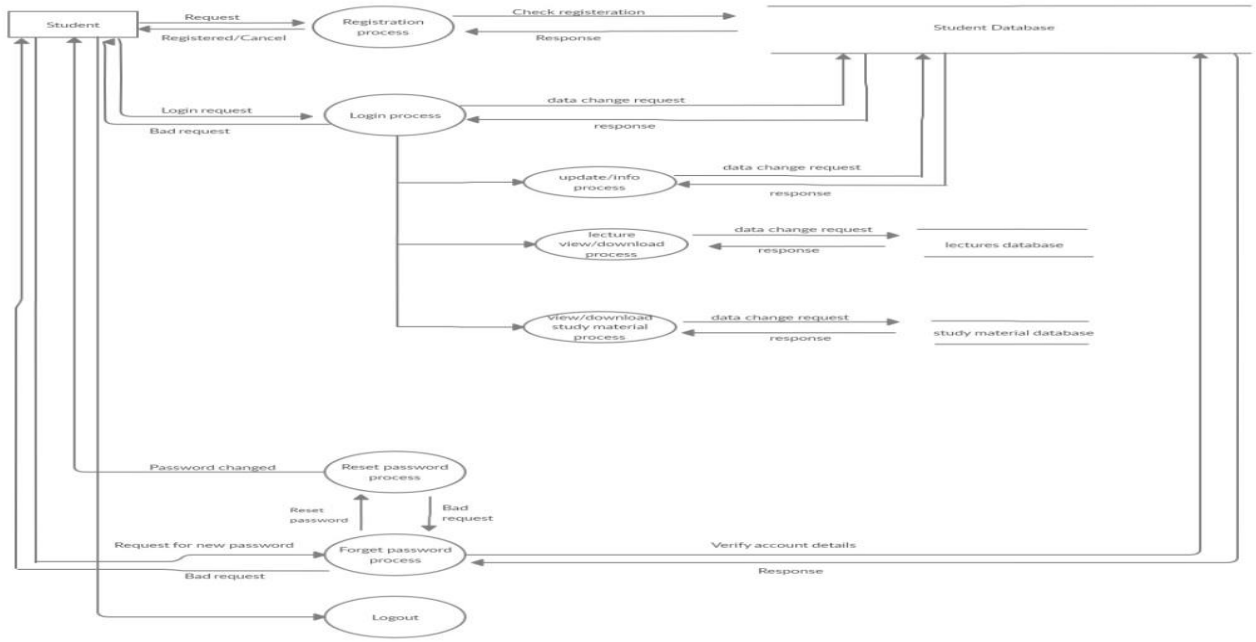


Level 2 DFDs simply break processes down into more detailed sub processes. In theory, DFDs could go beyond level 3, but they rarely do. Level 3 data flow diagrams are detailed enough that it doesn't usually make sense to break them down further.

DFD 2 :-

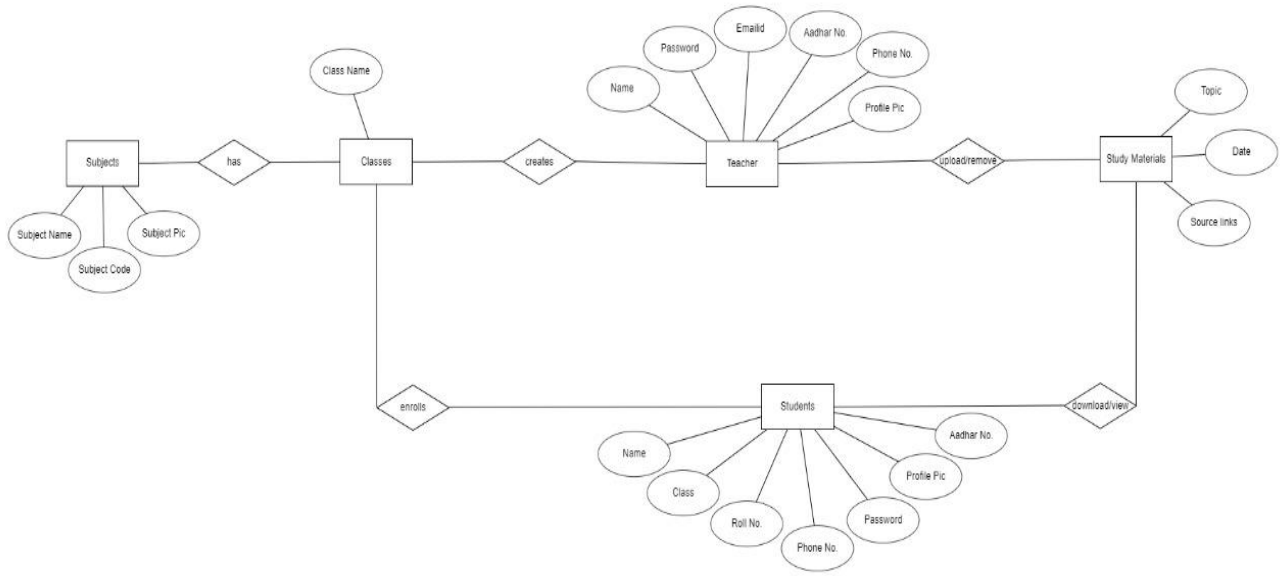


DFD 3 :-



Entity-relationship diagrams (ERD) are essential to modelling anything from simple to complex databases, but the shapes and notations used can be very confusing. This guide will help you to become an expert in ER diagram notation, and you will be well on your way to model your own database.

Entity Relationship Diagram of GuruKul :



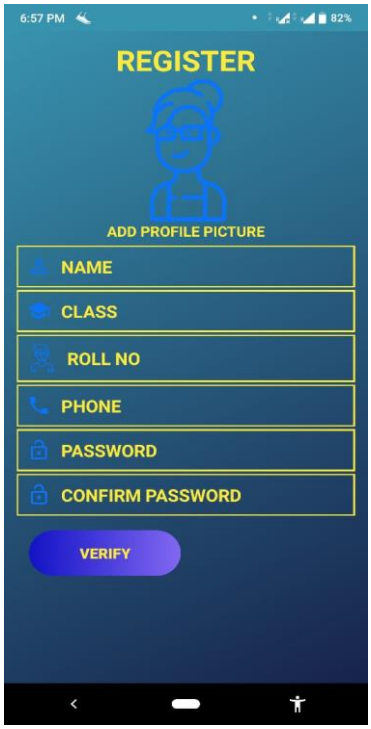
SCREENSHOTS

OF

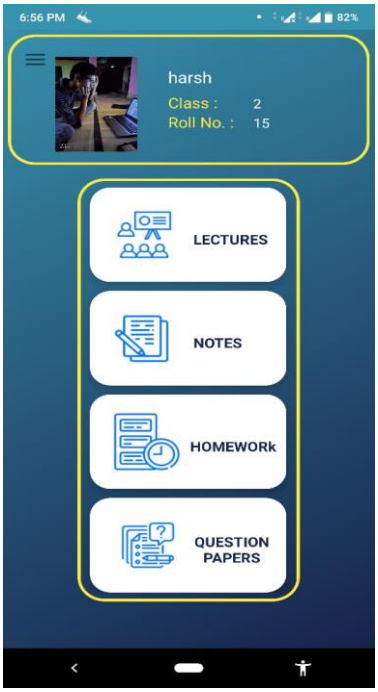
Gurukul
(Students)

Student Login Activity

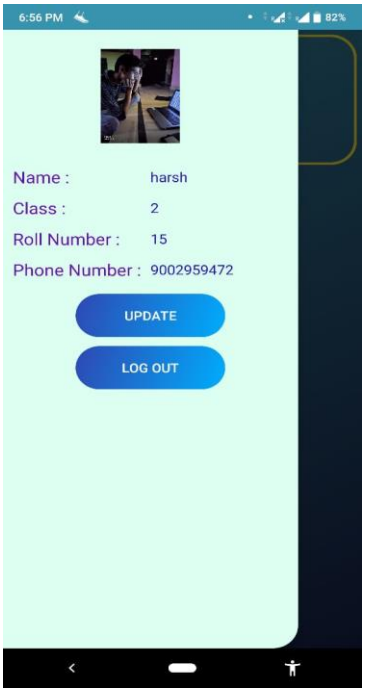
Register Activity



Student Home Activity



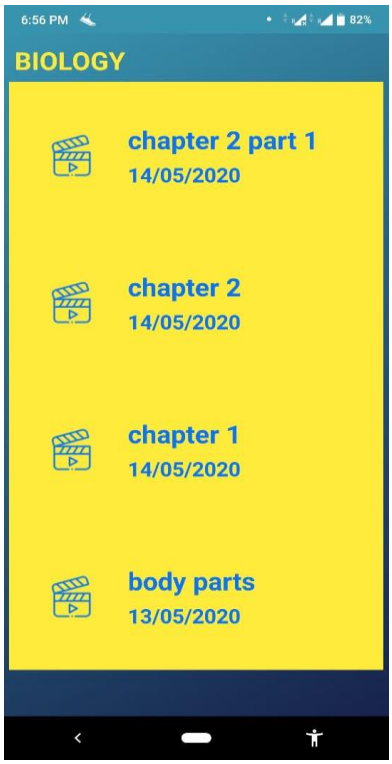
Student Navigation Drawer



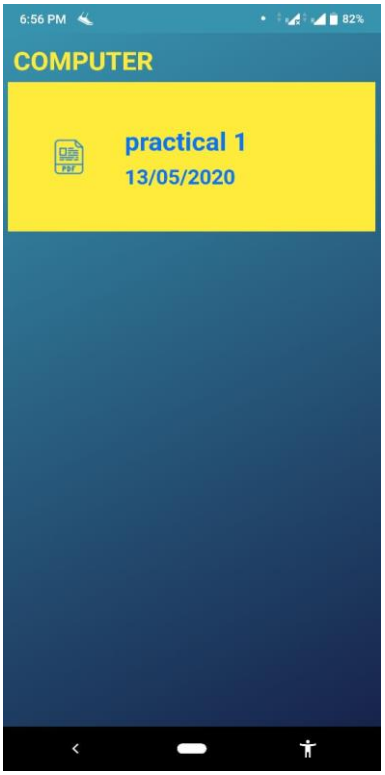
Update Activity

Lectures Activity

Lectures View



Notes View



Login Activity :

```
package com.gurukul;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;

import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import com.gurukul.helpers.MyStaticClass;
import com.gurukul.modals.Students;
```

```
import io.paperdb.Paper;

public class LoginActivity extends AppCompatActivity {
    Button login_btn;
    EditText Class_edittext,Roll_edittext,Password_edittext;
    TextView Register_textview,Forget_password_textview;
    int backpress=1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
        Password_edittext=findViewById(R.id.password_login);
        Class_edittext=findViewById(R.id.class_login);
        Roll_edittext=findViewById(R.id.roll_login);
        Register_textview=findViewById(R.id.register_text);
        Forget_password_textview=findViewById(R.id.forget_text);
        Paper.init(this);
        login_btn=findViewById(R.id.school_verify_btn);
        login_btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                login_user();
```

```
        }
    });


    Register_textview.setOnClickListener(new
View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new
Intent(LoginActivity.this,RegisterActivity.class));
        }
    });


    Forget_password_textview.setOnClickListener(new
View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new
Intent(LoginActivity.this,ForgetPassword.class));
        }
    });
  }


  private boolean check( String password_txt,String std_class_txt,
String roll_no_txt) {
    Boolean result = false;
```

```
    int n = 0;

    if (password_txt.length() == 0) {
        Password_edittext.setError("PLEASE ENTER PASSWORD");
        n = n + 1;
    }
    if (std_class_txt.length() == 0) {
        Class_edittext.setError("PLEASE ENTER CLASS");
        n = n + 1;
    }
    if (roll_no_txt.length() == 0) {
        Roll_edittext.setError("PLEASE ENTER ROLL NO");
        n = n + 1;
    }

    if (n == 0) {
        result = true;
    }
    return result;
}

private void login_user() {
String ClassNO=Class_edittext.getText().toString();
String RollNo=Roll_edittext.getText().toString();
```

```
String Password=Password_edittext.getText().toString();
if(check(Password,ClassNO,RollNo))
{
   Connect_to_database(ClassNO,RollNo,Password);
}
}

private void Connect_to_database(final String ClassNO, final String RollNo, final String password) {
    final DatabaseReference rootref;
    rootref = FirebaseDatabase.getInstance().getReference().child("CLASSES").child("CLASS"+ClassNO).child("ROLLNO");
    rootref.addListenerForSingleValueEvent(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot dataSnapshot)
        {
           if(dataSnapshot.child(RollNo).exists())
           {
              Students students=dataSnapshot.child(RollNo).getValue(Students.class);
              if(students.getPASSWORD().equals(password))
              {
```

```
                MyStaticClass.currentOnlineStudent=students;

Paper.book().write(MyStaticClass.userRollNoKey,RollNo);

Paper.book().write(MyStaticClass.userClassKey,ClassNO);

Paper.book().write(MyStaticClass.userPasswordKey,password);
                startActivity(new
Intent(LoginActivity.this,StudentHome.class));
                finishAffinity();
              }
              else {
                Toast.makeText(LoginActivity.this, "Wrong Password",
Toast.LENGTH_SHORT).show();
              }
            }
            else {
              Toast.makeText(LoginActivity.this, "User Don't Exists",
Toast.LENGTH_SHORT).show();
            }
          }
          @Override
          public void onCancelled(@NonNull DatabaseError
databaseError)
          {
          }
```

```
        });
    }

    @Override
    public void onBackPressed() {

        if (backpress > 1) {
            finishAffinity();
        } else {
            backpress = (backpress + 1);
            Toast.makeText(getApplicationContext(), " Press Back again to Exit ", Toast.LENGTH_SHORT).show();


        }
    }
}
```

Register Activity :

```
package com.gurukul;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityOptionsCompat;

import android.content.ContentResolver;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.webkit.MimeTypeMap;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.ProgressBar;
import android.widget.Toast;
```

```
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.FirebaseException;
import com.google.firebase.auth.PhoneAuthCredential;
import com.google.firebase.auth.PhoneAuthProvider;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;
import com.gurukul.modals.Students;

import java.util.concurrent.TimeUnit;

public class RegisterActivity extends AppCompatActivity {
    private static final int gpick = 1;
    EditText name, phone,std_class, confirm_password, password,
roll_no;
    Button verify_btn, register_btn;
    LinearLayout add_image_layout;
```

```
    ImageView profile_pic;
    String name_txt, phone_txt, std_class_txt, confirm_password_txt,
password_txt, roll_no_txt;
    Uri imguri;
    StorageReference storageref;
    static String profilePicUrl;
    DatabaseReference reference;
    ProgressBar progressBar;



    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register);


        add_image_layout = findViewById(R.id.profile_add_layout);
        profile_pic = findViewById(R.id.profile_pic_add);
        verify_btn = findViewById(R.id.verify);
        register_btn = findViewById(R.id.register);
        name = findViewById(R.id.name);
        phone = findViewById(R.id.phone);
        std_class = findViewById(R.id.std_class);
        confirm_password = findViewById(R.id.confirm_password);
        password = findViewById(R.id.reg_password);
```

```
        roll_no= findViewById(R.id.rollno);
        progressBar=findViewById(R.id.progressBar_register);
        progressBar.setVisibility(View.INVISIBLE);


        reference =
FirebaseDatabase.getInstance().getReference().child("CLASSES");
        storageref =
FirebaseStorage.getInstance().getReference("USER");
        register_btn.setVisibility(View.INVISIBLE);
        register_btn.setEnabled(false);
        verify_btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                name_txt = name.getText().toString();
                phone_txt = phone.getText().toString();
                password_txt = password.getText().toString();
                confirm_password_txt =
confirm_password.getText().toString();
                roll_no_txt = roll_no.getText().toString();
                std_class_txt = std_class.getText().toString();
                if (check(name_txt, password_txt, phone_txt, roll_no_txt,
std_class_txt, confirm_password_txt)) {
                    //if all are filled then here
                    if (!(phone_txt.length() == 10)) {
                        phone.setError("ENTER CORRECT PHONE");
```

```
                } else {
                    if ((password_txt.equals(confirm_password_txt))) {
                        name.setEnabled(false);
                        phone.setEnabled(false);
                        confirm_password.setEnabled(false);
                        password.setEnabled(false);
                        std_class.setEnabled(false);
                        roll_no.setEnabled(false);
                        add_image_layout.setEnabled(false);
                        verify_btn.setEnabled(false);
                        verify_btn.setVisibility(View.INVISIBLE);
                        checkAccount();
                        progressBar.setVisibility(View.VISIBLE);
                    } else {
                        password.setText("");
                        confirm_password.setText("");
                        confirm_password.setError("BOTH PASSWORDS
MUST BE SAME");
                    }
                }
            }

        }
    });
```

```
        add_image_layout.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                OpenGallery();
            }
        });
        register_btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                progressBar.setVisibility(View.VISIBLE);
                if (imguri == null) {
                    profilePicUrl = "https://firebasestorage.googleapis.com/"
+
                            "v0/b/gurukul-
f0593.appspot.com/o/USER%2Fstudent.png?alt=media&token=b619
6674-18d7-4899-9dab-a33eddab93ae";
                    AddUserData();
                } else {
                    SavePicInDatabase();
                }
            }
        });
    }
```

```
   private void checkAccount() {
      DatabaseReference rootref;
      rootref = FirebaseDatabase.getInstance().getReference();
      rootref.addListenerForSingleValueEvent(new
ValueEventListener() {
         @Override
         public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {
            if
(dataSnapshot.child("CLASSES").child("CLASS"+std_class_txt).child("R
OLLNO").child(roll_no_txt).exists()) {
               Toast.makeText(RegisterActivity.this,"Account Already
Exists,Redirecting to Login Page",Toast.LENGTH_SHORT).show();
               startActivity(new
Intent(RegisterActivity.this,LoginActivity.class));
            }
            else {

if(dataSnapshot.child("CLASSES").child("CLASS"+std_class_txt).exists(
))
               {
                  VerifyMyNumber("+91" + phone_txt);
                  new Handler().postDelayed(new Runnable() {
                     @Override
                     public void run() {
                        name.setEnabled(true);
```

```
                    phone.setEnabled(true);
                    confirm_password.setEnabled(true);
                    password.setEnabled(true);
                    std_class.setEnabled(true);
                    roll_no.setEnabled(true);
                    add_image_layout.setEnabled(true);
                    verify_btn.setEnabled(true);
                    verify_btn.setVisibility(View.VISIBLE);
                    progressBar.setVisibility(View.INVISIBLE);
                    Toast.makeText(RegisterActivity.this
                            , "Phone Number Must be in the Phone You
are using to Register",
                            Toast.LENGTH_LONG).show();
                }
            }, 30000);
        }
        else {
            name.setEnabled(true);
            phone.setEnabled(true);
            confirm_password.setEnabled(true);
            password.setEnabled(true);
            std_class.setEnabled(true);
            roll_no.setEnabled(true);
            add_image_layout.setEnabled(true);
```

```
                verify_btn.setEnabled(true);
                verify_btn.setVisibility(View.VISIBLE);
                progressBar.setVisibility(View.INVISIBLE);
                Toast.makeText(RegisterActivity.this, "Class
"+std_class_txt+ " Do not Exist,Ask your teacher to create One.",
Toast.LENGTH_SHORT).show();
            }


          }
        }

        @Override
        public void onCancelled(@NonNull DatabaseError
databaseError) {

        }
      });
   }

   private void AddUserData() {
      Students students=new
Students(name_txt,profilePicUrl,std_class_txt,roll_no_txt,phone_txt,
password_txt);
```

```
reference.child("CLASS"+std_class_txt).child("ROLLNO").child(roll_no
_txt).setValue(students);
      Toast.makeText(this, "NOW U CAN LOGIN",
Toast.LENGTH_SHORT).show();
      startActivity(new Intent(RegisterActivity.this,
LoginActivity.class));
      finish();
   }

   private void SavePicInDatabase() {
      final StorageReference storageReference =
storageref.child("CLASS"+std_class_txt).child(roll_no_txt);
      storageReference.putFile(imguri).addOnSuccessListener(new
OnSuccessListener<UploadTask.TaskSnapshot>() {
         @Override
         public void onSuccess(UploadTask.TaskSnapshot
taskSnapshot) {

storageReference.getDownloadUrl().addOnSuccessListener(new
OnSuccessListener<Uri>() {
               @Override
               public void onSuccess(Uri uri) {
                  profilePicUrl = String.valueOf(uri);
                  AddUserData();
               }
```

```
            });
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            Toast.makeText(RegisterActivity.this, e.getMessage(),
Toast.LENGTH_SHORT).show();
        }
    });
  }


  private void OpenGallery() {
    Intent galaryint = new Intent();
    galaryint.setAction(Intent.ACTION_GET_CONTENT);
    galaryint.setType("image/*");
    startActivityForResult(Intent.createChooser(galaryint, "select"),
gpick);
  }


  @Override
  protected void onActivityResult(int requestCode, int resultCode,
@Nullable Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
```

```
        if (requestCode == gpick && resultCode == RESULT_OK &&
data.getData() != null) {
            imguri = data.getData();
            profile_pic.setImageURI(imguri);
        }
    }


    private void VerifyMyNumber(final String s) {
        PhoneAuthProvider.getInstance().verifyPhoneNumber(s, 30L,
TimeUnit.SECONDS, RegisterActivity.this, new
PhoneAuthProvider.OnVerificationStateChangedCallbacks() {
            @Override
            public void onVerificationCompleted(@NonNull
PhoneAuthCredential phoneAuthCredential) {
                Toast.makeText(RegisterActivity.this, "Phone Verified",
Toast.LENGTH_SHORT).show();
                register_btn.setEnabled(true);
                register_btn.setVisibility(View.VISIBLE);
                progressBar.setVisibility(View.INVISIBLE);
            }


            @Override
            public void onCodeSent(@NonNull String s, @NonNull
PhoneAuthProvider.ForceResendingToken forceResendingToken) {
                super.onCodeSent(s, forceResendingToken);
```

```
            Toast.makeText(RegisterActivity.this, "OTP Sent...",
Toast.LENGTH_SHORT).show();
        }


        @Override
        public void onVerificationFailed(@NonNull FirebaseException
e) {
            name.setEnabled(true);
            phone.setEnabled(true);
            confirm_password.setEnabled(true);
            password.setEnabled(true);
            std_class.setEnabled(true);
            roll_no.setEnabled(true);
            add_image_layout.setEnabled(true);
            verify_btn.setEnabled(true);
            verify_btn.setVisibility(View.VISIBLE);
            progressBar.setVisibility(View.INVISIBLE);
        }
      });
   }
```

```
    private boolean check(String name_txt, String password_txt, String
phone_txt, String std_class_txt, String roll_no_txt, String
confirm_password_txt) {
        Boolean result = false;
        int n = 0;
        if (name_txt.length() == 0) {
            name.setError("PLEASE ENTER NAME");
            n = 1;
        }
        if (password_txt.length() == 0) {
            password.setError("PLEASE ENTER PASSWORD");
            n = n + 1;
        }
        if (phone_txt.length() == 0) {
            phone.setError("PLEASE ENTER NUMBER");
            n = n + 1;
        }
        if (std_class_txt.length() == 0) {
            std_class.setError("PLEASE ENTER CLASS");
            n = n + 1;
        }
        if (roll_no_txt.length() == 0) {
            roll_no.setError("PLEASE ENTER ROLL NO");
            n = n + 1;
```

```
        }
        if (confirm_password_txt.length() == 0) {
            confirm_password.setError("PLEASE ENTER PASSWORD
AGAIN");
            n = n + 1;
        }
        if (n == 0) {
            result = true;
        }
        return result;
    }
}
```

SCREENSHOTS

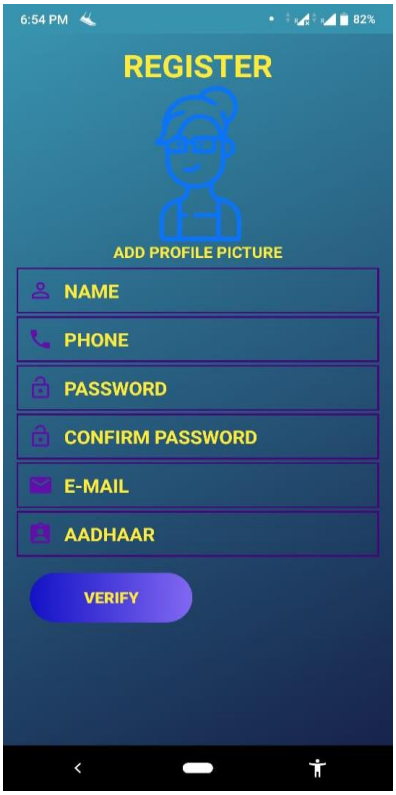
OF

Gurukul

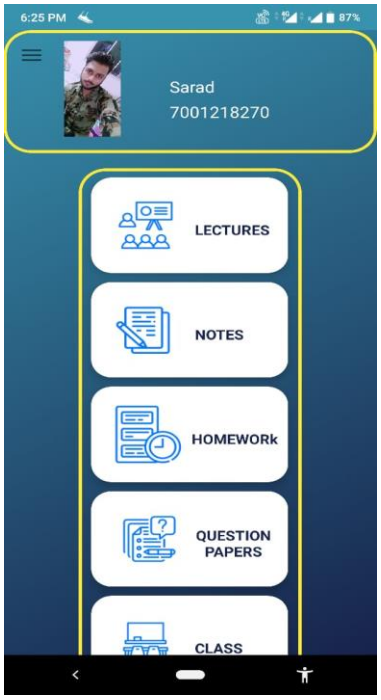
(Teachers)

Login Activity

Register Activity



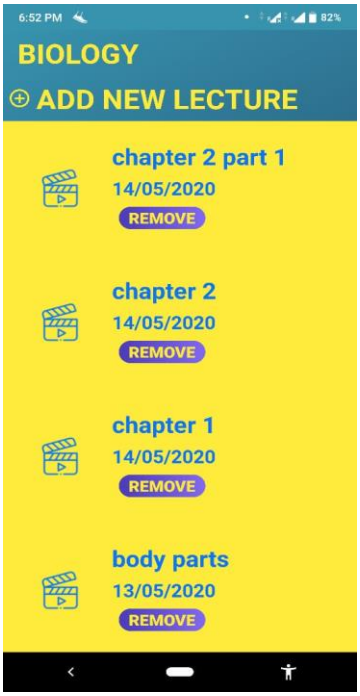
Teacher Home



Class Detail Activity

Class Creation Activity

Lectures Detail Activity



Homework Details Activity

Upload Activity

Login Activity :

```
package com.example.gurukulguru;

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

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import com.example.gurukulguru.ModelClasses.Teachers;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import io.paperdb.Paper;
```

```
public class TeacherLoginActivity extends AppCompatActivity {
    Button teacher_login_btn;
    TextView ForgetPassword,not_register;
    EditText phone,password;
    String Phone_text,Password_text;
    DatabaseReference rootref;
    int backpress=1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_teacher_login);



        teacher_login_btn = findViewById(R.id.teacher_login_button);
        ForgetPassword = findViewById(R.id.forget_password);
        not_register=findViewById(R.id.not_registered_text);
        phone=findViewById(R.id.phone_number);
        password=findViewById(R.id.password);

        Paper.init(this);



        not_register.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
```

```
            startActivity(new Intent(TeacherLoginActivity.this,
RegisterActivity.class));
        }
    });

    rootref= FirebaseDatabase.getInstance().getReference().child("CLASSES");

    ForgetPassword.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent intent = new Intent(TeacherLoginActivity.this,
ForgetPassword.class);
            startActivity(intent);
        }
    });
    teacher_login_btn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Phone_text=phone.getText().toString();
            Password_text=password.getText().toString();
            if((Phone_text.length()==10))
            {
                if((Password_text.length()!=0))
                {
                    password.setEnabled(false);
                    phone.setEnabled(false);
                    teacher_login_btn.setEnabled(false);
```

```
                LoginUser();
            }
            else {
                password.setError("Password can't be Null");
            }
        }
        else {

         phone.setError("Please Enter Correct Phone Number");
        }
      }
    });
  }

  private void LoginUser() {
    final DatabaseReference rootref;
    rootref=FirebaseDatabase.getInstance().getReference();
    rootref.addListenerForSingleValueEvent(new ValueEventListener() {
      @Override
      public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
        if (dataSnapshot.child("TEACHERS").child(Phone_text).exists()) {
            Teachers
teachers=dataSnapshot.child("TEACHERS").child(Phone_text).getValue(Teache
rs.class);
            if(teachers.getPassword().equals(Password_text))
            {
                MyStaticClass.currentOnlineTeacher=teachers;
```

```
                Paper.book().write(MyStaticClass.userPhoneKey,Phone_text);

Paper.book().write(MyStaticClass.userPasswordKey,Password_text);
                startActivity(new
Intent(TeacherLoginActivity.this,TeacherHome.class));
                finishAffinity();
              }else {
                Toast.makeText(TeacherLoginActivity.this, "Wrong Passowrd !!!",
Toast.LENGTH_SHORT).show();
                password.setEnabled(true);
                phone.setEnabled(true);
                teacher_login_btn.setEnabled(true);
              }
            }
            else {
              Toast.makeText(TeacherLoginActivity.this, "Account With This
Number Don't Exists", Toast.LENGTH_SHORT).show();
              password.setEnabled(true);
              phone.setEnabled(true);
              teacher_login_btn.setEnabled(true);
            }
          }

          @Override
          public void onCancelled(@NonNull DatabaseError databaseError) {

          }
```

```
        });
    }
    @Override
    public void onBackPressed() {


        if (backpress > 1) {
            finishAffinity();
        } else {
            backpress = (backpress + 1);
            Toast.makeText(getApplicationContext(), " Press Back again to Exit ",
Toast.LENGTH_SHORT).show();
        }
    }
}
```

Register Activity :

```
package com.example.gurukulguru;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

import com.example.gurukulguru.ModelClasses.Teachers;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.FirebaseException;
import com.google.firebase.auth.PhoneAuthCredential;
import com.google.firebase.auth.PhoneAuthProvider;
```

```
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;

import java.util.concurrent.TimeUnit;

public class RegisterActivity extends AppCompatActivity {
    private static final int gpick = 1;
    EditText name, phone, aadhaar, confirm_password, password, email;
    Button verify_btn, register_btn;
    LinearLayout add_image_layout;
    ImageView profile_pic;
    String name_txt, phone_txt, aadhaaar_txt, confirm_password_txt,
password_txt, email_txt;
    Uri imguri;
    StorageReference storageref;
    static String profilePicUrl;
    DatabaseReference reference;

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

```
        setContentView(R.layout.activity_register);


        add_image_layout = findViewById(R.id.profile_add_layout);
        profile_pic = findViewById(R.id.profile_pic_add);
        verify_btn = findViewById(R.id.verify);
        register_btn = findViewById(R.id.register);
        name = findViewById(R.id.name);
        phone = findViewById(R.id.phone);
        aadhaar = findViewById(R.id.aadhaar);
        confirm_password = findViewById(R.id.confirm_password);
        password = findViewById(R.id.reg_password);
        email = findViewById(R.id.email);


        reference =
FirebaseDatabase.getInstance().getReference().child("TEACHERS");
        storageref = FirebaseStorage.getInstance().getReference("TEACHERS");
        register_btn.setVisibility(View.INVISIBLE);
        register_btn.setEnabled(false);
        verify_btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                name_txt = name.getText().toString();
                phone_txt = phone.getText().toString();
                password_txt = password.getText().toString();
                confirm_password_txt = confirm_password.getText().toString();
                email_txt = email.getText().toString();
                aadhaaar_txt = aadhaar.getText().toString();
```

```
            if (check(name_txt, password_txt, phone_txt, aadhaaar_txt,
email_txt, confirm_password_txt)) {
                //if all are filled then here
                if (!(phone_txt.length() == 10)) {
                    phone.setError("ENTER CORRECT PHONE");
                } else {
                    if ((password_txt.equals(confirm_password_txt))) {

                        if (aadhaaar_txt.length() == 12) {
                            name.setEnabled(false);
                            phone.setEnabled(false);
                            confirm_password.setEnabled(false);
                            password.setEnabled(false);
                            aadhaar.setEnabled(false);
                            email.setEnabled(false);
                            add_image_layout.setEnabled(false);
                            verify_btn.setEnabled(false);
                            verify_btn.setVisibility(View.INVISIBLE);
                            checkAccount();

                        } else {
                            aadhaar.setError("ENTER CORRECT AADHAAR");
                        }
                    } else {
                        password.setText("");
                        confirm_password.setText("");
```

```
                    confirm_password.setError("BOTH PASSWORDS MUST BE
SAME");
                }
            }
        }

    }
});
add_image_layout.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        OpenGallery();
    }
});
register_btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (imguri == null) {
            profilePicUrl = "https://firebasestorage.googleapis.com/" +
                    "v0/b/gurukul-
f0593.appspot.com/o/USER%2Fstudent.png?alt=media&token=b6196674-
18d7-4899-9dab-a33eddab93ae";
            AddUserData();
        } else {
            //if upload pic
            SavePicInDatabase();
        }
```

```
        }
      });
    }

    private void checkAccount() {
      DatabaseReference rootref;
      rootref = FirebaseDatabase.getInstance().getReference();
      rootref.addListenerForSingleValueEvent(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
          if (dataSnapshot.child("TEACHERS").child(phone_txt).exists()) {
            Toast.makeText(RegisterActivity.this,"Account Already
Exists,Redirecting to Login Page",Toast.LENGTH_SHORT).show();
            startActivity(new
Intent(RegisterActivity.this,TeacherLoginActivity.class));
          }
          else {
            VerifyMyNumber("+91" + phone_txt);
          }
        }

        @Override
        public void onCancelled(@NonNull DatabaseError databaseError) {

        }
      });
    }
```

```
    private void AddUserData() {
        Teachers teachers = new Teachers(name_txt, phone_txt, aadhaaar_txt,
profilePicUrl, email_txt, password_txt);
        reference.child(phone_txt).setValue(teachers);
        Toast.makeText(this, "NOW U CAN LOGIN",
Toast.LENGTH_SHORT).show();
        startActivity(new Intent(RegisterActivity.this, TeacherLoginActivity.class));
        finish();
    }


    private void SavePicInDatabase() {
        final StorageReference storageReference = storageref.child(phone_txt);
        storageReference.putFile(imguri).addOnSuccessListener(new
OnSuccessListener<UploadTask.TaskSnapshot>() {
            @Override
            public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {
                storageReference.getDownloadUrl().addOnSuccessListener(new
OnSuccessListener<Uri>() {
                    @Override
                    public void onSuccess(Uri uri) {
                        profilePicUrl = String.valueOf(uri);
                        AddUserData();
                    }
                });
            }
        }).addOnFailureListener(new OnFailureListener() {
```

```
        @Override
        public void onFailure(@NonNull Exception e) {
            Toast.makeText(RegisterActivity.this, e.getMessage(),
Toast.LENGTH_SHORT).show();
        }
      });
   }

   private void OpenGallery() {
      Intent galaryint = new Intent();
      galaryint.setAction(Intent.ACTION_GET_CONTENT);
      galaryint.setType("image/*");
      startActivityForResult(Intent.createChooser(galaryint, "select"), gpick);
   }

   @Override
   protected void onActivityResult(int requestCode, int resultCode, @Nullable
Intent data) {
      super.onActivityResult(requestCode, resultCode, data);

      if (requestCode == gpick && resultCode == RESULT_OK && data.getData()
!= null) {
         imguri = data.getData();
         profile_pic.setImageURI(imguri);
      }
   }
```

```
    private void VerifyMyNumber(final String s) {
        PhoneAuthProvider.getInstance().verifyPhoneNumber(s, 30L,
TimeUnit.SECONDS, RegisterActivity.this, new
PhoneAuthProvider.OnVerificationStateChangedCallbacks() {
            @Override
            public void onVerificationCompleted(@NonNull PhoneAuthCredential
phoneAuthCredential) {
                Toast.makeText(RegisterActivity.this, "Phone Verified",
Toast.LENGTH_SHORT).show();
                register_btn.setEnabled(true);
                register_btn.setVisibility(View.VISIBLE);
            }

            @Override
            public void onCodeSent(@NonNull String s, @NonNull
PhoneAuthProvider.ForceResendingToken forceResendingToken) {
                super.onCodeSent(s, forceResendingToken);
                Toast.makeText(RegisterActivity.this, "OTP Sent...",
Toast.LENGTH_SHORT).show();
            }

            @Override
            public void onVerificationFailed(@NonNull FirebaseException e) {
                Toast.makeText(RegisterActivity.this, e.getMessage(),
Toast.LENGTH_SHORT).show();
            }
        });
    }
```

```
   private boolean check(String name_txt, String password_txt, String
phone_txt, String aadhaaar_txt, String email_txt, String confirm_password_txt)
{
      Boolean result = false;
      int n = 0;
      if (name_txt.length() == 0) {
         name.setError("PLEASE ENTER NAME");
         n = 1;
      }
      if (password_txt.length() == 0) {
         password.setError("PLEASE ENTER PASSWORD");
         n = n + 1;
      }
      if (phone_txt.length() == 0) {
         phone.setError("PLEASE ENTER NUMBER");
         n = n + 1;
      }
      if (aadhaaar_txt.length() == 0) {
         aadhaar.setError("PLEASE ENTER AADHAAR");
         n = n + 1;
      }
      if (email_txt.length() == 0) {
         email.setError("PLEASE ENTER EMAIL");
         n = n + 1;
      }
```

```
        if (confirm_password_txt.length() == 0) {
            confirm_password.setError("PLEASE ENTER PASSWORD AGAIN");
            n = n + 1;
        }
        if (n == 0) {
            result = true;
        }
        return result;
    }
}
```

BIBLIOGRAPHY

www.wikipedia.com

www.google.com

www.geeksforgeeks.org

www.guru99.com

www.lucidchart.com

www.youtube.com

www.devoloper.android.com

CONCLUSION

While developing the system, a conscious effort has been made to create and develop a software package, making use of available tools, techniques and resources that would make a proper application.

While making the system, an eye has been kept on making it as user friendly and flexible as possible. As such one may hope that the system will be acceptable to any user and will adequately meet his/her needs. For designing the system, we have used simple data flow diagrams. Overall the project teaches us the essential skills like using system analysis and design techniques like data flow diagram in designing the system.