CHAPTER -1

INTRODUCTION

1.1 OVERVIEW OF THE PROJECT

A platform where we let the trainer and trainee to communicate with one another and eradicate the communication gap between them for the betterment of the society in becoming literate without paying much. An app that would help a trainee to easily interact with the trainer and can easily resolve the problems that he/she is facing. In any organisation or any institution this will help students and employees to learn new things and resolve their issues that they may face.

1.2 DRAWBACK OF THE SYSTEM

The main problem faced by the students is the lack knowledge about the teaching staff outside their department and the busy schedule of the teacher at work. Often students are shy about asking the teacher about their doubts.

1.3 ANDROID

What is Android?

Android is the world's most popular operating system for mobile devices and tablets. It is an open source operating system, created by Google, and available to all kinds of developers with various expertise levels, ranging from rookie to professional.

From a developer's perspective, Android is a Linux-based operating system for smartphones and tablets. It includes a touch screen user interface, widgets, camera, network data monitoring and all the other features that enable a cell phone to be called a smartphone. Android is a platform that supports various applications, available through the Android Play Store. The Android platform also allows end users to develop, install and use their own applications on top of the Android framework. The Android framework is licensed under the Apache License, with Android application developers holding the right to distribute their applications under their customized license.

Android Architecture

Android is architected in the form of a software stack comprising applications, an operating system, run-time environment, middleware, services and libraries. This architecture can, perhaps, best be represented visually as outlined in figure. Each layer

of the stack, and the corresponding elements within each layer, are tightly integrated and carefully tuned to provide the optimal application development and execution environment for mobile devices.

The Linux Kernel

Positioned at the bottom of the Android software stack, the Linux Kernel provides a level of abstraction between the device hardware and the upper layers of the Android software stack. Based on Linux version 2.6, the kernel provides pre-emptive multitasking, low-level core system services such as memory, process and power management in addition to providing a network stack and device drivers for hardware such as the device display, Wi-Fi and audio.

The original Linux kernel was developed in 1991 by Linus Torvalds and was combined with a set of tools, utilities and compilers developed by Richard Stallman at the Free Software Foundation to create a full operating system referred to as GNU/Linux. Various Linux distributions have been derived from these basic underpinnings such as Ubuntu and Red Hat Enterprise Linux.

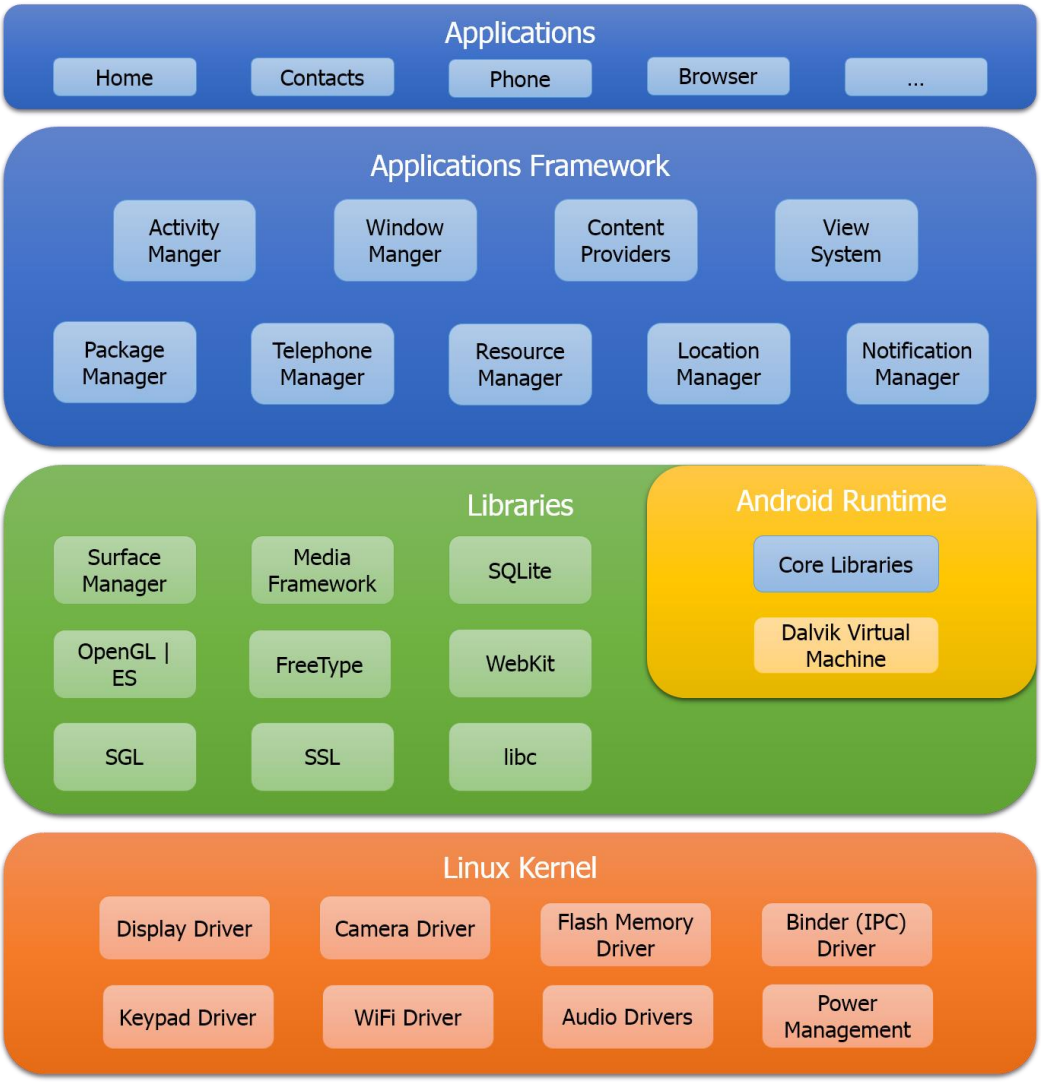
Android Runtime - Dalvik Virtual Machine

As previously noted, the Linux kernel provides a multitasking execution environment allowing multiple processes to execute concurrently. It would be easy to assume, therefore, that each Android application simply runs as a process directly on the Linux kernel. In fact, each application running on an Android device does so within its own instance of the Dalvik virtual machine (VM).

Running applications in virtual machines provides a number of advantages. Firstly, applications are essentially sandboxed, in that they cannot detrimentally interfere (intentionally or otherwise) with the operating system or other applications, nor can they directly access the device hardware. Secondly, this enforced level of abstraction makes application platform neutral in that they are never tied to any specific hardware.

The Dalvik virtual machine was developed by Google and relies on the underlying Linux kernel for low-level functionality. It is more efficient than the standard Java VM in terms of memory usage, and specifically designed to allow multiple instances to run efficiently within the resource constraints of a mobile device.

In order to execute within a Dalvik VM, application code must be transformed from standard Java class files to the Dalvik executable (.dex) format, which has a 50% smaller memory footprint than standard Java byte code. Standard Java class files can usually (though not always) be converted to Dex format using the dx tool included with the Android SDK.



CHAPTER-2

SYSTEM FEATURES

In the working part which we mentioned earlier as well but that info wasn't complete so as far as the whole working of this model is concerned it must be briefed as: -

We will provide one platform like we can think of one android app where there would be all the details of the staff as who is professional at what and thus the student can directly communicate to the faculty member instead of roaming around the college being unknown about the right person to talk to.
There are some good things that we would provide in our app, some of them are: -

- One chat window where the student and the staff member can communicate
- If the problem is too complex and requires the attention of more than one faculty member then we can create a window of the desired people in the conversation
- A proper meeting can be scheduled among the participant parties.
- Proper information regarding the staff member would be available (for example if a person is good in java then this trait would be present there with the name of the person/staff member)
- Students can give points to the teachers or their mentors, by having this feedback the students can know of how good a person is in his/her job and would it be beneficial for them
- Special reward facility should be there for the students based on whether who is more active and who has shown more progress or any other criteria
- Award of honour for the staff member would also be there
- No fees for the students to pay for being perfect in what they want
- Showcasing the efforts and achievements of both the student and the teacher/staff member
- The app must show proper time schedule of the teachers so that students may interact with them in their free time and thus there won't be a lot of chaos for the teachers
- Notifications must be delivered (like if a teacher that a student is looking for is free at the moment or notification at the time which is set for the meeting)
- A proper login window available for the student's security
- Feature of rating the teacher's capability on certain criteria's

2.1 USE CASES

- Mainly in educational premises
- In offices
- In industries
- In society

2.2 PRESENT REQUIREMENT

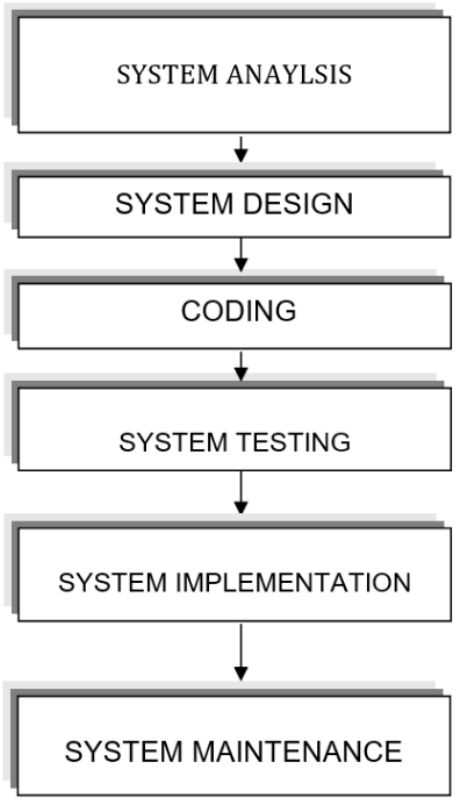
- It is required to lessen the communication gap
- Required for those who are not sound enough on the cost criteria
- Platform for the students to know more about their interests from the professionals
- Older solutions don’t have these facilities
- Cheap and useful
- Students can go to the best possible person for their doubts
- Time management

CHAPTER -3

PROCESS DESCRIPTION

The basic idea of software development life cycle (SDLC) is that there is a well-defined process by which an application is conceived, developed and implemented. The phases in the SDLC provide a basis for the management and control because they define segments of the flow of work, which can be identified for the managerial purpose and specifies the documents or other deliveries to be produced in each phase.

System Development revolves around a life cycle that begins with the recognition of user needs. In order to develop good software, it has to go through different phases. There are various phases of the System Development Life Cycle for the project and different models for software development, which depict these phases. We decided to use waterfall model, the oldest and the most widely used paradigm for software engineering. The Various relevant stages of the System Life Cycle of this Application Tool are depicted in the following flow diagram.



Let us have a look on each of the above activities:

3.1 SYSTEM ANALYSIS

System Analysis is the process of diagnosing situations, done with a defiant aim, with the boundaries of the system kept in mind to produce a report based on the findings. Analysis is fact-finding techniques where problem definition, objective, system requirement specifications, feasibility analysis and cost benefit analysis are carried out. The requirement of both the system and the software are document and reviewed with the user.

3.2 SYSTEM DESIGN

System Design is actually a multistep process that focuses on four distinct attributes of a program: data structures, software architecture, interface representations, and procedural (algorithmic) detail. System design is concerned with identifying the software components (Functions, data streams, and data stores), specifying relationships among components, specifying software structure, maintaining a record of design decisions and providing a blueprint for the implementation phase.

3.3 CODING

Coding step performs the translations of the design representations into an artificial language resulting in instructions that can be executed by the computer. It thus involves developing computer programs that meet the system specifications of design stage.

3.4 SYSTEM TESTING

System testing process focuses on the logical internals of the software, ensuring that all statements have been tested on the functional externals, that is conducting tests using various tests data to uncover errors that defined input will produce actual results that agree with required results.

3.5 SYSTEM IMPLEMENTATION

System Implementation is a process that includes all those activities that take place to convert an old system to a new system. The new system may be totally new system replacing the existing system or it may be major modification to the existing system. Coding performs the translations of the design representations into an artificial language resulting in instructions that can be executed by the computer. It thus involves developing computer programs that meet the system design specifications. System

implementation involves the translation of the design specifications into source code and debugging, documentation and unit testing of the source code.

3.6 SYSTEM MAINTENANCE

Maintenance is modification of a software product after delivery to correct faults to improve performance or to adopt the product to a new operating environment. Software maintenance cannot be avoided due to ware & tear caused by users. Some of the reasons for maintaining the software are

1. Over a period of time, software original requirements may change.
2. Errors undetected during software development may be found during user & require correction.
3. With time, new technologies are introduced such as hardware, operating system etc. The software therefore must be modified to adapt new operating environment.

CHAPTER-4

PROPOSED SYSTEM

One of the tools of the structure analysis is the Date Flow Diagrams. A DFD is a graphical representation of the system. The Data Flow Diagram is used by the system analyst to explain the flow of the data in the system.

Process

A process represents some Amount of work being performed on the data. A process does transformation of data from one form to another. A circle represents a process. The process must be named and numbered appropriately.

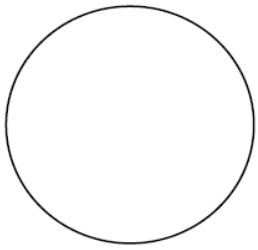
Data flow

A Data Flow designates an interface among different components in the DFD. It represents the path of data as it flows through the system. An Arrow represents a data flow. The name of the data flow is written along the line.

Data source

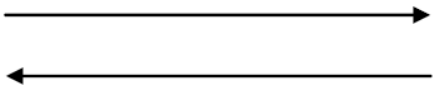
A Data Source is a repository of data. An open-ended rectangle or two horizontal parallel lines represent it. A DFD, which describes the system at a very general level, is called the Context Diagram. It contains a single process, but it plays a very important role in studying the system. The following pages displayed the context diagram and the DFD's of subsequent levels of the Hard & Soft Solution

PROCESS

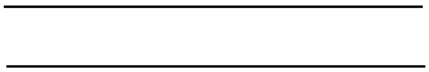


ENTITY

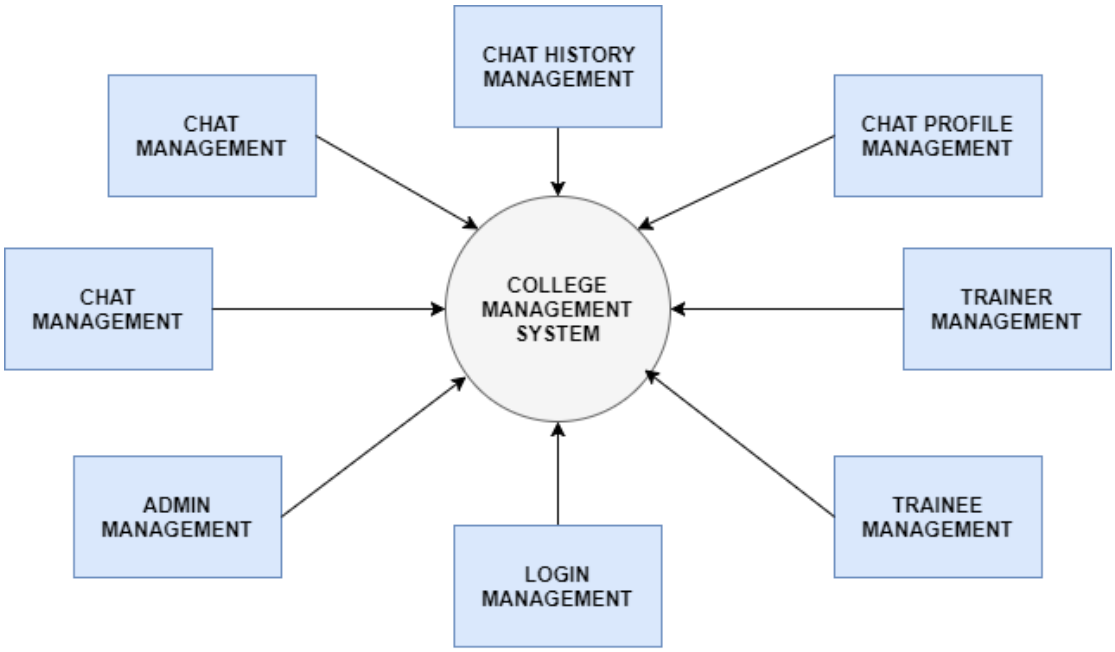
DATA FLOW



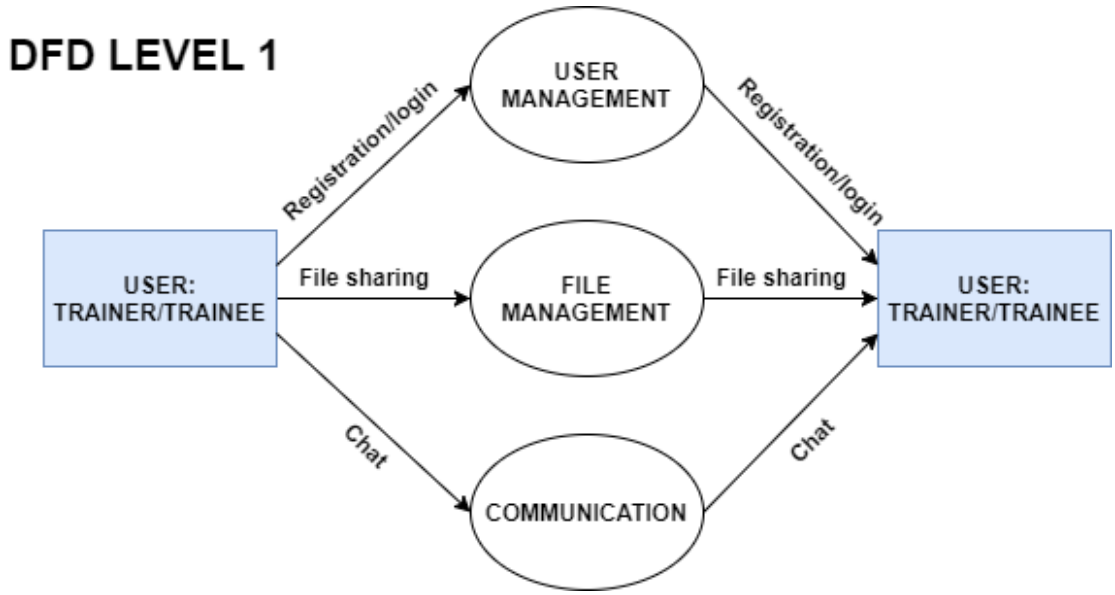
DATA STOCK / DATA BASE



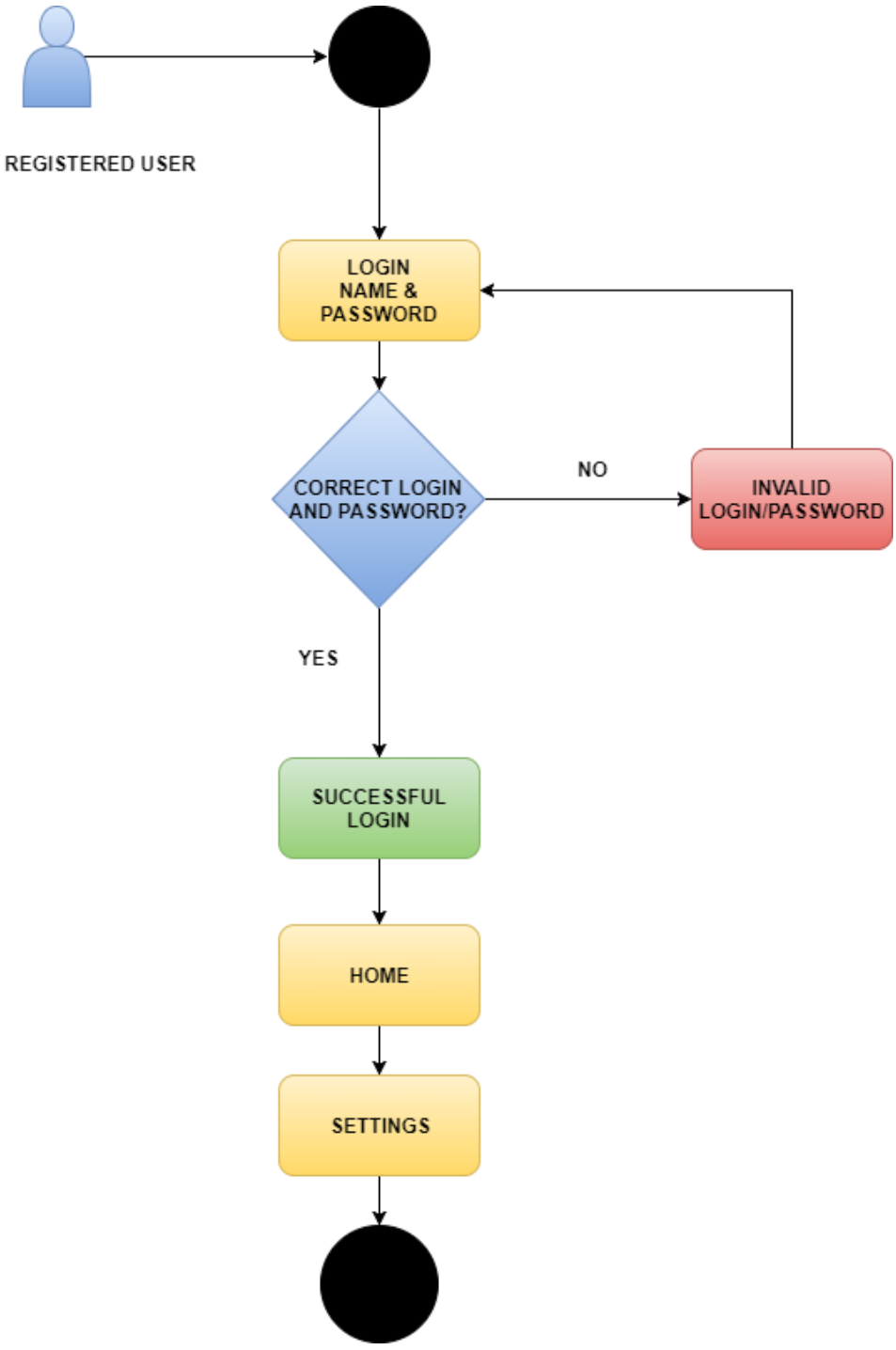
4.1 DFD



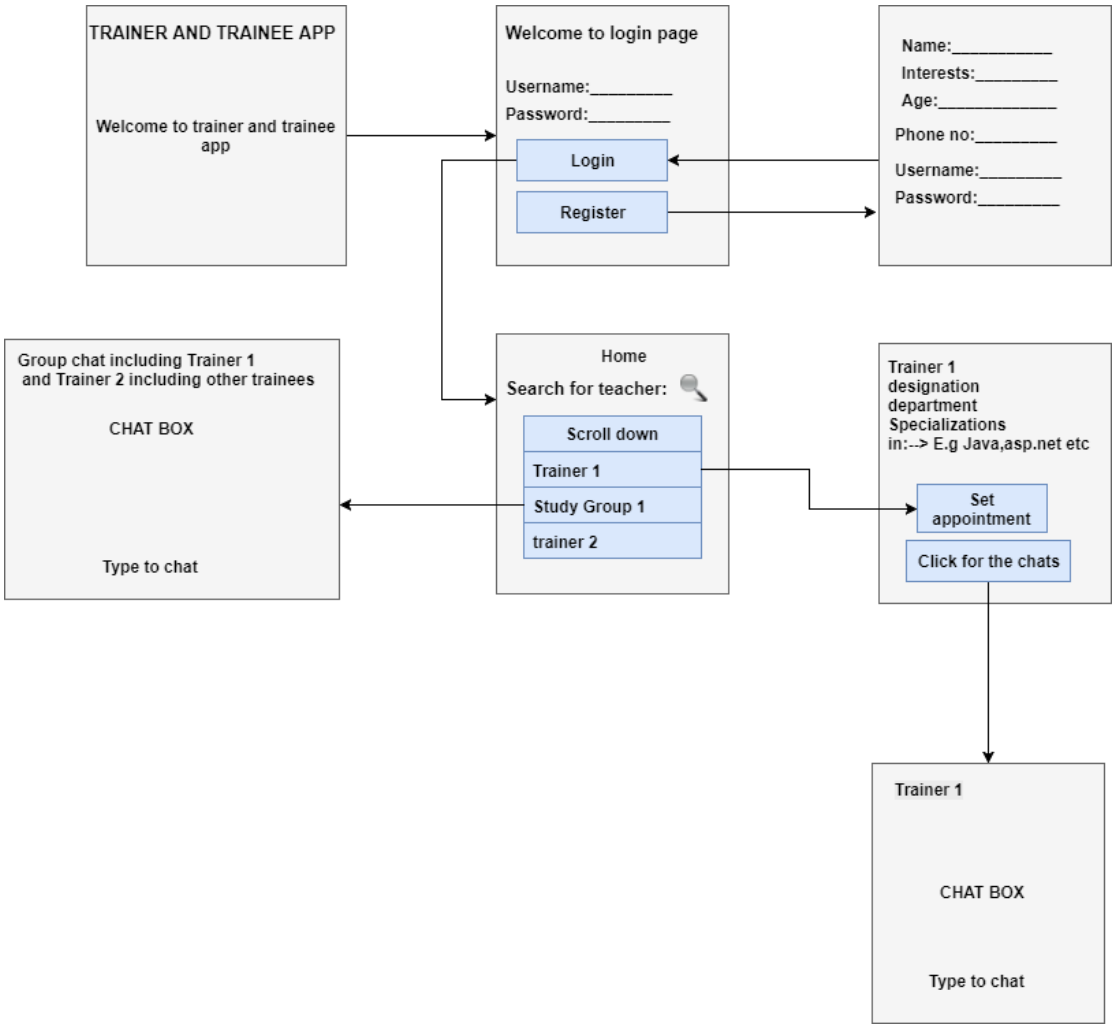
ZERO LEVEL DFD FOR TRAINER AND TRAINEE CHAT APPLICATION



SYSTEM ACTIVITY DIAGRAM



SCREEN FLOW DIAGRAM

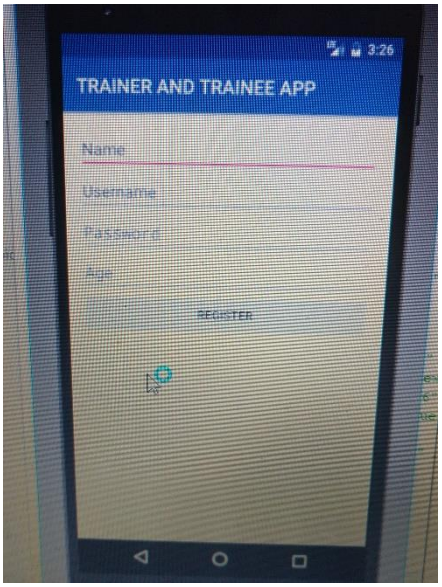


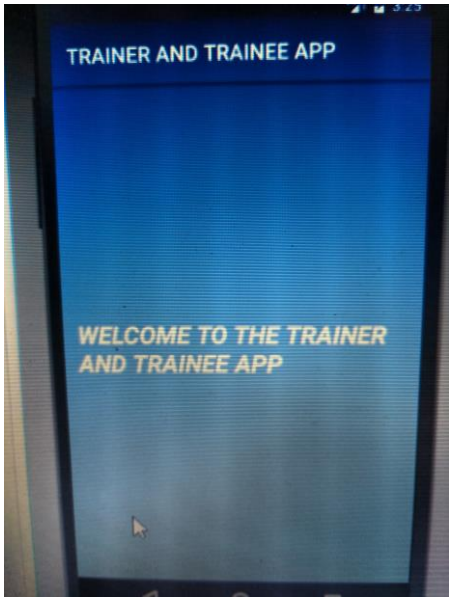
4.2 RESOURCES REQUIRED

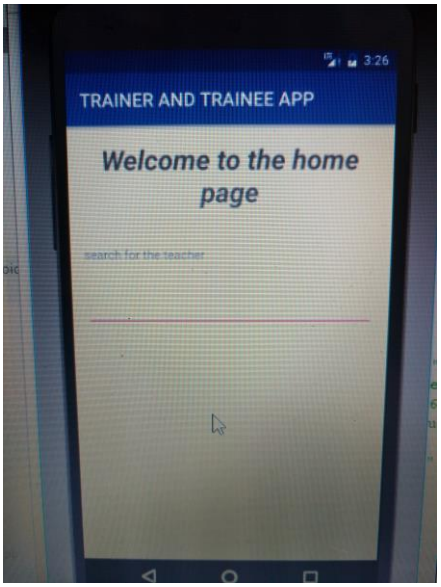
For the implementation of App following are the resources used for its development and successful implementation:

1. System will be developed in Java Platform.
2. Data Management is done using MySQL.
3. Mobile Operating system will be Android.
4. Technologies to be used are XML, AJAX, JSON, and PHP.
5. Tools used are Android Studio.

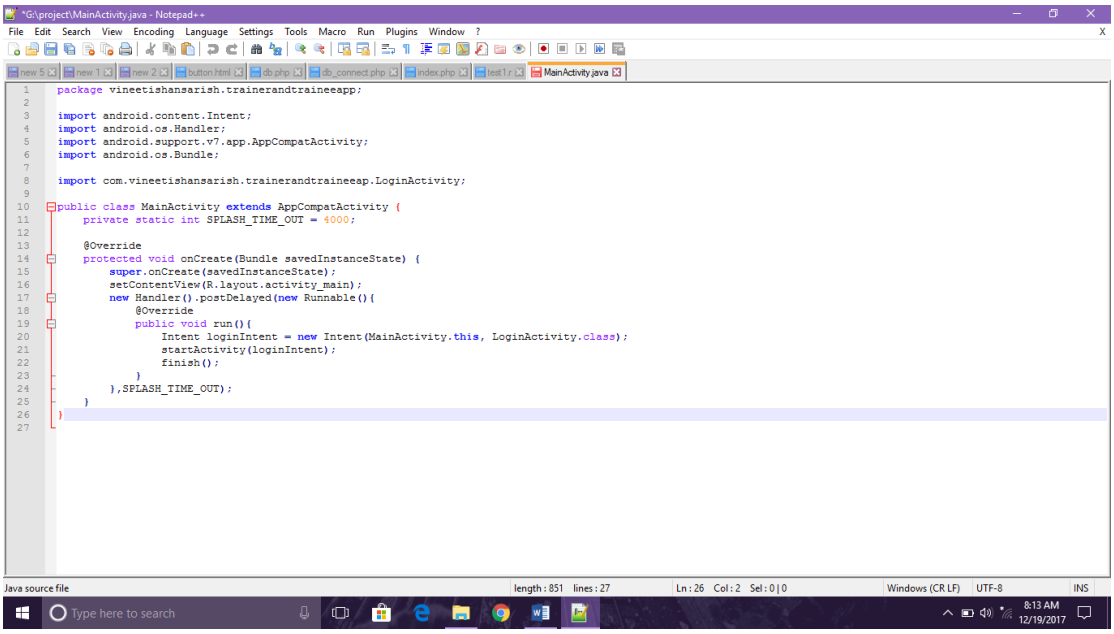
4.3 SCREEN SHOTS







4.4 CODE SAMPLE



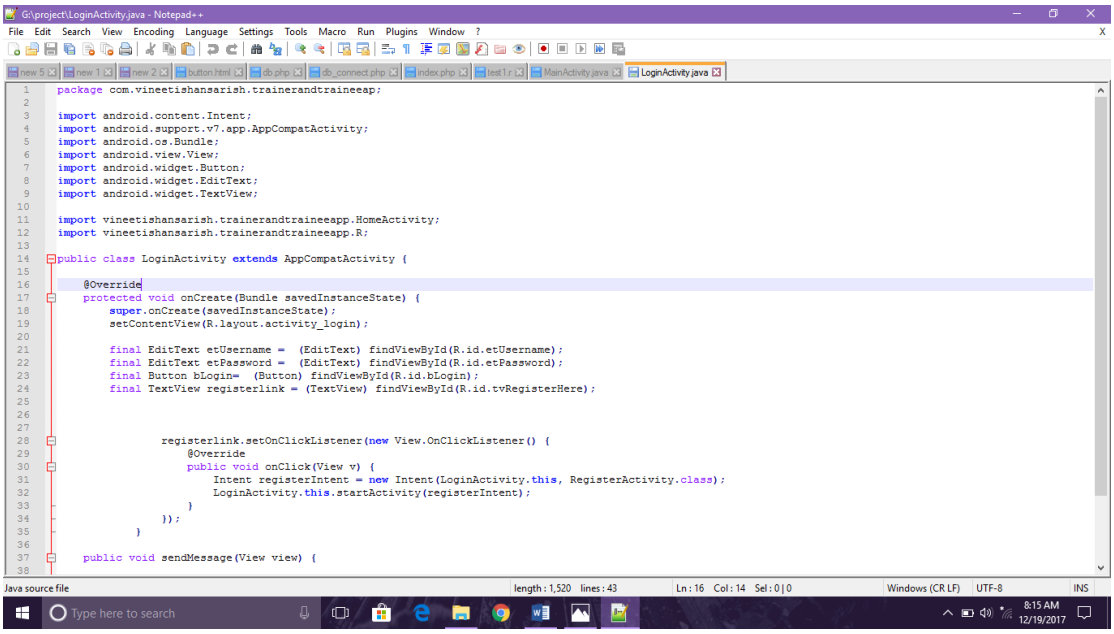
```
package vineetishansarish.trainerandtraineeapp;

import android.content.Intent;
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

import com.vineetishansarish.trainerandtraineeap.LoginActivity;

public class MainActivity extends AppCompatActivity {
    private static int SPLASH_TIME_OUT = 4000;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        new Handler().postDelayed(new Runnable(){
            @Override
            public void run(){
                Intent loginIntent = new Intent(MainActivity.this, LoginActivity.class);
                startActivity(loginIntent);
                finish();
            }
        },SPLASH_TIME_OUT);
    }
}
```



```
package com.vineetishansarish.trainerandtraineeap;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

import vineetishansarish.trainerandtraineeapp.HomeActivity;
import vineetishansarish.trainerandtraineeapp.R;

public class LoginActivity extends AppCompatActivity {

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

        final EditText etUsername =  (EditText) findViewById(R.id.etUsername);
        final EditText etPassword =  (EditText) findViewById(R.id.etPassword);
        final Button bLogin=  (Button) findViewById(R.id.bLogin);
        final TextView registerlink = (TextView) findViewById(R.id.tvRegisterHere);


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

    public void sendMessage(View view) {
```

CHAPTER-5

FUTURE WORKS

- As said earlier as well in the use cases part as well that this facility in the near future would not only be adhered to the educational premises but we can implement this thing in offices in universities, in industries and moreover in any area as well like for the common individuals as well.
- This we can think of implementing in the foreign colleges and industries as well.
- This would be beneficial for everyone who wants to learn something and that too free.
- In the future we can have some contacts with the mighty tech geeks and professionals whose services would be provided to the local students with some cost factor indulging here but not that high it would be only a bit of which we think everyone can afford.
- Video calling and video conference feature can be also added

CHAPTER-6

CONCLUSION

At the end we want to say that this idea would be really impactful in the future. It would prove beneficial not only to some students but to the entire society. It would eradicate the communication failure among the people and will be help people to achieve their desired goal. We don't have any competition in the industry and thus we are the first ones to put forward this idea. People can have these incentives through our platform without any cost. People can enjoy the facilities within very less time and prosper in quick time. This application not only benefits the student but also the trainer because through this application people can help other people and gain recognition and respect in the society.

REFERENCES

- Software Engineering book by Pearson
- O'Reilly - PHP in a Nutshell by Paul Hudson
- An introduction to Database Systems :Bipin C. Desai
- Principles of Data Base Management Systems: Aho Ullman
- www.developer.android.com
- www.androidcookbook.com
- www.google.com