ANDROID APPLICATION FOR GRAPHIC ERA GLOBAL SCHOOL AND APPLY TESTING USING UFT

A PROJECT REPORT

Submitted in partial fulfillment of the
Requirement for the award of degree of B.Tech in
Information Technology & Computer Science Engineering

BY

SHITANSHU SHEKHAR (I.T, 2007722)
HARSH VARDHAN (C.S.E, 2007464)
RAJAT GOYAL (I.T, 2007716)

Under the Guidance of
Ms. Neha Garg
Assistant Professor

Department of Computer Science and information Technology Engineering
Graphic Era University
(Under Section 3 of UGC Act, 1956)
Dehradun-248002
2017

GraphicEra
UNIVERSITY
University under section 3 of UGC Act, 1956
Accredited by NAAC with Grade A

CERTIFICATE

We hereby certify that the work which is being presented in the major project report titled **"Android Application for Graphic Era Global School and apply testing using UFT "** in partial fulfillment for the award of the Degree of Bachelor of Technology in Computer Science and Engineering / Information Technology of Graphic Era University and submitted in the Department of Computer Science and Information Technology Engineering of Graphic Era University, Dehradun is a record of our own work carried out during a period from August 2016 to May 2017, under the supervision of **Ms. Neha Garg, Assistant Professor,** Department of Information Technology and engineering of Graphic Era University, Dehradun.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree or Diploma.

Shitanshu Shekhar	Harsh Vardhan	Rajat Goyal
E.No - 2007722	E.No - 2007464	E.No -2007716
Dept. of I.T	Dept. of C.S.E	Dept. of I.T

(Ms. Neha Garg)
Assistant Professor
Graphic Era University

Head of Department
Dept. of CS/IT

ABSTRACT

A school is a place designed to provide learning spaces and learning environments for the teaching of students under the direction of teachers. Meanwhile parents are concern about their children and wants to keep the track of their children. So, we developed an application which provide a complete information about the child. The rapid proliferation of mobile computing technology has massive potential for providing access to different services at any time and from anywhere. The system is built to be an application for **Graphic Era Global School**. The main objective of this application is to provide the progress of the children to the parents.

The Project is developed in Java Programming Language by using the Android Studio 2.3. We use the Android Software Development Kit (SDK) which includes a variety of custom tools that help us develop mobile applications on the Android platform. The most important of these are the Android Emulator and the Android Development Tools (ADT) plug-in for Android Studio. We also use the HP Unified Functional Testing Tool for the testing of the application.

ACKNOWLEDGEMENT

We feel ourselves honored to have had suchan opportunity as to work under the capable guidance of **Ms. Neha Garg, Assistant Professor**, without which this project would not have been possible. Her exemplary counseling, monitoring, insipiration and continuous encouragment are only of a few ways he played a substantial role in our endeavor and her teaching shall go a long mile to help us in making informed and succesful choices.

We also take this moment to heartly thanks everyone surrounding us for keeping us to the mark and always look to the better future . We are indebted to them for their love and support through this work.

We convey our sincere thanking to the staff of Graphic Era University, especially Prof. Manish Sarma, Prof. Vijay Singh and Mr. Vidit Kumar for his invaluable assistance. Their Motivation has been a stepping stone in making this undertaking a successs.

Date: Shitanshu Shekhar
Place: Dehradun Harsh Vardhan
Rajat Goyal

TABLE OF CONTENT

PAGE NO.

CHAPTER 1 : Introduction

CHAPTER 2 : Technologies Used

CHAPTER 5 : Coding

CHAPTER 6 : Testing and Implementation

CHAPTER 7 : Conclusion and Future Scope

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATION

GEGS	Graphic Era Global School
UFT	Unified Functional Testing Tool
IDE	integrated development environment
ALM	Application Lifecycle Management
AUT	Application under test
SDK	software development kit
APK	Android application package
NDK	Native Development Kit
POI	points of interest
SRS	Software Requirements Specification
DFD	Data Flow Diagram

Chapter 1

Introduction

1.1 Introduction

Our plan in this project was to develop an Android mobile application for Graphic Era Global University which allows the users or parents to browse easily the available information about the school and their children at on their smart phones. We hope our work can give parents enough flexibility to surf the fee status, attendance and any enquiry easily through the application. We tested the application through automated testing tool Unified functional testing tool (UFT) so that it did not create any further error.

1.2 Who are we ?

The background of our group members are different from each other, but what we share is a common desire for production and development which motivates all of us to start this challenging project. We belong to different study groups varying from Information Technology to Computer Science at Graphic Era University. What we all want here is to learn more about Android development world, UFT testing environment and learn how to use all the possibilities around it, either in later at work or in our master thesis.

1.3 Project Description

This project is developed just to give realistic environment. Each section is divided into small modules, which are worked upon as with user choice. The application must be needed installed in his/her mobile phone. The user runs the application while visiting. He accesses the information from a simple interface or the information about any location. This project **Graphic Era Global School Application** provides the students information. This proposed application does not require any internet access and thus eliminates the disadvantage of single point failure**.** The system gives the basic details that will be required such as an image of that student along with basic details like the photo, contact no, attendance, fee status etc. It helped the parents for continues growth of student. The user can also zoom in and zoom out to seek a better view. The application must be now installed in his mobile phone. The user runs the application while visiting. The application now allows user to select a type of details from the following list:

- Student information
- Facility information
- Profile information
- Gallery
- Contact Us
- Notice
- Academic Calendar
- Time Table
- Syllabus

1.4 Approach Done

The project was done in two stages. First we have to learn about working of Unified Functional testing tool a product of Hewlett Packard (HP). This tool helps testers to perform an automated functional testing seamlessly without monitoring once script development is complete. HPE Unified Function Testing (UFT) software automates functional testing through an intuitive, visual user experience that ties manual testing, automated software testing, and framework-based testing together in one integrated development environment (IDE). HPE UFT is the industry standard automated software testing solution, with proven capabilities for meeting the challenges of today's agile, modern application teams. After that we have to learn about the android studio and some basic concepts of java like creation of database, connectivity of database to the android application and at last after successfully testing the application by HPE UFT, how to deploy the application in certain environment.

Second we learned the working of the Android Studio and made the android application for the school and test it for the better performance of module.

Chapter 2

Technologies Used

2.1 Introduction

Technology is the collection of techniques, skills, methods and processes used in the production of goods or services or in the accomplishment of objectives, such as scientific investigation. Technology can be the knowledge of techniques, processes, and the like, or it can be embedded in machines which can be operated without detailed knowledge of their workings.

I used the Android and Unified functional testing tool during the development of the project. Both of the technology are widely used in the industry for the growth of the cutting age software.

2.2 HP Unified Functional Testing Tool (UFT)

HPE Unified Function Testing (UFT) software automates functional testing through an intuitive, visual user experience that ties manual testing, automated software testing, and framework-based testing together in one integrated development environment (IDE). The capabilities in HPE UFT help you significantly reduce the cost, time spent, and complexity of the functional testing process while driving continuous quality.

HPE UFT is the industry-standard automated software testing solution, with proven capabilities for meeting the challenges of today's agile, modern application teams. HPE UFT leverages a broad ecosystem of integrations—from source control management tools such as Subversion or Git, to continuous integration tools such as Jenkins. Functional tests can be triggered as part of the regular build process and run results reported in HPE Application Lifecycle Management (ALM) or HPE Quality Center. Teams are instantly alerted to issues to keep Agile projects on track. HPE UFT also includes the HPE Lean Functional Testing solution designed for Agile testing where test creation is done in Microsoft® Visual Studio/C# and Eclipse/Java.

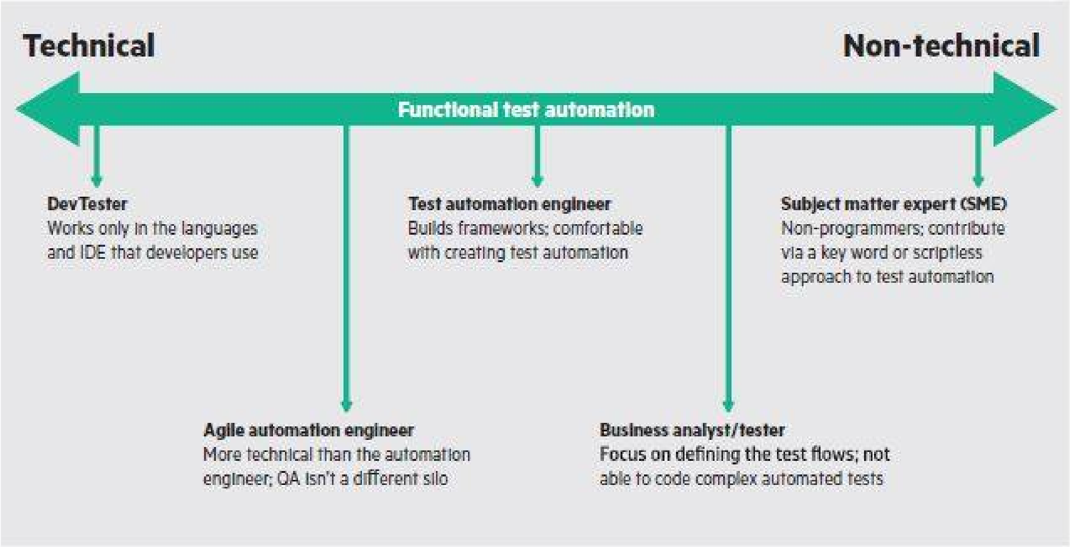
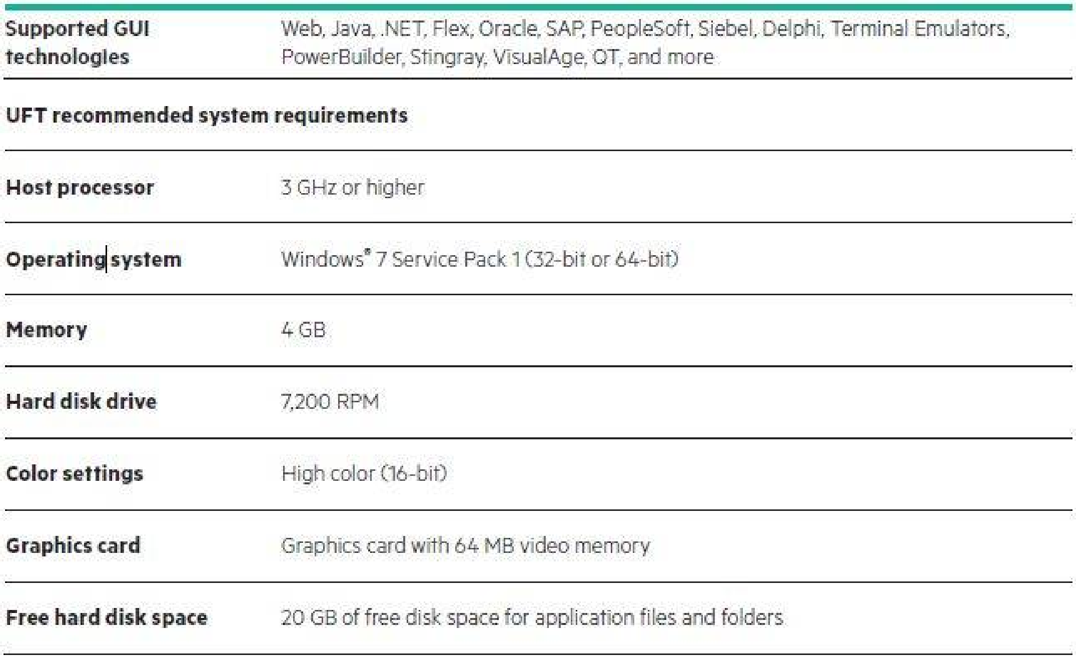

Technical
Non-technical
Functional test automation
DevTester
Works only in the languages and IDE that developers use
Test automation engineer
Builds frameworks; comfortable with creating test automation
Subject matter expert (SME)
Non-programmers; contribute via a key word or scriptless approach to test automation
Agile automation engineer
More technical than the automation engineer; QA isn't a different silo
Business analyst/tester
Focus on defining the test flows; not able to code complex automated tests

Figure 2.1 HPE Unified Functional Testing addresses the testing needs of any organization regardless of skillset.

2.2.1 Key features

- Single unified interface: Automate testing for both GUI functionality and the back-end service parts of an application.
- Manual test conversion: Jump-start new automations by importing manual tests from HPE Sprinter; convert the results into reusable test automation assets for regression and integration tests.
- Full browser coverage: Run tests on the latest versions of Chrome, IE, Firefox, and Safari on Mac.
- SAP testing: Leverage extensive support for SAPUI5 objects and methods, SAP Web Dynpro, ABAP, and the SAP NWBC Desktop application.
- Version control: Version control can be done with HPE ALM or a standard source control system. As an example, HPE UFT works directly within your Subversion (SVN) or Git repositories to update and commit changes to your testing documents, run a difference comparison, revert to the last committed version, or resolve version conflicts with Subversion.
- BPT integration: Create, edit, and maintain BPT framework tests directly from HPE UFT.



Supported GUI technologies	Web, Java, .NET, Flex, Oracle, SAP, PeopleSoft, Siebel, Delphi, Terminal Emulators, PowerBuilder, Stingray, VisualAge, QT, and more
UFT recommended system requirements	
Host processor	3 GHz or higher
Operating system	Windows® 7 Service Pack 1 (32-bit or 64-bit)
Memory	4 GB
Hard disk drive	7,200 RPM
Color settings	High color (16-bit)
Graphics card	Graphics card with 64 MB video memory
Free hard disk space	20 GB of free disk space for application files and folders

Table 2.1 Technology platform support

2.2.2 Working Environment and Limitation

UFT Working Environment:

- UFT runs only in windows environment.
- It uses VB script as a scripting language.
- The technologies it supports are Web, Java, .Net, SAP, Oracle, Siebel, Web Services, and many major languages.

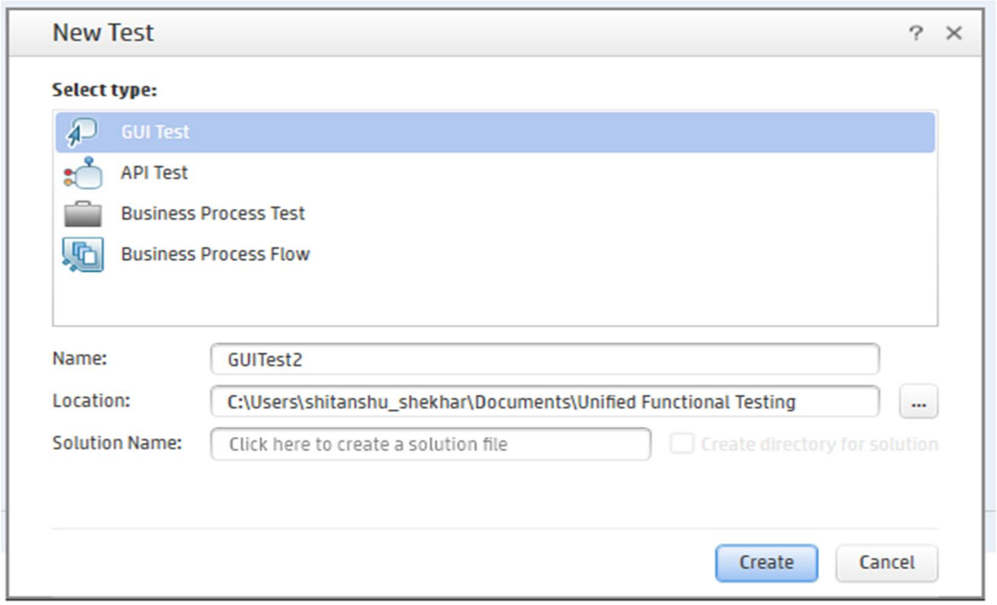
Limitations:

- Cost is extremely high – License and maintenance.
- It does not support the UNIX/ Linux flavour operating system.
- Challenges in object detection.

2.2.3 Working procedure of UFT

- The basic test creation method is record and playback.
- When a tester launches UFT and performs a series of operations on the Application under test. It generates lines of code that corresponds to each operation performed. This will be the basic test script**.**
- Moving on to Playback, when the test script created is run, it performs the exact same operations on the AUT thus playing back the sequence of steps already recorded.
- UFT identifies the various objects in the AUT by a name or handler ID that the object possesses. During the record phase it captures all these properties and during playback, it performs the desired operations like mouse click, checkbox checking etc.

2.2.4 Build a simple test

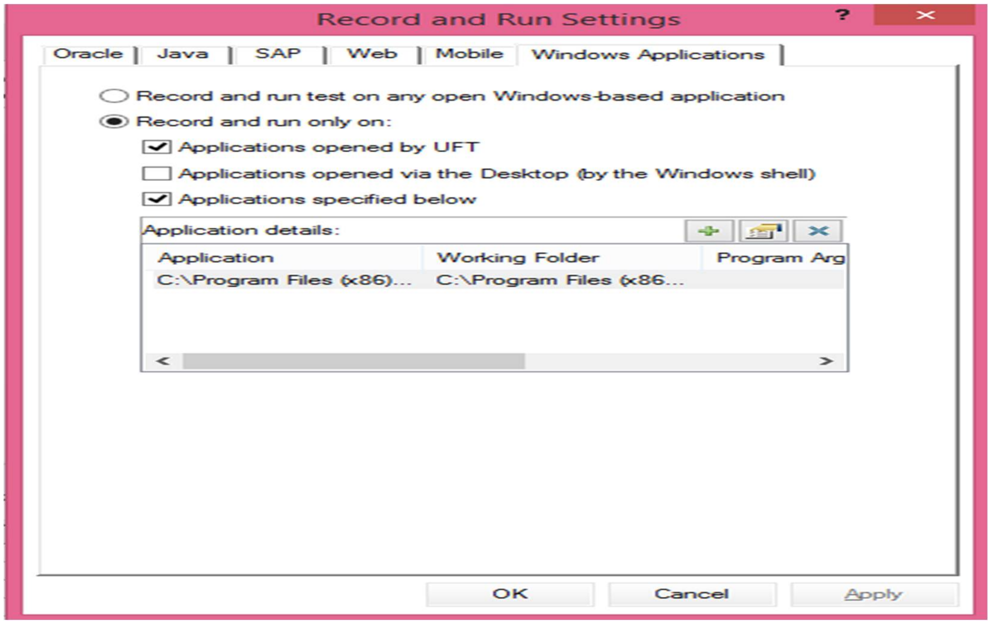


Name: GUITest2

Location: C:\Users\shitanshu_shekhar\Documents\Unified Functional Testing

Solution Name: Click here to create a solution file

Figure 2.2 creating a new GUI test segment



Record and Run Settings

○ Record and run test on any open Windows-based application

● Record and run only on:

☑ Applications opened by UFT

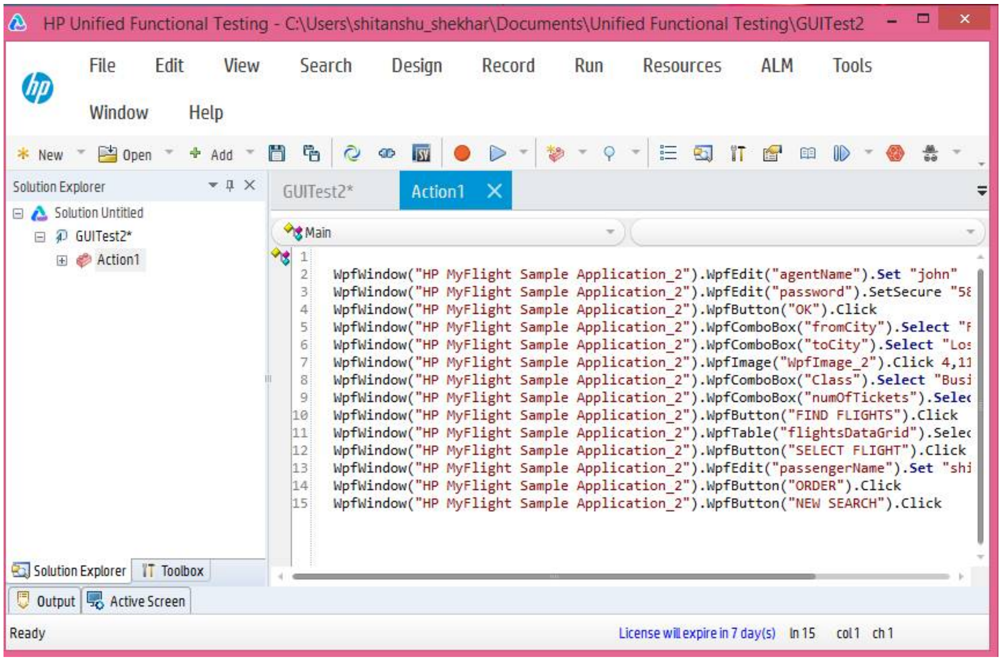
□ Applications opened via the Desktop (by the Windows shell)

☑ Applications specified below

Application details:

Application	Working Folder	Program Arg
C:\Program Files (x86)...	C:\Program Files (x86...	

Figure 2.3 Setting the Record feature for desired test

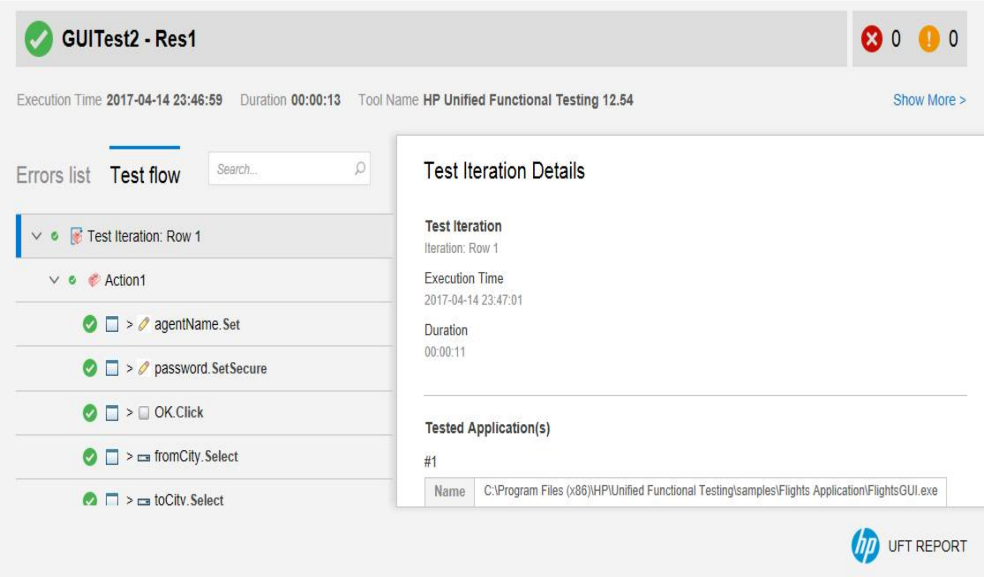


HP Unified Functional Testing - C:\Users\shitanshu_shekhar\Documents\Unified Functional Testing\GUITest2

```

WpfWindow("HP MyFlight Sample Application_2").WpfEdit("agentName").Set "john"
WpfWindow("HP MyFlight Sample Application_2").WpfEdit("password").SetSecure "5
WpfWindow("HP MyFlight Sample Application_2").WpfButton("OK").Click
WpfWindow("HP MyFlight Sample Application_2").WpfComboBox("fromCity").Select "F
WpfWindow("HP MyFlight Sample Application_2").WpfComboBox("toCity").Select "Los
WpfWindow("HP MyFlight Sample Application_2").WpfImage("WpfImage_2").Click 4,11
WpfWindow("HP MyFlight Sample Application_2").WpfComboBox("Class").Select "Busi
WpfWindow("HP MyFlight Sample Application_2").WpfComboBox("numOfTickets").Selec
WpfWindow("HP MyFlight Sample Application_2").WpfButton("FIND FLIGHTS").Click
WpfWindow("HP MyFlight Sample Application_2").WpfTable("flightsDataGrid").Selec
WpfWindow("HP MyFlight Sample Application_2").WpfButton("SELECT FLIGHT").Click
WpfWindow("HP MyFlight Sample Application_2").WpfEdit("passengerName").Set "shi
WpfWindow("HP MyFlight Sample Application_2").WpfButton("ORDER").Click
WpfWindow("HP MyFlight Sample Application_2").WpfButton("NEW SEARCH").Click
```

Ready   License will expire in 7 day(s)   ln 15   col 1   ch 1

Figure 2.4 Recoding the event of test cases for automation



GUITest2 - Res1

✖ 0 ! 0

Execution Time **2017-04-14 23:46:59** Duration **00:00:13** Tool Name **HP Unified Functional Testing 12.54**

- ✓ Test Iteration: Row 1
  - ✓ Action1
    - ✓ □ > agentName.Set
    - ✓ □ > password.SetSecure
    - ✓ □ > OK.Click
    - ✓ □ > fromCity.Select
    - ✓ □ > toCity.Select

Test Iteration Details

Test Iteration
Iteration: Row 1

Execution Time
2017-04-14 23:47:01

Duration
00:00:11

Tested Application(s)

#1

Name	C:\Program Files (x86)\HP\Unified Functional Testing\samples\Flights Application\FlightsGUI.exe

Figure 2.5 Showing the report after applying automated testing

2.3 Android

Android is a mobile operating system (OS) currently developed by Google, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. It has been the best-selling OS on tablets and on smartphones since 2013, and has the largest installed base. Android's user interface is mainly based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input. The operating system's current design language is Google's Material Design. Android's primary app store is Google Play, with over one million Android applications ("apps") published and 50 billion downloads as of July 2013. In addition to touchscreen devices, Google has further developed Android for television, cars, and wristwatches, each with a specialized yet similar interface. Variants and forked versions of Android are also used on notebooks, game consoles, digital cameras, and other electronics. Android was initially authored by Android, Inc., which Google bought in 2005 and unveiled it in 2007, along with the founding of the Open Handset Alliance a consortium of hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices. A 2013 survey of mobile application developers found that 71% of developers create applications for Android from April to May, and a 2015 survey found that 40% of full-time professional developer’s priority see Android as their target platform, compared to Apple's iOS on 37% with both platforms far above others. In September 2015, Android had 1.4 billion monthly active users. Android's source code is released by Google under open source licenses, although most Android devices ultimately ship with both open source and proprietary software, including required proprietary components for Google's services. It is popular with technology companies that require an optimized, low-cost and customizable operating system for high-tech devices. Its open nature has encouraged a large community of developers and enthusiasts to use the open-source code as a foundation for community-driven projects, which add new features for advanced users or bring Android to devices originally shipped with other operating systems. However, Android has no central update protocol and most devices fail to receive security updates: 2015 research concluded that almost 90% of Android phones in use had known but unpatched security vulnerabilities due to lack of updates and support. The success of Android has made it a target for patent litigation as part of the so-called "smartphones wars" between technology companies.

2.3.1 List of features in Android

- **Web browser:** The web browser available in Android is based on the open-source Blink (previously Web Kit) layout engine, coupled with Chrome's V8 JavaScript engine. Then the Web Kit-using Android Browser scored 100/100 on the Acid3 test on Android 4.0 ICS; the Blink-based browser currently has better standards support. The browser is variably known as 'Android Browser', 'AOSP browser', 'stock browser', 'native browser', and 'default browser'. Starting with Android 4.4 Kit Kat, Google has mandated that the default browser for Android proper be Google Chrome. Since Android 7.1.2 Nougat, the Web View browser that apps can use to display web content without leaving the app has been separated from the rest of the Android firmware in order to facilitate separate security updates by Google.

- **Voice-based features:** Google search through voice has been available since initial release. Voice actions for calling, texting, navigation, etc. are supported on Android 2.2 onwards. As of Android 4.1, Google has expanded Voice Actions with ability to talk back and read answers from Google's Knowledge Graph when queried with specific commands. The ability to control hardware has not yet been implemented.

- **Multi-touch**: Android has native support for multi-touch which was initially made available in handsets such as the HTC Hero. The feature was originally disabled at the kernel level (possibly to avoid infringing Apple's patents on touch-screen technology at the time). Google has since released an update for the Nexus One and the Motorola Droid which enables multi-touch natively.

- **Multitasking:** Multitasking of applications, with unique handling of memory allocation, is available.

- **Multiple language support:** Android supports multiple languages.

- **Storage:** SQLite, a lightweight relational database, is used for data storage purposes

- **Applications:** Android applications are composed of one or more application components (activities, services, content providers, and broadcast receivers). Each component performs a different role in the overall application behavior, and each one can be activated individually (even by other applications). The manifest file must declare all components in the application and should also declare all application requirements, such as the minimum version of Android required and any hardware configurations required. Non-code application resources (images, strings, layout files, etc.) should include alternatives for different device configurations (such as different strings for different languages).

2.4 Software Requirements for Development

2.4.1 Android Studio

Android Studio is the official integrated development environment (IDE) for Android platform development. It was announced on May 16, 2013 at the Google I/O conference. Android Studio is freely available under the Apache License 2.0 Based on JetBrains' IntelliJ IDEA software, Android Studio is designed specifically for Android development.

2.4.1.1 Android Studio Features

- 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 Google Cloud Messaging and App Engine.

2.4.2 Android SDK

The Android software development kit (SDK) includes a comprehensive set of development tools. These include a debugger, libraries, a handset emulator based on QEMU, documentation, sample code, and tutorials. Currently supported development platforms include computers running Linux (any modern desktop Linux distribution), Mac OS X 10.5.8 or later, and Windows XP or later. As of March 2015, the SDK is not available on Android itself, but the software development is possible by using specialized Android applications.

Until around the end of 2014, the officially supported integrated development environment (IDE) was Eclipse using the Android Development Tools (ADT) Plug-in, though IntelliJ IDEA IDE (all editions) fully supports Android development out of the box, and NetBeans IDE also supports Android development via a plug-in. As of 2015, Android Studio, made by Google and powered by IntelliJ, is the official IDE; however, developers are free to use others. Additionally, developers may use any text editor to edit Java and XML files, then use command line tools (Java Development Kit and Apache Ant are required) to create, build and debug Android applications as well as control attached Android devices (e.g., triggering a reboot, installing software package(s) remotely). The SDK also supports older versions of the Android platform in case developers wish to target their applications at older devices. Android applications are packaged in .apk format and stored under /data/app folder on the Android OS (the folder is accessible only to the root user for security reasons). APK (Android application package) package contains .dex files (compiled byte code files called Dalvik executables), resource files, etc.

2.4.3 Android NDK

Libraries written in C, C++ and other languages can be compiled to ARM, MIPS or x86 native code and installed using the Android Native Development Kit (NDK). Native classes can be called from Java code running under the Dalvik VM using the System.load Library call, which is part of the standard Android Java classes. Complete applications can be compiled and installed using traditional development tools. However, according to the Android documentation, NDK should not be used solely for developing applications only because the developer prefers to program in C/C++, as using NDK increases complexity while most applications would not benefit from using it. Java application development based on an IDE such as Eclipse, the NDK is based on command-line tools and requires invoking them manually to build, deploy and debug the apps. Several third-party tools allow integrating the NDK into Eclipse and Visual Studio.

2.4.4 Java

Android applications are developed using the Java language. Java is a very popular programming language developed by Sun Microsystems (now owned by Oracle). Developed long after C and C++, Java incorporates many of the powerful features of those powerful languages while addressing some of their drawbacks. Still, programming languages are only as powerful as their libraries. These libraries exist to help developers build applications. Android relies heavily on these Java fundamentals. The Android SDK includes many standard Java libraries (data structure libraries, math libraries, graphics libraries, networking libraries and everything else you could want) as well as special Android libraries that will help you develop awesome Android applications.

2.4.5 Extensible Mark-up Language

XML stands for Extensible Mark-up Language. XML was designed to store and transport data.XML was designed to be both human- and machine-readable. XML is commonly used as a data format on the Internet. If you want to access data from the Internet, chances are that the data will be in the form of XML. If you want to send data to a Web service, you might also need to send XML. An XML attribute can only have a single value and each attribute can appear at most once on each element. In the common situation where a list of multiple values is desired, this must be done by encoding the list into a well-formed XML attribute with some format beyond what XML defines itself Usually this is either a comma or semi-colon delimited list or, if the individual values are known not to contain spaces a space-delimited list can be used. The advantage to declaring your UI in XML is that it enables you to better separate the presentation of your application from the code that controls its behavior. Your UI descriptions are external to your application code, which means that you can modify or adapt it without having to modify your source code and recompile. For example, you can create XML layouts for different screen orientations, different device screen sizes, and different languages. Additionally, declaring the layout in XML makes it easier to visualize the structure of your UI, so it's easier to debug problems.

2.4.6 Android - SQLite Database

SQLite is an open source SQL database that stores data to a text file on a device. SQLite is a relational database management system contained in a C programming library. It is embedded in android by default. In contrast to many other database management systems, SQLite is not a client–server database engine. Rather, it is embedded into the end program. There is no need to perform any database setup or administration task. Android comes in with built in SQLite database implementation. SQLite supports all the relational database features. In order to access this database, you don't need to establish any kind of connections for it like JDBC, ODBC e.t.c .The main package is android.database.sqlite that contains the classes to manage your own databases. In order to create a database you just need to call this method openOrCreateDatabase with your database name and mode as a parameter. It returns an instance of SQLite database which you have to receive in your own object. we can create table or insert data into table using execSQL method defined in SQLiteDatabase class. SQLite is one way of storing user data. SQLite is a very light weight database which comes with Android OS. The android.database.sqlite.SQLiteOpenHelper class is used for database creation and version management. For performing any database operation, you have to provide the implementation of onCreate() and onUpgrade() methods of SQLiteOpenHelper class. SQLite is a popular choice as embedded database software for local/client storage in application software.

2.4.7 Hardware and Software Requirement specifications

2.4.7.1 Software specification

Operating System:

- Microsoft Windows 7 or later version.
- Mac OS X 10.5.8 or later version with Intel chip.
- Linux including GNU C Library 2.7 or later.

Required tools to develop Android applications

- Minimum SDK API Level 17Android 4.2 (Jelly Bean).
- Maximum SDK API level 24 Android 7.1 (Nougat).
- ADB driver.
- HAXB installer and Google USB Driver.

2.4.7.2 Hardware specification

- Processor – i3
- Hard Disk – 5 GB
- Memory – 4GB RAM
- Android device

2.5 Project Life Cycle

2.5.1 Software Development Life Cycle

Software development organization follows some process when developing a software product. A key component of any software development process is the life cycle model on which the process is based. The particular life cycle model can significantly affect overall life cycle costs associated with a software product. Life cycle of the software starts from concept exploration and at the retirement of the software.

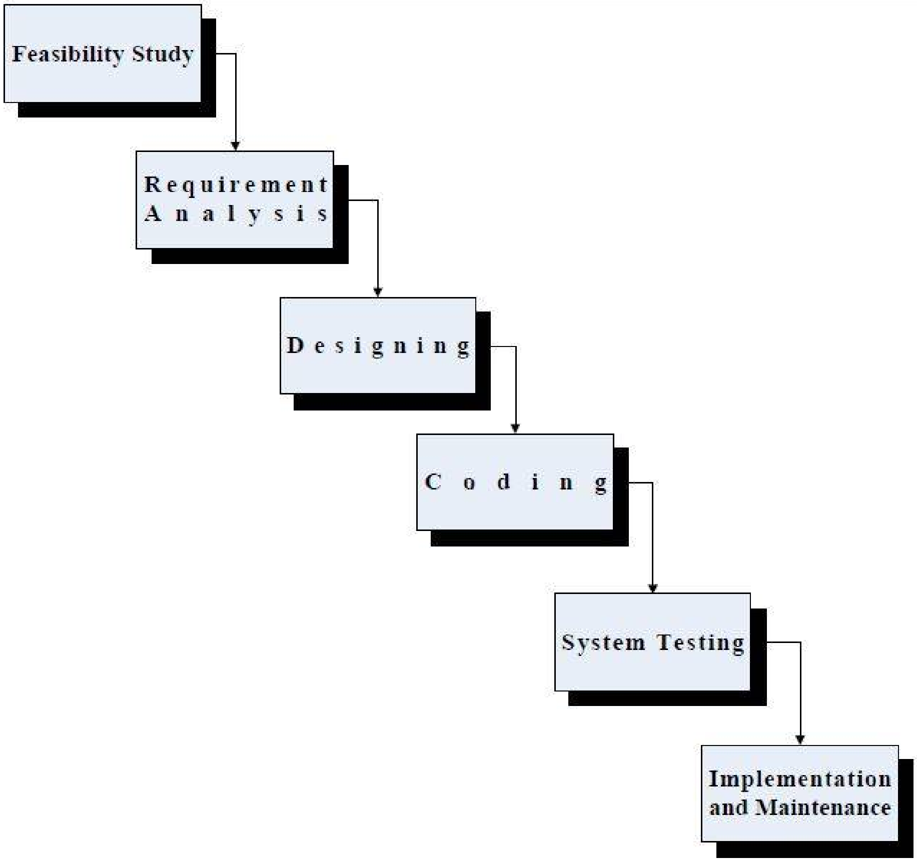


Figure 2.6 Phases of software development life cycle

3.2 Phases of Software Development Life Cycle

The software development life cycle is classically thought of as the set of activities that Market-Place project report analysts, designers and users carry out to develop and implement an information system. The software development life cycle consists of the following activities:

- Requirement Gathering.
- Feasibility Study.
- System Design.
- Coding.
- System Testing.
- Implementation and Maintenance.

Requirement Gathering

An important outcome of the requirement gathering is to determination the need of user or find the deficiency in the current working system. After getting the right need of user develop an application which fulfill the need of user and fill the fault in the previous system. In the conduct of feasibility study, there are major distinct and interrelated areas were taken into consideration.

Technical Feasibility

The System of operation which was functioning earlier was totally manual, with no kind of automation or computerization. But this system will make it automatic and easy for the parents to keep the eyes on their children progress. It successfully satisfies the user requirement.

Economical Feasibility

The computerized system is economically feasible in the sense the cost to develop the system using hardware and software. The system is user friendly so there is no need to specially train a person to open it. A person with a little knowledge of android phone and internet can use this system.

Requirements analysis:

The requirement of the system is to create a utility that will be provide desire information about the school and their infrastructure. When the user begins his journey with the application connects to his personal space, an execution of the application begins and the information is provided according to their need. A user can download an application which will be installed in his mobile device. In this scenario, the parents can also go through the fee status and attendance of his baby.

System Designing

The system is designed to satisfied all the requirements with user friendly way and it is easy to use. The system is designed Java 2 micro Edition (J2ME) and Android applications are written in Java. J2ME uses the JVM and Android has its own VM which is called Dalivik. It has a special byte code format that corresponds to Android devices requirements. The generated .dex files are smaller than the generated using usual Java byte code because the .dex files contain unique data. If several classes share the same string, the string exists only one time in the .dex file and next occurrences will be just a pointer to this string. It uses the SQLite as database to store the data.

Coding

The coding of this GEGS application is done using the android studio IDE, java xml 1.

To code with the J2ME and Android Architecture we have some new ideas and we does not use a traditional approach of android studio that is writing all the logic in controller. Instead we have written all our validation in Models so that we can easily call a single function to perform operation. We have write logics for all repetitive tasks in a parent model and inherit that in all other individual models that are written for particular tables according to tables. We have write a class to handle exceptions. We need to handle the exceptions to perform the insert and update operations which are posting data because a posted data can break validations but we don’t need to write a code for validation in every insertion and updating function. We just need to call the function inside try block and if posted data violate the validation that will be handled by catch.

System Testing

During system testing, the system is used experimentally to ensure that Graphic era global school project does not fail. Special test data are input for processing, and the result examined. We used Unified functional testing tool for the testing of this application.

Implementation, Evaluation and Maintenance:

Implementation is the process of having systems personnel check out and put new equipment into use, train users, install the new application and construct any files of data needed to use it Evaluation of the system is performed to identify its strength and weaknesses. Maintenance is necessary to eliminate errors in the working system during its working life and to tune the system to any variations in its working environment. The importance of maintenance is to continue to bring the new system to standards.

Chapter 3

Requirements Analysis

3.1 Overview

Analysis is a Fact Finding Technique where studies like User’s need, System Requirement Specifications, Feasibility Analysis and Cost-Benefit Analysis are carried out.

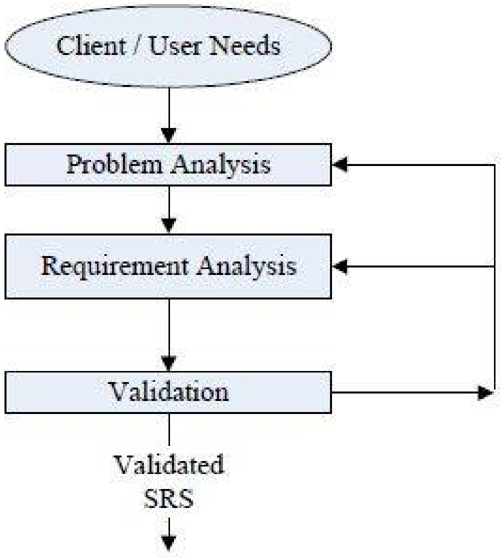


Figure 3.1 flow of information

In the education, information is obtained mainly through newspaper, magazines, radio and other simple ways those are available easily. But problem is that parents are not able to get children daily activity timely when they are in the home or office. While today's mobile devices are becoming more intelligent, compared with PC, they still have the following limitations like small screen and tiny keyboard, limited CPU capacity, limited memory space, slow and fitful Internet connection. But the application on these mobiles works slow due to continues acquisition of the bandwidth. Therefore, the mobile end - user’s operation is very difficult, and the contents display on the screen of mobile device is limited.

Graphic era global school Application to develop detailed texts, pictures, and other guidance information are provided, and so people can better understand the icon and make decision objectively. This section analyses ImageView, Activity and presents a simple design example to show the detailed information on Android. ImageView is used to display a view of the Images. It can accept the keyboard events.

3.2 Objective of Requirement Analysis

Requirement analysis was conducted with the following objectives in mind:

- Identification of need.
- Information Gathering
- Evaluate the system concept of feasibility

3.2.1 Identification of need

First we have to identify the need of the application**.** In this application a user can find attraction point at the bases of interest (Student Info, Facility Info, Gallery, Academic Calendar, Attendance and Notice).

3.2.2 Information Gathering

First we need to gather the information about the application what are problems with the existing application. There are so many problems with the existing application is that tourists are not able to get travel information timely when they are on the move. Many mobiles of recent decades have school application. But the application on these mobiles works slow due to continues acquisition of the bandwidth. Therefore, the mobile end - user's operation is very difficult, and the contents display on the screen of mobile device is limited. So we have develop an application which work off line also.

3.2.3 Feasibility Study:

As other application work on online mode and but my application has offline feature where user can be connected with my app weather there is internet con is not. This features will be helpful at any natural clematises.

3.3 Software Requirements Specification (SRS)

3.3.1 Purpose

Retrieves the user’s current geological coordinates. Flexible map display and control functions and location support are provided in Android for mobile system design. This module is responsible to retrieve the weather information from Google and display it to user. Weather forecasting is the application of science and technology to predict the state of the atmosphere for a given location.

3.3.2 Developer’s responsibility

The developer is responsible for:

- Developing the system.
- Installing the application on the client’s hardware.
- Maintaining the system

3.3.3 Goals and objective

The goal of the project is provide basic functionalities About Graphic era global school application such as showing a detailed information about the school, locating points of interest (POIs), retrieving the information about the children at any time. This app is view in offline mode so there is no need of internet. When a people wants to get updated information than they have to go online. It copy the database record in a temporary repository so that it can be run offline. Parents are also directly mail the class teacher for any query.

Chapter 4

System Design

4.1 Database Design

The collection of data is usually referred to as the database. The database contains the information about one particular enterprise. Database system of data involves both the definitions of structures for the storage of information, processing and mechanism for the manipulation of information. In addition, the database system provides for the safety of information stored in the database despite system crashes or attempts of unauthorized access.

4.2 Human Machine Interface Design

The design of the human machine interface in one of the most important aspects of system design. A good interface design should take into account the following factors:

- **User characteristics**

  It includes consideration of the kinds of the users who will use the equipment, their diverse backgrounds and skills, the user expectations as well as their physical characteristics. The users who possess high degree of skill often prefer more powerful functions which usually means greater complexity, unskilled operations, on the other hand, would simple functions which are easier to learn and use.

- **Task Characteristics**

  The nature of the users tasks differ and therefore the needs for specific-kinds of service from the system. For example unstructured tasks usually require a more flexible mode of interaction to meet the varying needs of users as opposed to structured tasks, which are more predictable and repetitive. The sequence and frequency with which certain tasks are performed will also affect the optimal design of the user interface.

- **Functional Characteristics**

  It refers to the various functions required to perform the tasks and the ease with which these functions can be learnt made use of by the users while ascertaining the functional characteristics, the support facilities required to perform the functions, also need to be taken into account. These include facilities like training, on line help, documentation, expert system etc. the other aspect of functional characteristics of a system its performance criteria like response time, fault tolerance etc.

- **Input Design**

  The most common cause of errors in data processing is inaccurate input data. Errors entered by data entry operators can be controlled by the input design. Input design is the process of converting user-oriented inputs to computer based formats. The goal of input design to make data entry easy logical and free from errors.

- **Output Design**

  Computers are the most important source of information to the user. Inputs are fed into computers to acquire the required outputs. The computers can provide valuable information's in the form of well-documented outputs for various values. The major form of output is a hardcopy (reports) from the printer. Reports are around the output requirements of the user.

4.3 Software Design Specification

This Graphic era global school project section provides an overview of the entire design document. This Graphic era global school project document describes all data, architectural, interface and component-level design for the software.

4.3.1 Database Design

4.3.1.1 Tables Description

S.NO.	NAME	DATA TYPE	NULL?	DEFAULT
1	Email id	Text	No	
2	Password	Text	No	
3	Admission id	Text	No	
4	Name	Text	No	
5	Class section	Text	No	
6	Class teacher	Text	No	
7	Father	Text	No	
8	Mother	Text	No	
9	Phone	Integer	No	

Table 4.1 (a) This table used to store the data about the profile information. Primary key of this table is Email id.

S.NO.	NAME	DESCRIPTION
1	Email id	Store the email of user
2	Password	Store the password
3	Admission id	Store the admission id given by school
4	Name	Name of children
5	Class section	Child class and section

6	Class teacher	Class teacher name
7	Father	Father’s name of child
8	Mother	Mother’s name of child
9	Phone	Contact number of parents

Table 4.1 (b) This table describe the detail of field

S.NO.	NAME	DATA TYPE	NULL?	DEFAULT
1	Admission id	Text	No	
2	Total class	Integer	No	
3	Class done	Integer	No	
4	Class attendance	Integer	No	
5	Percentage	Integer	No	

Table 4.2 (a) This table used to store the data about the attendance. Primary key of this table is Admission id.

S.NO.	NAME	DESCRIPTION
1	Admission id	Store the admission id of user given by school
2	Total class	Total class will be done in academic year
3	Class done	Total class done
4	Class attendance	Class attend by child

5	Percentage	Percentage of attendance

Table 4.2 (b) This table describe the detail of field of attendance table

S.NO.	NAME	DATA TYPE	NULL?	DEFAULT
1	Admission id	Text	No	
2	Total fee	Integer	No	
3	Paid fee	Integer	No	
4	Due fee	Integer	No	

Table 4.3 (a) This table used to store the data about the Fee status. Primary key of this table is Admission id.

S.NO.	NAME	DESCRIPTION
1	Admission id	Store the admission id of user given by school
2	Total fee	Total fee will be pay in academic year
3	Paid fee	Total fee paid
4	Due fee	Due fee

Table 4.3 (b) This table describe the detail of field of fee

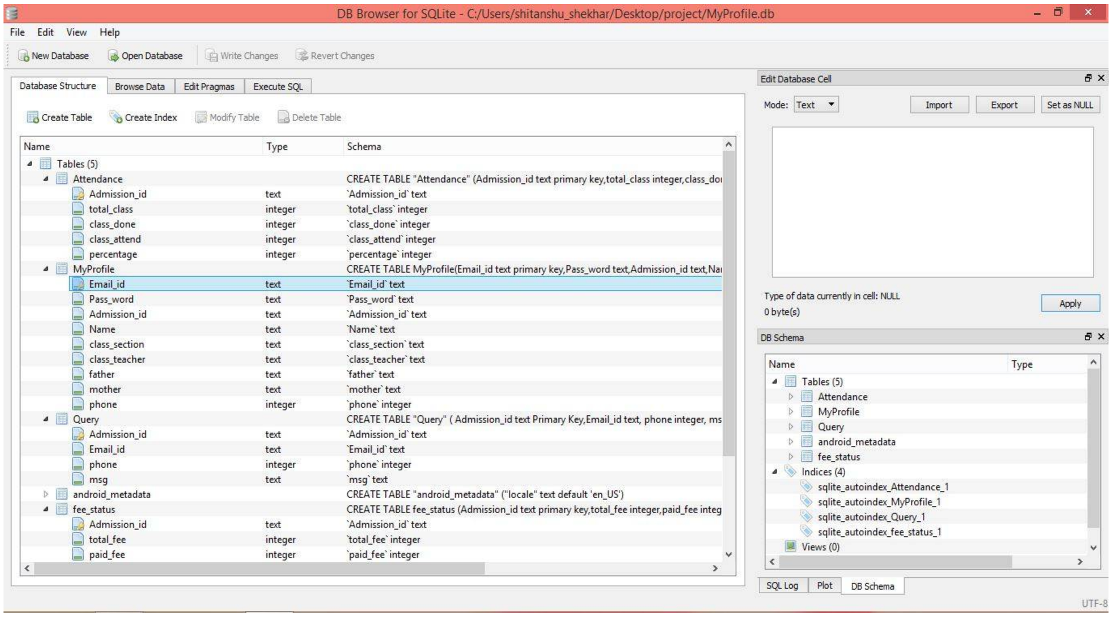
S.NO.	NAME	DATA TYPE	NULL?	DEFAULT
1	Admission id	Text	No	
2	Email id	Text	No	

3	Phone	Integer	No	
4	Message	Text	No	

Table 4.4 (a) This table used to store the data about the query. Primary key of this table is Admission id.

S.NO.	NAME	DESCRIPTION
1	Admission id	Store the admission id of user given by school
2	Total fee	Total fee will be pay in academic year
3	Paid fee	Total fee paid
4	Due fee	Due fee

Table 4.4 (b) This table describe the detail of field of query



Name	Type	Schema
Tables (5)		
Attendance		CREATE TABLE "Attendance" (Admission_id text primary key,total_class integer,class_do
Admission_id	text	\`Admission_id\` text
total_class	integer	\`total_class\` integer
class_done	integer	\`class_done\` integer
class_attend	integer	\`class_attend\` integer
percentage	integer	\`percentage\` integer
MyProfile		CREATE TABLE MyProfile(Email_id text primary key,Pass_word text,Admission_id text,Na
Email_id	text	\`Email_id\` text
Pass_word	text	\`Pass_word\` text
Admission_id	text	\`Admission_id\` text
Name	text	\`Name\` text
class_section	text	\`class_section\` text
class_teacher	text	\`class_teacher\` text
father	text	\`father\` text
mother	text	\`mother\` text
phone	integer	\`phone\` integer
Query		CREATE TABLE "Query" (Admission_id text Primary Key,Email_id text, phone integer, ms
Admission_id	text	\`Admission_id\` text
Email_id	text	\`Email_id\` text
phone	integer	\`phone\` integer
msg	text	\`msg\` text
android_metadata		CREATE TABLE "android_metadata" ("locale" text default 'en_US')
fee_status		CREATE TABLE fee_status (Admission_id text primary key,total_fee integer,paid_fee integ
Admission_id	text	\`Admission_id\` text
total_fee	integer	\`total_fee\` integer
paid_fee	integer	\`paid_fee\` integer

Type of data currently in cell: NULL
0 byte(s)

Name	Type
Tables (5)	
Attendance	
MyProfile	
Query	
android_metadata	
fee_status	
Indices (4)	
sqlite_autoindex_Attendance_1	
sqlite_autoindex_MyProfile_1	
sqlite_autoindex_Query_1	
sqlite_autoindex_fee_status_1	
Views (0)	

Figure 4.1 Database screen for SQLite

4.3.2 Process Model

A Process Model tells us about how the data is processed and how the data flows from one table to another to gather the required information. This GEGS Android Application project model consists of the Functional Decomposition Diagram and Data Flow Diagram.

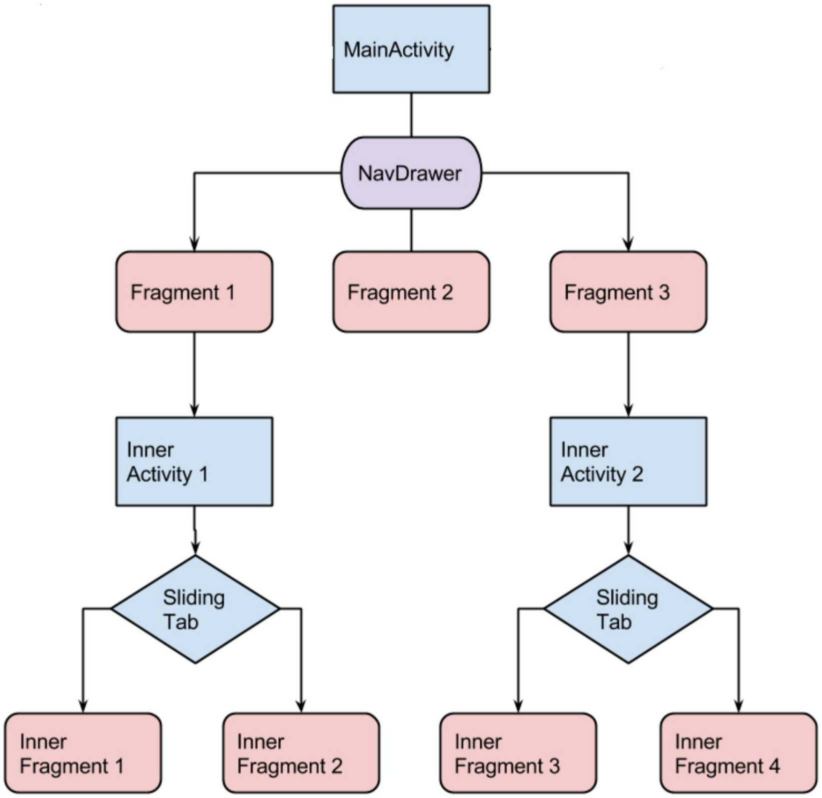


Figure 4.2 Functional Decomposition Diagram

4.3.4 Data Flow Diagram

Data Flow Diagrams show the flow of data from external entities into the system, and from one process to another within the system. There are four symbols for drawing a DFD

- Rectangles representing external entities, which are sources or destinations of data.

- Ellipses representing processes, which take data as input, validate and process it and output it.

- Arrows representing the data flows, which can either, be electronic data or Physical items.
- Open-ended rectangles or a Disk symbol representing data stores, including electronic stores such as databases or XML files and physical stores such as filing cabinets or stacks of paper.

Data Flow Diagrams for the current system. Each process within the system is first shown as a Context Level DFD and later as a Detailed DFD. The Context Level DFD provides a conceptual view of the process and its surrounding input, output and data stores. The Detailed DFD provides a more detailed and comprehensive view of the interaction among the sub-processes within the system.

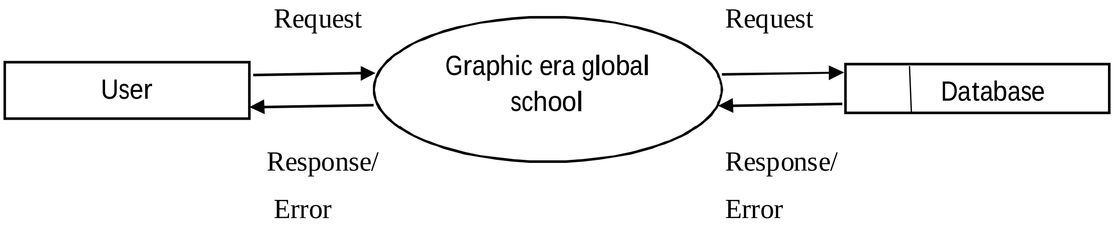


Figure 4.3 0 Level DFD

4.3.5 Flow chart

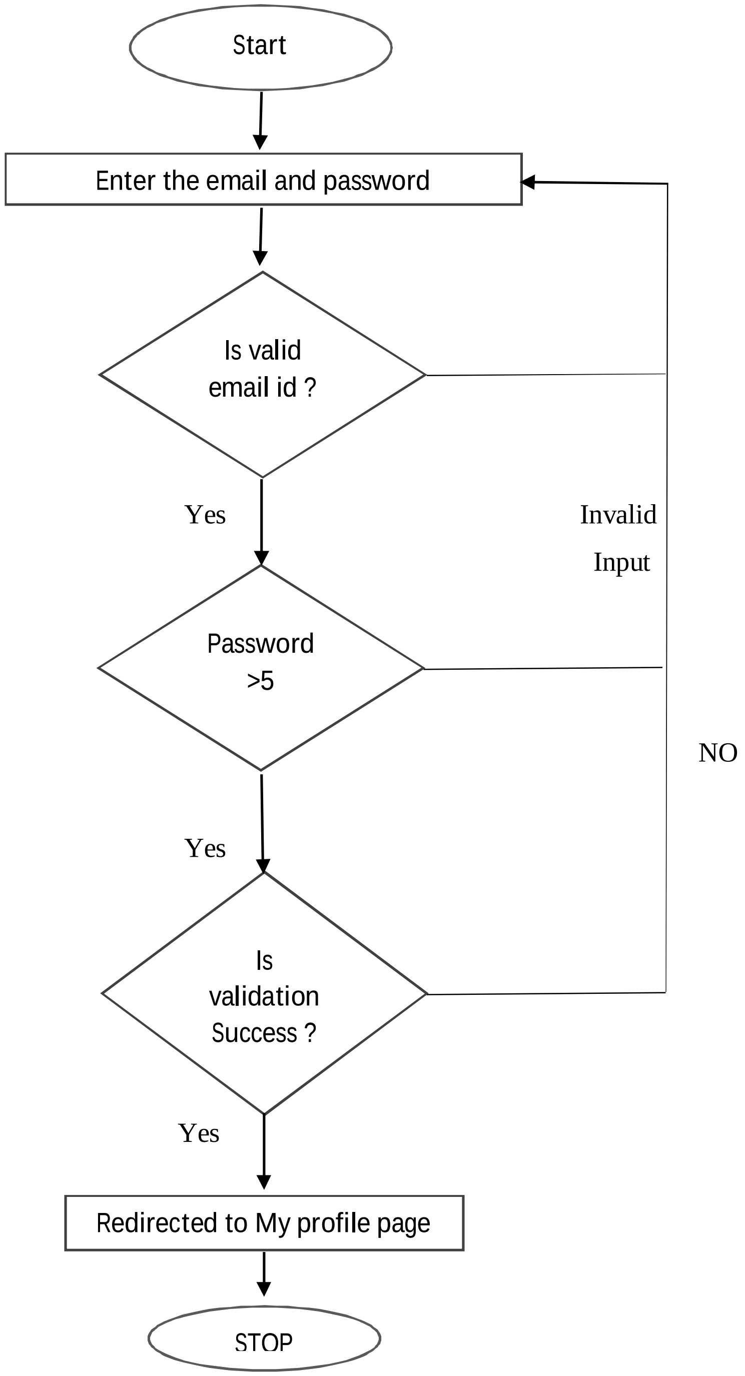


Figure 4.4 Flow diagram of login validation

Chapter 5

Coding

5.1 Introduction

The coding of this GEGS is done using the Android Studio 2.3, java, xml and SQLite for the client side scripting.

We have write a class to handle exceptions. We need to handle the exceptions to perform the insert and update operations which are posting data because a posted data can break validations but we don't need to write a code for validation in every insertion and updating function. We just need to call the function inside try block and if posted data violate the validation that will be handled by catch.

5.2 Android Architecture

Android architecture or **Android software stack** is categorized into five parts:

- Linux kernel
- Native libraries (middleware)
- Android Runtime
- Application Framework
- Application

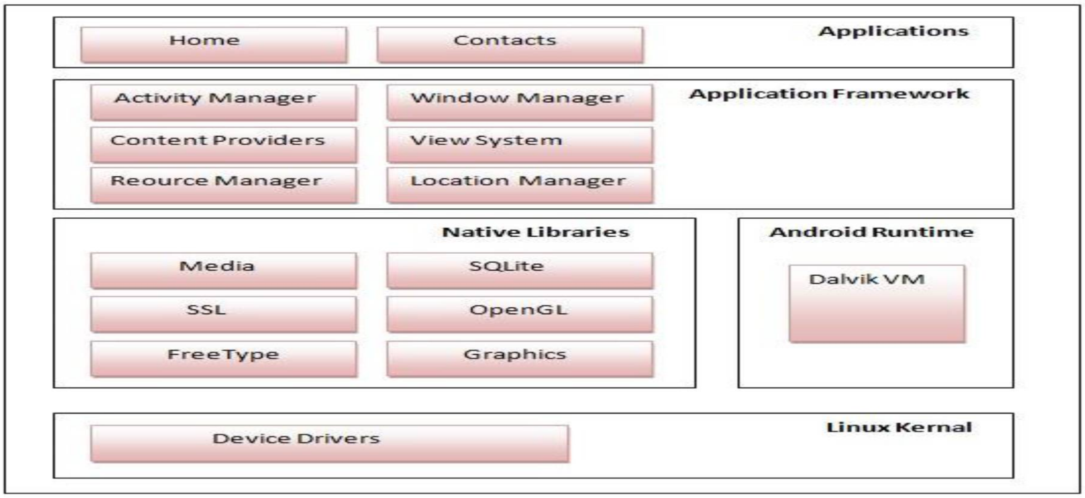


Figure 5.1 Android Architecture

- **Linux kernel**

  It is the heart of android architecture that exists at the root of android architecture. **Linux kernel** is responsible for device drivers, power management, memory management, device management and resource access.

- **Native libraries**

  On the top of Linux kernel, there are **Native libraries** such as WebKit, OpenGL, Free Type, SQLite, Media, C runtime library (libc) etc.

  The WebKit library is responsible for browser support; SQLite is for database, Free Type for font support, Media for playing and recording audio and video formats.

- **Android Runtime**

  In android runtime, there are core libraries and DVM (Dalvik Virtual Machine) which is responsible to run android application. DVM is like JVM but it is optimized for mobile devices. It consumes less memory and provides fast performance.

- **Application Framework**

  On the top of Native libraries and android runtime, there is android framework. Android framework includes **Android API's** such as UI (User Interface), telephony, resources, locations, Content Providers (data) and package managers. It provides a lot of classes and interfaces for android application development.

- **Application**

  On the top of android framework, there are applications. All applications such as home, contact, settings, games, browsers are using android framework that uses android runtime and libraries. Android runtime and native libraries are using Linux kernel.

5.3 Activity Life Cycle

If you have worked with C, C++ or Java programming language then you must have seen that your program starts from **main()** function. Very similar way, Android system initiates its program within an **Activity** starting with a call on *onCreate()* callback method. There is a sequence of callback methods that start up an activity and a sequence of callback methods that tear down an activity as shown in the below Activity life cycle diagram:

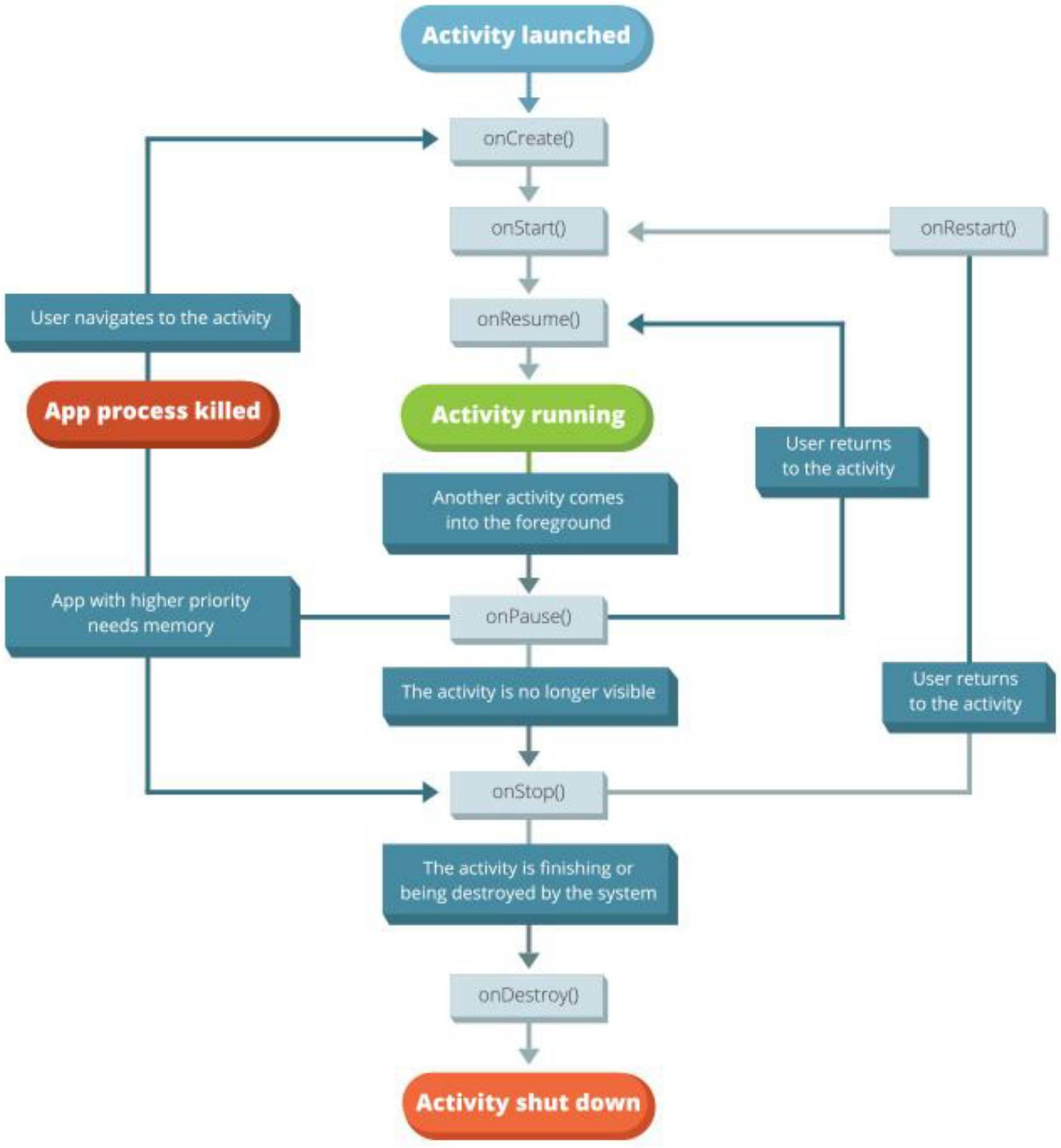

Activity launched
onCreate()
onStart()
onRestart()
User navigates to the activity
onResume()
App process killed
Activity running
User returns to the activity
Another activity comes into the foreground
App with higher priority needs memory
onPause()
The activity is no longer visible
User returns to the activity
onStop()
The activity is finishing or being destroyed by the system
onDestroy()
Activity shut down

Figure 5.2 Activity Life Cycle

- **onCreate()** This is the first callback and called when the activity is first created.
- **onStart()** This callback is called when the activity becomes visible to the user.
- **onResume()** This is called when the user starts interacting with the application.
- **onPause()** The paused activity does not receive user input and cannot execute any code and called when the current activity is being paused and the previous activity is being resumed.
- **onStop()** This callback is called when the activity is no longer visible.
- **onDestroy()** This callback is called before the activity is destroyed by the system.
- **onRestart()**This callback is called when the activity restarts after stopping it.

5.3.1 AndroidManifest.xml

Every application must have an AndroidManifest.xml file (with precisely that name) in its root directory. The manifest file presents essential information about your app to the Android system, information the system must have before it can run any of the app's code. Among other things, the manifest does the following:

- It names the Java package for the application. The package name serves as a unique identifier for the application.
- It describes the components of the application — the activities, services, broadcast receivers, and content providers that the application is composed of. It names the classes that implement each of the components and publishes their capabilities (for example, which Intent messages they can handle). These declarations let the Android system know what the components are and under what conditions they can be launched.
- It determines which processes will host application components.
- It declares which permissions the application must have in order to access protected parts of the API and interact with other applications.

- It also declares the permissions that others are required to have in order to interact with the application's components.
- It lists the Instrumentation classes that provide profiling and other information as the application is running. These declarations are present in the manifest only while the application is being developed and tested; they're removed before the application is published.
- It declares the minimum level of the Android API that the application requires.
- It lists the libraries that the application must be linked against.

5.3.2 Source of Manifest File

```
<?xml version="1.0" encoding="utf-8"?>

<manifest>

    <uses-permission />
    <permission />
    <permission-tree />
    <permission-group />
    <instrumentation />
    <uses-sdk />
    <uses-configuration />
    <uses-feature />
    <supports-screens />
    <compatible-screens />
    <supports-gl-texture />

    <application>

        <activity>
            <intent-filter>
                <action />
                <category />
                <data />
            </intent-filter>
            <meta-data />
        </activity>

        <activity-alias>
            <intent-filter> . . . </intent-filter>
            <meta-data />
        </activity-alias>
        <service>
            <intent-filter> . . . </intent-filter>
            <meta-data/>
        </service>

        <receiver>
            <intent-filter> . . . </intent-filter>
            <meta-data />
        </receiver>

        <provider>
            <grant-uri-permission />
            <meta-data />
            <path-permission />
        </provider>

        <uses-library />

    </application>

</manifest>
```

An application can have one or more activities without any restrictions. Every activity you define for your application must be declared in your*AndroidManifest.xml* file and the main activity for your app must be declared in the manifest with an <intent-filter> that includes the MAIN action and LAUNCHER category.

5.3.3 Code

We have write all main logic and functions in app model and extend that model in all the individual models so now we see that code of app model:

Application Module

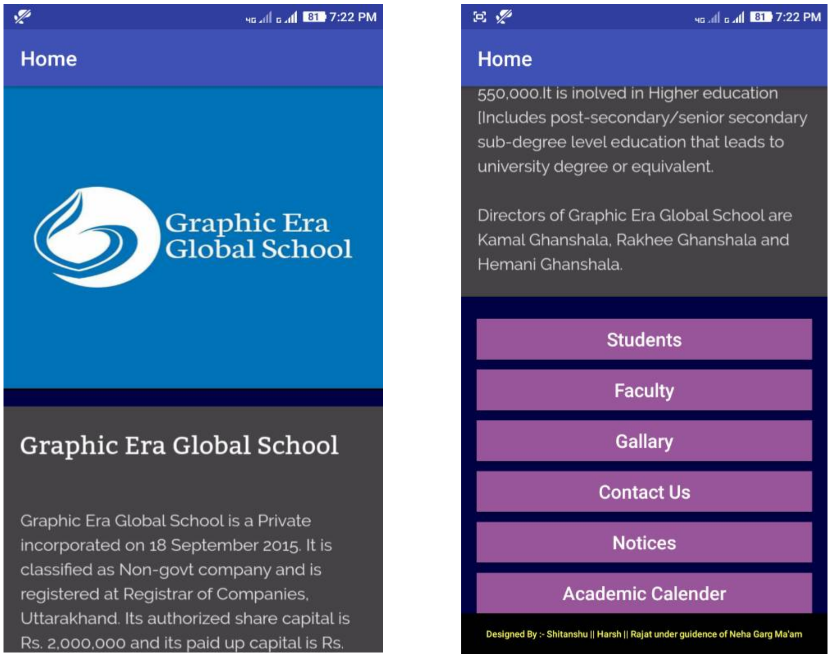
activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView 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/scrollView1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_gravity="center"
    android:fillViewport="true"
    android:adjustViewBounds="true"
    android:background="#004"
    tools:context="com.shitanshu_shekhar.myapplication.MainActivity">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <ImageView
            android:id="@+id/iv_bollywood"
            android:layout_width="wrap_content"
            android:layout_height="300dp"
            android:layout_alignParentTop="false"
            android:layout_centerHorizontal="true"
            android:background="@drawable/gegs" />

        <ImageView
            android:id="@+id/iv_hollywood"
            android:layout_width="wrap_content"
            android:layout_height="490dp"
            android:src="@drawable/gegs5"
            android:layout_below="@+id/iv_bollywood"
            android:layout_alignParentStart="true" />

        <Button
            android:id="@+id/button"
            android:layout_width="330dp"
            android:layout_height="40dp"
            android:background="#959"
            android:clickable="?attr/isLightTheme"
            android:onClick=""
            android:text="Students"
            android:textAlignment="gravity"
            android:textAllCaps="false"
            android:textColor="#ffffff"
            android:textSize="10pt"
            android:layout_below="@+id/iv_hollywood"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="6dp" />

        <Button
            android:id="@+id/button1"
            android:layout_width="330dp"
            android:layout_height="40dp"
            android:background="#959"
            android:clickable="?attr/isLightTheme"
            android:onClick=""
            android:text="Faculty"
            android:textAlignment="gravity"
            android:textAllCaps="false"
            android:textColor="#ffffff"
            android:textSize="10pt"
            android:layout_marginTop="10dp"
            android:layout_below="@+id/button"
            android:layout_alignStart="@+id/button" />

        <Button
            android:id="@+id/button2"
            android:layout_width="330dp"
            android:layout_height="40dp"
            android:text="Gallary"
            android:textAllCaps="false"
            android:textSize="10pt"
            android:background="#959"
            android:textColor="#ffffff"
            android:layout_marginTop="10dp"
            android:layout_below="@+id/button1"
            android:layout_alignStart="@+id/button1"
            android:onClick=""/>

        <Button
            android:id="@+id/button3"
            android:layout_width="330dp"
            android:layout_height="40dp"
            android:layout_marginTop="10dp"
            android:background="#959"
            android:clickable="?attr/isLightTheme"
            android:text="Contact Us"
            android:textAllCaps="false"
            android:textColor="#ffffff"
            android:textSize="10pt"
            android:layout_below="@+id/button2"
            android:layout_alignStart="@+id/button2" />

        <Button
            android:id="@+id/button4"
            android:layout_width="330dp"
            android:layout_height="40dp"
            android:text="Notices"
            android:textAllCaps="false"
            android:textSize="10pt"
            android:background="#959"
            android:textColor="#ffffff"
            android:layout_marginTop="10dp"
            android:layout_below="@+id/button3"
            android:layout_alignStart="@+id/button1"
            android:onClick=""/>

        <Button
            android:id="@+id/button5"
            android:layout_width="330dp"
            android:layout_height="40dp"
            android:text="Academic Calender"
            android:textAllCaps="false"
            android:textSize="10pt"
            android:background="#959"
            android:textColor="#ffffff"
```

```
            android:layout_marginTop="10dp"
            android:layout_below="@+id/button4"
            android:layout_alignStart="@+id/button1"
            android:onClick=""/>

        <Button
            android:id="@+id/editText"
            android:layout_width="wrap_content"
            android:layout_height="40dp"
            android:ems="10"
            android:inputType="textPersonName"
            android:text="Designed By :- Shitanshu || Harsh || Rajat under
guidence of Neha Garg Ma'am"
            android:background="#000"
            android:textColor="#ff5"
            android:textSize="1.7mm"
            android:textAllCaps="false"
            android:clickable="false"
            android:enabled="false"
            android:layout_alignParentBottom="true"
            android:layout_alignParentStart="true"
            android:layout_below="@id/button5"
            android:layout_alignParentEnd="true" />


    </RelativeLayout>


</ScrollView>
```

MainActivity.java

```
package com.shitanshu_shekhar.myapplication;

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

public class MainActivity extends AppCompatActivity {

    Button btn,btn2,btn3,btn4,btn5,btn6;
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        btn=(Button)findViewById(R.id.button);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i=new Intent(MainActivity.this,LoginActivity.class);
                startActivity(i);
            }
        });

        btn2=(Button)findViewById(R.id.button2);
        btn2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i=new Intent(MainActivity.this,gallery.class);
                startActivity(i);
            }
        });

        btn3=(Button)findViewById(R.id.button1);
        btn3.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i=new Intent(MainActivity.this,faculty.class);
                startActivity(i);
            }
        });

        btn4=(Button)findViewById(R.id.button3);
        btn4.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i=new Intent(MainActivity.this,contact.class);
                startActivity(i);
            }
        });

        btn5=(Button)findViewById(R.id.button4);
        btn5.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i=new Intent(MainActivity.this,notice.class);
                startActivity(i);
            }
        });

        btn6=(Button)findViewById(R.id.button5);
        btn6.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i=new Intent(MainActivity.this,calender.class);
                startActivity(i);
            }
        });

    }


}
```



Graphic Era Global School

Graphic Era Global School is a Private incorporated on 18 September 2015. It is classified as Non-govt company and is registered at Registrar of Companies, Uttarakhand. Its authorized share capital is Rs. 2,000,000 and its paid up capital is Rs.

550,000.It is inolved in Higher education [Includes post-secondary/senior secondary sub-degree level education that leads to university degree or equivalent.

Directors of Graphic Era Global School are Kamal Ghanshala, Rakhee Ghanshala and Hemani Ghanshala.

Designed By :- Shitanshu || Harsh || Rajat under guidence of Neha Garg Ma'am

Figure 5.3 Home Screen

activity_login.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/scrollView5"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_gravity="center"
    android:fillViewport="true"
    android:adjustViewBounds="true"
    tools:context="com.shitanshu_shekhar.myapplication.LoginActivity"
    android:background="@drawable/sadow">


    <TextView android:text = "Login" android:layout_width="wrap_content"
        android:layout_height = "wrap_content"
        android:id = "@+id/textview"
        android:textSize = "30dp"
        android:layout_alignParentTop = "true"
        android:layout_centerHorizontal = "true" />

    <TextView
        android:layout_width = "wrap_content"
        android:layout_height = "wrap_content"
        android:text = "Graphic Era Global School"
        android:id = "@+id/textView"
        android:textColor = "#ff7a"
        android:textSize = "30dp"
        android:layout_below="@+id/textview"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="10dp" />

    <EditText
        android:layout_width = "wrap_content"
        android:layout_height = "wrap_content"
        android:id = "@+id/editText"
        android:hint = "Registered Email"
        android:focusable = "true"
        android:textColorHighlight = "#ff7eff15"
        android:textColorHint = "#985"
        android:layout_below="@+id/textView"
        android:layout_alignStart="@+id/textView"
        android:layout_marginTop="15dp"
        android:layout_alignEnd="@+id/textView" />

    <EditText
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:ems="10"
        android:id="@+id/editText2"
        android:layout_below="@+id/editText"
        android:layout_alignRight="@+id/editText"
        android:layout_alignEnd="@+id/editText"
        android:textColorHint="#985"
        android:hint="Password"
        android:layout_alignStart="@+id/editText" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Attempts Left:"
        android:id="@+id/textView2"
        android:textSize="25dp"
        android:textColor="#235"
        android:layout_below="@+id/editText2"
        android:layout_alignStart="@+id/editText2"
        android:layout_marginTop="12dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="New Text"
        android:id="@+id/textView3"
        android:layout_alignTop="@+id/textView2"
        android:layout_alignParentRight="true"
        android:layout_alignParentEnd="true"
        android:layout_alignBottom="@+id/textView2"
        android:layout_toEndOf="@+id/textview"
        android:textSize="25dp"
        android:layout_toRightOf="@+id/textview" />

    <Button

        android:text="Login"
        android:textColor="#fff"
        android:textStyle="bold"
        android:background="#959"
        android:layout_width="100dp"
        android:layout_height="35dp"
        android:textAllCaps="false"
        android:id="@+id/button"
        android:layout_below="@+id/textView2"
        android:layout_centerHorizontal="true"
```

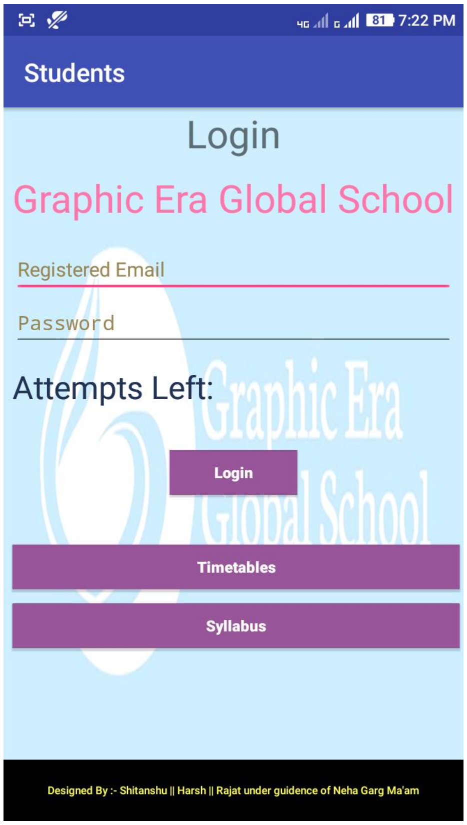
```
        android:layout_marginTop="33dp" />

    <Button
        android:id="@+id/button9"
        android:layout_width="350dp"
        android:layout_height="35dp"
        android:background="#959"
        android:onClick=""
        android:text="Timetables"
        android:textAllCaps="false"
        android:textColor="#fff"
        android:textStyle="bold"
        android:layout_marginTop="39dp"
        android:layout_below="@+id/button"
        android:layout_alignStart="@+id/textView2" />

    <Button
        android:id="@+id/button10"
        android:layout_width="350dp"
        android:layout_height="35dp"
        android:background="#959"
        android:onClick=""
        android:text="Syllabus"
        android:textAllCaps="false"
        android:textColor="#fff"
        android:textStyle="bold"
        android:layout_below="@+id/button9"
        android:layout_alignStart="@+id/button9"
        android:layout_marginTop="11dp" />

    <Button
        android:id="@+id/button6"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:text="Designed By :- Shitanshu || Harsh || Rajat under guidence
of Neha Garg Ma'am"
        android:layout_alignParentStart="true"
        android:layout_alignParentEnd="true"
        android:textColor="#ff5"
        android:textAllCaps="false"
        android:textSize="1.6mm"
        android:background="#000"
        android:clickable="false"/>


</RelativeLayout>
```

LoginActivity.java

```
package com.shitanshu_shekhar.myapplication;

import android.content.Intent;
import android.graphics.Color;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;import java.util.regex.Matcher;
import java.util.regex.Pattern;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.AutoCompleteTextView;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import java.lang.*;

import static android.Manifest.permission.READ_CONTACTS;

/**
 * A login screen that offers login via email/password.
 */
public class LoginActivity extends AppCompatActivity {


    Button b1,b2;
    EditText ed1,ed2;

    TextView tx1,tx2;
    int counter =5,i;
    int warning=10;

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

        b1 = (Button)findViewById(R.id.button);
        ed1 = (EditText)findViewById(R.id.editText);
        ed2 = (EditText)findViewById(R.id.editText2);

        b2 = (Button)findViewById(R.id.button2);
        tx1 = (TextView)findViewById(R.id.textView3);
        tx1.setVisibility(View.GONE);


        Button btn1, btn10;
        btn1 = (Button) findViewById(R.id.button9);
        btn1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i = new Intent(LoginActivity.this, time_table.class);
                startActivity(i);
            }
        });

        btn10 = (Button) findViewById(R.id.button10);
        btn10.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i = new Intent(LoginActivity.this, syllabus.class);
                startActivity(i);
            }
        });

        findViewById(R.id.button).setOnClickListener(new OnClickListener() {

            @Override
            public void onClick(View arg0) {

                final String email = ed1.getText().toString();
                if (!isValidEmail(email)) {
                    ed1.setError("Invalid Email");
                }

                final String pass = ed2.getText().toString();
                if (!isValidPassword(pass)) {
```

```
          ed2.setError("Invalid Password");
        }

        if(ed1.getText().toString().equals("abc@gmail.com") &&
            ed2.getText().toString().equals("admin1")) {

Toast.makeText(getApplicationContext(),"Redirecting...",Toast.LENGTH_SHORT).sho
w();
            Intent i = new
Intent(LoginActivity.this,Profile1.class);
            startActivity(i);
        }
        else if(ed1.getText().toString().equals("abc2@gmail.com") &&
            ed2.getText().toString().equals("admin2")) {

Toast.makeText(getApplicationContext(),"Redirecting...",Toast.LENGTH_SHORT).sho
w();
            Intent i = new
Intent(LoginActivity.this,Profile2.class);
            startActivity(i);
        }
        else if(ed1.getText().toString().equals("abc3@gmail.com") &&
            ed2.getText().toString().equals("admin3")) {

Toast.makeText(getApplicationContext(),"Redirecting...",Toast.LENGTH_SHORT).sho
w();
          Intent i = new Intent(LoginActivity.this,Profile3.class);
          startActivity(i);
        }
        else
          {
          Toast.makeText(getApplicationContext(), "Wrong
Credentials",Toast.LENGTH_SHORT).show();
          tx1.setVisibility(View.VISIBLE);
          tx1.setBackgroundColor(Color.RED);
          counter--;
          tx1.setText(Integer.toString(counter));

          if (counter == 0) {
            b1.setEnabled(false);
            Toast.makeText(getApplicationContext(),"Try after 10
second",Toast.LENGTH_LONG).show();
          }
        }

      }
    });

  }


  // validating email id
  private boolean isValidEmail(String email) {
    String EMAIL_PATTERN = "^[_A-Za-z0-9-\\+]+(\\.[_A-Za-z0-9-]+)*@"
        + "[A-Za-z0-9-]+(\\.[A-Za-z0-9]+)*(\\.[A-Za-z]{2,})$";

    Pattern pattern = Pattern.compile(EMAIL_PATTERN);
    Matcher matcher = pattern.matcher(email);
    return matcher.matches();
  }

  // validating password with retype password
  private boolean isValidPassword(String pass) {
    if (pass != null && pass.length() > 5) {
      return true;
```

```
        }
        return false;

    }
}
```



Designed By :- Shitanshu || Harsh || Rajat under guidence of Neha Garg Ma'am

Figure 5.4 Login Page

activity_profile.xml

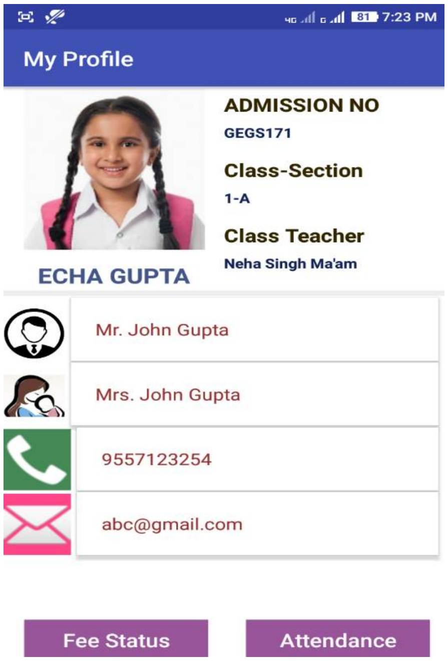
```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.shitanshu_shekhar.myapplication.Profile1">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="550dp">

    <RelativeLayout
        android:id="@+id/relative_layout1"
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:background="#fff">

        <RelativeLayout
            android:layout_width="180dp"
            android:layout_height="match_parent"
            android:layout_alignParentBottom="true"
            android:id="@+id/relativeLayout">

            <ImageButton
                android:layout_width="150dp"
                android:layout_height="160dp"
                android:layout_alignParentTop="true"
                android:layout_centerHorizontal="true"
                android:src="@drawable/profile1_pic"
                android:id="@+id/imageButton" />

            <TextView
                android:layout_width="match_parent"
                android:layout_height="30dp"
                android:text="ECHA GUPTA"
                android:gravity="center"
                android:textColor="#458"
                android:textSize="24sp"
                android:textStyle="bold"
                android:layout_alignParentBottom="true"
                android:layout_alignParentEnd="true" />

        </RelativeLayout>

        <RelativeLayout
            android:layout_width="210dp"
            android:layout_height="match_parent"
            android:layout_alignParentTop="true"
            android:layout_toEndOf="@+id/relativeLayout">

            <TextView
                android:layout_width="180dp"
                android:layout_height="30dp"
                android:layout_alignParentTop="true"
                android:layout_centerHorizontal="true"
                android:layout_marginTop="5dp"
                android:text="ADMISSION NO"
                android:textColor="#312500"
                android:textSize="21sp"
                android:textStyle="bold"
                android:id="@+id/textView13" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="GEGS171"
                android:textColor="#125"
                android:textSize="15sp"
                android:textStyle="bold"
                android:layout_marginTop="1dp"
                android:id="@+id/textView15"
                android:layout_below="@+id/textView13"
                android:layout_alignStart="@+id/textView14" />
            <TextView
                android:layout_width="wrap_content"
                android:layout_height="30dp"
                android:text="Class-Section"
                android:textColor="#312500"
                android:textSize="21sp"
                android:textStyle="bold"
                android:id="@+id/textView14"
                android:layout_marginTop="15dp"
                android:layout_below="@+id/textView15"
                android:layout_alignStart="@+id/textView13" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="1-A"
                android:textColor="#125"
                android:textSize="15sp"
                android:textStyle="bold"
                android:layout_marginTop="1dp"
                android:id="@+id/textView16"
                android:layout_below="@+id/textView14"
                android:layout_alignStart="@+id/textView15" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="30dp"
                android:text="Class Teacher"
                android:textColor="#312500"
                android:textSize="21sp"
                android:textStyle="bold"
                android:layout_marginTop="15dp"
                android:id="@+id/textView17"
                android:layout_below="@+id/textView16"
                android:layout_alignStart="@+id/textView16" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Neha Singh Ma'am"
                android:textColor="#125"
                android:textSize="15sp"
                android:textStyle="bold"
                android:id="@+id/textView18"
                android:layout_marginTop="1dp"
                android:layout_below="@+id/textView17"
                android:layout_alignStart="@+id/textView17" />

        </RelativeLayout>

    </RelativeLayout>
        <RelativeLayout
            android:layout_width="fill_parent"
            android:layout_height="420dp"
            android:orientation="vertical"
            android:background="#fff"
            android:id="@+id/linearLayout"
            android:layout_below="@+id/relative_layout1"
            android:layout_alignParentStart="true"
            android:layout_marginTop="5dp">

            <TextView
                android:layout_width="300dp"
                android:layout_height="wrap_content"
                android:background="#fff"
                android:elevation="4dp"
                android:padding="20dp"
                android:text="Mr. John Gupta"
                android:textSize="3mm"
                android:id="@+id/textView8"
                android:layout_marginTop="2dp"
                android:textColor="#963231"
                android:layout_marginLeft="5dp"
                android:layout_marginRight="5dp"
                android:layout_alignParentTop="true"
                android:layout_alignParentEnd="true" />

            <TextView
                android:layout_width="300dp"
                android:layout_height="wrap_content"
                android:background="#fff"
                android:clickable="true"
                android:elevation="4dp"
                android:padding="20dp"
                android:text="Mrs. John Gupta"
                android:textSize="3mm"
                android:layout_marginTop="2dp"
                android:textColor="#963231"
                android:layout_marginLeft="5dp"
                android:layout_marginRight="5dp"
                android:layout_below="@+id/textView8"
                android:layout_alignParentEnd="true"
                android:id="@+id/textView9" />

            <TextView
                android:layout_width="300dp"
                android:layout_height="wrap_content"
                android:background="#fff"
                android:layout_marginTop="2dp"
                android:clickable="true"
                android:elevation="4dp"
                android:padding="20dp"
                android:text="9557123254"
                android:textSize="3mm"
                android:textColor="#963231"
                android:layout_marginLeft="5dp"
                android:layout_marginRight="5dp"
                android:id="@+id/textView10"
                android:layout_below="@+id/textView9"
                android:layout_alignStart="@+id/textView9"
                android:layout_alignParentEnd="true" />

            <TextView
                android:layout_width="300dp"
                android:layout_height="wrap_content"
                android:background="#fff"
                android:clickable="true"
                android:elevation="4dp"
                android:padding="20dp"
                android:textSize="3mm"
                android:textColor="#963231"
                android:layout_marginTop="2dp"
                android:text="abc@gmail.com"
                android:layout_marginLeft="5dp"
                android:layout_marginRight="5dp"
                android:id="@+id/textView11"
                android:layout_below="@+id/textView10"
                android:layout_toEndOf="@+id/imageView20" />
```

```xml
<ImageView
    android:id="@+id/imageView17"
    android:layout_width="75dp"
    android:layout_height="60dp"
    android:layout_marginTop="5dp"
    app:srcCompat="@drawable/father_symbol"
    android:layout_alignTop="@+id/textView8"
    android:layout_toStartOf="@+id/textView9" />

<ImageView
    android:id="@+id/imageView18"
    android:layout_width="80dp"
    android:layout_height="65dp"
    android:layout_marginTop="5dp"
    app:srcCompat="@drawable/mother_symbol"
    android:layout_below="@+id/textView8"
    android:layout_toStartOf="@+id/textView9"
    android:layout_above="@+id/textView10" />

<ImageView
    android:id="@+id/imageView19"
    android:layout_width="85dp"
    android:layout_height="60dp"
    android:background="#f485"
    app:srcCompat="@android:drawable/stat_sys_phone_call"
    android:layout_alignBottom="@+id/textView10"
    android:layout_toStartOf="@+id/textView10" />

<ImageView
    android:id="@+id/imageView20"
    android:layout_width="75dp"
    android:layout_height="60dp"
    android:background="#f48"
    app:srcCompat="@android:drawable/ic_dialog_email"
    android:layout_alignBottom="@+id/textView11"
    android:layout_toStartOf="@+id/textView10" />

<TextView
    android:id="@+id/textView12"
    android:layout_width="160dp"
    android:layout_height="wrap_content"
    android:textSize="25dp"
    android:textStyle="bold"
    android:gravity="center"
    android:layout_marginTop="20dp"
    android:layout_below="@+id/textView11"
    android:layout_alignEnd="@+id/button7" />

<TextView
    android:id="@+id/textView22"
    android:layout_width="160dp"
    android:layout_height="wrap_content"
    android:textSize="25dp"
    android:textStyle="bold"
    android:gravity="center"
    android:layout_marginTop="20dp"
    android:layout_alignBaseline="@+id/textView12"
    android:layout_alignBottom="@+id/textView12"
    android:layout_toEndOf="@+id/button7"
    android:layout_marginStart="21dp" />

<Button
    android:id="@+id/button7"
    android:layout_width="150dp"
    android:layout_height="40dp"
```

```
                android:background="#959"
                android:clickable="?attr/isLightTheme"
                android:text="Fee Status"
                android:textAllCaps="false"
                android:textColor="#ffffff"
                android:textSize="10pt"
                android:layout_marginStart="17dp"
                android:layout_below="@+id/textView12"
                android:layout_alignStart="@+id/imageView20"
                android:layout_marginTop="10dp" />

            <Button
                android:id="@+id/button8"
                android:layout_marginTop="10dp"
                android:layout_width="150dp"
                android:layout_height="40dp"
                android:background="#959"
                android:clickable="?attr/isLightTheme"
                android:text="Attendance"
                android:textAllCaps="false"
                android:textColor="#ffffff"
                android:textSize="10pt"
                android:layout_above="@+id/button11"
                android:layout_alignEnd="@+id/textView22" />

            <Button
                android:id="@+id/button11"
                android:layout_width="280dp"
                android:layout_height="35dp"
                android:textColor="#fff"
                android:textSize="10pt"
                android:textAllCaps="false"
                android:background="#092156"
                android:text="***  For any query  ***"
                android:layout_marginTop="10dp"
                android:layout_below="@+id/button7"
                android:layout_centerHorizontal="true" />


        </RelativeLayout>
    </RelativeLayout>
</ScrollView>
```

Profile.java

```
package com.shitanshu_shekhar.myapplication;

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

public class Profile1 extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_profile1);

        Button b1,b2,b3;
        b1=(Button)findViewById(R.id.button7);
        b2=(Button)findViewById(R.id.button8);
        b3=(Button)findViewById(R.id.button11);
        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                TextView tx1=(TextView)findViewById(R.id.textView12);
                tx1.setText("No fee due");
            }
        });

        b2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                TextView tx2=(TextView)findViewById(R.id.textView22);
                tx2.setText("78.65%");
            }
        });

        b3.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i = new Intent(Profile1.this,query.class);
                startActivity(i);
            }
        });
    }
}
```



7:23 PM

My Profile

ADMISSION NO
GEGS171

Class-Section
1-A

Class Teacher
Neha Singh Ma'am

ECHA GUPTA

Mr. John Gupta

Mrs. John Gupta

9557123254

abc@gmail.com

Figure 5.5 Profile page

activity_query.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.shitanshu_shekhar.myapplication.query"
    android:background="@drawable/sadow"
    android:orientation="vertical">

    <TextView
        android:id="@+id/contact_form_title"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_marginBottom="16dp"
        android:layout_marginTop="5dp"
        android:text="Graphic Era Global School"
        android:textColor="#ce3232"
        android:textSize="26sp"
        android:typeface="serif" />

    <EditText
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:backgroundTint="@color/colorPrimaryDark"
        android:hint="Admission ID"
        android:inputType="textPersonName" />

    <EditText
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="10dp"
        android:layout_marginTop="10dp"
        android:backgroundTint="@color/colorPrimaryDark"
        android:hint="Email"
        android:inputType="textEmailAddress" />

    <EditText
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:backgroundTint="@color/colorPrimaryDark"
        android:hint="Phone"
        android:inputType="phone" />

    <EditText
        android:layout_width="fill_parent"
        android:layout_height="150dp"
        android:layout_marginBottom="10dp"
        android:layout_marginTop="10dp"
        android:backgroundTint="@color/colorPrimaryDark"
        android:gravity="top"
        android:hint="Your Message"
        android:fitsSystemWindows="true"
        android:breakStrategy="balanced"
        android:inputType="textMultiLine"
        android:singleLine="false"
        android:padding="5dp" />

    <Button
        android:id="@+id/sumitquery"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:background="@color/colorPrimaryDark"
        android:elevation="4dp"
        android:paddingLeft="70dp"
        android:paddingRight="70dp"
        android:text="Submit"
        android:textColor="#fff" />


</LinearLayout>
```

query.java

```
package com.shitanshu_shekhar.myapplication;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

public class query extends AppCompatActivity {

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

        Button button;
        button=(Button)findViewById(R.id.sumitquery);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Toast.makeText(getApplicationContext(),"Your Response is 
Recorded, We answered you as soon as possible",Toast.LENGTH_LONG).show();
            }
        });
    }
}
```

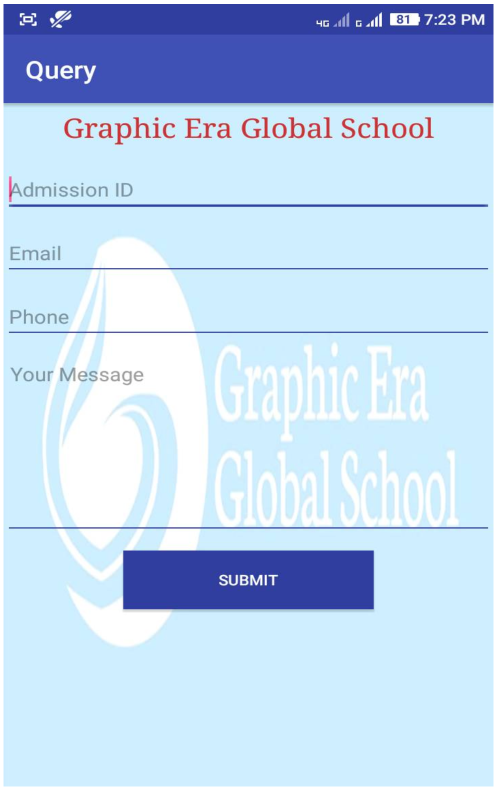


Figure 5.6 Query page for response

MyDBHandler.java

```
package com.shitanshu_shekhar.hero;

import android.database.*;
import android.content.*;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class MyDBHandler extends SQLiteOpenHelper {

    private static final int DATABASE_VERSION=1;
    private static final String DATABASE_NAME="product.db";
    private static final String TABLE_PRODUCTS="products";
    private static final String COLUMN_ID="_id";
    private static final String COLUMN_PRODUCTNAME="productname";

    public MyDBHandler(Context context, String name,
SQLiteDatabase.CursorFactory factory, int version) {
        super(context, DATABASE_NAME, factory, DATABASE_VERSION);
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        String query="CREATE TABLE"+TABLE_PRODUCTS+"("+
                COLUMN_ID+"INETEGER PRIMARY KEY AUTOINCREMENT"+
                COLUMN_PRODUCTNAME+"TEXT"+
                ");";
        db.execSQL(query);
    }

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

    public void addProduct(Products products)
    {
        ContentValues values=new ContentValues();
        values.put(COLUMN_PRODUCTNAME,products.get_productname());
        SQLiteDatabase db = getWritableDatabase();
        db.insert(TABLE_PRODUCTS,null,values);
        db.close();
    }


    public String databaseToString() {
        String dbString = "";
        SQLiteDatabase db = getWritableDatabase();
        String query = "SELECT * FROM " + TABLE_PRODUCTS + "WHERE 1";

        Cursor c = db.rawQuery(query, null);
        c.moveToFirst();

        while (!c.isAfterLast()) {
            if (c.getString(c.getColumnIndex("productname")) != null) {
                dbString += c.getString(c.getColumnIndex("productname"));
                dbString += "\n";
            }
        }
        db.close();
        return dbString;
    }
}
```

Chapter 6

Testing and Implementation

6.1 Testing

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and code generation. The increasing visibility of software as a system element and the attendant "costs" associated with a software failure are motivating forces for well-planned through testing.

Once source code has been generated, software must be tested to uncover as many errors as possible before delivery to customer. The goal is to design a series of test cases that GEGS Android Application project reports have a high likelihood of finding errors but how? That GEGS Android Application project report' where software testing techniques enter the data.

6.1.1 Testing Objectives

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that tourism system project report has a high probability of finding an as-yet-undiscovered error.

6.1.2 Testing Principle

- Exhaustive testing is not possible.
- All tests should be user requirement.
- Exhaustive testing is not possible.
- To be most effective, an independent third party should conduct testing.

6.2 Unit Testing

Unit testing focuses verification effort on the smallest unit of software design-the software component or module. Using the component – level design description as a guide, important control paths are tested to uncover errors within the boundary of the module. The relative complexity of tests and uncovered errors is limited by the constrained scope established for unit testing. The unit test is white-box oriented and the step can be conducted in parallel for multiple components.

6.2.1 Home page

S. No.	Test Case	Input	Excepted Behavior	Observed Behavior	Test Result
1	Student button is working or not?	Button Pressed	Perform send processing	Successfully Send	Success
2	Faculty button is working or not?	Button Pressed	Perform send processing	Successfully Send	Success
3	Gallery button is working or not?	Button Pressed	Perform send processing	Successfully Send	Success
4	Contact Us button is working or not?	Button Pressed	Perform send processing	Successfully Send	Success
5	Notice button is working or not?	Button Pressed	Perform send processing	Successfully Send	Success
6	Calendar button is working or not?	Button Pressed	Perform send processing	Successfully Send	Success

Table 6.1 Description of test cases of home under unit testing

6.2.2 Student Page

S. No.	Test Case	Input	Excepted Behavior	Observed Behavior	Test Result
1	Can Email id field be Null?	Null Email id field	Email id cannot be NULL	Warning" User name can't Null.	Success
2	Can password field be Null?	Null password field	Password cannot be NULL	Warning" password can't Null.	Success
3	Login button is working or not?	Button Pressed	Perform redirection processing	Successfully Send	Success

4	Time table button is working or not?	Button Pressed	Perform send processing	Successfully Send	Success
5	Syllabus button is working or not?	Button Pressed	Perform send processing	Successfully Send	Success

Table 6.2 Description of test cases of student under unit testing

6.2.3 Profile Page

S. No.	Test Case	Input	Excepted Behavior	Observed Behavior	Test Result
1	Is able to retrieve name and email, contact number etc from database	Valid Email id and password	Able to fetch data from Database	No error found during data fetching	Success
2	Attendance button is working or not?	Button Pressed	Fetch data from database and show attendance	No error found during data fetching	Success
3	Fee status button is working or not?	Button Pressed	Fetch data from database and show attendance	No error found during data fetching	Success
4	Query button is working or not?	Button Pressed	Perform send processing	Successfully Send	Success

Table 6.3 Description of test cases of profile under unit testing

6.3 Integration testing

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing.

Incremental integration is the antithesis of the big bang approach. The program is constructed and tested in small increments, where errors are easier to isolate and correct, interfaces are more likely to be tested completely, and a systematic test approach may be applied.

6.5 System testing

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. Although each test has a different purpose, all work to verify that GEGS system elements have been properly integrated and perform allocated functions.

S. No.	Test Case	Input	Excepted Behavior	Observed Behavior
1	Can user add record	Personal info of user	User personal information should be stored in Database	No modification valid authority responsible for this operation
2	Can user view record	Personal info of user	User personal information retrieve from Database	Data shown
3	Can user send email by contact us button	Button pressed	Redirected to Gmail box	Successfully done
4	Can user update record	Personal info of user	User personal information should be updated in Database	No modification valid authority responsible for this operation

Figure 6.4 Describing System test cases

6.6 Validation Checks

implements validations when we input an data when we input a data then we have to check that data is proper and has right value to store in database we have to check that data has right input and has no cross site scripting and also check for SQLite for security purpose. We put validations on input boxes because we prevent out database from fake values so we implement validation for input. For example it is compulsory to input contact number and email id during the send feedback then we put set a required validation for the user if the user left the field empty then feedback will not complete and a message will be shown that contact number and email id is required.

6.6.1 Validations for User:

- **Email:** Email is required field and it must be unique value and it should be a valid email value.
- **Password:** Password is required field and it must be unique value and it should be a valid Password value and greater than five.
- **Contact Number:** Name is also a required field.
- **Admission id:** This field is required.

Chapter 7

Conclusion and Future Scope

7.1 Conclusion

This is presents the advantages of new popular technology, Android, that will be the dominate technology. We presented already the reasons that make Android popular. These applications are still restricted to mobile resources and to the connection of the Internet. In addition they may suffer from reliability and scalability. The user can share his experience and contribute to enrich the information in the database. For this reason, the number of updates may be overloaded the server if the number of users is grown quickly. To make the system scalable, the new promising technology that will help to alleviate several problems encounter the mobile applications is cloud computing.

The system is designed to allow parent to access in online mode from the android app or offline mode from the installed application over the mobile device keep the tracks of progress of their children day by day.

This technology will conquer the poverty of mobile resources by using different services of cloud computing, it provides also flexibility and scalability to use the computing resources on-demand. Therefore, we will modify the system design and implementation to be cloudy.

7.2 Further Enhancement in Project

There is some more functions that we have not applied till now but these are common functions of marketplace software. We have put these functions in the list of further enhancement and we will exactly put these functions and implement.

- Make the application more attractive and user friendly.
- Add courses for the learning.
- Apply query management system which handle the query.
- Add chat box for messaging.

References

[1] https://stackoverflow.com/questions/32507534/ddms-cant-bind-to-local-8600-for-debugger-on-android-studio

[2] https://www.sqlite.org/lang_analyze.html

[3] https://www.tutorialspoint.com/sqlite/sqlite_pragma.htm

[4] https://developers.google.com/android/?hl=en

[5] Addison-Wesley the Android Developer's Cookbook, Building Applications with the Android SDK (2011)

[6] Android SDK: http://developer.android.com/sdk/installing.html

[7] Using Hardware Devices, Android Developers website: http://developer.android.com/guide/developing/device.html

[8] XML sample, UiO website: http://www.uio.no/studier/emner/matnat/ifi/INF3100/v11/beskjeder.xml

[9] IBM tutorial for XML parsing on Android: http://www.ibm.com/developerworks/opensource/library/x-android/

[10] https://www.youtube.com/results?search_query=android+sqlite+tutorial

[11] Automated Testing Tool: https://saas.hpe.com/en-us/software/uft