Project Details - Department of Computer Science

S.No	Reg.No	Name of the Student	Department	Title of the Project	Name of the Guide with designation
1	15BS4515	ABIRAMI R	B.Sc Computer Science	ONLINE MOBILE PHONE SALES SYSTEM	Mrs.S.SHOBANA, Head & Assistant Professor in Computer Science
2	15BS4516	AISWARIA S	B.Sc Computer Science	PANDAY EAT CALCULATE	Mrs.S.SARANYA, Assistant Professor in Computer Science
3	15BS4517	ANUPRIYA A	B.Sc Computer Science	ONLINE TRANSPORT OF LUGGAGES	Mrs.D.PAVITHRA, Assistant Professor in Computer Science
4	15BS4518	APSARA PARVEEN A	B.Sc Computer Science	MARK INFORMATION THROUGH SMS	Mrs.D.PAVITHRA, Assistant Professor in Computer Science
5	15BS4520	DELSI J	B.Sc Computer Science	INDIA TOURIST GUIDE	Ms.S.PONMALAR, Assistant Professor in Computer Science
6	15BS4521	DURAIANANDHI N	B.Sc Computer Science	FARMING ASSISTANT WEB SERVICE	Mrs.D.PAVITHRA, Assistant Professor in Computer Science
7	15BS4522	GEETHAMANI K	B.Sc Computer Science	ONLINE GATEPASS SYSTEM	Ms.E.KOKILMANI, Assistant Professor in Computer Science
8	15BS4523	GNANA SHREE K	B.Sc Computer Science	ONLINE JEWELLERY SHOP	Mrs.S.SHOBANA, Head & Assistant Professor in Computer Science
9	15BS4524	JAYASEELI J	B.Sc Computer Science	GRAPHICAL PASSWORD BY IMAGE SEGMENTATION	MRS.S.SARANYA, Assistant Professor in Computer Science
10	15BS4525	KARTHIKA C	B.Sc Computer Science	PET SHOP MANAGEMENT SYSTEM	Mrs.S.SHOBANA, Head & Assistant Professor in Computer Science
11	15BS4526	KAVIYA C	B.Sc Computer Science	BUS TICKET RESERVATION SYSTEM	Ms.S.PONMALAR, Assistant Professor in Computer Science
12	15BS4527	KIRUTHIKA E	B.Sc Computer Science	FOOD DONATING SYSTEM	Mrs.S.SARANYA, Assistant Professor in Computer Science
13	15BS4528	KOUSALYA M	B.Sc Computer Science	MEMORABLE GALLERY COLLECTION	Mrs.B.SASIKALA, Assistant Professor in Computer Science
14	15BS4529	LAVANYA M	B.Sc Computer Science	ONLINE HEALTH CHECKUP REGISTRATION SYSTEM	Mrs.S.SHOBANA, Head & Assistant Professor in Computer Science
15	15BS4531	MALARVIZHI V	B.Sc Computer Science	ONLINE COMPETATIVE EXAMINATION TESTING	Ms.E.KOKILAMANI, Assistant Professor in Computer Science
16	15BS4532	MANJULA G	B.Sc Computer Science	ONLINE LAUNDARY SYSTEM	Mrs.B.SASIKALA, Assistant Professor in Computer Science
17	15BS4533	MASILAMANI M	B.Sc Computer Science	ONLINE FURNITURE SHOP	Ms.S.PONMALAR, Assistant Professor in Computer Science
18	15BS4534	NANDHINI M	B.Sc Computer Science	RESUME BUILDER APPLICATION	Mrs.D.PAVITHRA, Assistant Professor in Computer Science
19	15BS4535	NANDHINI N	B.Sc Computer Science	ONLINE CONSTRUCTION SYSTEM	Ms.E.KOKILAMANI, Assistant Professor in Computer Science
20	15BS4536	NANDHINI V	B.Sc Computer Science	ONLINE HEALTH CARE SYSTEM	Mrs.S.SARANYA, Assistant Professor in Computer Science
21	15BS4537	NANTHINI R	B.Sc Computer Science	MARRIAGE MATRIMONAL PORTAL	Ms.S.PONMALAR, Assistant Professor in Computer Science

22	15BS4538	NARMADHA P	B.Sc Computer Science	E-GIFT SHOP/E-HOME DECORATIONS	Mrs.S.SARANYA, Assistant Professor in Computer Science
23	15BS4539	NATHIYA N	B.Sc Computer Science	CAMPUS REQUIREMENT SYSTEM	Mrs.B.SASIKALA, Assistant Professor in Computer Science
24	15BS4540	NATHIYA P	B.Sc Computer Science	UNIVERSITY BULLETIN BOARD	Mrs.D.PAVITHRA, Assistant Professor in Computer Science
25	15BS4541	NEYA VARSHINI S	B.Sc Computer Science	ONLINE JOB PORTAL-PLACEMENT FOR COMPANIES	Ms.E.KOKILMANI, Assistant Professor in Computer Science
26	15BS4542	NITHYA V	B.Sc Computer Science	ONLINE VOTING SYSTEM	Mrs.S.SHOBANA, Assistant Professor in Computer Science
27	15BS4543	NIVETHINI M	B.Sc Computer Science	PASSENGER MESSAGING SYSTEM	Mrs.N.SATHYA PRIYA, Assistant Professor in Computer Science
28	15BS4544	PAVITHRA M	B.Sc Computer Science	HYPERACTIVITY SYSTEM	Mrs.B.SASIKALA, Assistant Professor in Computer Science
29	15BS4545	PONMOZHI U N	B.Sc Computer Science	CHILD CARE INFORMATION SYSTEM	Mrs.B.SASIKALA, Assistant Professor in Computer Science
30	15BS4546	PREETHI K	B.Sc Computer Science	FINGER PRINT ATM	Mrs.B.SASIKALA, Assistant Professor in Computer Science
31	15BS4548	ROHINI M	B.Sc Computer Science	BEAUTY CARE AND TREATMENT SYSTEM	Ms.S.PONMALAR, Assistant Professor in Computer Science
32	15BS4549	SANGAVI B	B.Sc Computer Science	INTERCOLLEGE EVENT	Mrs.S.SARANYA, Assistant Professor in Computer Science
33	15BS4550	SANGEETHA P	B.Sc Computer Science	MUSICAL INSTUMENT SYSTEM	Ms.S.PONMALAR, Assistant Professor in Computer Science
34	15BS4551	SANTHIYA E	B.Sc Computer Science	BLOOD BANK APPLICATION	Mrs.D.PAVITHRA, Assistant Professor in Computer Science
35	15BS4552	SARANYA M	B.Sc Computer Science	ONLINE MARRIAGE PLANNING	Ms.E.KOKILAMANI, Assistant Professor in Computer Science
36	15BS4553	SARANYA R	B.Sc Computer Science	ONLINE BLOOD BANK SYSTEM	Mrs.S.SHOBANA,Head & Assistant Professor in Computer Science
37	15BS4554	SELVI S	B.Sc Computer Science	PHOTO COMPETITION WEBSITE WITH VOTING	Ms.P.YASODHA, Assistant Professor in Computer Science
38	15BS4555	SHENBAGA PRIYA M	B.Sc Computer Science	INTERIOR DESIGN	Mrs.S.MAHALAKSHMI, Assistant Professor in Computer Science
39	15BS4556	SHOUNDARIA B	B.Sc Computer Science	ONLINE LAPTOP SALES AND SERVICE SYSTEM	MS.E.KOKILAMANI, Assistant Professor in Computer Science
40	15BS4557	SOPHIYA SRI A	B.Sc Computer Science	DEFECT FREE ONLINE SHOPPING	Mrs.D.PAVITHRA, Assistant Professor in Computer Science
41	15BS4558	SUMATHI V	B.Sc Computer Science	ONLINE GAS BOOKING SYSTEM	Mrs.S.SHOBANA, Head &Assistant Professor in Computer Science
42	15BS4559	SWARNA LAKSHMI B	B.Sc Computer Science	MEDIFACTS AGENCY	Mrs.B.SASIKALA, Assistant Professor in Computer Science
43	15BS4560	VENNILA A	B.Sc Computer Science	ONLINE HOUSE RENTING SYSTEM	MS.E.KOKILMANI, Assistant Professor in Computer Science
44	15BS4561	YASINA PARVEEN M	B.Sc Computer Science	QUIZ APPLICATION	MS.S.PONMALAR, Assistant Professor in Computer Science

TUITION CENTRE MANAGEMENT SYSTEM

A PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the Degree of

Bachelor of Science in Computer Science

Submitted By

Nivethini M

(Reg. No: 15BS4543)

Under the Guidance of

Mrs. N. Sathyapriya M.Sc(CS).,MBA(ISM).,M.Phil.,

Assistant Professor

Department of Computer Science

DEPARTMENT OF COMPUTER SCIENCE

SRI G.V.G VISALAKSHI COLLEGE FOR WOMEN

(Autonomous)

Accredited at 'A' Grade by NAAC

An ISO Certified Institution

Udumalpet – 642 128

(2017 -2018)

CERTIFICATE

This is to certify that the project entitled **"Tuition Centre Management System"** is the record work done by **Nivethini M (15BS4543)** in partial fulfillment of the requirements for the award of the Degree of Bachelor of Science in Computer Science in **Sri G.V.G Visalakshi College for Women (Autonomous), Udumalpet** during the academic year 2017 – 2018.

Submitted for the viva- voce held on _____

Signature of the Guide **Signature of the HOD**

Internal Examiner **External Examiner**

DECLARATION

I here by declare that the project entitled **"Tuition Centre Management System"** submitted to the Department of Computer Science, Sri G.V.G Visalakshi College for Women (Autonomous), Udumelpet in partial fulfillment of the requirements for the award of the Degree of Bachelor of Science in Computer Science, is a record of original project work done by me under the supervision and guidance of **Mrs. N. Sathyapriya M.Sc(CS).,MBA(ISM).,M.Phil.,** Assistance Professor, Department of Computer Science, Sri G.V.G Visalakshi College for Women (Autonomous), Udumalpet.

Place : Udumalpet **Signature of the Candidate**

Date : **(NIVETHINI M)**

ACKNOWLEDGEMENT

First of all I record my sincere gratitude to the Almighty for His blessings for the successful completion of this project.

I take immense pleasure in expressing my thanks to the management for their continuous support throughout the project.

At the outset, I express my sincere thanks to our principal **Dr.(Mrs.) K. Punithavalli, Sri G.V.G Visalakshi College for Women (Autonomous)** for motivating us to take project activities and granting us permission to access materials inside the college.

I am extremely thankful to Director **Dr.K.Kannan, Sri G.V.G Visalakshi College for Women (Autonomous)** for providing all the facilities to do this project.

I am greatly indebted to **Mrs. S. Shobana,** Head, Department of Computer Science for her continuous support throughout the Project.

I wish to express my sincere thanks to my guide, **Mrs. N. Sathyapriya,** Assistant Professor, Department of Computer Science for her continuous encouragement, involvement and valuable suggestions to improve and finish the project successfully.

I take this opportunity to thank my Parents and Friends for their support, contribution and motivation which helped me a lot to complete this project successfully.

SYNOPSIS

The project entitled **"Tuition Centre Management System"** is developed with an aim of managing the tuition centre details through a mobile application. The project is developed using **"Android Studio"** as the Front End and **"SQLite"** as the Back End. The system helps effectively in managing the data of students. The students and their parents are the users of this application. The mobile application can be accessed from anywhere and at anytime.

This project contains the following modules and pages,

Admin module

- Student details
- Announcements
- Test and class time table
- Marks and attendance

Student Module

- View test and class time table
- View new announcements
- Download notes
- View attendance and marks

Parent Module

- View attendance and marks

The tuition Centre management system allows the administrator to view and edit data as required. It maintains the marks, attendance and the student details. Students can view their test time table and class time table through this application. They can view the announcements made by the tuition center. This system gives details about attendance and marks of the children to their parents. All this functionality ensures no errors.

CONTENT

1 . INTRODUCTION

The project entitled as **"Tuition Centre Management System"** is developed using **"Android Studio"** as Front End and **"SQLite"** as Back End. Tuition Centre Management System is an mobile applicaiton that can be used in tuition centre. This system is basically specializing on managing one tuition centre. This system is handled by administrator to enter the details of the students. The parents of the students can know the marks and attendance.This system can keep huge data organised and secured. It offers faster process and reduces the cost. The modules in this project are admin module, parent module and student module. Separate portals for admin, parent and student are provided. The authentication requires ID and password.

1.1 Problem Definition

The main problem in the existing system is that there is no mobile application for the management of the tuition centre activities. All the details are maintained manually in note and paper and it is the difficult to maintain. In this system, finding and retrieving of the data is difficult. There is no provision to notify the students and parents about the student activities. To overcome all these difficulties the proposed system has been developed.

2 . SYSTEM ANALYSIS

System analysis is a problem-solving technique that breaks down a system into its component pieces for the purpose of the studying how well those component parts work and interact to accomplish their purpose. The field of system analysis relates closely to requirements analysis or to operations research.

2.1 EXISTING SYSTEM

The main problem in the existing system is that there is no mobile application for the management of the tuition centre activities. All the details are maintained manually in note and paper which makes maintainance more difficult. The use of manual system requires more space of working area and use of additional equipments and facilities such as files, cabinets, racks and others. By using the manual process, people have to deal more with time and effort.

Disadvantages of Existing System:

- Lot of paper works.
- Takes more time to enter data.
- Needs more effort.
- Duplication of data may occur.
- Tracing the data is difficult.
- No provision for the parents to know the attendance and mark of their children on time.

2.2 PROPOSED SYSTEM

The proposed system is used to replace the manual system. This system has the ability to add record, save, delete and edit record. The task is centralised which means the process of entering, updating and maintaining data can be done by the admin. The informaton in the system is kept secured. The authentication requires ID and password for each user.The use of database will avoid redundant of data.

Advantages of Proposed System:

- Less paper work
- On time notification
- Duplication of data is avoided
- Tracing the data is easy
- Takes less time and effort to enter data
- Data maintainance is easy
- Data is kept secured
- Separate portals for admin, parent and student is provided.
- Separate authentication is provided
- Announcement page to pass information to the students
- Parents can view attendance and marks of the students

2.3 PROJECT DESCRIPTION

The project entitled "**Tuition Centre Management System**" is developed with an aim of managing the tuition centre details through a mobile application. The project is developed using "**Android Studio**" as the Front End and "**SQLite**" as the Back End. The system helps effectively in managing the data of students. The students and their parents are the users of this application. The mobile application can be accessed from anywhere and at anytime.

This project contains the following modules and pages,

Admin module

- Student details
- Announcements
- Test and class time table
- Marks and attendance

Student Module

- View test and class time table
- View new announcements
- Download notes
- View attendance and marks

Parent Module

- View attendance and marks

Student Details:

The page contains the details of the students like Student Id, Name, Student and Parent phone number, Parent Name and Address.

Announcements:

Students can view new announcements posted by the tuition center. Through this students can know any information from their home.

Test Time Table:

The students can know information about test date and time and in what subjects.

Class Time Table:

Students can view in which time and in which day they have classes and they can also know what subject classes are there.

Attendance:

Both parents and students can view the attendance of the students. It contains the student attendance date wise.

Notes:

Student can download the notes uploaded by the admin.

Marks:

The marks page allow both the parents and students to view the mark of the students in a particular test.

3. SYSTEM REQUIREMENTS

3.1 Hardware Requirements:

For Desktop:

Processor	:	Intel(R) Core(TM) i5-2410M CPU
Speed	:	2.30 GHz
Hard Disk	:	400 GB
RAM	:	4 GB
Monitor	:	15.6” LED
Keyboard	:	Multimedia keyboard
Mouse	:	Optical Mouse

For Mobile:

Processor	:	Qualcomm MSM8916 Quad Core 3G and above
Speed	:	2.30 GHz
Main Memory	:	8 GB
RAM	:	1 GB
Operating System	:	Lollipop and above

3.2 Software Requirements:

Operating System	:	Windows 7 & above
Front End	:	Android Studio 3.0.1
Back End	:	SQLite

3.3 Software Specifications:

3.3.1.Android Studio:

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA . On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, such as:

- A flexible Gradle-based build system.
- A fast and feature-rich emulator.
- A unified environment where you can develop for all Android devices.
- Instant Run to push changes to your running app without building a new APK.
- Code templates and GitHub integration to help you build common app features and import sample code.
- Extensive testing tools and frameworks.
- Lint tools to catch performance, usability, version compatibility, and other problems
- C++ and NDK support.
- Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine.

Project Structure

Each project in Android Studio contains one or more modules with source code files and resource files. Types of modules include:

- Android app modules
- Library modules
- Google App Engine modules

All the build files are visible at the top level under **Gradle Scripts** and each app module contains the following folders:

- **manifests**: Contains the AndroidManifest.xml file.
- **java**: Contains the Java source code files, including JUnit test code.
- **res**: Contains all non-code resources, such as XML layouts, UI strings, and bitmap images.

The Android project structure on disk differs from this flattened representation. To see the actual file structure of the project, select **Project** from the **Project** dropdown.

You can also customize the view of the project files to focus on specific aspects of your app development. For example, selecting the **Problems** view of your project displays links to the source files containing any recognized coding and syntax errors, such as a missing XML element closing tag in a layout file.

The User Interface

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

Gradle Build System

Android Studio uses Gradle as the foundation of the build system, with more Android-specific capabilities provided by the Android plugin for Gradle. This build system runs as an integrated tool from the Android Studio menu, and independently from the command line. You can use the features of the build system to do the following:

- Customize, configure, and extend the build process.
- Create multiple APKs for your app, with different features using the same project and modules.
- Reuse code and resources across sourcesets.

3.3.2 SQLite:

SQLite is a relational database management system contained in a C programming library. In contrast to many other database management systems, SQLite is not a client–server database engine. Rather, it is embedded into the end program.
SQLite is ACID-compliant and implements most of the SQL standard, using a dynamically and weakly typed SQL syntax that does not guarantee the domain integrity.
SQLite is a popular choice as embedded database software for local/client storage in application software such as web browsers. It is arguably the most widely deployed database engine, as it is used today by several widespread browsers, operating systems, and embedded systems (such as mobile phones), among others. SQLite has bindings to many programming languages.

Design:

Unlike client–server database management systems, the SQLite engine has no standalone processes with which the application program communicates. Instead, the SQLite libraryis linked in and thus becomes an integral part of the application program. The library can also be called dynamically. The application program uses SQLite's functionality through simple function calls, which reduce latency in database access: function calls within a single process are more efficient than inter-process communication. SQLite stores the entire database (definitions, tables, indices, and the data itself) as a single cross-platform file on a host machine. It implements this simple design by locking the entire database file during writing. SQLite read operations can be multitasked, though writes can only be performed sequentially.

Due to the server-less design, SQLite applications require less configuration than client-server databases. SQLite is called zero-confbecause it does not require service management startup or access control based on GRANT and passwords. Access control is handled by means of file system permissions given to the database file itself. Databases in client-server systems use file system permissions which give access to the database files only to the daemon process.

SQLite uses PostgreSQL as a reference platform. “What would PostgreSQL do” is used to make sense of the SQL standard. One major deviation is that, with the exception of primary keys, SQLite does not enforce type checking; the type of a value is dynamic and not strictly constrained by the schema (although the schema will trigger a conversion when storing, if such a conversion is potentially reversible). SQLite strives to follow Postel's Rule.

History:

D. Richard Hipp designed SQLite in the spring of 2000 while working for General Dynamics on contract with the United States Navy. Hipp was designing software used aboard guided missile destroyers, which originally used HP-UX with an IBM Informix database back-end. SQLite began as a Tcl extension.

The design goals of SQLite were to allow the program to be operated without installing a database management system or requiring a database administrator. Hipp based the syntax and semantics on those of PostgreSQL 6.5. In August 2000, version 1.0 of SQLite was released, with storage based on gdbm (GNU Database Manager). SQLite 2.0 replaced gdbm with a custom B-tree implementation, adding transaction capability. SQLite 3.0, partially funded by America Online, added internationalization, manifest typing, and other major improvements.

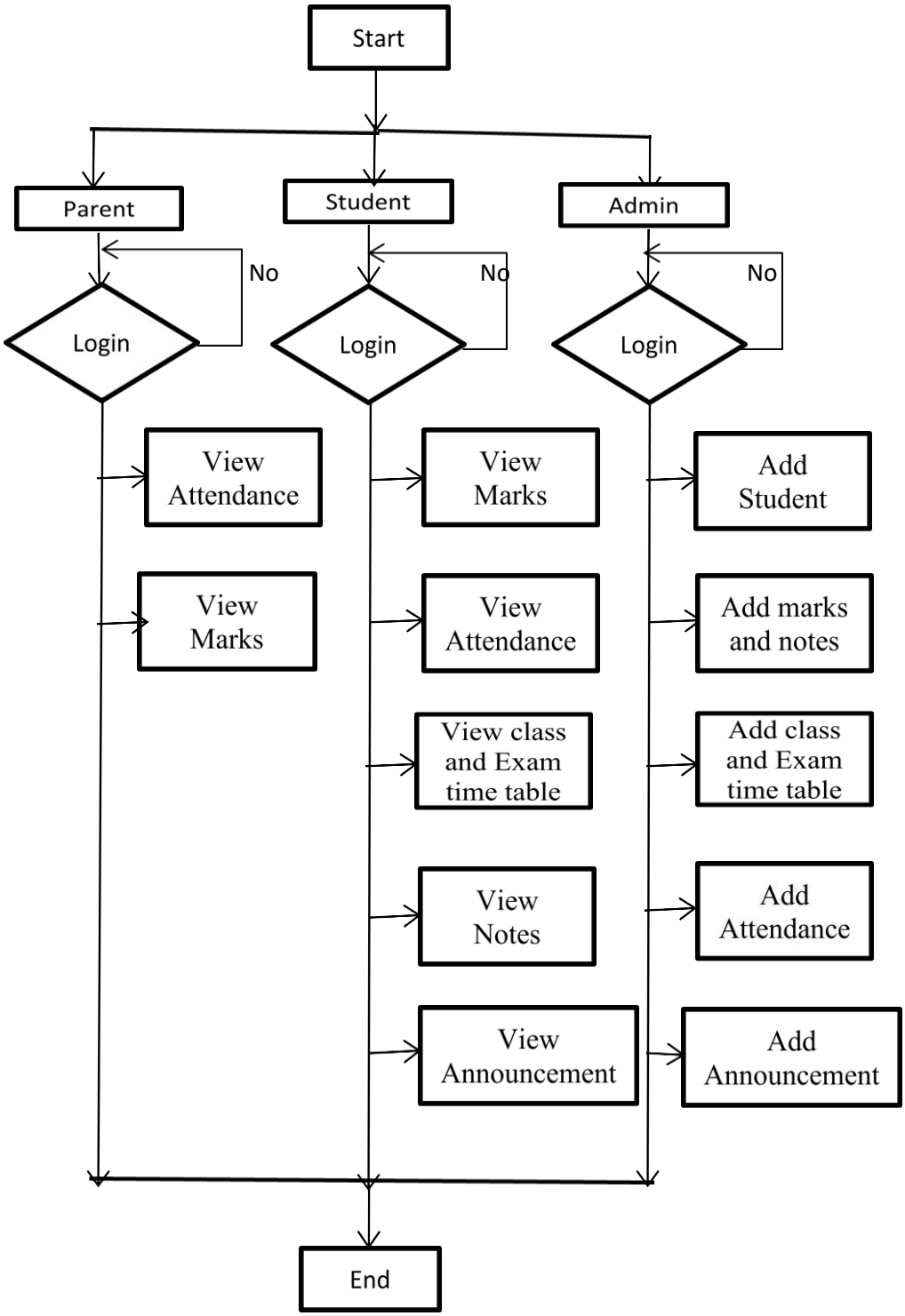
In 2011 Hipp announced his plans to add a NoSQL interface (managing documents expressed in JSON) to SQLite databases and to develop UnQLite, an embeddable document-oriented database. UnQLite was released as an independent database.

Features:

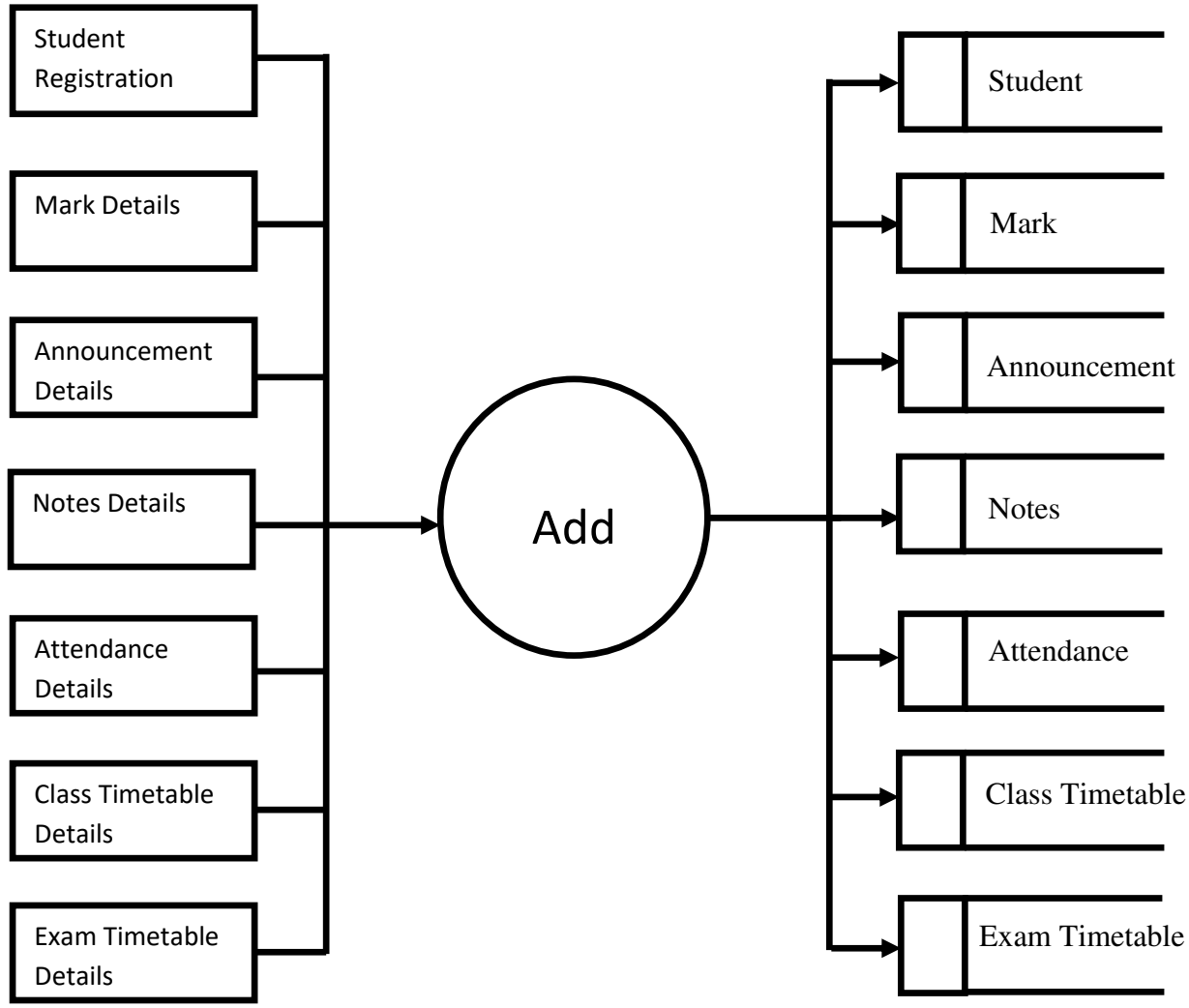
SQLite implements most of the SQL-92 standard for SQL but it lacks some features. For example, it partially provides triggers, and it cannot write to views (however it provides INSTEAD OF triggers that provide this functionality). While it provides complex queries, it still has limited ALTER TABLE function, as it cannot modify or delete columns.

4 . SYSTEM DESIGN

4.1 Sitemap:



4.2 Data Flow Diagram:



4.3 Table Design:

Table Name : Student_details

Primary Key: Id

Fields	Data type	Size	Description
Id	Integer	10	Student Id
Name	Text	30	Student Name
DOB	Text	15	Student Date of Birth
Phone	Text	10	Student PhoneNumber
Email	Text	30	Student Email
ParentName	Text	30	StudentParentName
ParentPhone	Text	10	StudentParentPhone
Regid	Text	10	Student RegistrationId
UserPassword	Text	25	UserPassword
ParentPassword	Text	25	ParentPassword

Table Name : Subjects

Primary Key: Id

Fields	Data type	Size	Description
Id	Integer	10	Subject Id
Subject_Name	Text	15	Subject Name

Table Name : Notes

Primary Key: Id

Fields	Data type	Size	Description
Id	Integer	10	Notes Id
Notes	Text	50	Notes Description
Subject_Name	Text	15	Subject
File_Path	Text	30	File path of Notes
Format	Text	20	Notes File Format
Date	Text	15	Date updated

Table Name : Marks

Primary Key: Id

Fields	Data type	Size	Description
Id	Integer	10	Mark Id
Subject_Name	Text	15	Subject Name
Reg_Id	Integer	15	Student RegistrationId
Grade	Text	10	Grade Obtained
Total_Marks	Integer	5	Total Marks
Marks	Integer	5	Marks Obtained

Table Name : Exam Time Table

Primary Key: Id

Fields	Data type	Size	Description
Id	Integer	10	Exam Id
StartDateTime	Text	15	Start Date Time
EndDateTime	Text	15	End Date Time
Subject	Text	15	Subject

Table Name : Class Time Table

Primary Key: Id

Fields	Data type	Size	Description
Id	Integer	10	Timetable Id
Session_name	Text	15	Session Name
StartTime	Text	15	Class Starting time
EndTime	Text	15	Class Ending time
Day_Type	Text	20	Weekend or Weekdays

Table Name : Announcements

Primary Key: Id

Fields	Data type	Size	Description
Id	Integer	10	AnnouncementId
Announcement_Title	Text	20	Announcement Title
Announcement_Date	Text	15	Announcement Date
Description	Text	50	Announcement Description

Table Name : Attendance

Primary Key: Id

Fields	Data type	Size	Description
Id	Integer	10	Attendance Id
Reg_Id	Integer	10	Student RegistrationId
Date	Text	15	Attendance Date
Status	Text	10	Present or Absent

5 . SYSTEM TESTING

Testing is a broad subject, with various methodologies combining various approaches, but it embodies attempts to ensure that the user's experience is satisfactory: intuitive, reliable, responsive, secure, etc. The testing process focuses on the logical intervals of the software ensuring that all statements have been tested and on functional interval is conducting tests to uncover errors and ensure that defined input will produce actual results that agree with the required results.

Unit Testing

Unit tests are tests that exercise a class in isolation from (most of) the rest of the application, and form a significant element of Test Driven Development where failing tests are written to allow the application to be debugged into existence. Unit tests focuses on the smallest unit of software design,the module. In this testing each module is found to be working satisfactory are regard to the expected output from the module.

In this project, each form is tested individually by giving necessary data and checked for its correctness. Error messages are displayed for the errors generated.

Integration Testing

Integration testing is the systematic testing for constructing the program structure, while the same time conducting the tests to uncover errors associated with the interfere. Whereas unit testing tests classes in isolation from each other, and system testing checks that the entire assembly works in the target platform when operated as a user, integration testing checks that selected groups of parts fit together properly in controlled conditions that are easier to set up and manipulate with direct access to the program code.

In this project, Admin module and student module are integrated together and the details updated in admin can be viewed by the students.

Validation Testing

Validation is done at the end of the development process and takes place after verifications are completed. At the culmination of the black box testing software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of test i.e., validation succeeds when the software functions in a manner that can be reasonably accepted by the customer.

In this project, the phone number field is validated to read only 10 characters. Each portal is authenticated by proper user name and password.

System Testing

System tests exercise the system end-to-end: the application will be on the device and will talk to your server if you use one. They are slower to run than unit tests and are based on driving the application through the UI in a 'black box' way. They form part of Behaviour Driven Development.

Because they run on a phone, the compile/execute cycle of the unit test becomes a compile/package/install/execute cycle, but they run against the real implementation and thus don't suffer from the danger of implementation inaccuracy unit test stubs and mocks present.

Usability Testing:

Usability testing is a way to see how easy to use something is by testing it with real users.Users are asked to complete tasks, typically while they are being observed by a researcher, to see where they encounter problems and experience confusion. If more people encounter similar problems, recommendations will be made to overcome these usability issues. This testing mainly focuses on the user's ease to use the application, flexibility in handling controls and ability of the system to meet its objectives. It is also called User Experience Testing.

This system can be used efficiently and it is easy to handle for all types of users.

Compatibility Testing:

Compatibility Testing is a type of Software testing to check whether your software is capable of running on different hardware, operating systems, applications , network environments or Mobile devices. Compatibility testing is a type of software testing used to ensure compatibility of the system/application/website built with various other objects such as other web browsers, hardware platforms, users (in case if it's very specific type of requirement, such as a user who speaks and can read only a particular language), operating systems etc. This type of testing helps to find out how well a system performs in a particular environment that includes hardware, network, operating system and other software etc.

This system is tested by installing it in different mobile operating systems and versions and it works efficiently.

6 . SYSTEM IMPLEMENTATION

Implementation is the final stage of the project. System Implementation is the major part of the project in which theoretical design is turned into a working system. System Implementation is the process of converting a new or a revised system design into an operational one. System implementation includes Planning, Investigation of scenario, Constraints and implementation of the project. Initially, Our task is to plan the concept which we yet to execute. Planning the concept includes execution methods, time taken to complete the task. Project Concept is completely based on Staff, Student and the Parents.

Once the planning is completed, ensure that all the functionalities are working when staff creates student profile, attendance and mark details. So that both the ends communicates effectively.

The implementation plan consists of the following sample steps

- Testing the developed system with sample data
- Detection and correction errors
- Making necessary changes in the system
- Checking the reports with that of the existing system
- Training and involvement of user personnel
- Installation of hardware and software utilities

7 . CONCLUSION

This project contributes in providing effective solutions to current **"Tuition Centre Management System"**. The process of Tuition management data are retrieved from database. Information updation becomes very easy and convenient with this application. By using the current mobile technology and network technology, this project has created a new era of Tuition management application.

The package was designed in such a way that future modifications can be done easily. The following conclusions can be deduced from the development of the project. Automation of the entire system improves the efficiency. It provides a friendly graphical user interface which proves to be better when compared to the existing system. It gives appropriate access to the authorized users depending on their permissions.

It effectively overcomes the delay in communications. Updating of information becomes so easier. System security, data security and reliability are the striking features. The System has adequate scope for modification in future if it is necessary.

8 . FUTURE ENHANCEMENT

The **"Tuition Centre Management System"** is developed in order to have better data management education center. It is more to management tools which help user to store data accordingly and maintaining it. Later some telecom concepts, can be implemented so that the user will get the information about the activities of the tuition management in a mobile device. As technology grows, new functionality and new requirements can be implemented. Notifications through Email and SMS facility can added. Online test feature can be added for students. Later Fees management system can be added. The system is designed in such a way that it is flexible to change any further requirements prescribed by the user.

9 . BIBLIOGRAPHY

- Zigurd Mennieks, **"Programming Android: Java Programming for the New Generation of Mobile Devices"** by O'Reilly 3rd Edition.
- Marko Gargenta, **"Learning Android Building Applications for the Android Market"** by O'Reilly 2nd Edition..
- Bill Phillips and Chris Stewart, **"Android Programming: The Big Nerd Ranch Guide"** by Big Nerd Ranch 2nd Edition.

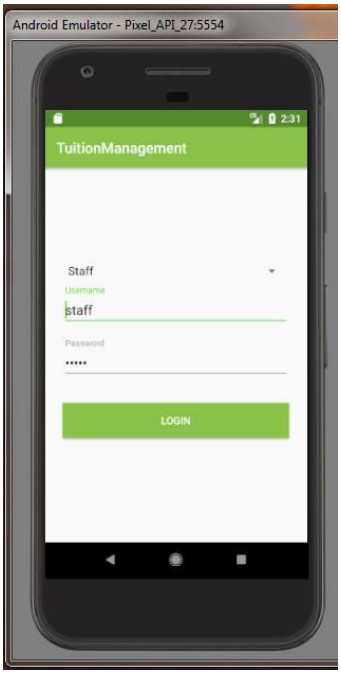
The following Websites were referred during the analysis and execution phase of the project

- Introduction to Android: http://developer.android.com/guide/index.html.
- Android API: http://developer.android.com/reference/packages.html
- Android User Interfaces: http://developer.android.com/guide/topics/ui/index.html
- Layout: http://developer.android.com/guide/topics/ui/declaring-layout.html

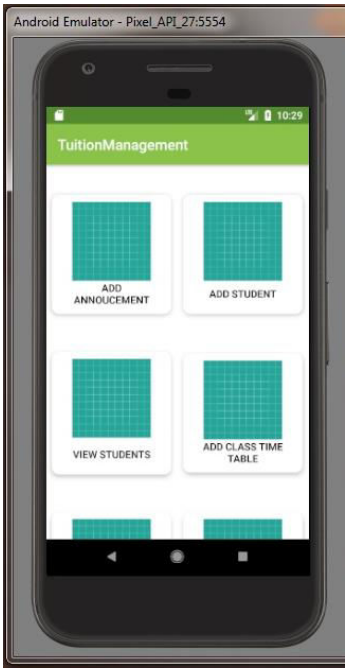
APPENDIX

Sample Form Design:

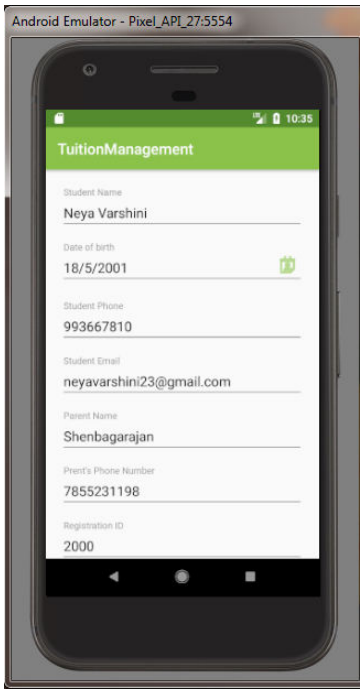
Admin LogIn Page:

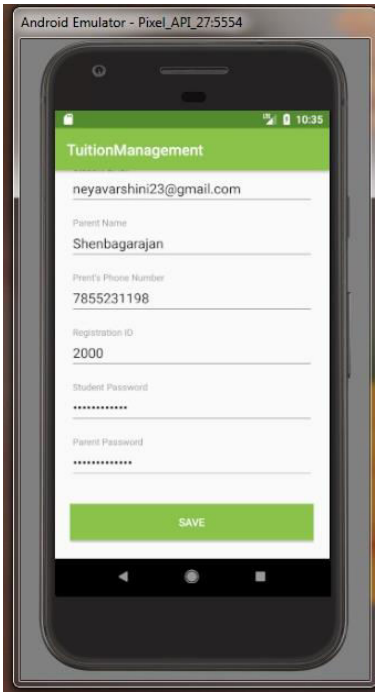


Admin Page:

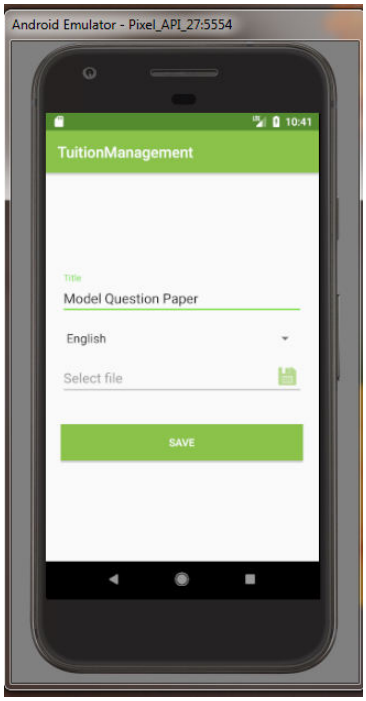


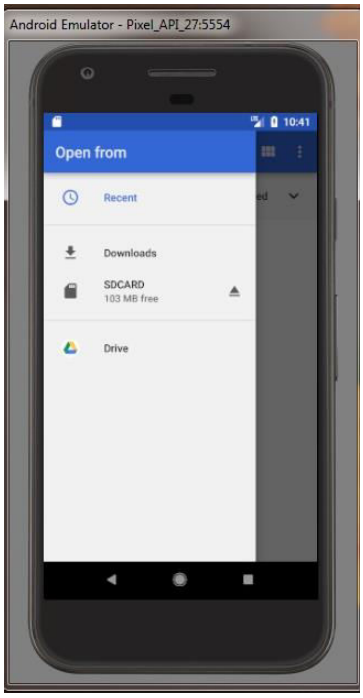
Add Student Page:



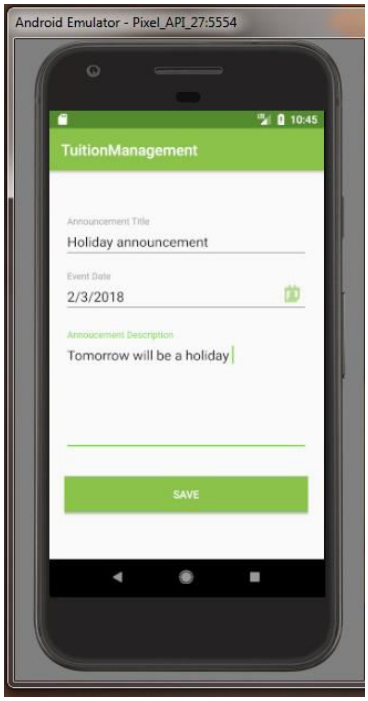


Add notes Page:

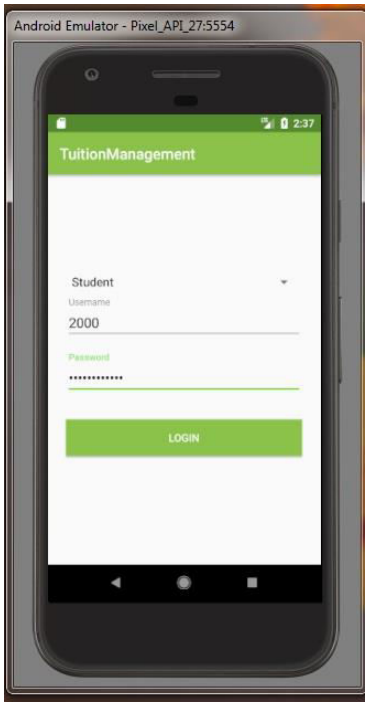




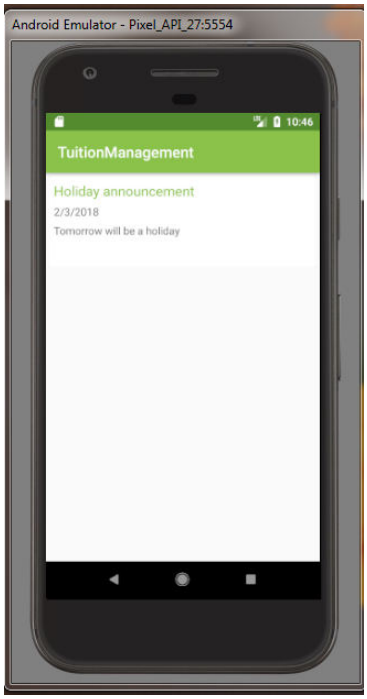
Add Announcement Page:



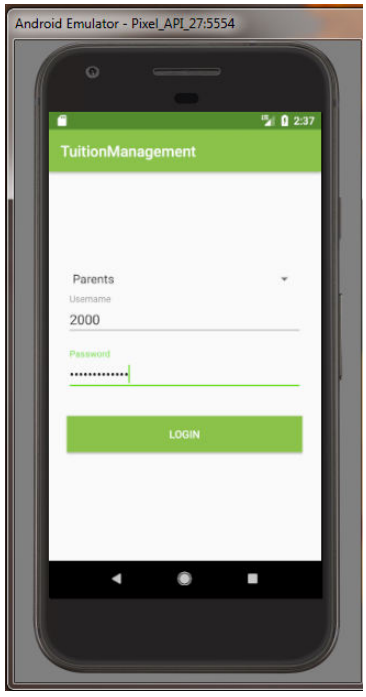
Student LogIn Page:



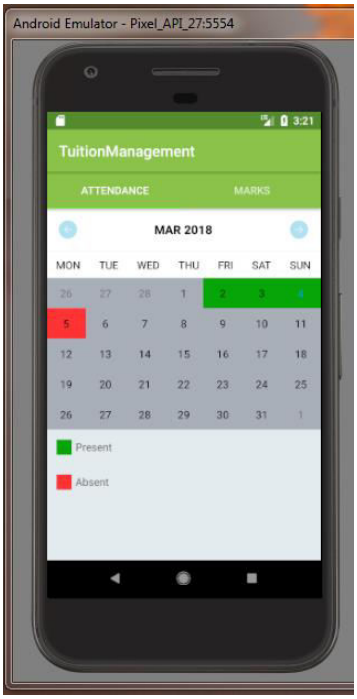
View Announcement Page:



Parent LogIn Page:



View Attendance Page:



Sample Coding:

Adding Marks:

```
public class MarksAddActivity extends AppCompatActivity implements
View.OnClickListener {
    private EditText inputRegId;
    private Spinner subjectSpinner;
    private EditText inputMarksObtained;
    private EditText inputTotalMarks;
    private EditText inputgrade;
    private AppCompatButton btnSave;
    private DatabaseHandler myDb;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_marks);
        myDb = new DatabaseHandler(this);
        findViews();
    }
    private void findViews() {
        inputRegId = (EditText) findViewById(R.id.input_reg_id);
        subjectSpinner = (Spinner) findViewById(R.id.subject_spinner);
        inputMarksObtained = (EditText) findViewById(R.id.input_marks_obtained);
        inputTotalMarks = (EditText) findViewById(R.id.input_total_marks);
        inputgrade = (EditText) findViewById(R.id.inputgrade);
        btnSave = (AppCompatButton) findViewById(R.id.btn_save);
        btnSave.setOnClickListener(this);
        String regId = getIntent().getStringExtra("studentid");
        inputRegId.setText(regId);
        List<String> subjects = new ArrayList<String>();
        subjects.add("COMP3301");
        subjects.add("INFS3200");
        subjects.add("COMP3702");
```

```
        subjects.add("DECO3500");
        subjects.add("Algorithms");
        subjects.add("Datastructures");
        subjects.add("M1");
        subjects.add("English");
        subjects.add("C Programming");
       ArrayAdapter<String> dataAdapter = new ArrayAdapter<String>(this,
android.R.layout.simple_spinner_item, subjects);
dataAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
        subjectSpinner.setAdapter(dataAdapter);
    }
    @Override
    public void onClick(View v) {
        if (v == btnSave) {
            insertMarksData();
        }
    }
    private void insertMarksData() {
        String regId = inputRegId.getText().toString();
        String subjectName = subjectSpinner.getSelectedItem().toString();
        String marksObtained = inputMarksObtained.getText().toString();
        String totalMarks = inputTotalMarks.getText().toString();
        String grade = inputgrade.getText().toString();
        if (!AppUtils.isValidText(regId)) {
            inputRegId.setError("Invalid");
        } else if (!AppUtils.isValidText(totalMarks)) {
            inputTotalMarks.setError("Invalid");
        } else if (!AppUtils.isValidText(marksObtained)) {
            inputMarksObtained.setError("Invalid");
        } else if (!AppUtils.isValidText(grade)) {
            inputgrade.setError("Invalid");
        } else {
            if (myDb.insertMarksData(regId, subjectName, marksObtained, totalMarks, grade)) {
                inputTotalMarks.setText("");
```

```
            inputMarksObtained.setText("");
            inputgrade.setText("");
            subjectSpinner.setSelection(0);
        }
    }
  }
}
```

Adding Student:

```
public class Student {
   int sutdentId;
   String studentName;
   String dob;
   String studentPhone;
   String studentEmail;
   String parentName;
   String parentPhoneno;
   String regId;
   String studentPassword;
   String parentPassword;
   public int getSutdentId() {
      return sutdentId;
   }
   public void setSutdentId(int sutdentId) {
      this.sutdentId = sutdentId;
   }
   public String getStudentName() {
      return studentName;
   }
   public void setStudentName(String studentName) {
      this.studentName = studentName;
   }
   public String getDob() {
```

```
    return dob;
}
public void setDob(String dob) {
    this.dob = dob;
}
public String getStudentPhone() {
    return studentPhone;
}
public void setStudentPhone(String studentPhone) {
    this.studentPhone = studentPhone;
}
public String getStudentEmail() {
    return studentEmail;
}
public void setStudentEmail(String studentEmail) {
    this.studentEmail = studentEmail;
}
public String getParentName() {
    return parentName;
}
public void setParentName(String parentName) {
    this.parentName = parentName;
}
public String getParentPhoneno() {
    return parentPhoneno;
}
public void setParentPhoneno(String parentPhoneno) {
    this.parentPhoneno = parentPhoneno;
}
public String getRegId() {
    return regId;
}
public void setRegId(String regId) {
    this.regId = regId;
```

```
    }
    public String getStudentPassword() {
        return studentPassword;
    }
    public void setStudentPassword(String studentPassword) {
        this.studentPassword = studentPassword;
    }
    public String getParentPassword() {
        return parentPassword;
    }
    public void setParentPassword(String parentPassword) {
        this.parentPassword = parentPassword;
    }
}
```