Bachelor of Science in Information Technology

Gyalpozhing College of Information Technology

Software Requirement Specification Document
for
Mini Project

GCIT Note

Submitted by:
Cheki Lhamo
12190043

Bachelor of Science in Information Technology

Introduction

Purpose

The purpose of this document is to present a detailed description of the GCIT Note application. It will explain the purpose and features of the system, the workflow of the system and it's system designs which are mandatory for the development of the system.

GCIT Note is an android based application which will allow all the tutors to add their modules and it's notes. The notes added by the tutors will be visible to students and can be accessed by the students. Students can access the notes of all the other modules added by the tutors.

Aim

The main aim of my project is to develop an android based application through which the tutors can add notes and students can access the notes.

Objectives

The main objectives of my application are:

- To provide a platform for the students to access the notes of all the modules (all semesters' module).
- To make it easier for the students and tutors to access the notes.
- These notes can be used as a reference for both the tutors and students.
- To transform mobile phones into an effective tool that can be used as your notebook.

Scope

User Scope

The scope(user) of my project is limited to the students and tutors(teaching staffs) of Gyalpozhing College Of Information Technology.

System Scope

1. Login
2. Add Modules
3. Add Notes
4. Delete Notes
5. Favourites
6. Rename
7. Share Notes
8. About
9. Logout

Requirements

Functional Requirements

Followings are the features of my application:

1. For Tutors
    - Login – The tutors can login to the system using the provided credentials.
    - Add modules – The tutors can add modules to upload the notes.
    - Add notes – The tutors can add notes to the modules that they added.
    - Delete Notes – The tutors can also delete their notes.
    - Logout – The tutors can logout from the system once they are done with their activities.

2. For Students
   - Login – The students have to login to the system using the provided credentials.
   - Add to favourites – The students can also view the notes added by all the tutors and the can add the notes to favourites section.
   - Rename – Students can rename the notes if they encounter notes with same file name.
   - Logout – The students can logout from the system once they are done with their task.

Non-functional Requirements

1. Portability
   The app will be easily portable to different versions of android and it is independent of the size of any android phone and tablets.

2. Usability
   The application will be simple and easy for the users to use. The users will be provided with icons and descriptions so that they can access the features easily. It will be user friendly.

3. Reliability
   The application is offline based and it can work without internet connection. The application will be reliable.

4. Scalability
   The application will be able to change in size depending on the number of notes that the tutors provides.

5. Accessibility
   Any user can access the notes without having to register/login to my system.

6. The orientation of the app will be in both portrait and landscape.

7. The application will include developers information and it can also be shared. It will also have exit option for the users to exit from the application once they are done with their task.

Software Requirements

Software that will be used for the development of this application are:
1. Android Studio version 4.1.2 or above.
Android Studio is the official Integrated Development Environment(IDE) for Android App Development.
It provides features such as:
- A flexible Gradle-based build system.
- An Emulator.
- Extensive testing tools and frameworks.
- Build in support for google cloud platform.

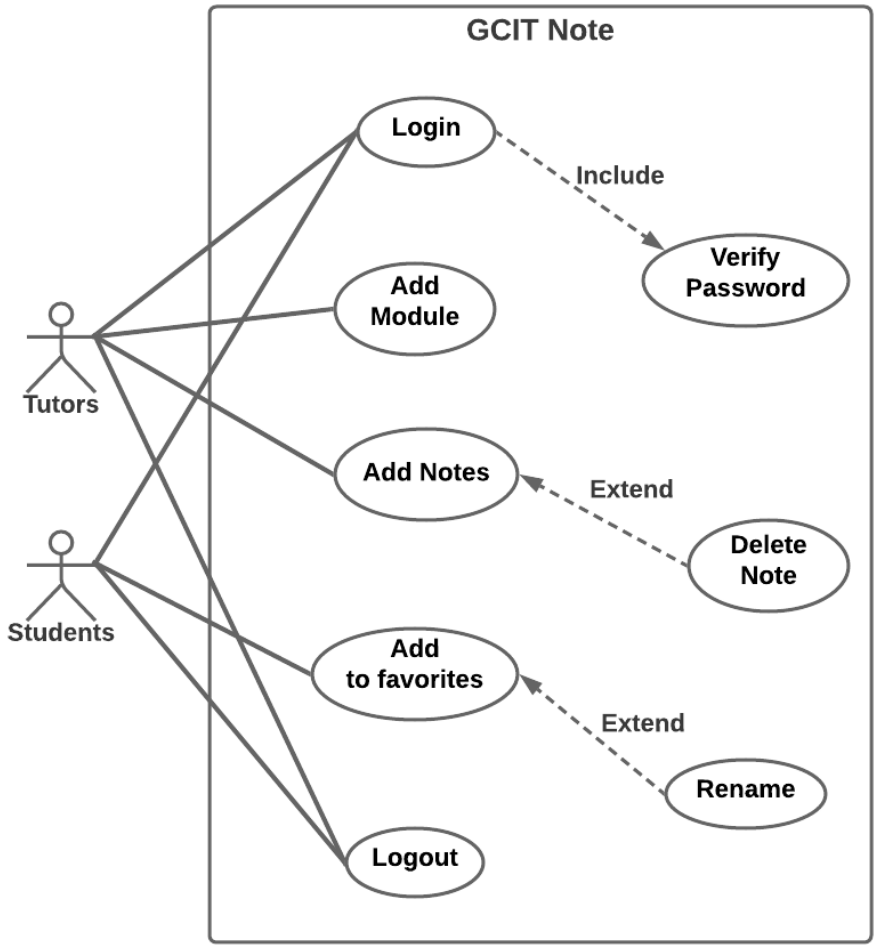
Java SE jdk, Android SDK-25 and code editor will be also be used and these software are embedded in with Android Studio.

2. Firebase database 19.1.2
The Firebase realtime database is a cloud-hosted No-SQL database that allows to store and synchronize the data between the users in realtime. The data is stores as JSON and synchronized in realtime to every connected client.

Hardware Requirements
Hardware requirements for developers:
1. Laptop with 4 - 8GB RAM, 2.00GHz*4 Processors and with 1.0 TB/above disk capacity.
2. An Android phone to be used as Emulator.

For users:
Android Phone

System Designs
1. Usecase

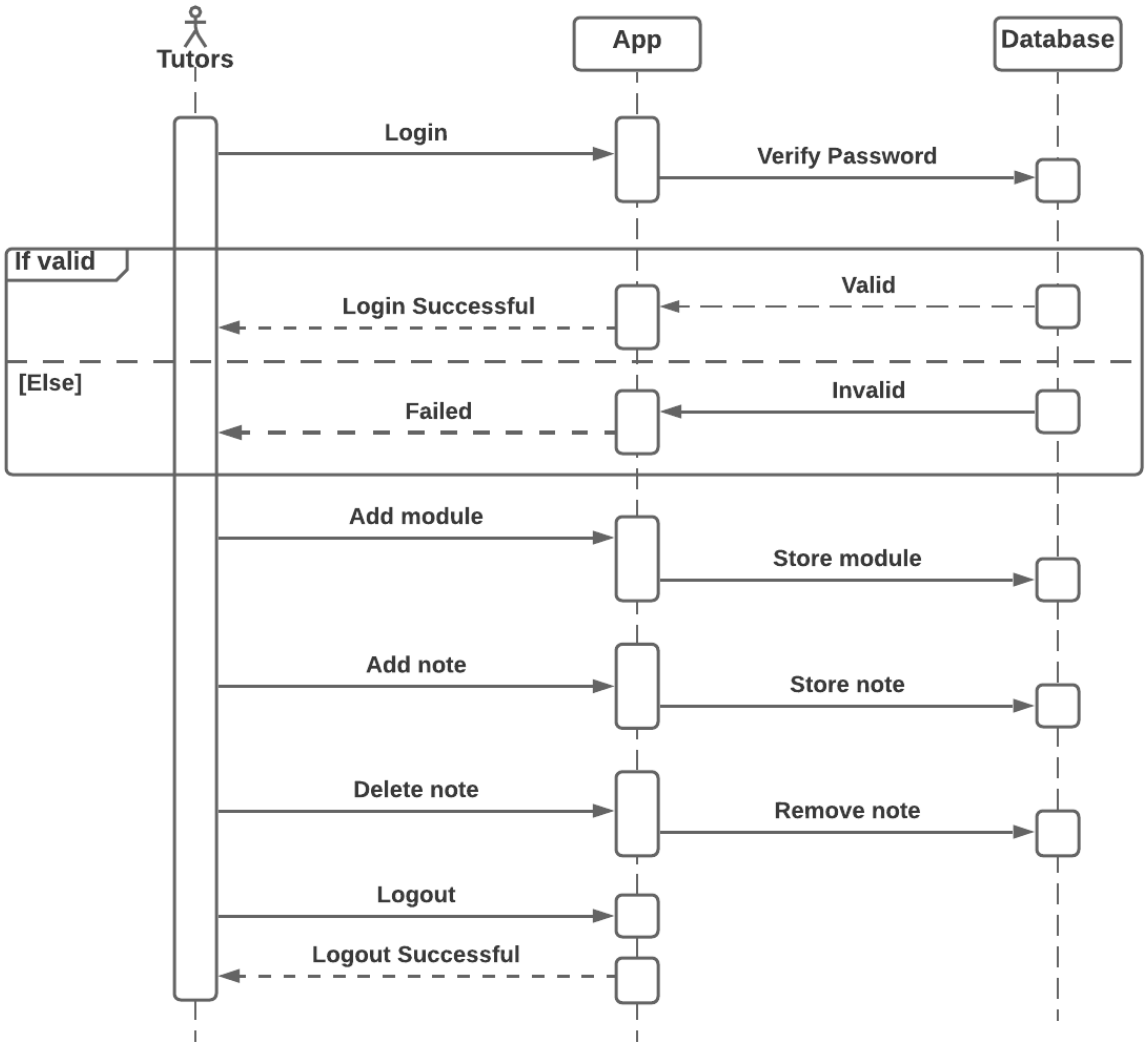


Usecase is a representation of user's interaction with the system that shows the relationship between the user and the different usecases in which the user is involved.
The two actors of the system:
1. Tutors
2. Students

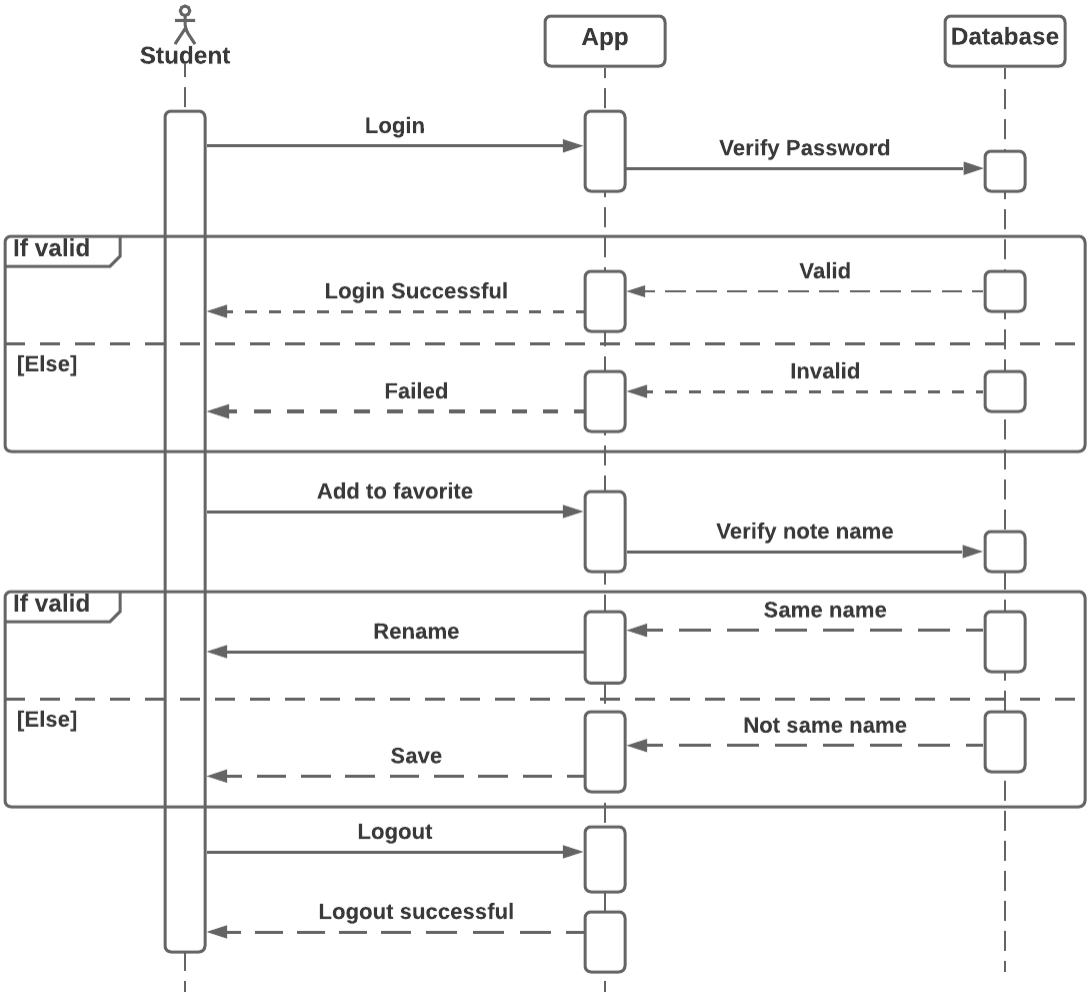
The functionalities of the actors are:
For Tutors:
Login – has verify password with included relationship.
Add Modules
Add Notes – has delete notes with excluded relationship.
Logout

For Students:
Login
Add to Favourites – has rename with excluded relationship.
Rename
Logout

2. Sequence diagram
   I. For Tutors

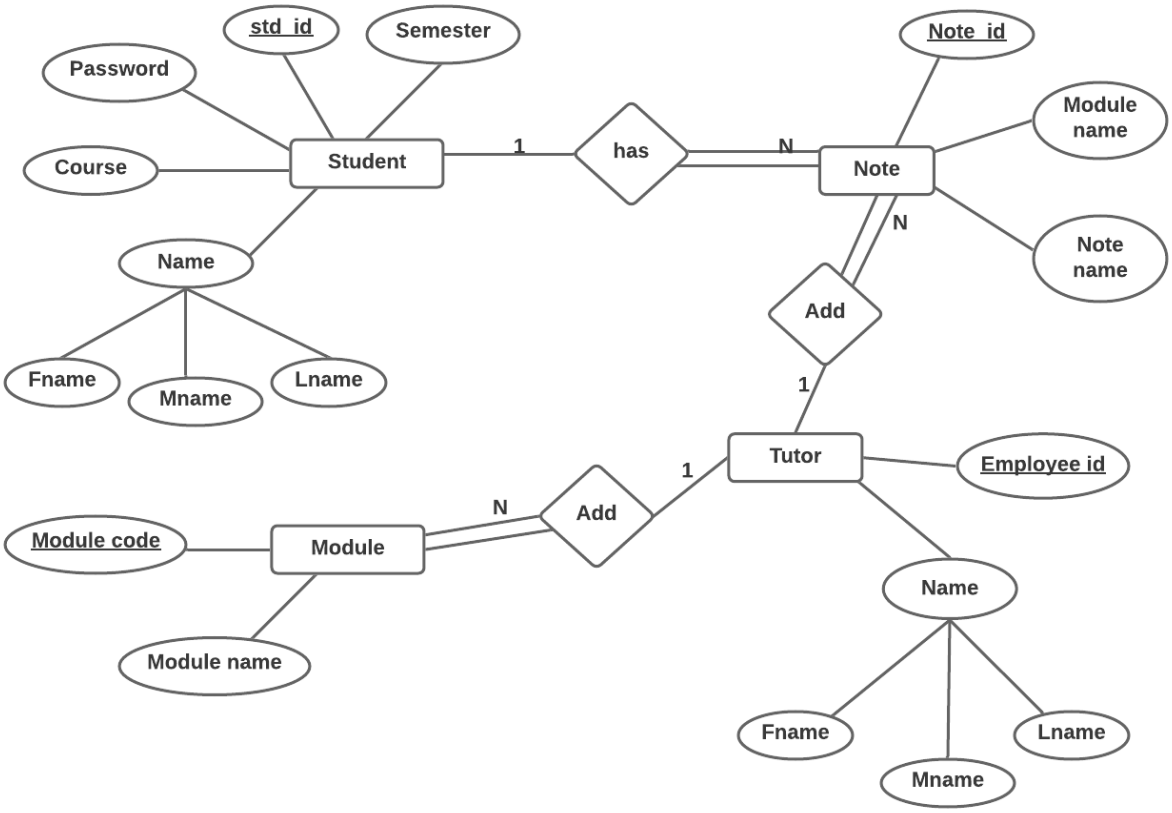


II. For Student



Sequence Diagrams are interaction diagrams that describes the details about how the operations are carried out in the system. The sequence diagram captures the interaction between the objects in context of a collaboration. The above sequence diagrams shows the system workflow and explains the detail logic behind GCIT Note application.

3. Entity Relationship diagram



The ERD is a structural diagram for the use in database design which contains different symbols and connectors. ERD is used for visualizing two important information that is te major entities and relationships among these entities.

There are four major entities in the above ERD with various attributes:

- Student – It has students _id as it's primary key and other attributes like name, course, password and semester.
- Tutor – Employee_id id the primary key for tutor entity with another attribute name.
- Module – It has module_code as it's primary key and it has another attribute called module_name.
- Note – Note_id is the primary key for note entity with other attributes module_name and note_name.

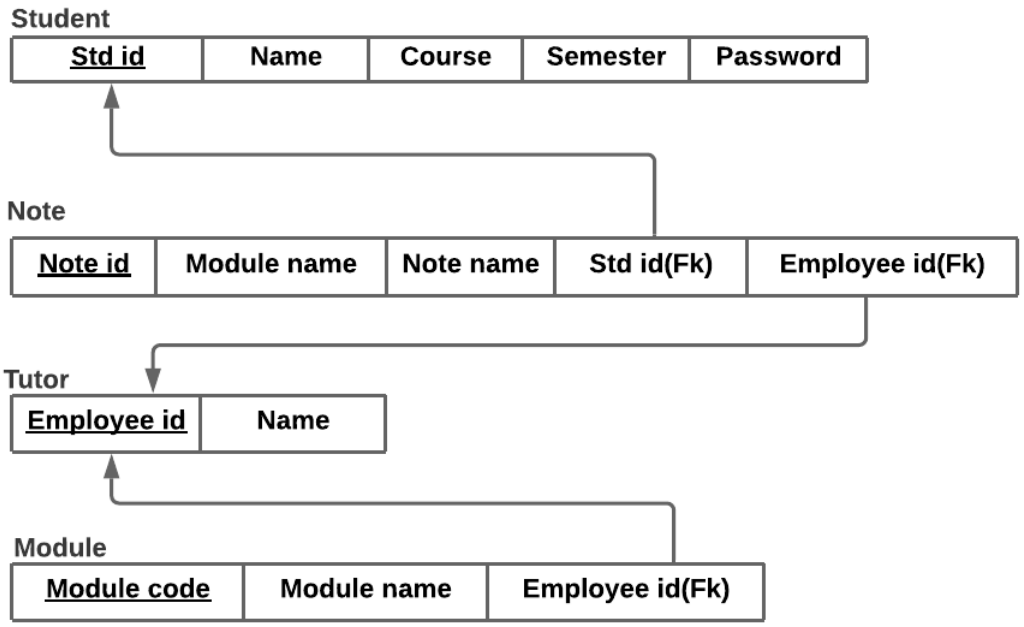
There are three relationships between the entities:

- Add – relationship between tutor and note with carditionality ratio 1:N. Every tutor can add many notes. Tutor can either add note or not(partial participation). Notes have to be added by the tutors(total participation).

- Has – relationship between student and note with carditionality ratio 1:N. Every student can have many notes. Student can either have note or not(partial participation). The notes should belong to a student(total participation).

- Add - relationship between tutor and module with carditionality ratio 1:N. Every tutor can add many modules. Tutor can either add module or not(partial participation). Module has to be added by the tutors(total participation).

4. Relational Schema Diagram



There are four tables:

- Student – student_id as it's primary key.
- Tutor – Employee_id as it's primary key.
- Note – Note_id as it's primary key. Student_id and employee_id as it's foreign key.
- Module – module_code as it's primary key and employee_id as it's foreign key.

Conclusion

A system requirements specification is a must when it comes to developing software. SRS is useful for both customers and software development team, it is important for developing a complete and clear system requirements document. SRS helps the customers to define their needs with accuracy, and it helps the development team understand what the customers need in terms of development. SRS should be well documented as it will lead to successful development of the software.