School of Electrical and Information Engineering, University
of the Witwatersrand, Private Bag 3, 2050, Johannesburg, South Africa

ELEN 4009 - Software Engineering

# Student Marks/Records Management Software - Requirements Gathering

**Project Team :** Khosa Masana (559990), Sanele Gcaba (459380),
Tshegofatso Misapitso (600313) and Londiwe Ngema (448871)

April 8, 2016

.

**Contents**

**Document Status Sheet**

| | |
|---|---|
| **Document Title** | Software Requirements |
| **Authors** | M. Khosa, S. Gcaba, T. Misapitso, L. Ngema |
| **Version** | 0.01 |
| **Document Status** | Draft |

| **Version** | **Date** | **Author(s)** | **Summary** |
|---|---|---|---|
| 0.0.1 | 25-02-2016 | M. Khosa, S. Gcaba, T. Misapitso, L. Ngema | Document Creation. |

.

## 1. Introduction (Sanele Gcaba)

### *1.1 Methodology*

The System Development Life Cycle (SDLC) methodology to be used for the project follows the Agile Model, this models breaks down the product into a cycle, it quickly delivers a working product and is a more realistic methodology. Ongoing releases are produced with small incremental changes and it depends heavily on customer interaction [1]. Specifically the Scrum Agile Model will be adopted for the project, Scrum comprises of short sprints and it enables the software team to be able to prioritize on what matters most. It is basically about delivering more often and getting feedback that is regularly responded to [2].

### *1.2 Purpose*

The purpose of this document is to present a detailed description of the requirements for the student marks/record system (SMMS). It will state the purpose and features of the system, the interfaces of the system, what the system will do, the limitations under which it must operate, define inputs and the expected reaction of the system (that is the outputs of the system).

Student Marks/Records Management Software provides online services that enable students to view their marks as they complete assignments, tests and laboratory work, the software system allows staff such as course coordinators the rights of to adding/editing marks obtained by the student on any of the mentioned forms of assessment. It is a convenient way for students to have access to their marks in a safe and confidential way as opposed to accessing them on notice boards where everyone else can publicly see them. The system ensures that each student can track his/her tentative progress throughout the year,it also helps in establishing errors errors in the record as early as possibly.

### *1.3 Project Scope*

- There are five basic users of the system - Students, Course coordinators, School administrator, Database Administrator, System Administrator.
- The primary function of the application is to allow students and staff to log-in using their details (Student/Staff number and password) and be able to access and view student records, the records include student marks for all forms of assessments for all courses registered for.
- The system database stores user profiles and student marks records. Marks records are retained for a period of 10 yrs.
- The software program should have domains assigned, i.e. each user can be able to access relevant and information they are authorised to view/edit/add based on what they are authorisation as per recognised domain. The system would be accessed online via a Browser and/or a Smart-phone App.

Below is a list of privileges per user as specified on the project brief [3].

**The Course Coordinator should be able to:-**

- Register himself/herself.
- Add various assessment method for the course and weighting for each assessment.
- Enter student marks on a user-friendly interface.
- Display or print out the table of students and their marks.
- Generate a summary statistics of the performance of the students - maximum, minimum, average, standard deviation or variants of each assessment.
- View projected pass rate based on the assessment marks accumulated students in class.

**The School Administrator should be able to:-**

- Register himself/herself.
- Display or print out table of students and their marks.
- Generate a summary statistics of the performance f the students.
- Generate a comparative chart of the assessment marks of selected courses being taken by students of a particular group.
- Histogram of assessment marks of all courses taken a specific student.

- Any recorded offences (e.g. plagiarism) for a student.
- The performances in the same course across different years may be compared.

**The Student should be able to:-**

- Register himself/herself.
- Display assessment marks for a course and statistics for that assessment.
- Display assessment marks for all the courses
- Based on current assessment marks, give what performance goals are needed to pass the course.

**The Database Administrator should be able to:-**

- Set-up the database.
- Manage and implement relevant API functionalities for accessing the database from a client.
- Responsible for backups and re-configuring the databases etc.

**The System Administrator should be able to:-**

- Configure Hardware, network and software.
- Ensure system security.

### *1.4 List of Definitions and abbreviations*

#### *1.4.1 Definitions*

| **Term** | **Definition** |
|---|---|
| Database | A collection of records stored within the system |
| Table | A collection of related data consisting of columns and rows |

#### *1.4.2 Abbreviations*

**SMMS** - Student Marks/record management system

**HTTP** - Hypertext Transfer Protocol

**HTML** - HyperText Markup Language

**RDBMS** - Relational Database Management System

**SQL** - Structured Query Language

### *1.5 Tools used*

**Web server** - Apache2

The Apache web server is the worlds most used web server software program, it uses HTTP to serve files that form web pages to users in response to their requests. It is an open source program [4, 5].

**Development tools (Front-End)** - HTML, CSS and Javascript

**Development tools (Back-End)** - PHP

PHP is a server scripting language, it is widely used, free, and efficient tool.

**Database Platform** - MySQL and PHPmyadmin

MySQL is an open-source RDBMS, it stores data in tables and PHPmyadmin is a graphical tool intended to handle administration of MySQL over the web.

## 2. Expanded Description of the project

The software system will follow a Two-Tier Architecture due to its ease of use and maintainability as compared to a Three-Tier Architecture. However, the performance of a Two-Tier Architecture slows down with an increase in users [6], hence a Three-Tier Architecture will be implemented with an increase of the number of users. Figure 1 below shows the Two-Tier Architecture.

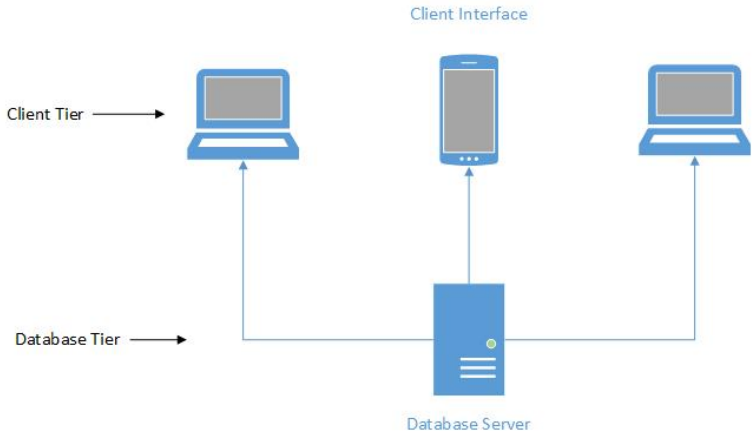


Figure 1 : Two-Tier Software Architecture

A Two-Tier Architecture is a software architecture where the interface runs on client and the data layer is stored on a server[7].

### *2.1 Front-End (Londiwe Ngema and Masana Khosa)*

Khosa Masana (559990) and Londiwe Ngema (448871) are going to be responsible for the design, development and documentation of the front end (user interface).

Both the user interface of the browser and the smart-phone app are going to be designed using HTML5, CSS and JavaScript. The user interface will offer a platform whereby the users are going to interact with the server. The users (i.e students and staff) will be prompted for a student/staff number and a password to view or change the results. Only the staff members (lecturers) will be given the right to change the results, students can only view the results. Validation of credentials will be done in Javascript before they are submitted to the server.

### *2.2 Back-End (Sanele Gcaba and Tshegofatso Misapitso)*

Sanele Gcaba (459380) and Tshegofatso Misapitso (600313) are going to be responsible for the design, development and documentation of the back-end (Server and Database).

The back-end: The proposed design of the back-end is to include a Linux Apache MySQL PHP(LAMP) installation.

The Linux operating system is chosen for the server to run on because of its Stability, Security and Cost of operation [8]. As a result Linux Mint operating system was chosen. The Linux is Just the base of the project that will allow the server to run off. The server proposed is an Apache server. This server is chosen because it is easy to install and operate. Apache will provide a secure, efficient and extensible server that provides HTTP services. The project requires that data is stored and later on read from. There is multiple datasets that need to be considered: for example multiple students that may be doing multiple courses and as a result a need for storing this data. MySQL was chosen because it is an open source database and large companies use it to save money and time powering their high volume websites [9].

PHP is selected as a server the scripting language. This is chosen because of the simplicity of the language and in addition JavaScript can be used to do client side validation to avoid overloading the server with server side validation. Validation would be necessary for authentication.

PHPMyadmin may be used in order to get a visual look of the database instead of having to type out queries in order to check that the data one expects to be in the database is actually in the database. Interacting: the client tries logging in:

- Client enters credentials, these credentials get validated onsubmit.
- If the credentials are correct: a PHP script is used to determine if the user is a student or staff member.
- If the user is a student then the student will only have certain privileges such as reading from the database only.
- If the user is an administrator or a staff member then they may be allowed to have different privileges to edit the database: such as alter student results and add new student onto the database.

### *2.3 Constraints (Londiwe Ngema)*

- The User Interface language is English only.
- Students cannot view test marks, only the final marks.

### 2.4 *ER Diagram (Londiwe Ngema)*

The ER Diagram shows how data is organized within the database.

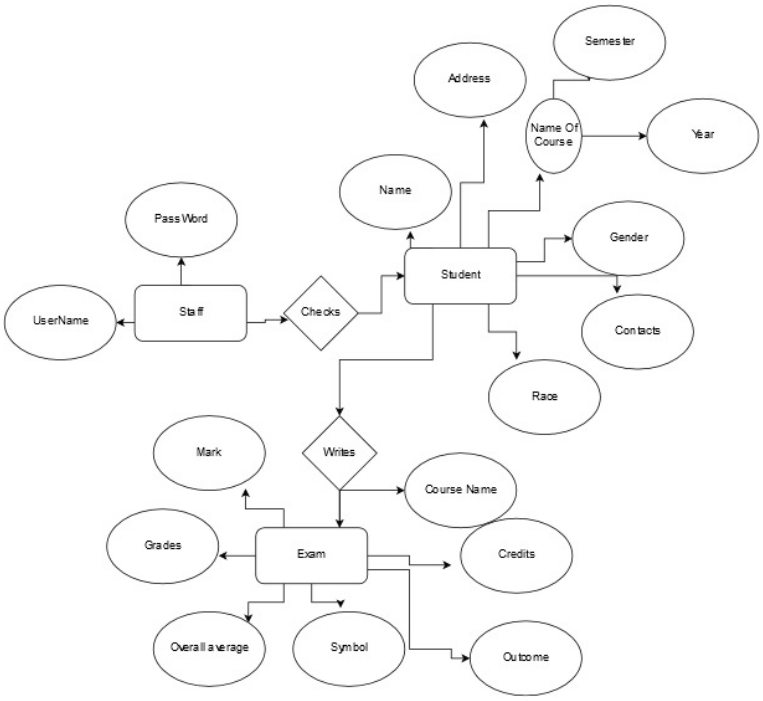


Figure 2 : ER Diagram

### 2.5 *Database Design - (T.Misapitso 600313)*

The design of the database schematic diagram is illustrated in the figure below. It shows the contents database and how they are linked to one another.

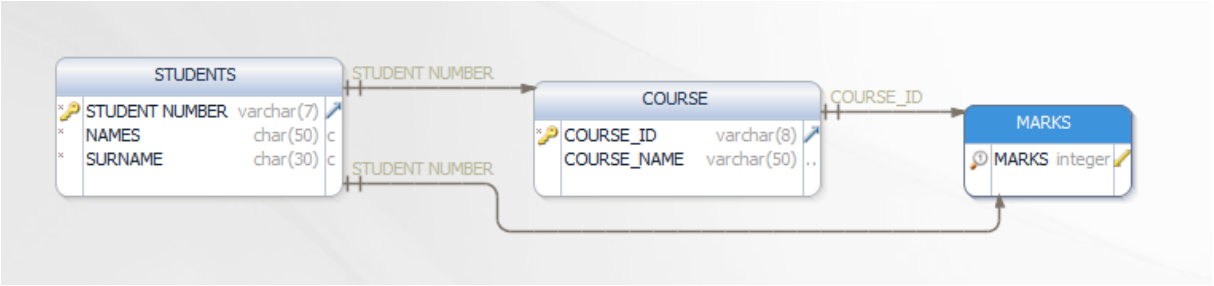


Figure 3 : Database Design

*2.6 Hardware Specifications (T. Misapitso 600313)*

**Minimum Requirements**

**Client side**

| **Windows** | **Processor** | **RAM** | **Disk Space** |
|---|---|---|---|
| Internet Explorer | Intel Pentium III or AMD - 800 MHz | 128 MB | 100 MB |
| Google Chrome | Intel Pentium IV - SSE2 capable | 128 MB | 100 MB |

| **Mac** | **Processor** | **RAM** | **Disk Space** |
|---|---|---|---|
| Google Chrome | 64 bit Intel processor | 128 MB | 100 MB |

| **Linux** | **Processor** | **RAM** | **Disk Space** |
|---|---|---|---|
| Google Chrome | Intel Pentium III processor | 128 MB | 100 MB |

**Server side**

| **Linux** | **Processor** | **RAM** | **Disk Space** |
|---|---|---|---|
| Apache 2 | 2 GHz processor or faster processor | 1 GB (32 bit) or 2 GB (64 bit) | |

**Recommended Requirements**

**Client side**

| **Windows** | **Processor** | **RAM** | **Disk Space** |
|---|---|---|---|
| Internet Explorer | 1 GHz or faster | 1 GB (32 bit) or 2 GB (64 bit) | 256 MB |

| **Mac** | **Processor** | **RAM** | **Disk Space** |
|---|---|---|---|
| Google Chrome | 64 bit Intel processor | 1 GB | 256 MB |

| **Linux** | **Processor** | **RAM** | **Disk Space** |
|---|---|---|---|
| Google Chrome 1 GHz or faster | 1 GB (32 bit) or 2 GB (64 bit) | 256 MB | |

**Server side** Server specifications will depend on the development of the project.

## 3. Specific Requirements

*3.1 UML Activity Diagram*

Figure 2 Shows the UML activity diagram of the how the software should work. As shown in the UML activity diagram, the user will first be ask to enter a user-name and a password. This is done to keep track of the type of user who will be interacting with the software i.e student, staff and admin. Each type of user will be given different rights to interact with the server as shown in the UML diagram. Students are only allowed to view the results, staff is allowed to change marks or add results and the administrator will be responsible of adding or removing users.

*3.2 Use Case Reports (Masana Khosa)*

Since we have only three users of the software, namely student, staff and administrator, three use case reports have been made for each user.

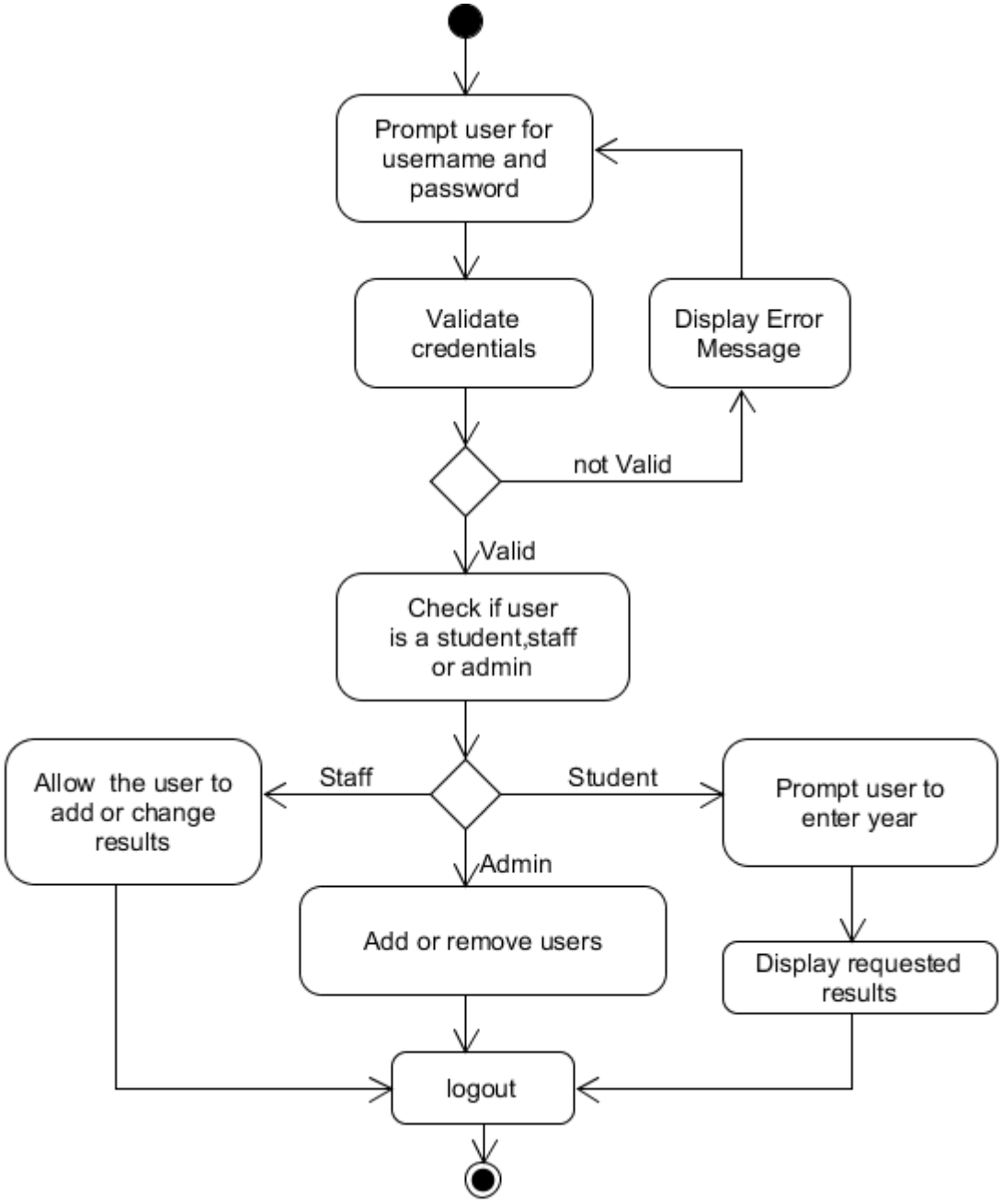


Figure 4 : UML Activity Diagram

### 3.2.1 *Student Use-Case Report*

A use case report for a student is shown in Figure 3.

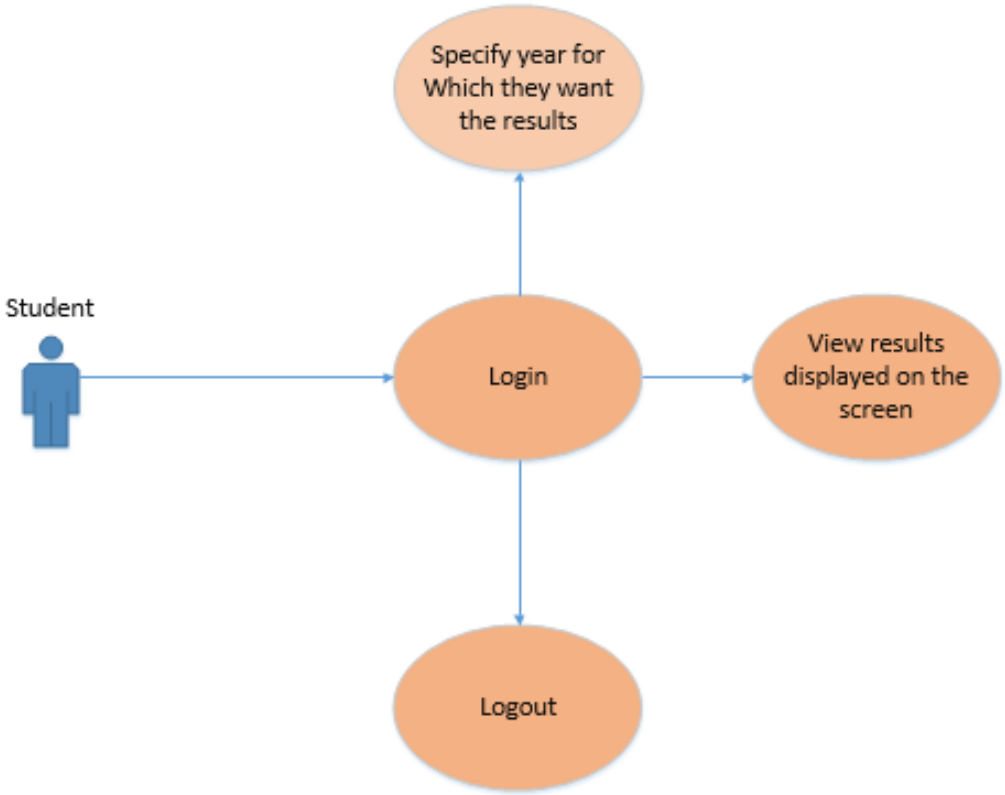


Figure 5 : Use Case Diagram For Student

| **Use case** | **Description** |
|---|---|
| login | The student need to sign in in order to view the results |
| Key in Year | The student must be able to specify the year which they need the results for |
| Display results | The student must be able to view the results displayed on the screen |
| Logout | The student must be able to logout |

### 3.2.2 *Staff Use-Case Report*

A use case report for staff is shown in Figure 4.

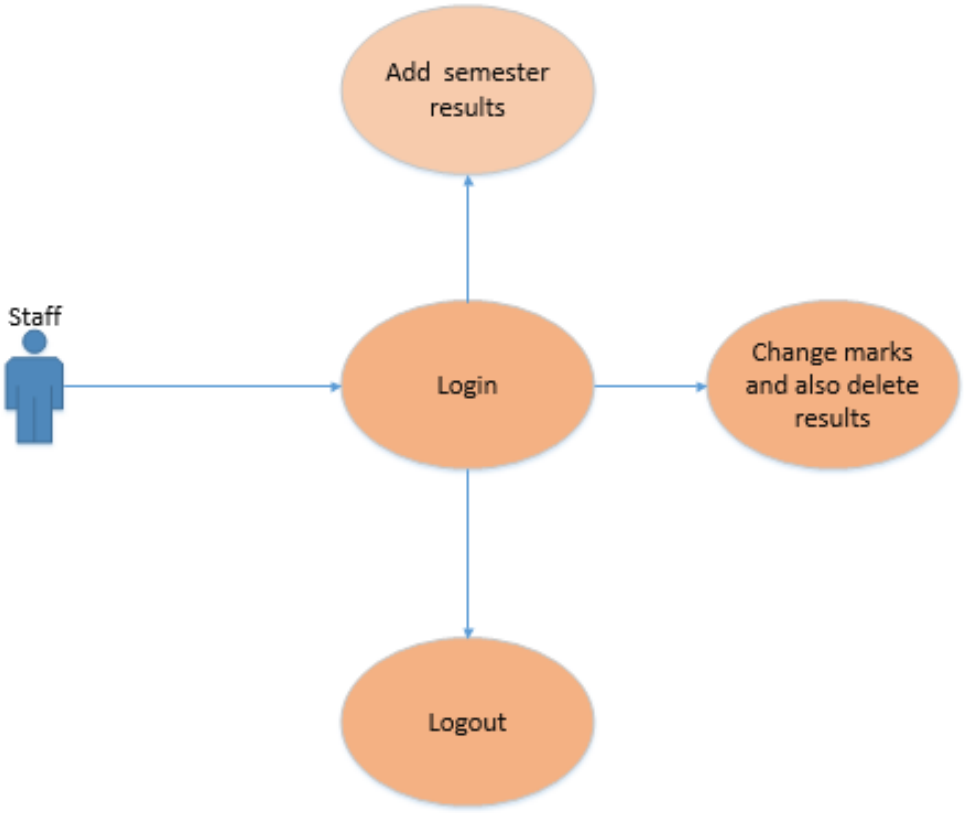


Figure 6 : Use Case Diagram For Staff

| **Use case** | **Description** |
|---|---|
| login | Staff need to sign in order to modify results |
| Change Results | Staff must be able to modify or add results |
| Logout | Staff member must be able to logout |

### *3.2.3 Administrator Use-Case Report*

A use case report for an Administrator is shown in Figure 5.

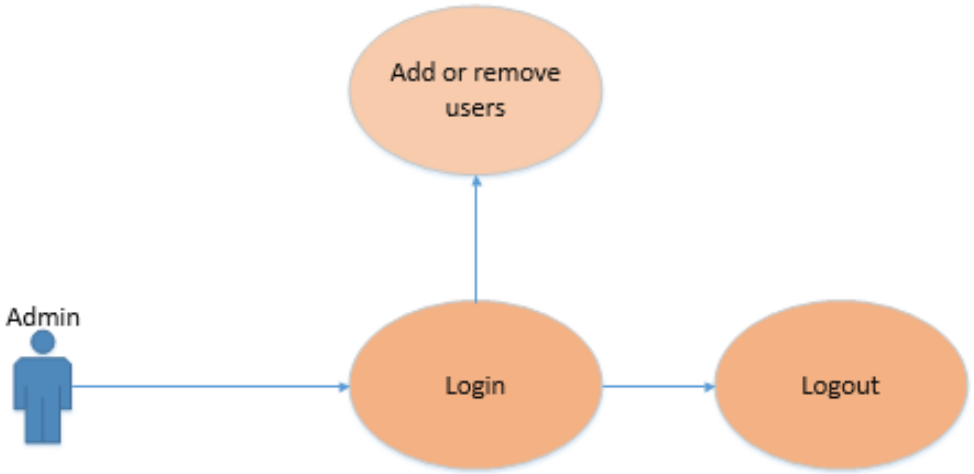


Figure 7 : Use Case Diagram For Administrator

| **Use case** | **Description** |
|---|---|
| login | Administrator need to sign in |
| Add users | The administrator must be able to add or remove users |
| Logout | Administrator must be able to logout |

## 4. Sequence Diagram

The sequence diagram in Figure 6 shows how each user interact with the interface and how the interface interact with database.

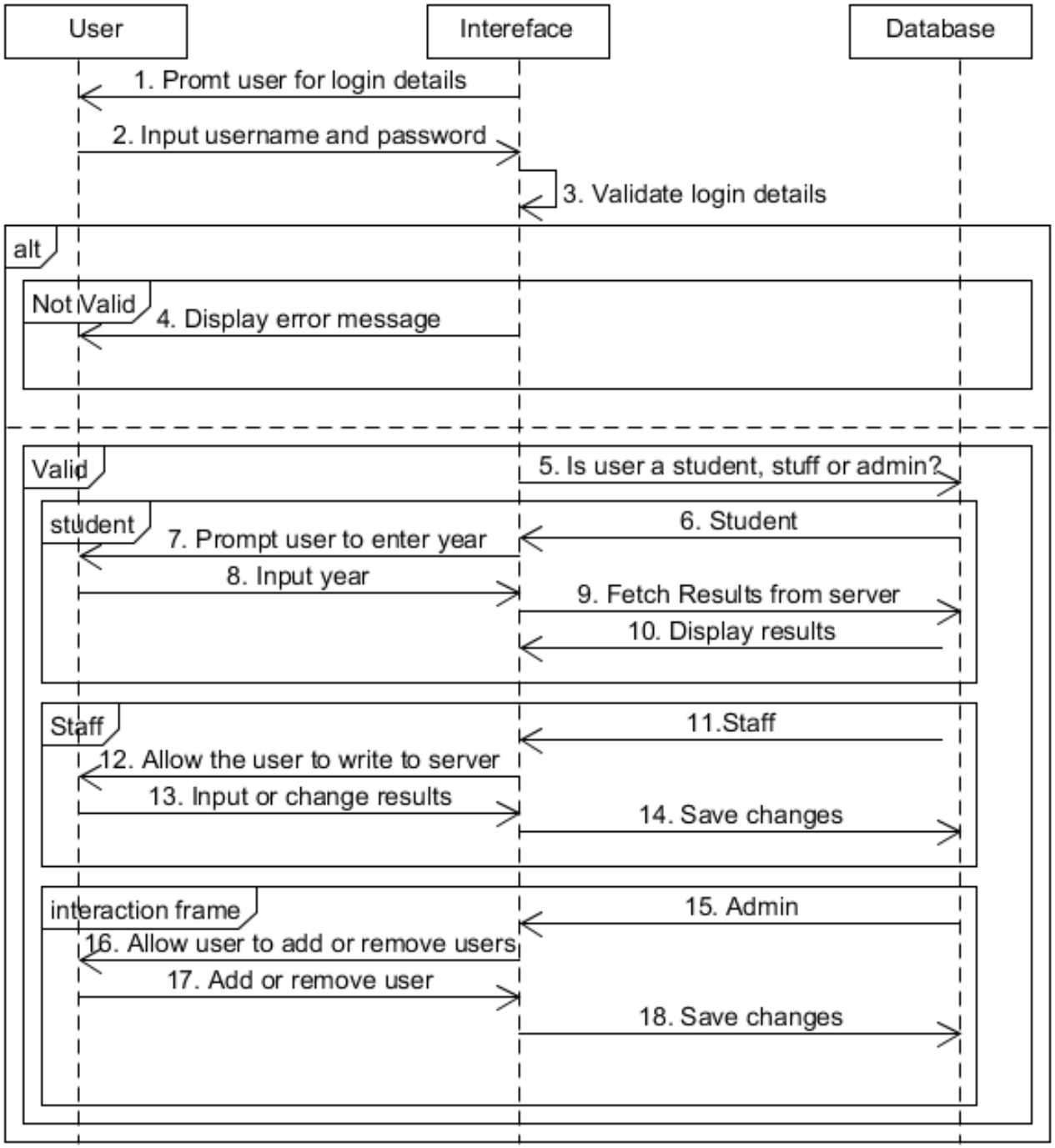


Figure 8 : Sequence Diagram

## REFERENCES


[1] R. H. Technology. "6 Basic SDLC Methodologies: Which One is Best?" https://www.roberthalf.com/technology/blog/6-basic-sdlc-methodologies-the-pros-and-cons, 2014. [Online; accessed 10-March-2016].

[2] VERSIONONE. "Understanding Scrum Project Management." https://www.versionone.com/scrum-project-management/, 2016. [Online; accessed 10-March-2016].

[3] E. Otoo. "Expanded Project Description." https://cle.wits.ac.za/access/content/group/ELEN4009_SoftwareEngineering2016/Project%20Descriptions/EnhancedProjDesc.pdf, 2016. [Online; accessed 22-March-2016].

[4] Apache. "Apache - HTTP Server Project." https://httpd.apache.org, 2016. [Online; accessed 9-March-2016].

[5] WhatIs.com. "Web server." http://whatis.techtarget.com/definition/Web-server, 2016. [Online; accessed 9-March-2016].
[6] G. Palaniswamy. "Two-Tier and Three-Tier Architecture with example." http://www.c-sharpcorner.com/UploadFile/gowth/two-tier-and-three-tier-architecture-with-example/, 2010. [Online; accessed 2-March-2016].
[7] techopedia. "Two-Tier Architecture." https://www.techopedia.com/definition/467/two-tier-architecture, 2016. [Online; accessed 2-March-2016].
[8] PCWorld. "Five Reasons Linux Beats Windows for Servers." http://www.pcworld.com/article/204423/why_linux_beats_windows_for_servers.html, 2015. [Online; accessed 24-February-2016].
[9] MySQL. "Why MySQL?" https://www.mysql.com/why-mysql/, 2015. [Online; accessed 24-February-2016].