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

1. Project Scope

- There are three basic users of the system - Students, Course coordinators and School 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 information and they are allowed to view/edit/add based on what their 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 [**?**].

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.

2. List of Definitions and abbreviations

2.0.1 Definitions

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

2.1 Abbreviations

SMMS - Student Marks/record management system

HTTP - Hypertext Transfer Protocol

HTML - HyperText Markup Language

RDBMS - Relational Database Management System

3. 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 [1], hence a Three-Tier Architecture will be implemented with an increase of the number of users. Figure 1 below shows the Two-Tier Architecture.

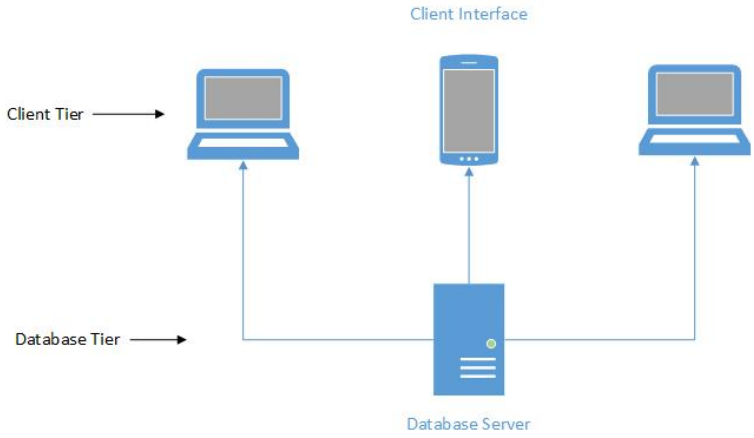

Figure 1 : Two-Tier Software Architecture

A Two-Tier Architecture is a software architecture where the interface runs on client and the data layer is stored on a server[2].

4. Responsibilities

Masana Khosa and Londiwe Ngema are responsible for the front end, they designed the website page and the interfaces for the individual users. They also created domains to ensure that the user only has access to what they need.

Sanele Gcaba and Tshegofatso Misapitso are responsible for the back end that is the database. The next step is to do the querying logic to link the back and front ends. The program is now able to log the registered users in and out based on the information from the database. Different interfaces can now be called upon for different users, i.e different interfaces for the school administrator, course co-ordinator and the students.

5. The current prototype

5.1 Front-End

The current prototype allows users to log in and they are redirected to the different pages based on the scope above. Since there are types of users with different privileges, they are distinguished using a domain. A student cannot log on the system as a stuff member and vice versa. When a user logs out they are directed to the home page.

5.2 Back-End

The developed Back-End constitute of a LAMP web server, this is hosted within a PC and accessible using a local host of a machine. The server that hosts the database was created using PHPmyadmin. The Back-end has achieved creating a database for users, this database has a Username, Password and Domain. The back-end team has been able ensure a link between the front end interface and the login database.

Figure below shows the user database design.

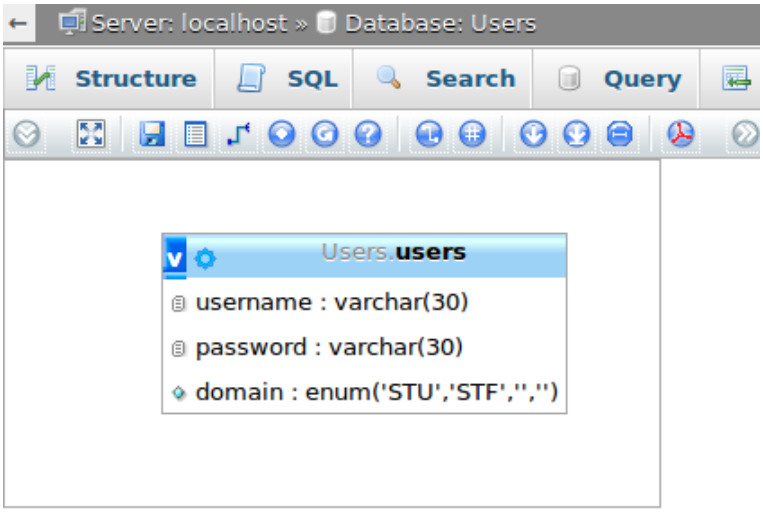

Figure 2 : User Database

Additionally a student record Database was created and a figure below shows a structure for the database

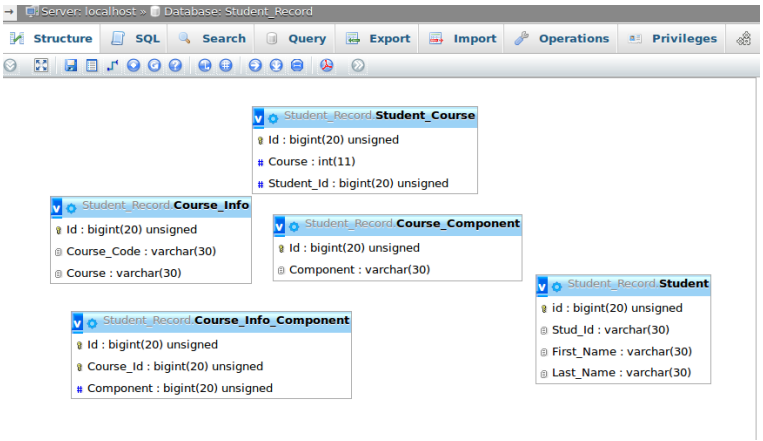

Figure 3 : Student Record Database

REFERENCES

[1] 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].

[2] techopedia. "Two-Tier Architecture." https://www.techopedia.com/definition/467/two-tier-architecture, 2016. [Online; accessed 2-March-2016].