SOFTWARE ENGINEERING: STUDENTS MARKS/RECORDS MANAGEMENT SOFTWARE

Khosa Masana 559990, Sanele Gcaba 459380, Tshegofatso Misapitso 600313, Londiwe Ngema 448871

1. Project Requirements

The aim of the project is to design, implement and document a student Marks/Record management system. This would be accessed online via a Browser and a Smart-phone App. The primary function of the application is to allow students to log-in using their details and be able to access their course marks. The users will be able select the academic year which they require marks for and the specific courses, course code, mark and symbol will be displayed to them. The program should have domains i.e. staff members can be able to access and edit the data base while students can only view their individual marks.

2. Expanded Description of the project

2.1 Front End

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.

With the student number, password and year entered as an input, the relevant information(results) of that particular student and the particular year will be searched from the server and displayed to the student. Results will be displayed in a form of a table with course name, course code, results and a symbol as columns. PHP will be used to link back end and front end

2.2 Back End

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

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

The Linux operating system is chosen for the server to run on because of its Stability, Security and Cost of operation [1]. As a result centOS (minimal), a Linux 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. This 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 [2].

PHP is selected as 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 querries in order to check that the data one expects to be in the database IS ACTUALLY IN THE DATA BASE. As a result PHP my admin could be used in testing of the script in terms of database manipulations.

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 will only have certain privellages 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.

REFERENCES

[1] 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].

[2] MySQL. "Why MySQL?" `https://www.mysql.com/why-mysql/`, 2015. [Online; accessed 24-February-2016].