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)

March 9, 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.

.

<h1 align="center">1.   Introduction</h1>

1.1   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 to students and allows staff the rights of editing course marks on the system. 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

1.2   Project Scope

- There are three basic users of the system - Students, Staff (Lectures) and Admin.
- The primary function of the application is to allow students to log-in using their details (Student number and password) and be able to access and view their course marks as per requested year of study and courses registered for.
- The system database stores user profiles and student marks records.
- The program should have domains assigned by Admin i.e. Staff members can be able to access and edit the database while Students can only view their individual marks. The system would be accessed online via a Browser and a Smart-phone App.
- Staff log-in using a staff number and password gives rights of editing student records through the student database.
- The students 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 through a user interface.
- Admin can add/delete users, grant editing and viewing permission to staff and only viewing rights to students.

1.3   List of Definitions and abbreviations

1.3.1   Definitions

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

1.3.2   Abbreviations

SMMS - Student Marks/record management system

HTTP - Hypertext Transfer Protocol

HTML - HyperText Markup Language

RDBMS - Relational Database Management System

SQL - Structured Query Language

1.4   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 [? ?].

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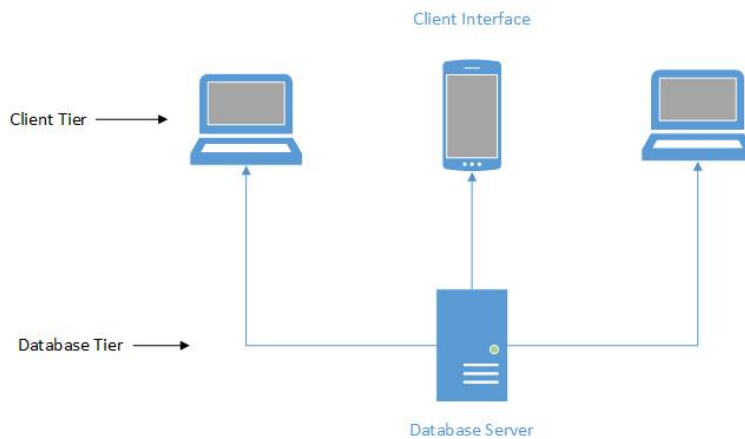

Figure 1 : Two-Tier Software Architecture

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

2.1   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.

2.1.1   *System Input and Output:*   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 (Output) 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 the 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 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 [?]. 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 [?].

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.