Software Requirements Specification

Name: Group 7

August 27, 2017

Contents

1 Introduction

This section provides a general summary of everything included in this SRS document.

1.1 Purpose

The purpose of this document is to present a detailed description of the Student Marks Management System software that will guide other developers to be able to design a full-scale application from this documentation.The purpose of the system is to help the student to track his/her academic progress through out the year and to minimize the errors in recorded marks.It will describe the features and the interfaces of the software and what the software will do and how the software will respond external input.The software will be built on HTML,PHP,CSS,JavaScript and PostgreSQL. It will be accessible with any browser.

1.2 Scope

This document covers the requirements for the Student Marks Management System. The software will provide a platform in which the users will be able to enter,update,retrieve,manage and archive the tentative(unofficial) Student marks as the students complete various assessments throughout the year. The software will record the student's, course-coordinators and the school administrator's information. It will also record courses information.

1.3 Definitions and abbreviations

Term	Definition
HTML	Hypertext Mark-up Language
PHP	Hypertext Preprocessor
CSS	Cascading style sheet
SRS	Software Requirements Specification
PostgreSQL	an object-relational database management system
Apache	a web server software
DBMS	Database Management System

1.4 Reference

A Concise Introduction to Software Engineering by Pankaj Jalote

1.5 Overview

This document has the general and the detailed requirements specification for the front-end, which are mostly the user interface requirements. The document also has the back-end requirements such as Database requirements. The scope of the system, definitions, the purpose of the document and the purpose of the system. It also includes a chapter that provides a general view of the software functionality and also introduces different users and their interaction with the system.

2 Overall Description

2.1 Product Functions

Our software will have

1) Student
Students will be able to register online with an email address and password which will then be authenticated with the information in our database and the user added to the database.

Students will able to display their marks and statistics for a given course, at the same time be able to view marks for all their courses and get performance goals needed to pass their respective courses.

2) Course Coordinator
Course Coordinators will be able to register online with an email address and password which will then be authenticated with the information in our database and the user added to the database.

Course Coordinators will be able to add various assessment methods for the course and the weighting for each, enter student marks for each assessment and display or print out the table of students and their marks.

Course Coordinators will be able to generate a summary statistics of a performance of the students and the projected pass rate based on the accumulated assessment marks.

3) School Administrator
School Administrators will be able to register online with an email address and password which will then be authenticated with the information in our database and the user added to the database.

School Administrators will be able to display or print out the table of students and their marks, generate a summary statistics of the performance of the students, generate comparative charts of the assessment marks of courses taken by a particular group of students.

School Administrators will be able to display histogram of assessment marks for all courses taken by a specific student and display any recorded offences for a student. And lastly, they will be able to generate the performance statistics of a particular course across different years.

2.2 User Characteristics

Though our website will be user friendly, a basic knowledge of using a computer and an internet browser is required to use this system as it is a web based application.

2.3 General Constraints

- An internet connection is a constraint since the website will retrieve data from the database over the Internet.
- Another constraint is that the users should be registered in the university system as either staff or students.
- It is hard to get community support for PostgreSQL.
- The ram of the hardware component of the server that will be hosting the application will be a constraint, because it determines the number of concurrent users the application can have.

2.4 Assumptions

We assume that the user have a basic computer knowledge and familiar with using an internet browser. We also assume that the students.

3 Requirements

3.1 General Requirements

- Apache shall be used as the system's server.
- The system shall be developed using PHP, HTML, CSS and Javascript.
- PostgreSQL shall be used as the Database Management System of the web application.
- The system shall have a login page.
- The system shall have a sign up page.
- If logged in as a course coordinator, the system shall allow administrator privileges.
- The system shall be able to produce graphs of the course statistics such as the maximum, minimum, average and standard deviation for each course.
- The system shall be able to store the marks that are entered by the course coordinator into the database.

- The system shall be able to give a student his marks if requested by that student.
- The system shall produce the marks currently needed for student to pass.
- The system should print out the table of students and their marks. When requested by the school administrator.
- The system should display marks for all assessments if it is requested by the student.
- The system should be able to register the user as the school administrator, course coordinator or student.
- If the user is a school administrator or course coordinator the system shall allow the user to change and view the plagiarism status of a student.

3.2 User Interface Requirements

3.2.1 Login page

- The login page shall have a form with an input field for email address.
- The login page shall have a form with an input field for password.
- The login page shall have a link to the sign up page.
- The login page shall have a button named "LOG IN".
- If the credentials entered by the user are correct, the login button shall take the user to the home page when pressed.
- If the credentials entered by the user are incorrect, the login button an error message will appear.
- The login page should have a header with the text "Record Management System".

3.2.2 Sign-up page

- The sign-up shall have an input form that takes in the email address.
- The sign-up shall have an input form that takes in a password.
- The sign-up shall have an input form for confirmation of password.
- The sign-up shall have an input form that takes in the name.
- The sign-up shall have an input form that takes the surname.
- The sign-up shall have an input form that takes in the name of the school that the user is in.
- The sign-up shall have an input form that takes in the occupation of the user.

3.2.3 Home page for students

- The home page shall have a logout button that takes you back to the sign-up page.
- The home page shall display a table with all the student's courses.
- The home page shall have a button with the text "view all course marks"
- Each course shall have a button next to it with the text "view marks "
- When clicked, the "view marks" button shall take the user to a screen with all the marks for each assessment, perfomance goals and statistics for that particular course.

3.2.4 Home page for Course Coordinator

- The home page shall have a logout button that takes you back to the sign-up page.
- The home page shall have a "Add assessments" button.
- When the "Add assessments" button is clicked, a pop-up with an input form will appear where the user can add a course into a table.
- When the "Add assessments" button is clicked, a pop-up with an input form will appear where the user can add the weight of the assessment into a table.
- When the "Add assessments" button is clicked, a pop-up with an input form will appear where the user can add the marks for each assessment into a table.
- When the user clicks on the button with the text "display statistics" on it, the summary statistics of the performance of the students will be displayed by the system.
- When the user clicks on the button with the text "display projection" on it, the projected pass rate based on the assessment marks will be displayed by the system.

3.2.5 Home page for School Administrator

- The home page shall have a logout button that takes you back to the sign-up page.
- When the user clicks on the button with the text "Display Marks" on it, a table with the students' marks shall appear.
- When the user clicks on the button with the text "Display Statistics" on it, the summary statistics of the performance of the students will be displayed by the system.

- When the user clicks on the button with the text "Generate Comparative Chart", a comparative chart of the assessment marks of selected courses being taken students shall be displayed by the system.
- When the user clicks on the button with the text "Display Histogram" on it, a histogram of assessment marks of all courses taken by a specific student shall be displayed by the system.
- When the user clicks on the button with the text "Check Plagiarism Status", The status of recorded offences for plagiarism will be displayed by the system.
- When the user clicks on the button with the text "Display Yearly Statistics " on it, a table with the students' marks shall appear.

3.3 Back-end Requirements

- The back-end system shall be able to store the data coming from the input forms into the database.
- The back-end system shall be able check if the password sent from the user interface is the same as the one stored in the database.
- When a student request his or her marks for a particular course, the back-end system shall be able to send the individual marks of that student.
- The back-end system shall be able to send the list of class marks of students to the school administrator.
- The back-end system shall be able to calculate the standard deviation, maximum,minimum and the average of the marks of the class.
- The back-end system shall be able to send the standard deviation, maximum, minimum and the average of the marks of the class to the client-side of the application.
- The server shall be able to receive data from concurrent user connections.
- The server shall be able to send data concurrently to users when requested.
- The database management system shall follow the ER-Model, where the entities are the student, the school administrator, the course coordinator and the courses.

3.4 Database Requirements

3.4.1 User's table

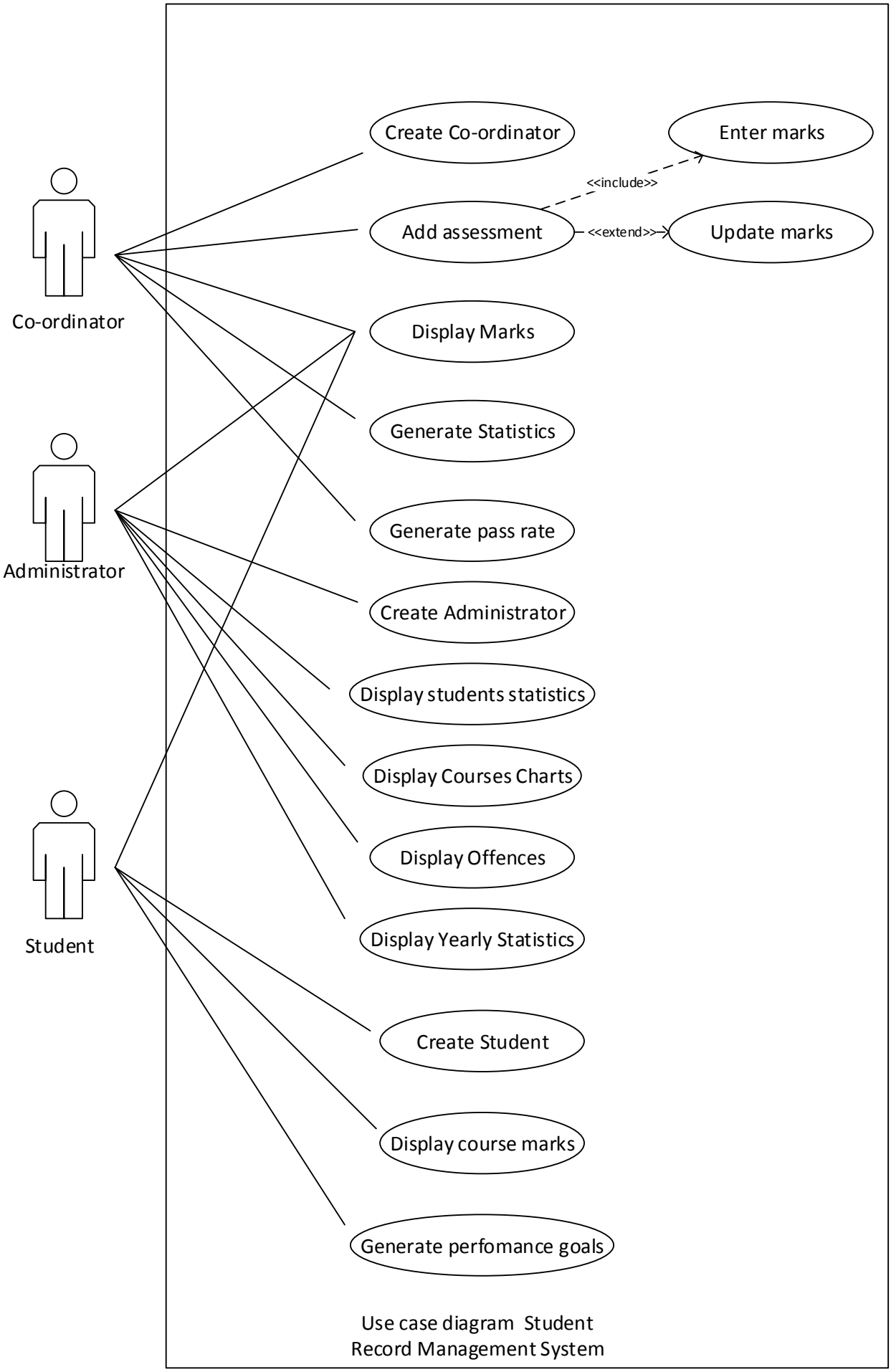
- The user's table must have a field titled email address that stores input data from the form with the same name.

- The email address field must be the primary key that stores input data from the form with the same name.
- The user's table must have a field titled name that stores input data from the form with the same name.
- The user's table must have field titled surname that stores input data from the form with the same name.
- The user's table must have field titled school that stores input data from the form with the same name.
- The user's table must have field titled "occupation" that stores input data from the form with the same name.
- The user's table must have field titled password.

3.4.2 Student table

- The student table must have a field titled email address.
- The field email address will be the primary key.
- The student table must have a field titled name.
- The student table must have a field titled surname.
- The student table must have a field titled total marks.

3.5 Use Cases



Use case diagram Student Record Management System