

PROJECT REPORT

SOFTWARE SYSTEM DESIGN DOCUMENTS

Group 1: Zehao Liu, Jialong Zhang, Huiqiao Yang

Course: cp476 Internet computing

Instructor: Shawn Gao, PhD

Wilfrid Laurier University

Revision History

Date	Version	Description	Arthur(F+L)
Jun6	0.2	Discuss backend file and code	ZL, JZ, HY
July13	0.4	Wrote introduction part	ZL
July21	0.6	Wrote General Description part	JZ
July22	0.8	Wrote Main Design Features	HY
July23	1.0	Wrote Application Architecture	ZL
July24	1.2	Wrote Technology Architecture	JL
July25	1.4	Wrote Standard	HY
July25	1.6	Wrote Database design	JL
July26	1.8	Wrote User Interface, Error Handling, Security	ZL
July27	2.0	Wrote Maintainability	HY
July27	2.2	Wrote conclusion	ZL, JZ, HY

1. Introduction

1.1. Purpose

1.2. Scope

1.3. Definitions

1.4. Overview

1.5 reference

2. General Description

2.1. Product Perspective

2.2. Tools used

2.3. General Constraints

2.4. Assumptions

3. Design Details

3.1. Main Design Features

3.2. Application Architecture

3.3. Technology Architecture

3.3.1. Web Application Architecture

3.3.2. Presentation Layer

3.3.3. Data Access Layer

3.4. Standards

3.5. Database design

3.6. Files

3.7. User Interface

3.8. Error Handling

3.9. Portability

4. security

4.1. Performance (auto update and refresh: auto check login states & log out)

4.2 prepared statement

4.3 string escape

5. Maintainability

6.conclusion

1.Introduction

1.1 Purpose

The Document describes the architecture and system design for Student Information Management System (SIMS), a web application provides a reliable connection between web browser and database server. All current and future developers, especially implementers, carrying out future development should follow the guidelines of this document.

1.2 Scope

The documentation presents the system's structure, including the database architecture, web page architecture, and interaction architecture. This document uses mildly-technical terms, which should be understandable to the system administrators.

1.3 Definitions, Acronyms OR Abbreviations

SIMS: Student Information Management System

LOG: log in and log out

internal web pages: all web pages except login.php

Entity: each table in database

Utility: function render, insert, delete, edit

CI: contact_information

SS: scholarship

SP: specialization

1.4 overview

To provide a user-friendly environment, web pages are used by the system to manage student information effectively. The system can be used without programming knowledge and handle some text preprocessing to avoid irregular behaviors. SIMS will consist of six primary functions: home, search, insert, delete, update, and LOG. Each function will be explained in detail in this document. All components in SIMS will be programmed without third-party language.

1.5. Reference

Shaun, G. (2021) CP476B Internet Computing-ProjectReportExamples-web-SDD_RoadTrip. Mylearningspace.

<https://mylearningspace.wlu.ca/d2l/le/content/439248/viewContent/2983559/View>

Shaun, G. (2021) CP476B Internet Computing-week-1e 1.Mylearningspace.

<https://mylearningspace.wlu.ca/d2l/le/content/439248/viewContent/2954217/View>

Shaun, G. (2021) Database_Design_Document. Mylearningspace.

<https://mylearningspace.wlu.ca/d2l/le/content/399848/viewContent/2661730/View>

Shaun, G. (2021) CP363A Database-I-week-5-2-DB-design-1. Mylearningspace.

<https://mylearningspace.wlu.ca/d2l/le/content/399848/viewContent/2685191/View>

Shaun, G. (2021) CP363A Database I-week-6-3-Tutorial. Mylearningspace.

<https://mylearningspace.wlu.ca/d2l/le/content/399848/viewContent/2698623/View>

Shaun, G. (2021) CP363A Database-I-week-7-1-DB-design-4. Mylearningspace.

<https://mylearningspace.wlu.ca/d2l/le/content/399848/viewContent/2699245/View>

2. System overview

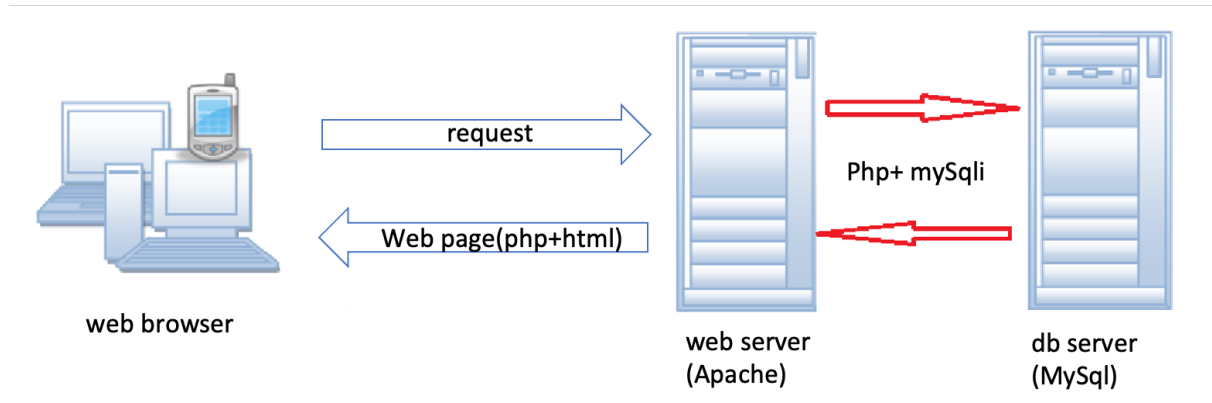


Figure1

2.1. Product Perspective

Figure 1 illustrates how the servers and the browser interacts. The front is the web page the user can interact with based on the HTML, and the backend is the relational database storing user data based on MySQL.

The administrative and user interfaces will use PHP to display the pages and MySQL to retrieve, insert, delete, and update the database. MySQL will be used to submit SQL commands for database management. The local Apache server is responsible for receiving data from MySQLi. PHP will allow multiple users to log in and interact with the program simultaneously due to its multithreading property. SIMS will be set up using five user levels, and each group will have different permission to view or change data. Approval will be illustrated in section 2.3.

2.2. Tools used

1. mySQL, a relational database backend that is SQL based.
2. Interfacing with the database to display information on the user's web browser will be done using PHP. It can connect to the database and parse it into viewable HTML code.
3. Apache, An open source web server that will display requested pages.

4. Lucid app, an online intelligent diagramming application
5. visual studio code is the development platform

2.3. General Constraints

The system running based on Apache, MySQL and PHP embedded in html.

Only university administrators have full access to all the data. A login password verification function is required, and no user can access the internal web pages without login. Figure 2 indicates which is available for a different identity. R represents render or read, a represent add, d represents delete, and e represents an edit.

entity	student				CI				scholarship				specification			
utility	r	a	d	e	r	a	d	e	r	a	d	e	r	a	d	e
admin	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
student	√			√					√				√			
advisor	√				√								√	√	√	√
professor	√				√								√			
service	√	√	√		√				√	√	√	√	√			

Figure2

2.4. Assumption

This is the second iteration of the application, and the assumption for SIMS is a software developed based on the web server and omits the other function, which allows developers to release a minimal viable product quickly.

3. System component

3.1. Main Design Features

-logIn.php: ask the user to select identity, enter username and password. It will automatically destroy the session if the session exists

-index.php: navigate user to different utilities including xxRender, xxInsert, xxDelete, xxEdit.

-logOut.php: An error handling page direct user back to login page if user status is not logged.

-_function.php: internal helping functions for logIn.php and index.php. It has three functions, including timer, filter, logout. Timer records the times the same user accesses the index page. Filter filtrate all unexpected input. Logout will destroy session information.

-connect.php: use mysqli to connect web server and mysql. It is used to connect frontend and backend.

-preState.php: a file containing all prepared statements

-xxRender.php: use direct statements show all data in database. It also has a method to select specific row with given primary key.

-xxInsert.php (insert): it is used to add new rows in database, and it executes insert sql statement.

-xxDelete.php (delete): it is used to delete existing rows in database, and it executes deleted sql statement.

-xxEdit.php (update): it is used to edit existing rows in database, and it executes an update statement.

search, insert, delete, edit in above four utilities used prepared statement since they all have binding parameters.

Notes: xx represents entities name. ('student'/'SS'/'SL'/'CI')

3.2. Application Architecture

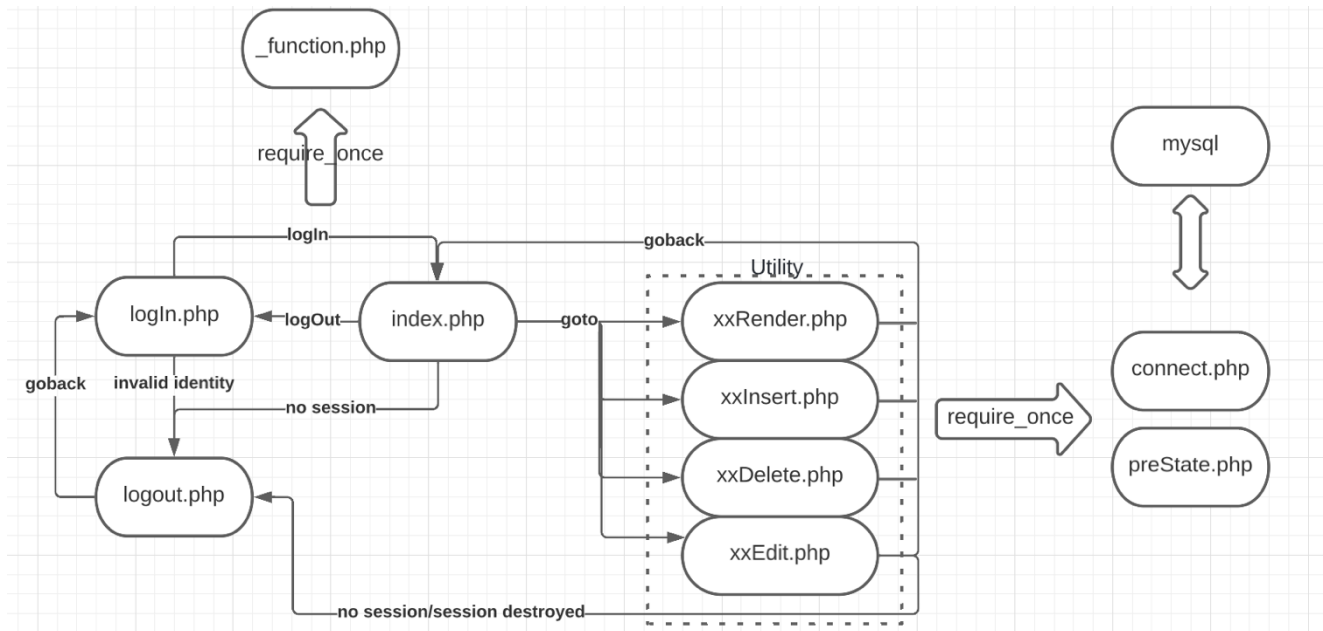


Figure3

Figure 3 demonstrates how the system works. Users start from `logIn.php` to direct log into `index.php`. Both `logIn.php` and `index.php` require `_function.php` once to access the helper function. `Index.php` can go to any utilities. These utilities need `connect.php` and `preState.php` once to connect to the MySQL database. All internal web pages have a `goback` option to return the upper layer PHP file, and these pages also auto-check whether the user status is logged or not so that users will not be able to access any page if they know the web page address.

3.3. Technology Architecture

3.3.1. Web Application Architecture

The system uses Apache and a local server for the web server. Through the server, application rights such as adding, deleting, and updating data can be performed, and ordinary users cannot change and delete data information without logging in. The system pages are built with HTML and PHP. Through a web browser, users can access the system and view the information in the

relevant sections. MySQL is used as the database to store the information, and the administrator can view and manage all the data information in the database.

3.3.2. Presentation Layer

Information will include all relevant information under student, contact, scholarship and specialization. Users can choose to search by id, add, delete and modify information within the scope of their authority.

3.3.3. Data Access Layer

Users can access data information by logging in. They can confirm changes in the browser but cannot access the database directly to make changes. Only the system developer can directly change the database.

3.4 Standards

Database – Relational.

Login – Select correct identity and enter account and password.

Input: Entered through text fields and stored in a database, users can query using the system's students or manually update and modify based on their information.

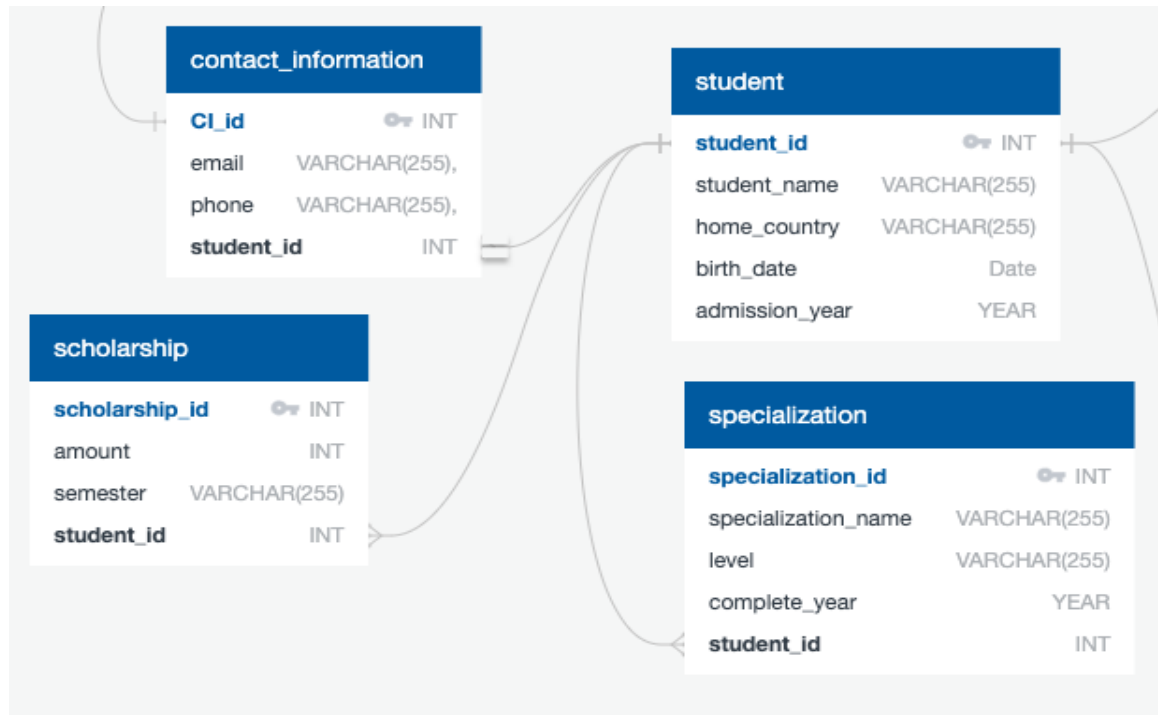
Output - The successful modification information is directly prompted through the web page and can be viewed.

Security - Access to the system requires the correct username and password.

Quality - the interface is straightforward, with no additional loading.

3.5. Database design

The student has contact information, student receives scholarship, student chooses specialization, and the student has contact information. The relationship between the student and the contact information is a one-to-one model. All other connections are in a one-to-many model.



3.6. User Interface

The user interface is a straightforward and clear structure, mainly through text input and mouse click, without redundant functions. SIMS can output information to the user clearly, and quickly help users find the place that needs to be modified. It mainly outputs information to users through HTML pages and prompts the user for incorrect input.

3.7. Error Handling

If a student encounters an error logging in, the page will display the error description. Errors will not affect the data in the database.

3.8. Portability

The system is portable, highly adaptable, and can be used for any operating system if PHP, Apache, and MySQL are installed. The built-in connection file can easily connect to the database, and the file does not take up much space. However, to perform system optimization or page modification, you must make code changes from the source file.

4. Security

4.1. Performance

Each PHP file will automatically check whether the user is logged. As the user log out, all session history in the web server for the user will be automatically removed. Users cannot access an internal web page without creating a session on the web server or the destroyed session.

4.2 prepared statement

As the document introduced in section 3.3.1, all SQL statement required binding parameter before execution use prepared statement. SQL injection can be efficiently defended.

4.3 string escape

String escape convert converts predefined characters to HTML entities which means script language could not be executed in any input.

5. Maintainability

In scalability, the system takes only four entities on the database. With the increasing demand for universities or students, entity size should be more larger.

Current password verification is incomplete, and only the identity will be verified. Further implementation should create an account verification database to store and check different user accounts and passwords.

In the advanced, the project is deployed on the local server. With the increasing size of the user and portability requirement, SIMS should be deployed on the web server.

6.conclusion

The student information management system is a critical system that supports the university in collecting student information and it is mainly used to maintain the daily operation of the university. This system is easy to operate and has a clear web page. Because the system settings have low technical requirements for users, it is relatively straightforward for administrators to maintain. During the development process, the web page design can be quickly changed according to the different needs of the university to collect information to make the data collection more standardized.