

Online University Registration System for SMU

Abel Asmare, Abenezer Desalegn, Dagmawi Yewlsew, and Hamrafel Wogi, St. Mary's University

1. Introduction

1.1 Background

St. Mary's University (SMU) is an outgrowth of St. Mary's Language School, which was established in 1991 in Addis Ababa, Ethiopia. The Language School was upgraded to a Language Center in 1995 with an exclusive focus on the improvement of the English language proficiency of students, establishing itself as a leading language center in the capital. It was in this language center that the University was born.

St. Mary's University was established in 1998 under St. Mary's University General Educational Development PLC with its head office in Hawassa and a branch in Addis Ababa.

The registration process that is being used currently is mainly using paper. During registration period, students collect their respective slips that contain the courses that they are going to take and the total amount of money they need to pay. Then they go to banks that are working with the University and pay the necessary amount of money. Afterwards they take the receipt of the payment that they have made to the University back to the Registrar Office; their slips are stamped at the Registrar Office, and one copy of the slip remains there. Finally, they give one copy to their respective department and keep the remaining one slip for themselves.

1.2 Statement of the Problem

The current registration process is mainly carried out manually. This system does not allow students to register at their convenient time and place with in the registration schedule. In addition, it takes a lot of time and effort as students have to go back and forth between the bank and the Registrar Office to pay the necessary payment. Furthermore, students are expected to register in person every semester. Another problem students face is that they cannot register any time they want within the registration time, they can only register during weekdays and office hours.

On the other hand, the Registrar Office is always crowded during registration period due to the errors made by the registrar workers.

All these reasons cause an inconvenience both for the students, registrar workers and the university in general.

1.3 Objective of the Project

The general objective of this project is to automate the current manual system carried out by the university to register students using our online university registration system.

The specific objectives for our project are listed below:

- To reduce the time students take to register;
- To reduce the inconvenience students face during registration;

- To allow students to be able to pay all the necessary fees using their bank accounts, without the need to physically go to a bank;
- To avoid the need to collect registration slips from the university; and
- To help the University in reducing the labor needed to give services to students particularly during registration period.

1.4 Feasibility Study

For this project, we have considered two types of feasibility studies: technical and economic feasibility studies.

1.4.1 Technical Feasibility

Technical feasibility is about whether the technical resources needed to complete the project are met by the team members. For this project, the team, has all the necessary technical resources needed, whether it is technical knowledge, hardware or software needs, to accomplish the project we are undertaking. Therefore, our project is technically feasible to carry out.

1.4.2 Economic Feasibility

Economic feasibility is concerned with the overall cost of the project. We will be using free (open source) software tools to develop the proposed web-based system; therefore, there will not be any software cost for this project. Additionally, we are going to use our own hardware devices (laptops for both developing and as a server using local servers) for this project. Therefore, we do not need a separate server so we do not have hardware cost.

The only costs we will have are for papers needed for requirement gathering and documenting the project.

Actually, the cost of the project is well within the amount of money the team can spend for the project. Therefore, it can be said that the proposed system is economically feasible to undertake.

1.5 Significance of the Project

This project has the following significances:

- It reduces the time it takes students to register.
- It allows students to register at anytime and anywhere.
- It allows the University to keep students information in a well-documented way.
- It reduces the time the University spends during registration time.
- It eliminates the intermediary (registrar workers and bank clerks) needed to register, giving students full control of the registration processes.
- It allows the University to attract more prospective university students because it offers online registration that most universities do not.
- Overall, it makes registration convenient for all students of the University.

1.6 Beneficiaries of the Project

The main beneficiaries of this project are the students and the University.

- Students
 - It helps students to register without the need for any intervention.
 - The project will reduce the time it used to take for the student to register.
 - The students will have 24/7 service to register.
- The University
 - It reduces paper (manual) work.
 - It reduces the crowd seen at the registrar during registration.
 - It reduces man-power and money used for the registration purpose.

1.7 Methodology

1.7.1 System Development Methodologies

System development methodology is used as a roadmap for any software project. It is used to structure, plan, and control the process development of a project. It is an important part of software engineering. In fact, we will be using one methodology for our project- a spiral method, which is a type of evolutionary process model.

Spiral model is an evolutionary software process model that couples the iterative nature of prototyping with the controlled and systematic aspects of the waterfall model. It provides the potential for rapid development of increasingly more complete versions of the software. We chose this methodology because it is a combination of iterative model and waterfall model. Specifically, we chose spiral methodology for the following reasons:

- It offers iterative capability, which we are going to use in future increment of the project.
- Since this model is partly based on waterfall model, it allows us to follow linear process at first iteration.
- We can add additional functionality or changes that can be done at a later stage.
- It will help us in developing the software fast.
- If any change occurs in the requirement of the project at later phases, those changes can be incorporated using this model.

1.8 Development Environment/programming tools and other tools

For our project, we will be using the following tools:

- Windows 10 is our choice of operating system on which to develop our project.
- For the front-end part of our web-based system, we will use HTML, CSS, JavaScript and bootstrap as front-end framework.
- We will use PHP as back-end language.
- For database, we will use MySQL.
- For text-editor, we will use Visual studio Code.
- For browser, we will use Google Chrome and Mozilla Firefox.
- We will use Apache as our server.

- We will be using Microsoft project and Microsoft Visio to draw different charts that are going to be used in our project.

We will use the following hardware tools for our project:

- Laptops
- USB Flash drive

1.9 Scope and Limitations of the Project

1.9.1 Scope of the Project

The scope of this project is listed below:

- Our project is only targeted for St. Mary's University Computer Science and Accounting departments.
- This system will be web based only.
- Our proposed system will allow new students and existing students to register.

1.9.2 Limitations of the Project

Some of the limitations of our project are:

- It does not allow students to add and drop for the time being.
- It will not allow students from departments other than Computer Science and Accounting to register.
- It does not include mobile apps and desktop applications.

1.10 Risks, Assumptions and Constraints

1.10.1 Risks

The risks we may face during this project are:

- We may not be able to finish the project according to our schedule because we are carrying out our classes side by side, and we have to study other courses, too.
- The requirement for the project may suddenly change if the University decides to change the way it carries the registration process, making it difficult for us to cope up with these sudden changes.
- Some team members may be unable to participate in the project because of personal issues or problems.
- If we get behind the project schedule, we may leave some features of the proposed system.
- The scope of the project might become too large and result in failure.

1.10.2 Assumptions

The assumptions that we are going to make for this project are:

- We are assuming that the requirement for this project will not change in a way that will have a considerable effect on our project.
- Another assumption we are making is that we will be able to finish the project within the schedule.
- Additionally, we are assuming that we will get the necessary information needed for our project from the Registrar Office and from students of the University.

1.10.3 Constraints

Some of the constraints of this project are:

- Since we will be carrying out the project side by side with attending class, we may have a limited time to carry out the project.
- Another constraint for this project is that our proposed system is web based and only works for St. Mary's University Department of Computer Science, and Accounting.
- We will not be able to connect our web-based system to the existing system that the University uses currently.

2. Business Area Analysis and Requirement Definition

2.1 Introduction

This chapter mainly deals with the inner working of the system used by the Registrar Office and about the requirements this project must satisfy.

To get detailed and thorough information about the existing system, we have used our own observation in addition to carrying out interviews with personnel at the Registrar Office.

2.2 Business Area Analysis

2.2.1 Business Rule Identification

Rule 1

- The person should be a student of the Computer Science or Accounting Department at St. Mary's University and should have an ID card.

Rule 2

- To register for a course, the student should have taken and passed the prerequisite for the course.

Rule 3

- Registration is carried out within a time limit specified by the Registrar Office.

Rule 4

- The number of credit hours per semester the student is registering for should not exceed that of the number set by the University.

Rule 5

- If a student did not register within the set deadline for the registration, he/she can register during the late registration period, with penalty.

Rule 6

- For new students their original documents should be verified and a copy of their original documents should be submitted.

Rule 7

- New students should have a result above the entrance point set by the Ministry of Science and Higher Education.

Rule 8

- In order to enroll, the courses the student is registering for the semester should be available.

2.2.2 Current System Vs Proposed System

Current System	Proposed System
➤ Students cannot register by themselves, which means there is the need for a registrar clerk as an intermediary for any student to register.	➤ Our proposed system will allow students to register all by themselves eliminating the need for another person in the registration process.
➤ Students can only register in-person during registration period.	✓ Since our proposed system will be an online system students will be able to register from anywhere, they are and they do not have to be present in-person.
➤ Because the registrar office only works during workdays and work hours, students cannot register outside the workdays and workhours.	✓ The system will be available for twenty-four hours and as such, students can register at any time they want within the registration period.
✓ The current system is not fully automated and therefore students' information is kept in papers.	✓ The proposed system is fully automated and because of that, all of students' information is kept in a database.

2.2.3 Players of the Existing System

Student

- Existing student
 - Takes a course payment slip
 - Completes the payment process
 - Returns the bank receipt to the registrar
- New student
 - Fills the registration forms
 - Submits the relevant documents to the registrar
 - Completes the payment process
 - Returns the bank receipt to the Registrar

Staff

- Instructor(Lecturer)
 - Uploads a student grade in to the system
- Department
 - Print the course payment slip from the system and give it to the student
 - If a student fulfills the whole payment, he/she receives one of the payment slips
 - Sets an instructor for every section

Registrar

- It checks the student's grade.
- If a student passes the previous course, it allows the student to take a new course.

- If a student fails all or one of the previous courses, it allows a student to add that course.
- Updates the student course slip into the system
- Receives a bank receipt from student
- Updates the student's Id Card
- Sends the students' attendance list to the department

2.3 Requirement Definition

2.3.1 Functional Requirements

The following are the functional requirements that the system should be able to check and verify:

- It allows new students to enter their personal information and upload necessary documentation.
- It checks the student's eligibility for a course that he/she wants to register based on prerequisite and grade.
- It stores students' information in a database.
- It allows new students to create a new account when they try to register for the first time.
- Whenever students try to login, their account is checked if the account exists or not; if it does, then it is authenticated.
- The system will show all the available courses for the current semester and allow students to choose any course that is given for that semester.
- The system will keep record of the courses that the student has or has not taken.
- The student can only see his/her own information while the system administrator can see information of all registered students.
- The system will keep the information of any new applicant/ new student in a separate table temporarily until all the documents are verified by the Registrar Office. If the documents are eligible, then all the record is transferred to another database permanently. If it is not then the record is deleted from the temporary database.
- If a student cannot register for a semester because of an academic or disciplinary suspension, then their access to their account will temporarily be revoked.
- The system will prevent students from registering for a course if:
 - They have taken the course previously.
 - They do not fulfill the prerequisite for the course.
 - The total credit hours the student registered for the semester is above what is specified by the University.

2.4 Requirement Gathering

Requirements gathering are one of the most essential parts of any project. It adds value to a project on multiple levels because it is where all the requirements for the project are gathered. It is an exploratory process that involves researching and documenting the project's exact requirements from start to finish.

In our project, to identify the requirements expected from all stakeholders, including students and registrar staff, we have used different techniques.

2.4.1 Requirement Gathering Techniques

When it comes to gathering requirements, the best way to do so differs among projects. Some requirement gathering tools and templates may work better for some projects but not for others. In fact, we have used different techniques that we think will help us in extracting as much information as we can get.

Below are the requirement gathering techniques we used in this project:

- Interview
- Brainstorming
- Observation
- Document Analysis
- Questionnaire

2.5 Method of Communication

Communication is a critical factor in project management. There are instances where projects have failed because of miscommunication and communication gaps. A project that is effectively communicated among team members will proceed smoothly and without much disagreement between members. This will eventually lead to achieving the project's goal within the deadline.

3. System Analysis and Design

3.1 Introduction

System Analysis and Design is an object-oriented approach that elaborates the overall design and structure of any system. System Analysis and Design helps in having a clear image of a system.

This chapter is about the design of the system, and it includes many structural designs in different forms such as activity diagram, use case diagram, and sequence diagram. It is the most important part of any software project because if a system is well analyzed and designed before it is implemented, then the remaining stages will be relatively easy to do. Therefore, we have spent a lot of time and effort to come up with a good design after analyzing the overall system. This chapter includes:

- Use case Diagram
- Activity Diagram
- Sequence Diagram
- Class Diagram

These diagrams and descriptions help us to map out what the system should include, how it should work, and what it will look like after it is implemented.

3.2 System Use Case

3.2.1 Actor Description

- **Existing student** is a person who has the right to register, view his or her grade and other things on the system.
- **New student** is a person who has the right to create account and register to the system.
- **Instructor** is a person who has the right to upload course result, change course result and other things on the system.
- **System admin** is a person who has the right to create account for the Registrar Officer, instructors, department staff and have the mandate to delete their account.
- **Department** is a person who has the responsibility for uploading courses.

3.2.2 Use Case Diagram Design

Use case diagram is used to show the interaction between different actors and use cases. In the above diagram, six actors interact with different use cases. All the actors except new students interact with login and change password use case; on the other hand, system admin interacts with use cases that are related to creating accounts for different users. Instructor interacts with upload course result, view course result, while students both new and existing register, and verify payment. The use cases related to registrar officers are edit calendar, approve new student and view grade of students.

3.3 Activity Diagram Design

An activity diagram visually presents a series of actions or flow of control in a system. They can also describe the steps in a use case diagram. Each activity diagram shows the actions taken to achieve a specific activity.

3.4 Sequence Diagram Design

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects. Sequence Diagrams are time focused and they show the order of the interaction visually by using the vertical axis of the diagram to represent what messages are sent and when.

Every sequence diagram we designed for our project shows the interaction between an actor and different pages of the system, and the database where all data is stored.

3.5 Class Diagram Design

A class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects. For this project, there are eight classes: System Admin, Account, Student, Department, Instructor, Department staff, Course, and registrar officers. Each of the classes has different methods.

3.6 Database Modeling

Data modeling is the process of creating a data model for the data to be stored in a database. A data model describes information in a systematic way that allows it to be stored and retrieved efficiently in a Relational Database System. The goal is to illustrate the types of data used and stored within the system, the relationships among these data types, the ways the data can be grouped and organized, and its formats and attributes.

3.6.1 Database Schema

We have the following database tables in our system that is used to store user information.

- Student
- Course
- Department
- Instructor
- Registrar officer
- System Admin
- Account

3.6.2 Normalization

Normalization is a database design technique that reduces data redundancy and eliminates undesirable characteristics like Insertion, Update and Deletion Anomalies. Normalization rules divide larger tables into smaller tables and links them using relationships. For this project, we have tried to normalize up to third normal form.

3.7. Implementation

This is the phase where actual production of the proposed project is done according to the functional requirements outlined in Chapter Two and the system design done in Chapter Three. In relation to the software process model used in this project, which is spiral model, this phase maps into the construction and deployment phase of spiral model.

Upon the start of this phase, the tools and environment needed for the implementation is prepared and ready. On the completion of this phase, the system will be fully implemented and ready to be used. Furthermore, we are assuming that the system will be installed and end users are introduced to the system and have the knowhow of the system, since we do not have the permission to access the University's server to actually install the system.

3.4.1 Technologies

Once our website is up and running, end users like students, instructors and registrar employees will want to access the website. To do this, they must provide an address to the URL (Uniform Resource Locator) in the web browser, the browser in return requests the needed file via HTTP. When the request reaches the web server, the HTTP server accepts the request, finds the requested document, and sends it back to the browser through HTTP. If the

server doesn't find the requested file, it returns a 404 response instead which indicates that the server cannot find the requested resource.

For our project, we are using Apache HTTP Server to locally host our website and all the related files. The purpose of a web server is to present the content of the website to the user's browser. These contents can be web pages or media files such as images, videos or plain text files. These files each have their own file extension, such as a file with "CSS" extension indicates that the file is cascading style sheet and "php" extension indicates that the file is a php file and such files will be processed by the web server and render the necessary html file and send it to the web browser.

Our website is a dynamic website which means that the end users can interact with it by filling different forms when there is the need to store information that the users entered. So, a database is needed, and for our project we are using MySQL database server. We will be using MySQL database to store students', instructors', and other users' data persistently. MySQL database is a popular database that works well with Apache Server.

For our project we will be using XAMPP server which consists mainly of the Apache HTTP Server, Maria DB database which is the same as MySQL database, and interpreters for scripts written in the PHP and Perl programming languages. XAMPP will help in developing our website in a fast pace since the time needed to configure the web server, database server and php engine separately is reduced significantly. Using XAMPP sever ,we are able to mimic a real live server in a local host hence making it easy to transition from local to live server during deployment phase.

3.4.2 Apache Server

Apache HTTP Server is a free and open-source web server that delivers web content through the internet. The Apache HTTP Server ("httpd") was launched in 1995 and it has been the most popular web server on the Internet since April 1996. Currently, it is on its 2.4.48 version. Its purpose is to establish a connection between a physical server and the browsers of website visitors (Firefox, Google Chrome, Safari, etc.) while delivering files back and forth between them. Apache servers process files written in different program languages such as Python, Java, and in our case PHP and others which the web browser cannot understand. Then it turns them to static HTML files and serves these files in the browser for web users.

Features of Apache Web Server:

- Handling of static files
- Loadable dynamic modules
- Compatible with IPv6
- Supports HTTP/2
- FTP connections
- Gzip compression and decompression
- Bandwidth throttling
- Compatible with different languages like php, python and Perl...etc.
- Load balancing

- Session tracking
- URL rewriting
- Geo location based on IP address

Initially Apache server was developed for UNIX but throughout the years it has become compatible with Windows operating system as well. But usually, it is used with Linux operating system to form the LAMP stack which includes Linux OS, Apache server, MySQL database, and PHP as the backend scripting language.

3.4.3 PHP

PHP (PHP: Hypertext Preprocessor) is an open-source, interpreted, and object-oriented scripting language that can be executed at the server-side which makes it well suited for web development. PHP code is processed on a web server by a PHP interpreter implemented as a module or a daemon. On a web server, the result of the interpreted PHP code which may be any type of data, such as generated HTML would form the whole or part of an HTTP response.

PHP is used to manage dynamic content, databases, and session tracking. PHP can handle forms, gather data from files, save data to a file, send data, and return data to the user. It can add, delete, modify elements within the database. It can access cookies variables and set cookies. Using PHP, we can restrict users to access some pages of your website.

Features of PHP:

- Database support
- Error reporting
- Platform independent
- Security
- Loosely typed language

Since PHP is platform independent it can work on any OS and web server.

3.4.4 Integrating Apache server and PHP

Upon a request from the user web browser, a request is sent to the web server (Apache), then this request is sent to the Apache web server through the internet. Once Apache receives the request, it analyzes the file extension of the requested file and decides whether to send the file to the PHP interpreter.

If the file has php extension, then it will be sent to the PHP interpreter. The PHP interpreter then executes the php file. During execution of this file, PHP may interact with the database to fetch information. After PHP finishes executing the file it sends the output back to Apache server. Finally, Apache receives the output from PHP through the internet and sends back the response to the web browser where the response is displayed to the computer.

3.4.5 MySQL Server Database and Connectivity

MySQL is an open-source SQL relational database management system that's developed and supported by Oracle. It is one of the most popular languages for accessing and managing the

records in the table. It is based on a client-server model. The core of MySQL is MySQL server, which handles all of the database instructions (or commands). MySQL server is available as a separate program for use in a client-server networked environment and as a library that can be embedded (or linked) into separate applications.

MySQL operates along with several utility programs which support the administration of MySQL databases. Commands are sent to MySQL Server via the MySQL client, which is installed on a computer.

MySQL runs on virtually all platforms, including Linux, UNIX and Windows and is supported by different web servers including Apache Server and IIS.

Features of MySQL Server database:

- Scalable
- Speed
- Memory efficiency
- High performance
- Secure
- Stored procedures, triggers, allows roll-back
- Multiple storage engines like InnoDB (default since MySQL version 5.5), MyISAM, Merge, Memory (heap), Federated, Archive, CSV, Black hole, NDB Cluster.

To connect to MySQL database using PHP, we can use MySQLi extension (MySQL improved) or PDO (PHP Data Objects), for our project we have used both PDO extension and MySQLi extension.

MySQLi supports both procedural and object-oriented programming.

PHP Data Objects (PDO) extension defines a lightweight, consistent interface for accessing databases in PHP. PDO_MYSQL is the driver that is used to implement the PHP Data Objects (PDO) interface to enable access from PHP to MySQL databases.

To use this driver in our project, we must enable the extension in the php.ini file.

3.4.6 Sessions

A session is a way to store information (in variables) to be used across multiple pages. By default, session variables last until the user closes the browser. Session identifiers or SID is a unique number which is used to identify every user in a session-based environment. These session IDs are randomly generated by the PHP engine. If we want to replace the system-generated session id with our own, we can supply it to the first argument of the session id function.

3.4.7 MVC

The Model-View-Controller (MVC) framework is an architectural pattern that separates an application into three main logical components. Model, View, and Controller. Each architectural component is built to handle specific development aspects of an application. MVC separates the business logic and presentation layer from each other.

- Model: It includes all the data and its related logic
- View: Present data to the user or handles user interaction
- Controller: An interface between Model and View components

In our project, we have built our own MVC framework to implement the MVC pattern.

3.4.8 Testing

Software testing is the process of evaluating and verifying to check that a software product or application does what it is supposed to do. It is Important because if there are any bugs or errors in the software, it can be identified early and can be solved before delivery of the software product. Properly tested software product ensures reliability, security and high performance which further results in time saving, cost effectiveness and customer satisfaction.

For our project, we have used continuous testing as we kept adding functionality and features. Continuous testing is testing each build as it becomes available. This will allow us to find errors early on the development stage and fix these errors in time.

In our project, we have carried out the following testing methods:

- Unit Testing
- Integration Testing
- System Testing

4. Conclusion and Recommendations

4.1 Conclusion

The current system used by the university to register students involves the student filling out a Paper based form and then paying the necessary amount of fee to a bank then return with the payment receipt to finish the process. The data that the student fills on the paper will then be encoded by registrar office workers to the existing system.

Our project's main objective was to allow students to register by themselves with minimal intervention from registrar workers. And throughout this project we have implemented this proposed system. In addition to registering students it allows instructors to enter students grade, and view grades for the course they have taught; and the department uploads the courses students must take for a specific year and semester; while registrar officers are able to approve new students, suspend students, view report for a specific year and semester. Furthermore, the system administrator is able to create accounts for employees that use the system like instructors, department officers, and registrar officers and as a report he/she will be able to see the details of the account that they have created and how many accounts they have created.

Even though we have implemented most of what we have set out to do, there still remains part of the system that is not implemented and we are leaving this as groundwork for future implementation. Additionally, we have assumed different scenarios in which we were not able to fully implement the needed logic. These constraints include connecting the payment verification with a bank system and as an assumption we have left out all the departments

except computer science assuming that they use the same logic and managing academic calendar is also left out due to shortage of time.

4.2 Recommendations

This system makes the registration process very easy for new students and accessible to qualified students. In addition, it integrates grade viewing by the student, managing students by the registrar, and uploading course result by the instructor making it easy for management and for the university's decision makers to improve their decision-making process by seeing reports generated by the system. So, we recommend that this system is implemented in real environment to allow every actor involved and the university get the full benefit of the system.

5. References

- The Apache Software Foundation. (2021, August 19). *Documentation* . Retrieved from Apache HTTP Server Project: <https://httpd.apache.org/>
- Boehm, B. (1988). A Spiral Model for Software Development and Enhancement. *Computer*, 21, 61-72.
- Bootstrap. (2021, July 25). *Documentation* . Retrieved from Bootstrap: <https://getbootstrap.com/docs/4.0/getting-started/introduction/>
- Hernandez, J. (2019, May 8). *Apache Webserver Introduction*. Retrieved from Sumo Logic: <https://www.sumologic.com/blog/apache-web-server-introduction/>
- MDN. (2021, September 20). *MVC*. Retrieved from MDN Web Docs: <https://developer.mozilla.org/en-US/>
- phptutorial. (2021, August 14). *Connecting to MySQL*. Retrieved from phptutorial: <https://www.phptutorial.net/php-pdo/pdo-connecting-to-mysql/>
- Pressman, R. S. (2010). *Software Engineering: A Practitioner's Approach* (7th ed.). New York: The McGraw-Hill Companies.
- Rob, P., & Coronel, C. (2009). *Database Systems: Design, Implementation, and Management* (8th ed.). Boston: Thompson Course Technology.
- Satzinger, J. W., Jackson, R. B., & Burd, S. D. (2012). *System Analysis and Design in a Changing World* (6th ed.). Boston: Cengage Learning.
- Sommerville, I. (2011). *Software Engineering* (9th ed.). Boston: Pearson Education.
- Tatroe, K., & MacIntyre, P. (2020). *Programming PHP* (4th ed.). Sebastopol: O'Reilly Media.
- Valacich, J. S., & George, J. F. (2017). *Modern Systems Analysis and Design* (8th ed.). Boston: Pearson Education.
- w3schools. (2021, February 23). *php mysql intro*. Retrieved from w3schools: https://www.w3schools.com/php/php_mysql_intro.asp