QAIWAN BLOG SYSTEM

(QBS)

PAIWAND HADI H. SAEED

FINAL YEAR PROJECT 1

School of Computing

Faculty of Engineering and Science

Qaiwan International University

January 2023



DECLARATION OF THESIS / UNDERGRADUATE PROJECT REPORT AND COPYRIGHT

Author’s full name : Paiwand Hadi H.Saeed

Date of Birth : 21/8/2000

Title : QAIWAN BLOG SYSTEM

Academic Session : Semester 1 2022 /2023

I declare that this thesis is classified as:

|  |  |  |
| --- | --- | --- |
|  | CONFIDENTIAL | (Contains confidential information under the Official Secret Act 1972)\* |
|  |  |  |
|  | RESTRICTED | (Contains restricted information as specified by the organization where research was done)\* |
|  |  |  |
| ✓ | OPEN ACCESS | I agree that my thesis to be published as online open access (full text) |

1. I acknowledged that Universiti Teknologi Malaysia reserves the right as follows:
2. The thesis is the property of Universiti Teknologi Malaysia
3. The Library of Universiti Teknologi Malaysia has the right to make copies for the purpose of research only.
4. The Library has the right to make copies of the thesis for academic exchange.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | Certified by: |  |
|  |  |  |  |  |
|  | SIGNATURE OF STUDENT |  | SIGNATURE OF SUPERVISOR |  |
|  | QU182SCSJ055 |  | Mrs.Shakar Hussein |  |
|  | **MATRIX NUMBER** |  | **NAME OF SUPERVISOR** |  |
|  | Date: 15 jun., 23 |  | Date: 15 jun., 23 |  |

# DECLERATION

“I hereby declare that we have read this thesis and in my opinion this thesis is sufficient in term of scope and quality for the award of the degree of Bachelor of Software Engineering.”

|  |  |  |
| --- | --- | --- |
| Signature | : |  |
| Name of Supervisor | : | Mrs.Shakar Hussein |
| Date | : | 15 February 2022 |

I declare that this thesis entitled **"QAIWAN BLOG SYSTEM" (QBS)**is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

|  |  |  |
| --- | --- | --- |
| Signature | : |  |
| Name | : | Paiwand Hadi H.Saeed |
| Date | : | 15 junuary 2023 |

# DEDICATION

This thesis is dedicated to my father, who taught me that the best kind of knowledge to have is that which is learned for its own sake. It is also dedicated to my mother, who taught me that even the largest task can be accomplished if it is done one step at a time.

# ACKNOWLEDGEMENT

In preparing this thesis, I was in contact with many people, researchers, academicians, and practitioners. They have contributed towards my understanding and thoughts. In particular, I wish to express my sincere appreciation to my main thesis supervisor, for encouragement, guidance, critics and friendship. Without their continued support and interest, this thesis would not have been the same as presented here. My fellow student should also be recognised for their support. My sincere appreciation also extends to all my colleagues and others who have provided assistance at various occasions. Their views and tips are useful indeed. Unfortunately, it is not possible to list all of them in this limited space. I am grateful to all my family member

# ABSTARC

University blogging platforms can result in management systems that are error-free, safe, dependable, and quick. It may enable the user to put less emphasis on record-keeping and more on other tasks. As a result, it will help businesses use their resources more effectively. The company can keep computerized records updated without making duplicate inputs. This implies that in order to access the knowledge, one does not need to be sidetracked by irrelevant information.

Table of Contents

[DECLERATION 3](#_Toc127002511)

[DEDICATION 4](#_Toc127002512)

[ACKNOWLEDGEMENT 5](#_Toc127002513)

[ABSTARC 6](#_Toc127002514)

[LIST OF Figure 10](#_Toc127002515)

[List of Table 10](#_Toc127002516)

[CHAPTER 1 11](#_Toc127002517)

[INTRODUCTION 11](#_Toc127002518)

[**1.1** **INTRODUCTION** 11](#_Toc127002519)

[**1.2 Problem Background** 12](#_Toc127002520)

[**1.3 Project aim** 12](#_Toc127002521)

[**1.4 Objectives** 12](#_Toc127002522)

[**1.5 Scope** 13](#_Toc127002523)

[**1.6 Importance of the project** 13](#_Toc127002524)

[**1.7 Report organization** 13](#_Toc127002525)

[Chapter 2 14](#_Toc127002526)

[LITERATURE REVIEW 14](#_Toc127002527)

[**2.1 INTRODUCTION** 14](#_Toc127002528)

[**2.2 Inter-Organization Case Study** 15](#_Toc127002529)

[**2.3 Current System Analysis** 16](#_Toc127002530)

[**2.3.1 Current System Analysis** 16](#_Toc127002531)

[**2.3.2 Proposed System** 16](#_Toc127002532)

[**2.4 Reviewing Similar System** 16](#_Toc127002533)

[**2.4.1 American University of Iraq Sulaymaniyah (AUIS) Blog system** 17](#_Toc127002534)

[**2.4.2 Suleimani Polytechnic University Blog system** 17](#_Toc127002535)

[**2.5 Compare between existing systems.** 18](#_Toc127002536)

[**2.5 Literature review on the technology used** 19](#_Toc127002537)

[**2.6 Chapter summary** 20](#_Toc127002538)

[Chapter 3 21](#_Toc127002539)

[**SYSTEM DEVELOPMENT METHODOLOGY** 21](#_Toc127002540)

[**3.1 Introduction** 21](#_Toc127002541)

[**3.2 Overview of Methodology and justification** 22](#_Toc127002542)

[**3.2.1 Planning & Analysis** 23](#_Toc127002543)

[**3.2.2 Design** 23](#_Toc127002544)

[**3.2.3 Implementation and coding** 23](#_Toc127002545)

[**3.2.4 Testing** 24](#_Toc127002546)

[**3.2.5 Deployment** 24](#_Toc127002547)

[**3.2.6 Maintenance** 24](#_Toc127002548)

[**3.3 Justification of using Iterative Development or tools to develop the system** 25](#_Toc127002549)

[**3.3.1 Visual studio** 25](#_Toc127002550)

[**3.3.2 HTML** 25](#_Toc127002551)

[**3.3.3 CSS** 25](#_Toc127002552)

[**3.3.4 Tailwind** 25](#_Toc127002553)

[**3.3.5 PHP** 26](#_Toc127002554)

[**3.3.6 MYSQL** 26](#_Toc127002555)

[**3.4 System requirement analysis:** 26](#_Toc127002556)

[**3.4.1 Hardware justification** 26](#_Toc127002557)

[**3.4.2 Software justification** 28](#_Toc127002558)

[**3.4.3 Project Planning** 28](#_Toc127002559)

[**3.5 Chapter summary** 29](#_Toc127002560)

[Chapter 4 30](#_Toc127002561)

[Requirement Analysis & Design 30](#_Toc127002562)

[**4.1 Introduction** 30](#_Toc127002563)

[**4.2 Requirement analysis** 30](#_Toc127002564)

[**4.2.1 Use case diagram** 31](#_Toc127002565)

[**4.2.2 Sequence diagram** 33](#_Toc127002566)

[**4.2.2.1 Register Sequence Diagram** 33](#_Toc127002567)

[**4.2.2.2 Login Sequence Diagram** 34](#_Toc127002568)

[**4.2.2.3 Logout Sequence Diagram** 35](#_Toc127002569)

[**4.2.2.4 Search Sequence Diagram** 36](#_Toc127002570)

[**4.2.2.5 Control Post Sequence Diagram** 37](#_Toc127002571)

[**4.2.2.6 Feedback Sequence Diagram** 38](#_Toc127002572)

[**4.2.2.7 Add Post Sequence Diagram** 39](#_Toc127002573)

[**4.2.3 Activity Diagram** 40](#_Toc127002574)

[**4.2.3.1 Control Post Activity Diagram** 40](#_Toc127002575)

[**4.2.3.2 Login Activity Diagram** 41](#_Toc127002576)

[**4.2.4 Data Dictionary** 42](#_Toc127002577)

[**4.3 Design** 45](#_Toc127002578)

[**4.4.1 ERD Design** 47](#_Toc127002579)

[**4.4.2 Normalized Table** 48](#_Toc127002580)

[**4.5 Interface Design** 49](#_Toc127002581)

[**4.6 Chapter Summary** 57](#_Toc127002582)

[Chapter 5 58](#_Toc127002583)

[Conclusion 58](#_Toc127002584)

[**5.1 Introduction** 58](#_Toc127002585)

[**5.2 Achievement of the Project** 58](#_Toc127002586)

[**5.3 suggestions for future Improvements** 58](#_Toc127002587)

[References: 59](#_Toc127002588)

[Appendix A 60](#_Toc127002589)

[SOFTWARE REQUIREMENT SPECIFICATION (SRS) 60](#_Toc127002590)

[Appendix B 88](#_Toc127002591)

[Software Design Description (SDD) 88](#_Toc127002592)

[Appendix C 97](#_Toc127002593)

[Software Testing Document (STD) 97](#_Toc127002594)

# LIST OF Figure

|  |  |  |
| --- | --- | --- |
| Figure NO. | TITLE | PAGE |
| Figure 0.1 | Use Case Diagram (QAIWAN BLOG SYSTEM) | 26 |
| Figure 0.2 | Register Sequence Diagram | 27 |
| Figure 0.3 | Login Sequence Diagram | 28 |
| Figure 0.4 | Logout Sequence Diagram | 29 |
| Figure 0.5 | Search Sequence Diagram | 30 |
| Figure 0.6 | Control Post Sequence Diagram | 31 |
| Figure 0.14 | Home Page | 43 |
| Figure 0.15 | Log in page | 43 |
| Figure 0.16 | Register page | 44 |

# List of Table

|  |  |  |
| --- | --- | --- |
| TABLE NO. | TITLE | PAGE |
| Table 1 | Hardware Minimum Requirement | 22 |
| Table 2 | Software Minimum Specification | 23 |
| Table 5 | Admin | 42 |
| Table 6 | Post | 42 |
| Table 7 | Contact | 42 |
| Table 8 | Student | 42 |

# CHAPTER 1

# INTRODUCTION

## **INTRODUCTION**

A fresh and cutting-edge method of information sharing among university students is blogging. Blogs are the fourth-generation Internet intercommunication format that differentiates from email. They are journal-style websites that consistently publish new articles over time. Blogs run by organizations, journals, or publications dedicated to disseminating scientific knowledge, as well as blogs run by individual scientists. Numerous blogs were early tweeters. They enable the distribution of not just study-related material but also of individual stories and experiences linked to higher education, especially those written by professionals' learners. This Project is about implementing an online blogging website in the university, user can use this website for knowing departments, courses, The project that its students done or having details on it, HTML, CSS and PHP are used in this project for the creation of the project. The students' personal blogs then serve as a venue for them to publish and archive blog items, whether they are written debates on various themes, answers to questions, free writing, or drafts of assignments.

## **1.2 Problem Background**

Blogs are frequently updated webpages that offer crucial insights into a certain issue. This blog system consists of distinct postings on topics that are more specialized within the blog's area of expertise, and it can make the university more known among the students, which gives a benefit for the student and the university because Qaiwan University is new to the city and also the departments like Software engineering and biomedical engineering are the one and only departments that QIU offers in the country, and the benefits for the graduated students, they can share their ideas and projects so their works will remain.

## **1.3 Project aim**

This system’s project aim is to develop an online Blog system using HTML, CSS, and PHP for Qaiwan international university. To have the proposed project of the ability to showcase the information about departments, courses, and student’s projects. The system will give all the information for the students regarding the university, for instance if a student wants to know about courses, they can find all the information they need in a detailed way, and it will also let the graduation students can get benefit based on this system, they can have contact with the new idea of their carrier.

## **1.4 Objectives**

* To inform university personnel and students of the news as soon as possible.
* To design a blog system for the University.
* To analyze an online blogging platform for university.
* To test the necessary information from the faculty and the students.

## **1.5 Scope**

University blogs, like any blog system, seem to function better when they focus on a particular topic, such as a certain department, school, or even a specific degree program. Universities can display certain works or a portfolio of students' work on course-specific blogs. In order to give international students moving from overseas more information, local university blogs can also be a useful resource. This system makes it easier for Qaiwan International University students to know more about the university’s activities and events.

## **1.6 Importance of the project**

The content on this website, which is based on university departments and activities, will be useful to students because it can:

* Provide a system that can be accessed from a mobile and PCs.
* It will give the newsiest students information and the newsiest activity inside the university.
* Help the university to be more popular.
* Get the new information earlier.

## **1.7 Report organization**

This chapter contains an overview of the system, the history of the issue, the project's purpose, its goals, its scope, and its significance. The project's concept is about the process of the system and how it will be the university's latest innovation is the creation of a blog website system that will include all of the staff and student activities as well as other university-related materials. The implementation of a literature review will be done in the following chapter, along with comparisons between this project and other projects and papers.

# Chapter 2

# LITERATURE REVIEW

## **2.1 INTRODUCTION**

In this chapter, a literature review will be implemented as a report, and in the interorganizational case study part, requirements to understand the nature and source of problems will be gathered. A survey is conducted among university students to better understand their problems, such as when new students want to choose a department and understand the courses and credits that they must take. The current system analysis involves reviewing currently available system(s) or prototype(s) and determining the system's advantages and disadvantages. There are few systems in Kurdistan that have blog systems for their universities. In comparing existing systems to ensure the strengths and weaknesses of the systems will be implemented, some research has been done to find systems about blog websites or applications that are similar to the proposed system but do not have the desired functions that we want to implement in the proposed system, so we will discuss the differences between the current systems and the proposed system. A literature review on technologies used will be written, as well as a declaration of the technologies that have been used in making the project by specifying what the technologies are, what they do, and how they implement the project.

## **2.2 Inter-Organization Case Study**

Based on a survey that was made through Blog system, Courses, Credits, Information of University, together with university students, both male and female, ages 18 and older, there are many answers to the questions that were planned to be asked were gathered through them.

While conducting the survey, it was found that most of the students have a mutual problem. The problem is that when students want to apply, they never find the complete answer to their questions. So, the blog system is the perfect system in order to get the knowledge that they are looking for about the universities departments, activities, courses, and any other information needed to fix the vagueness that the student have.

One of the main problems that is facing the students in Kurdistan region is that the majority come from a governmental background, so they are not familiar when they come to a private establishment. when it comes to UTM’s system the students that have a governmental background are not really used to many of the rules and regulations regarding the UNI. For example, the grading system is one of those obstacles, because students don’t really follow GPA system in Kurdistan. The system will also include this information in a nice, detailed way.

## **2.3 Current System Analysis**

## **2.3.1 Current System Analysis**

A blog system for university already exists in Kurdistan, it contains details about the university as well as information on its programs and activities. Students can profit from this blog system since it enables them to find everything there is to know about the department they wish to enroll in and become familiar with the university's system before they even go through the registration process.

## **2.3.2 Proposed System**

The most major benefit of the project to construct a blog system website for posting university information, and communication on a blog website for a university is improving the details for students about the information of the departments and more. In this paper, we talk about a website that helps students to gain a better understanding of the university system.

## **2.4 Reviewing Similar System**

This section will address the systems that are currently in use that have been studied or developed in the past. The discussion will contain the name of the article, a brief explanation that includes their weaknesses and strengths, and, in the conclusion, they will be compared.

## **2.4.1 American University of Iraq Sulaymaniyah (AUIS) Blog system**

This website is for their university and, it gathered the knowledge of the AUIS university. Students can get fully information just by visiting of the website. Also, they can contact the university throw the system by email, phone call and social media. And there are used three languages inside the website which is (Kurdish, Arabic and English).

## **2.4.2 Suleimani Polytechnic University Blog system**

This website is also giving the students information and knowledge them based on their departments, and you can contact the university by email, social-media, and phone call. There is one language used in this website which is English.

So, based on this existing website and the deference between my website will be the same, except that inside this website I will put the link of the Moodle and the alumni. The students will get full information of their university and aware of all knowledge of UTM system. They can also see the information of the sinners’ students and their experiences with the courses based on their departments and they can contact them directly by their profile information they share with the website. And view the activities based on the clubs of QIU university.

## **2.5 Compare between existing systems.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Features** | **AUIS Blog** | **SPU Blog** | **QIU Blog** |
| **Deployment** | website | website | website |
| **Specific Functions** | Allow users to view events.  Has environment view. | Allow users to view events.  Users can see courses | Allow users to view and apply for events.  Allow users to add posts. |
| **Technology** | Web technology | Web technology | Web technology |
| **Platform** | windows | windows | windows |
| **Connectivity** | Online | Online | Online |
| **Security features** | Normal password | Normal password | OTP Verification |
| **Friendly** | Yes | No | Yes |

Figure 2.1 Comparison between existing systems

## **2.5 Literature review on the technology used**

Coding: In layman's terms, Visual Studio Code is a code editor. Visual Studio Code is "a free editor that helps the programmer write code, helps in debugging, and corrects the code using the intelli-sense method." In normal terms, it facilitates users' ability to write the code in an easy manner. Many people say that it is half of an IDE and an editor, but the decision is up to the coders. Any program or piece of software that we see, or use works with code that runs in the background. Traditionally, coding was done in the traditional editors or even in the most basic editors, like Notepad! These editors used to provide basic support to the coders. Visual Studio will support HTML, which stands for Hypertext Markup Language and is a language for making or creating web pages. It describes the structure of a web page; it instructs the browser on how to display the contents on the website; CSS, which is used to style HTML contents in a website, specifies how the HTML contents should be displayed on the website. JavaScript is a text-based programming language used both on the client and server sides to allow you to design websites interactively. JavaScript web pages provide interactive elements that appeal to users. PHP stands for "Hypertext Preprocessor," and its scripts are executed on the server. We use this code so that we can create the website with its code. And Visual Studio is an easy-to-use tool to work with, which is why it is used in this project.

Database MySQL—is used to allow us to connect to the database through MySQL and save the system's contents into the database. And MySQL is an easy tool to use that's why it is used in this project.

A database is generally a structured collection of data; it is a location where data is saved and sorted. MySQL uses Structured Query Language, a domain-specific language, and implements a client-server model. The server responds to the client's request by getting information from the database.

## **2.6 Chapter summary**

The literature review, an inter-organizational study employing a survey, and feedback from the people we interviewed and surveyed were all discussed in this chapter. We looked at the existing system in Kurdistan and its positive and negative elements as described by the people we interviewed and questioned for our analysis of the current system. We discussed the contrasts between the articles and our proposed strategy while compared the current systems. In the literature review on the technologies used, we looked at the technologies used to produce this project and how they work.

# Chapter 3

## **SYSTEM DEVELOPMENT METHODOLOGY**

## **3.1 Introduction**

So that the project can be managed quickly and effectively, this chapter will describe the methodology kind of approach to the system development as well as how the system will be produced step-by-step and how it will be executed. Every project needs a methodology type that is best suited for the project at hand, and every methodology type has advantages and disadvantages. The phases of the methodology type will be covered in the phases of the selected methodology along with the implementation of this project.

## **3.2 Overview of Methodology and justification**

This project will employ the agile approach type since it is primarily intended to create a blog system, and the system will be created in accordance with their criteria. After that, you can change the system to a new one if you receive any feedback. A project is broken down into its many stages so that it may be handled effectively using the agile methodology. It is very necessary to continue working together with the many stakeholders and to do better on every level. Once a project has gotten underway, the teams will go through a process that includes planning, carrying out, and evaluating. Project stakeholders and team members must regularly collaborate.

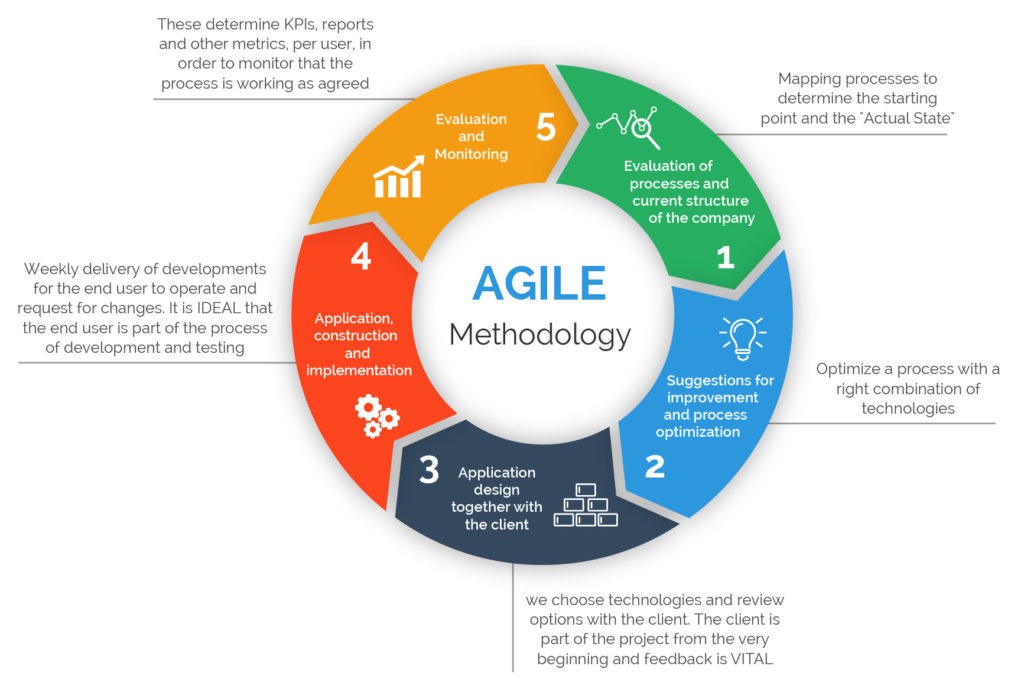


Figure ‎0‑1 Agile

### **3.2.1 Planning & Analysis**

The planning stage of the SDLC consists of two stages: the requirement gathering stage, where you collect requirements from your client or other stakeholders, and the requirement analysis stage, where you evaluate the suitability of producing the product, taking into consideration factors like revenue potential, production costs, user needs, and so on.

### **3.2.2 Design**

When you are in the design phase, you really begin to write. The initial concept and vision are expanded upon in the basic structure of the software, It includes the application security measures, the programming language, the templates, and the platform that will be used. You can also draw a flowchart illustrating how the application responds to human input.

### **3.2.3 Implementation and coding**

The programmers' labor rises during the coding stage. All elements must be put into place, and all previously designed functionality must be converted into code. If more than one developer is working on the project, which is the most frequent circumstance, then an emphasis on teamwork is also necessary because it is necessary to translate every previously specified capability into code. If more than one developer is working on the project, which is the most common scenario, a focus on teamwork is also necessary. Another important objective for publishing high-quality code is to find and repair errors and problems as soon as possible. It is beneficial to create extensive documentation as a guide to help users understand the aim and function of the program and to make the work of the developers easier.

### **3.2.4 Testing**

Before deploying the software, your quality assurance team must test it to make sure it is working properly and serving its intended purpose. Any serious security or user experience issues may be resolved with the help of the testing process. Before the product is made available to users, this stage concludes with test-driven development, or it may even begin before coding (TDD).

The types of testing to do in this phase:

* performance testing
* functional testing
* security testing
* unit-testing
* usability testing

### **3.2.5 Deployment**

First of all, you must realize that the first deployment is never simple. The application is made public and made accessible to users or customers after successful testing. The time is here for improving scenarios based on real-world happenings. Even though deployment is often automated, you and your staff should exercise caution because it is a sophisticated procedure. Compared to bigger networks, devices need to be linked often to complete this phase. More time and effort can be needed in some situations.

### **3.2.6 Maintenance**

The SDLC process's maintenance phase is perhaps its most important. After customers have had a chance to use your product in a real environment, you can use their input to add new features, fix recurring issues, and close any potential security holes. The development team's current job is to keep the existing product by keeping it current with both user preferences and technology demands.

## **3.3** **Justification of using Iterative Development or tools to develop the system**

In this section, we shall justify the technologies and tools that will be employed during the system's development phase, as well as their utilization, advantages, and potential effects on user needs.

### **3.3.1 Visual studio**

Microsoft's integrated development environment (IDE), Microsoft Visual Studio, is used to construct a range of products, including software programs, sites, web applications, online services, and smartphone apps. Visual Studio contains complementary tools, compilers, and other capabilities to help with the software development process. It supports different programming languages for coding and Visual Studio is used for this project to implement all the coding in the IDE.

### **3.3.2 HTML**

HTML, short for "Hypertext Markup Language," is a programming language used to develop websites. It specifies the structure of a website page and instructs the browser how to display the data on the website. It is used to create the website for this work and defines the structure of the website Most of the coding for the project is written in HTML.

### **3.3.3 CSS**

CSS stands for Cascading Style Sheets, which are used for designing the website, styling the HTML contents of a website, describing how the HTML contents should be shown on the website, and coloring the website contents. CSS is used for styling in this proposed project.

### **3.3.4 Tailwind**

The code of your application may be written and maintained more quickly using Tailwind CSS. You may style your application without writing bespoke CSS by utilizing this utility-first framework. To manage the padding, margin, color, font, shadow, and other aspects of your application instead, utilize utility classes.

### **3.3.5 PHP**

PHP stands for "Hypertext Preprocessor." Its scripts are executed on the server; it is used in this project so that it can connect to the database. It is used for server-side scripting.

### **3.3.6 MYSQL**

The relational database management system MySQL is free and open source. It is used so that we can connect to the database via MySQL save the data of the contents of the system into the database. And MySQL is an easy tool to use that's why it is used in this project. A database is a place in which data is stored and organized, it is simply a structured collection of data The client queries the server, and the server answers to the client from the database using the client-server architecture that MySQL implements. SQL is a domain-specific language used by MySQL.

## **3.4 System requirement analysis:**

The system requirements for this project cover both software and hardware. Hardware is any physical item, such as a device or piece of computer hardware, that is utilized to carry out many tasks, comprising calculation, processing, storage, and input and output. Software, on the other hand, is a collection of guidelines that a computer may follow to do certain tasks, such as producing code and building a system.

### **3.4.1 Hardware justification**

Hardware requirements are essential throughout project development to guarantee that the project or product will operate as efficiently as possible in any given user situation. The necessary hardware is shown in the following table:

|  |  |
| --- | --- |
| **Hardware** | **Minimum Specification** |
| **Processor** | Apple A8, Core (TM), intel(R)i5-8400U @  1.90 GHz, 1.75GHz |
| **Random Access Memory** | 4 GB |
| **Hard Drive Capacity** | 256 GB |
| **Operating System Architecture** | 64-bit. |
| **Input Device** | Mouse, Node, and Keyboard |
| **Output Device** | Screen, Monitor, etc. |

Table 1

### **3.4.2 Software justification**

The basic specifications for a platform that must be met to operate the proposed project are known as software requirements. The project's minimum requirements are shown in the table below:

Table 2 Description of the Software system requirements.

|  |  |
| --- | --- |
| **Software** | **Minimum Specification** |
| **Operating System** | **Windows 10, IOS 12** |
| **Integrated Development Environment** | **Microsoft Visual Studio.** |
| **Database Management System** | **MySQL** |
| **Web Browser** | **Safari, Google Chrome, Brave, Opera** |
| **Visual Modelling & Design Tool** | **Enterprise Architect, Lucid Chart** |
| **High Fidelity Prototype** | **Adobe XD** |

### **3.4.3 Project Planning**

Graphical user interface, table

Description automatically generated with medium confidence The project's timeframe will be scheduled using a Gantt chart for the planning-related component of the project. One well-liked depiction representing a project schedule is the Gantt chart. It resembles a bar chart in that it shows the beginning and ending dates of several project components. The Gantt chart is the most often used chart in project management. These diagrams are essential for planning projects and specifying the order in which activities must be completed. Frequently, a horizontal bar chart is used to display the chart.

Figure ‎0‑2

### **3.5 Chapter summary**

This chapter concludes with a discussion of how to use the Agile development process technique that was selected for the project. Each phase of the development process is then explained and supported by examples from the implementation of the phases in this project. Each technology or tool used in the project is justified and explained, along with its purpose. The proposed system, a website, can be viewed by mobile phones, tablets, or desktops, and the system requirements list both the software and hardware requirements.

# Chapter 4

# Requirement Analysis & Design

## **4.1 Introduction**

Requirements analysis and design requirements are covered in this chapter. which includes the use case, sequence diagrams, and activity diagrams will be described. The design part, which displays the class diagram and overall system architecture, will next be explained. The ERD database architecture, normalized tables, and eventually the website's interface and interactions will be covered after that.

## **4.2 Requirement analysis**

The user functions and the user's interaction with the system are described in the requirement analysis. Admin and students are the only two users of the system. The admin will oversee the whole system, including the pupils. When a student logs onto the system, they may see all of the information and materials related to the Qaiwan university.

## **4.2.1 Use case diagram**

The use case diagram depicts the interaction between the actors, users, head of club, and system. The requirements may be explained using case diagrams. The Qaiwan Blog System, which is my system, is the subject of this use case. It depicts the dialogue between the administrator and the student.

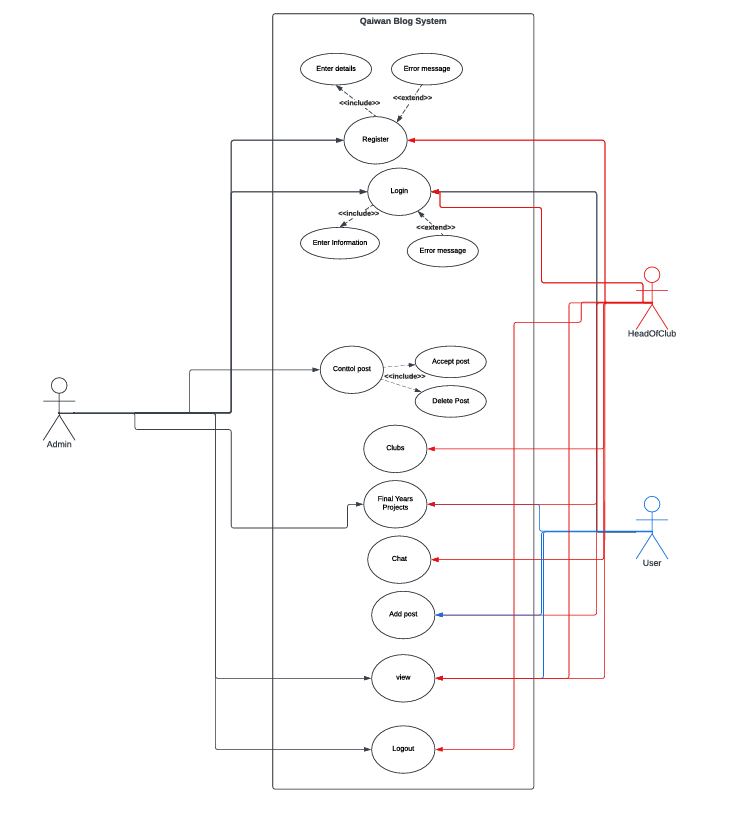


Figure ‎0‑1 Use Case Diagram (Qaiwan Blog System)

The student is permitted to register and login to the website according to the use case design in Figure 3. The student may view university content, post in the system, see events, departments, and activities after logging in, as well as provide system feedback. Students may also view profiles of each other. The head of the club can add the activities inside the blog system and they will control. The student can use chat inside the club and view the events in the clubs. The administrator will be in charge of the system; the registration is just for the administrator, and the administrator will offer the student an account.

## **4.2.2 Sequence diagram**

Sequence diagrams in UML are a type of interaction diagram that are used to depict the stages that are required to finish an operation. They show the ways in which the many components of a partnership interact with one another. In order to graphically show the progression of the interaction, sequence diagrams center their attention on time and utilize the vertical axis of the diagram to represent time in the form of the messages that are delivered and when they are sent.

### **4.2.2.1 Register Sequence Diagram**

In this figure 4, in diagram below is about Register sequence diagram based on my blog system, How it works with this system and the data base for admin.

Diagram

Description automatically generated

Figure ‎0‑2 Register Sequence Diagram

### **4.2.2.2 Login Sequence Diagram**

In this figure 5, it’s about login sequence diagram, and the connection of the system with the system and database for both admin and user. Diagram

Description automatically generated

Figure ‎0‑3 Login Sequence Diagram

### **4.2.2.3 Logout Sequence Diagram**

This figure 6 is Logout sequence diagram.

Diagram

Description automatically generated

Figure ‎0‑4 Logout Sequence Diagram

### **4.2.2.4 Search Sequence Diagram**

In figure 7, is about the search sequence diagram in this system and the connections between the system and database, user can search inside the website by the name that related on the website.

Diagram

Description automatically generated

Figure ‎0‑5 Search Sequence Diagram

### **4.2.2.5 Control Post Sequence Diagram**

In Figure 8, the admin will be handling the posts according to their kind. There are two options for the admin: one is to approve the post, and the other is to remove the post, taking into account all of the system and database ideas.Diagram

Description automatically generated

Figure ‎0‑6 Control Post Sequence Diagram

### **4.2.2.6 Feedback Sequence Diagram**

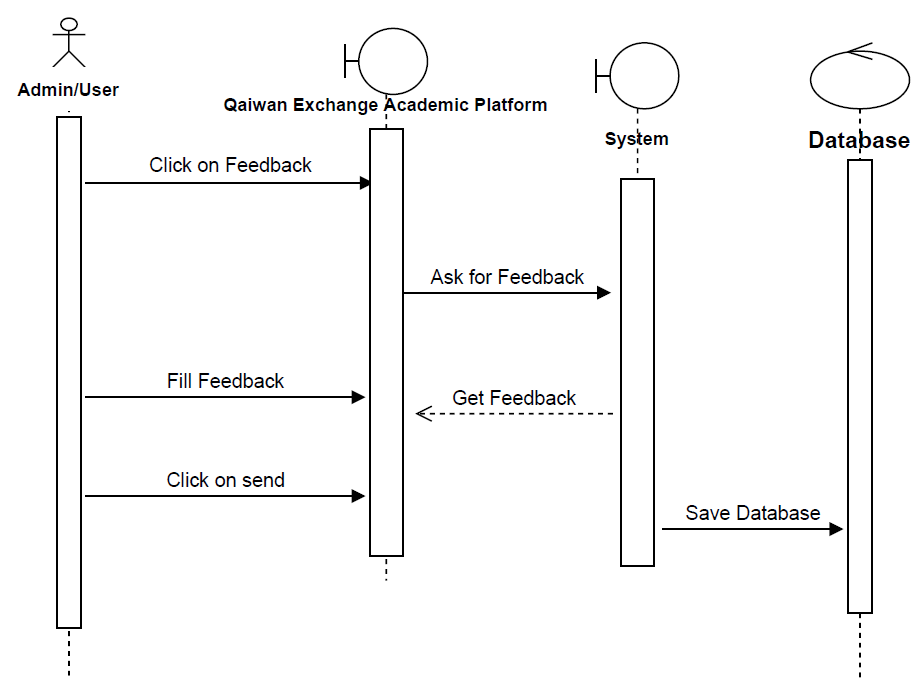
The feedback sequence diagram can be seen in Figure 9. This figure shows how the user provides feedback to the system, after which it is sent to the database before being sent back to the system admin.

Figure ‎0‑7 Feedback Sequence Diagram

### **4.2.2.7 Add Post Sequence Diagram**

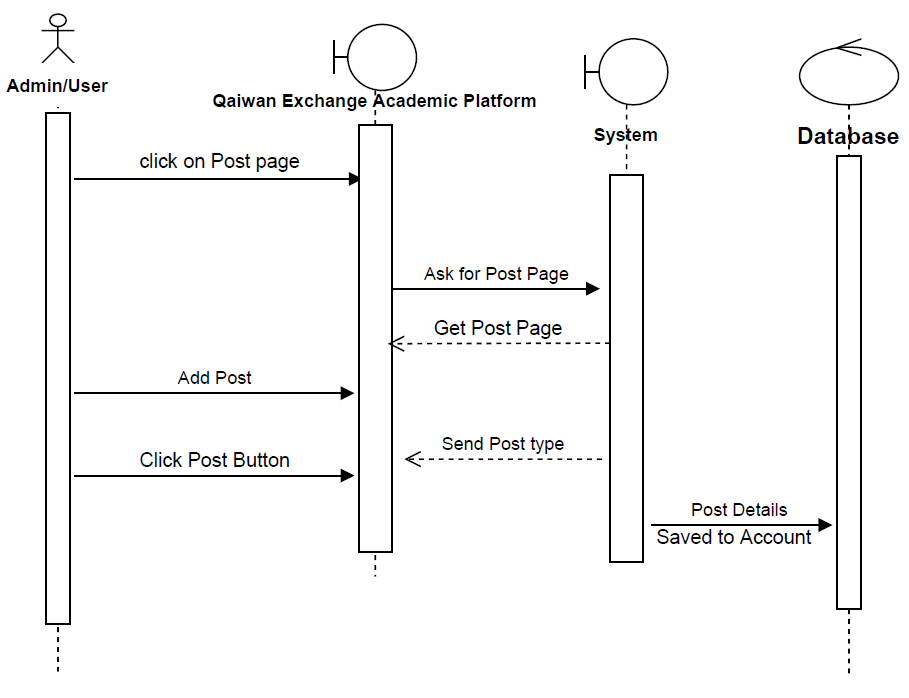
The add-post sequence diagram may be seen here in picture number 10. This website allows users as well as admins to submit content.****

Figure ‎0‑8 Add Post Sequence Diagram

### **4.2.2.8 Chat Sequence Diagram**

In this figure 9, showing the chat sequence diagram that students can use chat inside the system.

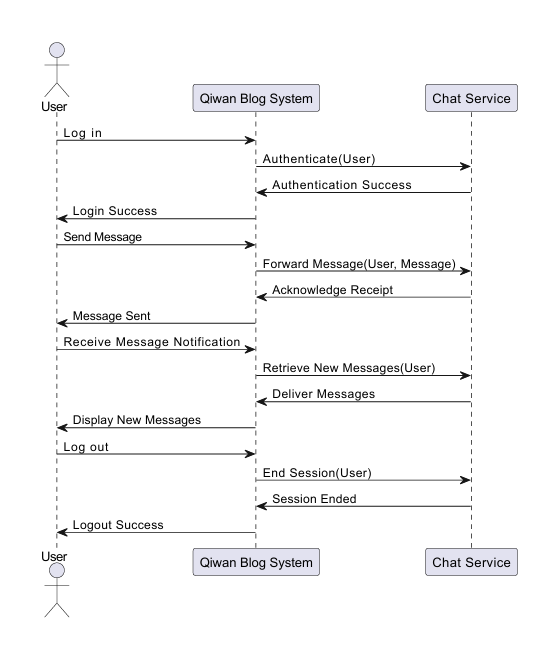


Figure 0-9 chat Sequence Diagram

## **4.2.3 Activity Diagram**

When attempting to represent the dynamic characteristics of a system using a UML diagram, activity diagrams are another key behavioral diagram to include. Modeling the transition from one activity to another is the primary purpose of an activity diagram, which is basically an advanced form of a flow chart.

### **4.2.3.1 Control Post Activity Diagram**

Figure 11 shows that admins always start at the home site and make their way to the control post. After that, administrators have the ability to modify the postings, at which point they may either remove them or approve them; the post will be stored if all went according to plan. If this is not the case, an error notice will be shown.

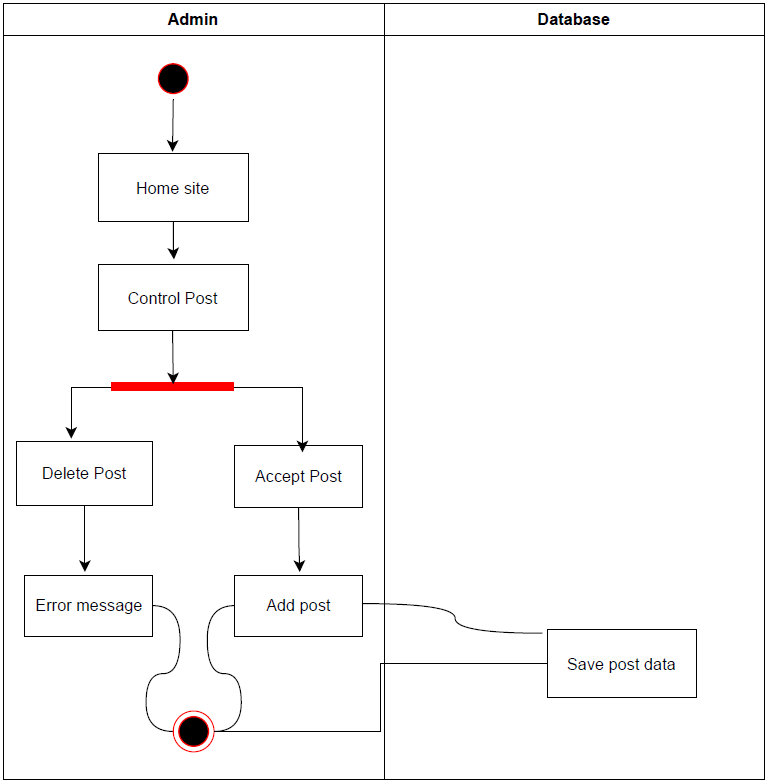


Figure ‎0‑9 Control-Post Activity Diagram

## **4.2.3.2 Login Activity Diagram**

According to this figure 12, the admin has already registered the system. When the user first logs in to the system, there are two conditions that must be met. If the first condition is met, the login information will be saved to the database, and the user may then log in to the system. If this is not the case, an error message will be sent.

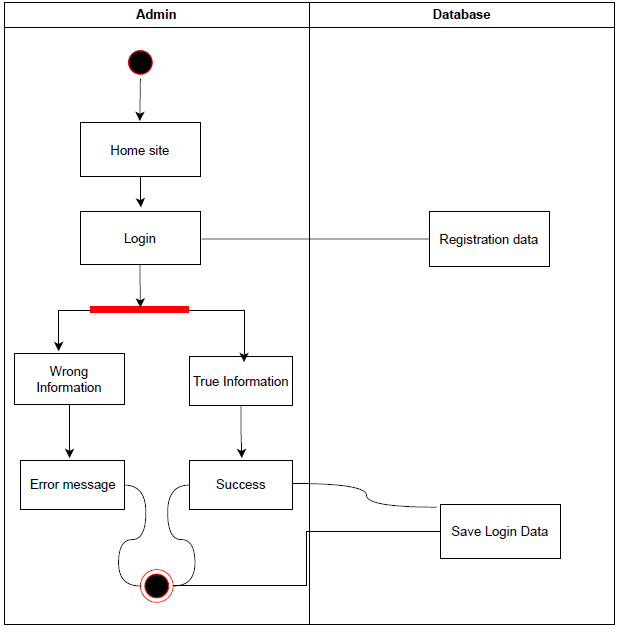


Figure ‎0‑10 Login Activity Diagram

## **4.2.4 Data Dictionary**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Table** | **Attribute** | **Type** | **Length** | **PK/FK** | **NULL** |
| ADMIN | A\_ID | INT | 30 | PK |  |
|  | A\_EMAIL | VARCHAR | 350 |  |  |
|  | A\_CODE | INT | 30 |  |  |
|  |  |  |  |  |  |
| STUDENT | S\_ID | INT | 30 | PK |  |
|  | S\_EMAIL | VARCHAR | 350 |  |  |
|  | S\_CODE | INT | 30 |  |  |
|  |  |  |  |  |  |
| Search | SR\_ID | INT | 30 |  |  |
|  | SR\_INFO | VARCHAR | 350 |  |  |
|  |  |  |  |  |  |
| CONTROL  POST | P\_ID | INT | 30 | FK |  |
|  | P\_INFO | VARCHAR | 350 |  |  |
|  | P\_TYPE | VARCHAR | 350 |  |  |
|  | P\_DATE | DATE |  |  |  |
|  |  |  |  |  |  |
| FEEDBACK | F\_ID | INT | 30 |  |  |
|  | F\_INFO | VARCHAR | 350 |  |  |
|  |  |  |  |  |  |
| PROFILE | P\_ID | INT | 30 | PK |  |
| P\_INFO | VARCHAR | 350 |  |  |  |

Table 3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Table** | **Attribute** | **Type** | **Length** | **PK/FK** | **NULL** |
| ADMIN | A\_ID | INT | 30 | PK |  |
|  | A\_EMAIL | VARCHAR | 350 |  |  |
|  | A\_CODE | INT | 30 |  |  |
|  |  |  |  |  |  |
| STUDENT | S\_ID | INT | 30 | PK |  |
|  | S\_EMAIL | VARCHAR | 350 |  |  |
|  | S\_CODE | INT | 30 |  |  |
|  |  |  |  |  |  |
| Search | SR\_ID | INT | 30 |  |  |
|  | SR\_INFO | VARCHAR | 350 |  |  |
|  |  |  |  |  |  |
| CONTROL  POST | P\_ID | INT | 30 | FK |  |
|  | P\_INFO | VARCHAR | 350 |  |  |
|  | P\_TYPE | VARCHAR | 350 |  |  |
|  | P\_DATE | DATE |  |  |  |
|  |  |  |  |  |  |
| FEEDBACK | F\_ID | INT | 30 |  |  |
|  | F\_INFO | VARCHAR | 350 |  |  |
|  |  |  |  |  |  |
| PROFILE | P\_ID | INT | 30 | PK |  |
| P\_INFO | VARCHAR | 350 |  |  |  |

Table 4

## **4.3 Design**

**4.3.1 Class Diagram**

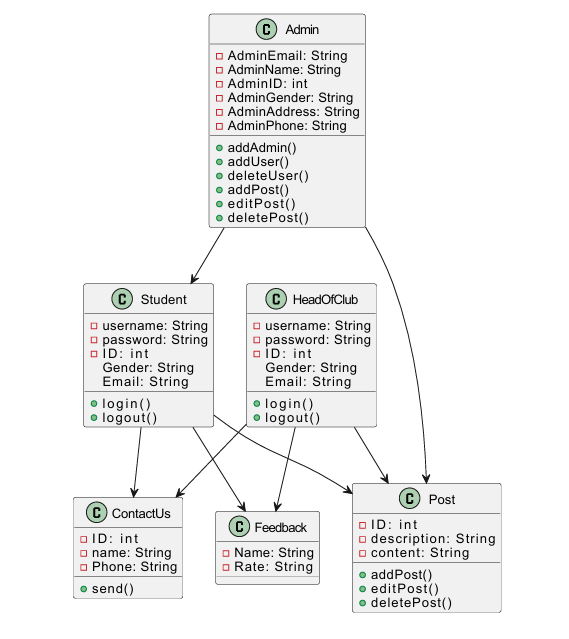
In order to facilitate the construction of object-oriented systems as well as their visualization, a graphical notation that is known as the UML Class Diagram is utilized. The Unified Modeling Language (UML) contains something called a class diagram, which may be thought of as a static structural diagram. The purpose of this kind of diagram is to describe the structure of a system by exhibiting the system's components in the picture that can be found further down on this page: 

Figure ‎0‑11 Class Diagram

**4.3.1 System Architecture**

The phrase "client-server architecture" describes a type of computer system that not only satisfies the majority of the requests for resources and services that are made by clients, but also hosts, delivers, and maintains the majority of those resources and services. This model of computing is sometimes referred to as the networking computing model or the client-server network. It may be separated from other paradigms by the fact that all requests and services are transmitted across a network in order to be processed.

A client-server architecture, also known as a client-server model, is a type of network application that divides responsibilities and workloads between clients and servers that are either co-located on the same computer or are linked to one another through the implementation of a computer network. This kind of application is generally referred to as a client-server model.****

Figure ‎0‑12 (System Architecture Design)

**4.4 Database Design**

### **4.4.1 ERD Design**

The classes that are contained inside the system, as well as the connections that link the classes to one another, are represented by class diagrams. Class diagrams contain not only the class itself, but also its attributes and methods, as well as the relationships that link the three of them together. These relationships may include inheritance, aggregation, composition, and a great number of others; however, these three are the most important ones. Figure 0-13 depicts a class diagram for the project that is being proposed.**Diagram

Description automatically generated**

Figure ‎0‑13 System ERD diagram

### **4.4.2 Normalized Table**

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Type** | **PK/FK** |
| ID | Int | PK |
| Email | Text |  |
| name | Varchar |  |
| Phone | Text |  |
| Gender | Text |  |
| Address | Text |  |

Table 5 (Admin)

|  |  |  |
| --- | --- | --- |
| Attribute | Type | PK/FK |
| ID | int | PK |
| Description | Text |  |
| Content | Text |  |

Table 6 (Post)

|  |  |  |
| --- | --- | --- |
| Attribute | Type | PK/FK |
| ID | int | PK |
| Name | Varchar |  |
| Phone | Type |  |

Table 7 (Contact)

|  |  |  |
| --- | --- | --- |
| Attribute | Type | PK/FK |
| ID | Int | PK |
| username | Varchar |  |
| Password | Text |  |
| Gender | Text |  |
| Email | Text |  |

Table 8 (Student)

## **4.5 Interface Design**

These figures below are the design of the (Qaiwan Blog System):

Graphical user interface

Description automatically generated

Figure ‎0‑14 (Home Page)

Graphical user interface

Description automatically generated

Figure ‎0‑15 (Login Page)

Graphical user interface, application

Description automatically generated

Figure ‎0‑16 (Register Page)

Graphical user interface, website

Description automatically generated

Figure ‎0‑17 (Dashboard)

Graphical user interface, application, website

Description automatically generated

Figure ‎0‑18 (View Project)

Graphical user interface, application

Description automatically generated

Figure ‎0‑19 (View Activity)

Graphical user interface, application

Description automatically generated

Figure ‎0‑20 (View Event)

Graphical user interface, website

Description automatically generated

Figure ‎0‑21 (Control Post)

Graphical user interface

Description automatically generated

Figure ‎0‑22 (Activity)

Graphical user interface

Description automatically generated

Figure ‎0‑23 (Project)

Graphical user interface

Description automatically generated

Figure ‎0‑24 (Event)

Graphical user interface, website

Description automatically generated

Figure ‎0‑25 (Register for clubs)

Graphical user interface, application

Description automatically generated

Figure ‎0‑26 (Contact us)

Graphical user interface, text

Description automatically generated

Figure ‎0‑27 (Departments)

Table

Description automatically generated

Figure ‎0‑28 (Feedback)

## **4.6 Chapter Summary**

The development of schematics and plans for the system was the primary focus of this chapter's discussion of important subjects. Such design for class diagrams, ERD, use cases, design for databases, etc. These diagrams are going to be put to use in the system implementation for the subsequent tasks (FYP2).

# Chapter 5

# IMPLEMENTATION AND TESTING

## **5.1 Introduction**

This chapter outlines the implementation and testing of the Qaiwan Blog System (QBS). The primary functionalities of the system, including registration, login, adding posts, and activity management, are implemented using HTML, CSS, PHP, and MySQL. The system's interfaces are designed to be user-friendly and efficient. Each section below details the coding and interfaces of the system's main functions.

## **5.2 Coding of System Main Function**

## **5.2.1 Register**

The registration function allows new users to create an account in the Qaiwan Blog System. The code ensures that all necessary information is collected and stored securely in the database.

## **5.2.2 Login**

The login function verifies user credentials and grants access to the system if the provided information matches the records in the database.

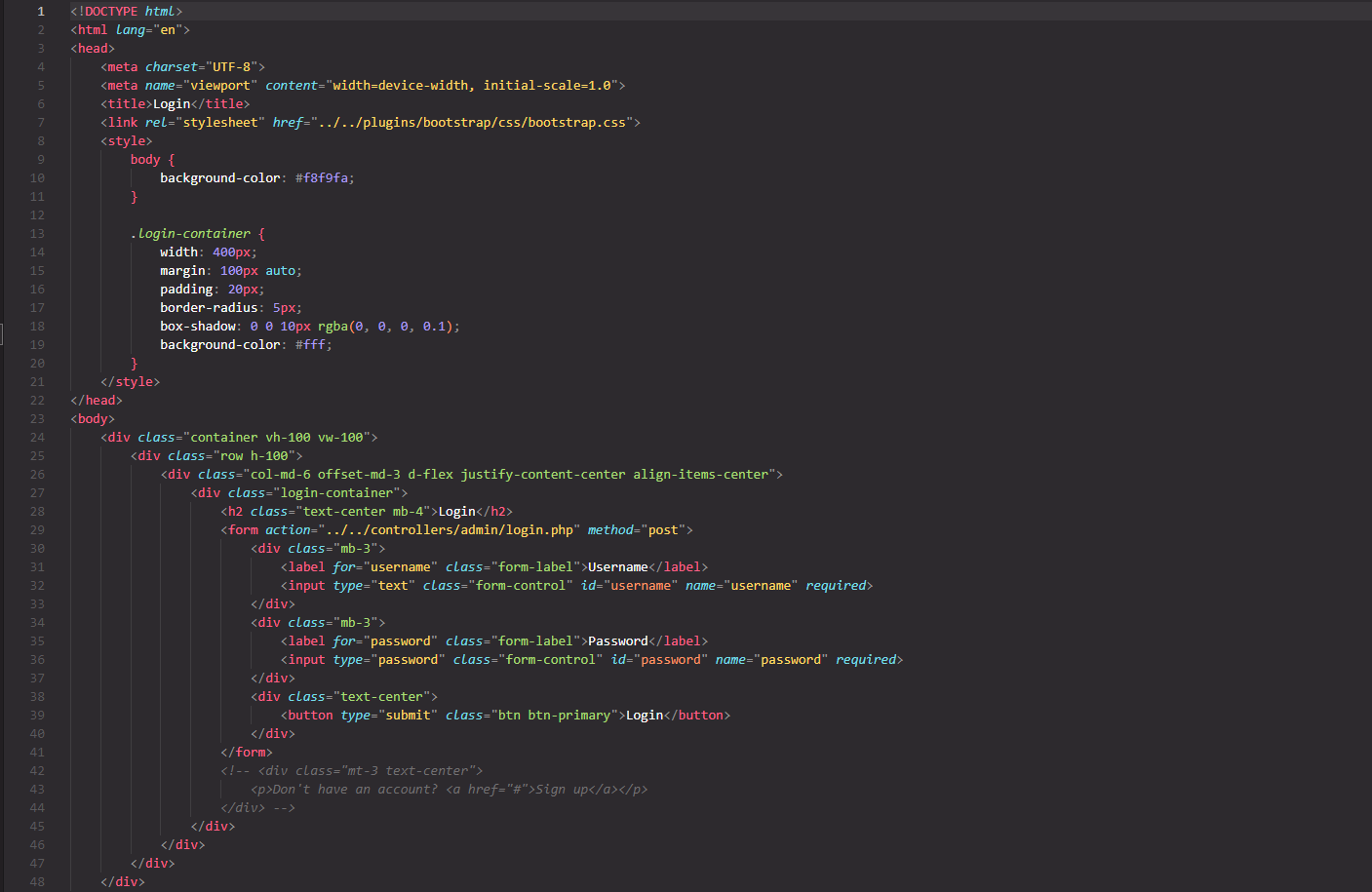


Figure 29

## **5.2.3 Add Post**

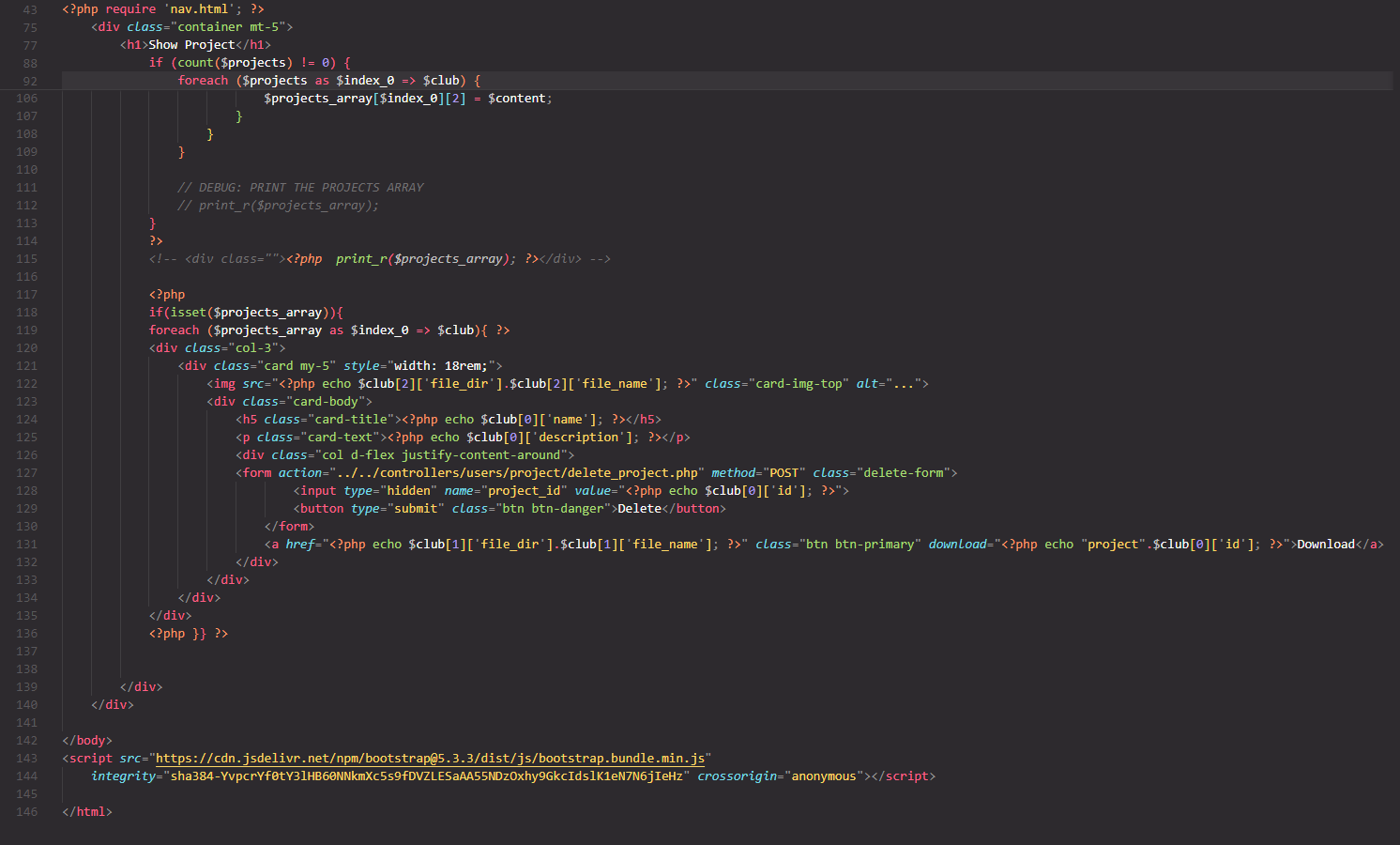
The add post function allows head of clubs to create new blog posts. The code ensures that the post content is saved to the database.

Figure 30

## **5.2.4 Activity**

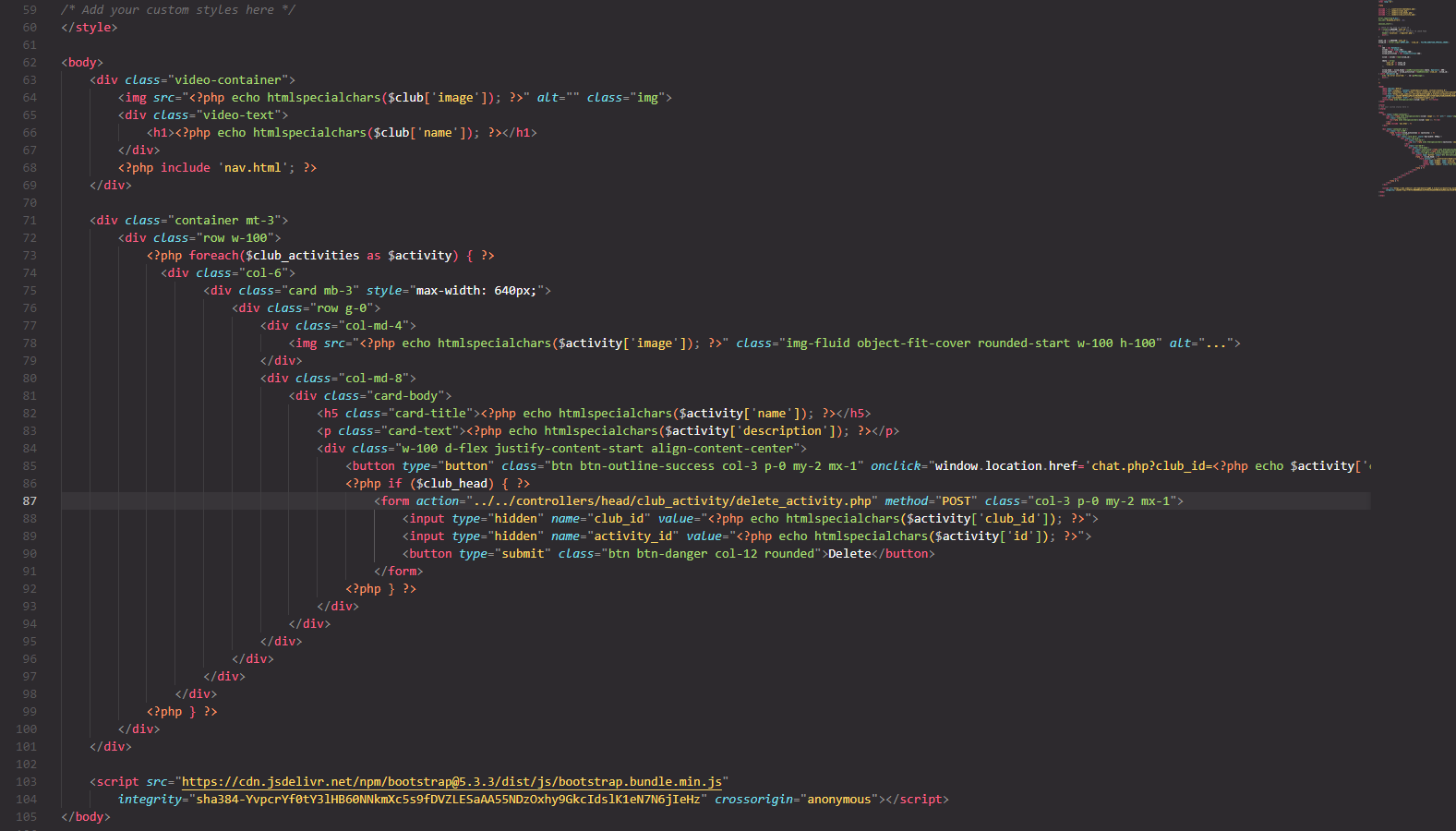
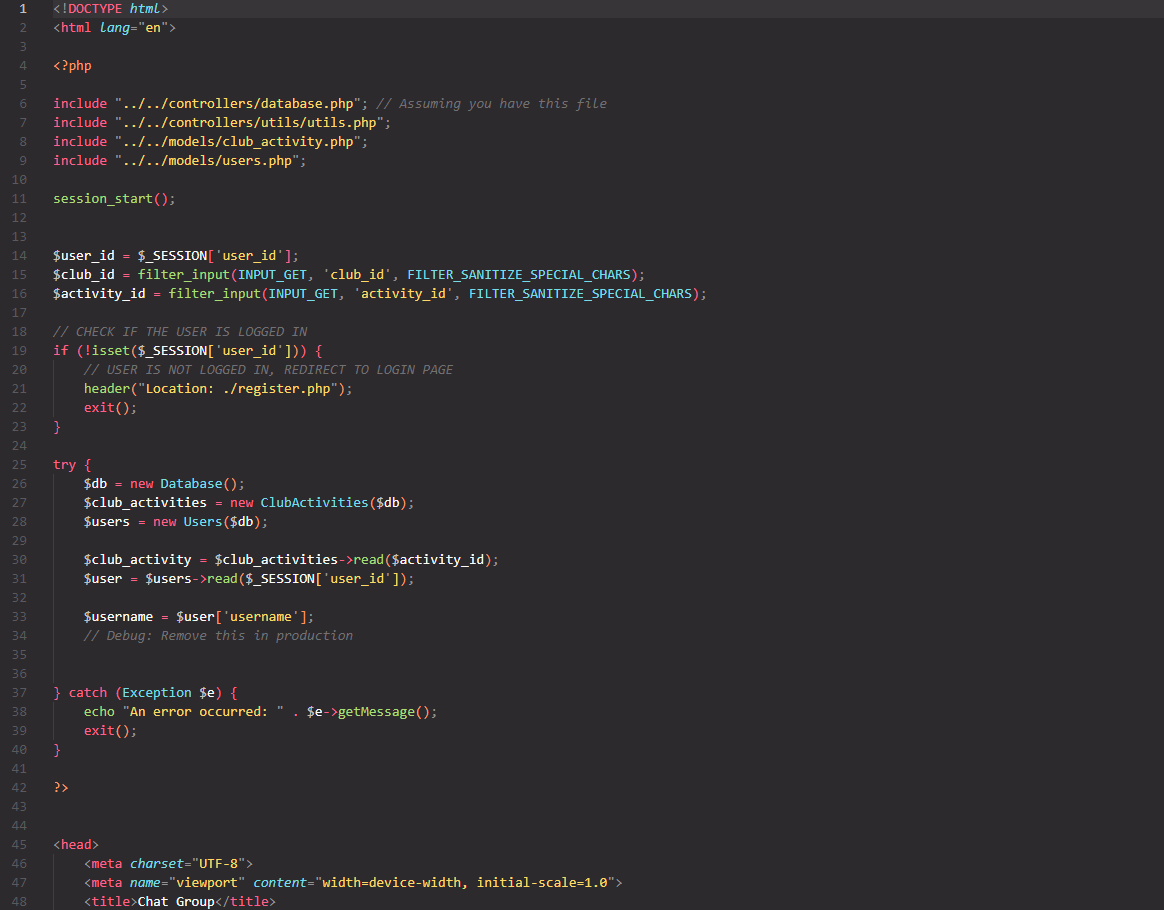
The activity function logs head of clubs activities such as posting, chatting, and logging in, which are then displayed on the user's activity feed.

Figure 31

## **5.2.5 Chat**

The chat interface allows users to communicate in real-time. This interface displays a chat window where users can send and receive messages.

## **5.3 Interfaces of System Main Function**

The user interfaces for the Qaiwan Blog System (QBS) are designed to be intuitive and user-friendly. The main functions of the system—registration, login, adding posts, and viewing activities—each have their own dedicated interface. Below are the detailed descriptions and HTML code for these interfaces.

## **5.4 Testing**

The website is now complete and functional, but in order to have a decent website, you must test it to see if the input you provide results in the intended output or not.

## **5.4.1 Black Box Testing**

Black boxing is a method for evaluating the performance of the system and testing your program. In this method, the tester chooses a function, provides inputs, and tests the function's output to determine if it succeeded or failed.

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** | **Status** |
| Username, password, email Valid | Successful Register, Showing the Login page | Successful Register, Showing the Login page | pass |
| username, password and email, Invalid | unsuccessful register and showing error message | unsuccessful register and showing error message | pass |
| Password match | Successful Register, Showing the Login page | Successful Register, Showing the Login page | pass |
| Password does not match | unsuccessful register and showing error message | unsuccessful register and showing error message | pass |
| Left out required Field | User should fill out the fields | User should fill out the fields | pass |

Table 9 Black box testing (Register Page)

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** | **Status** |
| username and password valid | Successful to login  Showing the home page | Successful to login  Showing the home page | pass |
| Invalid username and password | unsuccessful login and showing error message | unsuccessful login and showing error message | pass |
| Left out required Field | User should fill out the fields | User should fill out the fields | pass |

Table 10 Black box testing (Login page)

**5.4.2 White Box Testing**

White-box testing is a kind of testing that looks at how a system works on the inside. This kind of testing looks at how much a code statement, branch, path, or condition is used. White-box testing is a term for testing at a low level.

|  |  |
| --- | --- |
| **Use Case Name** | Login |
| **Use Case ID** | UC01 |
| **Description** | The use case talks about how the admin and user Login to the website. |
| **Pre-Condition** | The user must have an account on the website. |
| **Date** | 11- Feb - 2023 |
| **Tester:** | Paiwand Hadi |

Table 11 White box testing <login Page>

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| username and password valid | Session created and redirected to Homepage page | Session created and redirected to Homepage page |
| wrong username or password | Display error message | Display error message |

Table 12 White box testing <login Page>

**2.1.3 User Testing**

User testing is the process of putting a website or app's features and user interface to the test by having real people do certain tasks in real life. For testing, the users are regular people who are testing the Qaiwan Blog System website.

|  |  |  |
| --- | --- | --- |
| Tester: Paiwand Hadi | | |
| Date: 11/2/2023 | | |
| Module: Contact us | | |
| Instruction | Expected Result | Result |
| * 1. Click on Contact us page.   2. Fill out the fields.   3. Click Send Button | 1. Show the waiting icon. 2. Show successful message | pass |

Table 13 User Testing <Contact Us>

|  |  |  |
| --- | --- | --- |
| Tester: Paiwand Hadi | | |
| Date: 11/2/2023 | | |
| Module: Search | | |
| Instruction | Expected Result | Result |
| * 1. Click on a search bar.   2. Search by default   3. Search by writing a specific name.   4. Click on it | * 1. Showing same items   2. Going the direct page that will be searched | pass |

Table 14 User Testing <Search>

|  |  |  |
| --- | --- | --- |
| Tester: Paiwand Hadi | | |
| Date: 11/2/2023 | | |
| Module: Apply | | |
| Instruction | Expected Result | Result |
| * 1. View activities   2. Join clubs.   3. Fill the fields.   4. Click on apply | Successfully joining the club | pass |

Figure 15 User Testing <Apply>

# Chapter 6

# Conclusion

## **6.1 Introduction**

The overall results of the Qaiwan Blog System will be the subject of the primary discussion in this chapter's major topic. This system will be available for our university (Qaiwan International University).

## **6.2 Achievement of the Project**

The success that is attained as a result of the requirements, the literature review, the resources based on the blog system, and the chances for growth that it provides to both the teaching staff and the students and by contrasting those systems with this one, we have identified the problems that exist inside it, which will help us learn more about it and attempt to solve them.

## **6.3 suggestions for future Improvements**

There are a lot of different aspects of it that I want to learn more about, and one of those aspects, which will be covered throughout the next semester, is the practical consequences of the website. And if it were up to me, I'd want to take this website to the next level by adding features like the ability to pay for books at our university.

# References:

1. “What Is Agile Methodology in Project Management?” *What Is Agile Methodology in Project Management?*, [www.wrike.com/project-management-guide/faq/what-is-agile-methodology-in-project-management. Accessed 14 Jan. 2023](http://www.wrike.com/project-management-guide/faq/what-is-agile-methodology-in-project-management.%20Accessed%2014%20Jan.%202023).
2. “https://codilime.com/blog/the-stages-of-the-sdlc/.” *https://codilime.com/blog/the-stages-of-the-sdlc/. Accessed 14 Jan. 2023.* Accessed 14 Jan. 2023.
3. “The 7 Stages of the Software Development Life Cycle - CodiLime.” *CodiLime*, 18 Aug. 2022, codilime.com/blog/the-stages-of-the-sdlc.
4. “What Is Sequence Diagram?” *What Is Sequence Diagram?*, www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram. Accessed 14 Jan. 2023.
5. Bandakkanavar, Ravi. “Software Requirements Specification Document With Example - Krazytech.” *Krazytech*, 10 June 2021, krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database.
6. “What Is Activity Diagram?” *What Is Activity Diagram?*, www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-activity-diagram. Accessed 14 Jan. 2023.
7. “6 Reasons to Include a Blog on Your University Website.” *6 Reasons to Include a Blog on Your University Website*, 17 Feb. 2020, campuspress.com/6-reasons-to-include-a-blog-on-your-university-website.
8. “How to Start Blogging at University.” *Top Universities*, www.topuniversities.com/blog/how-start-blogging-university. Accessed 14 Jan. 2023.
9. “UML Class Diagram Tutorial.” *UML Class Diagram Tutorial*, www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-class-diagram-tutorial. Accessed 14 Jan. 2023.
10. “Entity Relationship Diagram (ERD) - What Is an ER Diagram?” *Entity Relationship Diagram (ERD) - What Is an ER Diagram?*, www.smartdraw.com/entity-relationship-diagram/#whatIsERD. Accessed 14 Jan. 2023.
11. “A Guide to Dependencies, Constraints and Assumptions (Part 3): Project Assumptions - InLoox.” *A Guide to Dependencies, Constraints and Assumptions (Part 3): Project Assumptions - InLoox*, www.inloox.com/company/blog/articles/a-guide-to-dependencies-constraints-and-assumptions-part-3-project-assumptions. Accessed 15 Jan. 2023.

# Appendix A

# SOFTWARE REQUIREMENT SPECIFICATION (SRS)

**1. Introduction**

**1.1 Purpose**

The purpose of this SRS (Software Requirement Specification) was to outline the exact requirements for the system that was developed. It is a document that lists the features and requirements for the system's performance.

**1.2 Scope**

The university community is the target audience for this project, which intends to assist people stay informed about everything that occurs on campus, including changes to the campus landscape, new club activities, and other things.

**1.3 Definitions, Acronyms and Abbreviation**

QBS: Qaiwan Blog System

UC: Use Case

SRS: Software Requirement Specification

**1.4 Overview**

The following chapter of this publication, titled "Overall Description," provides an overview of the productive viewpoints, system features, user characteristics, and constraints. User interfaces, hardware interfaces, and software interfaces are all covered in detail in Chapter Three, Specific Requirements. The use cases, use case requirements, and activity diagram are all discussed in the system features.

**2. Overall Description**

This whole document is about the problem that facing in Kurdistan. The information from a product standpoint is in the SRS that follows. With the user characteristics allowed limits and assumptions, it offers all information and details regarding Qaiwan Blog System.

Diagram

Description automatically generated

2.1 1 Use case

**2.1 Product Perspective**

Because the QAEP is a web-based tool, a browser like Google Chrome, Safari, etc. is necessary. The MySQL database server is linked to the website. The system retrieves the data from the database and displays it on the webpage.

**In this system has 2 users:**

* **Admin**
* **Student**

**2.1.1 System Interfaces**

**2.1.2 User Interfaces**

Graphical user interface

Description automatically generated

2.1 2 : Login Page

Graphical user interface

Description automatically generated

2.1 3 Home page

Graphical user interface, website

Description automatically generated

2.1 4 Dashboard

Graphical user interface, text

Description automatically generated

2.1 5 Department

Graphical user interface, application

Description automatically generated

2.1 6 Contact Us

**2.1.3 Hardware Interfaces**

The website requires an internet connection, either through cellular or Connected by LAN services. To use internet browsers, the system needs a device that can connect to the internet.

**2.1.4 Software Interfaces**

Because PHP is the greatest option for us to connect our website with the database because we are using MySQL, we have used HMTL, CSS, PHP, and java script because they are more interactive and support our website.

Database: We kept all of our information as well as the admin's and students' records in the MySQL data base.

**2.1.5 Communication Interfaces**

This system will supper all types of web browser.

**2.1.6 Memory**

The primary memories should have at least 2GB of RAM, while secondary memories should have at least 32GB and ideally 64GB.**2.2 Product Functions**

The use case for the system and how it will communicate with one another based on this blog system are described in the following statement.

**2.2.1 Register Use Case**

Diagram

Description automatically generated

2.1 7 Register Use case

Description: Admin can sign up and add another admin or student to the system.

**2.2.2 Login Use Case**

Diagram

Description automatically generated

2.1 8 Login Use case

Description: Both admins and students can login in, but administrators must first created student accounts.

**2.2.4 Search Use Case**

Diagram

Description automatically generated

2.1 9 Search Use case

Description: Student can search in the search form by the names that related to the system.

**2.2.5 Feedback**

Diagram

Description automatically generated

2.1 10 Feedback Use case

Description: Student can give feedback for the events and the activities, Admin can check the feedbacks.

**2.2.6 Contact us Use Case**

Diagram

Description automatically generated

2.1 11 Contact us Use case

Description: Student will be able to contact the system by contacting with the admin.

**2.2.7 Control Post Use Case**

Diagram

Description automatically generated

2.1 12 Control Post Use case

Description: Admin will control the posts, by accepting and deleting the posts.

**2.2.8 Logout Use Case**

Diagram

Description automatically generated

2.1 13 Logout Use case

Description: The logout for both users in the system is the same.

**2.3 User Characteristics**

**2.4 Constraints**

Most laptops and desktop computers that support HTML, CSS, PHP, and Java Script should be able to use the system.

User Interface: Navigating the website is far too easy and natural. A user-friendly interface with simple navigation should allow users to utilize all the system's features.

Security constraints: There is a good effort made to ensure the security of this system; students cannot register themselves. Each student needs to have their own login for the website.

**2.5 Assumption and Dependencies**

1. Availability of key project team members: All key project team members are available and equipped to work on the project.
2. Availability of the budget: The estimated budget is complete and precise.
3. Accurate scheduling: The project may be completed on time and the deadlines and milestones are attainable.
4. Performance of vendors, suppliers, and contractors: You can always get the tools and supplies you need.

**2.6 Apportioning of Requirements**

If the servers are down or the database is not connected, users from other universities won't be able to access the system.

**3. Specific Requirements**

Diagram

Description automatically generated

2.1 14 <Qaiwan Blog System>

**3.1 System Features**

**3.1.1 Module <user module>**

**3.1.1.1 UC01: Use Case <Register>**

Diagram

Description automatically generated

Figure 3.1 1 Use case Diagram <Register >

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-01 | | |
| **Use Case Name:** | Register | | |
| **Created By:** | Paiwand Hadi | **Last Updated By:** | Paiwand Hadi |
| **Date Created:** | 14/1/2023 | **Last Revision Date:** | 15/1/2023 |
| **Actors:** | Admin | | |
| **Description:** | The use case talks about how the admin register to the website. | | |
| **Pre-conditions:** | 1. The admin must have the access to the internet. 2. The admin must access the website. | | |
| **Flow of events:** | 1. The admin opens the website in the internet browser. 2. The admin clicks on the Register button. 3. The admin will fill up the form and they will send the registration form. 4. Then the click on register button. 5. The system will send the inputs to the data base. 6. If registration is successful, the admin will be informed. 7. The admin receives notification that the registration has failed if the register is not completed. | | |
| **Exception:** | 1. If the email exists, the system gives an error that the email already in use. | | |

Table 3.1 1 Use Case Description <Register>

Diagram

Description automatically generated

Table 3.1 2 System Sequence Diagram <Register>

**3.1.1.2 UC02: Use Case <Login>**

Diagram

Description automatically generated

Table 3.1 3 System Sequence Diagram <Login>

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-02 | | |
| **Use Case Name:** | Login | | |
| **Created By:** | Paiwand Hadi | **Last Updated By:** | Paiwand Hadi |
| **Date Created:** | 14/1/2023 | **Last Revision Date:** | 15/1/2023 |
| **Actors:** | Admin, Student | | |
| **Description:** | The use case talks about how the admin and user Login to the website. | | |
| **Pre-conditions:** | 1. The admin and user must have the access to the internet.   1. The admin and user must access the website. 2. The student must get register by the admin. | | |
| **Flow of events:** | * 1. The admin and user open the website in the internet browser.   2. The admin and user Enter the Username and password.   3. The system validates the username and the password.   4. The username and password will be verified by the system. The system will show the website's main page if the login and password are valid. The system will show an error notice if the username or password are wrong. | | |
| **Exception:** | 1. The home page will change based on the admin login or user. | | |

Table 3.1 4 Use Case Description <Login>

Diagram

Description automatically generated

Table 3.1 5 System Sequence Diagram <Login>

**3.1.1.3 UC03: Use Case <Search>**

Diagram

Description automatically generated

Figure 3.1 2 Use Case Diagram <Search>

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-03 | | |
| **Use Case Name:** | Search | | |
| **Created By:** | Paiwand Hadi | **Last Updated By:** | Paiwand Hadi |
| **Date Created:** | 14/1/2023 | **Last Revision Date:** | 15/1/2023 |
| **Actors:** | Student | | |
| **Description:** | The use case talks about how the users Search in the website. | | |
| **Pre-conditions:** | 1. The user must have the access to the internet. 2. The user must access the website. 3. The User must be logged in to the website | | |
| **Flow of events:** | 1. the users must login to the website. 2. The users will click on the search in the website Navbar. 3. The users will write down the name. 4. The system will show the all the information by the name. | | |
| **Exception:** | 1. The system will not show any parts that the searched by the user inside the website. | | |

Table 3.1 6 Use Case Description <Search>

Diagram

Description automatically generated

Figure 3.1 3 System Sequence Diagram <Search>

**3.1.1.4 UC04: Use Case < Feedback>**

Diagram

Description automatically generated

Figure 3.1 4 Use Case Diagram <Feedback>

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-04 | | |
| **Use Case Name:** | Feedback | | |
| **Created By:** | Paiwand Hadi | **Last Updated By:** | Paiwand Hadi |
| **Date Created:** | 14/1/2023 | **Last Revision Date:** | 15/1/2023 |
| **Actors:** | Admin, Student | | |
| **Description:** | The use case indicates the section where feedback can be given by the student, and admin will check the feedbacks. | | |
| **Pre-conditions:** | 1. The admin and user must have the access to the internet. 2. The admin and user must access the website. 3. The admin and user must be logged in to the website | | |
| **Flow of events:** | 1. the admin and user must login to the website. 2. The student will click on the activities or events and get their feedbacks. 3. The admin will check the feedbacks that given by the students. | | |
| **Exception:** | N/A. | | |

Table 3.1 7 Use Case Discerption <Feedback>

**3.1.1.5 UC05: Use Case <Contact>**

Diagram

Description automatically generated

Figure 3.1 5 Use Case Diagram <Contact>

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-05 | | |
| **Use Case Name:** | Contact us | | |
| **Created By:** | Paiwand Hadi | **Last Updated By:** | Paiwand Hadi |
| **Date Created:** | 14/1/2023 | **Last Revision Date:** | 15/1/2023 |
| **Actors:** | Student | | |
| **Description:** | The use case talks about how the users can contact the admins throw out the website. | | |
| **Pre-conditions:** | 1. The user must have the access to the internet. 2. The user must access the website. 3. The User must be logged in to the website 4. The user clicks on contact page 5. The user enter the required text | | |
| **Flow of events:** | 1. The users must login to the website. 2. The users will click on the contact us page. 3. The user will be able to see a text to fill in. | | |
| **Exception:** | 1. the system will not show any error is the process was successful. | | |

Table 3.1 8 Use Case Description <Contact us>

**3.1.1.5 UC05: Use Case <Control Post>**

Diagram

Description automatically generated

Figure 3.1 6 Use Case Diagram <Control Post>

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-05 | | |
| **Use Case Name:** | Control post | | |
| **Created By:** | Paiwand Hadi | **Last Updated By:** | Paiwand Hadi |
| **Date Created:** | 14/1/2023 | **Last Revision Date:** | 15/1/2023 |
| **Actors:** | Admin | | |
| **Description:** | The use case talks about how the admin can control the users post. | | |
| **Pre-conditions:** | 1. The admin must have the access to the internet. 2. The admin must access the website. 3. The admin must be logged in to the website 4. The admin clicks on control post 5. The admin chooses between accept or reject post. | | |
| **Flow of events:** | 1. The admin must login to the website. 2. The admin will click on control post to be redirected to the page. 3. The admin will see two options to choose between. | | |
| **Exception:** | 1. the system will not show any error is the process was successful. | | |

Table 3.1 9 Use Case Discerption <Control Post>

Diagram

Description automatically generated

Figure 3.1 7 System Sequence Diagram <Control Post>

Diagram, schematic

Description automatically generated

Figure 3.1 8 Use Case Diagram <View>

Diagram

Description automatically generated

Figure 3.1 9 Use Case Diagram <Apply>

**3.2 Performance Requirements**

Considering that the system would be web-based. Both internet and high-speed internet connection are necessary.

**3.3 Design Constraints**

To create a website, the system makes use of many pieces of software. A list of the system's software is provided below:

* The system has a user-friendly interface because to the integration of HTML, CSS, and Tailwind.
* The data is stored in a MySQL database, which was created for the system.

**3.4 Software System Attributes**

Availability: The system will be accessible, and all operations will be completed without interruption from the internet.

Usability: The system is simple to use and navigate in the manner that is often followed without delay.

Using MySQL services, the system makes sure that user information is secure in the database.

# Appendix B

# Software Design Description (SDD)

**Introduction**

**1.1 Purpose**

The architecture and intricate design of the Qaiwan Blog System website are described in this SDD Purpose. The SDD is introduced in part one of this publication, the system architectural design is shown in part two, and the comprehensive description is presented in part three. This paper serves as a comprehensive design guide for a piece of software. It provides details on a product's abilities and the creator's goals.

**1.1 Purpose**

This SDD outlines the first part, which includes the introduction, scope, definition of acronyms, references, and overview; part two, which consists of the architectural model for the system being designed; and part three, which consists of the entity relationship diagram, data dictionary, and user interface design for the database being designed.

**1.3 Definitions, Acronyms and Abbreviation**

SDD: Software Design Description

**1.4 Overview**

This article discusses the architecture of the system using system analysis. The system's data design, which describes the type of data model used in this system and contains the entity relationship diagram (ERD) and data dictionary, is also included in this document. and the primary function interfaces of the system, which allow for a clearer view of the website.

1. **System Architectural Design** 
   1. **Architecture Model**

****

Figure 3.1 10 System Architecture

1. **Database Design**
   1. **Entity Relationship Diagram (ERD)**

**Diagram

Description automatically generated**

Figure 3.1 11 ERD Diagram (Qaiwan Blog System)

**3.2 Data Dictionary**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Table** | **Attribute** | **Type** | **Length** | **PK/FK** | **NULL** |
| ADMIN | A\_ID | INT | 30 | PK |  |
|  | A\_EMAIL | VARCHAR | 350 |  |  |
|  | A\_CODE | INT | 30 |  |  |
|  |  |  |  |  |  |
| STUDENT | S\_ID | INT | 30 | PK |  |
|  | S\_EMAIL | VARCHAR | 350 |  |  |
|  | S\_CODE | INT | 30 |  |  |
|  |  |  |  |  |  |
| Search | SR\_ID | INT | 30 |  |  |
|  | SR\_INFO | VARCHAR | 350 |  |  |
|  |  |  |  |  |  |
| CONTROL  POST | P\_ID | INT | 30 | FK |  |
|  | P\_INFO | VARCHAR | 350 |  |  |
|  | P\_TYPE | VARCHAR | 350 |  |  |
|  | P\_DATE | DATE |  |  |  |
|  |  |  |  |  |  |
| FEEDBACK | F\_ID | INT | 30 |  |  |
|  | F\_INFO | VARCHAR | 350 |  |  |
|  |  |  |  |  |  |
| PROFILE | P\_ID | INT | 30 | PK |  |
| P\_INFO | VARCHAR | 350 |  |  |  |

Table 3.1 10 Data dictionary

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Table** | **Attribute** | **Type** | **Length** | **PK/FK** | **NULL** |
| ADMIN | A\_ID | INT | 30 | PK |  |
|  | A\_EMAIL | VARCHAR | 350 |  |  |
|  | A\_CODE | INT | 30 |  |  |
|  |  |  |  |  |  |
| STUDENT | S\_ID | INT | 30 | PK |  |
|  | S\_EMAIL | VARCHAR | 350 |  |  |
|  | S\_CODE | INT | 30 |  |  |
|  |  |  |  |  |  |
| Search | SR\_ID | INT | 30 |  |  |
|  | SR\_INFO | VARCHAR | 350 |  |  |
|  |  |  |  |  |  |
| CONTROL  POST | P\_ID | INT | 30 | FK |  |
|  | P\_INFO | VARCHAR | 350 |  |  |
|  | P\_TYPE | VARCHAR | 350 |  |  |
|  | P\_DATE | DATE |  |  |  |
|  |  |  |  |  |  |
| FEEDBACK | F\_ID | INT | 30 |  |  |
|  | F\_INFO | VARCHAR | 350 |  |  |
|  |  |  |  |  |  |
| PROFILE | P\_ID | INT | 30 | PK |  |
| P\_INFO | VARCHAR | 350 |  |  |  |

Table 3.1 11 Data dictionary

**3.3 User Interface Design**

Graphical user interface

Description automatically generated

Figure 3.1 12 Login page (Admin)

Graphical user interface

Description automatically generated

Table 3.1 12 Home page

Graphical user interface, website

Description automatically generated

Table 3.1 13 Dashboard

Graphical user interface, text

Description automatically generated

Table 3.1 14 Department

Graphical user interface, application

Description automatically generated

Table 3.1 15 Contact Us

# Appendix C

# Software Testing Document (STD)

**1. Introduction**

**1.1 Purpose**

Each stage of the development of a software product, especially a website, is crucial, but system testing is extremely crucial. Before launching the website, it is crucial to test it to ensure that everything functions perfectly. We tested this website using three techniques during development: user acceptability testing, white box testing, and black box testing.

* 1. **Scope**

This STD is about testing the system, and the system will be testing by the Qaiwan universities students.

* 1. **Definitions, Acronyms and Abbreviation**

STD: Software Testing Document

* 1. **Overview**

The website's STD is introduced in this paper, and in the second section, we cover the three testing techniques that were utilized to test the website: user acceptability testing, white box testing, and black box testing.

**2. Test Cases, Data and Expected Results**

**2.1 Testing**

The website is now complete and functional, but in order to have a decent website, you must test it to see if the input you provide results in the intended output or not.

**2.1.1 Black Box Testing**

Black boxing is a method for evaluating the performance of the system and testing your program. In this method, the tester chooses a function, provides inputs, and tests the function's output to determine if it succeeded or failed.

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** | **Status** |
| Username, password, email Valid | Successful Register, Showing the Login page | Successful Register, Showing the Login page | pass |
| username, password and email, Invalid | unsuccessful register and showing error message | unsuccessful register and showing error message | pass |
| Password match | Successful Register, Showing the Login page | Successful Register, Showing the Login page | pass |
| Password does not match | unsuccessful register and showing error message | unsuccessful register and showing error message | pass |
| Left out required Field | User should fill out the fields | User should fill out the fields | pass |

Table 3.1 16 Black box testing (Register Page)

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** | **Status** |
| username and password valid | Successful to login  Showing the home page | Successful to login  Showing the home page | pass |
| Invalid username and password | unsuccessful login and showing error message | unsuccessful login and showing error message | pass |
| Left out required Field | User should fill out the fields | User should fill out the fields | pass |

Table 3.1 17 Black box testing (Login page)

### 

**2.1.2 White Box Testing**

White-box testing is a kind of testing that looks at how a system works on the inside. This kind of testing looks at how much a code statement, branch, path, or condition is used. White-box testing is a term for testing at a low level.

|  |  |
| --- | --- |
| **Use Case Name** | Login |
| **Use Case ID** | UC01 |
| **Description** | The use case talks about how the admin and user Login to the website. |
| **Pre-Condition** | The user must have an account on the website. |
| **Date** | 11- Feb - 2023 |
| **Tester:** | Paiwand Hadi |

Table 3.1 18 White box testing <login Page>

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Result** | **Actual Result** |
| username and password valid | Session created and redirected to Homepage page | Session created and redirected to Homepage page |
| wrong username or password | Display error message | Display error message |

Table 3.1 19 White box testing <login Page>

**2.1.3 User Testing**

User testing is the process of putting a website or app's features and user interface to the test by having real people do certain tasks in real life. For testing, the users are regular people who are testing the Qaiwan Blog System website.

|  |  |  |
| --- | --- | --- |
| Tester: Paiwand Hadi | | |
| Date: 11/2/2023 | | |
| Module: Contact us | | |
| Instruction | Expected Result | Result |
| * 1. Click on Contact us page.   2. Fill out the fields.   3. Click Send Button | 1. Show the waiting icon. 2. Show successful message | pass |

Table 3.1 20 User Testing <Contact Us>

|  |  |  |
| --- | --- | --- |
| Tester: Paiwand Hadi | | |
| Date: 11/2/2023 | | |
| Module: Search | | |
| Instruction | Expected Result | Result |
| * 1. Click on a search bar.   2. Search by default   3. Search by writing a specific name.   4. Click on it | * 1. Showing same items   2. Going the direct page that will be searched | pass |

Table 3.1 21 User Testing <Search>

|  |  |  |
| --- | --- | --- |
| Tester: Paiwand Hadi | | |
| Date: 11/2/2023 | | |
| Module: Apply | | |
| Instruction | Expected Result | Result |
| * 1. View activities   2. Join clubs.   3. Fill the fields.   4. Click on apply | Successfully joining the club | pass |

Figure 3.1 22 User Testing <Apply>