International University, VNU-HCM

School of Computer Science and Engineering

Software Engineering

CONTENT MANAGEMENT SYSTEM FOR A BLOG

Final Report

Group 3:

Pham Hong Dang	ITITIU19009
Le Quy Minh Duc	ITITIU20188
Nguyen Vo Thanh Tam	ITITWE19012
Dang Quang Vinh	ITITIU19247
Vo Trung Duong	ITITIU20193

Advisor: Assoc. Prof. Nguyen Thi Thuy Loan

Content Management System for a Blog

Contributor

Pham Hong Dang	Le Quy Minh Duc	Nguyen Vo Thanh Tam	Dang Quang Vinh	Vo Trung Duong
20%	20%	20%	20%	20%

Table 1: Contribution

GitHub repo: https://github.com/vinhkute24/blog_website

Table of Contents

List of Figures

List of Tables

I. PROJECT DESCRIPTION

1.1. Abstracts

Abstract - Several different types of publishers are responsible for the vast majority of the content found on the World Wide Web. Most of this data is not edited or peer-reviewed by specialists like books and journals. Finding reliable information online is more difficult than ever because of the proliferation of websites and the general absence of quality control on the web. Meanwhile, new web page production facilities like Blogs amplify this problem by providing non-experts with basic content management tools that allow them to create easily updatable web diaries or online journals. However, there has not yet been developed a content management system (CMS) for measuring information quality on Blogs, despite a decade of intense research into IQ. This paper introduces a new content management system (CMS) for evaluating Blog posts according to quality. The CMS has the necessary features for classifying Blogs in accordance with IQ standards. In the created CMS, Blogs are automatically collected, and their IQs are automatically calculated. Keywords: Information Quality, Weblog, Content Management System.

1.2. Project Overview

The WWW is becoming one of the most popular online resources for gathering data. Meanwhile, new web page production tools like Blogs have significantly boosted the rate. Due to Blogs' intuitive content management systems, even laypeople can create and maintain their own dynamic online diaries or journals with ease. Technorati, a blog search engine, was tracking over 70 million blogs as of May 2007. According to the results of a quarterly survey, there are 120,000 new blogs published every day, with 1.5 million entries added. Blogs have risen to prominence on the World Wide Web and have become a commonplace medium for disseminating information online. Such commonly shared information is referred to as "little contents" by Ohmukai. Each day, there are more and more blog posts with very few words and numerous citations. Topic identification, trend analysis, and content ranking are just some of the methods used on these massive data sets. A "Blog" (sometimes spelled "web log" or "Blog") is a type of website that features frequent updates to previously published articles on a certain topic, usually in reverse chronological order. The content may have been produced by the site's creator, culled from external websites, or collected through user contributions. Articles on blogs cover a wide range of subjects, from personal experiences to technological advances to political debates to global issues. Reading blog posts is a wonderful way to get honest, up-to-date thoughts on a wide range of topics, from technology to poetry to social issues. After a decade of study and implementation, the only tools at our disposal for gauging, assessing, and bettering the quality of information on the web are piecemeal, ad hoc approaches. The intelligence level of Blogs is not currently being evaluated by any established criteria. Since most blogs make use of the same basic framework, we reasoned that it would be a useful tool for judging the credibility of online sources. Owners of blogs are not typically experts in IT. Blog content is prioritized over concerns about layout and design, and they make use of premade templates to do so. Our study's ultimate goal is to create a Blog-specific CMS. To define and quantify information quality, it is not sufficient to simply identify the shared features of IQ frameworks as independent entities. In reality, the contexts of information development and intended use are required for an accurate evaluation of information quality. This is because the characteristics of high-quality information change depending on the intended application.

1.3. Project Goals

The project's ultimate goal was to construct a content management system that accounted for quality standards. As was indicated previously, a user interface was created after the system was developed and incorporated the chosen criteria; we tracked user activity during the beta phase. Our research demonstrated that the implemented system successfully collected data, which we posted to the Blog. Our original intent was for blog owners to be able to check their own blog's performance. It is clear from the comments that people were able to see the ratings as well. The current CMS's strength is in its experimental structure, which ranks and ranks Blogs in a real-world setting. The current system can automatically rank users' intelligence on Blogs. In order to rank the Blogs, the system designed for managing them automatically calculates and accumulates values for eighteen variables. Automatic (and therefore mostly noise-free) data gathering and score calculation were achieved. Nevertheless, nine subjective criteria were obtained by voting due to their significance. The current method's primary benefit is the abundance of data available through the CMS's system log files. Insights gleaned from the dataset will guide the direction of future investigations. Using data analysis and data mining methods like cluster analysis, we will develop a system for ranking blog posts. The unique characteristics of Blogs and the criteria of social networks lead us to believe that this model will shed light on previously unexplored aspects of evaluating Blog material.

1.4. Scope of Work

Numerous technologies are used in this project to manage, plan, store, and develop network applications, deploy websites, as well as a platform for developing, distributing, and running applications. Users will thus experience a more realistic rather than just perusing the demo of the development team.

Included

Self-study knowledge is also used to enhance the project in addition to the application of the specialized knowledge that was learned for it. This project currently only uses the website to implement all protocols and activities; it does not use the phone application.

There is one type of account. First, the user’s account will be allowed to be able to add textual or media content into placeholders that are attached to locations on the web page using drag and drop method. Users should be able to design the entire website in this way. Additionally, users must have the option to publish blog posts.

Excluded

There are still a lot of features in this project that we have not yet been able to finish. For instance, some common features like notifications number of people who visit other’s blogs to send directly to user's accounts

1.5. Organization

1.5.1. Organizational Boundaries and Interfaces

By setting up daily meetings based on the demands of the moment, the project management will oversee both the division of labor and the communication between each team member. These primary tasks will be managed by the task leaders, who will delegate the subtasks to other team members. One person in this project has the option of leading this task and participating in another

task. An individual will complete the subtask, and the task leaders and project management can review the results later.

Resource Owners

Resource owners have a steering group through the project sponsor, the customer/user project team by management. Receive requests for competencies and resources.

Receivers

Receivers are the individuals who accept responsibility for ongoing performance from suppliers, such as operation and maintenance, and who familiarize themselves with performance outcomes and corporate goals. Additionally, they outline and communicate their demands to the supplier management on the list of unresolved problems.

1.5.2. Project Organization

Task	Name
Project Manager	Đặng Quang Vinh
Business Analyst	Võ Trùng Dương
Designer	Nguyễn Võ Thanh Tâm
Software Engineer	Lê Quý Minh Đức
Software Engineer	Phạm Hồng Đăng

Table 2: Project Organization

1.6. Resource Requirements

1.6.1. Hardware Resources:

Item	Quantity	Description	Purpose	Price	Total Cost	Data needed
Laptop Dell Precision M4700	1	OS: Windows 10 Home CPU: Intel Core i7-3940XM Graphics: Intel Iris Xe + NVIDIA Quadro K2000M RAM: 16GB Onboard Storage: 256 GB SSD Screen: Full HD 15.6"	Data Server	$599.99	$599.99	Done
Dell XPS 9360	1	OS: window 11 CPU: Intel Core i7- 7600U Graphics: Intel HD graphics 620 Ram: 16 Gb onboard Storage: 512Gb SSD	Data Server	$672	$672	Done
Laptop Dell Inspiron 3493	1	OS: Window 10 Home CPU: Intel Core i5 – 1035G1 Graphics: Intel UHD RAM: 8GB	Data Server	$549.99	$549.99	Done

		Storage: 200GB				
MacBook Pro 2017	1	OS: macOS CPU: Intel Core i5-8230U RAM: 8 GB 2133 MHz LPDDR3 Storage: 256GB SSD Screen: 13,3" Retina	Data Server	$458.00	$458.00	Done
MacBook Pro 2017	1	OS: macOS CPU: Intel Core i5-8230U RAM: 8 GB 2133 MHz LPDDR3 Storage: 256GB SSD Screen: 13,3" Retina	Data Server	$458.00	$458.00	Done
Total					$2737.98	

Table 3: Hardware Resources

1.6.2. Software Resources

Application	Quantity	Description	Purpose	Price	Total cost	Date needed
Visual Studio Code	5	A system development environment that is integrated. Among the functions that are available are debug, syntax highlighting, completion of the intelligent code, fragments, code refactoring, and built-in git.	Implement code	Free	Free	Done
Microsoft Teams	5	A collaboration application	Meeting, reminding, and storing files	Free	Free	Done
NodeJS	5	An executing JS code server-side platform	Building an extensible network application	Free	Free	Done

Git	5	Website for sharing and uploading code	Managing code	Free	Free	Done
Hostinger VPS Server Plan 3	1	Cloud hosting server	Host project on the cloud server	$7.99/month	$383.52	Done
Figma	5	Wireframe drawing tool, UI design collaboration application	Handles file organization by displaying projects and their files in a dedicated view. Create a project for the feature theme. Create a file for an epic or large feature.	Free	Free	Done
Total:				$383.52		

Table 4: Software Resources

1.6.3. Human Resources

Name	Name of Organization	Role	Required skills	Phone	Email	Salary/ month
Võ Trùng Dương	Hitachi Ltd – Inspired the next.	Business Analyst	Technical skills: Basic understanding of OOP, OOAD, programming languages, and database structure. Understanding the Business goal and problem Using a wireframe drawing tool Soft skills: Collaboration Communication skills Analytical and Critical Thinking	0923921807	ITITIU20193@student.hcmiu.edu.vn	$721/ month

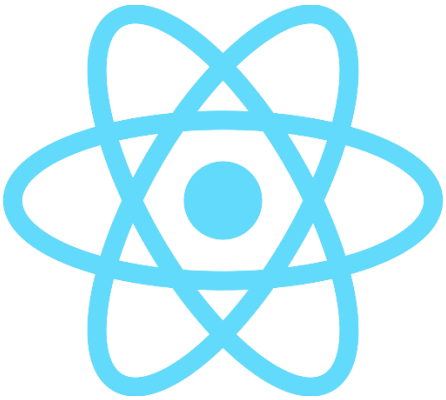
Đặng Quang Vinh	Hitachi Ltd – Inspired the next.	Project Manager	Technical skill: Basic understanding of OOP, OOAD, programming languages, and database structure. Knowledge of the development process. Familiar with project management methodologies, software Risk management Soft skills: Leadership Communication skills Problem-solving Time management	0905886245	ITITIU19247@student.hcmiu.edu.vn	$999/month
Nguyễn Võ Thanh Tâm	Hitachi Ltd – Inspired the next.	UI/UX, Analysis Design	Technical skill: Basic understanding of OOP, OOAD, programming languages. Knowledge of prototyping, mockup and wireframing. Proficient in using visual design software (Photoshop, AI, Xd) Soft skills: Collaboration, Communication skills Creativity Time management	0968552142	ITITWE19012@student.hcmiu.edu.vn	$535/month
Phạm Hồng Đăng	Hitachi Ltd – Inspired the next.	Front-end Developer	Technical skill: Basic understanding of OOP, OOAD. Deep knowledge of HTML, CSS & JS (framework) Understanding the fundamentals of responsive design Basic design skills Soft skills: Collaboration, Communication skills Creativity Problem-solving Time management	0968227204	ITITIU19009@student.hcmiu.edu.vn	$557/month

Lê Quý Minh Đức	Hitachi Ltd – Inspired the next.	Back-end Developer	Technical skill: Basic understanding of OOP, OOAD, programming languages, and database structure. Deep knowledge of Backend Programming Languages & Data Structures and Algorithms Familiar with database, API and servers Knowledge of HTML, CSS, JS Soft skills: Collaboration Communication skills Problem-solving Time management	03054 22899	ITITIU20188 @student.hcmiu.edu.vn	$650/ month
Total:						$6924.00

Table 5: Human Resources

1.7. Technologies

1.7.1. React JS



React (also known as React.js or ReactJS) is a free and open-source front-end JavaScript library for building user interfaces based on UI components. React can be used as a base in the development of single-page, mobile, or server-rendered applications with frameworks like Next.js. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality.

1.7.2. NodeJS

Node.js is an open-source server environment. Node.js is cross-platform and runs on Windows, Linux, Unix, and macOS. Node.js is a back-end JavaScript runtime environment. Node.js runs on the V8 JavaScript Engine and executes JavaScript code outside a web browser.

1.7.3. MongoDB

MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas. MongoDB is developed by MongoDB Inc. and licensed under the Server Side Public License (SSPL) which is deemed non-free by several distributions.

1.8. Schedule

1.8.1. Gantt Chart

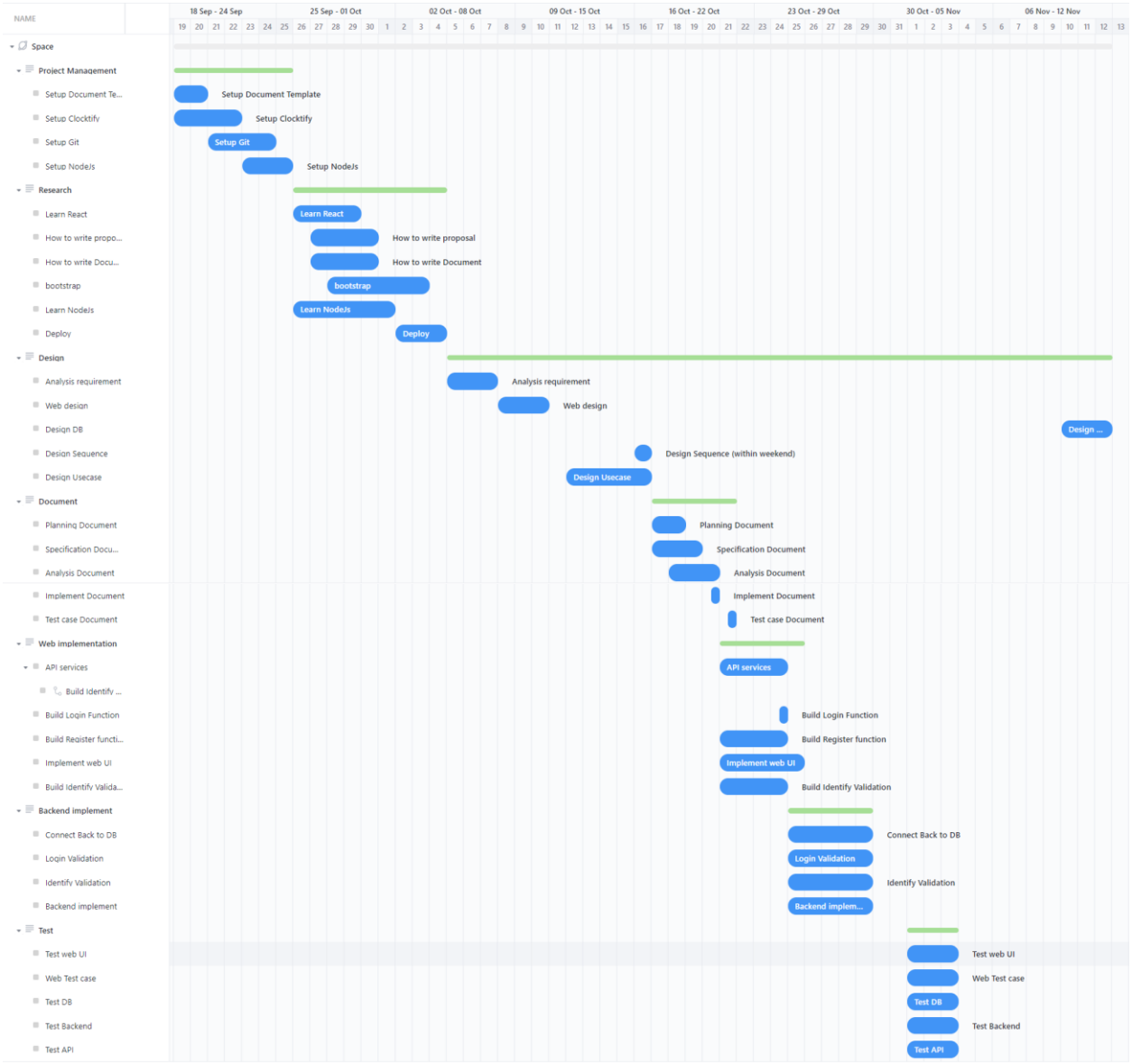


Figure 1: Gantt Chart

1.8.2. Work Breakdown Structure

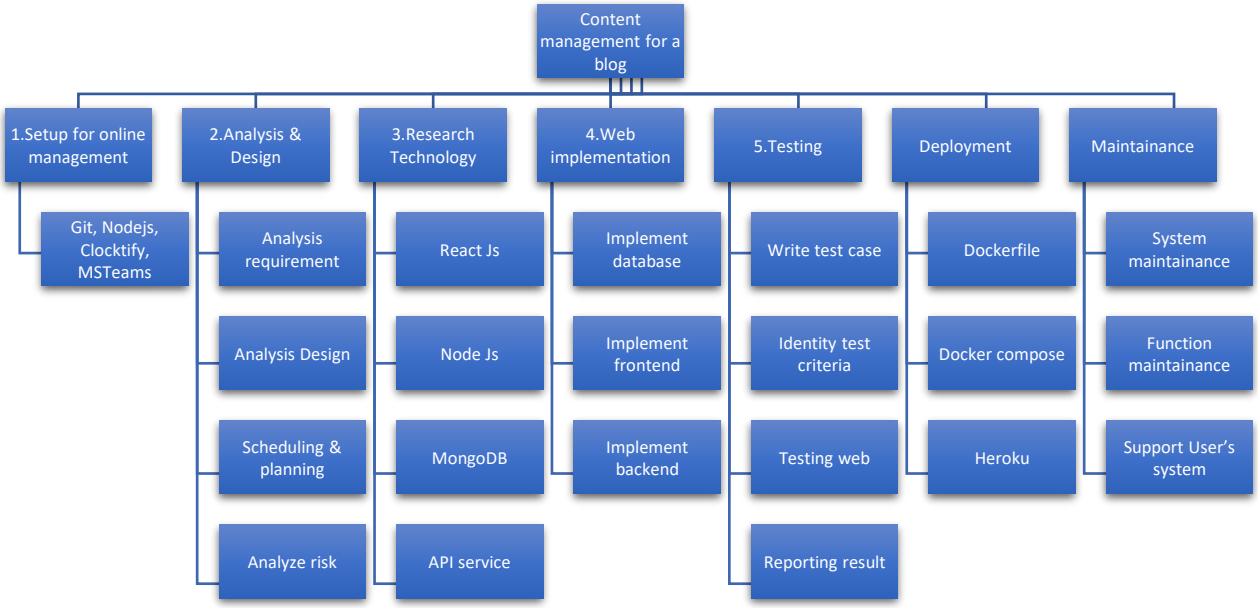


Figure 2: Work Breakdown Structure

1.8.3. Process Development

In order to do the project systematically, the agile planning method is required to use for team cooperation and interaction. One of the most prevalent agile frameworks is scrum. To make a complete scrum, the project manager will first spend time researching the project's requirements and planning the general product backlog and sprint backlog. Once the rudiment plan has been created, the project manager will hold a meeting for discussing and assigning tasks. A daily scrum and weekly scrum meeting will be held during the working process, which helps keep the project on target by doing regular checkups and reviews. In more detail, a daily scrum is reviewing what each person did yesterday and what work will be done today and tomorrow. The tasks will be implemented and tested within one sprint until the project manager feels satisfied with the product, then the subsequent print can start.

1.9. Product Backlog

StoryID	Story Name and description	Status	Sprint	Priority	Estimated days	Comments
1	As the user, I want to log in to the system for authority to perform his/her actions	Open	1	High	10 days	
2	As a user, I want to view other Blog activities.	In progress	2	High	10 days	

3	As the user, I want to modify some attributes of my blog	In progress	3	High	5 days	
4	As a member, I want the system can update the user via email. Also, other users can contact each other.	In progress	4	medium	2 days	
5	I want to search blogs or other users so that I can save time and effort.	Open	5	low	4 days	

Table 6: Product Backlog

1.10. Sprint Backlog

Sprint ID: 001 - System login						
Tasks	Responsibilities	Monday	Tuesday	Wednesday	Thursday	Friday
Define the use case and requirements for the login function	Team	3 hours	3 hours			
Define the use case and requirements for the homepage of the blog function	Team			3 hours	3 hours	
Define the use case and requirements for the modification of the blog	Team					4 hours
Design login function	Vo Trung Duong	6 hours	6 hours			
Design login homepage	Pham Hong Dang			6 hours	6 hours	
Create a database for user	Dang Quag Vinh					6 hours

Table 7: Sprint Backlog System login

Sprint ID: 002 – Reading the blog						
Tasks	Responsibilities	Monday	Tuesday	Wednesday	Thursday	Friday
Design the latest news feature – listing the 5 most recent articles.	Dang Quang Vinh	6 hours				
Design article listing page	Team	3 hours	3 hours	3 hours	3 hours	
Design ‘view article’ page, leads to single article	Nguyen Vo Thanh Tam		4 hours			
Design adding new article function	Pham Hong Dang				5 hours	
Design listing article function	Pham Hong Dang		6 hours			

Table 8: Sprint Backlog Reading the blog

Sprint ID: 003 – Content Modifier						
Tasks	Responsibilities	Monday	Tuesday	Wednesday	Thursday	Friday
Design the functionality for modifying after an article is created.	Team	4 hours	4 hours	4 hours		
Design preview display allows editing content.	Minh Duc		6 hours	6 hours		
Create a function to keep an audit log of all changes to content (including additions, modifications, and deletions)	Pham Hong Dang			6 hours	6 hours	

Table 9: Sprint Backlog Content Modifier

Sprint ID: 004 – Contact the users						
Tasks	Responsibilities	Monday	Tuesday	Wednesday	Thursday	Friday
Add a section on the user account to insert contact information	Duc	2 hours				
Develop a model which sends users emails whenever any update occurs.	Duc		3 hours			

Table 10: Sprint Backlog Contact the users

Sprint ID: 005 – Search-for-content model						
Tasks	Responsibilities	Monday	Tuesday	Wednesday	Thursday	Friday
Research available searching algorithm	Duc	4 hours				
Develop a model, add dataset for training	Team		4 hours	3 hours		
Deploy to the system	Team				4 hours	

Table 11: Sprint Backlog Search-for-content model

1.11. Risk Register

In this part, a list of identified risks and other information are defined

Risk Category: Some common categories are scope, schedule, cost, quality, HR.

Risk title: a one-sentence description of a risk.

Affect: A narrative description of the potential impact on the project.

Probability: the likelihood of risk to happen. You can use Low, Medium, High

Impact if the risk occurs: You can use Low, Medium, High

Risk response plan: Actions to respond to the risk.

Risk ID	Risk Category	Risk Title	Affect	Probability	Impact	Risk response plan

001	Team Corporation	Because this project is organizing online, so the meeting and tracking is difficult	Slowing down the project progress and produce unqualified products	Low	Medium	Set up an online management tool and organize a daily meeting
002	Pressure	Project managers want to reduce task durations, so they run task-parallel and have an additional requirement	Increasing number of errors and mistakes in the project	High	Low	Hold a weekly meeting in order to update the Schedule and plan
003	Misunderstood	Lack of communication, causing confusion and lack of clarity	Producing products with the wrong requirement and not meeting the criteria. This will end up wasting time.	Low	High	Project managers need to set clear expectation and criteria. Also, consistent asking and communicating with team members to ensure they all understand well

004	Changeable	Team members change the code, data, or documents without notifying	Task leader will lose track, and when there is a problem appears, the team has to spend time reading everything all over again	Medium	Low	Create backup files and doing a project with tools that help to mark the changes.
005	Hardware conflict	When the project goes on different computers, it may not run with some team members' computers because of the wrong version of the application	Spending time finding other tools to replace and build the project again	Low	Mediu m	Project managers will set up and check everything correctly at the beginning of the project so that the team can change the tools immediately
006	Wrong data	System updates wrong data (e.g., balance, location)	Causing lousy service to customers	High	High	Requiring the developer to create an efficient algorithm that the system can run more accurate
007	Request	Too many request from customers	Lagging systems take many time loading	High	Mediu m	Using technical that help system runs faster
008	Time	Besides the project, the members also	Lack of time to do the	High	High	Each member has to manage their schedule.

		have other classes. Also, we will have midterm and final exams	projects, reduce the quality of products.			There will be a punishment if they miss the deadline

Table 12: Risk Register

1.12. Security Aspect

The security aspect is essential in the design and development of web services. There are several types of strategies that can be distinguished to address communications protection between web services, and the two main categories include transport-level and message-level security. Advanced transport-level security means that the authenticity, integrity, and confidentiality of messages (E.g., SOAP messages, QR code scanning) are fully authorized for transactions with lower-level modes of self-transport and messages from sender to receiver. To encrypt packets over the network, the public-key techniques are used to authenticate endpoints and accepted to a symmetric key simultaneously by those protocols. Therefore, with this tight security, it can help users check and ensure the safety of their credit cards. However, those messages are vulnerable to attacks based on the rapid modification of messages, leading to several consequences such as unauthorized access, information disclosure, or identity theft. Hence, message level security and web services are covered in many standards, such as WS Security, which provides mechanisms to ensure end-to-end security and allow the protection of certain sensitive parts of information.

Learn each security option:

+ Allow the site to check if customers have saved their payment method.

+ Safe Browsing: Get an alert whenever banking employees see a credit card page; customers are visited by someone with malicious intent. When customers access an account, then the system will check the compatibility among the IP addresses. If there has no match, the bank will send a partial copy of the IP address to email or SMS to determine whether customers access their account. Furthermore, it helps increase security for everyone on the website. The system will periodically send warning emails to customers when there are suspicious activities.

1.13. Naming Conventions and Definitions

#	Name	Definition
1	HTML (Hypertext Markup Language)	The programming language used to build and restructure the components contained on the Website
2	CSS (Cascading Style Sheets)	A language used to find and reformat generated elements
3	JS (JavaScript)	Scripting or programming language that allows implementing complex features on a website
4	API (Application Programming Interface)	Middle method connecting different applications and libraries
5	DevOps	A combination of principles, practices, processes, and tools that automate the programming and delivery of software.
6	Agile	A flexible software development methodology that is a specific approach to software project management

7	Sub Task	The smaller task of User Story
8	Assigned	The status of task that did not assign for whom
9	Sprint	A repeatable fixed time-box during which a "Done" product of the highest possible value is created
10	Scrum	A framework for developing, delivering, and maintaining products in a complicated environment, with an initial emphasis on software development, while it has been used in other sectors, including research, sales, marketing, and advanced technologies.
11	User Story	An informal, natural language description of features of a software system.

Table 13: Naming Conventions and Definitions

1.14. Conclusion

Making changes to website content without a fully-featured content management system can be both costly and time-consuming. A CMS software serves as an all-in-one content marketing platform, allowing you to streamline your workflow and improve efficiency as you build your website. Today Content management systems (CMS) to help them to deliver targeted information for visitors interested in their products and services, there are many software packages available. With CMS, non-techical users can create functional pages upload and modify content themselves, without having to hire a web developer or know programming languages.

Obviously, your site and CMS are just a single little piece of the story. Additionally, having effective communication tools that enable omnichannel support would be helpful. CMS software allows you to create and edit content without the assistance of a developer. It is also simple to set up and maintain. It includes all of the necessary tools and functionality to increase your chances of success.

II. PROJECT REQUIREMENTS

2.1. Project User Cases

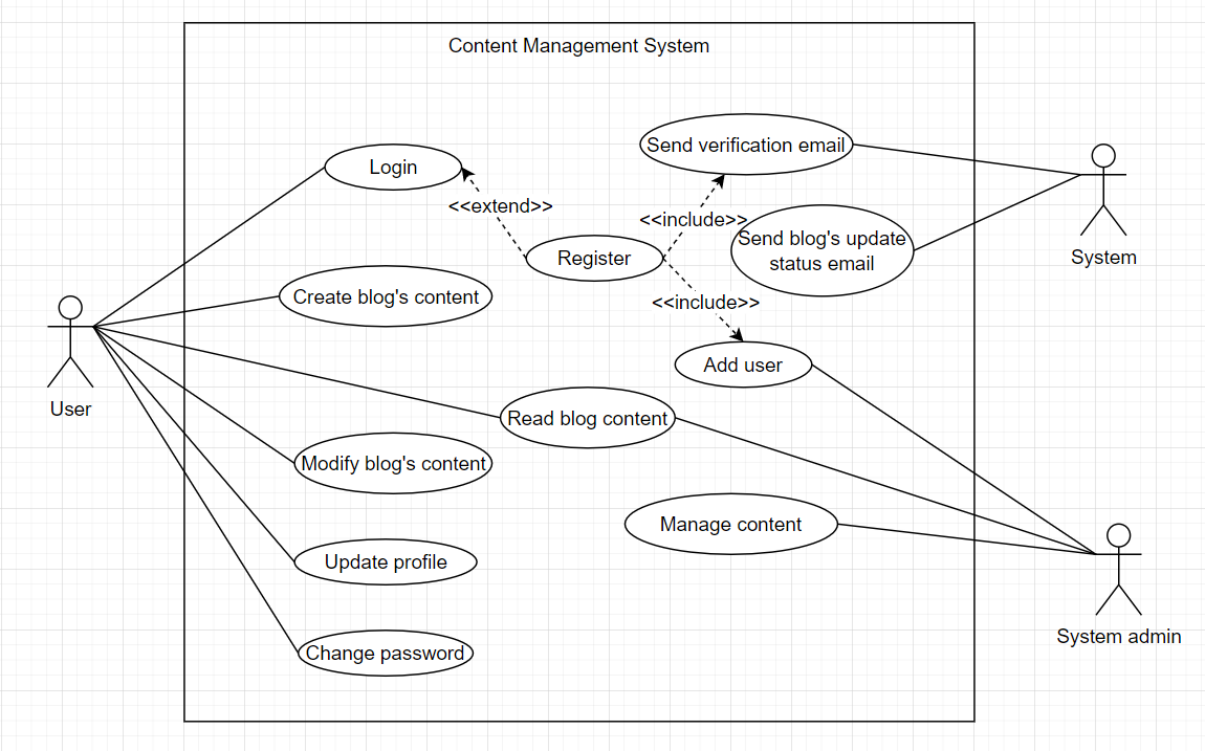


Figure 3: Project User Cases

User cases descriptions:

Use case 1: Log in the blog system

Identifier UC1

Inputs:

1. Username
2. Password

Outputs:

1. The home page with user's authorization [If success]
2. The login page
3. Register

Basic Course:

Actor: User (user/admin)

System

Open the login page

Display the login page

Enter username and password

Submit

- Check the user's info.

- If success, return the home page

- Else return the login page

- Else register new account.

Precondition

User has a registered account that is created earlier (username and password)

Post condition

None

User story:

As a blog visitor, I want to log in to the system so that I can use functions of the blog like create blog content, update profile, modify blog's content, read blog content…

As a system admin, I want to add user to the blog.

Use case 2: create blog content

Identifier UC1

Inputs:

Log in successful.

Outputs:

The main blog page.

Basic Course:

Actor: User

System

Open the main blog page

Display the main blog page

Precondition: User has to login already.

Post condition: None

User story:

As a blog visitor, I want create blog's content to another user can see my blog.

Use case 3: modify blog's content

Identifier UC1

Inputs:

Log in successful.

Outputs:

The main blog page.

Basic Course:

Actor: User

System

Open the main blog page

Display the main blog page

Modify user blog's content

Modify the blog's content

Precondition: User has to login already.

Post condition: None

User story:

As a blog visitor, I want to modify my blog's content.

Use case 4: Update profile

Identifier UC1

Inputs:

Log in successful.

Direct web address from blog main page to profile page.

Outputs:

The profile page.

Basic Course:

Actor: User

System

Open the profile page

Display the profile page

Edit the information on user's profile

Edit user's information on user's profile

Update the information on user's profile

Update user's information on user's profile

Precondition:

- User has to login already and direct web address from blog main page to profile page.

Post condition: None

User story:

As a blog visitor, I want to update my information of my profile.

Use case 5: Read blog content

Identifier UC1

Inputs:

Log in successful.

Outputs:

The main blog page

Basic Course:

Actor: User (user, admin)

System

Read all blog contents in blog website

Display all content of blog.

Precondition:

- User has to login already.

Post condition: None

User story:

As a blog visitor, I want to read all the blog content.

As a blog admin, I want to read all the blog content.

Use case 6: Change password

Identifier UC1

Inputs:

Log in successful.

Direct web address to profile page

Outputs:

The profile page.

Basic Course:

Actor: User (user)

System

Change password

Replace nthe ew password the for user.

Precondition:

- User has to login already.

- Direct web address from blog main page to profile page.

Post condition: None

User story:

As a blog visitor, I want to read all the blog content.

As a blog admin, I want to read all the blog content.

Use case 7: Manage content

Identifier UC1

Inputs:

Log in successful.

Outputs:

The admin blog page.

Basic Course:

Actor: Admin

System

Mange content of blog.

Allow admin account can manage content of the blog

Precondition:

- Admin has to login already.

- Direct web address from blog main page to admin blog page.

Post condition: None

User story:

As a blog admin, I want to manage all the blog content.

2.2. Functional and Non-functional Requirements:

Req.ID	Requirement Name	Detailed description	Type

001	Register an account	If the user doesn't have an account, then he/she will be suggested to register, User must have email for that account to be verification	Functional requirement
002	Create/modify blog's content	Users with an available account can create new blogs, modify or delete them.	Functional requirement
003	Add information to profile	Users can edit or add extra information to his/her profile.	Functional requirement
004	Email verification and notification	through registered email, users have the ability to change password, delete account. Also, user can get notification through that email.	Functional requirement
005	Security requirement	Server must have the ability to detect any attack (Cross site scripting, SQL injection, Clickjacking,...) and protect users from them.	Non-functional requirement
006		If a user cannot log in to an account, they would have the option to change password on request (user must be authenticated by email) (see Req.004). Otherwise, if a user exceeds 5 times logging in, he/she has to wait for 30 minutes for the next action.	Non-functional requirement

Table 14: Functional and Non-functional Requirements

III. Design

3.1. System Architecture:

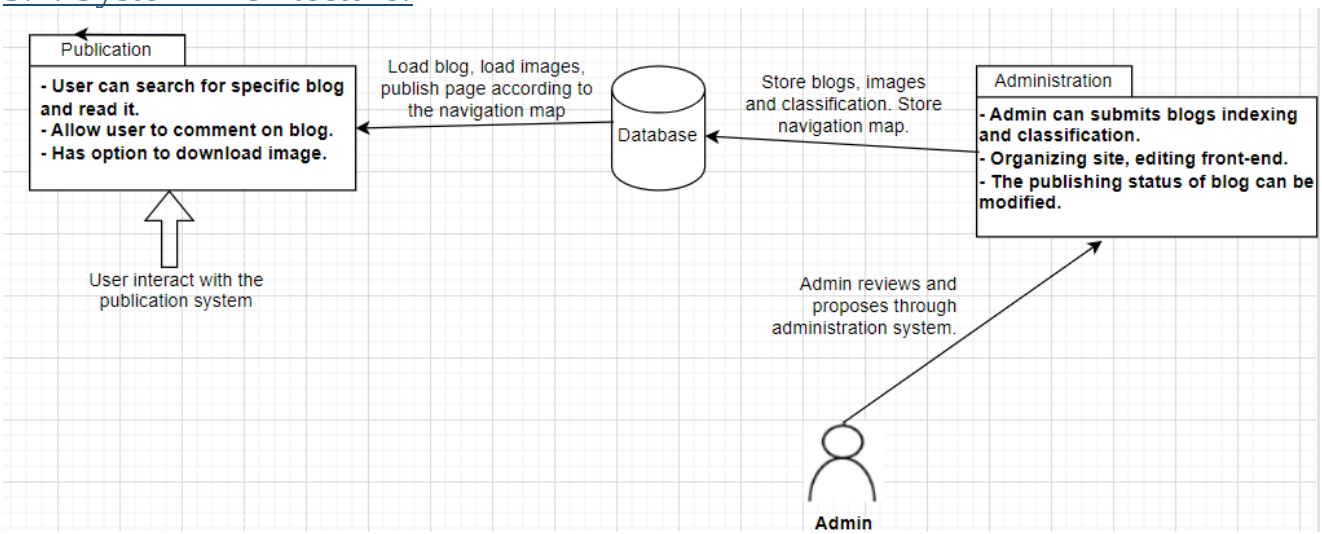


Figure 4: System Architecture

3.2. Database Structure

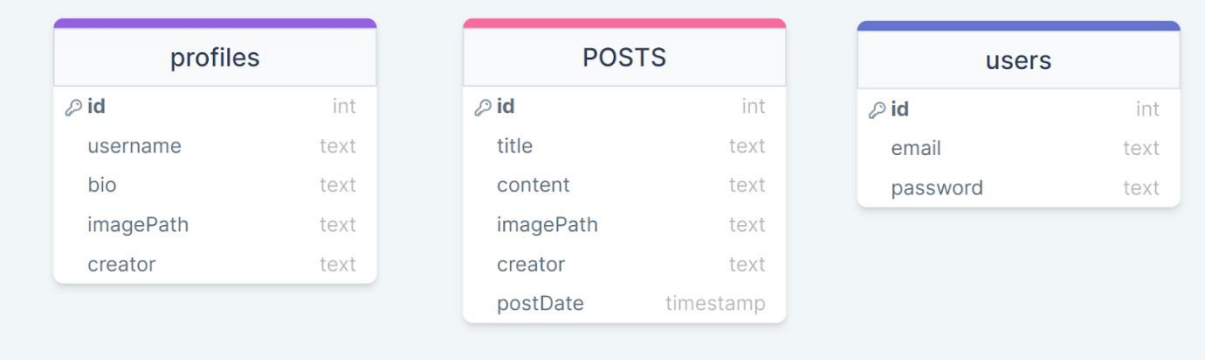


Figure 5: Database Structure

3.3. Entity relationship diagram:

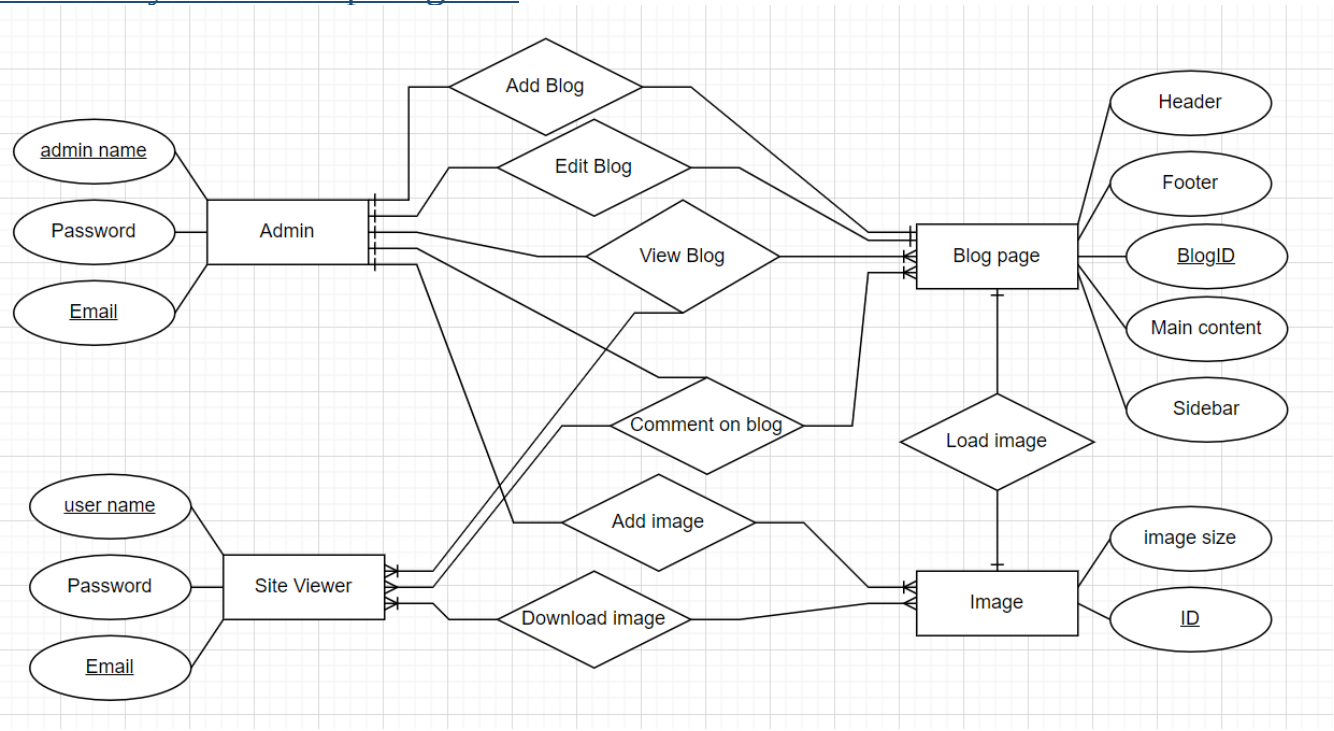


Figure 6: Entity relationship diagram

3.4. Sequence diagram:

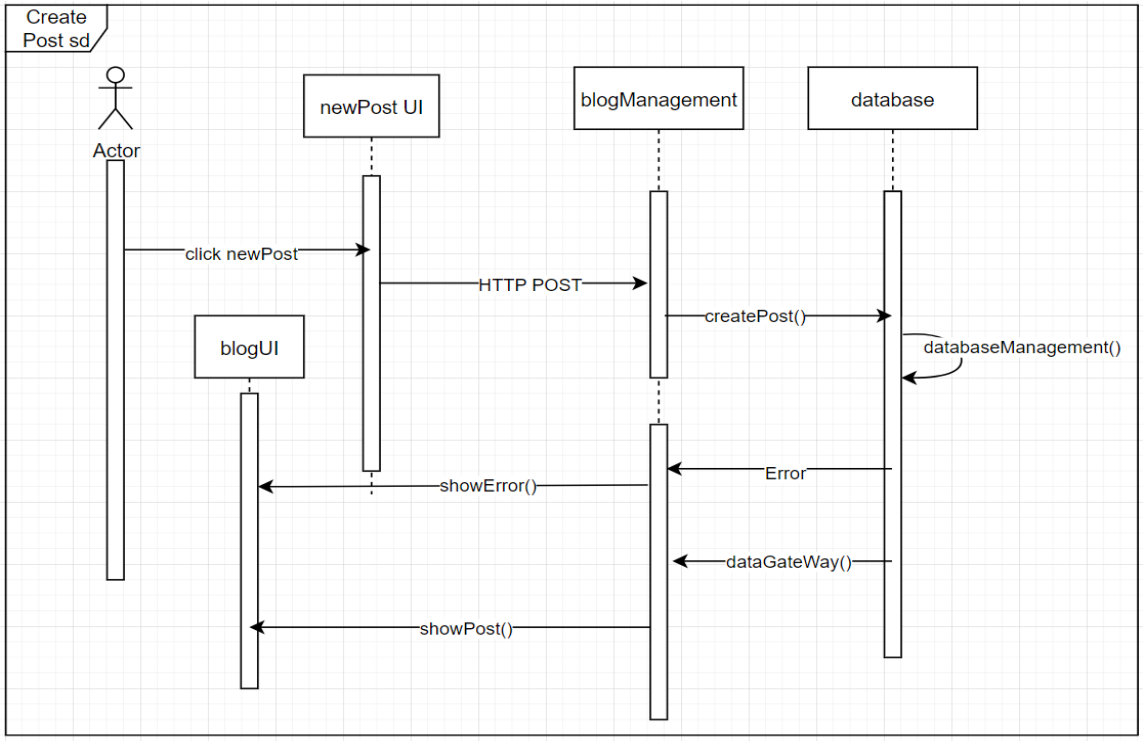


Figure 7: Create post Sequence diagram

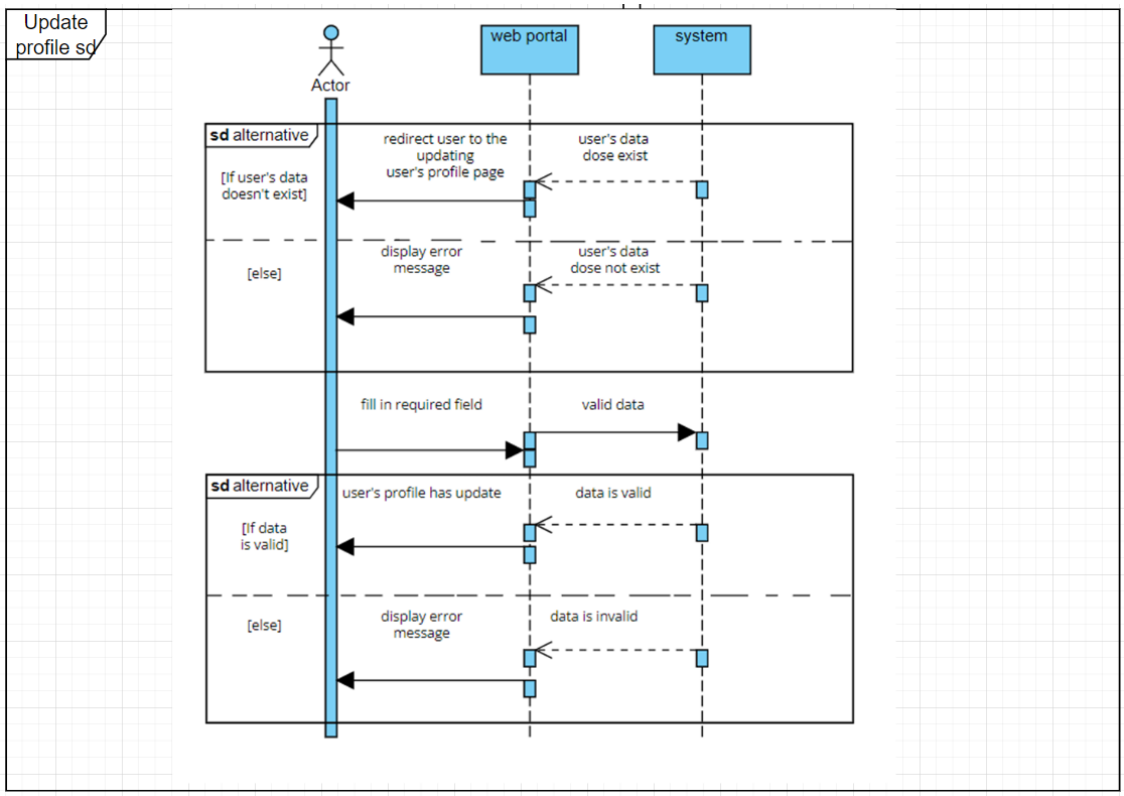


Figure 8: Profile Sequence diagram

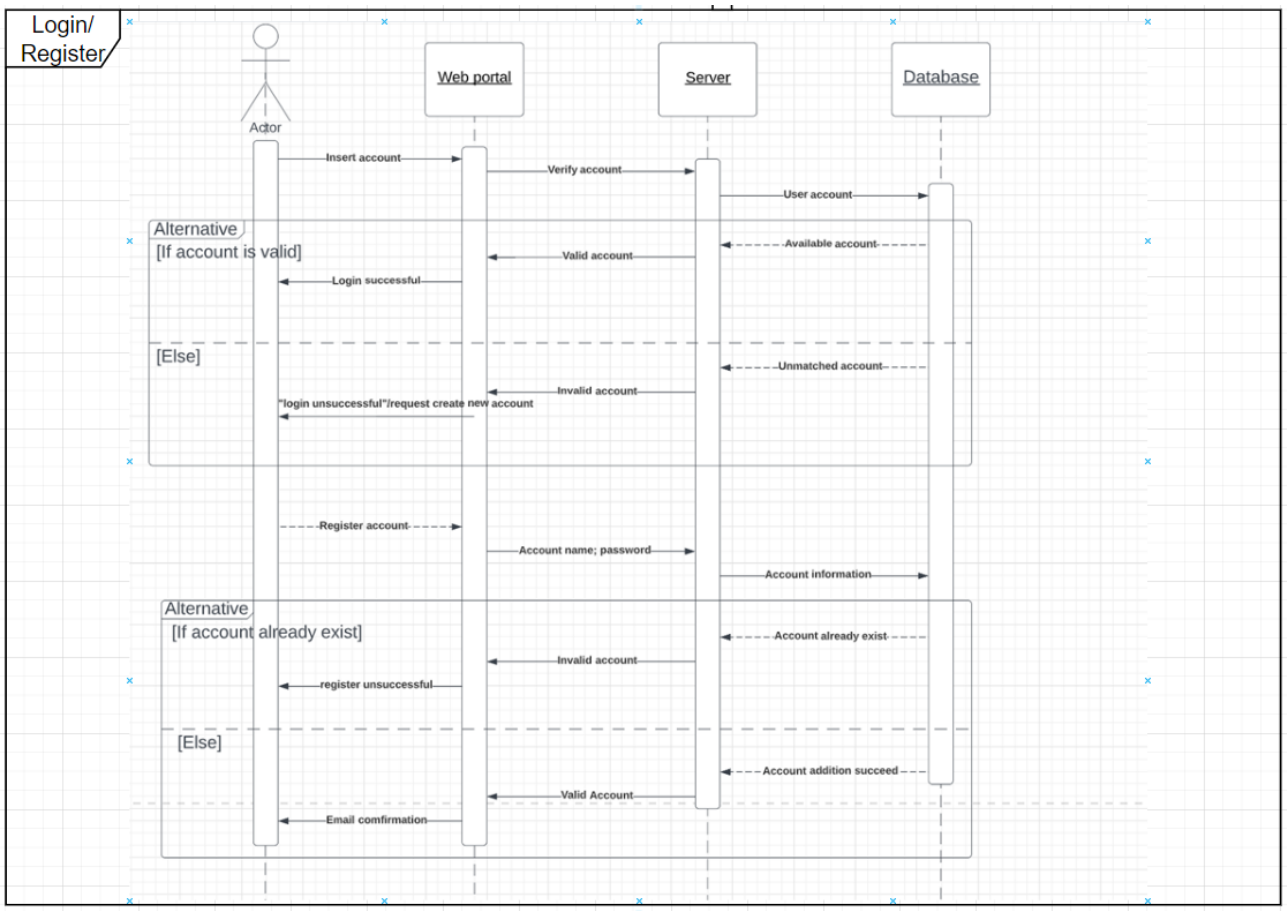


Figure 9: Login/Register Sequence Diagram

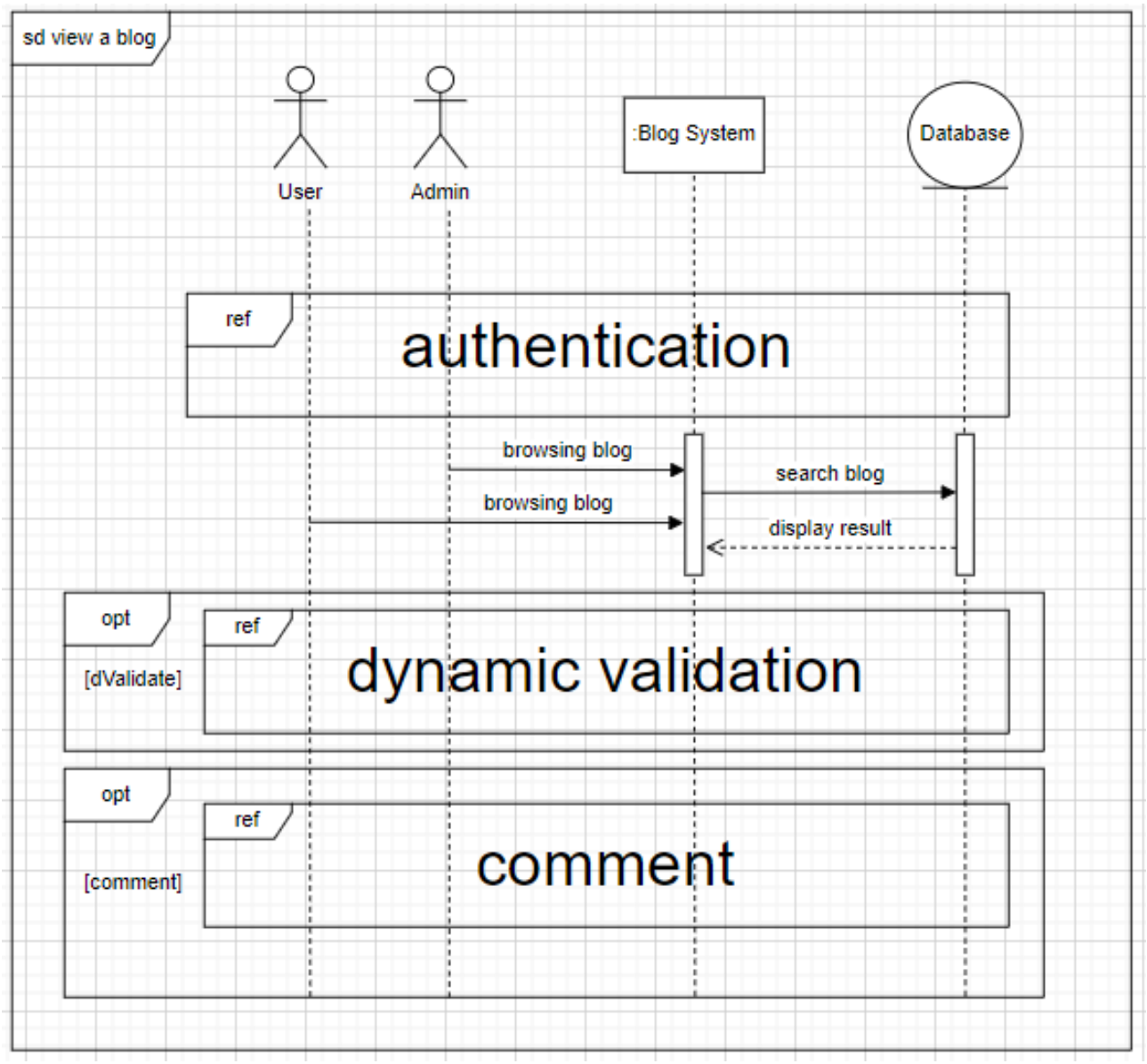


Figure 10: View a blog Sequence diagram

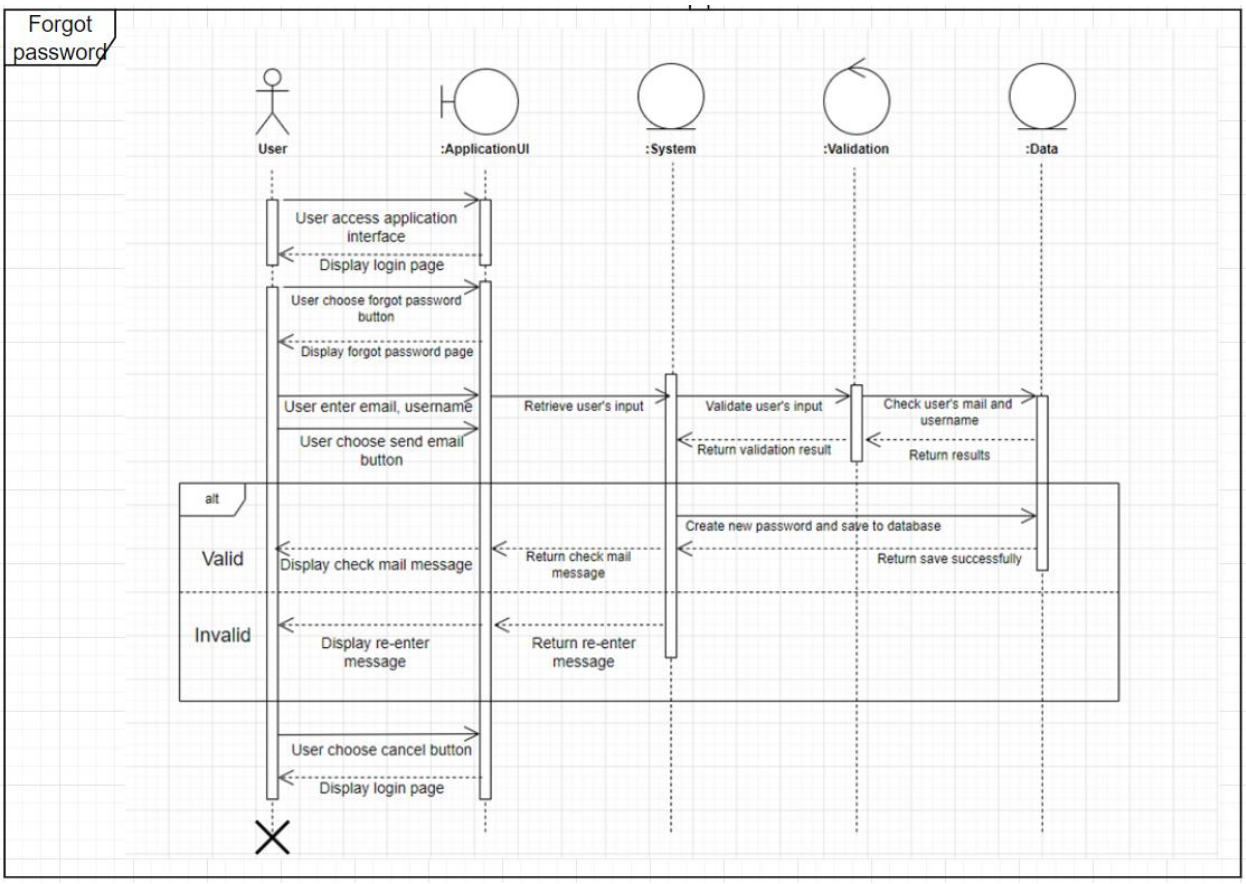


Figure 11: Forgot Password Sequence diagram

3.5. Class diagram:

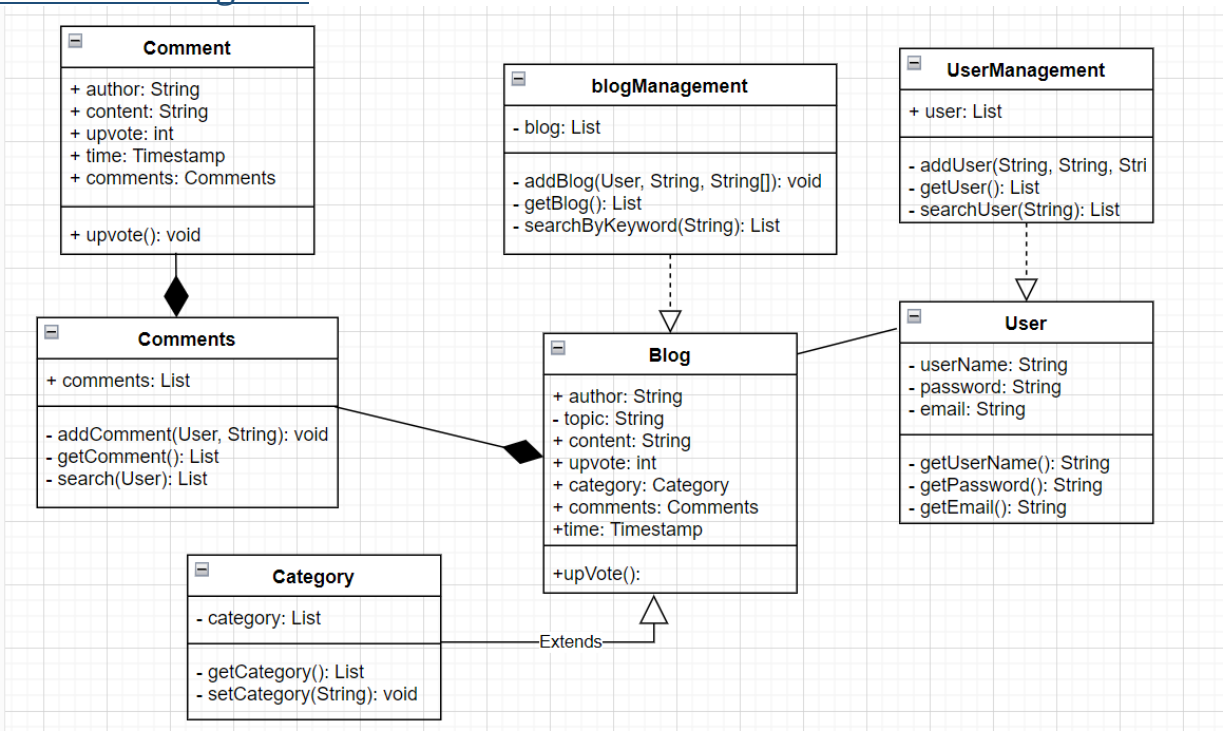


Figure 12: Class Diagram

IV. Implementation

4.1. Login function

When users have entered to the website, the first page appear to the users are home page which have a large title “Publish your passions, your way” and all the popular blogs and popular writers.

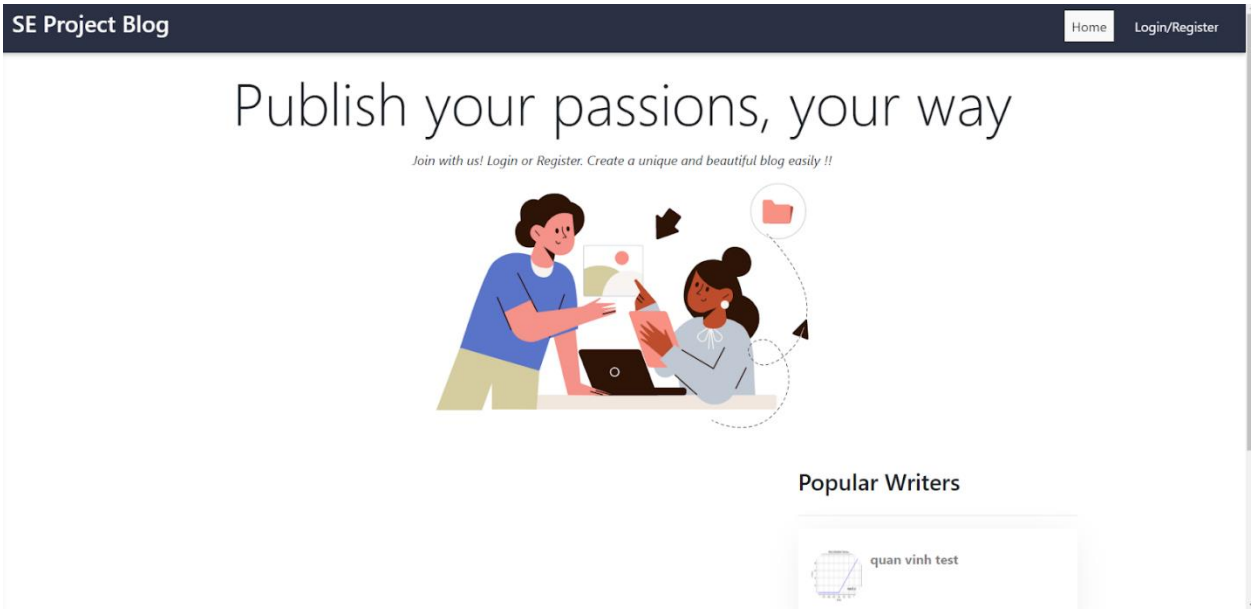


Figure 13: Home Page

If users already had an account, they can login to system by clicking in the login button and then providing their email and password, which they had registered before, into the form. If users haven't registered, they need to click the register button to create a new account. The system will show an announcement if there's any error occur.

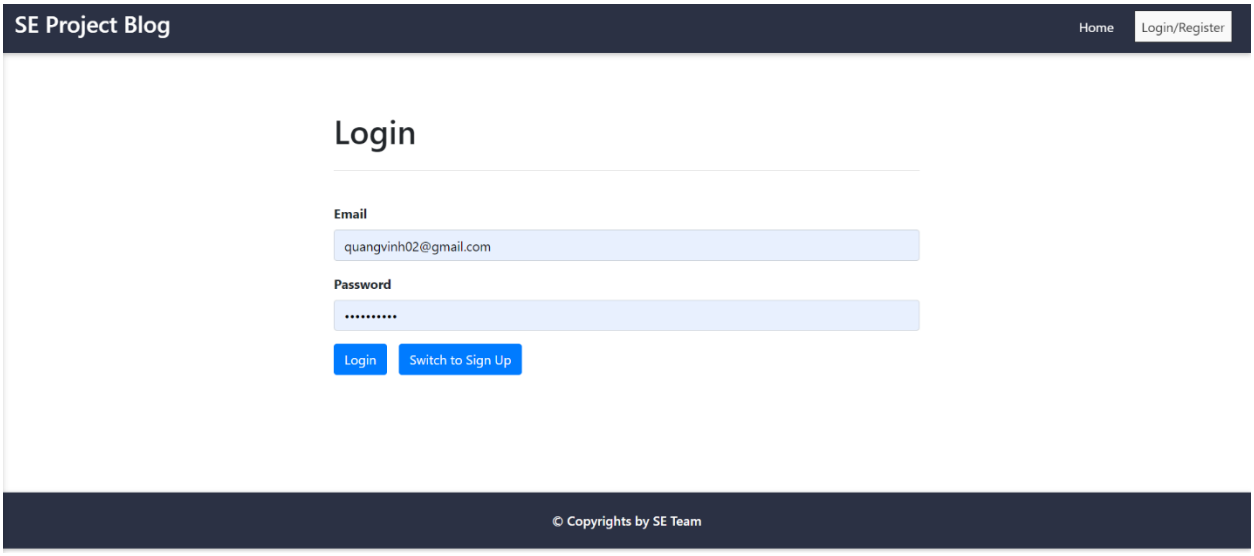


Figure 14: Login/Register page

When users can login, they will be transferred to homepage of blog website and the navigation bar will show more functions to transfer to another page (Profile page, My Post page, Create Post page).

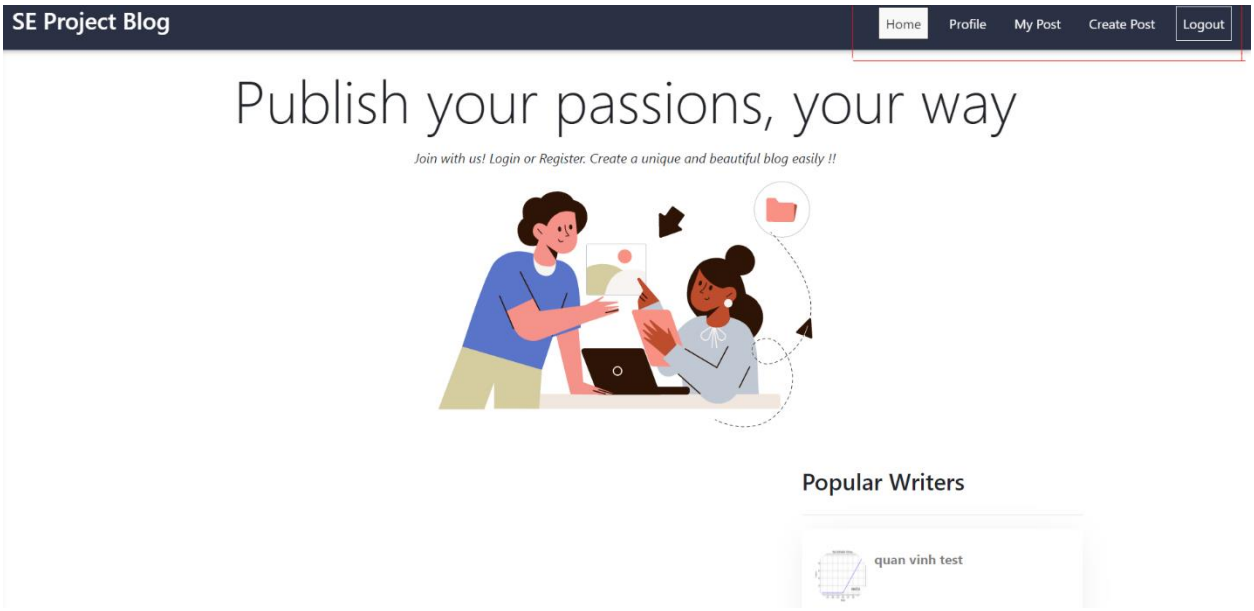


Figure 15: Home page after user login

4.2. User function

4.2.1. Read blog

At the homepage interface, users can read a Blog by clicking any blog in homepage.

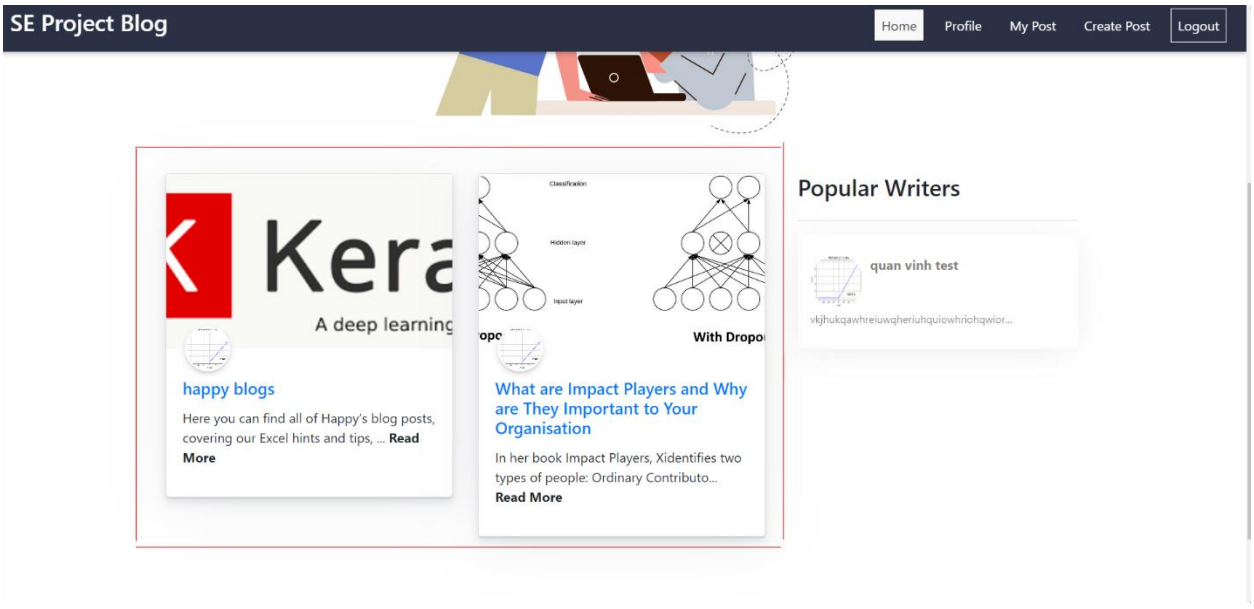


Figure 16: The Blog zone where user can read all blogs.

After clicking of the blog tile, the user will move to that blog page to read more about this blog.

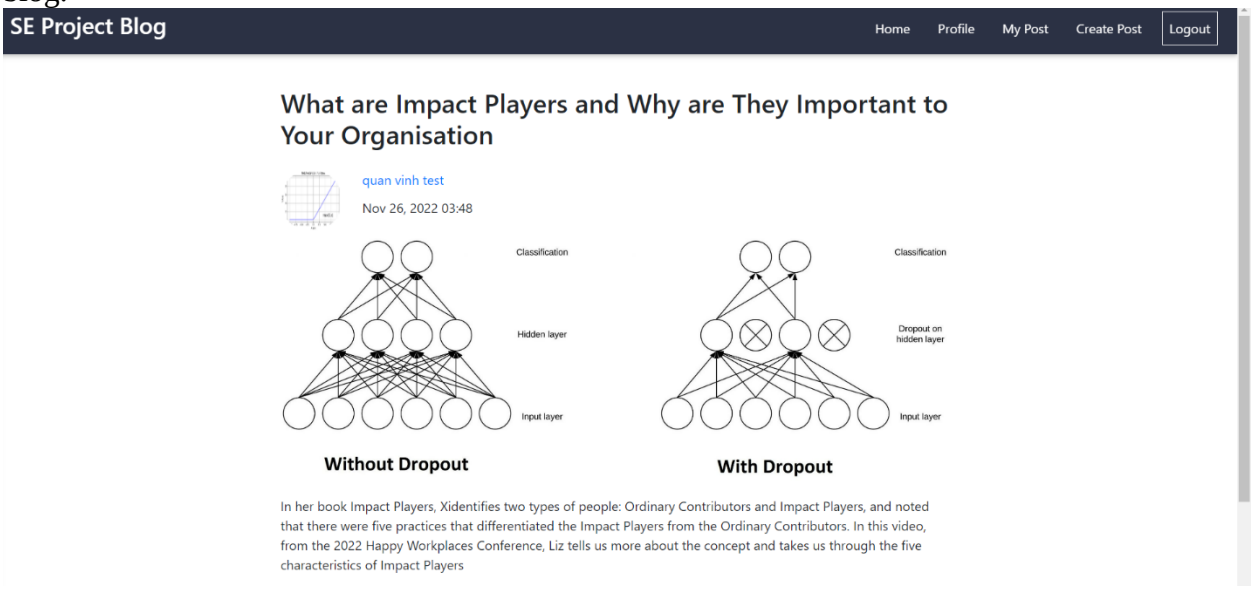


Figure 17: The page of blog has title "What are Impact Players and Why are They Important to Your Organisation"

4.2.2. User Profile

After click the Profile button, user will move to the profile page where show all the information of the user (username, description profile and user image, user's blog).

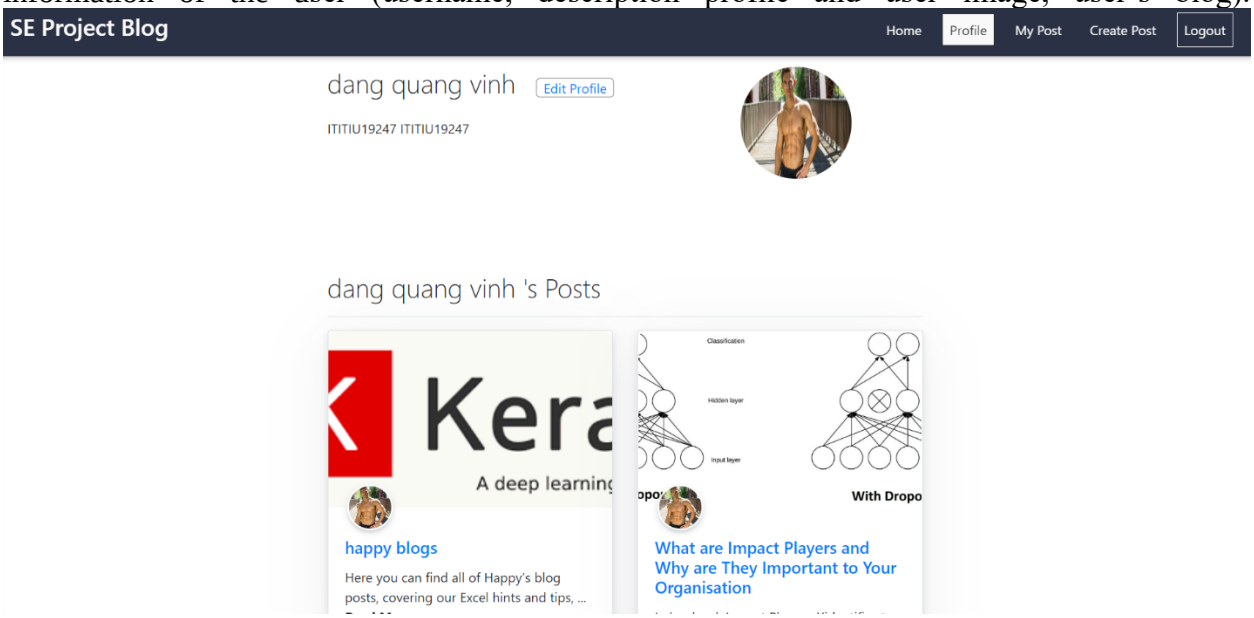


Figure 18: The profile page

If users want to edit their profile users can click the edit button to edit their information (username, description profile and upload image).

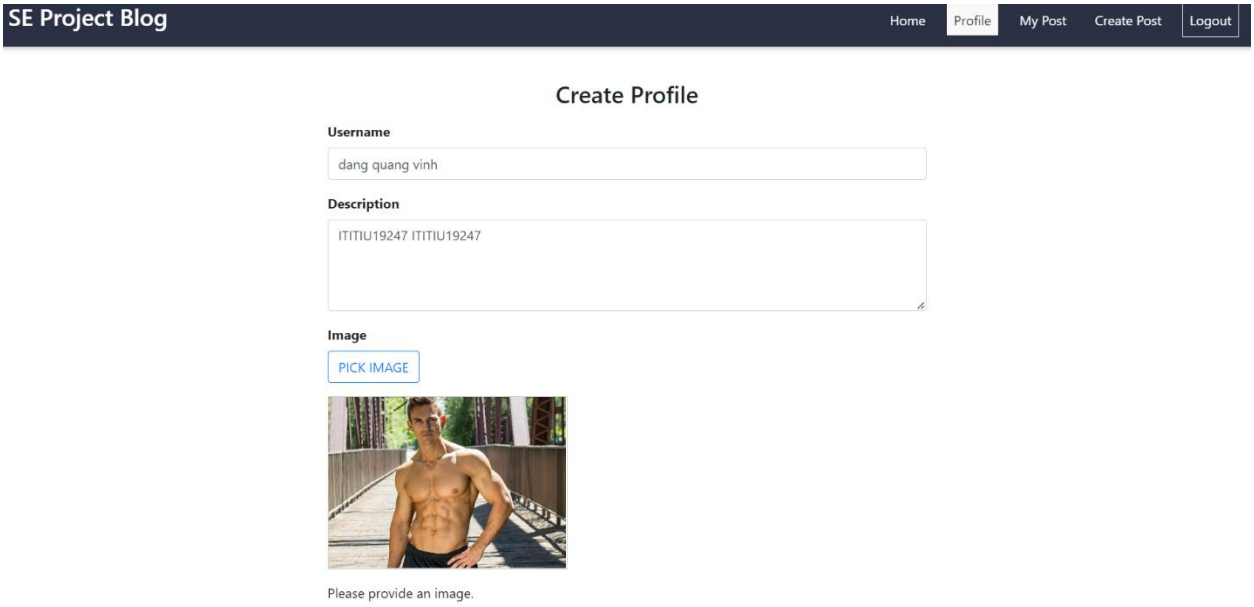


Figure 19: Edit profile page.

4.2.3. User's blog

If user want to see their blog clearly, they can move to the "My Post page" by clicking the My Post button in navigation bar.

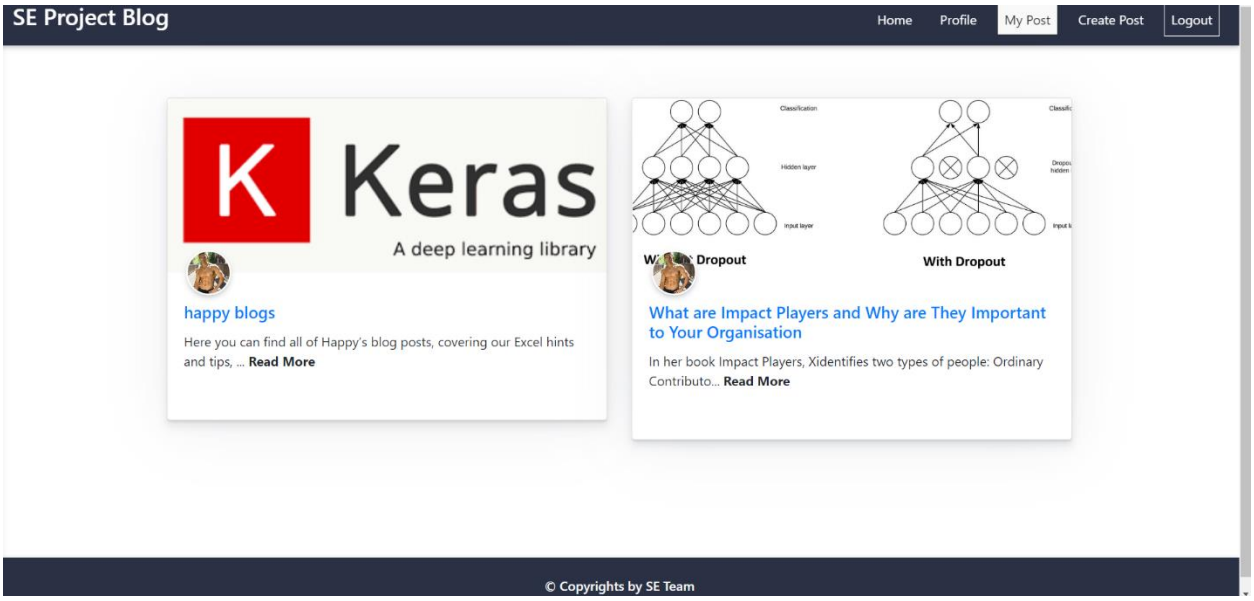


Figure 20: The My Post page

4.2.4. Create blog

At the navigation bar, users can click on the "Create Post" to proceed to create new post for their blogs. Next, the user has to fill all the information of their new post (title, description, image of the post) then clicking “submit” button to publish new post.

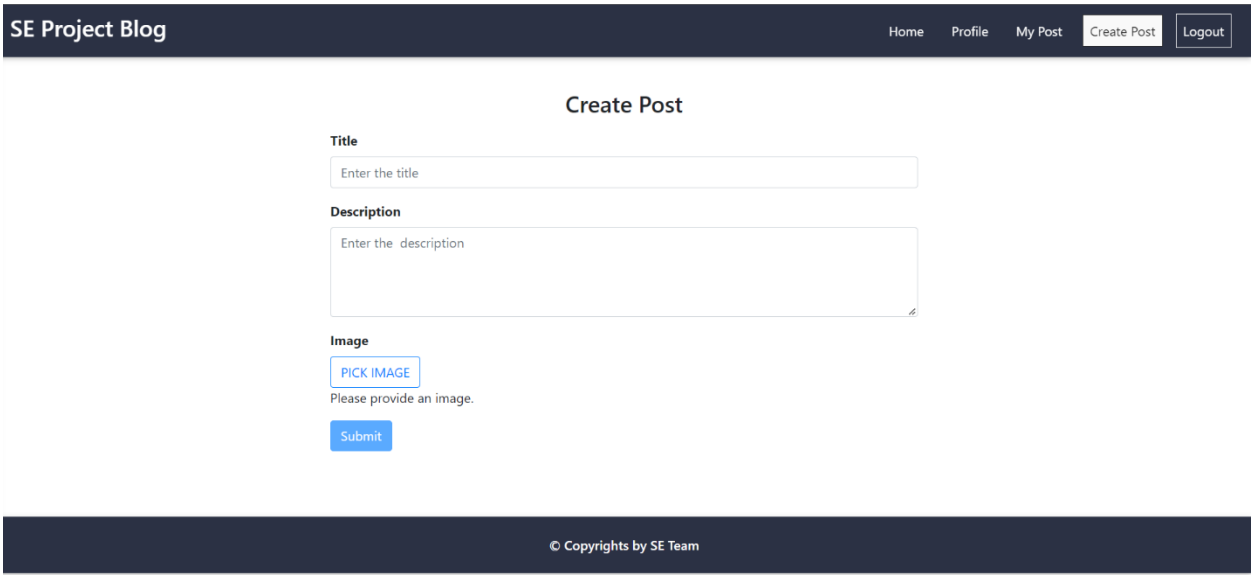


Figure 21: The create blog page

V. Test Plans

5.1. Feature to be tested

5.1.1. In scope:

All of the Content Management System For a Blog features that were outlined in the software requirements must be tested.

Module Name	Roles	Description
Login	User & Admin	User/Admin: A user and an admin are only permitted to have one account with a special email. They can access the program by entering their username and password.
Create Account	Admin	Admin: An admin is only permitted to have one account with a single email address. A manager can create accounts for several users.
Create Blog	User	User can click on the "Create Post" to proceed to create new post for their blogs. Next, the user has to fill all the information of their new post (title, description, image of the post) then clicking “submit” button to publish new post.
Forgot Password	User & Admin	User/Admin: If a user or administrator forgets their password for their own account, they can receive a new password via email.
Change Password	User & Admin	Users and administrators both have the ability to change passwords.
Manage Users	Admin	Administrator: By searching for a certain user's ID, an administrator may view all of that user's information.
Manage Blog History	Admin	Administrator: By searching for a specific user's ID, an administrator can view all of their blog history.

Table 15: System features

5.1.2. Out of scope:

Legal matters. Legal guidelines, frameworks, or standards do not determine the seriousness of a fault, and we are not legal consultants.

Issues with virus scanners, ad blockers, or browser extensions can cause them to prevent particular content or even app execution.

Setup issues are simply a sign that there is a problem with the test setup or the test environment; they are not actual defects.

There won't be any logical, performance, or stress testing of the database.

5.2. Pass/Fail Criteria

The case is designated as having passed after meeting the requirements. It is still recorded as passing even if there are a few minor issues with the anticipated outcome.

If the test case cannot execute due to an environmental condition or if the output does not match the intended output, the case is marked as failed.

Test Case	Cases	Pass	Fail
Login	Loading page	Can display login page when users choose login button	Page die even network is fine
	Enter valid username and password	1 Alert user if the account is not found or password is not correct 2. Login to the right account	1. Cannot login but do not have alert line for users 2. Login to the wrong account 3. Information is valid but cannot login
Create New Account	Loading page	Load and show main home page when users choose register	1. Stuck at Register page for users 2. Page die even network is fine
Create New Blog	Loading page	Can display Create Post page when user choose create post button	1. Stuck at Create Post page for users 2. Page die even network is fine
	Enter correct form: fill title, description and insert Image	1. Alert users if not meet the criteria 2. Continue if users input correct form	Not alert error to users and continue when users enter wrong information.
	Input field	Alert users when they leave the blank	
Manage User	Loading page	Load and show web page for admins after login by admin account	1. Load web page for users 2. Page die even network is fine
	View users' information	1. Check existence of user ID 2. If admin click to the button view, the field of	1. Not alert errors to users 2. Cannot show user information 3. Show the wrong user's information

		entering user ID cannot be empty 3. View all basic information of users 4. Alert errors to users	4. Show another user information
Manage Login History	Loading page	1. Load and show web page for admins after login by admin account 2. Load correct page when admin choose Manage login history button	1. Load web page for users 2. Page die even network is fine

Table 16: Pass/Fail Criteria

5.3. Approach

5.3.1. Process of testing

• Planning and Control

• Analysis and Design

• Implementation and Execution

• Evaluating exit criteria and Reporting

• Test Closure activities

5.3.2. Testing level

• Unit testing: test each separated part of the software to define whether the components are fulfilling functionalities or not

• Integration testing: test the data flows from one module to another modules, which is performed by testers.

• System testing: define functional as well as non-functional that are needed for testing

• Acceptance testing: check if the requirements of a specification are met the delivery or not.

• API testing: test all the APIs created for the software

5.3.3. Roles and responsibilities

No	Role	Name	Responsibilities
01	Test Manager	Quang Vinh	Prepares test strategy, defines the scope of work

02	Software Tester	Nguyen Vo Thanh Tam	Execute test scenarios for software applications
03	Database Administration	Pham Hong Dang	Test database, make sure that test data environment and assets are managed and maintained
04	Developer in Test	Vo Trung Duong	Implement the test cases, test program.

Table 17: Roles and responsibilities

5.3.4. Types of testing

Test Objective	Ensures application navigation, data entry, processing
Technique	Executing each use case, function Using valid and invalid data Displaying the warning messages when invalid data is used Making sure that business rules are applied
Completion Criteria	Planned tests have been executed
Special Considerations	(None)

Table 18: Function testing

Test Objective	Verify that users can only access to the data when they are provided permissions
Technique	Identify and list each user type and the functions/data that each type of user has permissions for. Create test for each user type
Completion Criteria	For each known user type, the function/data is available The function runs as expected
Special Considerations	Access to the system must be reviewed with the systems administrator.

Table 19: Security Testing

Test Objective	Verify the navigation through the application. Verify window objects and characteristics
Technique	Create tests for every window to verify navigation and object states for every application window
Completion Criteria	Each window successfully verified
Special Considerations	Not all properties can be accessed by third-party

Table 20: User Interface Testing

References

- Draw.io, ”Flowchart Making Tool”.[Online].Available: https://www.draw.io/
- Gantt Chart Builder (Free) - Create Gantt Charts – Free: https://clickup.com/features/gantt-chart-view
- MongoDB Atlas Database | Multi-Cloud Database Service: https://www.mongodb.com/atlas/database
- Requirements for a Web Content Management System: https://s3.amazonaws.com/digitalgov/_legacy-img/2014/07/cms-requirements-for-sow.docx
- Content Management System Requirements Checklist: https://www.qgcio.qld.gov.au/__data/assets/word_doc/0010/6004/cms_require.doc
- The Definitive Guide to CMS Architecture: https://developers.bloomreach.com/blog/2019/cms-architecture.html
- Gantt Chart, Work Breakdown Structure: https://kpvida.wordpress.com/tag/work-breakdown-structure/
- The Gantt chart, a working tool of management, by Wallace Clark.
- Pekka Abrahmson, Outi Salo, Jussi Ronkainen, Juhani Warsta “Agile Software Development Methods”, VTT Publications 478, ESPOO 2002.
- Miller, Steve, “Agile Scrum-An Overview”, Pragmatic Software Newsletter, Feb. 2008, Pragmatic Software, 17 Jun 2009.
- NodeJS: https://www.tutorialsteacher.com/nodejs
- Facebook React Documentation: https://github.com/facebook/react
- What is Entity Relationship Diagram (ERD)?: https://www.visual-paradigm.com/guide/data-modeling/what-is-entity-relationship-diagram/

- Ultimate Entity Relationship Diagram Tutorial (ER Diagrams): https://creately.com/blog/diagrams/er-diagrams-tutorial/
- UML Class Diagram Tutorial: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-class-diagram-tutorial/
- The Easy Guide to UML Class Diagrams | Class Diagram Tutorial: https://creately.com/blog/diagrams/class-diagram-tutorial/
- Sequence Diagram Tutorial – Complete Guide with Examples: https://creately.com/blog/diagrams/sequence-diagram-tutorial/
- Sequence Diagram Examples: https://circle.visual-paradigm.com/category/sequence-diagram/