



Alliance with FPT Education

UNIVERSITY OF GREENWICH
COMP-1640-M02-2022-23
Enterprise Web Software
Development
Group Report

Student name	To Nhat Duy
ID number (00xxxxxx)	001272680
Lecturer/Tutor name	LE THANH HUY
Student submission date	04/19/2023

Team name: Group VPQD

Table 1: Team member + Positions + Commitment point

Member's name	ID	Roles	Commitment point (0-10)
Tran Quan Vinh (Leader)	001272758	Scrum master - Fullstack	
Truong Van Phat	001272727	Product Owner - Tester	
Nguyen Huynh Hong Quan	001272731	Frontend - Tester	
To Nhat Duy	001272680	Database designer - Tester	

Table 2: Roles account

Roles	Email	Password
QA (Admin)	admin@gmail.com	123456
Staff	staff@gmail.com	123456

Table 3: URL

URL of the screencast	https://drive.google.com/file/d/1Iw3kGEb03CF-b372NiL3dem9TjRLFAq6/view?usp=sharing
URL of repository	https://github.com/VinhTranQuan/1640-code/tree/main/Idea_Management

Table of Contents

1.	Introduction.....	9
2.	Database.....	9
2.1.	Entity-relationship diagram (ERD).....	9
2.1.1.	“Account” entity.....	10
2.1.2.	“Feedback” entity	11
2.1.3.	“Post” entity	11
2.1.4.	“Category” entity	12
2.1.5.	“Department” entity	12
2.1.6.	“Comment” entity.....	13
2.1.7.	Relationship between entities.....	13
2.2.	Database design	19
3.	Interface design	21
3.1.	Site map system	21
3.2.	Final designs	22
4.	Functionality.....	35
4.1.	Use case diagram.....	35
4.1.1.	Admin.....	35
4.1.2.	Staff.....	36
4.2.	Activity diagram.....	37
4.2.1.	Admin.....	37
4.2.2.	Staff	52
4.3.	Demo function.....	57
5.	Testing.....	76
5.1.	Testing scope.....	76
5.2.	Test details	76
5.2.1.	Sprint 1.....	78
5.2.2.	Sprint 2.....	79
5.2.3.	Sprint 3.....	80
5.2.4.	Sprint 4.....	82
5.2.5.	Sprint 5.....	84

5.2.6. Sprint 6.....	85
6. Agile Scrum implementation	87
6.1 Product backlog	87
6.2 Sprint backlogs.....	88
6.2.1 Sprint 1.....	89
6.2.1 Sprint 2.....	91
6.2.3 Sprint 3.....	93
6.2.4 Sprint 4.....	95
6.2.5 Sprint 5.....	97
6.2.6 Sprint 6.....	100
6.3 Minute of daily meeting.....	102
6.3.1 Sprint 1.....	102
6.3.2 Sprint 2.....	102
6.3.3 Sprint 3.....	103
6.3.4 Sprint 4.....	103
6.3.5 Sprint 5.....	104
6.3.6 Sprint 6.....	105
7. References.....	106

List of Figure

Figure 1: ERD	9
Figure 2: “Account” entity	10
Figure 3: “Feedback” entity	11
Figure 4: “Post” entity	11
Figure 5: “Category” entity.....	12
Figure 6: “Department” entity	12
Figure 7: “Comment” entity	13
Figure 8: “account” and “comment” relationship.....	13
Figure 9: “account” and department relationship	14
Figure 10: “account” and “feedback” relationship	15
Figure 11: “account” and “post” relationship	15
Figure 12: “feedback” and “post” relationship	16
Figure 13: “feedback” and “department” relationship.....	17
Figure 14: “feedback” and “comment” relationship.....	17
Figure 15: “post” and “department” relationship	18
Figure 16: “post” and “category” relationship.....	19
Figure 17: “account” entity.....	19
Figure 18: “category” entity	20
Figure 19: “comment” entity	20
Figure 20: “department” entity	20
Figure 21: “feedback” entity.....	21
Figure 22: “post” entity.....	21
Figure 23: Site map system.....	21
Figure 24: Login	22
Figure 25: Login with admin account.....	23
Figure 26: Post interface	23
Figure 27: Create post interface.....	24
Figure 28: Edit post interface.....	24
Figure 29: Delete post interface	25
Figure 30: Category interface	25
Figure 31: Add category	26
Figure 32: Delete category.....	26
Figure 33: Staff interface	27
Figure 34: Add staff	27
Figure 35: Edit staff interface	28
Figure 36: Delete staff interface	28
Figure 37: Department interface	29
Figure 38: Add department	29
Figure 39: Edit department	30

Figure 40: Delete department interface	30
Figure 41: QA list interface	31
Figure 42: Update profile interface.....	31
Figure 43: Change password interface.....	32
Figure 44: Interface when logged in with staff account	32
Figure 45: View post and feedback interface	33
Figure 46: Submit idea interface.....	33
Figure 47: List feedback interface	34
Figure 48: Comment and like interface	34
Figure 49: Usecase diagram (Admin).....	35
Figure 50: Usecase diagram (Staff)	36
Figure 51: Update account activity	37
Figure 52: Change password activity.....	38
Figure 53: Add department activity	39
Figure 54: Edit department activity	40
Figure 55: Delete department activity	41
Figure 56: Add post activity	42
Figure 57: Edit post activity.....	43
Figure 58: Delete post.....	44
Figure 59: Add category activity	45
Figure 60: Edit category activity	46
Figure 61: Delete category activity.....	47
Figure 62: Add staff activity.....	48
Figure 63: Edit staff activity	49
Figure 64: Delete staff activity	50
Figure 65: Writing feedback.....	51
Figure 66: Update account activity	52
Figure 67: Change password.....	53
Figure 68: Add idea	54
Figure 69: Edit idea.....	55
Figure 70: Delete idea.....	56
Figure 71: Login interface	57
Figure 72: Login with admin account.....	57
Figure 73: After login with admin account.....	58
Figure 74: Category interface	58
Figure 75: Add category	59
Figure 76: Add category information	59
Figure 77: Show category information	60
Figure 78: Delete category.....	60
Figure 79: Department interface	61

Figure 80: Create department interface	61
Figure 81: Add department information	62
Figure 82: Show department information.....	62
Figure 83: Edit department	63
Figure 84: Delete department	63
Figure 85: Staff interface	64
Figure 86: Create staff interface	64
Figure 87: Create staff information	65
Figure 88: Show staff information.....	66
Figure 89: Edit Staff	66
Figure 90: After edit staff	67
Figure 91: Delete staff	67
Figure 92: Post interface	68
Figure 93: Create post.....	68
Figure 94: Add post information	69
Figure 95: Show post information	69
Figure 96: Edit post interface.....	70
Figure 97: After edit post.....	70
Figure 98: Delete post.....	71
Figure 99: QA list interface	71
Figure 100: Staff interface	72
Figure 101: View post and feedback interface	72
Figure 102: Submit idea information.....	73
Figure 103: Idea information interface	73
Figure 104: Idea information interface with admin account.....	74
Figure 105: Comment idea interface	74
Figure 106: After comment and like idea	75
Figure 107: Update profile.....	75
Figure 108: Change password interface.....	76
Figure 109: Burndown chart of sprint 1.....	90
Figure 110: Burndown chart of sprint 2.....	92
Figure 111: Burndown chart of sprint 3.....	94
Figure 112: Burndown chart of sprint 4.....	96
Figure 113: Burndown chart of sprint 5.....	99
Figure 114: Burndown chart of sprint 6.....	101

List of Table

Table 1: Team member + Positions + Commitment point.....	2
Table 2: Roles	2
Table 3: URL	2
Table 4: Test sprint 1	78
Table 5: Test sprint 2	79
Table 6: Test sprint 3	80
Table 7: Test sprint	82
Table 8: Test sprint 5	84
Table 9: Test print 6	85
Table 10: Product backlog	87
Table 11: Sprint 1.....	89
Table 12: Sprint 2.....	91
Table 13: Sprint 3.....	93
Table 14: Sprint 4.....	95
Table 15: Sprint 5.....	97
Table 16: Sprint 6.....	100
Table 17: Daily meeting sprint 1.....	102
Table 18: Daily meeting sprint 2.....	102
Table 19: Daily meeting sprint 3.....	103
Table 20: Daily meeting sprint 4.....	103
Table 21: Daily meeting sprint 5 (1).....	104
Table 22: Daily meeting sprint 5 (2).....	104
Table 23: Daily meeting sprint 6.....	105
Table 24: Daily meeting sprint 6 (2).....	105

1. Introduction

This team report is intended to document the development of a website application based on the given specifications. The report is divided into four main sections detailing the project phases. First, we'll cover creating a database for the site, highlighting key entities and stages from pre-normalization to a third normalization. Second, we will present the user interface design, outlining the three main phases including drawing the site map, developing the prototype, and designing the final interface. The third part will give an overview of the main functions of the website based on a detailed case study of different users, specifically QA, department manager and staff. The testing phase, which is very important to ensure functionality, will be presented, including all the conducted test cases and their results. Finally, we will discuss the Agile Scrum methodology used during the development of this project.

2. Database

2.1. Entity-relationship diagram (ERD)

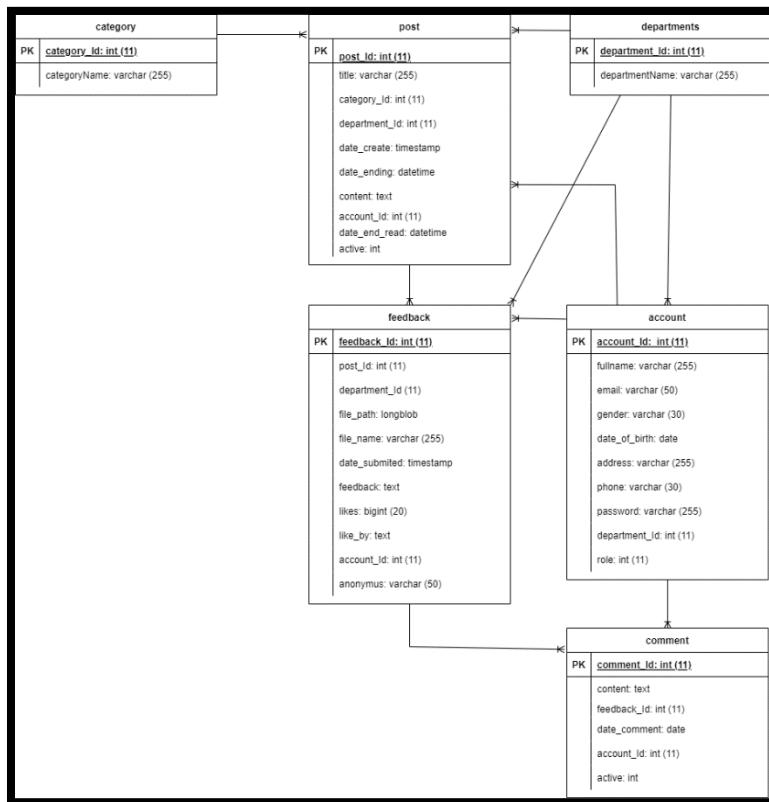


Figure 1: ERD

Explain: The project's database is comprised of six tables, with their respective entities and relationships visually represented in Figure 1. Each entity and its corresponding relationship will be meticulously outlined in detail below.

2.1.1. “Account” entity

account	
PK	account_Id: int (11)
	fullname: varchar (255) email: varchar (50) gender: varchar (30) date_of_birth: date address: varchar (255) phone: varchar (30) password: varchar (255) department_Id: int (11) role: int (11)

Figure 2: “Account’ entity

Explain: The primary objective of creating this entity is to facilitate the efficient management of account user details on a database. The entity comprises ten key attributes, including account identification (account_Id), full name (fullname), email address (email), gender, date of birth (date_of_birth), physical address (address), phone number (phone), password, department identification (department_Id), and role information. Overall, this entity plays a crucial role in simplifying the account information storage process.

2.1.2. “Feedback” entity

feedback	
PK	<u>feedback_Id: int (11)</u>
	post_Id: int (11) department_Id (11) file_path: longblob file_name: varchar (255) date_submited: timestamp feedback: text likes: bigint (20) like_by: text account_Id: int (11) anonymus: varchar (50)

Figure 3: “Feedback” entity

Explain: The primary function of this entity is to facilitate the organization and storage of employee comments on QA posts in a database. It comprises of 11 principal properties that include feedback_Id, post_Id, department_Id, file_path, file_name, date_submited, feedback, likes, like_by, account_Id, and anonymus. These properties serve to streamline the management process of the feedback, providing an efficient means of accessing and handling the data.

2.1.3. “Post” entity

post	
PK	<u>post_Id: int (11)</u>
	title: varchar (255) category_Id: int (11) department_Id: int (11) date_create: timestamp date_ending: datetime content: text account_Id: int (11) date_end_read: datetime active: int

Figure 4: “Post” entity

Explain: This entity plays a crucial role in streamlining the response management process by facilitating the efficient organization and storage of QA posts in the database. It is comprised of eight key attributes, including the post's title, post ID, department ID, category ID, creation date, expiry date, content, and account ID. These attributes work together to provide a seamless means of accessing and processing data, ultimately enhancing the overall efficiency of the response management process.

2.1.4. “Category” entity

category	
PK	category_id: int (11)
	categoryName: varchar (255)

Figure 5: “Category” entity

Explain: This entity serves as a means to streamline the organization of post types and their storage in the database. Its two key attributes, category_id and categoryName, aid in the effective processing of data by simplifying the response management process. This facilitates the efficient access and retrieval of data, contributing to a more streamlined and optimized system.

2.1.5. “Department” entity

departments	
PK	department_id: int (11)
	departmentName: varchar (255)

Figure 6: “Department” entity

Explain: This entity is designed to simplify the management of department types and their storage in a database. It is characterized by two key properties: department_id and departmentName. These properties help to streamline the handling of data by simplifying the process of managing responses. This, in turn, facilitates efficient access to and retrieval of data, leading to a more optimized and streamlined system.

2.1.6. “Comment” entity

comment	
PK	comment_Id: int (11)
	content: text feedback_Id: int (11) date_comment: date account_Id: int (11) active: int

Figure 7: “Comment” entity

Explain: The purpose of this entity is to make the management of comment types in staff’s idea easier and store them in the database. It has five primary attributes: comment_Id, content, feedback_Id, date_comment, and account_Id. These attributes help simplify data processing by making response management more efficient, leading to faster data access and retrieval. As a result, the system becomes more streamlined and optimized.

2.1.7. Relationship between entities

- “account” and “comment” relationship

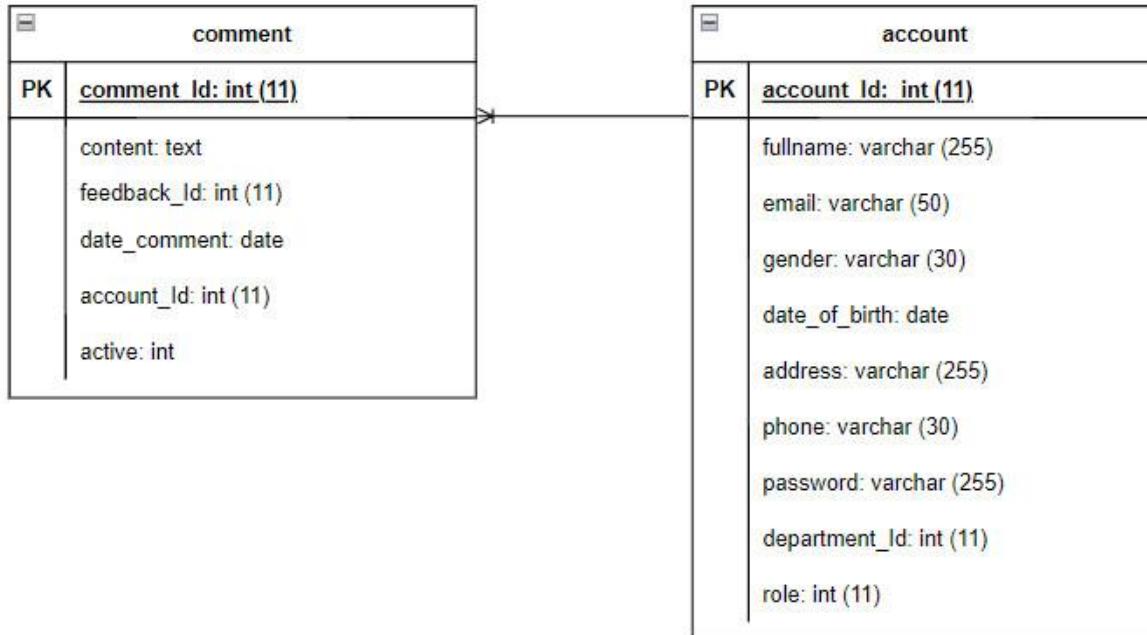


Figure 8: “account” and “comment” relationship

Explain: The `account` table contains information about the user accounts registered or used for the application or website. Meanwhile, the `comments` table contains information about comments posted by users on a certain content or article. On the relational side, the `comments` table is linked to the `account` table via the `account_Id` foreign key, which means that each comment will be linked to the user account that posted or used to log in and post comments. An account (account) can create many comments (comments). Each comment must belong to a specific account.

- “account” and “department” relationship

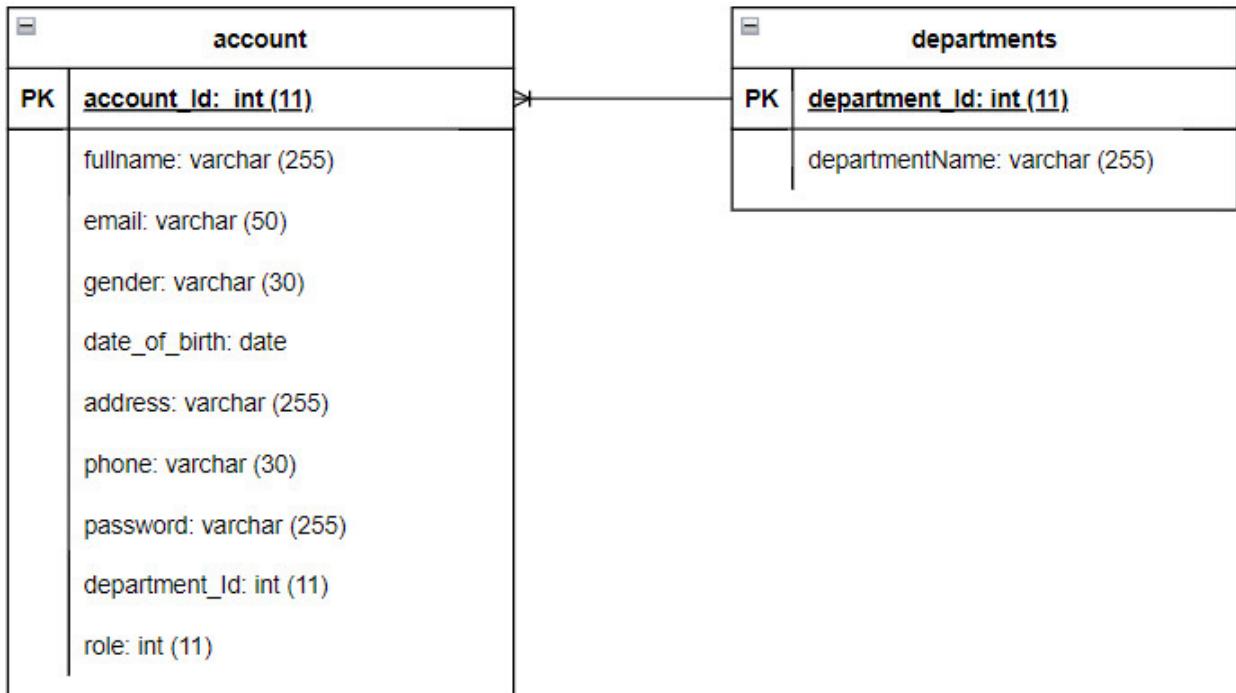


Figure 9: “account” and department relationship

Explain: Tables `accounts` and `departments` have a one-to-many relationship (one-to-many relationship). This means that a department can have many accounts, but one account belongs to only one department. To display this relationship, the table `department` has the primary keyword `department_id`, then the table `accounts` has the field `department` as a foreign keyword referencing the table `department`. Using exceptions allows us to link the records in these two tables together, making querying and data management more convenient.

- “account” and “feedback” relationship

account		feedback	
PK	<u>account_id</u> : int(11)	PK	<u>feedback_id</u> : int(11)
fullname: varchar (255) email: varchar (50) gender: varchar (30) date_of_birth: date address: varchar (255) phone: varchar (30) password: varchar (255) department_id: int (11) role: int (11)		post_id: int (11) department_id (11) file_path: longblob file_name: varchar (255) date_submited: timestamp feedback: text likes: bigint (20) like_by: text account_id: int (11) anonymus: varchar (50)	

Figure 10: “account” and “feedback” relationship

Explain: The account and feedback tables have a one-to-many relationship. This means that each account (account) can have many feedback (feedback), but each response can only belong to 1 account. To show this relationship, the account table has a primary key of id (representing each account), and the feedback table has a primary key of feedback_id (representing each response). However, the feedback table also has an account_id field to store the foreign key of the account table, associated with the account to which the response belongs.

- “account” and “post” relationship

post		account	
PK	<u>post_id</u> : int(11)	PK	<u>account_id</u> : int(11)
title: varchar (255) category_id: int (11) department_id: int (11) date_create: timestamp date_ending: datetime content: text account_id: int (11) date_end_read: datetime active: int		fullname: varchar (255) email: varchar (50) gender: varchar (30) date_of_birth: date address: varchar (255) phone: varchar (30) password: varchar (255) department_id: int (11) role: int (11)	

Figure 11: “account” and “post” relationship

Explain: The relationship between the account and post tables is an entity-association relationship in the database. On a simple database, the account table typically contains information about a user account such as name, email address, password, and other fields of information. Meanwhile, the posts table contains information about the posts on the website, such as title, content, date of publication, poster and other fields. The relationship between these two tables is one-to-many, which means that each user account can post multiple posts on the site. Each post is posted by a single account and each account can post multiple posts. Therefore, in order to display the posts on the site, the information from both tables must be combined using a common primary key. In this case, information about the user account will be stored in the account table and information about the posts will be stored in the posts table. Each post will be identified with a unique number, and all of these posts will contain an ID value corresponding to the user account it was posted by.

- “feedback” and “post” relationship

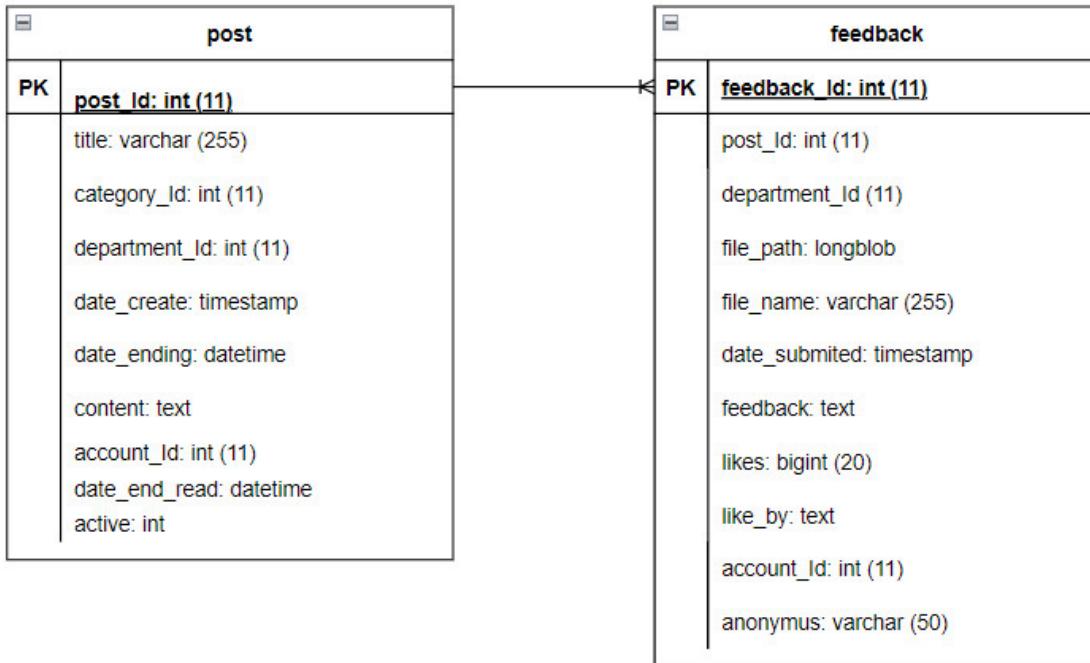


Figure 12: “feedback” and “post” relationship

Explain: The relationship between the Feedback and Post tables is a one-to-many relationship. This means that a post can have multiple feedback votes, but a vote can only belong to one post. In the database, to establish this relationship, the Feedback table will have a foreign key that refers to the primary key of the Post table. This allows the feedback sheets to be linked to the corresponding article. This relationship allows to manage user responses to articles, and to track and display user responses on specific articles.

- “feedback” and “department” relationship

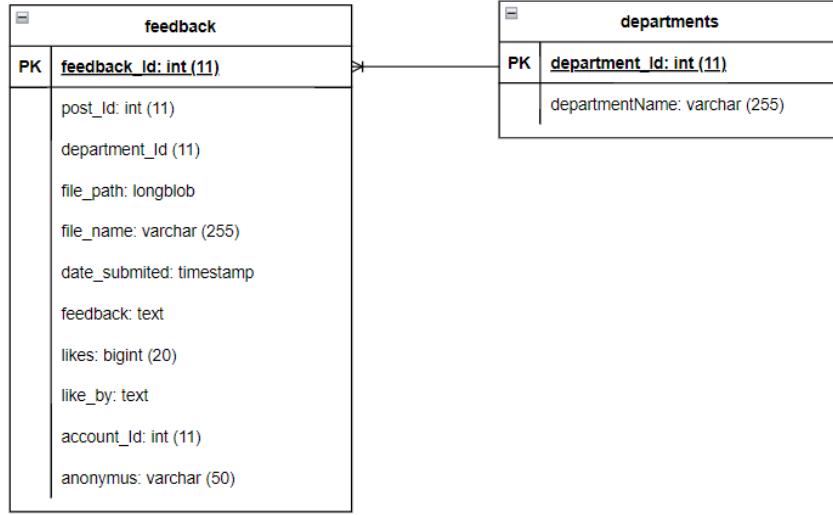


Figure 13: “feedback” and “department” relationship

Explain: The relationship between the two response tables and the department is a many (one-to-many) relationships, where a department can have many responses. This relationship is represented by a foreign key in the response table, which points to the primary key of the parts table that identifies the data unit of the response. A department (department) can have many different responses (responses) from many users. Each response is in the data unit of a particular department. The relationship between the two tables can be represented via foreign key (foreign key) **department_id** in response table, pointer to primary key (primary key) id of table **department**.

- “feedback” and “comment” relationship

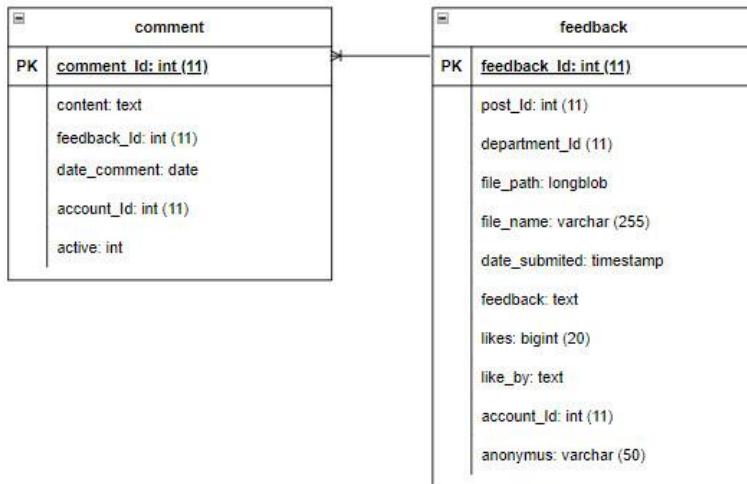


Figure 14: “feedback” and “comment” relationship

Explain: The “feedback” and “comments” tables have a one-to-many relationship. That is, a version in “feedback” can have many comments, but that one comment can only correspond

to one "feedback". Specifically, the "feedback" table contains information about user feedback, such as feedback_Id, feedback, creation date and creator. Meanwhile, the "comments" table contains user comments on a particular response, including the id_comment, body, creation date, creator, and feedback_Id associated with the corresponding response.

- “post” and “department” relationship



Figure 15: “post” and “department” relationship

Explain: The relationship between the post and department tables can be a "one-to-many" relationship. This means that each post only belongs to a certain department. However, each department can contain multiple posts. In the database, to represent this relationship, we can add a field "department_Id" to the table "posts", representing the department to which the post belongs. This field will act as the foreign key referring to the primary key of the "department" table, allowing us to easily retrieve related information between the two tables.

- “post” and “category” relationship



Figure 16: “post” and “category” relationship

Explain: The relationship between the post and category tables can be a "one-to-many" relationship. This means that each post has only a certain type. However, each category can belong to multiple posts. In the database, to represent this relationship, we can add a field "department_id" to the table "posts", representing the department to which the post belongs. This field will act as a foreign key that refers to the primary key of the "department" table, allowing us to easily retrieve related information between the two tables.

2.2. Database design

- “account” entity

The screenshot shows the MySQL database interface with the following details:

- Query results: Showing rows 0 - 1 (total, Query took 0.0004 seconds.)
- SQL query: SELECT * FROM `account`
- Table structure: account (account_id, fullname, email, gender, date_of_birth, address, phone, password, department_id, role)
- Data rows:

	account_id	fullname	email	gender	date_of_birth	address	phone	password	department_id	role
<input type="checkbox"/>	1	Vinh	admin@gmail.com	Other	2023-03-27	TP.Can Tho Ap so tai, xa thanh phu, huyen cao nuoc, tinh ca m...	021352876	e10adc3949ba59abbe56e057f20f883e	1	1
<input type="checkbox"/>	2	Tran Quan Vinh	staff@gmail.com	Other	2023-04-03		0914593431	e10adc3949ba59abbe56e057f20f883e	1	0

Figure 17: “account” entity

- “category” entity

		category_id	categoryName
<input type="checkbox"/>	Edit Copy Delete	1	Event of school
<input type="checkbox"/>	Edit Copy Delete	4	Club
<input type="checkbox"/>	Edit Copy Delete	5	Workshop

Figure 18: “category” entity

- “comment” entity

		comment_id	content	feedback_id	date_comment	account_id	active
<input type="checkbox"/>	Edit Copy Delete	18	test1	26	2023-04-10	1	0
<input type="checkbox"/>	Edit Copy Delete	20	test1	27	2023-04-10	1	0

Figure 19: “comment” entity

- “department” entity

		department_id	departmentName
<input type="checkbox"/>	Edit Copy Delete	1	Computer research
<input type="checkbox"/>	Edit Copy Delete	2	Bussiness

Figure 20: “department” entity

- “feedback” entity

Showing rows 0 - 0 (1 total). Query took 0.0005 seconds.											
SELECT * FROM `feedback`											
<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]											
UPDATE `feedback` SET `feedback` = 'feedback test' WHERE `feedback`.`feedback_Id` = 32;											
feedback_Id	post_Id	department_Id	file_path	file_name	date_submitted	feedback	likes	liked_by	account_Id	anonymous	
32	30	1	[BLOB - 47 B]	milewska_the e-commerce_60-2019 (1).pdf	2023-04-12 13:19:13	feedback test	0		1 anonymous		

Figure 21: “feedback” entity

- “post” entity

SELECT * FROM `post`											
<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]											
post_Id	title	category_Id	department_Id	date_create	date_ending	date_end_read	content	account_Id	active		
32	asasas	1	1	2023-04-14 14:43:27	2023-04-15 14:42:00	2023-04-15 14:42:00	aaaaaa	1	0		

Figure 22: “post” entity

3. Interface design

3.1. Site map system

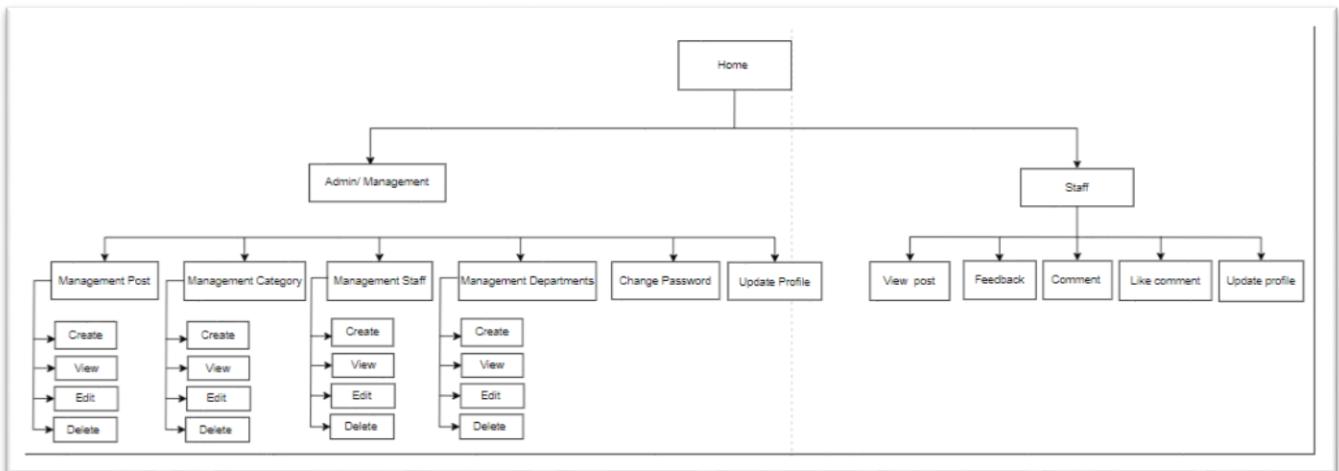
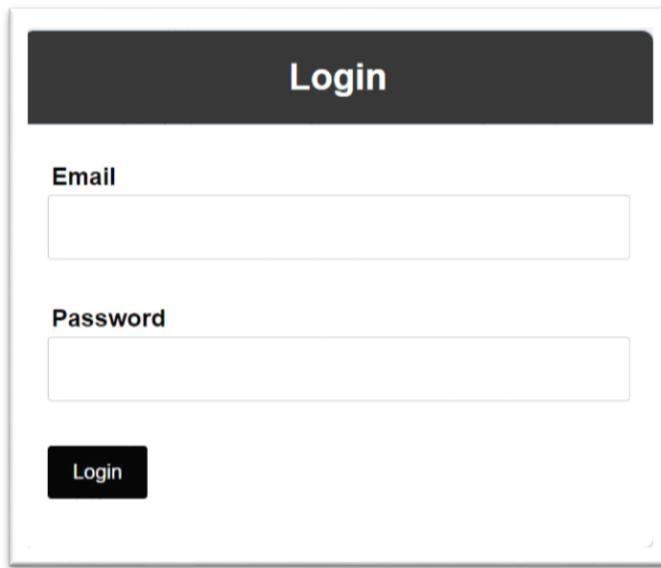


Figure 23: Site map system

The website system has been scientifically designed to ensure that the management becomes easier and more convenient than ever. The homepage of the website will be the Home page, where users can view basic information and learn about the website. However, to manage the website professionally, the system has been decentralized into two main objects, Admin and Staff. Admin will be granted access to the main management sections including: article management, staff management, department management and category management. For each management section, users will be provided with basic functions such as adding, editing and deleting to easily manage the content and information of the website. In addition, in the admin management section, there are additional functions to manage your own profile such as changing passwords and updating personal information. Meanwhile, Staff will be allowed to access and use basic functions including viewing posts, giving feedback, leaving comments, liking comments and updating personal information. These functions will be clearly and easily divided so that each user can use and access easily quickly and conveniently. At the same time, the system will provide users with detailed help and instructions to help them use the website most effectively.

3.2. Final designs

- Login



The image shows a simple login interface. It features a dark grey header bar with the word "Login" in white. Below this, there are two input fields: one for "Email" and one for "Password", both with placeholder text. At the bottom is a dark grey "Login" button.

Figure 24: Login

This is the login interface of the system, in this page will ask the user of his account to log in to the system.

- Logged in with admin account

The screenshot shows the 'Feedback' section of a web application. At the top, there's a navigation bar with links for Feedback, Post, Category, Staff, Department, QA List, and a greeting 'Hi, admin@gmail.com'. Below the navigation is a search bar labeled 'Search Feedback:' with a placeholder 'Enter feedback...'. A table displays a single row of data:

Post Title	Feedback	Category	Department	Date Submitted	Comment	Date end comment
organize an event for the holiday of 8/3	Feedback	Event of school	Computer research	2023-04-15 15:46:00	2023-04-15 15:46:00	Comment

At the bottom of the page, there are social media sharing icons (Facebook, Twitter, Google+) and a footer link '©2019 Idea Management | Privacy | Terms of Use'.

Figure 25: Login with admin account

This is the interface after the admin has successfully logged into the system

- Post interface

The screenshot shows the 'Post' section of the web application. The top navigation bar includes links for Feedback, Post, Category, Staff, Department, QA List, and a greeting 'Hi, admin@gmail.com'. Below the navigation is a search bar labeled 'Search Post:' with a placeholder 'Enter post...'. A table displays a single row of data:

Title	Department	Category	Create Date	End Date	Manage
organize an event for the holiday of 8/3	Computer research	Event of school	2023-04-14 15:46:29	2023-04-15 15:46:00	Edit Delete

At the bottom of the page, there are social media sharing icons (Facebook, Twitter, Google+) and a footer link '©2019 Idea Management | Privacy | Terms of Use'.

Figure 26: Post interface

This is the Post interface, which in this page belongs to the administrator's control. In this section, the administrator will perform tasks such as adding, editing, and deleting posts.

- **Create post interface**

The screenshot shows a 'Create Post' form. At the top, there are navigation links: Feedback, Post, Category, Staff, Department, QA List, and a user greeting 'Hi, admin@gmail.com'. The main form has the following fields:

- Title: A text input field.
- Category: A dropdown menu labeled 'Choose Category'.
- Department: A dropdown menu labeled 'Choose Department'.
- End submit: A date input field showing 'mm/dd/yyyy'.
- End comment: A date input field showing 'mm/dd/yyyy'.
- Content: A large text area for the post content.

A 'Post' button is located at the bottom right of the form.

Figure 27: Create post interface

This is the interface to add a new post, in this page will display the necessary information to create a new post.

- **Edit post interface**

The screenshot shows an 'Edit Post' form. The title is 'Edit Post'. The form contains the following fields:

- Title: A text input field containing 'Organize an event for the holiday of 8/3'.
- Category: A dropdown menu showing 'Event'.
- Department: A dropdown menu showing 'Computer'.
- End submit: A date input field showing '04/27/2023 11:59 PM'.
- End comment: A date input field showing '04/27/2023 11:59 PM'.
- Content: A text area containing 'Organize an event for the holiday of 8/3'.

At the bottom are two buttons: 'Post' and 'Cancel'.

Figure 28: Edit post interface

This is the post editing interface, in this page will display the information of a post so that the admin wants to edit any information of that post.

- **Delete post interface**

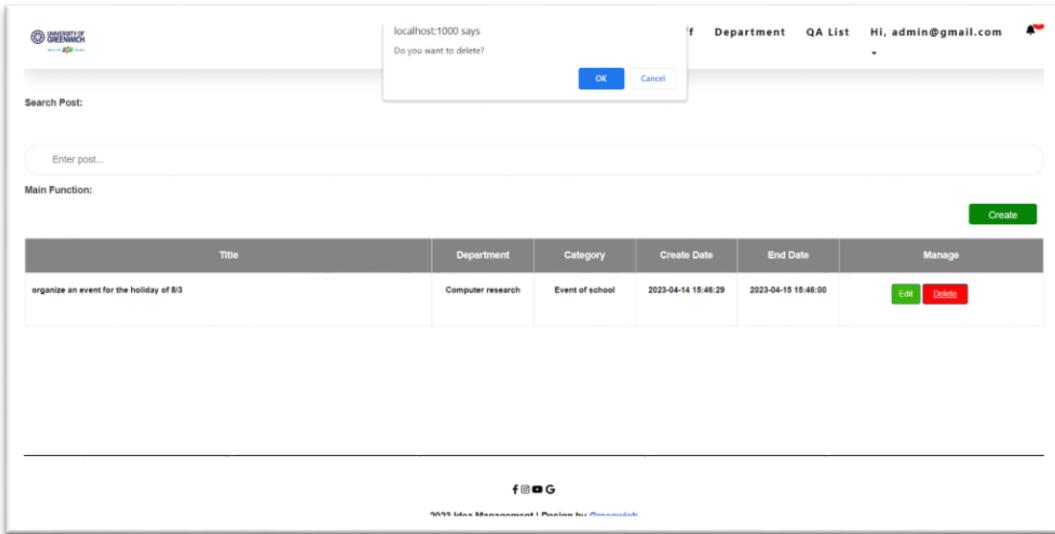


Figure 29: Delete post interface

This is the interface to delete a post, in this part the system will display a message to confirm that the administrator wants to really delete the post or not. If you choose Ok, the system will delete it, if you choose Cancel, you will return to the Post page

- **Category interface**

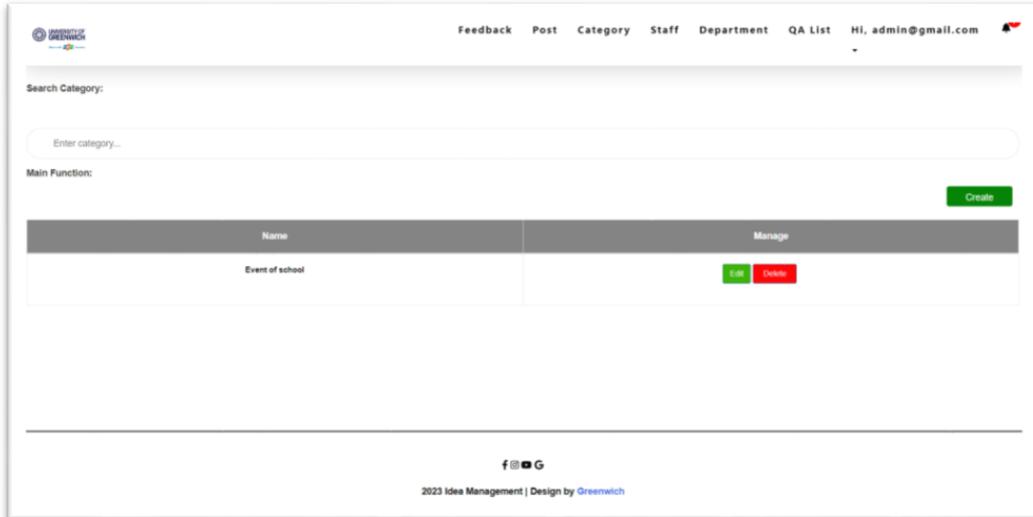


Figure 30: Category interface

This is the Category interface, which in this page belongs to the administrator's control. In this section, the administrator will perform tasks such as adding and deleting category.

- Add category interface

The screenshot shows a modal window titled "Add Category". Inside, there is a label "Category Name:" above an empty input field. At the bottom of the modal are two buttons: "Add Category" on the left and "Cancel" on the right.

This is the interface to add a new category, in this page will display the necessary information to create a new category

Figure 31: Add category

- Delete category interface

The screenshot shows a web application interface. At the top, there is a navigation bar with links for "Department", "QA List", and a user session "Hi, admin@gmail.com". Below the navigation, a modal dialog box is open, asking "localhost:1000 says Do you want to delete?". Below the dialog, there is a search bar labeled "Search Category:" and a text input field with placeholder "Enter category...". Underneath, there is a table with a single row. The table has columns for "Name" (containing "Event of school") and "Manage" (containing "Edit" and "Delete" buttons). On the far right of the table, there is a green "Create" button. At the bottom of the page, there is footer text: "f G" and "2023 Idea Management | Passion for Progress".

Figure 32: Delete category

This is the interface to delete a category, in this part the system will display a message to confirm that the administrator wants to really delete the category or not. If you choose Ok, the system will delete it, if you choose Cancel, you will return to the Category page.

- Staff interface

The screenshot shows a web-based staff management system. At the top, there is a navigation bar with links for Feedback, Post, Category, Staff, Department, QA List, and a greeting message "Hi, admin@gmail.com". Below the navigation bar is a search bar labeled "Search Staff:" with a placeholder "Enter staff name...". Underneath the search bar is a table displaying staff information. The table has columns for Full Name, Gender, Date of birth, Address, Phone number, Email, and Manage. A single row is visible, showing data for "Tran Quan Vinh" (Gender: Other, DOB: 2023-04-03, Address: Ap so tai, xa thanh phu, huyen cao nuoc, tinh ca mau, Phone number: 0914593431, Email: staff@gmail.com). To the right of the row are "Edit" and "Delete" buttons. Above the table is a "Create" button. At the bottom of the page are social media sharing icons (Facebook, Twitter, LinkedIn) and a "G" icon.

Figure 33: Staff interface

This is the Staff interface, which in this page belongs to the administrator's control. In this section, the administrator will perform tasks such as adding, editing, and deleting staff.

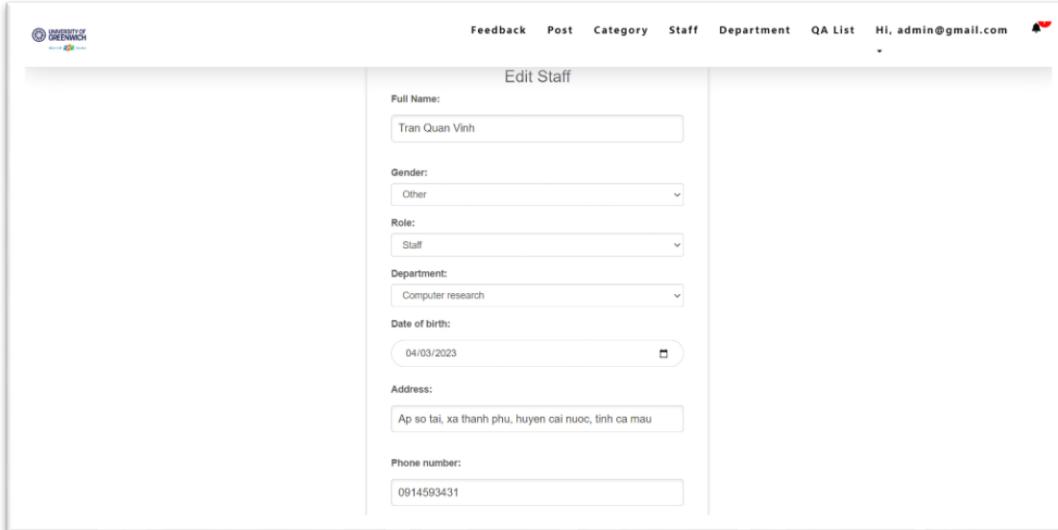
- Add staff interface

The screenshot shows a modal dialog titled "CREATE STAFF". The form contains the following fields: "Fullname" (input field), "Gender" (dropdown menu with placeholder "-- Choose Gender --"), "Department" (dropdown menu with placeholder "-- Choose Department --"), "Date of birth" (input field with placeholder "mm/dd/yyyy" and a clear button), "Address" (input field), "Phone number" (input field), and "Email" (input field).

Figure 34: Add staff

This is the interface to add a new staff, in this page will display the necessary information to create a new staff

- **Edit staff interface**

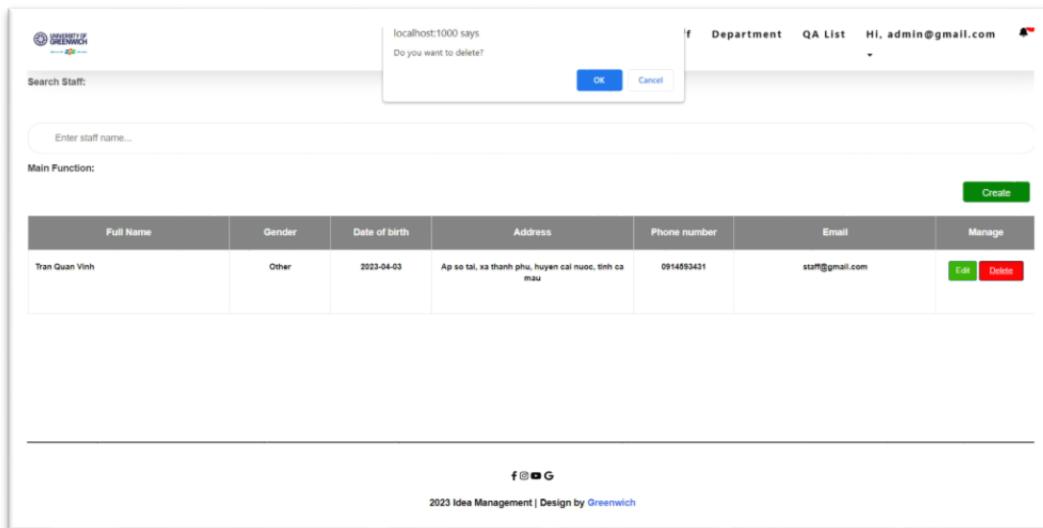


The screenshot shows the 'Edit Staff' form. At the top, there's a navigation bar with links for Feedback, Post, Category, Staff, Department, QA List, and a user greeting 'Hi, admin@gmail.com'. Below the navigation is a title 'Edit Staff'. The form contains several input fields: 'Full Name' (Tran Quan Vinh), 'Gender' (Other), 'Role' (Staff), 'Department' (Computer research), 'Date of birth' (04/03/2023), 'Address' (Ap so tai, xa thanh phu, huyen cai nuoc, tinh ca mau), and 'Phone number' (0914593431). There are also 'Create' and 'Cancel' buttons at the bottom.

Figure 35: Edit staff interface

This is the staff editing interface, in this page will display the information of a post so that the admin wants to edit any information of that staff.

- **Delete staff interface**



The screenshot shows the 'Delete Staff' interface. At the top, there's a navigation bar with links for Feedback, Post, Category, Staff, Department, QA List, and a user greeting 'Hi, admin@gmail.com'. A modal dialog box is open, displaying the message 'localhost:1000 says' and 'Do you want to delete?'. In the background, there's a search bar labeled 'Search Staff:' and a table titled 'Main Function:' showing staff information. The table has columns for Full Name, Gender, Date of birth, Address, Phone number, Email, and Manage. One row is selected, showing 'Tran Quan Vinh' in the Full Name column. At the bottom of the page, there are social media icons (Facebook, Instagram, Google+) and a footer note '2023 Idea Management | Design by Greenwich'.

Figure 36: Delete staff interface

This is the interface to delete a staff, in this part the system will display a message to confirm that the administrator wants to really delete the staff or not. If you choose Ok, the system will delete it, if you choose Cancel, you will return to the Staff page

- **Department interface**

The screenshot shows a web-based administrative interface for managing departments. At the top, there is a navigation bar with links for Feedback, Post, Category, Staff, Department, QA List, and a greeting 'Hi, admin@gmail.com'. Below the navigation, there is a search bar labeled 'Search Department:' with a placeholder 'Enter department...'. A section titled 'Main Function:' contains a 'Create' button. The main content area displays a table with one row of data:

No.	Name	Manage
1	Computer research	Edit Delete

At the bottom of the page, there are social media sharing icons (Facebook, Twitter, Google+) and a copyright notice: '©2019 Idea Management | Please see [Privacy Policy](#)'.

Figure 37: Department interface

This is the Department interface, which in this page belongs to the administrator's control. In this section, the administrator will perform tasks such as adding and deleting department.

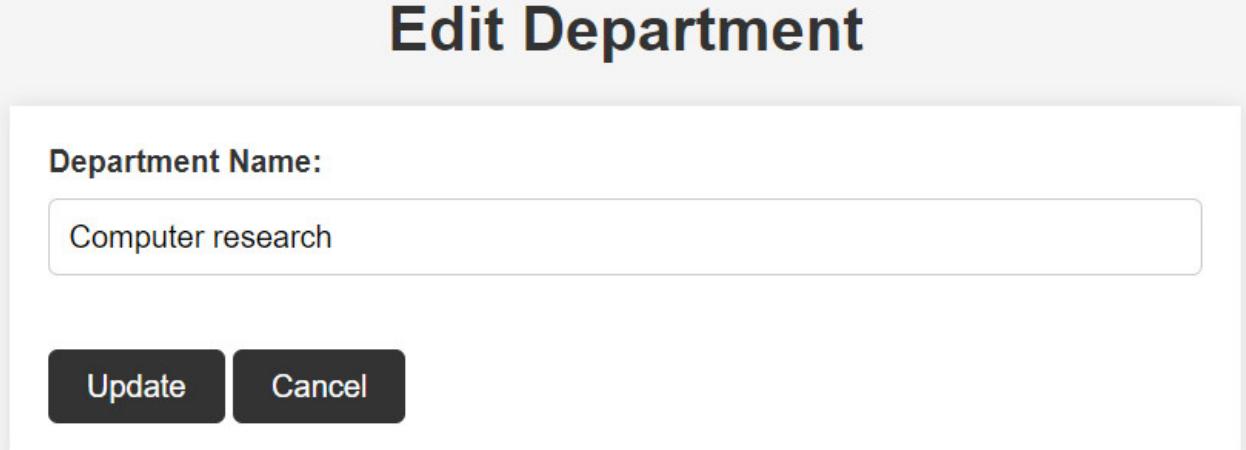
- **Add department interface**

The screenshot shows a modal dialog box titled 'Add Department'. Inside the dialog, there is a form field labeled 'Department Name:' with an empty input box. At the bottom of the dialog are two buttons: 'Add Department' and 'Cancel'.

Figure 38: Add department

This is the interface to add a new department, in this page will display the necessary information to create a new department

- Edit department interface

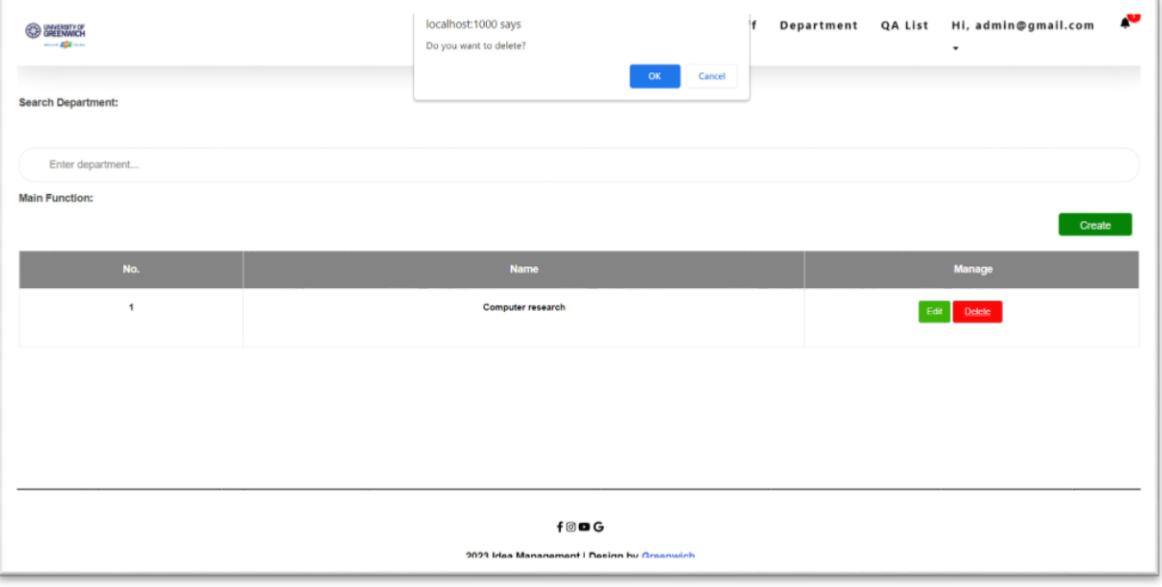


The screenshot shows a modal window titled "Edit Department". Inside, there is a label "Department Name:" followed by a text input field containing "Computer research". At the bottom are two buttons: "Update" and "Cancel".

Figure 39: Edit department

This is the department editing interface, in this page will display the information of a department so that the admin wants to edit any information of that department.

- Delete department interface



The screenshot shows a web page with a search bar for "Search Department" and a main table listing a department. The table has columns for "No.", "Name", and "Manage". The first row shows "1" in the No. column and "Computer research" in the Name column. In the Manage column, there are "Edit" and "Delete" buttons. A modal dialog box is open in the top right, asking "Do you want to delete?" with "OK" and "Cancel" buttons. The page footer includes social media icons and a copyright notice: "©2023 Idea Management | Design by Greenwich".

No.	Name	Manage
1	Computer research	Edit Delete

Figure 40: Delete department interface

This is the interface to delete a department, in this part the system will display a message to confirm that the administrator wants to really delete the department or not. If you choose Ok, the system will delete it, if you choose Cancel, you will return to the Department page

- QA list interface

The screenshot shows a web application interface for managing accounts. At the top, there is a navigation bar with links: Feedback, Post, Category, Staff, Department, QA List, and Hi, admin@gmail.com. Below the navigation bar is a search bar labeled "Search Department:" with a placeholder "Enter department name...". A table below the search bar displays account information for a user named "Vinh". The columns in the table are: Full Name, Gender, Date of birth, Address, Phone number, Email, and Manage. The "Manage" column contains two buttons: "Edit" (green) and "Delete" (red). The table row for "Vinh" has the following values: Vinh, Other, 2023-03-27, TP.Can Tho, 021352876, admin@gmail.com. At the bottom of the page, there are social media sharing icons (Facebook, Twitter, LinkedIn) and a copyright notice: "2023 Idea Management | Design by Greenwich".

Figure 41: QA list interface

This is the QA list interface, which in this page belongs to the administrator's control. In this section, the administrator will perform tasks such as editing, and deleting account.

- Update profile interface

The screenshot shows a "Update Profile" form. The form fields are: Full Name (Vinh), Gender (Other), Date of birth (03/27/2023), Address (TP.Can Tho), and Phone number (021352876). At the bottom of the form is a large "Add" button. The background of the form is white, and the text is black. The overall design is clean and modern.

Figure 42: Update profile interface

This is the update profile interface, in this page will display the information of account so that the user can edit any information of that account.

- **Change password interface**

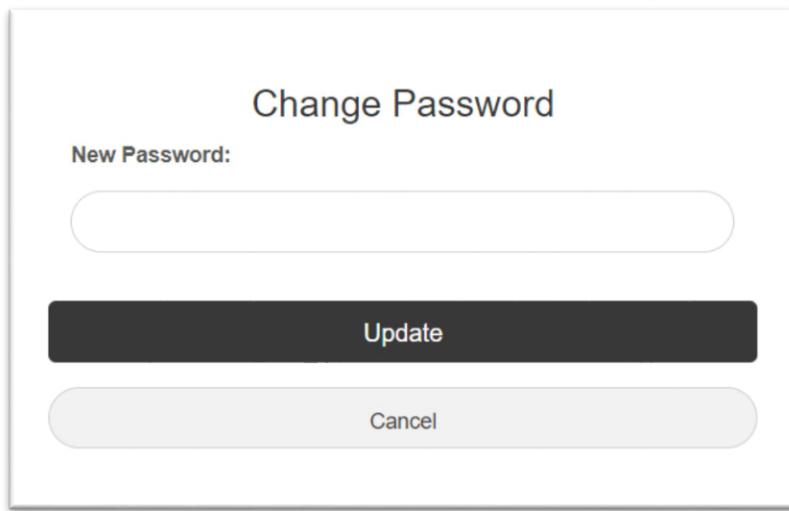


Figure 43: Change password interface

This is the change password of user account interface, in this page will display password of a account so that the user wants to change password of that account.

- **Interface when logged in with staff account**

The screenshot shows a web application interface for a staff account. At the top right, there are links for "Feedback" and "Hi, staff@gmail.com". Below the header is a search bar labeled "Search Feedback:" with the placeholder "Enter feedback...". A table displays a single row of feedback data:

Post Title	Feedback	Category	Department	Day Submitted	Comment	Date end comment
organize an event for the holiday of 8/3	Feedback	Event of school	Computer research	2023-04-15 15:46:00	2023-04-15 15:46:00	Comment

At the bottom of the page, there are social media sharing icons: Facebook, Twitter, LinkedIn, and Google+. The URL in the address bar is "http://www.english-test.net/feedback/staff/".

Figure 44: Interface when logged in with staff account

This is the interface after the user has successfully logged into the system, here is the staff account

- View post and feedback interface

The screenshot shows a web page with the University of Greenwich logo at the top left. At the top right, there are links for 'Feedback' and 'Hi, staff@gmail.com'. Below the header, there is a form for viewing a post. The 'Title' field contains 'organize an event for the holiday of 8/3'. The 'Content' field also contains 'organize an event for the holiday of 8/3'. A 'Submit Idea' button is located below the content area. At the bottom of the page, there are social sharing icons (Facebook, Twitter, LinkedIn, Google+) and a footer note: '2023 Idea Management | Design by Greenwich'.

Figure 45: View post and feedback interface

This is the interface after the user clicks the Feedback button.

- Submit idea interface

The screenshot shows the 'Submit idea interface'. It features a 'Content' field with the text 'organize an event for the holiday of 8/3'. Below it is a 'Title' field containing 'organize an event for the holiday of 8/3'. There is a file upload input field with 'No file chosen' and a 'Choose File' button. A dropdown menu is set to 'Anonymous'. A large 'Message' input field is present. At the bottom, a prominent black 'Submit' button is centered.

Figure 46: Submit idea interface

This is the interface after the user clicks on the Submit idea button.

- List feedback interface

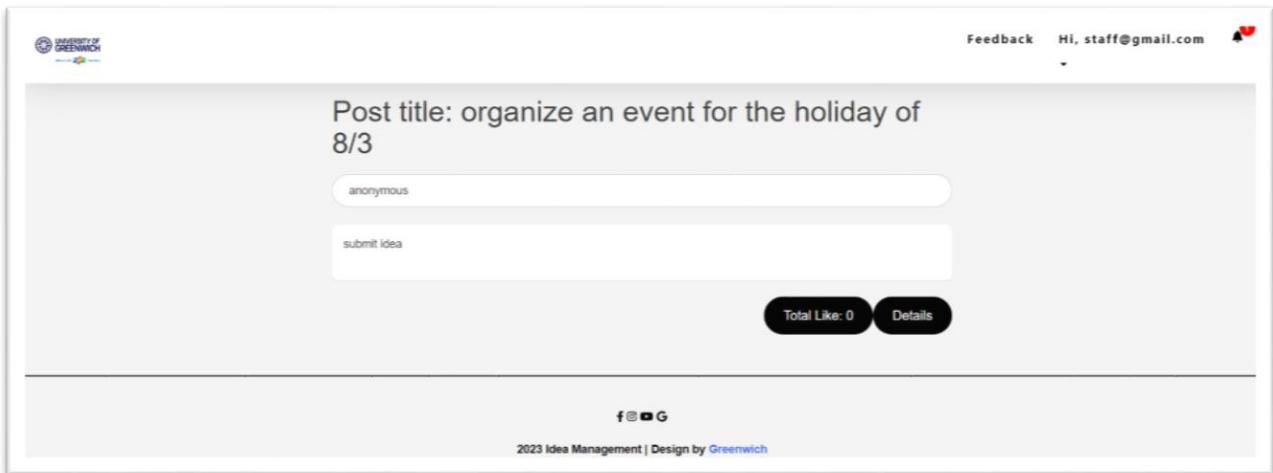


Figure 47: List feedback interface

Here is the following interface showing the list of responses.

- Comment and like interface

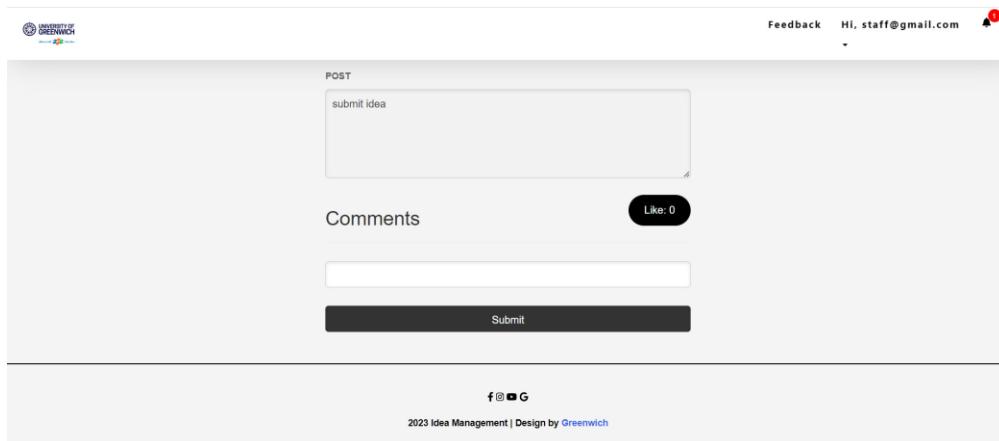


Figure 48: Comment and like interface

This is a command and like interface, this interface you will use to enter comment information in the empty boxes and click Submit to save the comment information or click like button to like this idea.

4. Functionality

4.1. Use case diagram

4.1.1. Admin

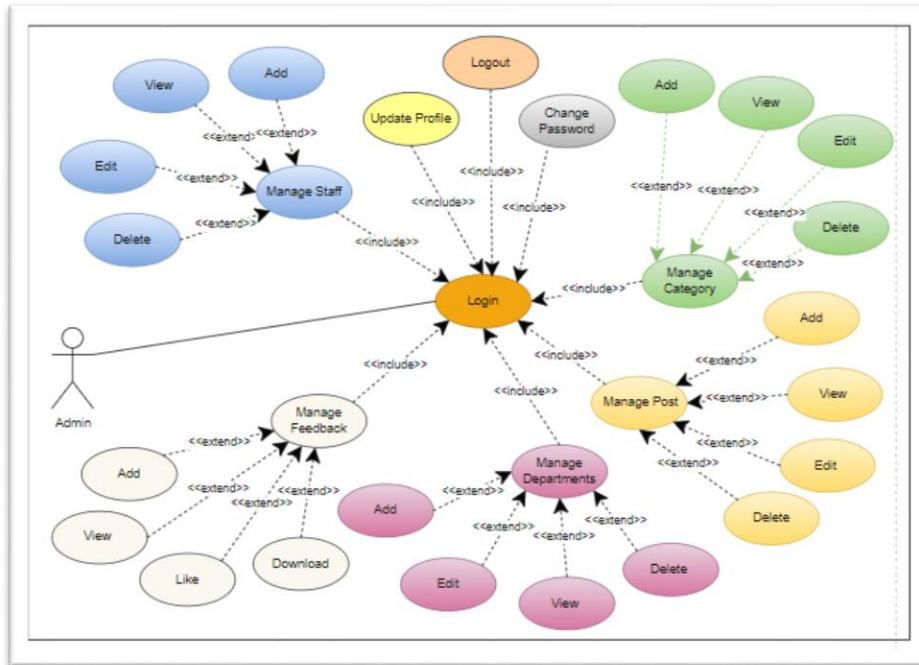


Figure 49: Usecase diagram (Admin)

Explain: In the Use case diagram Admin, users will be provided with many functions to manage and control the entire system of the website. When accessing the login section, users can use their account and password to log in to the system.

If the login information is correct, the system will move to the Manage Staff section to allow the user to perform staff management related tasks. Inside this section, users will have the functions of adding, editing, deleting, viewing to easily manage information about employees working in the company

Besides, the Manage Department section is designed to manage departments in the company. Users will also have functions to add, edit, delete, view to manage information about departments. These functions will help users manage the most easily and accurately.

The Manage Categories section is designed to manage the categories used in the website. Users will have the functions of adding, editing, deleting, viewing to be able to manage information about that category. This will make the management and classification of information on the website easier than ever

The Manage Post section will help users manage articles on the website. Users will have the functions of adding, editing, deleting, viewing to easily manage and display better articles on the website.

Besides, about the profile, users can perform the functions of updating personal information, changing passwords, and logging out. This helps users to keep their personal information safe and secure.

4.1.2. Staff

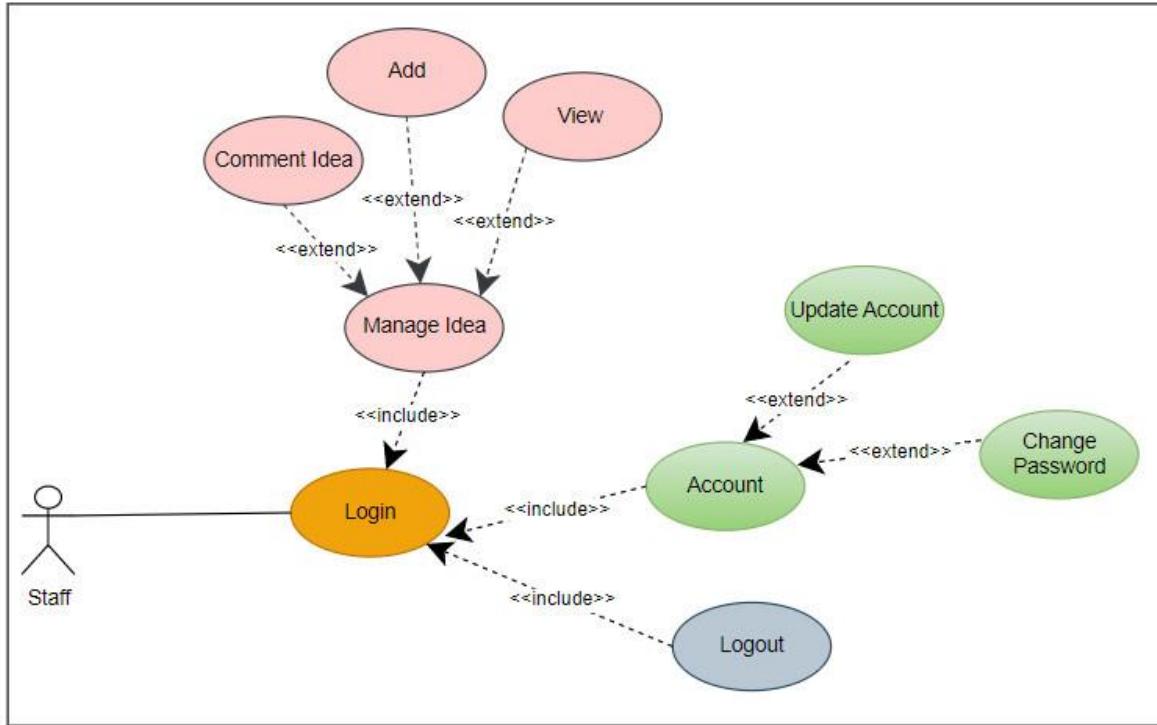


Figure 50: Usecase diagram (Staff)

Explain: In the Use case diagram Staff, users will be provided with a comment management section with many functions so that users can perform tasks related to comments and feedback from customers. In the comment management section, Staff will be provided with the functions of adding, viewing and writing comments to easily manage and display the comment information. Besides, Staff can also track and respond to customer feedback quickly and conveniently. In addition, the profile management section allows users to update personal information, change passwords. Employees can update their personal information to keep it safe and secure. Changing the password allows users to enhance the security of their accounts on the system.

4.2. Activity diagram

4.2.1. Admin

- Update Account

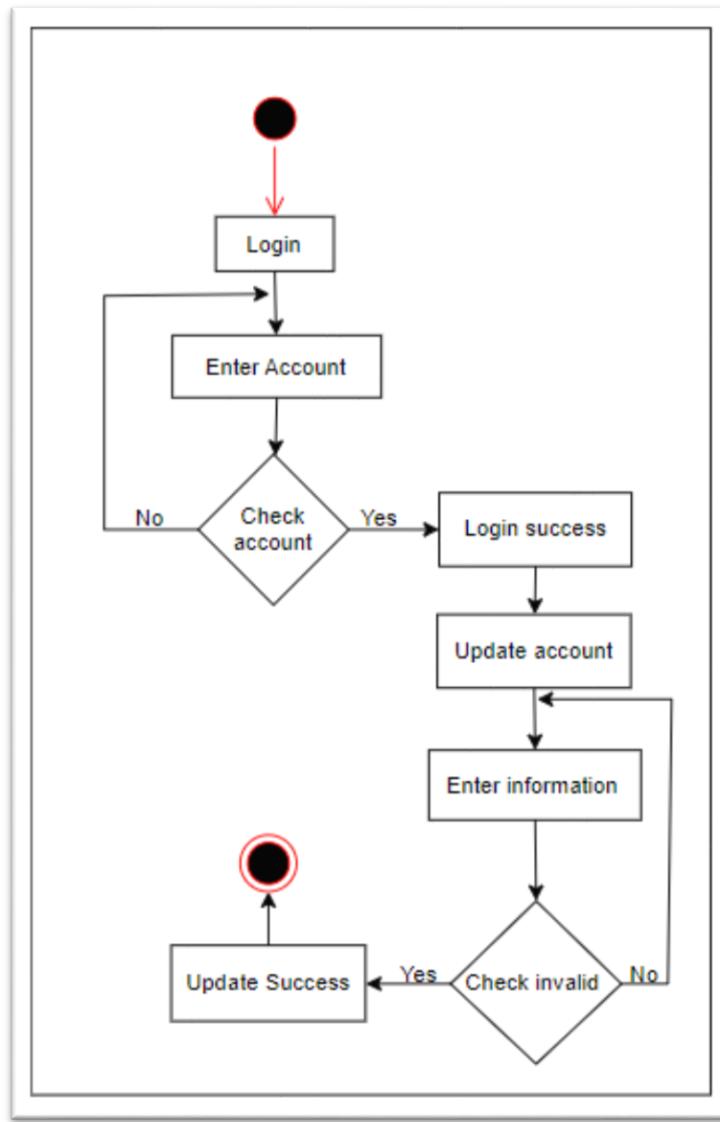


Figure 51: Update account activity

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the administrator's management system. Next, the admin needs to choose to update the account, enter the information to be edited, at this time the system will check the admin information just entered the system, if ok, the admin's information account will be updated, if not, the system will update. The system will not update and let the admin re-enter the information.

- Change Password

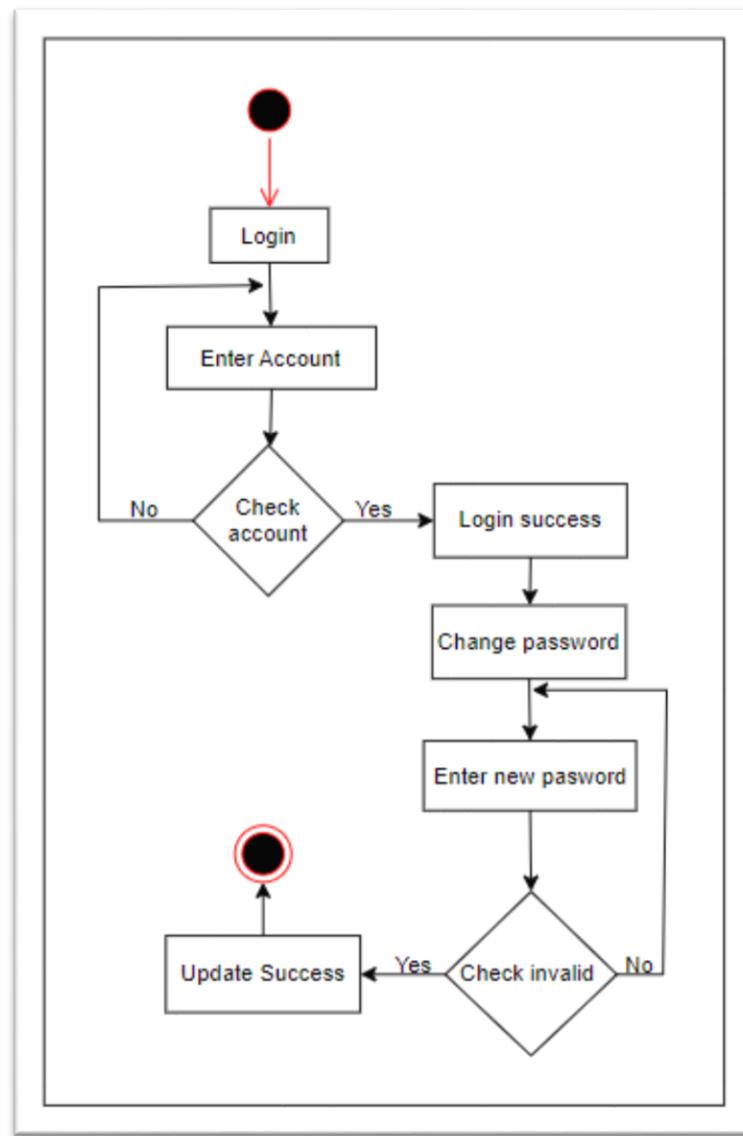


Figure 52: Change password activity

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the administrator's management system. Next admin needs to change his password, next will be to enter a new password, this time the system will check that admin's password will be updated, otherwise the system will not update and let admin re-enter information

- Add Department

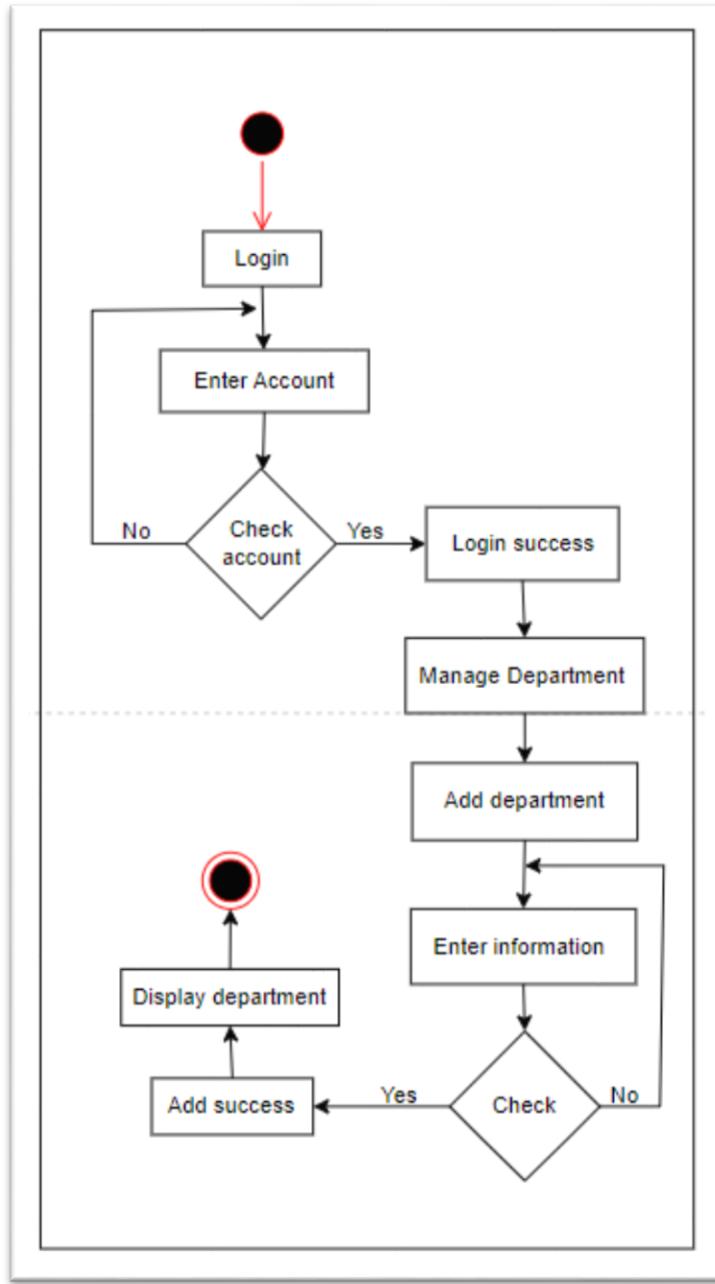


Figure 53: Add department activity

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the administrator's management system. Next, the administrator needs to manage the Department of the system. In this section, the administrator wants to add a new department, then choose to add new, then enter the department information into the system and add it, now the system will check if the

department's information is correct, it will add the information to the database. data, message display and department display have just been added.

- **Edit Department**

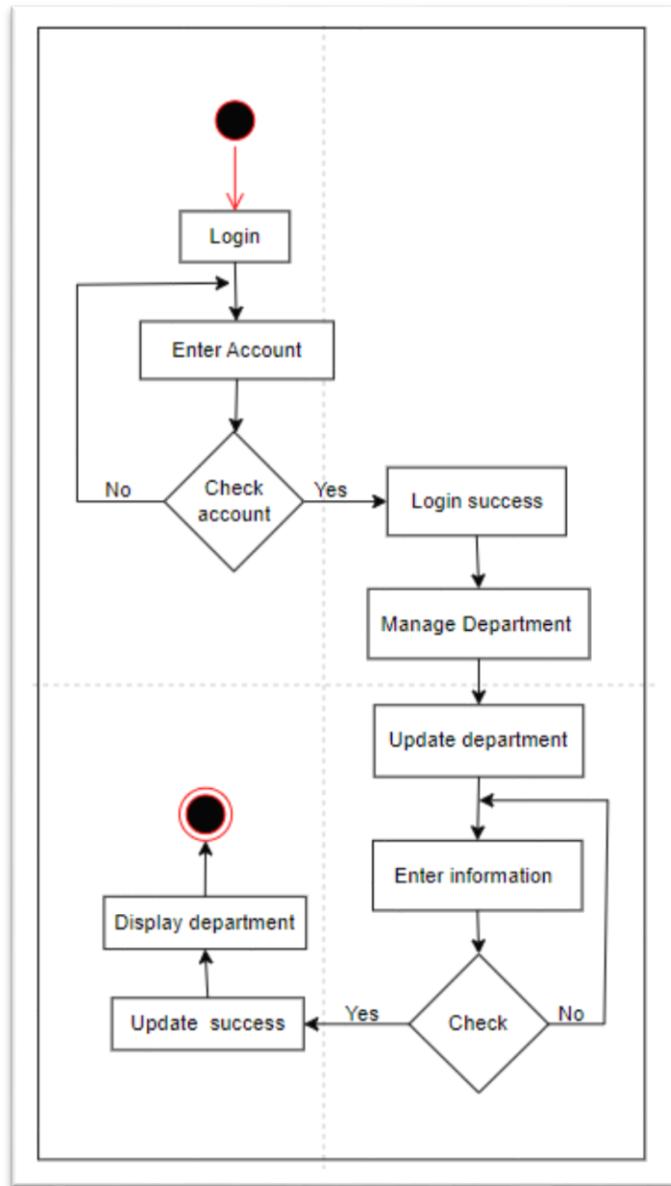


Figure 54: Edit department activity

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the administrator's management system. Next, the administrator needs to manage the Department of the system. In this section, the administrator wants to edit a department, the administrator selects that department. Next will

be to enter the information you need to edit and update it, this time the system will check if the department information has just been properly edited, it will be updated. The data will be updated in the database. Whether, then will display the department that has just been edited. If the system does not check properly, the administrator will re-enter the department information and not update the data.

- **Delete Department**

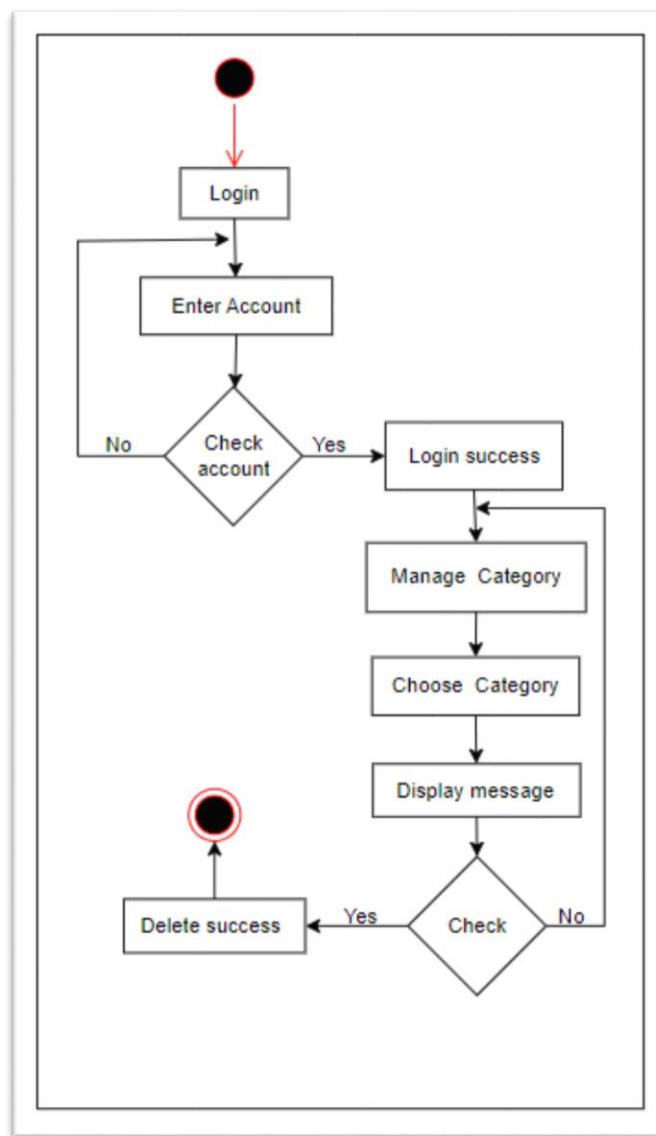


Figure 55: Delete department activity

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the administrator's management system. Next, the administrator needs to manage the Department of the system. In this section, the

administrator wants to delete a department, the administrator selects that department. Then the system will display a message to make sure the administrator wants to delete it or not. If yes is selected, the system will check and delete that department information from the database, if not, the system will return to the department management page.

- **Add Post**

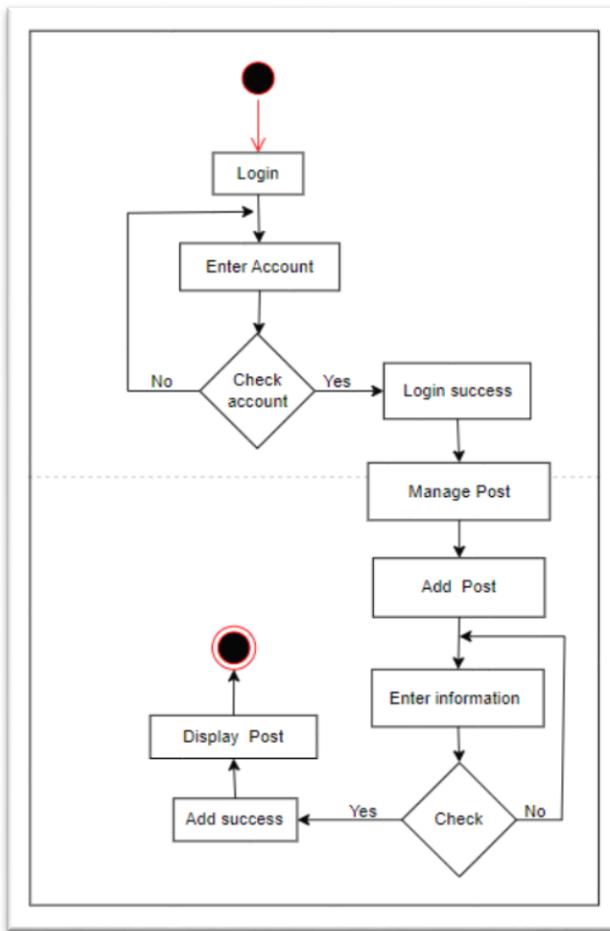


Figure 56: Add post activity

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the administrator's management system. Next, the administrator needs to manage the Post of the system. In this section, the administrator wants to add a new post, then choose to add new, then enter the post information into the system and add it, now the system will check if the post information is correct, it will add the information to the database. data, message display and post display have just been added.

- **Edit Post**

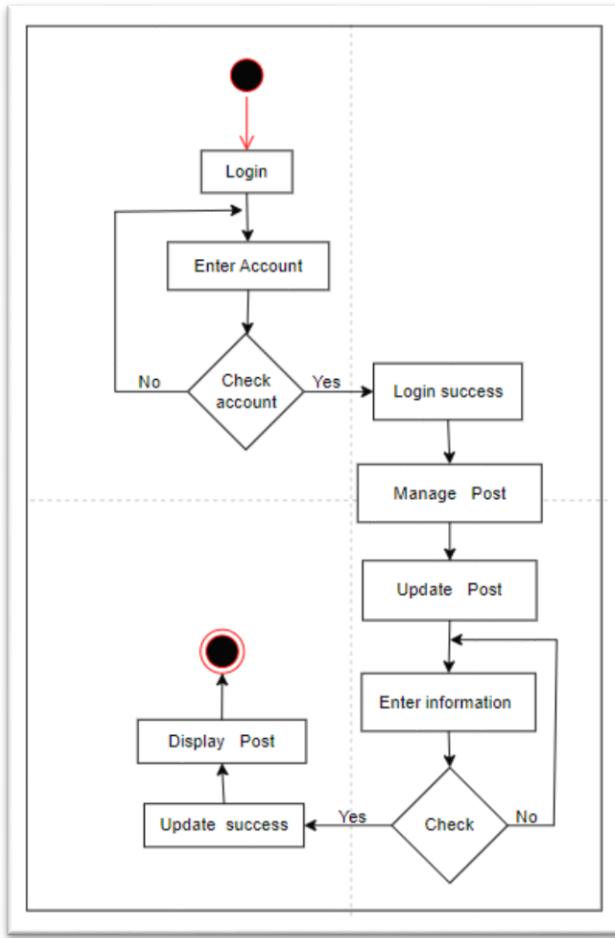


Figure 57: Edit post activity

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the administrator's management system. Next, the administrator needs to manage the post of the system. In this section, the administrator wants to edit a post, the administrator selects that post. Next will be to enter the information you need to edit and update it, this time the system will check if the post information has just been properly edited, it will be updated. The data will be updated in the database. Whether. then will display the post that has just been edited. If the system does not check properly, the administrator will re-enter the post information and not update the data.

- Delete Post

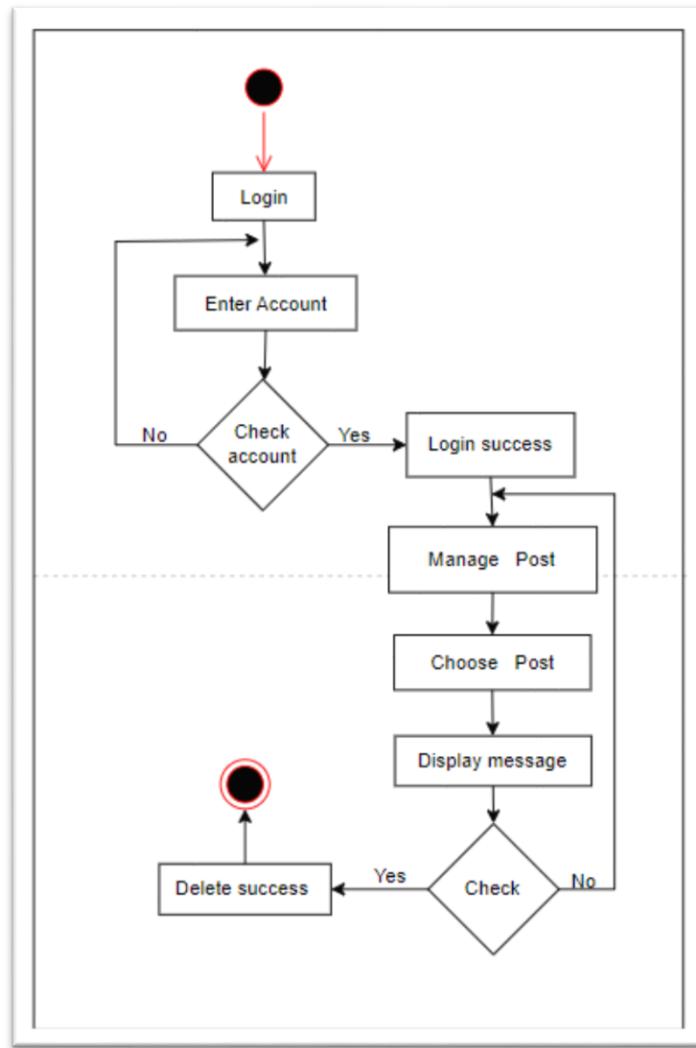


Figure 58: Delete post

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the administrator's management system. Next, the administrator needs to manage the post of the system. In this section, the administrator wants to delete post, the administrator selects that post. Then the system will display a message to make sure the administrator wants to delete it or not. If Yes is selected, the system will check and delete that post information from the database, if not, the system will return to the post management page.

- Add category

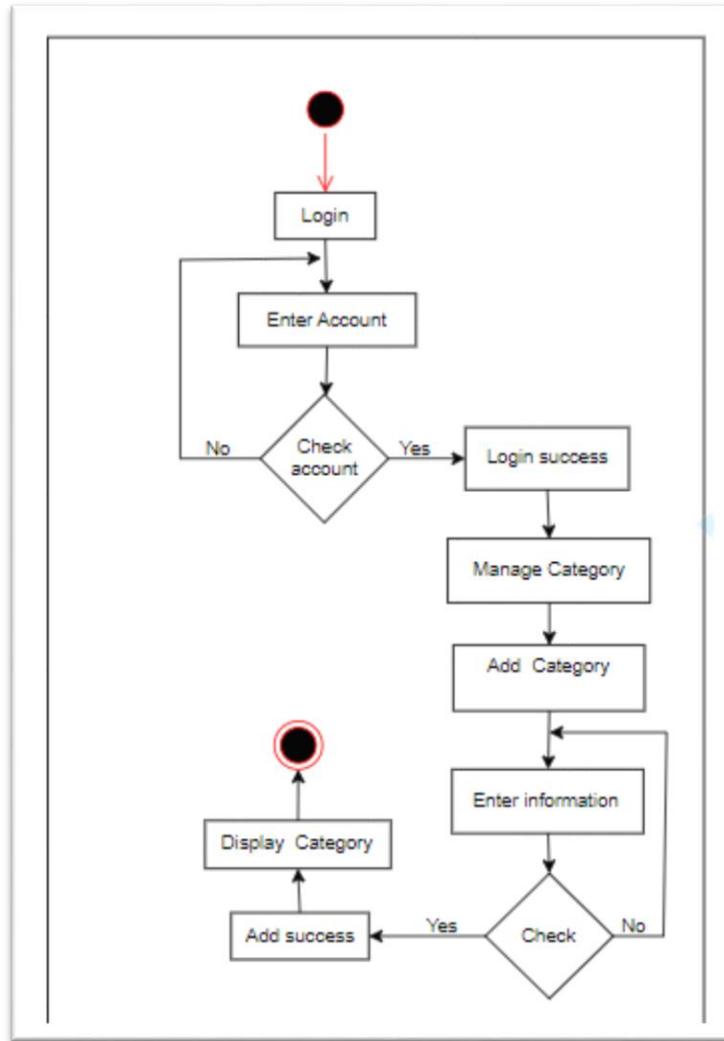


Figure 59: Add category activity

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the management system. manager's management. Next, the administrator needs to manage the category of the system. In this section, the administrator wants to add a new category, then select add new and then enter category information into the system and add, at this time the system will check if the category information is correct, then add the information. database. data, message display, and part display have just been added.

- **Edit category**

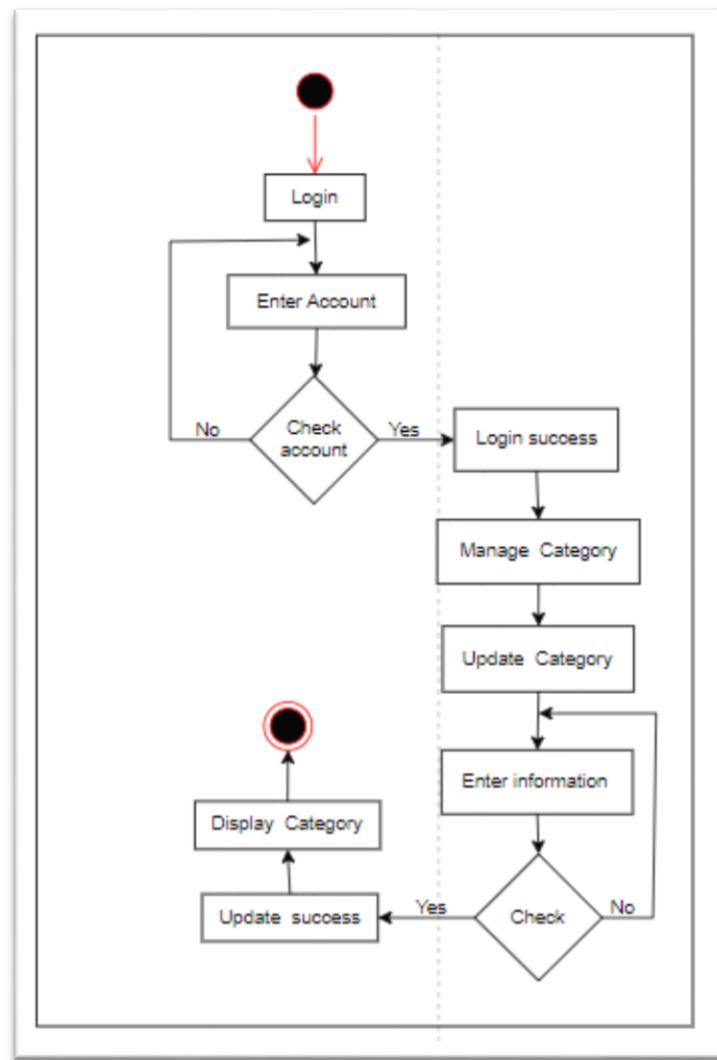


Figure 60: Edit category activity

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the administrator's management system. Next, the administrator needs to manage the category of the system. In this section, the administrator wants to edit a category, the administrator selects that department. Next will be to enter the information you need to edit and update it, this time the system will check if the category information has just been properly edited, it will be updated. The data will be updated in the database. Whether, then will display the category that has just been edited. If the system does not check properly, the administrator will re-enter the category information and not update the data.

- Delete category

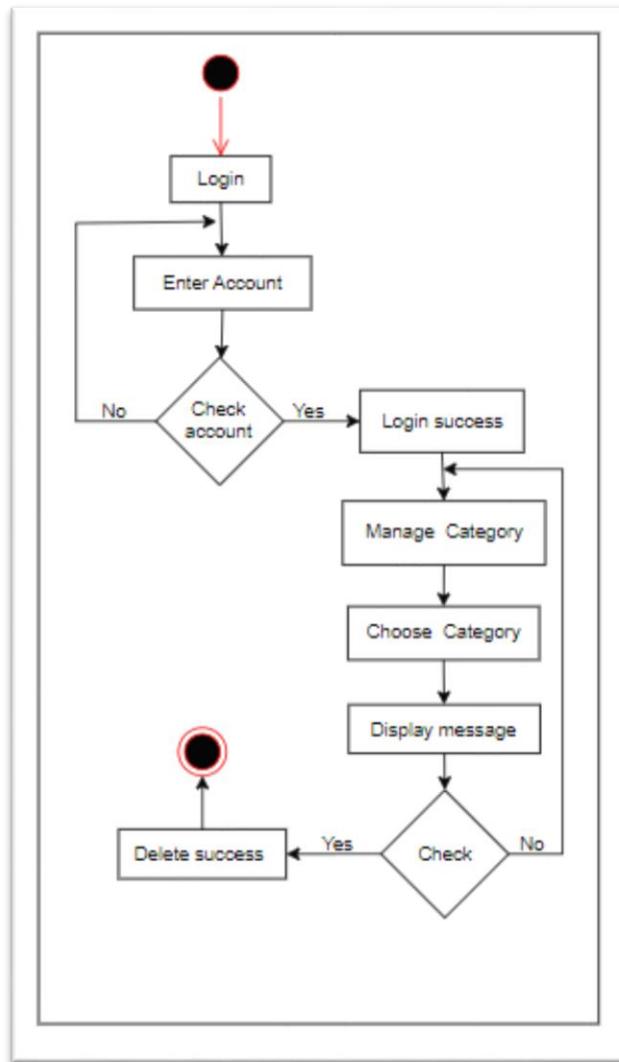


Figure 61: Delete category activity

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the administrator's management system. Next, the administrator needs to manage the category of the system. In this section, the administrator wants to delete a category, the administrator selects that category. Then the system will display a message to make sure the administrator wants to delete it or not. If Yes is selected, the system will check and delete that category information from the database, if not, the system will return to the category management page.

- Add staff

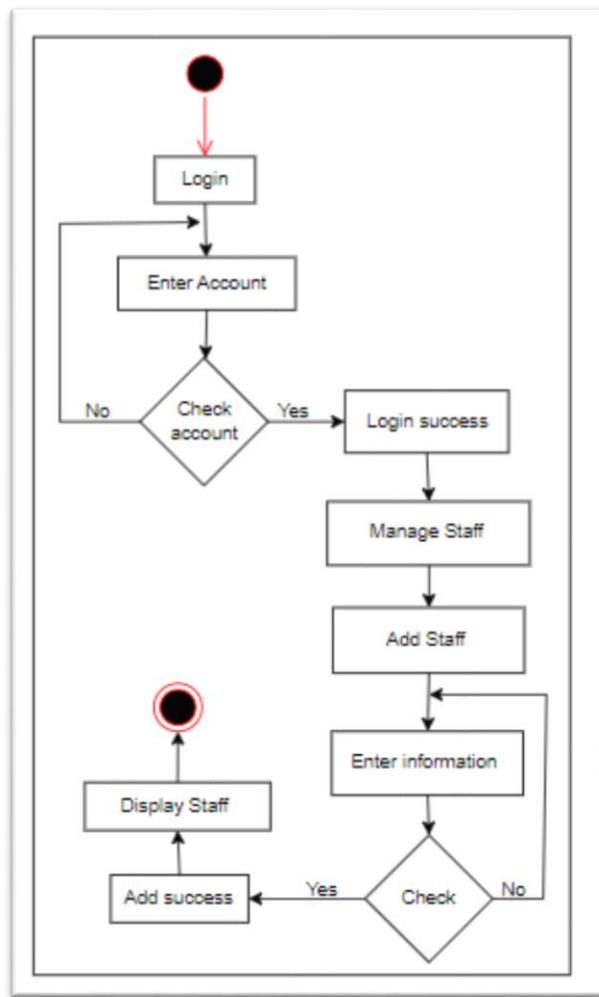


Figure 62: Add staff activity

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the management system. manager's management. Next, the administrator needs to manage the staff of the system. In this section, the administrator wants to add a new staff, then select add new and then enter the staff. Information into the system and add, at this time the system will check if the staff information is correct, then add the information. database. data, message display, and part display have just been added.

- Edit staff

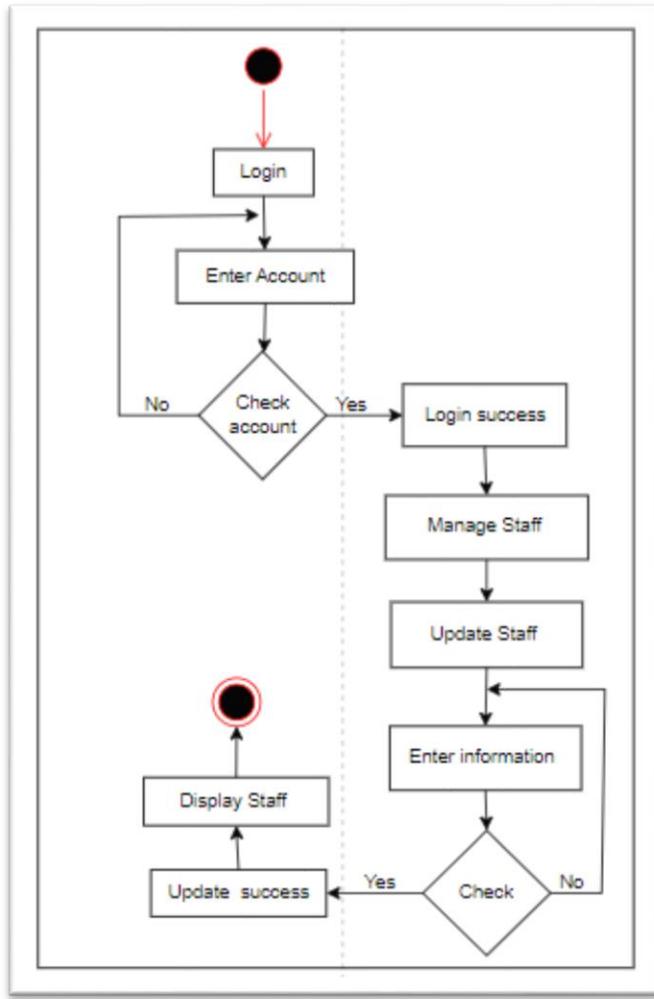


Figure 63: Edit staff activity

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the administrator's management system. Next, the administrator needs to manage the staff of the system. In this section, the administrator wants to edit a staff, the administrator selects that staff. Next will be to enter the information you need to edit and update it, this time the system will check if the staff information has just been properly edited, it will be updated. The data will be updated in the database. Whether, then will display the staff that has just been edited. If the system does not check properly, the administrator will re-enter the staff information and not update the data.

- Delete staff

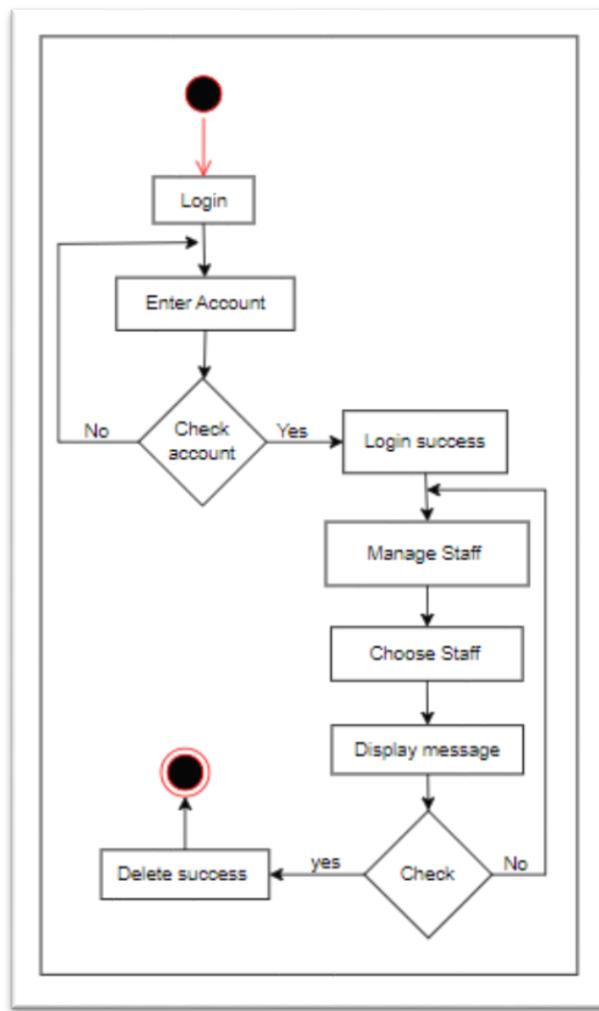


Figure 64: Delete staff activity

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the administrator's management system. Next, the administrator needs to manage the staff of the system. In this section, the administrator wants to delete a staff, the administrator selects that staff. Then the system will display a message to make sure the administrator wants to delete it or not. If yes is selected, the system will check and delete that staff information from the database, if not, the system will return to the staff management page.

- Writing feedback

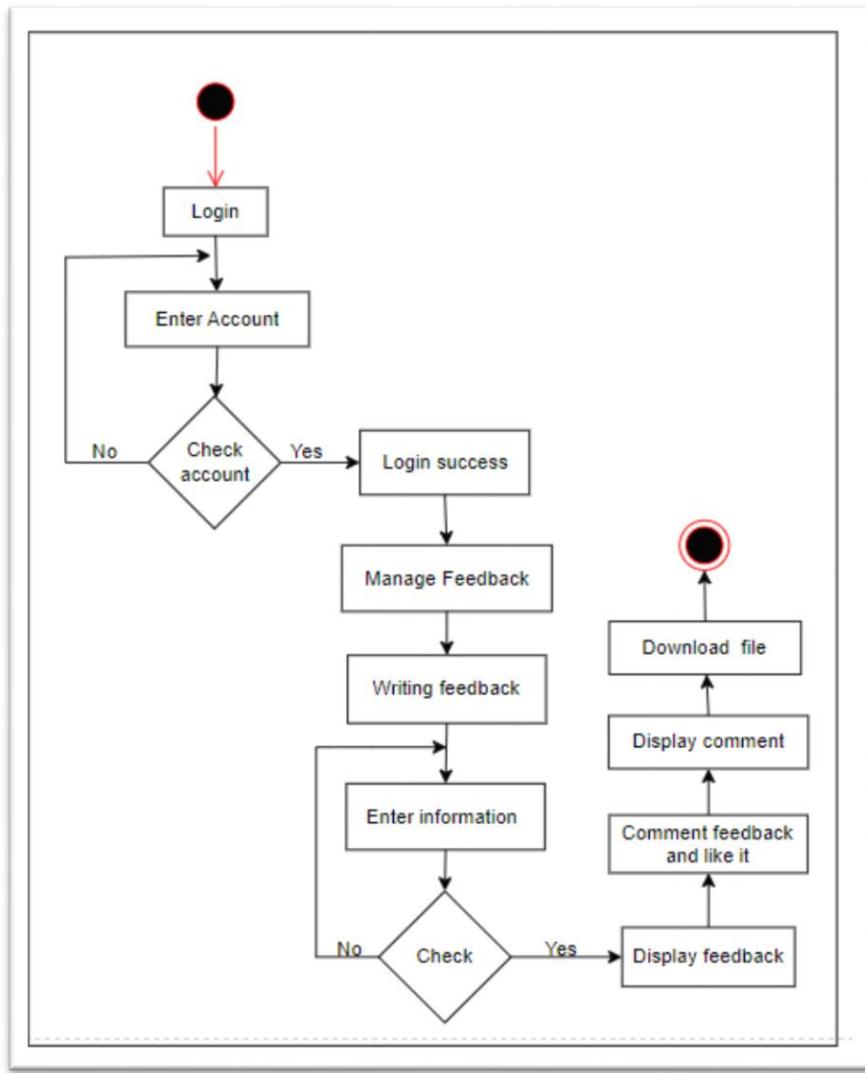


Figure 65: Writing feedback

Start admin need to log in, admin needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let admin re-enter the information, if it's correct, it will access the management system. manager's management. Next, the administrator will go to Manage Feedback and create new feedback. The administrator needs to enter the information of the feedback, now the system will check if the information is not valid, the system will let the manager re-enter the information of the feedback, if it is reasonable, the system will display the feedback and the data will be saved in the database, next the administrator will drop a like for the response and comment on that response. Then the system will display all comments and will download the file if the administrator wants.

4.2.2. Staff

- Update Account

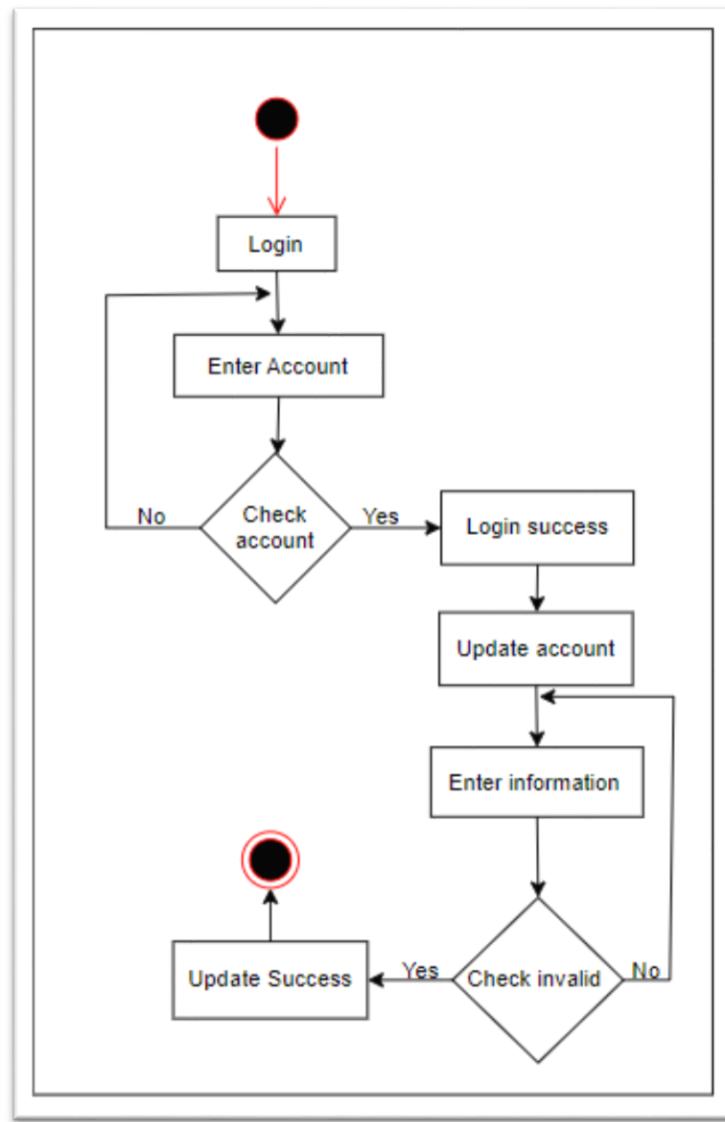


Figure 66: Update account activity

Start staff need to log in, staff needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let staff re-enter the information, if it's correct, it will access the staff management system. Next, the staff needs to choose to update the account, enter the information to be edited, at this time the system will check the staff information just entered the system, if ok, the admin's information account will be updated, if not, the system will update and let the staff re-enter the information.

- Change Password

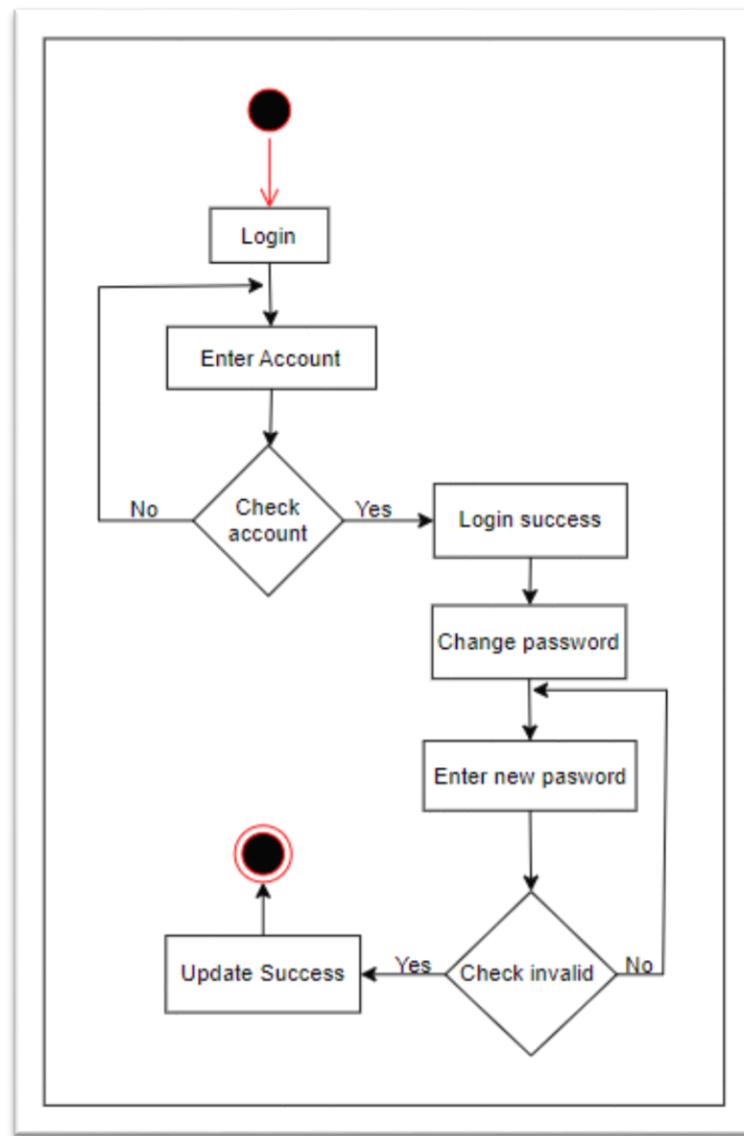


Figure 67: Change password

Start staff need to log in, staff needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let staff re-enter the information, if it's correct, it will access staff management system. Next staff needs to change his password, next will be to enter a new password, this time the system will check that admin's password will be updated, otherwise the system will not update and let staff re-enter information.

- Add Idea

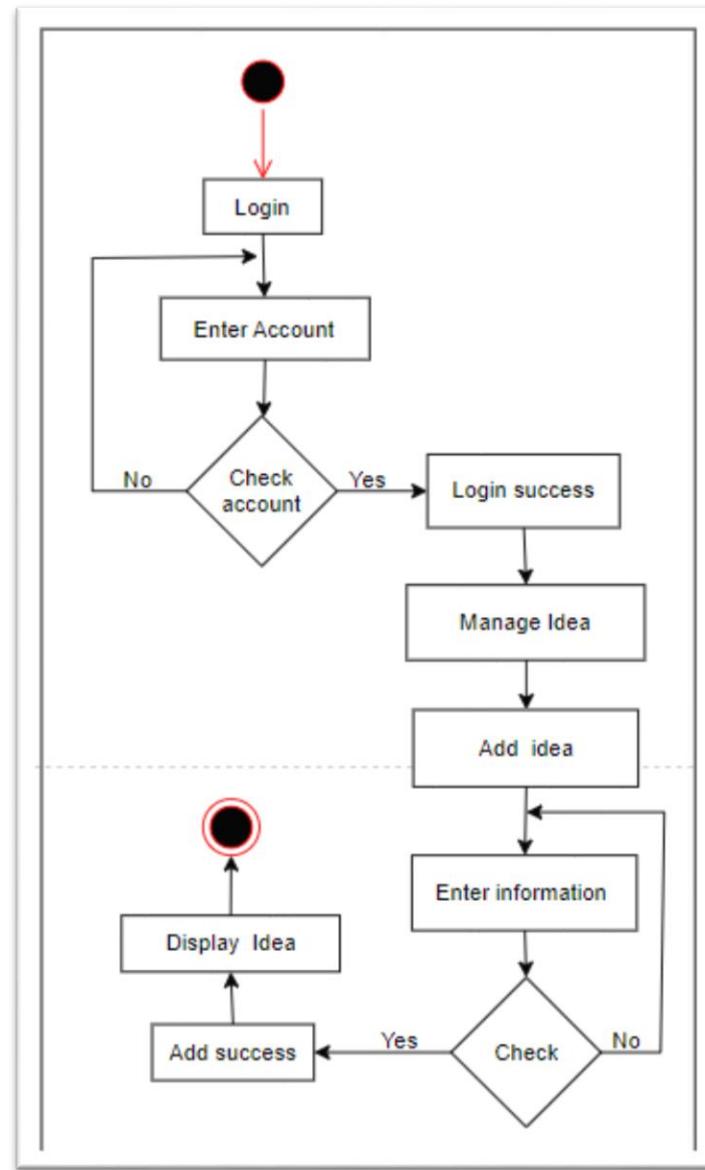


Figure 68: Add idea

Start staff need to log in, staff needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let staff re-enter the information, if it's correct, it will access a staff management system. Next, the staff needs to manage the idea of the system. In this section, the staff wants to add a new idea, then choose to add new, then enter the idea information into the system and add it, now the system will check if the idea information is correct, it will add the information to the database. data, message display and idea display have just been added.

- Edit idea

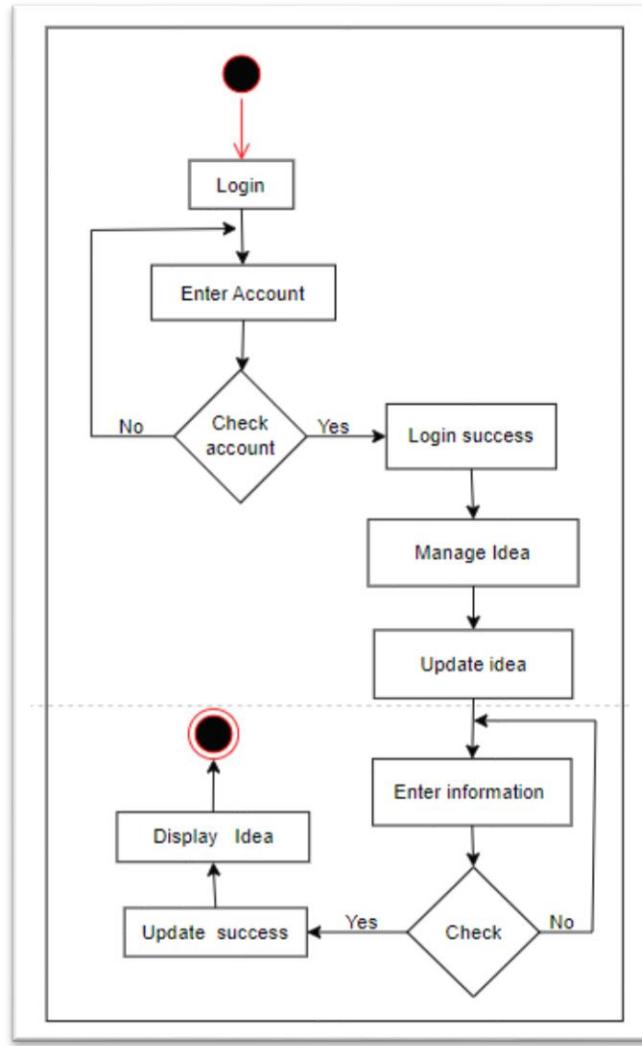


Figure 69: Edit idea

Start staff need to log in, staff needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let staff re-enter the information, if it's correct, it will access the staff management system. Next, the staff needs to manage the idea of the system. In this section, the staff wants to edit idea, the staff selects that idea. Next will be to enter the information you need to edit and update it, this time the system will check if the idea information has just been properly edited, it will be updated. The data will be updated in the database. Whether, then will display the idea that has just been edited. If the system does not check properly, the staff will re-enter the idea information and not update the data.

- Delete Idea

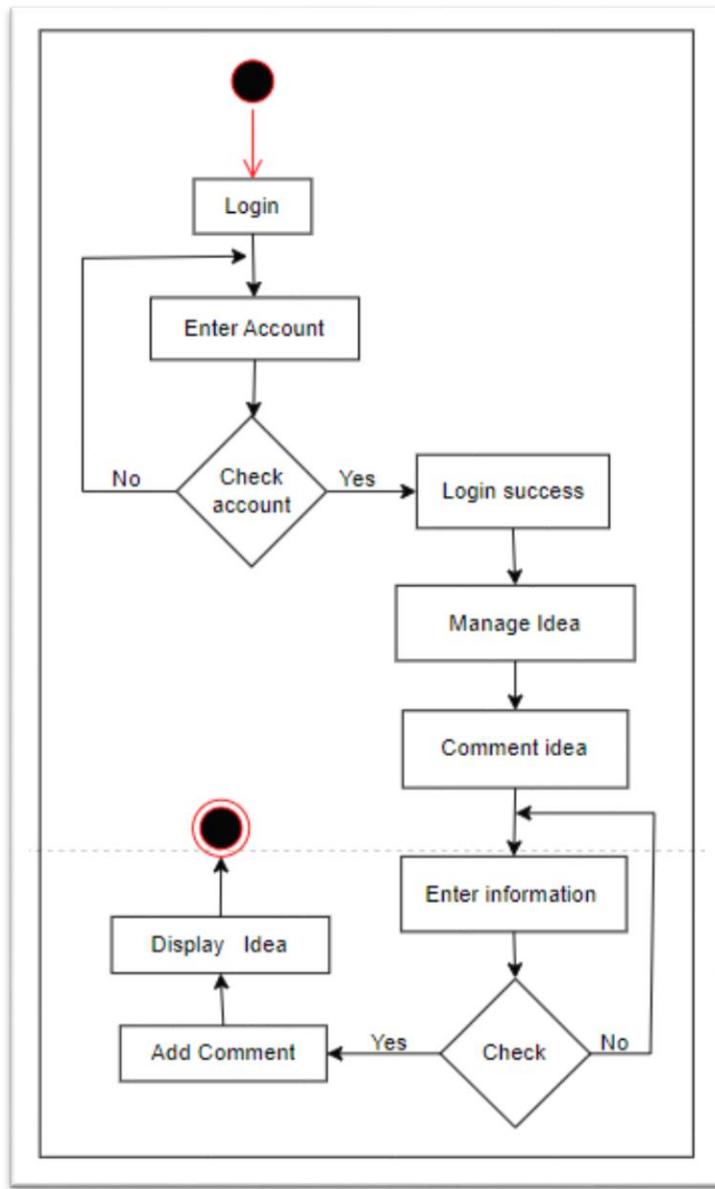


Figure 70: Delete idea

Start staff need to log in, staff needs to enter his account information, then the system will check if the account is correct or not, if it's wrong, the system will let staff re-enter the information, if it's correct, it will access the staff management system. Next, the staff needs to manage the idea of the system. In this section, the staff wants to delete idea, the staff selects that idea. Then the system will display a message to make sure the staff wants to delete it or not. If Yes is selected, the system will check and delete that idea information from the database, if not, the system will return to the idea management page.

4.3. Demo function

- Login interface

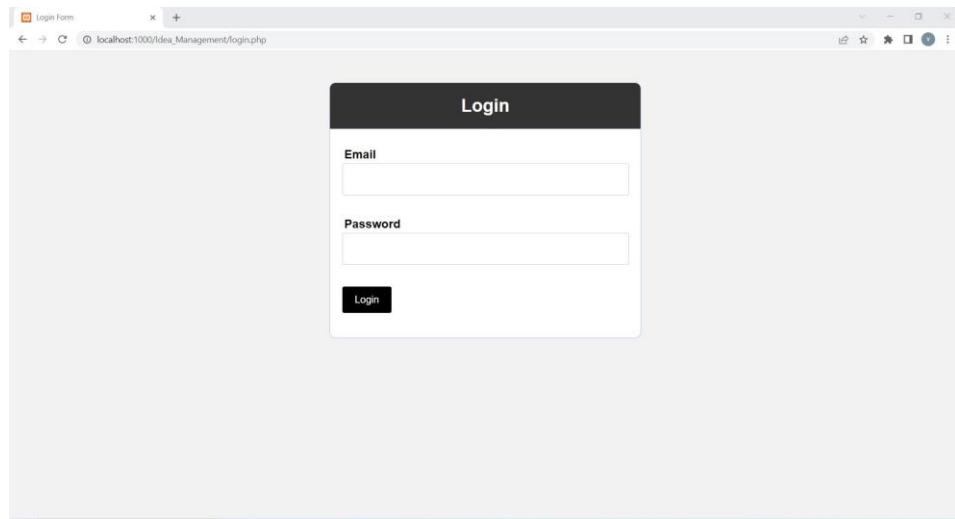


Figure 71: Login interface

Explain: This is login interface, in this page user can login with an existing account

- Login with account

A detailed view of the login interface. At the top is a dark header with the word "Login" in white. Below it is a light blue input field labeled "Email" containing the text "admin@gmail.com". Below that is another light blue input field labeled "Password" containing five dots ("....."). At the bottom is a black "Login" button.

Figure 72: Login with admin account

Explain: User can login to admin interface with admin account

- After login with admin account

The screenshot shows a web application interface for managing feedback. At the top, there is a navigation bar with links for Feedback, Post, Category, Staff, Department, QA List, and a greeting 'Hi, admin@gmail.com'. Below the navigation bar is a search bar labeled 'Search Feedback:' and a text input field with placeholder text 'Enter feedback...'. A horizontal table header row follows, containing columns for Post Title, Feedback, Category, Department, Day Submitted, Comment, and Date end comment. The main content area is currently empty, showing a light gray background.

Figure 73: After login with admin account

Explain: This is the interface after login with admin account and this is also an interface that displays information for staff so that staff can give feedback such as post title, category, department, day submitted and date end comment, admin can click on pages like post, category, staff, department, QA List to view or manage them.

- Category interface

The screenshot shows a web application interface for managing categories. At the top, there is a search bar labeled 'Search Category:' and a text input field with placeholder text 'Enter category...'. Below the search bar is a 'Main Function:' button. A horizontal table header row follows, containing columns for Name and Manage. The main content area is currently empty, showing a light gray background.

Figure 74: Category interface

Explain: This is the category interface, where admin can create category and manage them like edit and delete.

- Add category

Add Category

Category Name:

Add Category **Cancel**

Figure 75: Add category

Explain: This is the "Add Category" interface after the admin presses the create button in the post interface. Here admin can fill in the post's information and press the "Add Category" button to create a new category.

- Add category information

Add Category

Category Name:

Add Category **Cancel**

Figure 76: Add category information

Explain: This is what it looks like after I add information to the category.

- Show category information

The screenshot shows a web-based application interface for managing categories. At the top, there is a search bar labeled "Search Category:" with the placeholder "Enter category...". Below the search bar is a button labeled "Create". The main area displays a table with two columns: "Name" and "Manage". A single row is present in the table, containing the text "Event of school" under the "Name" column and two buttons, "Edit" and "Delete", under the "Manage" column. The entire interface is set against a light gray background.

Figure 77: Show category information

Explain: After filling in the category information, press the "Category" button on the "Create Category" page. Category information will be displayed here.

- Delete category

The screenshot shows a web-based application interface similar to Figure 77. It includes a search bar, a "Create" button, and a table with a single row for "Event of school". A prominent feature is a modal dialog box in the center. The dialog has a header "localhost:1000 says" and a message "Do you want to delete?". It contains two buttons: "OK" and "Cancel". The rest of the interface is visible in the background.

Figure 78: Delete category

Explain: After pressing the delete button in the interface, the message line "Do you want to delete?". Users can choose "yes" to delete the category or select "no" to cancel.

- Department interface

The screenshot shows a user interface for managing departments. At the top left is a search bar labeled "Search Department:" with the placeholder "Enter department...". Below it is a section labeled "Main Function:" with a "Create" button. A horizontal bar below these contains two buttons: "Name" and "Manage". At the bottom of the interface are social media sharing icons (Facebook, Twitter, LinkedIn, Google+) and a copyright notice: "2023 Idea Management | Design by Greenwich".

Figure 79: Department interface

Explain: This is the department interface, where admin can create department and manage them like edit and delete.

- Create department interface

The screenshot shows a modal window titled "Add Department". It has a label "Department Name:" followed by an empty text input field. At the bottom are two buttons: "Add Department" and "Cancel".

Figure 80: Create department interface

Explain: This is the "Create Department" interface after the admin presses the "create" button in the category interface. Here admin can fill in the category's information and press the "Add Department" button to create a new category.

- Add department information

Add Department

Department Name:

Add Department **Cancel**

Figure 81: Add department information

Explain: This is what it looks like after I add information to the department.

- Show department information

Search Department:	
Enter department...	
Main Function:	
Name	Manage
Computer research	Edit Delete

Figure 82: Show department information

Explain: After filling in the post information, press the "Add Department" button on the "Add Department" page. Department information will be displayed here.

- **Edit department**

Edit Department

Department Name:

Update **Cancel**

Figure 83: Edit department

Explain: This is the department edit interface, in this interface admin can change the department's information.

- **Delete department**

localhost:1000 says
Do you want to delete?

OK **Cancel**

Search Department:

Main Function:

Create

Name	Manage
Computer research	Edit Delete

Figure 84: Delete department

Explain: After pressing the delete button in the interface, the message line "Do you want to delete?". Users can choose "yes" to delete the department or select "no" to cancel.

- **Staff interface**

The screenshot shows a top navigation bar with links: Feedback, Post, Category, Staff, Department, QA List, and a user greeting "Hi, admin@gmail.com". A notification badge with the number 1 is visible. Below the navigation is a search bar labeled "Search Staff:" with a placeholder "Enter staff name...". To the right is a "Main Function:" dropdown menu. At the bottom is a table header row with columns: Full Name, Gender, Date of birth, Address, Phone number, Email, and Manage.

Full Name	Gender	Date of birth	Address	Phone number	Email	Manage
-----------	--------	---------------	---------	--------------	-------	--------

Figure 85: Staff interface

Explain: This is the staff interface, where admin can create staff and manage them like edit and delete

- **Create staff interface**

The screenshot shows a "Create Staff" form. It includes fields for Fullname (text input), Gender (dropdown menu with placeholder "-- Choose Gender --"), Department (dropdown menu with placeholder "-- Choose Department --"), Date of birth (text input with placeholder "mm/dd/yyyy" and a clear button), Address (text input), Phone number (text input), and Email (text input).

Figure 86: Create staff interface

Explain: This is the "Create staff" interface after the admin presses the create button in the staff interface. Here admin can fill in the staff's information and press the "Add" button to create a new staff.

- **Create staff information**

Feedback	Post	Category	Staff
Fullname:	Tran Quan Vinh		
Gender	Male		
Department	Computer research		
Date of birth:	06/08/2001		
Address:	ca mau		
Phone number:	0914593431		
Email:	tranquanvinh001@gmail.com		

Figure 87: Create staff information

Explain: This is what it looks like after I add information to the staff.

- Show staff information

Search Staff:

Main Function:

Full Name	Gender	Date of birth	Address	Phone number	Email	Manage
Tran Quan Vinh	Male	2001-06-08	ca mau	0914593431	tranquanvinh001@gmail.com	<button>Edit</button> <button>Delete</button>

Create

Figure 88: Show staff information

Explain: After filling in the staff information, press the "Add" button on the "Create Staff" page. Staff information will be displayed here.

- Edit staff

Edit Staff

Full Name:

Gender:

Role:

Department:

Date of birth:

 Calendar icon

Address:

Phone number:

Figure 89: Edit Staff

Explain: This is the staff edit interface, in this interface admin can change the staff's information.

- After edit staff

Search Staff:

Main Function:

Create

Full Name	Gender	Date of birth	Address	Phone number	Email	Manage
Quan Vinh	Male	2001-06-08	ca mau	0914593431	tranquanvinh001@gmail.com	Edit Delete

Figure 90: After edit staff

Explain: This is the interface after changing the fullname information of the staff.

- Delete staff

Search Staff:

Main Function:

Create

Full Name	Gender	Date of birth	Address	Phone number	Email	Manage
Quan Vinh	Male	2001-06-08	ca mau	0914593431	tranquanvinh001@gmail.com	Edit Delete

Figure 91: Delete staff

Explain: After pressing the delete button in the interface, the message line "Do you want to delete?". Users can choose "yes" to delete the staff or select "no" to cancel.

- Post interface

The screenshot shows a web-based application interface. At the top, there is a header with the University of Greenwich logo and navigation links: Feedback, Post, Category, Staff, Department, QA List, and a user greeting "Hi, admin@gmail.com". Below the header is a search bar labeled "Search Post:" followed by a text input field with placeholder text "Enter post...". To the right of the search bar is a "Main Function:" dropdown menu. A green "Create" button is located at the bottom right of the main content area. The main content area has a table-like structure with columns: Title, Department, Category, Create Date, End Date, and Manage. Below this table is a footer with social media icons (Facebook, Twitter, LinkedIn, Google+) and the text "2023 Idea Management | Design by Greenwich".

Figure 92: Post interface

Explain: This is the post interface, where admin can create posts and manage them like edit and delete

- Create post

The screenshot shows the "Create Post" form. At the top, there is a header with the University of Greenwich logo and navigation links: Feedback, Post, Category, Staff, Department, QA List, and a user greeting "Hi, admin@gmail.com". The main form area is titled "Create Post". It contains several input fields: "Title" (text input), "Category" (dropdown menu), "Department" (dropdown menu), "End submit :" (date input), "End Comment :" (date input), and "Content" (text area). At the bottom of the form is a large black "Post" button.

Figure 93: Create post

Explain: This is the "Create Post" interface after the admin presses the create button in the post interface. Here admin can fill in the post's information and press the "Post" button to create a new post.

- Add post information

Feedback Post Category Staff Department QA List Hi, admin@gmail.com

Create Post

Title: Organize an event for the holiday of 8/3

Category: Event

Department: Computer

End submit: 04/27/2023 11:59 PM

End Comment: 04/27/2023 11:59 PM

Content: Organize an event for the holiday of 8/3

Figure 94: Add post information

Explain: This is what it looks like after I add information to the post.

- Show post information

Feedback Post Category Staff Department QA List Hi, admin@gmail.com

Search Post: Enter post...

Main Function: Create

Title	Department	Category	Create Date	End Date	Manage
Event of school	Computer	Event	2023-04-15 17:00:11	2023-04-16 17:00:00	<button>Edit</button> <button>Delete</button>
Test 2	Computer	Study	2023-04-15 17:01:25	2023-04-20 17:01:00	<button>Edit</button> <button>Delete</button>
Organize an event for the holiday of 8/3	Computer	Event	2023-04-17 07:39:43	2023-04-27 23:59:00	<button>Edit</button> <button>Delete</button>

Figure 95: Show post information

Explain: After filling in the post information, press the "Post" button on the "Create Post" page. Post information will be displayed here.

- **Edit post**

Edit Post

Title:
Organize an event for the holiday of 8/3

Category
Event

Department
Computer

End submit :
04/27/2023 11:59 PM

End comment:
04/27/2023 11:59 PM

Content:
Organize an event for the holiday of 8/3

Post **Cancel**

Figure 96: Edit post interface

Explain: This is the post edit interface, in this interface admin can change the post's information.

- **After edit post**



Feedback Post Category Staff Department QA List Hi, admin@gmail.com

Search Post:

Enter post...

Main Function: Create

Title	Department	Category	Create Date	End Date	Manage
Event of school	Computer	Event	2023-04-15 17:00:11	2023-04-16 17:00:00	Edit Delete
Test 2	Computer	Study	2023-04-15 17:01:25	2023-04-20 17:01:00	Edit Delete
Create an event for the holiday of 8/3	Computer	Event	2023-04-17 07:41:17	2023-04-27 23:59:00	Edit Delete

Figure 97: After edit post

Explain: This is the interface after changing the title information of the post.

- **Delete post**

The screenshot shows a web application interface. At the top right, there is a user profile with the text "Hi, admin@gmail.com" and a notification badge with the number "1". Below the header, there is a navigation bar with links for "Department", "QA List", and "Search Post:". A search bar below the navigation bar contains the placeholder "Enter post...". On the right side of the page, there is a modal dialog box with the text "localhost:1000 says" and "Do you want to delete?". It has two buttons: "OK" and "Cancel". Below the modal, there is a table titled "Main Function:" with a "Create" button. The table has columns for Title, Department, Category, Create Date, End Date, and Manage. One row in the table is shown, containing the text "Create an event for the holiday of 8/3", "Computer research", "Event of school", "2023-04-15 09:30:38", "2023-04-27 23:59:00", and "Edit" and "Delete" buttons.

Title	Department	Category	Create Date	End Date	Manage
Create an event for the holiday of 8/3	Computer research	Event of school	2023-04-15 09:30:38	2023-04-27 23:59:00	<button>Edit</button> <button>Delete</button>

Figure 98: Delete post

Explain: After pressing the delete button in the interface, the message line "Do you want to delete?". Users can choose "yes" to delete the post or select "no" to cancel.

- **QA list interface**

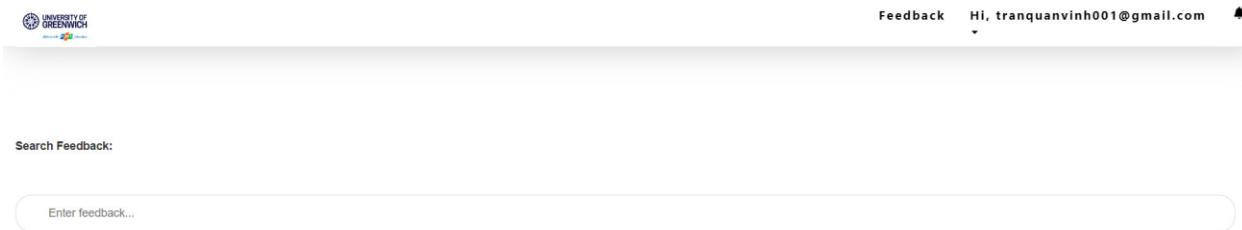
The screenshot shows a web application interface for managing QA data. At the top left, there is a search bar with the placeholder "Search QA name:". Below the search bar, there is another search bar with the placeholder "Enter department name...". On the right side of the page, there is a table titled "Main Function:" with a "Create" button. The table has columns for Full Name, Gender, Date of birth, Address, Phone number, Email, and Manage. One row in the table is shown, containing the text "Vinh", "Other", "2023-03-27", "TP.Can Tho", "021382876", "admin@gmail.com", and "Edit" and "Delete" buttons.

Full Name	Gender	Date of birth	Address	Phone number	Email	Manage
Vinh	Other	2023-03-27	TP.Can Tho	021382876	admin@gmail.com	<button>Edit</button> <button>Delete</button>

Figure 99: QA list interface

Explain: This is the QA list interface, where admin can manage them like edit and delete.

- Staff interface



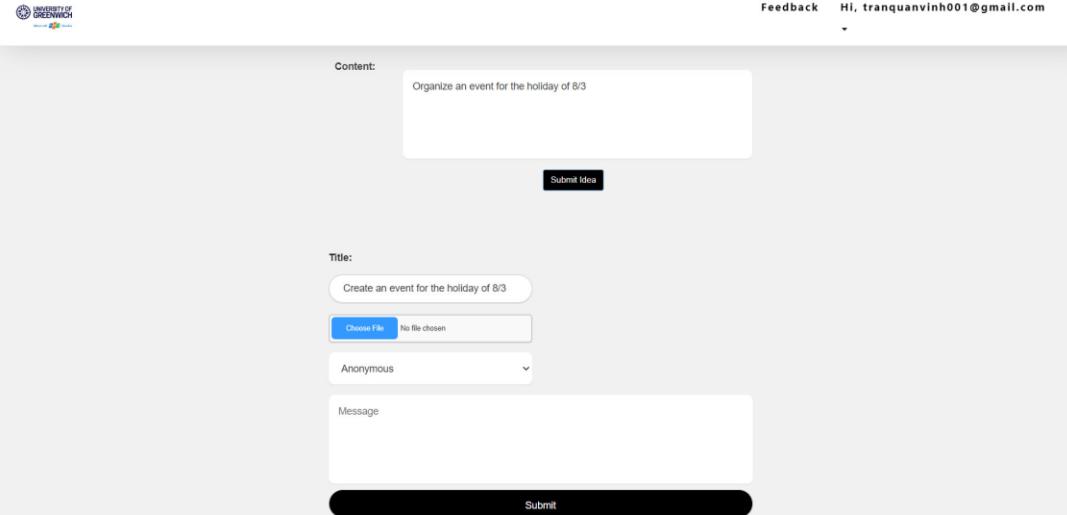
The screenshot shows a web-based feedback system. At the top right, there is a user profile with the name "Hi, tranquanvinh001@gmail.com" and a dropdown arrow. To the left of the profile, there is a "Feedback" button. On the far right, there is a small bell icon. Below the header, there is a search bar labeled "Search Feedback:" followed by a placeholder "Enter feedback...". A table below the search bar displays a single row of post information:

Post Title	Feedback	Category	Department	Day Submitted	Comment	Date end comment
Create an event for the holiday of 8/3	Feedback	Event of school	Computer research	2023-04-27 23:59:00	2023-04-29 23:59:00	Comment

Figure 100: Staff interface

Explain: This is the feedback interface when the user login with staff account, in this interface staff can see post information and feedback about this post. In addition, staff can also view feedback and comment on existing feedback.

- View post and feedback interface

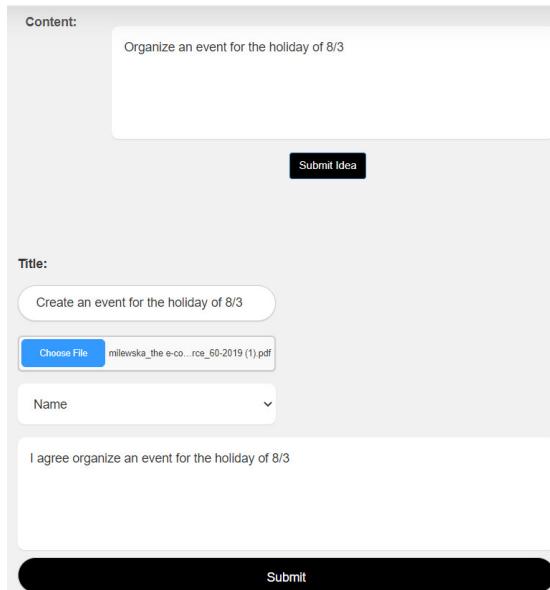


The screenshot shows a "View post and feedback interface". At the top right, there is a user profile with the name "Hi, tranquanvinh001@gmail.com" and a dropdown arrow. To the left of the profile, there is a "Feedback" button. On the far right, there is a small bell icon. Below the header, there is a "Content:" section containing a text area with the text "Organize an event for the holiday of 8/3" and a "Submit Idea" button. Below this, there is a "Title:" section with a text input field containing "Create an event for the holiday of 8/3", a file upload field with "Choose File No file chosen", and a dropdown menu set to "Anonymous". At the bottom, there is a "Message" text area and a large black "Submit" button.

Figure 101: View post and feedback interface

Explain: This is the view post and feedback interface when the staff click the “feedback” button in feedback interface, in this interface staff can see post information and click “submit idea” to feedback about this post.

- Submit idea information



A screenshot of a web-based form for submitting an idea. The form has a light gray background with several input fields and buttons.

Content:
Organize an event for the holiday of 8/3

Submit Idea

Title:
Create an event for the holiday of 8/3

Choose File: miliewska_the e-co...rcs_60-2019 (1).pdf

Name

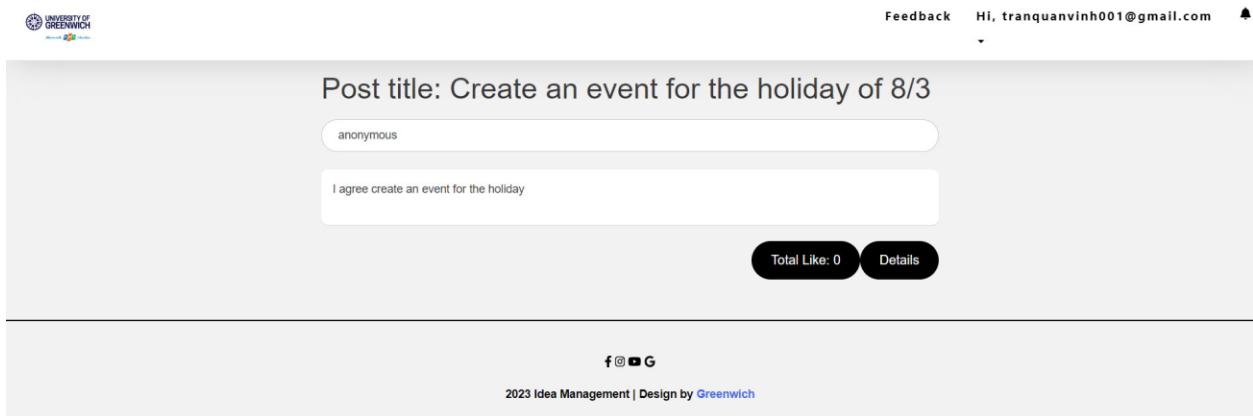
I agree organize an event for the holiday of 8/3

Submit

Figure 102: Submit idea information

Explain: This is what it looks like after I add information to the idea.

- Idea information interface



A screenshot of the idea information interface. The top navigation bar includes the University of Greenwich logo, a feedback link, and a user greeting (Hi, tranquanvinh001@gmail.com).

Post title: Create an event for the holiday of 8/3

anonymous

I agree create an event for the holiday

Total Like: 0 Details

f @ G

2023 Idea Management | Design by Greenwich

Figure 103: Idea information interface

Explain: Here is the idea information interface, here users can see full information of the idea such as total likes and content of the idea or click details to like and comment this idea.

- Idea information interface with admin account

The screenshot shows a web page for an idea titled "Create an event for the holiday of 8/3". The page includes a header with navigation links like Feedback, Post, Category, Staff, Department, QA List, and a greeting for "Hi, admin@gmail.com". Below the title, it says "anonymous". A text area contains the content "I agree organize an event for the holiday of 8/3". At the bottom, there are buttons for "Download file", "Total Like: 0", and "Details". Social sharing icons for Facebook, Twitter, LinkedIn, and Google+ are at the bottom left, along with a copyright notice: "2023 Idea Management | Design by Greenwich".

Figure 104: Idea information interface with admin account

Explain: Here is the idea information interface, here users can see full information of the idea such as total likes and content of the idea or click details to like and comment this idea and admin can click download file to see file of staff.

- Comment idea interface

The screenshot shows a comment form for the same idea. It has a header with "Feedback" and "Hi, tranquanvinh001@gmail.com". The main area is labeled "POST" and contains a text input field with the placeholder "I agree create an event for the holiday". Below the input field are "Comments" and "Like: 0" buttons. A large empty text area for a comment is followed by a "Submit" button. Social sharing icons and a copyright notice are at the bottom.

Figure 105: Comment idea interface

Explain: This is comment idea interface, in this interface user can comment and like idea

- After comment and like idea

UNIVERSITY OF GREENWICH

Feedback Hi, tranquanvinh001@gmail.com

POST

I agree create an event for the holiday

Comments

Quan Vinh 2023-04-15

OK

Like: 1

Submit

f @ G

2023 Idea Management | Design by Greenwich

Figure 106: After comment and like idea

Explain: After commenting and liking the idea, the interface will display like this.

- Update profile

UNIVERSITY OF GREENWICH

Feedback Hi, tranquanvinh001@gmail.com

Update Profile

Full Name: Quan Vinh

Gender: Other

Date of birth: 06/08/2001

Address: ca mau

Phone number: 0914593431

Update Cancel

Figure 107: Update profile

Explain: This is the profile update interface, at this interface the user can change the information and press the "Update" button to change it.

- Change password interface

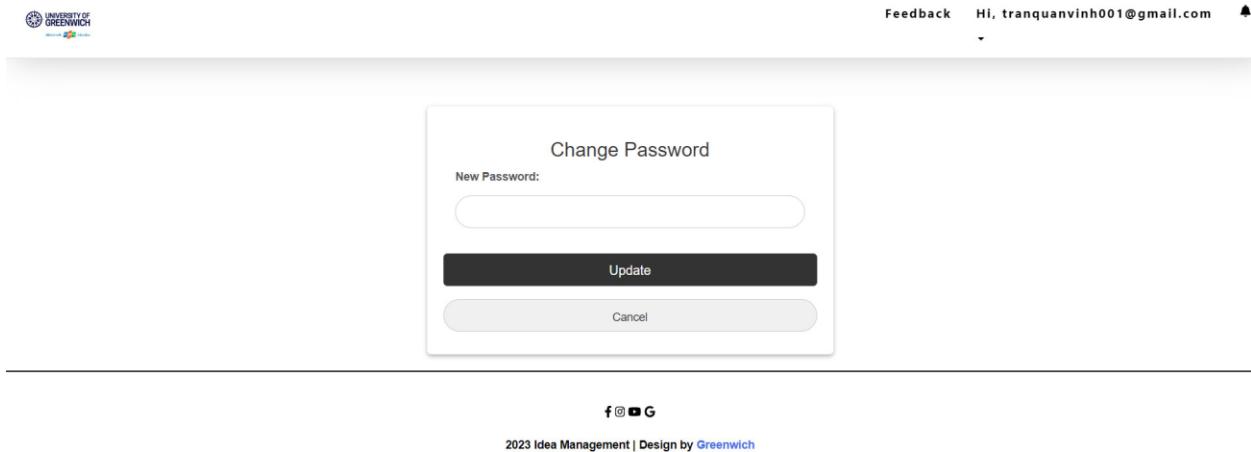


Figure 108: Change password interface

Explain: This is the password change interface, at this interface the user can change the account's password and press the "Update" button to change.

5. Testing

5.1. Testing scope

- Make sure the system functions are always working properly
- Assuring the effectiveness of the system.
- Testing time: 2/20/2023 – 4/17/2023.

5.2. Test details

- The goal is to create appropriate test cases for every sprint and carry out testing.
- Requirement testing:
 - Login/Logout
 - Add, edit, delete department
 - Add, edit, delete category
 - Add, edit, delete post
 - Deadline feedback
 - Deadline comment
 - Add, edit, delete staff

- Upload file
- Download file
- Update profile
- Change password
- Feedback
- Comment
- Like
- Notification
- Check button
- Test levels
 - System Test
 - Integration Test
 - Acceptance Test

5.2.1. Sprint 1

Date: 2/20/2023

Table 4: Test sprint 1

User's Story	Task	Test case description	Test input	Expected Output	Actual Output	Test Result
Managers can add, edit, delete and review existing departments in the system.	Test CRUD department function	Create department	Enter department name: Computer	The added department information will be displayed on the department page as "Computer"	The added department information will be displayed on the department page as "Computer"	Pass
		Edit department	Change department name information from "Computer" to "Business"	Department information that has been changed will be displayed from "Computer" to "Business"	Department information that has been changed will be displayed from "Computer" to "Business"	Pass
		Delete department	Delete department name "Business" from department page	Department with name "Business" will disappear from department list	Department with name "Business" will disappear from department list	Pass
	Check button	Delete department	Click delete button in department interface	The message "Do you want to delete" will be displayed	The message "Do you want to delete" will be displayed	Pass
		Edit department	Click edit button in department interface	The edit department interface will be displayed	The edit department interface will be displayed	Pass
		Create department	Click create button in	The create department interface will be displayed	The create department interface will be displayed	Pass

			department interface			
--	--	--	----------------------	--	--	--

- **Achieving after sprint 1:**

- Create department
- Edit department
- Delete department
- Check create department button
- Check edit department button
- Check delete department button

- **Not achieving after sprint 1:** None

5.2.2. Sprint 2

Table 5: Test sprint 2

User's Story	Task	Test case description	Test input	Expected Output	Actual Output	Test Result
Managers can add, edit, delete and review existing category in the system.	Test CRUD category function	Create category	Enter category name: Event	The added category information will be displayed on the category page as " Event "	The added category information will be displayed on the category page as " Event "	Pass
		Edit category	Change category name information from " Event " to "Study"	Category information that has been changed will be displayed from " Event " to "Study"	Category information that has been changed will be displayed from " Event " to "Study"	Pass
		Delete category	Delete category name " Study " from category page	Category with name "Business" will disappear from category list	Category with name " Study " will disappear from category list	Pass

	Check button	Delete category	Click delete button in category interface	The message "Do you want to delete" will be displayed	The message "Do you want to delete" will be displayed	Pass
		Edit category	Click edit button in category interface	The edit category interface will be displayed	The edit category interface will be displayed	Pass
		Create category	Click create button in category interface	The create category interface will be displayed	The create category interface will be displayed	Pass

- **Achieving after sprint 2:**
 - Create category
 - Edit category
 - Delete category
 - Check create category button
 - Check edit category button
 - Check delete category button
- **Not achieving after sprint 2:** None

5.2.3. Sprint 3

Table 6: Test sprint 3

User's Story	Task	Test case description	Test input	Expected Output	Actual Output	Test Result
Managers can add, edit, delete and	Test CRUD post function	Create post	Enter post name: Organize an event for the holiday of 8/3 Category: Event Department: Computer	The added post information will be displayed on the post page as: Post name: Organize an event for the holiday of 8/3	The added post information will be displayed on the post page as:	Pass

review existing post in the system.		End submit: 27/4/2023 11:59 PM End comment: 27/4/2023 11:59 PM Content: Organize an event for the holiday of 8/3	Category: Event Department: Computer End submit: 27/4/2023 11:59 PM End comment: 27/4/2023 11:59 PM Content: Organize an event for the holiday of 8/3	Post name: Organize an event for the holiday of 8/3 Category: Event Department: Computer End submit: 27/4/2023 11:59 PM End comment: 27/4/2023 11:59 PM Content: Organize an event for the holiday of 8/3	
	Edit post	Change post name from: Organize an event for the holiday of 8/3 to "Create Organize an event holiday of 8/3" Category: Event Department: Computer End submit: 27/4/2023 11:59 PM End comment: 27/4/2023 11:59 PM Content: Organize an event for the holiday of 8/3	Post information that has been changed will be displayed from Organize an event for the holiday of 8/3 to "Create Organize an event holiday of 8/3" Category: Event Department: Computer End submit: 27/4/2023 11:59 PM End comment: 27/4/2023 11:59 PM Content: Organize an event for the holiday of 8/3	Post information that has been changed will be displayed from Organize an event for the holiday of 8/3 to "Create Organize an event holiday of 8/3" Category: Event Department: Computer End submit: 27/4/2023 11:59 PM End comment: 27/4/2023 11:59 PM Content: Organize an event for the holiday of 8/3	Pass
	Delete post	Delete post name "Create Organize an event holiday of 8/3"	Post with name " Create Organize an event holiday of 8/3" will disappear from post list	Post with name " Create Organize an event holiday of 8/3" will disappear from post list	Pass
	Check button	Delete post	Click delete button in post interface	The message "Do you want to delete" will be displayed	The message "Do you want to delete" will be displayed

		Edit post	Click edit button in post interface	The edit post interface will be displayed	The edit post interface will be displayed	Pass
		Create post	Click create button in post interface	The create post interface will be displayed	The create post interface will be displayed	Pass

- **Achieving after sprint 3:**

- Create post
- Edit post
- Delete post
- Check create post button
- Check edit post button
- Check delete post button

- **Not achieving after sprint 3:** None

5.2.4. Sprint 4

Table 7: Test sprint

User's Story	Task	Test case description	Test input	Expected Output	Actual Output	Test Result
Managers can add, edit, delete and review existing staff (staff account)	Test CRUD staff function	Create staff	Enter staff fullname: Tran Quan Vinh Gender: Male Role: Staff Department: Computer Date of birth: 06/06/2001 Address: Ca Mau Phone number: 0123456789 Email: vinh@gmail.com	The added staff information will be displayed on the staff page as: Fullname: Tran Quan Vinh Gender: Male Role: Staff Department: Computer Date of birth: 06/06/2001 Address: Ca Mau Phone number: 0123456789 Email: vinh@gmail.com	The added staff information will be displayed on the staff page as: Fullname: Tran Quan Vinh Gender: Male Role: Staff Department: Computer Date of birth: 06/06/2001 Address: Ca Mau	Pass

in the system.					Phone number: 0123456789 Email: vinh@gmail.com	
		Edit staff	Change staff fullname from: "Tran Quan Vinh" to "Quan Vinh" Gender: Male Role: Staff Department: Computer Date of birth: 06/06/2001 Address: Ca Mau Phone number: 0123456789 Email: vinh@gmail.com	Staff information that has been changed will be displayed from "Tran Quan Vinh" to "Quan Vinh" Role: Staff Department: Computer Date of birth: 06/06/2001 Address: Ca Mau Phone number: 0123456789 Email: vinh@gmail.com	Staff information that has been changed will be displayed from "Tran Quan Vinh" to "Quan Vinh" Role: Staff Department: Computer Date of birth: 06/06/2001 Address: Ca Mau Phone number: 0123456789 Email: vinh@gmail.com	Pass
		Delete staff	Delete staff fullname "Quan Vinh"	Staff with fullname "Quan Vinh" will disappear from post list	Staff with fullname "Quan Vinh" will disappear from post list	Pass
		Check button	Delete staff	Click delete button in staff interface	The message "Do you want to delete" will be displayed	The message "Do you want to delete" will be displayed
			Edit staff	Click edit button in staff interface	The edit staff interface will be displayed	The edit staff interface will be displayed
			Create staff	Click create button in staff interface	The create staff interface will be displayed	The create staff interface will be displayed

- Achieving after sprint 4:

- Create staff
- Edit staff
- Delete staff
- Check create staff button

- Check edit staff button
- Check delete staff button
- **Not achieving after sprint 4:** None

5.2.5. Sprint 5

Table 8: Test sprint 5

User's Story	Task	Test case description	Test input	Expected Output	Actual Output	Test Result
All users will receive a notification when someone comments on their feedback or when QA create post for them	Test notifications for user	Check the notification function when someone comments on an staff feedback	Use another account to comment on staff feedback	The notification bell will display the number 1, and when pressed, it will display the message of the commenter.	The notification bell will display the number 1, and when pressed, it will display the message of the commenter.	Pass
		Check the notification function when admin create post for them	Use QA account and create post	The notification bell will display the number 1, and when pressed, it will display the message of the new post.	The notification bell will display the number 1, and when pressed, it will display the message of the new post.	Pass
		The number displayed in the bell will disappear when the user clicks to view the message	Click the number displayed on the bell to see new notifications	The number displayed in the bell will disappear	The number displayed in the bell will disappear	Pass

		Click the bell to see old notifications	Click the notification bell	Old notifications will be displayed again	Nothing happens when the bell is pressed	Fail
--	--	---	-----------------------------	---	--	------

- **Achieving after sprint 5:**

- Check the notification function when someone comments on an staff feedback
- Check the notification function when admin create post for them
- The number displayed in the bell will disappear when the user clicks to view the message
- Click the bell to see old notifications

- **Not achieving after sprint 5:** 1 (Click the bell to see old notifications)

5.2.6. Sprint 6

Table 9: Test print 6

User's Story	Task	Test case description	Test input	Expected Output	Actual Output	Test Result
Staff can post feedback on post they can see	Verify upload feedback function	Upload feedback	Title: Suggestion on IT Content: Should add more Java courses Choose anonymous or your name File upload: 1 PDF may or may not be available Enter feedback about: IT	All information about feedback will be displayed in the comments page and every can see and comment like as: Title: feedback on IT Content: "Should add more Java courses" Select topic: IT File upload: 1 PDF yes or no Enter feedback about: IT	All information about feedback will be displayed in the comments page and every can see and comment like as: Title: feedback on IT Content: "Should add more Java courses" Select topic: IT File upload: 1 PDF yes or no Enter feedback about: IT	Pass

		Upload file in PDF	Click the button “Choose file” and select file PDF and fill the information after that click submit button	The file will be added and QA can download it in the comment page.	The file will be added and QA can download it in the comment page.	Pass
Staff can comment and like on feedback they can see	Verify “Comment and like feedback” function	User can comment feedback	Username: Quan Vinh Enter text: “I agree” At: 4-17-2023 10:20:31 AM And click submit button	Comments show with the following data Commenter’s name: “Quan Vinh” Comment date: “4-17-2023” Comment text: “I agree”	Comments show with the following data Commenter’s name: “Quan Vinh” Comment date: “4-17-2023” Comment text: “I agree”	Pass
		User can like feedback	Click the like button in the comment pages	Depending on the number of likes, the like button will display from 0 to 1 or 1 to 2. Each individual may only press it once, and doing so a second time will cancel that turn.	Depending on the number of likes, the like button will display from 0 to 1 or 1 to 2. Each individual may only press it once, and doing so a second time will cancel that turn.	Pass

- **Achieving after sprint 6:**
 - Upload feedback
 - Upload file in PDF
 - User can comment feedback
 - User can like feedback
- **Not achieving after sprint 6:** None

6. Agile Scrum implementation

Scrum is a framework that provides users with a range of techniques and processes for effective application management and product development through continuous iteration. Our team adopted the Scrum methodology to manage and develop the application in this project, and we have been able to witness the effectiveness of this approach firsthand (Peek, 2023).

6.1 Product backlog

A list of system requirements considered necessary is maintained in the form of a Product Backlog, which serves as the central repository for tracking any changes to the product. The Product Owner has the responsibility of managing the availability, contents and priority of the requirements enlisted in the Product Backlog (Raeburn, 2022).

Table 10: Product backlog

PRODUCT BACKLOG							
ID	Categories	Role	Action	Explanation	Priority	Status	Note
1	CRUD	QA Manager	CRUD Department	Administrators can create a new department or update, view and delete existing departments	Must have	Done	
2	CRUD	QA Manager	CRUD Category	Admin can create new categories or update, view and delete existing categories	Must have	Done	
3	CRUD	QA Manager	CRUD Topic	Admin can create new topic or update, view and delete existing topics	Must have	Done	
4	CRUD	QA Manager	CRUD Staff account	Admin can create new staff account or update, view and delete existing staff account	Must have	Done	
5	Viewing	QA Manager	Viewing topic	They can view previously created topics and feedback.	Must have	Done	
6	Comment	QA Manager	Comment feedback	Managers can comment on feedback already posted by others	Must have	Done	

7	Notification	Staff	Notification for user	When someone comments on the feedback, a notification will be sent to the person who posted that feedback.	Must have	Done	
8	Viewing	Staff	View topic, feedback	Users can view topics belonging to their department and see feedback on those topics	Must have	Done	
9	Posting	Staff	Post feedback	When they see a topic in their department, they can post feedback on that topic.	Must have	Done	
10	Comment	Staff	Comment feedback	They can comment on feedback on topics in their department	Must have	Done	
11	Updating	Staff	Update account	Staff can update basic information in their account	Must have	Done	

The Product Backlog of the website, defined by the given specifications, is seamlessly presented in the figure. Each item in the backlog has four crucial components, namely role, action, explanation, and priority. Our application has effectively met all user requirements outlined in the Product Backlog.

During the development process, we gave the utmost priority to the 'Must have' requirements to ensure system reliability, given their criticality.

6.2 Sprint backlogs

The Sprint Backlog is a crucial management tool that assists the development team in executing work during a Sprint effectively. Each Sprint Backlog comprises Product Backlog Items that the development team has selected for the current Sprint, along with a comprehensive work plan to achieve the Sprint Goal.

Primarily, the Sprint Backlog is represented in the form of a task table that breaks down user stories and tasks that need completion. The development team has the authority to update the Sprint Backlog throughout the Sprint, by adding new tasks or removing unnecessary ones.

In the Sprint Backlog, tasks or jobs can be assigned a status, such as 'To Do', 'In Progress', and 'Done'. Moreover, when a task is marked as "Done", the estimated value of that task is instantly updated.

(Asana, 2022)

6.2.1 Sprint 1

Table 11: Sprint 1

SPRINT 1												
ID	User's story	Task	Owner	Estimated hours	Day 1 (20/2)	Day (21/2)	Day (22/2)	Day (23/2)	Day (24/2)	Day (25/2)	Day (26/2)	Status
1	Managers can add, edit, delete and review existing departments in the system.	Code department interface	Vinh	14	2	3	2	2	3	2	0	Completed
		Database design + connect	Duy	10	3	0	0	3	2	0	2	Completed
		Code create department	Quan	10	3	3	0	0	2	2	0	Completed
		Code update department	Duy	8	0	3	3	0	0	2	0	Completed
		Code delete department	Phat	10	2	2	0	0	3	3	0	Completed
		Testing	Phat + Quan	10	0	0	4	3	0	0	3	Completed

In Sprint 1, we started with coding necessary Department functions. Department is used to divide the staff in the company, they will be allocated in the department they work, then they will be able to see the categories of that department created by the QA Manager. QA Managers can create new parts or distribute existing ones to their staff.

We will first implement the interface code of the system as well as the interface of the department management page (front-end), then will design the database based on the previous designs to store the necessary data, next will be the functions in CRUD and finally will be the test to catch and fix the error.

All the goals we set in the Sprint were completed well, most of the early days we achieved many of the goals we set, but in the later times we started to slow down and balance with the goal. original pepper

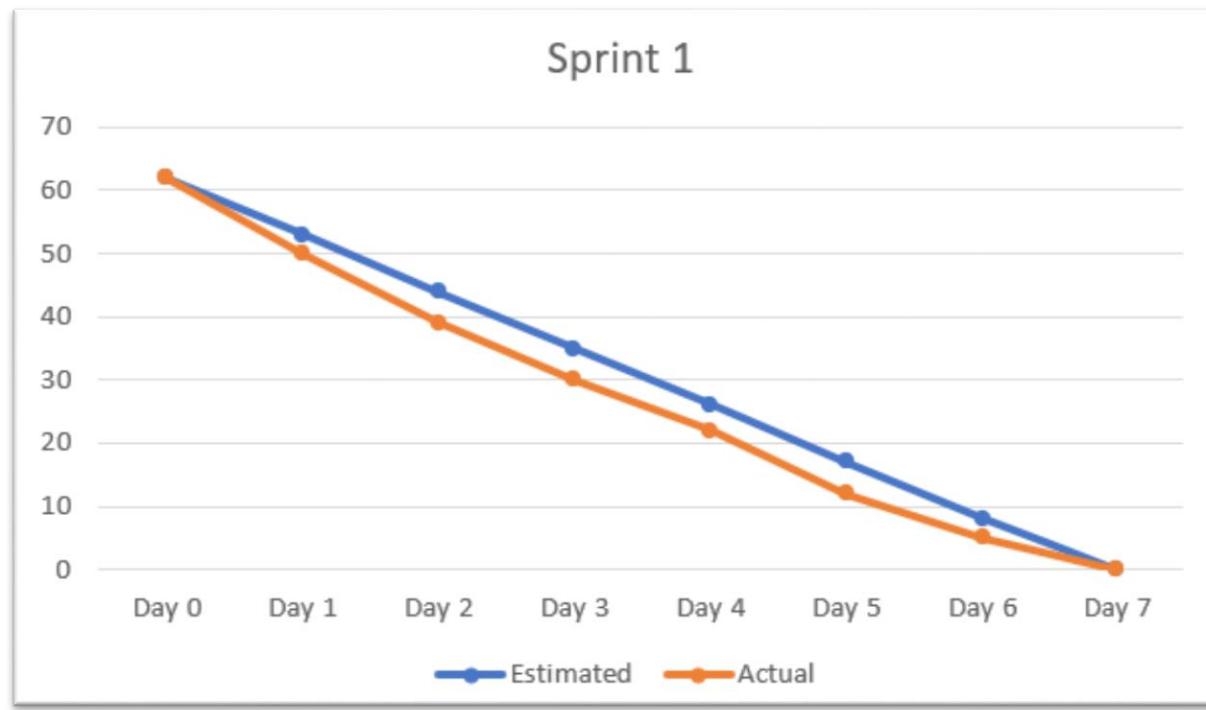


Figure 109: Burndown chart of sprint 1

6.2.1 Sprint 2

Table 12: Sprint 2

SPRINT 2												
ID	User's story	Task	Owner	Estimated hours	Day (27/2)	Day (28/2)	Day (1/3)	Day (2/3)	Day (3/3)	Day (4/3)	Day (5/3)	Status
2	Managers can add, edit, delete and review existing category in the system.	Code category interface	Vinh	12	3	0	3	2	2	2	0	Completed
		Database design + connect	Duy	9	3	0	2	2	2	0	0	Completed
		Code create category	Quan	10	0	3	0	4	3	0	0	Completed
		Code update category	Duy	8	0	3	0	0	0	3	2	Completed
		Code delete category	Phat	7	0	3	0	2	2	0	0	Completed
		Testing	Phat + Quan	10	3	0	2	0	0	2	3	Completed

Category is the category of the topic, a category can contain many topics, when creating a topic, you must choose a category for that topic, each topic can only choose one category.

We will first implement the interface code of the system as well as the interface of the category management page (front-end), then will design the database based on the previous designs to store the necessary data, next will be the functions in CRUD and finally will be the test to catch and fix the error.

All the goals that we had set for the Sprint have been successfully accomplished. We managed to achieve a significant portion of the goals set early on in the Sprint; however, towards the end, there was a slight slowdown where we had to make a conscious effort to balance and align ourselves with the original Sprint goal.

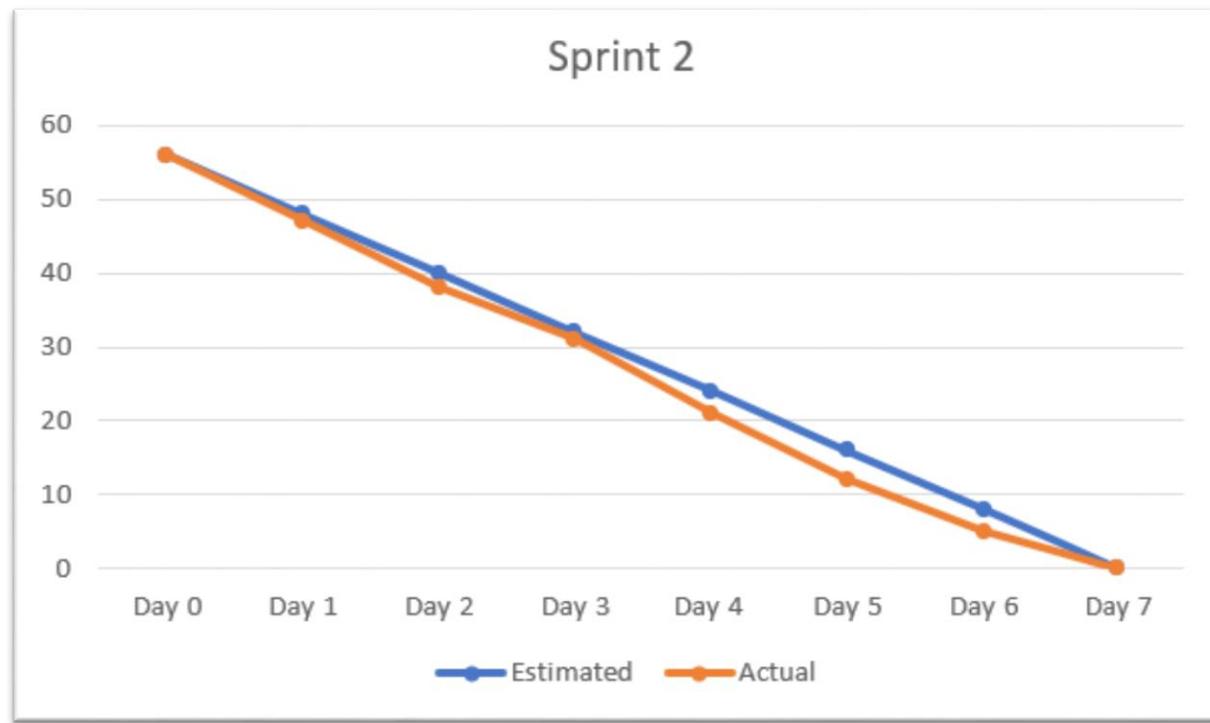


Figure 110: Burndown chart of sprint 2

6.2.3 Sprint 3

Table 13: Sprint 3

SPRINT 3												
ID	User's story	Task	Owner	Estimated hours	Day (6/3)	Day (7/3)	Day (8/3)	Day (9/3)	Day (10/3)	Day (11/3)	Day (12/3)	Status
3	Managers can add, edit, delete and review existing post in the system.	Code post interface	Vinh	11	3	3	0	0	2	0	3	Completed
		Database design + connect	Duy	9	3	0	2	2	0	0	2	Completed
		Code create post	Quan	8	3	3	0	2	0	0	0	Completed
		Code update post	Quan	7	0	0	2	0	3	2	0	Completed
		Code delete post	Phat	7	3	0	2	2	0	0	0	Completed
		Testing	Duy + Phat	10	0	2	0	0	2	3	3	Completed

Post are topics created by the QA Manager for a certain department and for a certain category, staff can comment on post belonging to their department.

We will first implement the interface code of the system (front-end), then will design the database based on the previous designs to store the necessary data, next will be the functions in CRUD and finally will be the test to catch and fix the error.

All the objectives of the Sprint were effectively met as intended. At the beginning of the Sprint, we were able to achieve a substantial part of the initial objectives swiftly. Nevertheless, towards the end, we experienced a slight deceleration, which required us to make a conscious effort to regain balance and align ourselves with the original Sprint goal.

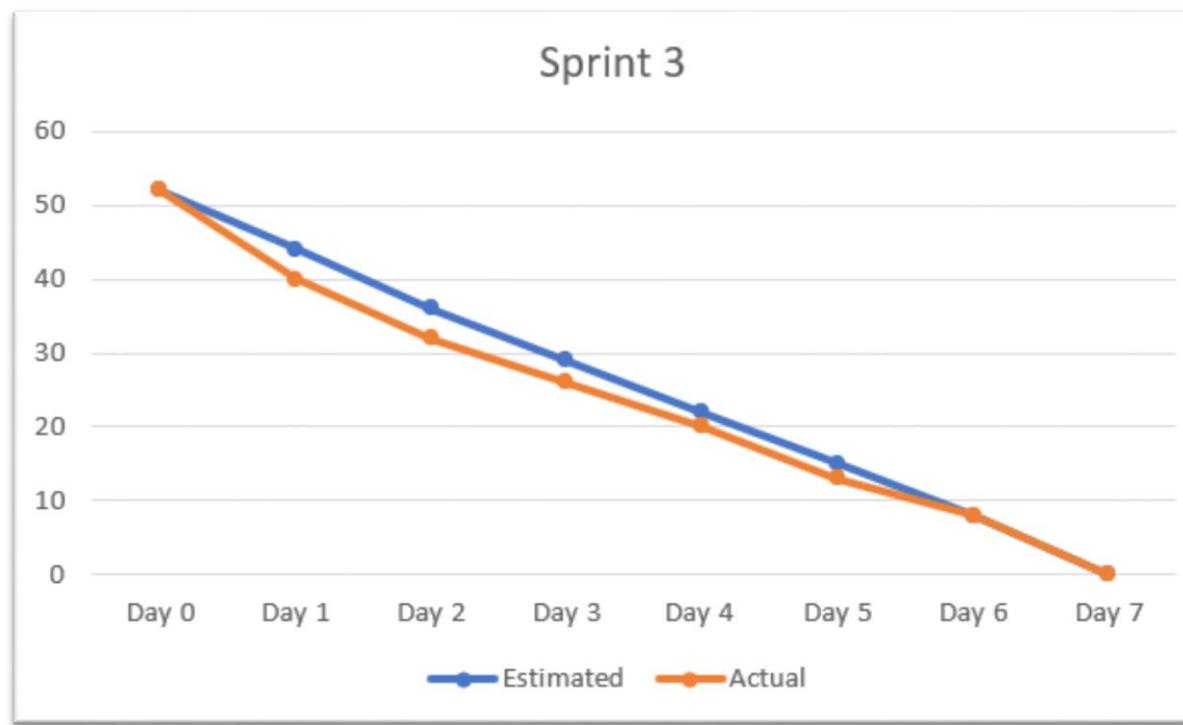


Figure 111: Burndown chart of sprint 3

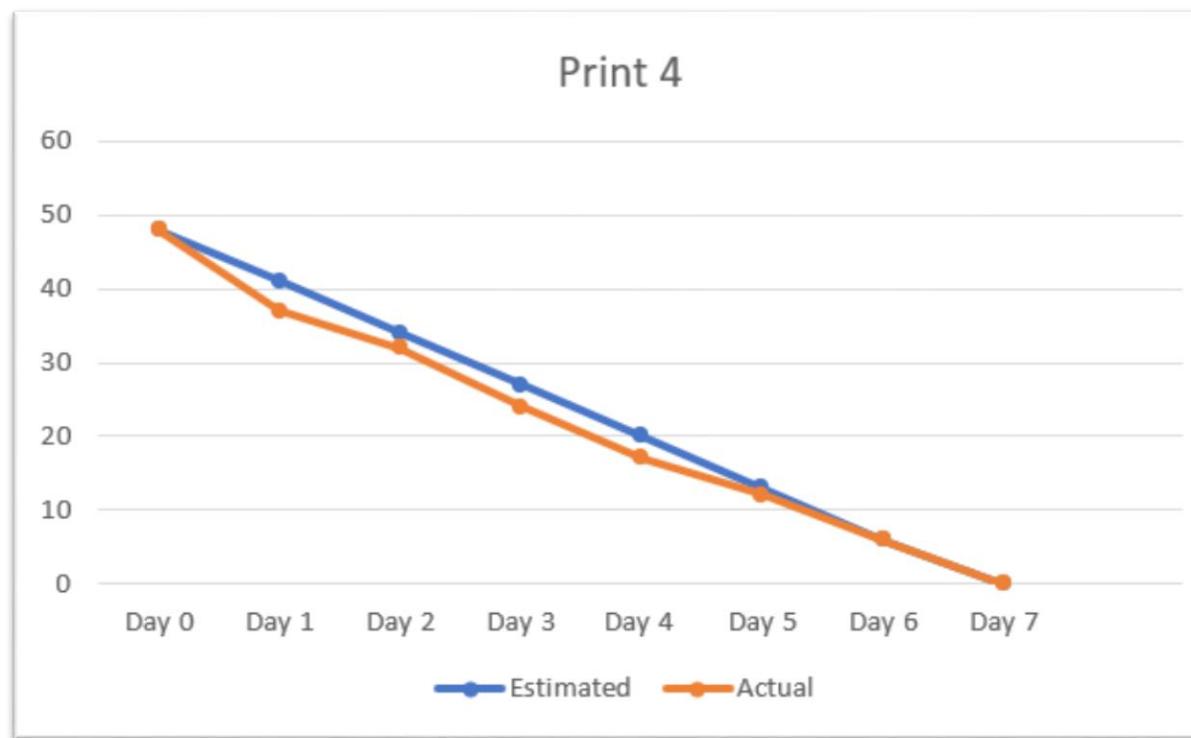
6.2.4 Sprint 4

Table 14: Sprint 4

SPRINT 4												
ID	User's story	Task	Owner	Estimated hours	Day (13/3)	Day (14/3)	Day (15/3)	Day (16/3)	Day (17/3)	Day (18/3)	Day (19/3)	Status
4	Managers can add, edit, delete and review existing staff (staff account) in the system.	Code staff interface	Vinh	10	3	0	2	3	3	0	0	Completed
		Database design + connect	Duy	8	2	0	2	2	0	0	2	Completed
		Code create staff	Quan	7	3	0	2	0	0	2	0	Completed
		Code update staff	Quan	6	0	2	0	2	2	0	0	Completed
		Code delete staff	Phat	7	3	0	2	2	0	0	0	Completed
		Testing	Duy + Phat	10	0	3	0	0	0	3	4	Completed

Staff is the manager staff in the company, the employee account will be created by the QA manager and given to the employee, they cannot create any for themselves. When creating an account, the QA Manager will select the department belonging to that employee.

We will first implement the interface code of the system (front-end), then will design the database based on the previous designs to store the necessary data, next will be the functions in CRUD and finally will be the test to catch and fix the error.



Estimated	48	41	34	27	20	13	6	0
Actual	48	37	32	24	17	12	6	0

Figure 112: Burndown chart of sprint 4

6.2.5 Sprint 5

Table 15: Sprint 5

SPRINT 5												
ID	User's story	Task	Owner	Estimated hours	Day (20-21/3)	Day (22-23/3)	Day (24-25/3)	Day (26-27/3)	Day (28-29/3)	Day (30-31/3)	Day (1-2/4)	Status
5	Manager can see all post added by them or another administrator	Code view post with admin account	Phat	15	3	4	0	4	4	0	0	Completed
		Testing	Quan	12	3	3	0	4	2	0	0	Completed
6	Manager can see all comments of all feedback	Code comment feedback with admin account	Duy	18	2	3	4	3	3	3	0	Completed
		Database design + connect	Duy	8	3	2	2	0	0	0	0	Completed
		Testing	Quan	12	0	0	3	0	2	3	4	Completed
7	All users will receive a notification when someone comments	Coded notification for user	Vinh	24	4	4	4	3	2	3	4	Completed
		Testing	Phat	8	0	0	2	0	0	3	3	Completed

	on their feedback										
--	-------------------	--	--	--	--	--	--	--	--	--	--

In Sprint 5, the time is 2 weeks for a Sprint, unlike the first 4 prints because its work is more complicated. In Print 5, we proceed to perform the necessary functions. In Sprint 5 and 6, the time is 2 weeks for a Sprint, unlike the first 4 prints because the work is more complicated. In Print 5, we proceed to perform the necessary functions. Almost in week 5 will be functions for QA Manager.

Viewing post will be 2 different modes, for QA manager, they can see all topics, followed by the comment section with each topic, almost everything is the same for staff. The next function we do in this sprint is the notification function, which is probably the most difficult function in this system.

For these functions, there is no need to create a new database because it will mostly be based on the databases created from that, first the code view post for the QA Manager account and its interface, then perform the testing step. , next will be the comment feedback code and create a database to store comments and likes of that comment, next will be testing. Next will be the functional code that notifies all users and performs testing.

The Sprint's objectives were successfully accomplished according to plan. We rapidly achieved a considerable portion of the initial objectives during the Sprint's onset. However, towards the end, we experienced a minor deceleration, which called for a deliberate attempt to restore equilibrium while staying true to the original Sprint goal.

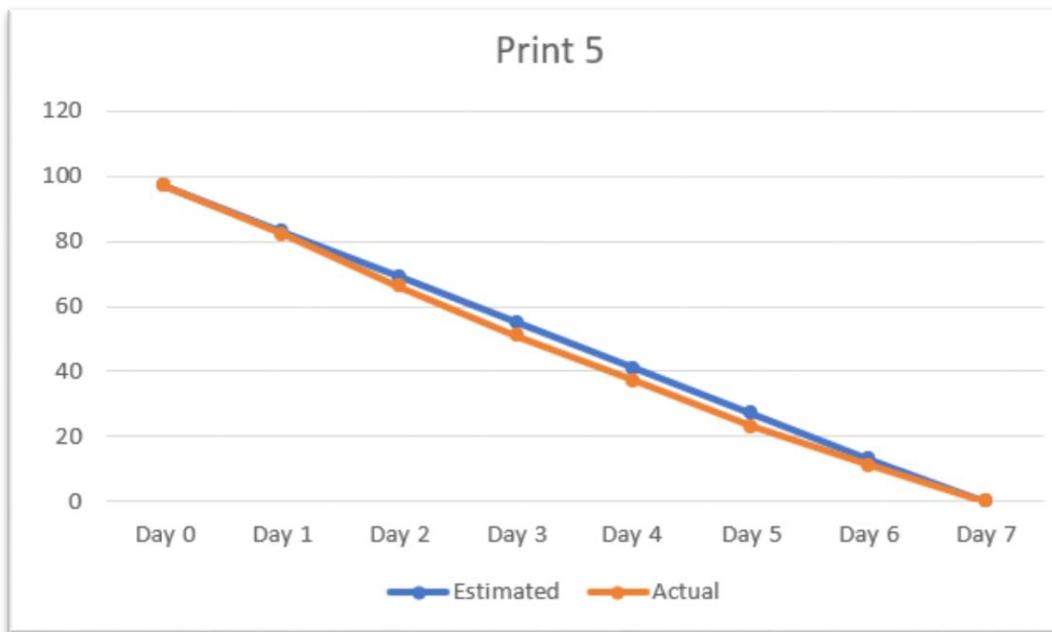


Figure 113: Burndown chart of sprint 5

6.2.6 Sprint 6

Table 16: Sprint 6

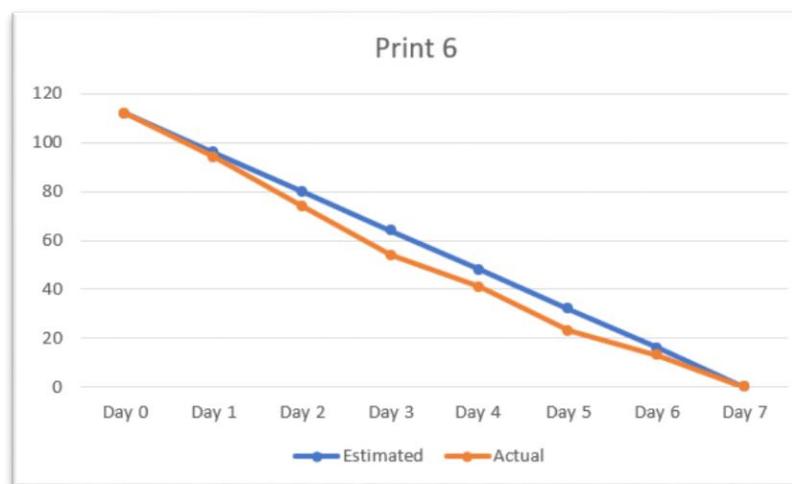
SPRINT 6												
ID	User's story	Task	Owner	Estimated hours	Day (3-4/4)	Day (5-6/4)	Day (7-8/4)	Day (9-10/4)	Day (11-12/4)	Day (13-14/4)	Day (15-16/4)	Status
8	Staff can view post and feedback based on their department	Code view post with staff account	Phat	14	4	5	4	0	1	0	0	Completed
		Code view feedback	Phat	10	2	0	3	0	5	0	0	Completed
		Testing	Quan + Duy	14	5	0	0	0	5	0	4	Completed
9	Staff can post feedback on post they can see	Code post feedback	Vinh	15	3	3	0	3	2	0	4	Completed
		Testing	Quan	10	4	4	2	0	0	0	0	Completed
10	Staff can comment and like on feedback they can see	Code comment feedback with staff account	Vinh	12	0	0	4	3	2	3	0	Completed
		Testing	Phat + Duy	15	0	0	3	3	0	4	5	Completed
11	They can update some basic information	Code update staff account	Quan	12	0	3	4	2	0	3	0	Completed
		Testing	Duy	10	0	5	0	2	3	0	0	Completed

	about themselves										
--	------------------	--	--	--	--	--	--	--	--	--	--

For Staff, they only see topics in their department and post feedback related to that post, they can comment on those feedbacks and drop the "Like" expression on the comments, Staff can Update some basic information based on the account created previously by QA Manager.

For the staff topic view function, they need the developer to divide the department, we will not conduct testing at this time but will continue to code the feedback view and then do the testing. Next will be the post feedback code, this is the most important part of the system. The last 2 functions are the function of commenting feedback and updating the staff's account.

The goals of the Sprint were successfully completed as expected. We quickly met a sizable chunk of the basic goals when the Sprint first started. We did, however, encounter a slight slowing at the conclusion, which required a concerted effort to restore equilibrium while adhering to the original Sprint objective.



Estimated	112	96	80	64	48	32	16	0
Actual	112	94	74	54	41	23	13	0

Figure 114: Burndown chart of sprint 6

6.3 Minute of daily meeting

The Daily Scrum is a short 15-minute meeting where the development team comes together to review their progress towards achieving the Sprint goal and assess their performance in accomplishing tasks listed in the Sprint backlog.

6.3.1 Sprint 1

Table 17: Daily meeting sprint 1

Date	Hours	Attendance
20-2-2023	14:00 – 15:00	4/4
Meeting face to face		
Topics		
Discuss the tasks to be done and divide the work		
Bringing out the best technology for each person's work		
Tasks		
Where will the team leader divide the work for 2 weeks		
Members have to give their own opinions		

In the first group meeting is the first day of work. We need to put the work that needs to be done on this project, and come up with the latest and most appropriate tools and technology to complete this project. Besides, divide each way of work for team members.

6.3.2 Sprint 2

Table 18: Daily meeting sprint 2

Date	Hours	Attendance
5-3-2023	14:00 – 15:00	4/4
Meeting face to face		
Topics		
Discuss the mistakes you are making		
Perform a check of previous jobs		
Tasks		
We'll be looking at work progress in both the previous Sprint and the current Sprint.		
Report and fix errors during work		

In the second meeting, we checked how far each member has done and what needs to be done next. Members will report their work progress and difficulties as well as things that cannot be done and cannot be completed, team members will give the best solution for that.

6.3.3 Sprint 3

Table 19:Daily meeting sprint 3

Date	Hours	Attendance
12-3-2023	8:00 – 9:00	3/4
Meeting face to face		
Topics		
Discuss problems encountered in the working process		
Discuss the next directions		
Tasks		
The team leader will check the progress of the members and urge them to speed up the work		
Double check the functions that have been implemented in 3 weeks		

The next meeting will not be too much work for us to do, simply report the status of each member and the team leader will proceed to urge his team members and conduct an inspection. simple functions completed.

6.3.4 Sprint 4

Table 20:Daily meeting sprint 4

Date	Hours	Attendance
18-3-2023	12:00 - 13:30	4/4
Meeting face to face		
Topics		
Discuss problems encountered in the working process		
Tasks		
Let's finish the unfinished work together		
Proceed to identify the next step (Sprint 5-Sprint 6) of complex jobs		

The meeting in print 4 was quite important while we had to finish all the unfinished business to move to the next stage. Proceed to reassign the work according to what the previous group has noticed in the members during the previous work to do the next work.

6.3.5 Sprint 5

Table 21:Daily meeting sprint 5 (1)

Date	Hours	Attendance
25-3-2023	16:00 – 17:00	3/4
Meeting face to face		
Topics		
Discuss what needs to be corrected		
Come up with a solution to the problem		
Tasks		
Members will discuss with each other about having difficulty in coding or performing their assigned work and other members will continue to help and overcome.		

In this Sprint, there will be 2 meetings, this is the first meeting, with new jobs, the complexity is considered higher than the previous ones, so it is necessary to discuss the mistakes that all members have. are experiencing and work on fixing those errors.

Table 22:Daily meeting sprint 5 (2)

Date	Hours	Attendance
31-4-2023	16:00 – 17:00	3/4
Meeting face to face		
Topics		
Report on assigned work		
Connecting related things together		
Tasks		
Each member in turn will report their work done to other members.		
The other members will have to supplement each other's unfinished work		
Proceed to assemble all system functions		

The members will report on the work they have been assigned, if not completed, the group will have to discuss and solve those tasks. After that, it will continue to connect the jobs from the first week to now.

6.3.6 Sprint 6

Table 23: Daily meeting sprint 6

Date	Hours	Attendance
7-4-2023	16:00 – 17:00	3/4
Meeting face to face		
Topics		
Discuss the mistakes you are making		
Perform a check of previous jobs		
Tasks		
We'll be looking at work progress in both the previous Sprint and the current Sprint.		

Table 24: Daily meeting sprint 6 (2)

Date	Hours	Attendance
16-4-2023	16:00 – 18:00	4/4
Meeting face to face		
Topics		
Completed work		
Discussion about the project		
Tasks		
Proceed to complete the unfinished business and complete the project.		
Discuss difficulties and lessons learned from completing this project		

We'll have to do our quick work to wrap up this project. First, the members must report their own work that has been assigned to conduct re-checking for each error, if there are unfinished things, the whole group must sit down to complete that work. Then each member will talk about their difficulties when working in the group as well as in the project, the experiences they have learned this time.

7. References

Asana, T. (2022, December 19). What is a sprint backlog? How to create one, with examples. Retrieved from asana: <https://asana.com/resources/sprint-backlog>

Peek, S. (2023, february 22). What Is Agile Scrum Methodology? Retrieved from businessnewsdaily: <https://www.businessnewsdaily.com/4987-what-is-agile-scrum-methodology.html>

Raeburn, A. (2022, november 29). What is a product backlog? (And how to create one). Retrieved from asana: <https://asana.com/resources/product-backlog>