

**STUDENT MANAGEMENT PROJECT**

**Software Requirement Specification**

Lecturer: ThS. Trịnh Huy Hiệp

Class: SE1807

***Member***

*Nguyễn Tăng Tài Phát (Leader)*

*Lưu Thế Vinh*

*Thái Bình Dương*

*Nguyễn Như Tài*

*Nguyễn Trung Kiên*

*Trần Anh Toàn*

*Lê Đức Lộc*

– Ho Chi Minh, May 2024 –

# Record of changes

|  |  |  |  |
| --- | --- | --- | --- |
| Date | A\* M, D | In charge | Change Description |
| 21/5 | A |  |  |
| 21/5 | A |  |  |
| 21/5 | A |  |  |

\*A - Added M - Modified D - Deleted

# Table of Contents

[Table of Contents 3](#_Toc162472549)

[I. Introduction 5](#_Toc162472550)

[1. Business Background 5](#_Toc162472551)

[2. Business Context 5](#_Toc162472552)

[3. Major Feature 6](#_Toc162472553)

[4. User Requirement 7](#_Toc162472554)

[4.1. Use case 7](#_Toc162472555)

[4.2. Actor list 7](#_Toc162472556)

[5. Assumptions & Dependencies 8](#_Toc162472557)

[6. Limitations and Exclusions 8](#_Toc162472558)

[7. Business Rules 8](#_Toc162472559)

[II. Use Case Specifications 9](#_Toc162472560)

[1. User detail feature 9](#_Toc162472561)

[2. Class detail feature 11](#_Toc162472562)

[3. Subject list feature 13](#_Toc162472563)

[4. Team detail feature: 14](#_Toc162472564)

[5. Iteration detail feature 16](#_Toc162472565)

[III. Functional Requirements 17](#_Toc162472566)

[1. System Functional Overview 17](#_Toc162472567)

[1.1 Screens Flow 18](#_Toc162472568)

[1.2 Screen Descriptions 19](#_Toc162472569)

[1.3 Screen Authorization 21](#_Toc162472570)

[1.4 Non-Screen Functions 21](#_Toc162472571)

[1.5 Entity Relationship Diagram 22](#_Toc162472572)

[2. Login and Register 23](#_Toc162472573)

[2.1. Login 23](#_Toc162472574)

[2.2. Register 24](#_Toc162472575)

[3. Setting 24](#_Toc162472576)

[4. User list and detail 25](#_Toc162472577)

[4.1. Function trigger 25](#_Toc162472578)

[4.2. Function description 25](#_Toc162472579)

[4.3. Screen layout 25](#_Toc162472580)

[4.4. Function detail 26](#_Toc162472581)

[5. Class list and detail 26](#_Toc162472582)

[5.1. Function trigger: 26](#_Toc162472583)

[5.2. Function description: 26](#_Toc162472584)

[5.3. Screen layout: 26](#_Toc162472585)

[5.4. Function Details 27](#_Toc162472586)

[6. Subject list and detail 27](#_Toc162472587)

[6.1. Function trigger: 27](#_Toc162472588)

[6.2. Function description: 27](#_Toc162472589)

[6.3. Screen layout: 27](#_Toc162472590)

[6.4. Function Details: 28](#_Toc162472591)

# I. Introduction

## 1. Business Vision

The Bakery Management System (BMS) is a groundbreaking solution designed to revolutionize the bakery industry's operational landscape. In a world empowered by the BMS, bakeries seamlessly integrate digital technologies into their daily operations, offering customers a modern and convenient ordering experience. Through an intuitive online platform accessible via web and mobile devices, customers can effortlessly place orders for their favourite bakery items from anywhere, anytime. Whether it's a customized cake for a special occasion or a daily selection of freshly baked goods, the BMS ensures that customers have access to a diverse range of bakery offerings at their fingertips. By leveraging advanced order processing and delivery management capabilities, the BMS optimizes bakery operations, reducing wait times and ensuring timely delivery of orders. Bakery staff are equipped with powerful tools to efficiently manage inventory, process orders, and engage with customers, enhancing overall productivity and service quality.

## 2. Business Context

### 2.1. Stakeholder Profiles

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Stakeholder | Major Value | Attitudes | Major Interests | Constraints |
| Bakery Owner/  Manager | Streamlined bakery operations; increased efficiency and profitability | Open to innovation, but cautious about disruptions to existing workflows | Cost-effectiveness; improved inventory management; enhanced customer service | Limited budget for implementation; concerns about staff training and adoption |
| Bakery Staff | Simplified work processes; reduced manual tasks | Open to change, but concerned about job security and workload implications | Job satisfaction; efficient use of time and resources | Training and support needed for transitioning to the new system; potential resistance to technology adoption |
| Customers | Expanded product offerings; convenience | Excited about new offerings, but may have concerns about pricing and quality | Variety of bakery items; ease of ordering and payment | Accessibility issues for customers without internet or mobile devices; concerns about privacy and security of personal information |
| Suppliers | Increased business opportunities; streamlined transactions | Supportive of initiatives that improve efficiency and communication with the bakery | Timely payments; clear order specifications | Integration challenges with existing supplier systems; potential resistance to changes in ordering and invoicing processes |
| IT Department | Enhanced system performance and security | Supportive of technology upgrades, but cautious about system compatibility and maintenance | Scalability; data integration with existing systems | Limited resources for system implementation and maintenance; concerns about data security and compliance with regulations |

### 2.1. Priorities

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dimension | Constraint | Driver | Degree of Freedom | Dimension |
| Features | All features scheduled for release 1.0 must be fully operational | Fulfillment of bakery operational needs and customer requirements | Flexibility in prioritizing features based on criticality and resource availability | Features |
| Quality | 95% of user acceptance tests must pass; all security tests must pass | Ensuring reliability, usability, and security of the BMS | Margin for addressing any issues identified during testing; continuous improvement | Quality |
| Schedule | Release 1 planned to be available by end of Q1 of next year, release 2 by end of Q2; overrun of up to 2 weeks acceptable without sponsor review | Timely delivery to meet bakery's operational needs and market demands | Buffer for unexpected delays or challenges; proactive monitoring and mitigation strategies | Schedule |
| Cost | Budget overrun up to 15% acceptable without sponsor review | Cost-effectiveness and financial viability | Contingency measures to manage unforeseen expenses; optimization of resource allocation | Cost |
| Staff | Team size is half-time project manager, half-time BA, 3 developers, and 1 tester; additional developer and half-time tester available if necessary | Adequate resource allocation and skill set diversity for project execution | Flexibility to scale team based on project requirements and workload fluctuations | Staff |

## 3. Major feature

**FE-1:** Users can order and pay for meals from the bakery's menu for pickup or delivery.

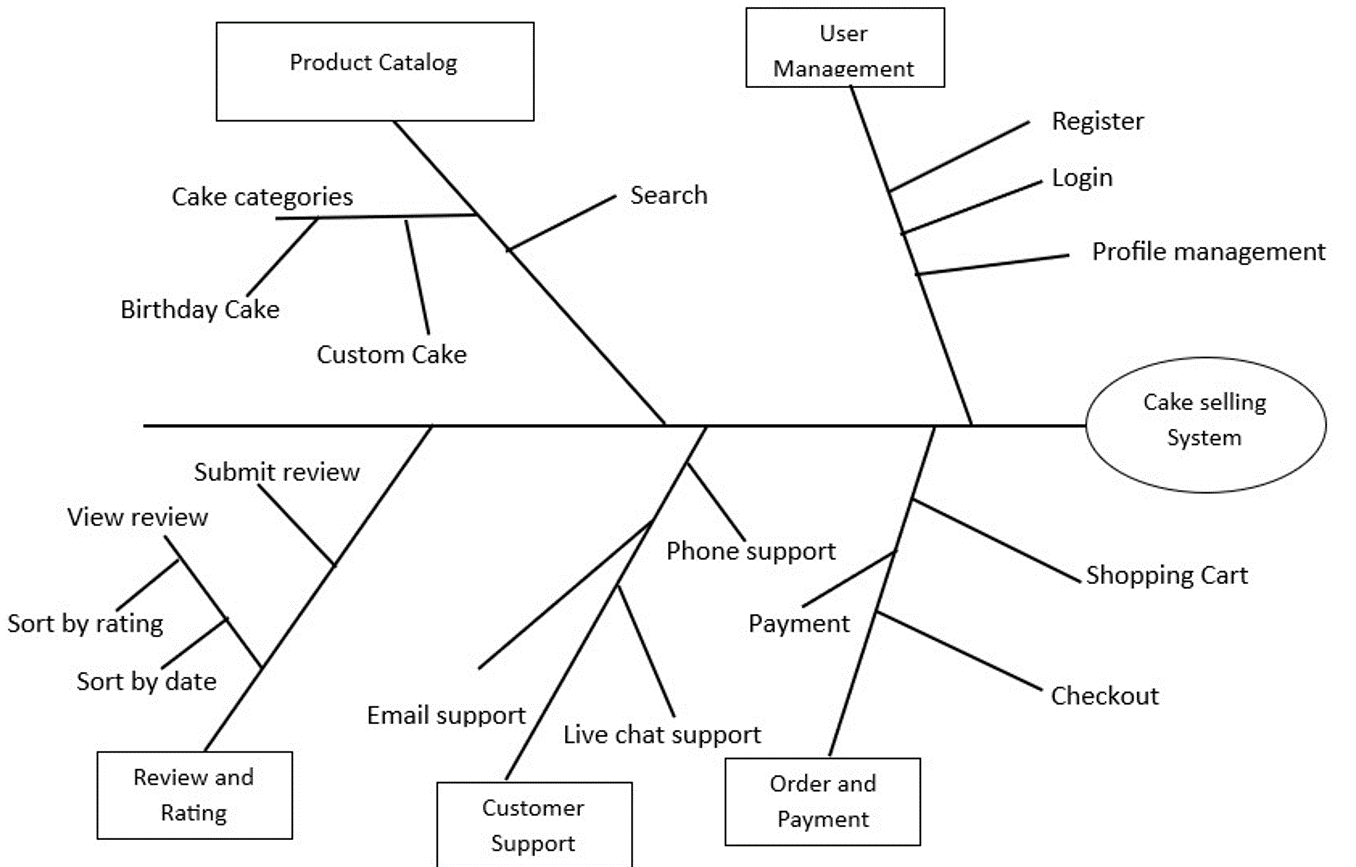
**FE-2:** Users can order and pay for meals from local restaurants for delivery.

**FE-3:** Users can manage subscriptions for standing or recurring meal orders, including special daily meals.

**FE-4:** Admins can create, view, modify, delete, and archive bakery menus.

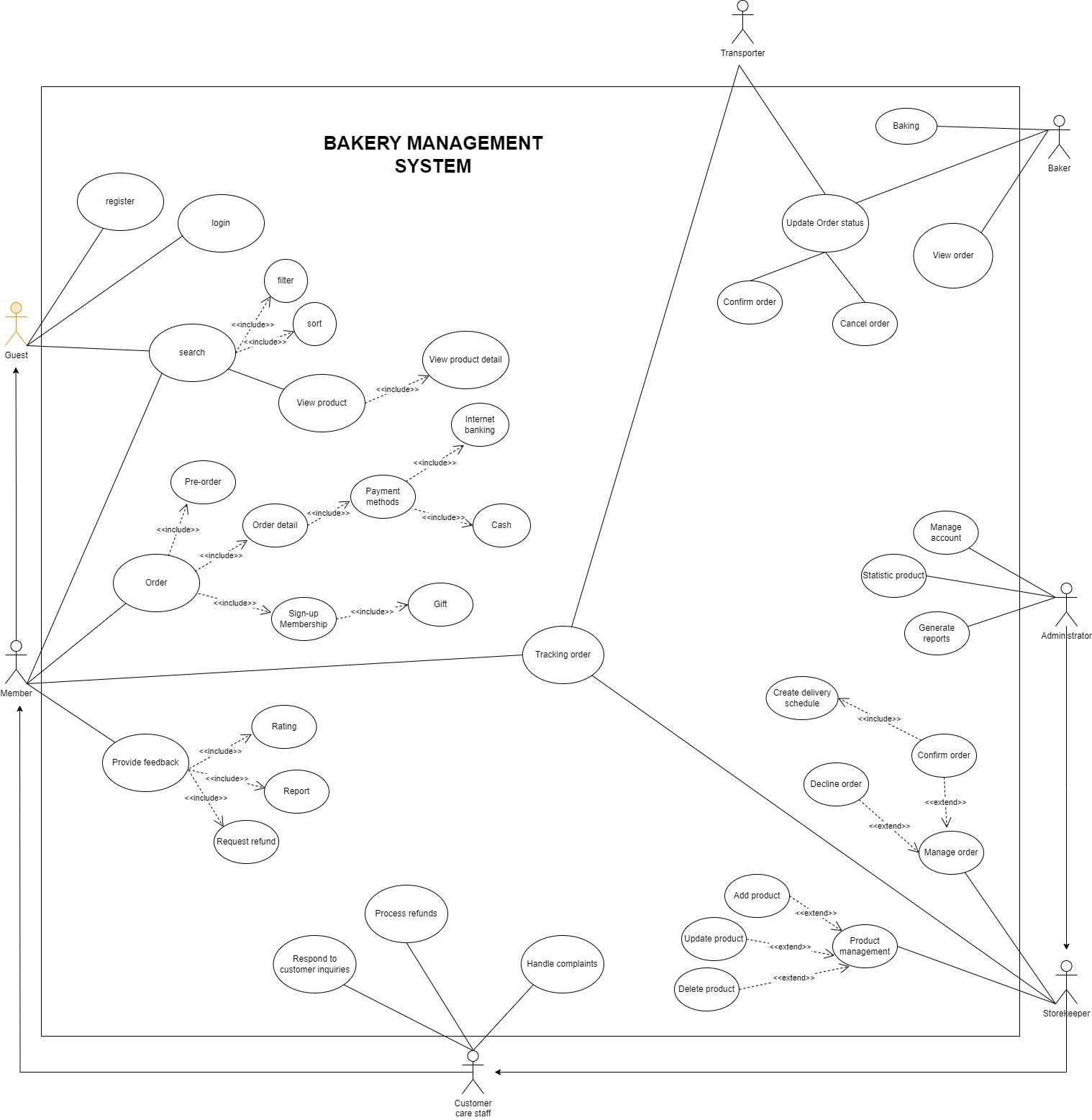
**FE-5:** Users can access ingredient lists and nutritional information for bakery menu items.

**FE-6:** System access is available through corporate intranet, smartphones, tablets, and the internet for authorized employees.



## 4. User Requirement

### 4.1. Use Case Diagram



### 4.2. Actor list

|  |  |  |
| --- | --- | --- |
| No | Actor | Description |
| 1 | Admin | User management, monitoring and website security, making a new class for subject managers and their students. |
| 2 | Subject  manager | Push content like projects, assignments for students. Make a deadline for students. Grading for student's project, comment in student’s project. Manage the quantity of submitted students. |
| 3 | Trainer | View the whole students in one class, download submitted projects of students, comment for students. |
| 4 | Student | View their score (LOC) on the dashboard. They can only view their information. See the time left on content, assignment of the subject manager created. Can push projects by .rar file to server, in running time they can update or delete their own project. |

## 5. Assumptions & Dependencies

* Users as role students can upload files to submit projects for lecture to grade and comment.
* Teachers can create deadlines for students, after the required time students can’t do any work on this process.

## 6. Limitations and Exclusions

* Users may require a real-time system counting down from the website.
* Users may require a larger file than file can accept.
* Users may need to rename a file or a note for their project.
* Users may out session in a long time not working.
* Student account just mush be a school’s email account

## 7. Business Rules

|  |  |
| --- | --- |
| **ID** | **Rule Definition** |
| |  | | --- | | BR-01 |  |  | | --- | |  | | Some features such as viewing order history and browsing the product catalog should be available even without an internet connection. |
| |  | | --- | | BR-02 |  |  | | --- | |  | | Customers must provide valid payment information at the time of order placement.   |  | | --- | |  | |
| |  | | --- | | BR-03 |  |  | | --- | |  | | Orders cannot be cancelled or modified once they reach the "Baking" status.   |  | | --- | |  | |
| |  | | --- | | BR-04 |  |  | | --- | |  | | Inventory levels must be updated in real-time as orders are placed and processed   |  | | --- | |  | |
| |  | | --- | | BR-05 |  |  | | --- | |  | | Discount codes must be applied at the time of checkout, and only one discount code can be used per order.   |  | | --- | |  | |
| |  | | --- | | BR-06 |  |  | | --- | |  | | Refunds are only available for orders that have not yet been started ("Pending" status).   |  | | --- | |  | |
| |  | | --- | | BR-07 |  |  | | --- | |  | | Delivery options must be chosen at the time of order placement, with delivery times being estimated and not guaranteed.   |  | | --- | |  | |
| |  | | --- | | BR-08 |  |  | | --- | |  | | Customers have the right to request deletion of their personal data from the system in compliance with data protection regulations. |

# II. Use Case Specifications

## 1. User detail feature

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID and Name: | UC-01 User detail | | | |
| Created By: |  |  |  |  |
| Primary Actor: |  | | | |
| Description: |  | | | |
| Trigger: |  | | | |
| Preconditions: |  | | | |
| Postconditions: |  | | | |
| Normal Flow: |  | | | |
| Exceptions: |  | | | |
| Priority: |  | | | |
| Frequency of Use: |  | | | |
| Business Rules: |  | | | |
| Other Information: |  | | | |
| Assumptions: |  | | | |

## 2. Class detail feature

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: |  | | |
| Created By: |  |  |  |
| Primary Actor: |  |  |  |
| Description: |  | | |
| Trigger: |  | | |
| Preconditions: |  | | |
| Postconditions: |  | | |
| Normal Flow: |  | | |
| Alternative Flows: |  | | |
| Exceptions: |  | | |
| Priority: |  | | |
| Frequency of Use: |  | | |
| Business Rules: |  | | |
| Other Information: |  | | |
| Assumptions: |  | | |

## 3. Subject list feature

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | UC-03 Subject list feature | | |
| Created By: | [Nguyen Minh Duc (K15 HL)](mailto:ducnmhe150901@fpt.edu.vn) | Date Created: | 30/05/2022 |
| Primary Actor: | Student | Secondary Actors: | Admin |
| Description: | When the user accesses the system and wants to see the list of all the subjects the system has. The system will respond to the user with a list of subjects with any information of this subject they want to see. The system also has the search function to help the user easier to find the subject they want to join in. If the user has the high permission like admin, they can add new subjects, update or delete subjects. Admin have more functions than normal users. Admin can see all the subjects including the subjects that are not active. And admin can modify things like add, update information or delete subjects. | | |
| Trigger: | A student wanna join to a subject | | |
| Preconditions: | PRE-1. That student logged in to the system.  PRE-2. That user account has the role and status of a student and the access to accept to learn that subject. | | |
| Postconditions: | POST-1. Are the subjects in the status of working.  POST-2. Did the subject have too many class and don’t have enough lecture to teach  POST-3. Did student done all the precondition subject what that subject required | | |
| Normal Flow: | 1.0 Join a Subject.  1. User asks to view the menu for all subjects.  2. System will check out the status and role of users.  3. System displays a menu of all the subjects.  4. Users will find and select one of all subjects from the menu.  5. System check out is the subject able to join.  6. System will check out all the classes that can be accessed in that subject and respond to the user. | | |
| Alternative Flows: | 1.1 User does not have permission to join subject  1. System respond to user a alert make them know the account does not in a status able to access  2. Return to the menu of subjects.  1.2 User did not login  2. Return to the login page. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 2000 users, average of one usage per day. Always ready to be access for everyone | | |
| Business Rules: |  | | |
| Other Information: | 1. Student can un-enrol anytime they want if the join in wrong subject  .  2.Subject List will list all the related information | | |
| Assumptions: | Assume about 1000 student enrol in a subject in the start of the course | | |

## 4. Team detail feature:

|  |  |  |  |
| --- | --- | --- | --- |
| ID and Name: | UC-04 Team detail | | |
| Created By: | Đan Công Duy | Date Created: | 30/5/2022 |
| Primary Actor: | Trainer | Secondary Actors: | Student, Author |
| Description: | Trainer accesses the system from the corporate intranet or from home, selects classes, and creates a team for students. | | |
| Trigger: | Trainer wants to create a new team or change team members. | | |
| Preconditions: | PRE-1. Trainer is logged into the system. | | |
| Postconditions: | POST-1. Trainer created a new team for students. | | |
| Normal Flow: | 1.0 Create a new team  1. Trainer views the list of students in a class.  2. System displays a list of students.  3. Trainer will select one or more students from the list to create a new team.  4. Trainer confirms the select (continue) or modify team member (return step 3).  5. System displays the team which was newly created. | | |
| Alternative Flows: | 1.1 Create a new team similar to an existing team  1. Trainer creates a new team similar to the existing team.  2. System notice error.  3. System will display a list of students. | | |
| Exceptions: | 1.1. User not logged into system.   1. System will notify the user to login. 2. System will return to the login page.   1.2. If a student is in a team   1. System notice error. 2. Users need to create a different team. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 1000 users, average of one usage per day. Peak usage load for this use case is between 9:00 A.M. and 16:00 P.M. local time. | | |
| Business Rules: | BR-2, BR-5, BR-7, BR-12 | | |
| Other Information: | 1. Trainer shall be able to cancel the creating team at any time. 2. Only the Trainer can edit the team. 3. Students can view their team after the Trainer creates the team. | | |
| Assumptions: | Assume that the system will automatically delete a team when a member of the team leaves class. | | |

## 5. Iteration detail feature

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID and Name: | UC-05 Edit Iteration | | | |
| Created By: | Thái Minh Quang | Date Created: | 30/5/2022 |  |
| Primary Actor: | Author | | | |
| Description: | A Author accesses the Project Management System from the corporate intranet or from home, list all the iteration that author can edit, add , remove, update iteration | | | |
| Trigger: | Author can edit the iteration as updating, deleting, adding,... | | | |
| Preconditions: | PRE-1. Author will login to the system.  PRE-2. Author choose iteration list and then update, add, or delete the iteration | | | |
| Postconditions: | POST-1. After updating, adding, deleting,... data will be saved into the database. Then this will be showed to student | | | |
| Normal Flow: | 1.0 Author will give a fixed amount of time for students to do.  Add, delete and update | | | |
| Alternative Flows: | 1.1 Author can create multiple iterations for students in a given time period  Students can ask the teacher to edit the time | | | |
| Exceptions: | 1.0. If students submit their assignment without clicking on the submit button, that assignment can be rejected. | | | |
| Priority: | medium | | | |
| Frequency of Use: | Only authors can access the system. And they can access whenever they want | | | |
| Business Rules: | BR-1, BR-2, BR-3, BR-4, BR-11, BR-12, BR-14, BR-15 | | | |
| Other Information: | 1. Students can comment on the iteration that they do | | | |
| Assumptions: | Assume system will automatically follow the history of the iteration that student have done | | | |

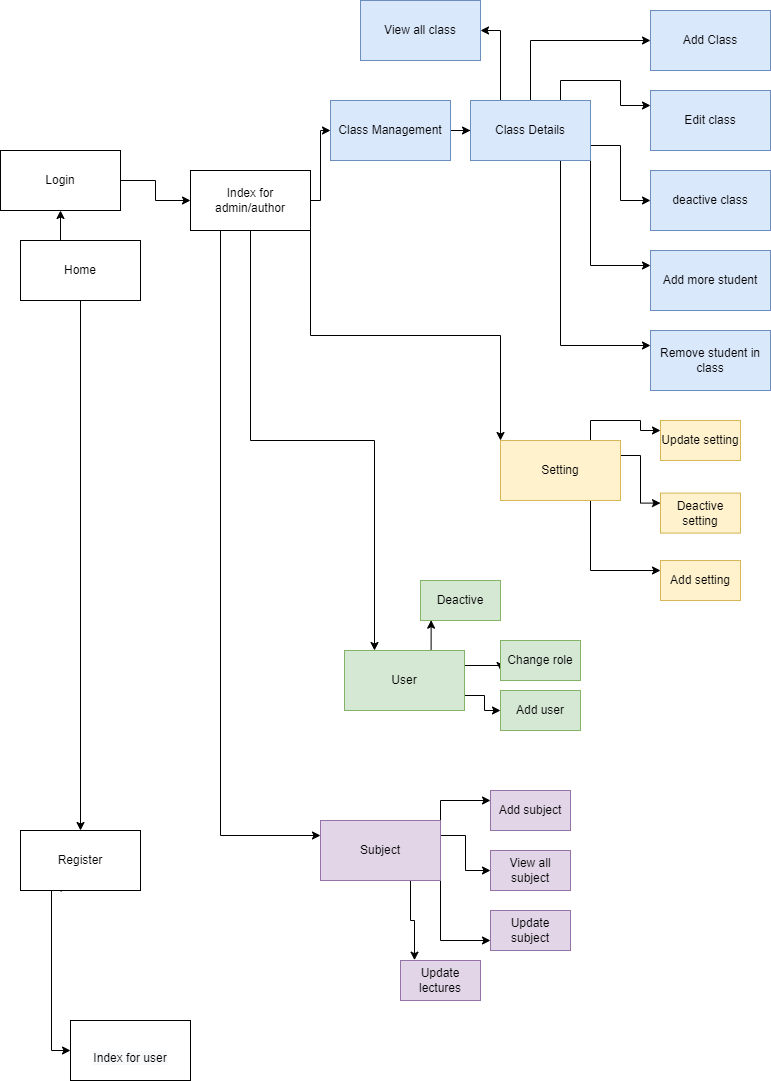
# III. Functional Requirements

## 1. System Functional Overview

[Provide functionality overview of software system: screen flow, screen descriptions, system user roles, screen authorization, non-screen functions, ERD]

### 1.1 Screens Flow

When a user login into the system, the system will check if his account is admin or not. If they are admin, they have permission to do something special with data like: class data, subject data,… They can see details and change anything on this function role.



After logging in as admin, admin can go to the settings list page to view and manage general configuration settings for the whole system. Admin can add, edit, delete settings to optimise the system

### 1.2 Screen Descriptions

|  |  |  |  |
| --- | --- | --- | --- |
| No | Feature | Screen | Description |
| 1 | Login | login | user login to system |
| 2 | Register | register | user create new account |
| 3 | Setting list | View All Settings | All Setting will display on the screen of setting list |
| 4 | Setting detail | Add setting | Add a new setting |
| 5 | Setting detail | View setting detail | View detail of each setting |
| 6 | Setting detail | Edit setting detail | admin can update or delete setting |
| 7 | User List | List all users | admin can view all users in database |
| 8 | User List | search user | admin can search user by full name or roll number |
| 9 | User List | Add new User | admin can add a new user |
| 10 | User List | change status, role | admin can change status and role of user |
| 11 | User detail | update user | admin can edit information of each user |
| 12 | Class List | list all class | view all class |
| 13 | Class List | search class | search class by class code |
| 14 | Class List | add new class | admin and author can add a new class |
| 15 | Class Detail | update class | edit information of each class |
| 16 | Subject list | list all subject | view all subject |
| 17 | Subject list | search | search subject by subject name, subject code |
| 18 | Subject list | add | add a new subject |
| 19 | Subject Detail | update | edit information of each subject |

### 1.3 Screen Authorization

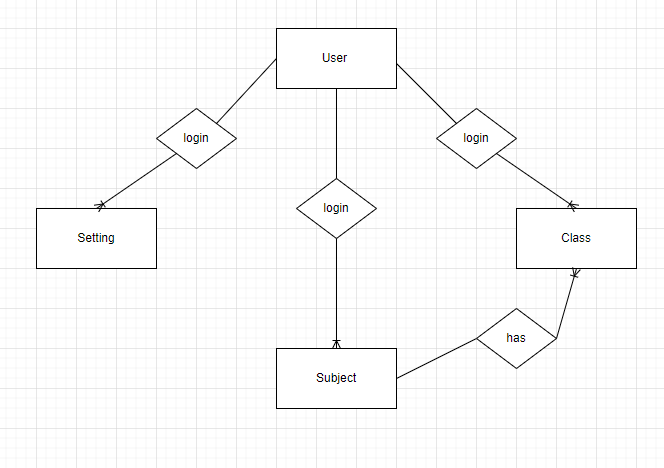
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Screen | Admin | Author | Trainer | Student |
| Login | X | X | X | X |
| Reset password | X | X | X | X |
| Change password | X | X | X | X |
| Register |  |  |  | X |
| Setting detail | X |  |  |  |
| Setting list | X |  |  |  |
| User detail | X |  |  |  |
| User list | X |  |  |  |
| Class detail | X | X | X |  |
| Class list | X | X | X |  |
| Subject detail | X |  |  |  |
| Subject list | X |  |  |  |

### 1.4 Non-Screen Functions

|  |  |  |  |
| --- | --- | --- | --- |
| No | Feature | System Function | Description |
| 1 | Search | Search | Search items in the list, then will display on the screen help boost user experience |
| 2 | Order, Filter | Order, Filter | Make all the list has filter help boost user experience |
| 3 | Paging | Paging | Paging for the list help boost efficiency |
| 4 | Authorize | Authorize | Return the login page if user did not login help boost security |
| 5 | Remember Account | Remember Account | Make the web remember the account, does not need to re login help boost user experience |
| 6 | Remember Input | Remember Input | Make the website remember any input you text in until the user delete them help boost user experience |

### 1.5 Entity Relationship Diagram

[Provide the entity relationship diagram and the entity descriptions in the table format as below]



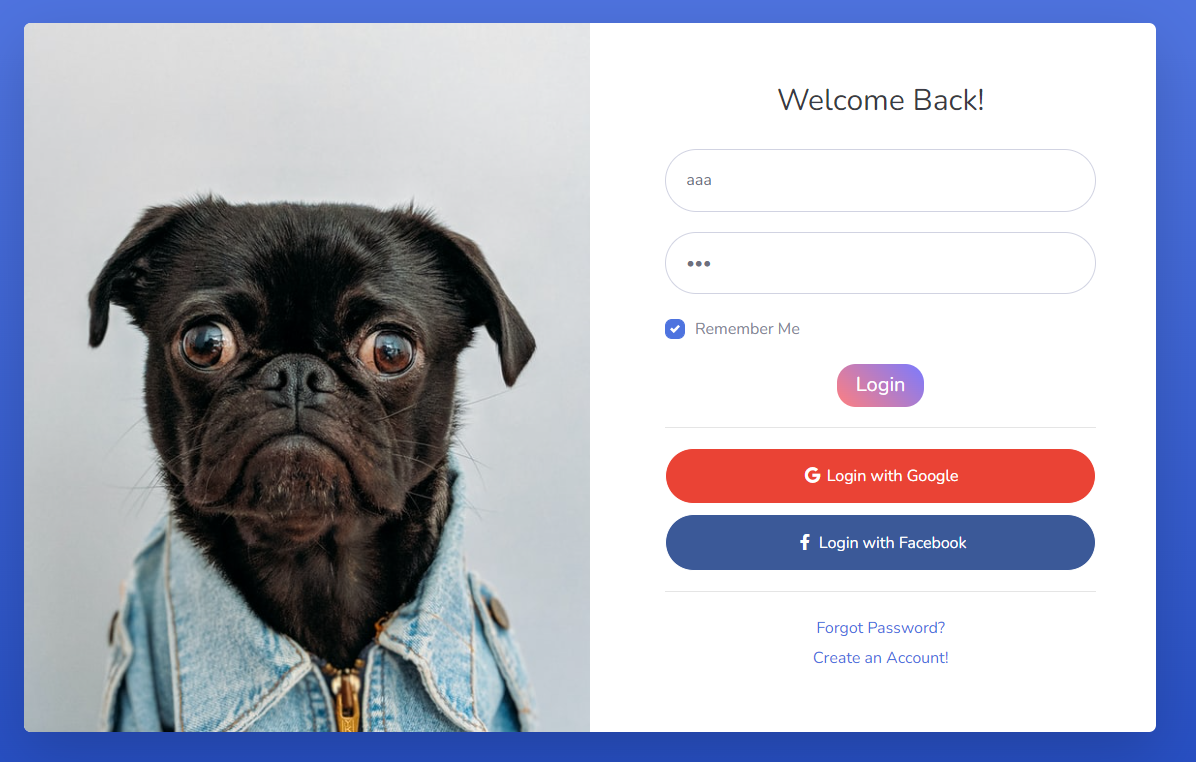
Entities Description

|  |  |  |
| --- | --- | --- |
| No | Entity | Description |
| 1 | User | who use the system |
| 2 | Setting | contain record and detail of setting |
| 3 | Class | contain record and detail of class |
| 4 | Subject | contain record and detail of subject |

## 2. Login and Register

### 2.1. Login

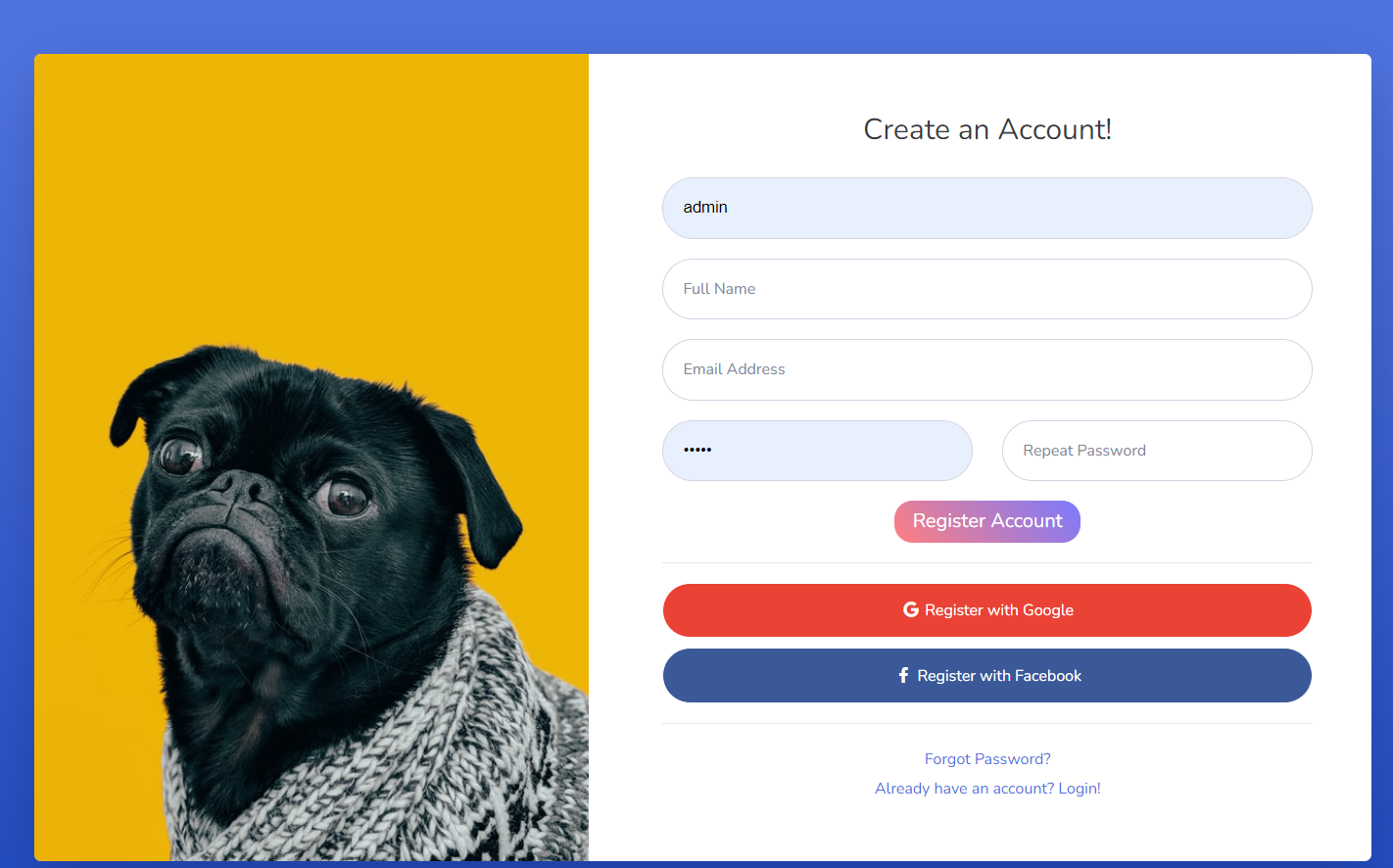
* Function trigger: When the users accessed the system and clicked the “Login” in the home page.
* Function description: Allow user use functions in the system.
* Screen layout:



* Function detail: User login into the system to use functions inside the system.

### 2.2. Register

* Function trigger: When the users accessed the system and clicked the “Register” in the Login page.
* Function description: Allows the user to create a new account to access the system.
* Screen layout:



* Function detail: User will fill in the fields of form to create a new account.

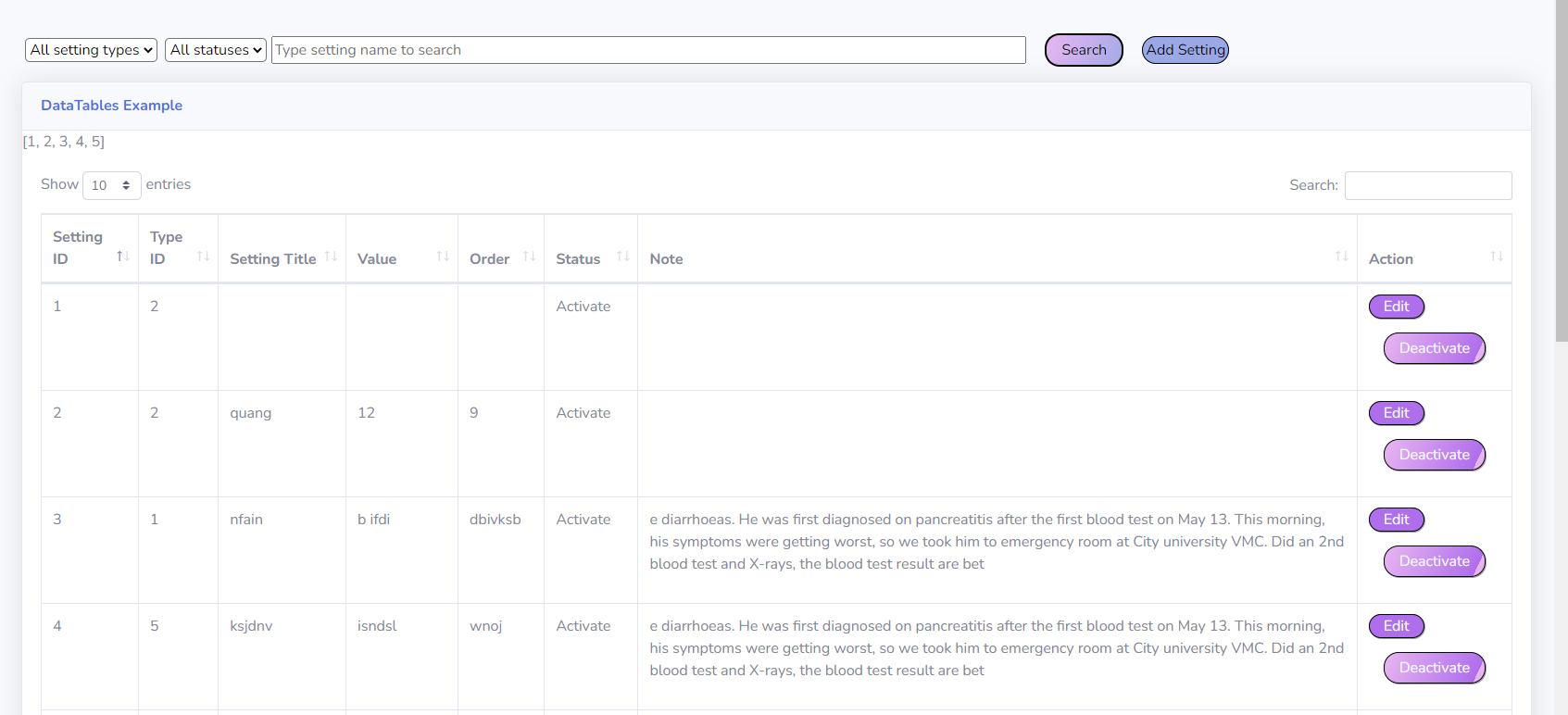
## 3. Setting

[A function can be a screen or a non-screen function (listed in the part 3.1.5 above). In this part, you need to provide the details on the related function, focus on mentioning below information

· Function trigger: When user add, update or delete a record

· Function description: admin will access the system to add, update or delete

· Screen layout: mock-up prototype of the screen, sample below is for Manage Products screen



· Function Details:

* List all settings
* add setting
* update setting
* delete setting
* search setting

## 4. User list and detail

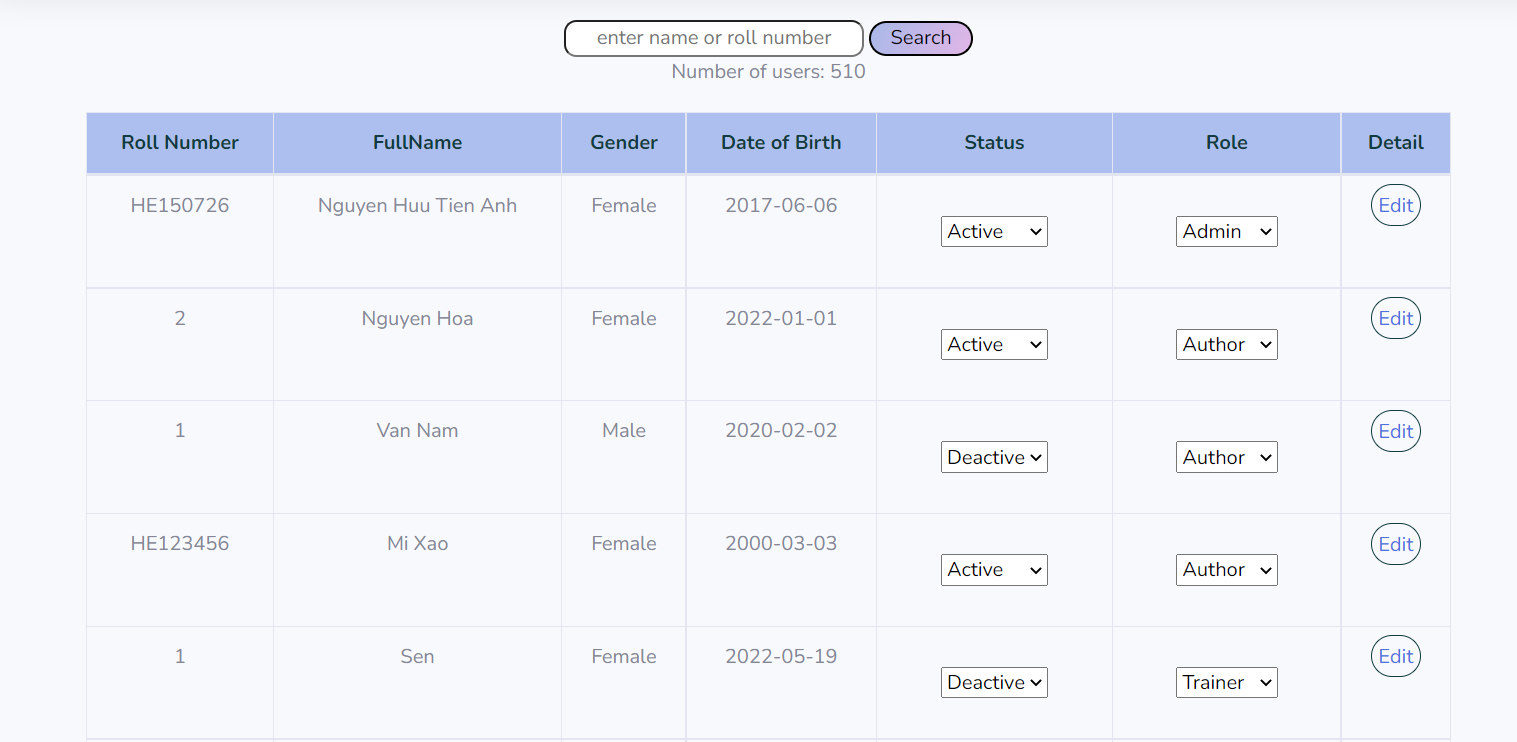
### 4.1. Function trigger

when the admin accesses the system and clicks the “User Management” in the home page.

### 4.2. Function description

Function is where admin can manage users access to the system. Admin can view all, search, delete and update each user.

### 4.3. Screen layout



### 4.4. Function detail

* Admin can view all users
* Admin can search user by full name or roll number
* Admin can deactivate user
* Admin can update information of each user
* Admin can add new user

## 5. Class list and detail

### 5.1. Function trigger:

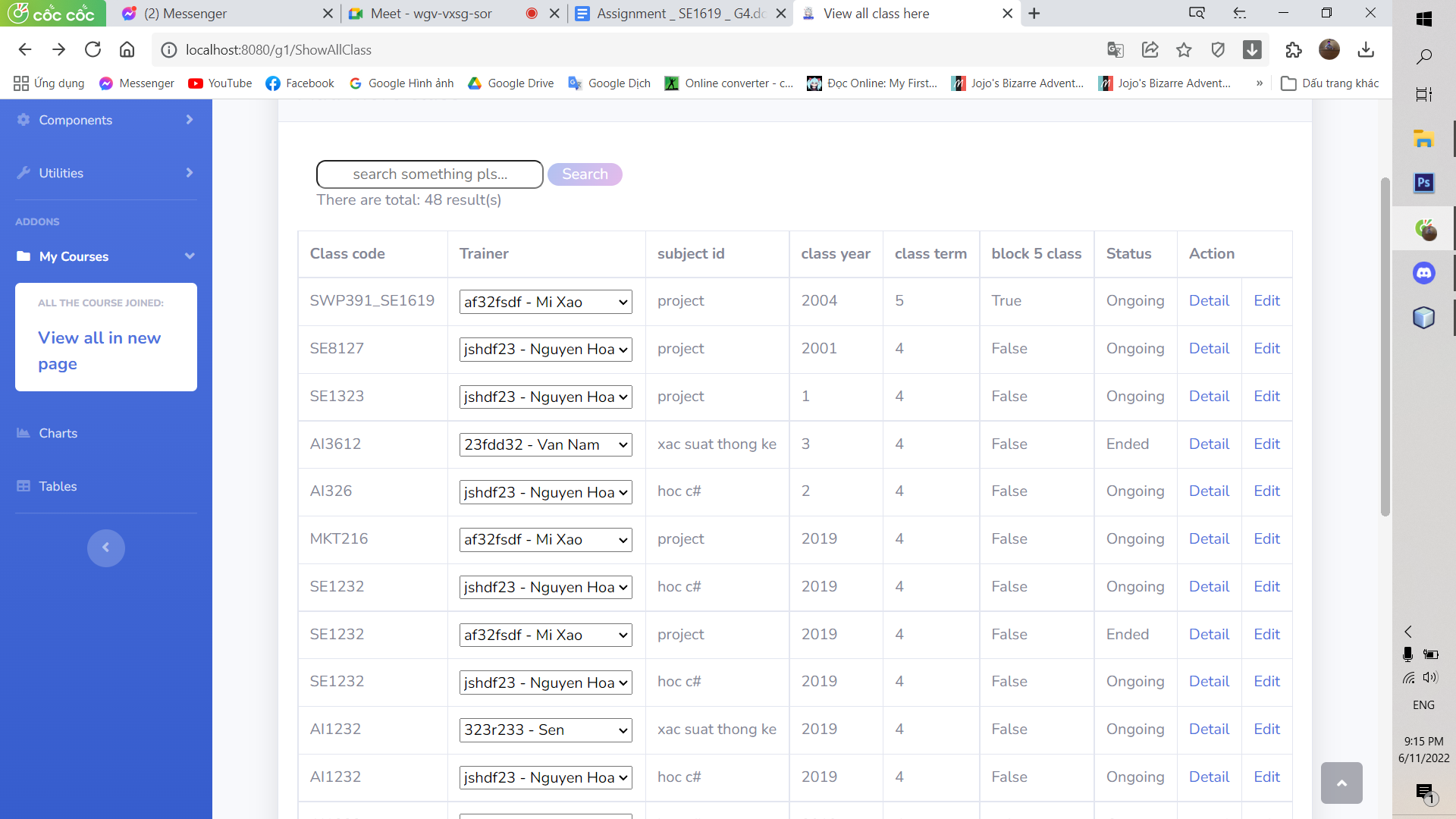
When admin access to the website, the system will show a box to access to class management

### 5.2. Function description:

A function where the admin can work with important data like class, user,... Normal users cannot access here to make changes. If they try to do so, the system will throw an error page and request the user come back to the homepage after a few seconds.

### 5.3. Screen layout:

It connects to the page home for admin to help admin change the information on this function.



### 5.4. Function Details

* Admin can see all the classes in school
* Each class has a detail button to see more detail in class (students…)
* Admin can manage all class
* admin can add more student in class
* admin can remove (kick) student in class

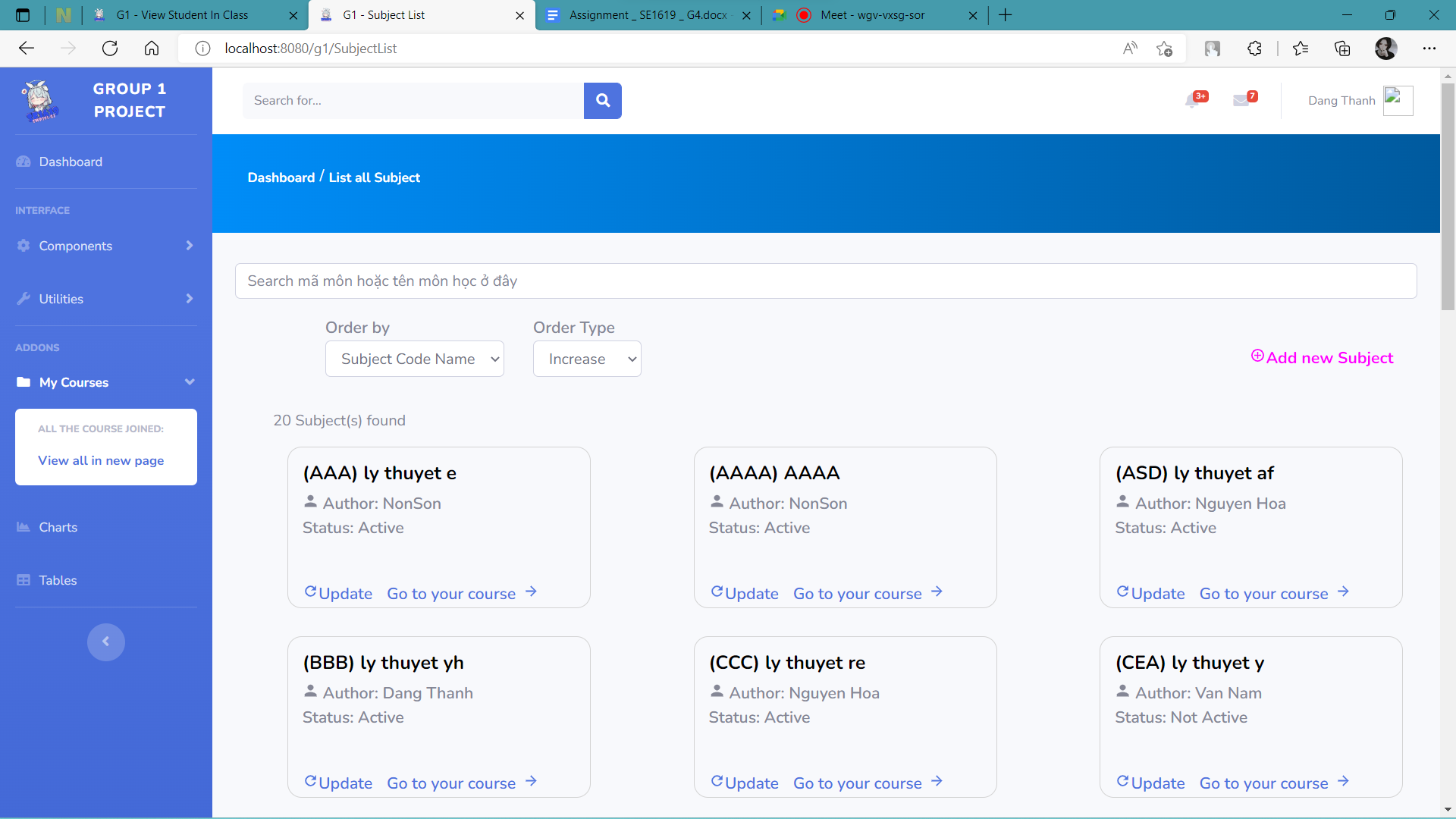
## 6. Subject list and detail

6.1. Function trigger: The user requests to see the list of subjects when clicking the button to view the list of subjects at the dashboard

### 6.2. Function description:

* Common users can see all the active subjects, search and follow the filter.
* Admin can see all the subjects including not active subjects, also search and follow the filter.
* Admin can modify the subject data like update and add new subject

### 6.3. Screen layout:



### 6.4. Function Details:

Common users just see all the classes of subjects that were activated. They also have any function of filter like search, order subject and paging

Admin can see all the list of subjects including those that are not active. This helps the admin to easily update and edit the subjects.

When the admin clicks on add a new subject, the system will display a table and ask the admin to enter the full information of the subject. If the admin input is valid, the system requirements are satisfied. The system will add a new subject to the class list and a message will appear indicating that the subject has been successfully added. If the admin's input values are not valid, the system will send a warning corresponding to the reason why the subject cannot be added and ask the admin to re-enter the subject information.

When the admin clicks on the subject update, the system will open a table to help the admin edit the selected subject for editing. Similar to how the admin adds a new subject, but when the admin enters it correctly, the system will replace the selected subject information with the new information that the admin has just entered.

# IV. Non-Functional Requirement

## 1. Usability

## 2. Reliability

## 3. Performance