TRƯỜNG ĐẠI HỌC FPT

MINISTRY OF EDUCATION AND TRAINING

FPT UNIVERSITY

Capstone Project Document

Summer Cultivation Course Management System for Co Loan Pagoda (SCCMS)

SEP490-G51	
Group Members	Nguyễn Tuấn Ninh - Leader- HE161847 Nguyễn Xuân Hậu - Member - He170950 Đỗ Thị Thanh Thủy - Member - HE161856 Đậu Đình Mạnh - Member - HE163126 Nguyễn Duy Phúc - Member - HE176098
Supervisor	Đào Thị Thanh
Ext Supervisor	Nguyễn Bá Hải

- Hanoi, December 2024 -

Table of Contents

Acknowledgement

We would like to express our deepest gratitude to all those who have supported and guided us throughout the development of this project.

First and foremost, we extend our heartfelt thanks to Ms. Dao Thi Thanh, our mentor from FPT University, for her invaluable support during the entire project. Her dedication and commitment to our success, along with her insightful guidance, have played a crucial role in shaping and completing our work. Her constructive feedback not only helped us improve our technical skills but also enhanced our project management abilities. We are deeply grateful for the time, effort, and knowledge she has devoted to our team and are proud to have her as our mentor.

We would also like to extend our sincere thanks to Mr. Nguyen Ba Hai, our mentor from FPT Software, for his valuable insights and professional support throughout the project. His practical experience and expertise helped us bridge the gap between theoretical knowledge and real-world application, providing us with a clearer direction for project development. His advice gave us the confidence to tackle challenges and resolve issues effectively. We greatly appreciate the inspiration and guidance he has provided us.

Our heartfelt gratitude also goes to Reverend Thich Thanh Dan, the Product Owner of this project, for his invaluable collaboration and support throughout the development process. His clear vision and guidance helped us thoroughly understand the requirements and objectives of the product, enabling the team to develop the project efficiently. Reverend Thanh Dan fostered a collaborative and supportive environment, inspiring us to create a product that meets the needs of users. We sincerely value his contributions and the assistance he has extended to us during this journey.

We would also like to acknowledge the efforts of our team members: Nguyen Tuan Ninh, Dau Dinh Manh, Do Thi Thanh Thuy, Nguyen Xuan Hau, and Nguyen Duy Phuc, for their relentless dedication, teamwork, and commitment to the success of this project. Each member has made significant contributions, and we are deeply grateful for the solidarity, collaboration, and team spirit we have shared throughout this journey.

Finally, we extend our sincere gratitude to the lecturers and students at FPT University who participated in interviews and shared their experiences, challenges, and insights regarding project management. Your valuable input helped us gain a deeper understanding of the real-world issues faced by students, which shaped the features and functionalities of the application to better meet practical needs. We hope this project will significantly simplify the management of group tasks and documents, facilitating more effective collaboration between students and lecturers.

SCCMS Team

Definition and Acronyms

Acronym	Definition
SCCMS	Summer Cultivation Course Management System
BA	Business Analysis
BR	Business Rule
GUI	Graphical User Interface
SDS	Software Design Specification
SRS	Software Requirement Specification
UC	Use Case
API	Application Program Interface

I. Project Introduction

1. Overview

1.1 Project Information

- Project name: **Summer Cultivation Course Management System for Co Loan Pagoda**
- Vietnamese name: **Hệ thống quản lý khóa tu mùa hè cho chùa Cổ Loan**
- Project code: **SCCMS**
- Group name: **SEP490-G51**
- Software type: **Web Application**

1.2 Project Team

Full Name	Role	Email	Mobile
Đào Thị Thanh	Lecturer	thanhdt59@fe.edu.vn	0346078237
Nguyễn Bá Hải	Lecturer	hainb4@fe.edu.vn	0976998086
Nguyễn Tuấn Ninh	Leader	ninhnthe161847@fpt.edu.vn	0338516217
Đỗ Thị Thanh Thuỷ	Member	thuydtthe161856@fpt.edu.vn	0973506889
Đậu Đình Mạnh	Member	manhddhe163126@fpt.edu.vn	0981661879
Nguyễn Duy Phúc	Member	phucndhe176098@fpt.edu.vn	0911468929
Nguyễn Xuân Hậu	Member	haunxhe170950@fpt.edu.vn	0969583865

Table 1: Project team member information

2. Product Background

Currently, summer cultivation courses are being organised nationwide. The cultivation program is designed to be rich with meditation activities, learning Buddhism, group activities, and culture such as performing arts, and folk games, helping participants practise their spirit and develop themselves. In particular, summer cultivation courses are increasingly attracting parents' attention, not only assisting students to have a meaningful summer but also helping them grow in morality, understand more about love for the Fatherland, filial piety to parents, and the spirit of dedication to society.

With the great values that summer cultivation courses bring, every year in the summer, Co Loan Pagoda, located in Ninh Tien Commune, Ninh Binh City, Ninh Binh Province hosts multiple cultivation courses each summer, attracting over 200 students and volunteers. Over the years, the manual registration and management processes have proven inefficient. However, despite the popularity and success of these courses, the pagoda's current manual registration and management processes have proven increasingly inefficient and burdensome over the years. Parents wishing to register their children often face considerable challenges.

The current business process is described below. Before the cultivation course begins, students need to prepare their complete documentation. The person who wants to be a supervisor -who is responsible for taking care of students- will register to participate and receive notifications from the secretary. They must also attend a training session to understand their responsibilities. Parents need to prepare their children's documentation to ensure everything is ready for the enrollment day.

Students will arrive at the reception desk on enrollment day to complete the enrollment process. Here, they will receive uniforms and student cards, and be guided to their accommodations by the supervisor. After changing into their uniforms and wearing their student cards, students will have lunch and participate in activities according to the established schedule. The supervisors will greet the students, check their documentation, confirm the number of students, and assist them in settling into their prepared accommodations. Parents will bring the necessary documentation to enrol their children and receive a pickup appointment for after the cultivation course.

Throughout the cultivation course, students will engage in activities according to the schedule. In the morning, they will wake up, attend to personal hygiene, meditate, have breakfast, chant, and study teachings. After lunch, students will participate in chores and take rest breaks. In the evening, they will bathe, have dinner, engage in group activities, and write in their journals. Supervisors will monitor and assist students in all daily activities and conduct meetings to check on the health and well-being of the students. On the other hand, pagoda also has volunteers, who will participate in different teams and coordinate tasks via Zalo, while the teachers will guide and oversee all activities.

On dismissal day, students will prepare their belongings to go home and write reflections on the cultivation course. Supervisors will handle the process of sending students back, ensuring that everyone is safe and accounted for. Parents will bring the pickup appointment slip to collect their children.

After the cultivation course concludes, supervisors will write a summary report detailing strengths, weaknesses, and suggestions for improvement for future courses. Volunteers will receive participation certificates (if available), and students will write about the changes they have experienced after the cultivation course, capturing their memories and lessons learned.

On the side of parents, in September 2024, the pagoda surveyed parents to gain deeper insights into how parents are experiencing the current summer cultivation course. The result of this survey are collected and presented in the sheet: *Khảo sát phụ huynh khóa tu chùa Cổ Loan*

Based on the more than 30 responses collected, the pagoda obtained the following results:

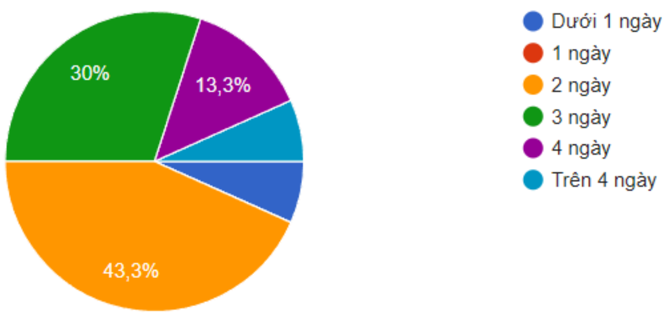


Figure 1: The time parents have to spend to gather cultivation course information

The data underscores the considerable amount of time that parents invest in researching and obtaining relevant information. A significant proportion, 43.3%, require an entire day to complete this task, while an additional 30% of parents need two full days to gather the necessary details. This means that more than 70% of parents spend between one and two days engaged in information-gathering, which can be seen as a substantial time investment. Furthermore, 13.3% of parents require four days to accomplish this, and while the percentage of those spending more than four days is minimal, the data suggests that for many, the process is protracted and inefficient. These findings point to a clear inefficiency in the current system of information dissemination, revealing a substantial need for a more streamlined and effective management system. Such a system would not only reduce the time burden placed on parents but also enhance accessibility and ensure that essential course information is provided in a more timely and organized manner. In light of these statistics, it becomes evident that developing a centralized, well-structured information management platform would greatly benefit parents, potentially increasing both satisfaction and participation in the course enrollment process.

The team also interviewed three secretaries currently working at the pagoda. The results showed that notifying parents of the registration results takes approximately five minutes per student. The pagoda's secretaries spend around five days organising personnel into various teams, assigning students to groups, and arranging night shifts appropriately. Creating cards for students and volunteers takes about ten minutes per card, extending the card preparation time for around 200 students to approximately four to five days. The administrative staff also face difficulties in sending invitations to 250 former volunteers to participate in the new course. Moreover, storing reports manually has made it challenging to review and retrieve past records.

3. Existing Solutions

Currently, after surveying existing solutions and cultivation course management systems, we have found that extremely little system is specifically designed or implemented to meet the particular needs of managing cultivation courses for pagodas in Vietnam, including Co Loan Pagoda. Moreover, existing solutions are available, they are usually general information management tools or event management tools, and they fail to meet the specific requirements of the pagoda in organising, coordinating, and overseeing cultivation processes.

3.1 Current Cultivation Management of Co Loan Pagoda

Currently, the pagoda manages all activities related to the Cultivation Courses and volunteer assignments manually, largely through direct interactions and traditional paper-based methods. This system includes handling registrations, volunteer allocations, and daily operational tasks without any centralised digital assistance.

As the pagoda's activities, such as organising Cultivation Courses and managing volunteers, expand, this manual approach has become increasingly challenging. The staff, including the Secretary and Managers, must manually review and update records, which is time-consuming and error-prone. This method also makes it difficult to efficiently track the skills and availability of volunteers and assign them appropriately to the various tasks or shifts required.

For registering as a cultivation course participant, the pagoda announces the necessary information and documents for registering for the cultivation course through social media or word of mouth. Students must prepare their complete application and then send it via the pagoda secretary’s Zalo. After receiving the application, the secretary will assess whether it meets the requirements for enrollment. If any documents are missing or extra, the secretary will notify the parents to prepare them. If the application is complete, the secretary will add the student to the registration sheet. Once the list of participants is finalized, the secretary will divide the students into different groups. This task is done using Excel to ensure that the groups are balanced in terms of age and that no group contains both male and female students. Afterward, the secretary will create student cards for the participants using Photoshop. On the registration day, students will bring their application to submit to the secretary. The secretary will mark that student as enrolled and then issue them their uniform and participation card.

For registering as a volunteer for the cultivation course, the pagoda will announce the recruitment of volunteers through social media. Volunteers will register via Google Forms, after which the secretary will review and select eligible volunteers and assign them to specific teams. The secretary will also send notifications to those who are not selected. For those who are selected, the secretary will send the list to team leaders through a shared sheet. The team leaders will then contact their volunteers and add them to a shared Zalo group.

For reports, the supervisors are required to write a report each day. Every evening, the supervisors will collect a report template from the secretary’s office, fill it out, and submit it. These reports will be compiled for review by the pagoda’s monks and the secretary. The report includes issues such as students’ psychological state, health, and eating habits, as well as any difficulties or arising problems.

For night shifts, first, the pagoda staff will register their availability through a sign-up form. Students will be assigned to different rooms, with male and female students separated. Each night, there will be approximately three night shifts. The secretary will use the form to allocate night shifts to staff members, assigning them to different shifts and rooms, and then notify the staff via a shared Zalo group. When it's time for their shift, if it's the first shift of the night, the staff member will collect the logbook from the secretary's office. For the second shift and beyond, the staff member will collect the logbook from the previous shift's person. During their shift, the staff must note down any issues during the night (such as room temperature, noise, etc.) and specific notes for each student. The last shift of the night will return the logbook to the secretary.

Advantages

- Simplicity in implementation without the need for technical infrastructure.
- Direct control over each process by the pagoda staff, providing a personalised touch.

Disadvantages

- Time-consuming, requiring significant manual effort to manage and update information.
- Prone to human error, leading to potential inconsistencies in volunteer assignments and course management.
- Lacks quick scalability and flexibility in response to changes or last-minute adjustments.
- No centralised data storage or backup, increasing the risk of data loss.
- Limited reporting capabilities, making it difficult to generate comprehensive reports for feedback and future improvements.

3.2 Ba Vang Pagoda Management System

The website system of Ba Vang Pagoda is a comprehensive platform designed to optimise the registration process for summer cultivation course courses. The site offers a user-friendly interface, making it easy for participants to register and select suitable courses based on their age. Specifically, the system categorises courses into groups such as: Course 1 for students from grades 7 to 9 (born between 2012 and 2010), Course 2 for students from grades 10 to 12 (born between 2009 and 2007), Course 3 for students from grades 2 to 6 (born between 2017 and 2013), and Course 4 for those from grade 12 to university and young adults under 30 (born between 2006 and 1994).

Participants simply need to click on the corresponding course to proceed with registration, with clear information displayed for each age group. The system supports user information collection through a simple process, including steps for entering personal information and selecting a course. Additionally, the website provides detailed information about the schedule, location, and activities of the cultivation courses, facilitating easy search and tracking for users.

Besides supporting participants, this system also aids the pagoda management by automatically collecting and organising data about registrants, including personal information, course preferences, and participant numbers. This helps the organisers monitor the number of students, manage capacity, and plan cultivation course activities effectively.

Furthermore, the website integrates news sections that provide updates on events and notable activities at the pagoda, especially programs related to summer cultivation courses, videos, and live events such as study sessions and prominent activities at the cultivation courses, including videos sharing Buddhist knowledge or Google Maps, allowing users to easily search for and get directions to Ba Vang Pagoda. Users can click on the "Directions" section to be guided directly through Google Maps, making it more convenient and accurate for those visiting the pagoda for the first time or those needing assistance in navigating from different locations.

With an intuitive design, bright colours, and a lively interface, the system helps users easily navigate through its functions. The system is also compatible with both computers and mobile devices, making registration quick and convenient. Additionally, support features through phone, email, or a hotline are available to address user inquiries promptly and effectively.

The system's link: https://chuabavang.com/dang-ky/dang-ky-khoa-tu-mua-he/

Figure 2: Registering for a summer cultivation course at Ba Vang Pagoda.

Advantages

- It has been accessible for multiple years and has demonstrated its efficacy for pupils.
- Designed with a user-friendly interface that requires minimal training, making it accessible for pagoda staff and volunteers.

Disadvantages

- Primarily focused on pagoda management in general, it lacks features for specialized management for cultivation courses that can be useful for intensive cultivation course management and organization activities
- The system does not have management functions like daily reports, the application form is for volunteers or night shift assignments.

4. Business Opportunity

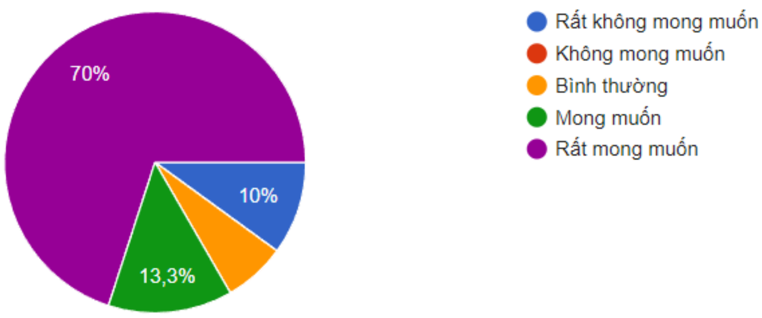


Figure 3: Parents' interest in a website that compiles information related to the cultivation course.

The result of this survey are collected and presented in the sheet: *Khảo sát phụ huynh khóa tu chùa Cổ Loan*

The pagoda also surveys to analyze parents' interest in a website that compiles information related to the cultivation course. The survey results demonstrate a strong positive response towards the creation of such a platform. A substantial 70% of parents are "very interested," and an additional 13.3% express moderate interest, meaning that 83.3% of parents overall support the idea. Only 10% of respondents are neutral, while opposition is negligible. These findings suggest a clear demand for a centralised system that would make it easier for parents to access course information, significantly reducing the time and effort required. The data strongly indicates that a website like this would be widely appreciated and highly beneficial for the majority of parents.

Moreover, the secretaries require a management system where they can reduce their manual workloads. The pagoda's secretaries currently spend around five days organising personnel, assigning students to groups, and arranging night shifts. A digital management system can streamline these tasks through automated scheduling, team assignments, and shift planning, significantly reducing the time and effort required while improving the accuracy of arrangements. Creating ID cards for around 200 students and volunteers currently takes about ten minutes per card, extending the card preparation time to four to five days. An integrated system that automates card generation, including photo processing and data input, can reduce the preparation time, allowing for faster and more efficient handling of this critical task.

The secretaries face challenges in sending out invitations to 250 former volunteers due to manual processes. By adopting an automated SMS or emails, the pagoda can easily manage mass invitations, follow-ups, and confirmations, ensuring timely and consistent outreach to past participants and reducing manual errors.

The manual storage of reports has made it difficult to review and retrieve past records, limiting the ability to make data-driven decisions and track historical trends. Implementing a digital report management system would enable easy access, searchability, and analysis of past data, enhancing decision-making and operational efficiency.

5. Software Product Vision

For parents of students who want to enroll their children in summer cultivation courses, the Summer Cultivation Course Management System for Co Loan Pagoda is a comprehensive web-based platform designed to streamline and simplify the registration process. The system allows parents to access and submit summer cultivation course registrations for their children with a user-friendly interface. Unlike the current manual methods where parents who want to enroll their children in a course usually spend days gathering course information, understanding the registration procedures, and receiving feedback on the registration results, this system provides complete information about the summer cultivation course, registration procedures, saving time, enhancing communication and improving efficiency by registering online.

For secretaries involved in managing summer cultivation courses, the Summer Cultivation Course Management System for Co Loan Pagoda is a comprehensive web-based platform designed to streamline and simplify the process of staffing different departments, assigning students to groups, scheduling appropriate night shifts, and returning registration results to parents. The Summer Cultivation Course Management System for Co Loan Pagoda allows secretaries to schedule night shifts in minutes, rather than days as they would have done manually. It also allows for automatic issuance of student and volunteer ID cards, which is a much easier process than doing it manually.

The pagoda's administrative team will be able to easily automatically send invitations to former volunteers to join new courses and allow administrators to review and retrieve records. The system provides a user-friendly management interface, saving time and increasing efficiency in management.

By addressing these key aspects, the Summer Cultivation Course Management System for Co Loan Pagoda aims to become an indispensable tool for Co Loan Pagoda, improving the efficiency and effectiveness of managing summer cultivation courses.

6. Project Scope & Limitations

The system will provide a comprehensive solution to support the organization and operation of summer cultivation course activities, including participant registration, volunteer registration, and night shift management. The system also provides functions to manage reports during the course. The primary objective of the software is to enhance management efficiency, reduce manual work, and improve the experience for both participants and the organising team.

The software will include features such as online registration, personal information management, and facilitating interaction between participants and instructors. Additionally, the software will integrate reporting and statistics functions to help organizers easily evaluate and adjust cultivation course activities.

However, the software will not include features related to financial management such as tuition collection, fund management, or health services. Any requests or features outside of this defined scope will not be considered unless they offer significant value and are approved along with necessary adjustments to the budget, timeline, and resources. The system does not manage food or volunteer tasks during the cultivation course. It also does not support scheduling and learning material management, detailed tasks of volunteers, and event management for the cultivation course

6.1 Major Features

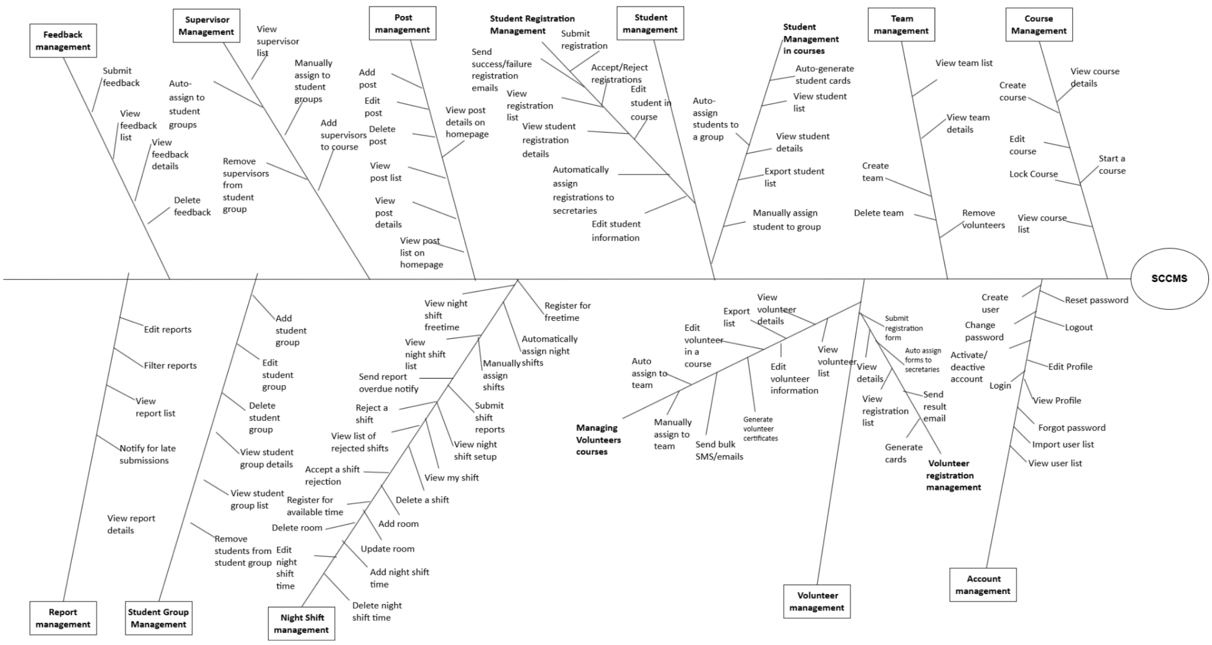


Figure 4: Feature tree of the project

The link of the feature tree: [FeatureTree.drawio]

Feature ID	Feature
Admin Features	
FE-01	**Account Management**: Enables admins to create, deactivate, and activate accounts. Admin can also login, password reset, create manager, and view or edit profiles.
Manager Features	
FE-01	**Account Management**: Enables managers to create, deactivate, and activate accounts of secretaries and staff, team leaders and supervisors. Managers can also do functionalities like login, password reset, import of user lists and viewing/editing profiles.

FE-02	**Course Management**: This feature allows managers to create, edit, view, and delete courses. It includes the ability to set start and end dates, expected number of students, and notify related parties of any changes (e.g., schedule changes). Managers can also lock/unlock courses and view detailed course information.
FE-03	**Team Management**: Allows managers to create and manage teams based on gender, age, and the expected number of participants. Teams can be edited, deleted, and assigned to leaders. Additionally, managers can view and filter the volunteer list in each team.
FE-04	**Volunteer Management**: Managers can accept or reject registrations, view the registration list, search/filter volunteer registration details, and send bulk notifications. Managers can auto-generate volunteer cards, assign volunteers to teams, view, filter, and export volunteer lists, and assign volunteers manually or automatically. This feature also provides a consolidated view of volunteers across a course, enabling managers to filter and search for volunteers, and send emails to all volunteers.
FE-05	**Supervisor Management**: Managers assign supervisors to student groups manually or automatically, and manage their details. Supervisors can also be added or removed from specific courses or groups.
FE-06	**Night Shift Management**: The system automatically assigns staff to shifts optimally. Managers can manually assign volunteers, view shift details. The system will use Greedy Algorithm and Brute Force to assign staffs to night shifts automatically.

FE-07	**Student Management**: Managers can send success/failure emails, accept/reject applications, view and filter registration lists, search registration details, and update registration status or confirm admissions. This feature also auto-generates student cards, allows managers to view and export student lists, filter and sort students, and send bulk messages/emails to parents. Students can also be assigned or removed from groups.
FE-08	**Student Group Management**: Enables managers to add, edit, and delete student groups, view students in a group, and manage the supervisors assigned to that group.
FE-09	**Report Management**: Managers can view and filter daily reports, reopen an overdue report
FE-10	**Post Management**: This feature provides functionality for managing content on a public-facing page. Managers can add, edit, and delete posts, view a post list of the page.
FE-11	**Feedback Management**: Enables managers to view the feedback list, and access details of submitted feedback. Managers can also delete feedback entries.
Secretary Features	
FE-01	**Account Management**: Secretaries can do functionalities like login, password reset, bulk Email sending, and viewing/editing their own profile.
FE-02	**Course Management**: This feature allows secretaries to view courses.
FE-03	**Team Management**: Secretaries can view, filter, add and remove the volunteer list in each team.

FE-04	**Volunteer Management**: Secretaries can accept or reject registrations, view the registration list, search/filter volunteer registration details, and send bulk notifications. Secretaries can auto-generate volunteer cards, assign volunteers to teams, view, filter, and export volunteer lists, and assign volunteers manually or automatically. Bulk notifications can be sent to volunteers. This feature also provides a consolidated view of volunteers across all courses, enabling secretaries to filter and search for volunteers, and send emails to all volunteers.
FE-05	**Supervisor Management**: Secretaries can auto-generate supervisor cards, assign supervisors to student groups manually or automatically, and manage their details. Supervisors can also be added or removed from specific courses or groups, and bulk notifications can be sent.
FE-06	**Night Shift Management**: The system automatically assigns staff to shifts optimally. Secretaries can manually assign volunteers, view shift details, submit and manage notifications for overdue or rejected shifts.
FE-07	**Student Management**: Secretaries can send success/failure emails, accept/reject applications, view and filter registration lists, search registration details, and update registration status or confirm admissions. This feature also auto-generates student cards, allows secretaries to view and export student lists, filter and sort students, and send bulk messages/emails to parents. Students can also be assigned or removed from groups.
FE-08	**Student Group Management**: View, filter and search students in a groups
FE-09	**Report Management**: This feature allows secretaries to view daily reports. Secretaries can also filter lists and view detailed report information.
FE-10	**Feedback Management**: Enables secretaries to view the feedback list, and

	access details of submitted feedback
Team Leader Features	
FE-01	**Account Management**: Team leaders can do functionalities like login, password reset, and viewing/editing their own profile.
FE-02	**Course Management**: This feature allows team leaders to view courses.
FE-03	**Team Management**: Team leaders scan view and filter the volunteer list in each team.
FE-04	**Volunteer Management**: View, filter volunteer lists.
FE-07	**Student Management**: Allows team leaders to view, filter and sort students.
FE-08	**Student Group Management** View, filter and search students in a groups
FE-09	**Report Management**: The team leader can view and filter daily reports.
Staff Features	
FE-01	**Account Management**: Staff can do functionalities like login, password reset, and viewing/editing their profile.
FE-02	**Course Management**: This feature allows staff to view courses.
FE-03	**Team Management**: Staff scan view and filter the volunteer list in each team.
FE-04	**Volunteer Management**: View, filter, and export volunteer lists.
FE-06	**Night Shift Management**: This feature allows staff to register for night shifts and submit night shift reports. Staff can also reject night shifts assigned.

FE-07	**Student Management**: Allows staff to view and export student lists, filter and sort students.
FE-08	**Student Group Management** View, filter and search students in a groups
FE-09	**Report Management**: Staff view and filter night shifts, submit night shift reports and can request opening in case it is overdue.
Supervisor Features	
FE-01	**Account Management**: Supervisor can do functionalities like login, password reset, and viewing/editing their own profile.
FE-02	**Course Management**: This feature allows supervisor to view courses.
FE-03	**Team Management**: Supervisor can view and filter the volunteer list in each team.
FE-04	**Volunteer Management**: View and filter volunteer lists.
FE-06	**Night Shift Management**: This feature allows supervisor to view night shift
FE-07	**Student Management**: Allows supervisors to view student lists, filter and sort students. Supervisor can also export the student list
FE-08	**Student Group Management** View, filter and search students in a groups
FE-09	**Report Management**: Supervisor can submit, view, edit, and filter daily reports.
Guest Features	
FE-04	**Volunteer Management**: This feature enables volunteers to submit registration forms and receive successful or failed registration notifications via email.

FE-07	**Student Management**: Enables online registration submission for students' parents. Parents can also see evaluations of students by filling in the form student code.
FE-10	**Post Management**: View the logo and posts to learn information about the pagoda, information about the cultivation course, how to register for the course, and the necessary documents for admission... on the page.
FE-11	**Feedback Management**: Enables parents to submit feedback about the cultivation course. Parents can write about their children's changes after (Parents must input their student's identity across the course)

Table 2: Major features

Conclusion:

The Summer Cultivation Course Management for Co Loan Pagoda is designed to enhance the management of summer cultivation courses by providing a solution that addresses key operational challenges. By streamlining the processes management, the system will significantly improve the overall efficiency and effectiveness of the course. This ensures smooth coordination, clear communication, and timely reporting for all stakeholders involved in the summer cultivation programs.

6.2 Limitations & Exclusions

6.2.1 Limitations

ID	Name	Description
LI-01	No management of food during the cultivation course	The software does not provide any functionality for managing food-related tasks during the cultivation course. This includes tracking food inventory, managing quantities, or handling meal planning and preparation. As a result, all food management tasks, such as procuring and preparing meals for

		participants, will need to be done manually or through a separate system. The software focuses solely on course-related management functions.
LI-02	No management of volunteers' tasks during the cultivation course	The software does not offer features to manage volunteers' activities, such as task assignment, progress tracking, or scheduling. There is no functionality to monitor volunteer hours or assign specific duties within the cultivation course. These responsibilities will be handled separately from the software, either manually or with another tool, allowing the system to remain focused on managing course content and participants.
LI-03	No support for scheduling and learning material management	The system lacks functionality to support scheduling or managing the distribution of learning materials for the cultivation course. There is no provision for organizing course schedules, setting up timetables, or managing the flow of educational materials during the course. Users will need to rely on external tools or manual methods to handle these tasks, as the system's scope is limited to other areas of course management.
LI-04	The system does not include special event management for the cultivation course	The system does not provide any support for organizing special events, such as guest lectures, workshops, or other activities that might occur during the cultivation course. These types of events, which could enrich the learning experience, will need to be planned and managed separately, using other tools or methods. The system's focus is strictly on core course

		management, ensuring that event planning falls outside its scope.
LI-05	No support for financial management and online payments	The system does not include any tools for managing finances or handling online payments. This means that any tasks related to tracking course fees, processing payments from participants, or managing the course budget will need to be done manually or through another system. The focus of the software remains on course management without delving into financial operations, which must be managed separately.
LI-06	No management of resources and facilities	The system does not provide any functionality for managing physical resources or facilities during the cultivation course. This includes tracking the usage, availability, and allocation of materials such as classroom equipment or other supplies. Any management of physical assets required for the course will need to be handled manually or through a separate system. The software focuses solely on managing course-related tasks.

Table 3: Limitations

6.2.2 Exclusions

ID	Name	Description
EX-01	No advanced analytics and reporting feature	The system will not include advanced data analytics, predictive modelling, or custom report generation. Only basic reporting functions are available and users should not expect complex data insights.

EX-02	No mobile application support	The project won't have mobile application support. The users can only access the system through a browser.
EX-03	Internet Dependency	The system requires a stable internet connection to function properly. Users may experience limited functionality or downtime if their internet connection is unstable or unavailable.

Table 4: Exclusions

II. Project Management Plan

1. Overview

1.1 Scope & Estimation

In this project, we define 1 man-day equals to 6 hours

#	WBS Item	Complexity	Est. Effort (man-days)
1	***Account Management***		***56***
1.1	View users list	Complex	10
1.2	Import user list	Complex	10
1.3	Forgot password	Complex	10
1.4	View Profile	Medium	8
1.5	Create account	Medium	8
1.6	Login	Complex	10
2	***Course Management***		***18***
2.1	View course list	Medium	8
2.2	Create course	Complex	10
3	***Student management***		***44***
3.1	View student registration list	Complex	10
3.2	Student registration details	Medium	8
3.3	View student details	Medium	8
3.4	View student list	Complex	10
3.5	View student result	Medium	8
4	***Student group management***		***43***
4.1	View student group list	Normal	5
4.2	Create student group	Complex	10
4.3	View student group details	Complex	10
4.4	Add students to a student group	Medium	8
4.5	Edit student group	Complex	10
5	***Supervisor management***		***18***

5.1	View supervisor list	Medium	8
5.2	Add supervisors to a course	Complex	10
6	***Team management***		***42***
6.1	View team list	Medium	8
6.2	Create team	Medium	8
6.3	View team details	Complex	10
6.4	Edit team	Medium	8
6.5	Add volunteers to a team	Medium	8
7	***Volunteer management***		***41***
7.1	View volunteer registration list	Complex	10
7.2	Volunteer registration details	Normal	5
7.3	Volunteer details	Medium	8
7.4	View volunteer list	Complex	10
7.5	Accept/Reject registration	Medium	8
8	***Night Shift management***		***68***
8.1	View night shift list	Complex	10
8.2	Register for available time	Complex	10
8.3	View staffs' freetime	Medium	8
8.4	View night shift details	Complex	10
8.5	View list of night shift rejection	Complex	10
8.6	View my night shift assignment	Complex	10
8.7	Night shift config	Complex	10
9	***Post management***		***37***
9.1	View post list (Guest's page)	Medium	8
9.2	Guest home page	Medium	8
9.3	View post list	Normal	5
9.4	View post details	Medium	8
9.5	View post details (Guest's page)	Medium	8

10	***Report management***		***20***
10.1	View report details	Complex	10
10.2	View report list	Complex	10
11	***Feedback management***		***18***
11.1	View feedback list	Normal	5
11.2	View feedback details	Normal	5
11.3	Send feedback	Medium	8
Total Estimated Effort (man-days)			**405**

Table 5: Scope and Estimation

1.2 Project Objectives

1.2.1 Allocated Effort

#	Members	Weekdays	Weekends
1	Nguyễn Tuấn Ninh	6 hours	8 hours
2	Nguyễn Xuân Hậu	6 hours	8 hours
3	Nguyễn Duy Phúc	6 hours	8 hours
4	Đỗ Thị Thanh Thủy	6 hours	8 hours
5	Đậu Đình Mạnh	6 hours	8 hours

Table 6: Allocated Effort

Total allocated effort: 428 man-days

1.2.2 Defect Rate

We identify 1 man-month (MM) as 20 man-days. Since the team is developing a new system, we keep the defect rate (No. Def/MM) at around 4.2. Therefore the number of defects we want to identify after delivering the product to the customer is around 90. We will divide the defects into categories as below:

#	Testing Stage	Test Coverage	No. of Defects	% of Defect	Notes
1	Review	100%	50	55.55%	Perform to identify defects at an early stage

1	Unit Test	100%	0	0%	They are written and run by software developers to ensure that a section of an application (known as the "unit") meets its design and behaves as intended.
2	Integration Test	100%	25	27.78%	Individual software modules are combined and tested as a group.
3	System Test	100%	10	11.11%	Performed on the entire system, tests not only the design but also the behavior and even the believed expectations of the customer.
4	Acceptance Test	100%	5	5.56%	Users of the system perform tests in line with what would occur in real-life scenarios

Table 7: Defect distribution

Review Stage (55.55%): This high percentage indicates that many defects were found at an early stage, highlighting the importance of the review process in identifying issues before formal testing begins, thus helping to prevent costly fixes later in the development process.

Unit Test Stage (0%): The defect is 0% at this stage means that all the bugs at this level are fixed before coming to the next stage. If there are any defects remaining at this level, the project cannot run properly.

Integration Test Stage (27.78%): The number of defects discovered drops further in integration testing, which implies that the individual modules, once tested, tend to work well together. This stage identifies issues that may arise when integrating different modules, but since the number of defects found is relatively low, it indicates that the earlier tests were effective.

System Test Stage (11.11%): The number of defects identified in system testing is the lowest among the stages, suggesting that the system, when tested as a whole, performed better. This stage evaluates overall functionality and compliance with specifications, highlighting effective testing practices in the earlier stages.

Acceptance Test Stage (5.56%): A small number of defects were found during acceptance testing, indicating that the system met user requirements and expectations. This is a strong indicator of quality and thoroughness throughout the development and testing process.

1.2.3 Timeliness

We meet at least 90% of the assigned deadlines to ensure the project is completed on time.

1.2.4 Requirement Completeness

With the number of use cases identified, we aim to complete at 100%.

1.3 Project Risks

#	Risk Description	Impact	Possibility	Response Plans
1	Poor communication among members leads to missing critical information	High	High	Set up meetings and use project management tools to keep information updated. Encourage direct and clear communication to avoid misunderstandings.
2	Insufficient time spent on the capstone project of team members	High	High	Each member regularly reports any abnormal events to the whole team to quickly support. Make a clear commitment measured by working hours per day from the project commencement.

3	Customers change requirements and scopes are expanded beyond expectation	Medium	Medium	Establish a clear scope and regularly review it to ensure that it remains under control. Any proposed changes should be carefully reviewed before being implemented.
4	Lack of members due to a member being forced to leave due to not being qualified to defend the project.	High	Low	Members agree to commit to spending enough time. Members have to learn additional skills necessary to serve the project and agree to fulfill their responsibilities, avoiding being forced to leave the group due to not completing the minimum tasks.

Table 8: Project risk

2. Management Approach

2.1 Project Process

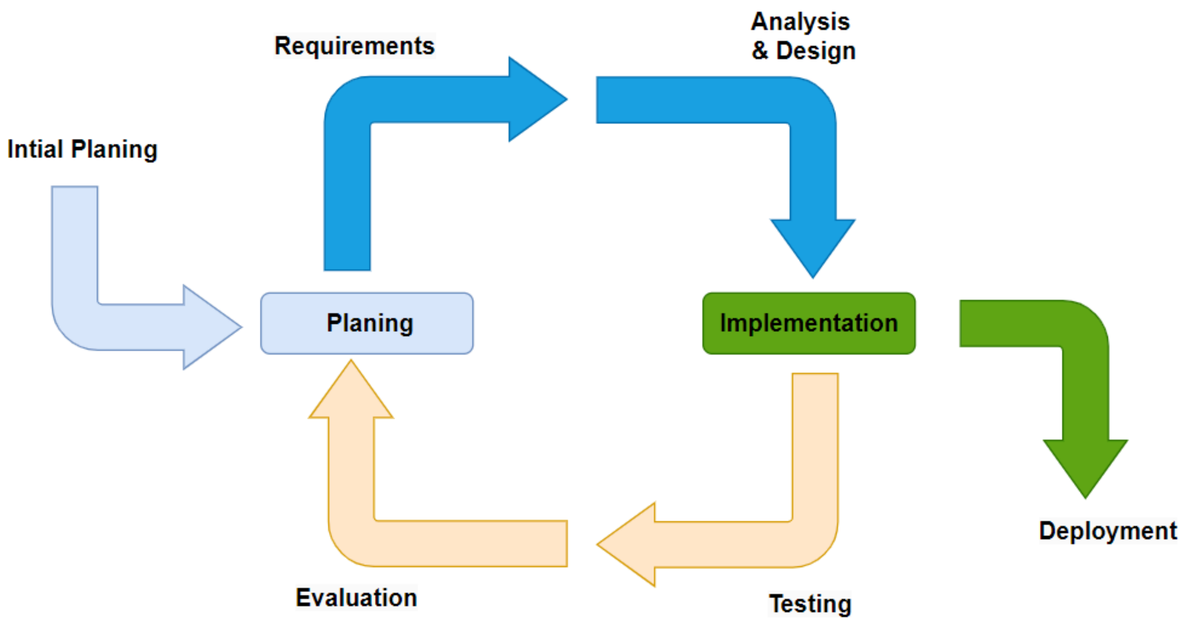


Figure 5: Iterative and Incremental Model Process

After carefully reviewing the models for software development, we decided to apply the **Iterative and Incremental model** for this project. This software development model is an approach to software development through building the system step by step and iterating many times. The software parts will be developed and improved gradually through each development cycle, continuously integrating user input and gradually adding functions. Each part of the project will have complete functions and can operate independently and each version will be supplemented with new and expanded functions. Some reasons for choosing this development model:

- Flexibility: Because the model is iterative, it is easy to change the requirements through iterations.
- Risk management: Reduce risks by detecting and handling problems at an early stage.
- Continuous feedback: By developing software in small steps, customers can test and give feedback continuously.
- Quality assurance: Because it is tested at each stage, the quality of the product is very reliable.
- Measurable progress: Set clear goals to track the progress of the project.

However, Iterative and Incremental model also has some disadvantages that we need to know when using it:

- Difficulty in managing the organization: Continuous iterations and step-by-step delivery can be difficult to manage and require close coordination.
- Need for a long-term vision: Continuous loops develop step by step, but to ensure smoothness between iterations, an overall view of the product is needed.

In this project, we plan to divide it into 4 specific iterations as follows:

- **Project Initiating (1 week - from 02/09/2024 to 08/09/2024)**
  - Kick-off Meeting
  - Determine Project Scope and Features
  - Prepare Project Introduction (Report 1)
  - Create preliminary Project Plan (Report 2 - Version 0.9)
- **Iteration 1 (4 weeks - from 09/09/2024 to 06/10/2024)**

- Identify functions for iteration 1, update Project Plan (Report 2 - Version 1.0)
- Create high-level requirements and analyze detailed requirements for iteration 1 and start Software Requirement Specification (Report 3 - Version 1.0)
- Create high-level design and analyze detailed design for iteration 1 features and start Software Design Document (Report 4- Version 1.0)
- Coding the planned functions
- Conduct testing and bug fixing (Report 5.1, 5.2, 5.3 - Version 1.0)
- Evaluate iteration effectiveness

● **Iteration 2 (3 weeks - from 07/10/2024 to 27/10/2024)**
- Identify functions for iteration 2, update Project Plan (Report 2 - Version 1.1)
- Revise requirements (if requirements change) and analyze detailed requirements for iteration 2 features and update Software Requirement Specification (Report 3 - Version 1.1)
- Analyze detailed design for iteration 2 features and update Software Design Document (Report 4 - Version 1.1)
- Proceed with coding the planned functions
- Conduct testing and bug fixing (Update report 5.1, 5.2, 5.3 - Version 1.1)
- Evaluate iteration effectiveness

● **Iteration 3 (3 weeks - from 28/10/2024 to 17/11/2024)**
- Identify functions for iteration 3, update Project Plan (Report 2 - Version 1.2)
- Revise requirements (if requirements change) and analyze detailed requirements for iteration 3 features and update Software Requirement Specification (Report 3 - Version 1.2)
- Analyze detailed design for iteration 3 features and update Software Design Document (Report 4 - Version 1.2)
- Proceed with coding the planned functions
- Conduct testing and bug fixing (Update report 5.1, 5.2, 5.3 - Version 1.2)
- Evaluate iteration effectiveness

● **Iteration 4 (3 weeks - from 18/11/2024 to 08/12/2024)**
- Identify functions for iteration 4, update project plan (Report 2 - Version 1.3)

- Revise requirements (if requirements change) and analyze detailed requirements for iteration 4 features and update Software Requirement Specification (Report 3 - Version 1.3)
- Analyze detailed design for iteration 4 features and update Software Design Document (Report 4 - Version 1.3)
- Proceed with coding the planned functions
- Conduct testing and bug fixing (Update report 5.1, 5.2, 5.3 - Version 1.3)
- Proceed with project deployment
- Prepare Report 6 - User Guide
- Evaluate iteration effectiveness

● **Project Closing (1 week - from 09/12/2024 to 15/12/2024)**
- Report 7 - Final Report
- Submit The Capstone Project
- Prepare Thesis Defense Slides and Practical Presentation
- Evaluate project effectiveness

2.2 Quality Management

2.2.1 Defect Prevention

- To minimize potential errors before developing the product
- Design the system architecture carefully
- Testers must collect a list of common errors and plan to make changes.
- There must be a plan to train and guide members on common problems and errors during the development process.
- Managers need to determine the appropriate plan and should make some changes if necessary.

2.2.2 Reviewing

- After understanding the customer needs, review the requirements from members, developers, BAs or customers to ensure all perspectives are in sync and can be thoroughly reviewed before proceeding to the next steps.
- Review documents thoroughly to ensure accuracy and completeness.
- Developers perform code reviews to detect logical or syntactic errors.

- Compare the application with existing systems to ensure the work is being done in the right direction.

2.2.3 Unit Testing

- Ensure that each small component of the software works as expected by writing small, separate unit tests with logical names for maintenance.
- Use automated testing frameworks to easily automate testing.
- Ensure that the tests run quickly and are reusable.
- Ensure that all edge cases are fully tested.

2.2.4 Integration Testing

- Make sure that unit testing is done before moving on to Integration Testing.
- Make sure the modules are working properly after integration.
- Prepare integration test plans and reports
- Fully prepare test cases and test data.

2.2.5 System Testing

- Ensure integration testing phases are completed.
- Ensure functional and non-functional requirements are met.
- To ensure smooth and reliable operation of the software, system testing must be planned and testing should mainly focus on performance, security, recovery, external interface, user-system interaction, etc.

2.2.6 Acceptance Testing

- Ensure system testing phases are completed.
- Ensure that all business, functional, and non-functional requirements are met.
- Ensure smooth and intuitive interaction between the user and the system, with workflows reflecting actual use cases.
- Verify that the software meets all acceptance criteria as defined by stakeholders for final approval and sign-off for deployment.

2.3 Project Training Plan

Training Area	Participants	When, Duration	Waiver Criteria

ASP.NET	NinhNT, ManhDD, PhucND, ThuyDTT	14/9- 20/9	Mandatory
React	NinhNT, ManhDD, HauNX	14/9- 20/9	Mandatory
Git, Github	All Member	14/9- 15/9	Mandatory
Testing	All Member	20/9- 25/9	Mandatory

Table 9: Training plan

3. Project Deliverables

#	Deliverable	Due Date	Notes
Project Initiating (from 02/09/2024 to 08/09/2024)			
1	Project Introduction	08/09/2024	Prepare an overview of the project: background, objective, and scope
2	Project Plan v0.9	08/09/2024	Prepare Project plan at a high level
Iteration 1 (from 09/09/2024 to 06/10/2024)			
3	Project Plan v1.0	09/09/2024	Identify functions for iteration 1, update Project Plan
4	Software Requirement Specification v1.0	16/09/2024	High-level requirements and analyze detailed requirements for iteration 1
5	Software Design Document v1.0	22/09/2024	High-level design and analyze detailed design for iteration 1 features and start Software Design Document
6	Software Test Document (STD)	01/10/2024	Prepare testing results: Unit Test, Integration Test, System Test, Acceptance Test
6	Product Release 1	06/10/2024	Product Release after Iteration 1
Iteration 2 (from 07/10/2024 to 27/10/2024)			

7	Project Plan v1.1	07/10/2024	Identify functions for iteration 2, update Project Plan
8	Software Requirement Specification v1.1	12/10/2024	Update Software Requirement Specification with analysis of detailed requirements for iteration 2
9	Software Design Document v1.1	17/10/2024	Update the Software Design Document with the analysis of the detailed design for iteration 2 features
10	Software Test Document (STD)	23/10/2024	Prepare testing results: Unit Test, Integration Test, System Test, Acceptance Test
11	Product Release 2	27/10/2024	Product Release after Iteration 2
Iteration 3 (from 28/10/2024 to 17/11/2024)			
12	Project Plan v1.2	28/10/2024	Identify functions for iteration 3, update Project Plan
13	Software Requirement Specification v1.2	02/11/2024	Update Software Requirement Specification with analysis of detailed requirements for iteration 3
14	Software Design Document v1.2	07/11/2024	Update the Software Design Document with analysis of the detailed design for iteration 3 features
15	Software Test Document (STD)	13/11/2024	Prepare testing results: Unit Test, Integration Test, System Test, Acceptance Test
16	Product Release 3	17/11/2024	Product Release after Iteration 3
Iteration 4 (from 18/11/2024 to 08/12/2024)			
17	Project Plan v1.3	19/11/2024	Identify functions for iteration 4, update Project Plan
18	Software Requirement Specification v1.3	24/11/2024	Update Software Requirement Specification with analysis of detailed requirements for iteration 4

19	Software Design Document v1.3	29/11/2024	Update the Software Design Document with analysis of the detailed design for iteration 4 features
20	Software Test Document (STD)	30/11/2024	Prepare testing results: Unit Test, Integration Test, System Test, Acceptance Test
21	Product Release 4	04/12/2024	Product Release after Iteration 4
22	User Guide Document	04/12/2024	Provide straightforward instructions for using the developed software, ensuring the language is simple and direct.
Project Closing (from 09/12/2024 to 15/12/2024)			
23	Final Project Report	09/12/2024	Includes all the previous project documents in combined, source code, and thesis presentation slide

Table 10: Project Deliverables

4. Responsibility Assignments

4.1 Project Organization

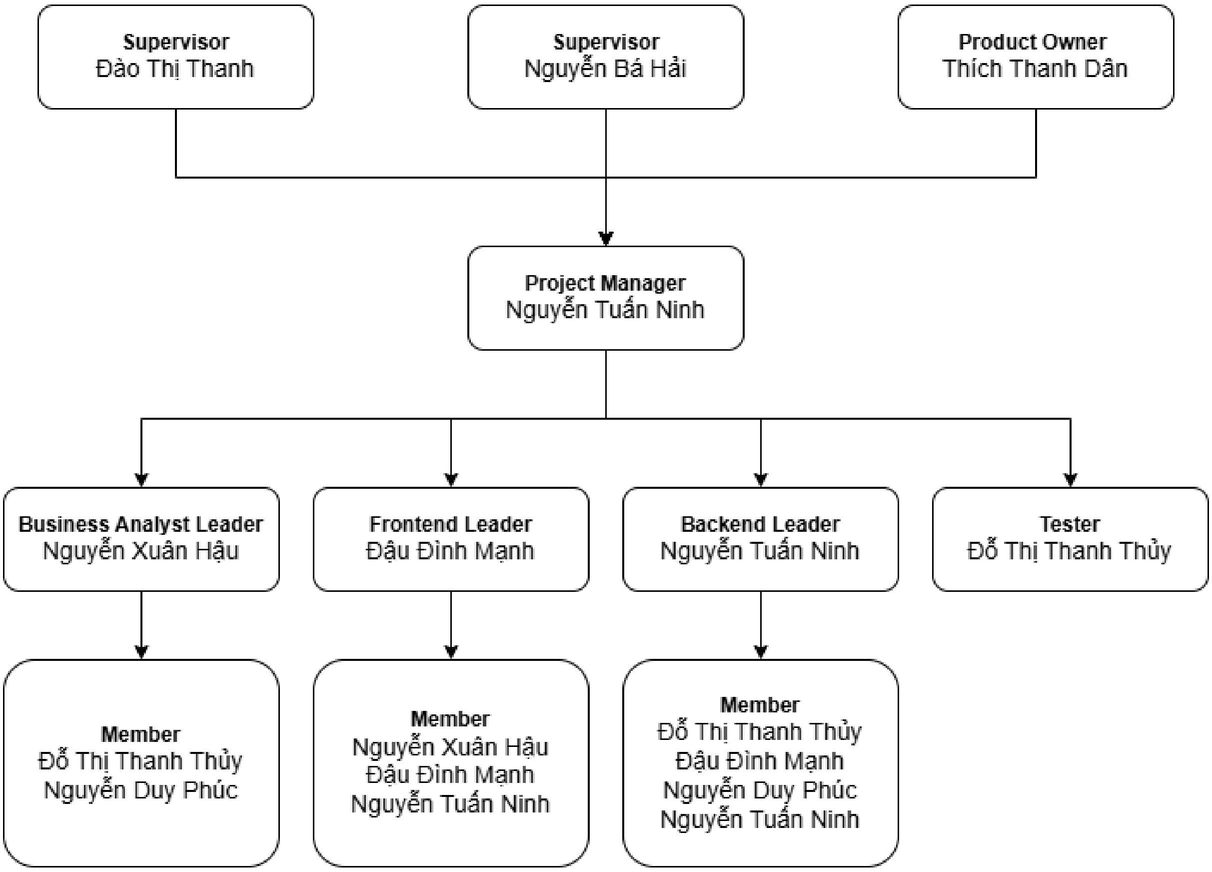


Figure 6: Project Organization

4.2 Role and responsibilities

Role	Responsibilities
Supervisor	Oversees the overall project to ensure alignment with strategic goals, provides guidance and mentorship to the project team, ensures that the project objectives are met, and approves major project deliverables and milestones.
Product Owner	The Product Owner defines and prioritizes the product backlog, ensuring alignment with business goals and user needs.
Project Manager	Manages the project from inception to completion, ensuring it stays on schedule, creates and maintains project plans, allocates resources and assigns tasks, and monitors progress and performance.
Business Analyst Leader	Leads the business analysis efforts, ensuring that all business requirements are properly understood and met, analyses and

	documents business processes and requirements, and works with stakeholders to gather feedback
Business Analyst	Supports the Business Analyst Leader in gathering and analyzing requirements, conducting interviews and workshops with stakeholders, and translating business requirements into technical specifications
Front end Leader	Leads the front-end development team, oversees the design and development of the front-end interface reviews code, and provides feedback to front-end members
Front end member	Develops and implements the front-end interface as per the project requirements, and writes front-end code.
Back end Leader	Leads the back-end development team, ensuring the proper functionality of the server, database, and application logic, Oversee back-end architecture and implementation.
Front end member	Implements the back-end services and ensures integration with the front end, writes back-end code
Tester leader	Leads the testing team to ensure the quality of the product before release, develops test plans and strategies, oversees the execution of test cases and reports defects, and ensures proper documentation of testing results.
Tester	Executes tests, and identifies defects in the system. Testers also documents, and reports bugs to the development team, and re-test fixes to ensure issues are resolved.

Table 11: Row and responsibilities

4.3 Responsibility assignments

D~Do; R~Review; S~Support; I~Informed; <blank>- Omitted

Responsibility	NinhNT	HauNX	PhucND	ThuyDTT	ManhDD
Document and Report					
Project Planning & Tracking	D	S, R	I	I	I
Prepare Project Introduction Document	I	D, R	D	S	S
Prepare SRS Document	D	D, R	D	D	D
Prepare SDS Document	D, R	D	D	S	D
Prepare Test Documents	D	D	D	D, R	D
Prepare User Guides	S	D, R	D	S	S
Prepare Final Report	D	D, R	D	D	D
Prepare Presentation Slide	R	D	S	D	S
Design					
System Architecture Design	D, R	S	I	I	I
Class Diagram	R	I	S	S	D
Sequence Diagram	D, R	I	D	D	D
Use Case Specification	D, R	S	D	D	D
Screen Design	R	S	D	D	I
Database Design	D, R	S	I	s	I
Implementation					

Control source code	D, R	I	I	I	I
Code front-end	D	D	I	D	D, R
Code back-end	D, R	I	D	D	D
Deploy web	D, R	I	I	I	I
Test					
Create test plan	R	I	I	D	I
Create test report	D	I	D	R	D

Table 12: Responsibility assignments

5. Project Communications

Communication Item	Who/ Target	Purpose	When, Frequency	Type, Tool, Method(s)
Daily meeting	All Member	- Check progress - Discuss problem	21:00 every day	Google Meet
Weekly meeting with supervisors	All Member, supervisor	- Report progress	3 times a week	Google Meet, Microsoft team
Team Messaging	All Member, supervisor	- Notice about unscheduled meeting - Ask supervisors about the issues to get advice	When necessary	Online - Messenger, Zalo
Unscheduled Meeting	All Member, supervisor	- Discuss and solve problems	When there are problems that needs to be resolved immediately	Google Meet/Offline
Meeting with customers	All Member, customer	- Discuss about requirements	1 time a month	Google Meet/Offline

		- Update the project status to customer		

Table 13: Communication Plan

6. Configuration Management

6.1 Document Management

The project uses Google Docs and Google Sheets as our primary tools for sharing, editing, and version control of our documents. It allows us to observe what has changed in the documents and who is responsible for that modification.

The project has a folder on Google Drive named Documents. It is shared between members and supervisors.

The folder includes following files:

- Report1_Project Introduction
- Report2_Project Management Plan
- Report3_Software Requirement Specification
- Report4_Software Design Document
- Report5.0_Test Documentation
- Report5.1_Unit Test
- Report5.2_Integration Test
- Report6_Software User Guides
- Report7_Final Project Report

6.2 Source Code Management

The project uses Gitlab for managing source code because It makes it easy for developers to share code files and collaborate with fellow developers on open-source projects. GitHub also serves as a social networking site where developers can openly network, collaborate, and pitch their work.

The project has 2 Gitlab repositories: SCCMS_API and SCCMS_WebApp

6.3 Tools & Infrastructures

Category	Tools / Infrastructure
Technology	React 18(FrontEnd), ASP.NET 8 (BackEnd), SignalR(notification)
Database	SQL Server
IDEs/Editors	Visual Studio Code, Visual Studio
Diagramming	DrawIO, Lucid Chart, Astah Professional
Documentation	Ms Office, Google Docs/Sheets/Slides
Version Control	GitLab (Source Codes), Google Drive (Documents)
Deployment server	Azure Web Service
Project management	Google Sheet (Schedule, Tasks, Defects), GitLab (Issue Management)

Table 14: Tools and Infrastructures

III. Software Requirement Specification

Below is how our team defined the main content for the report 3. Other details are defined in detail in the document:Report3_Software Requirement Specification

1. Requirement Overview

1.1 Context Diagram

The "Cultivation Course Management System for Co Loan Pagoda" is designed to streamline the organization and operation of summer cultivation course activities. The system provides key functionalities such as participant and volunteer registration, night shift management, and the ability to generate reports during the course. It aims to increase efficiency, reduce manual tasks, and improve the overall experience for participants and organizers. Core features include online registration, personal information management, and tools for fostering communication between participants and instructors. It also integrates reporting and statistical analysis to assist organizers in evaluating and adjusting activities.

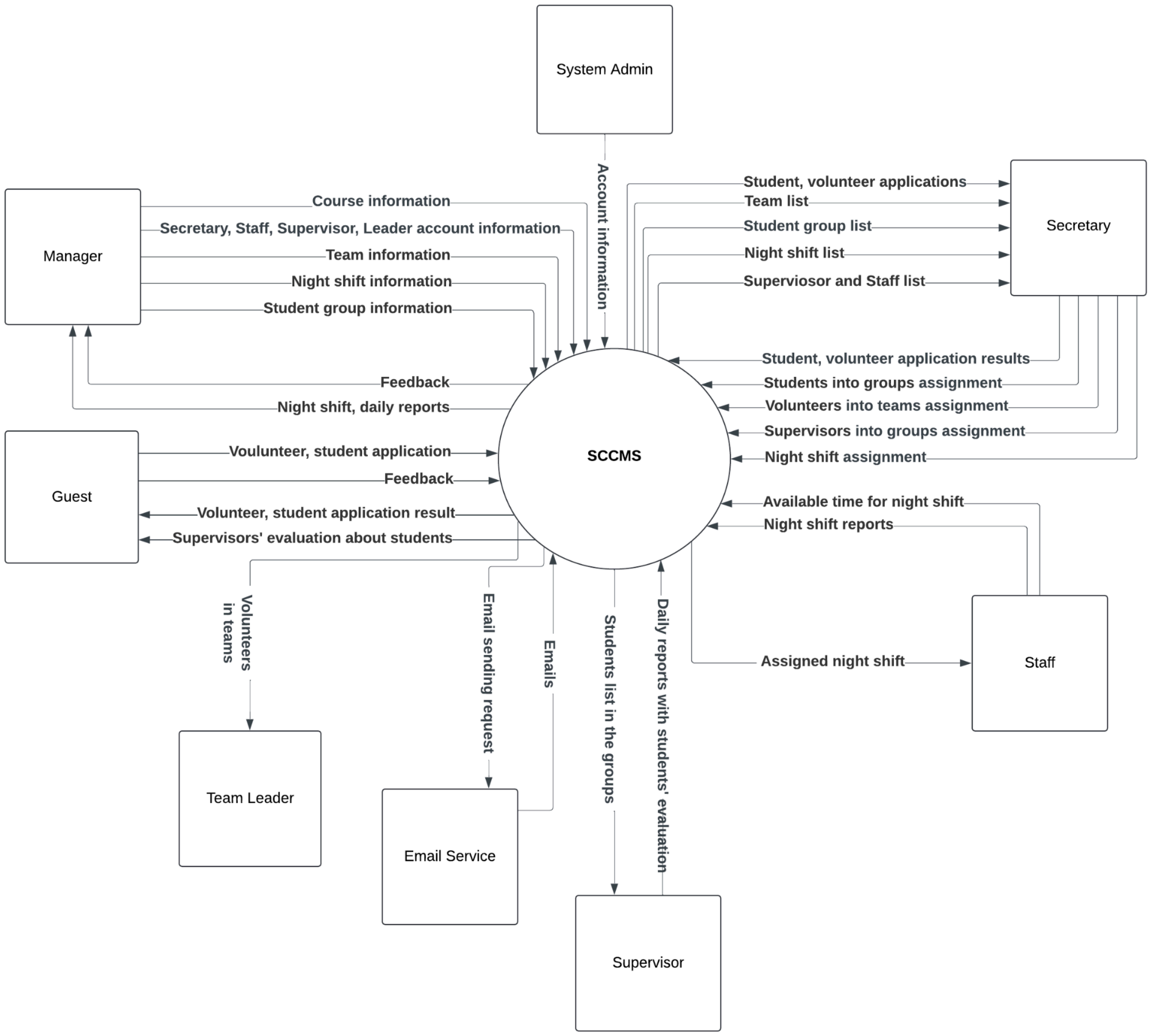


Figure 7: Context Diagram for the system

The link of context diagram: [Context Diagram]

The system is primarily built for seven main user roles: system admin, manager, secretary, staff, team leaders, supervisor and guest, each with customized features. This application is especially beneficial for secretaries because it can reduce the time spent on manual activities to a minimum, such as automatically creating cards for students, volunteers and leaders, automatically dividing classes, and sending mass mails. At the same time, the system also helps improve the process of viewing and sending reports to managers and staff.

1.2 Main Workflows

1.2.1 Initialize and setup cultivation course

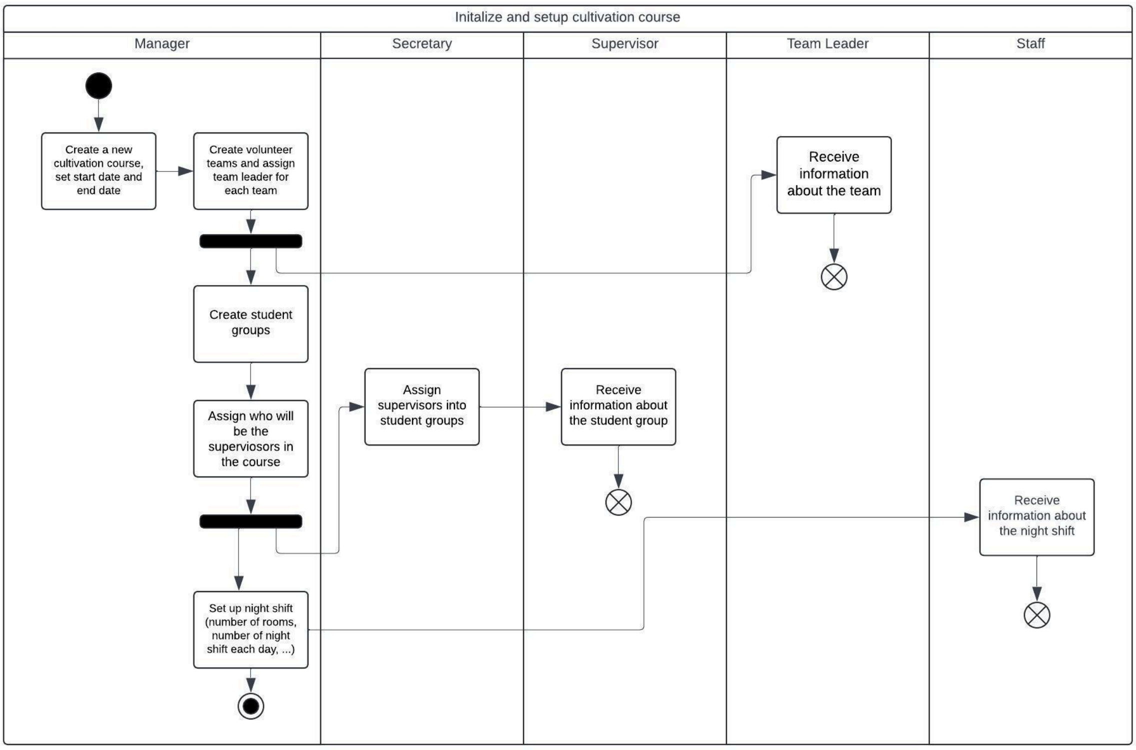


Figure 8 : Swimlane diagram of initializing and setuping cultivation course

The link of swimlane diagram: [Swimlane - Initialize and setup cultivation course]

1.2.2 Manage student from registration to graduation

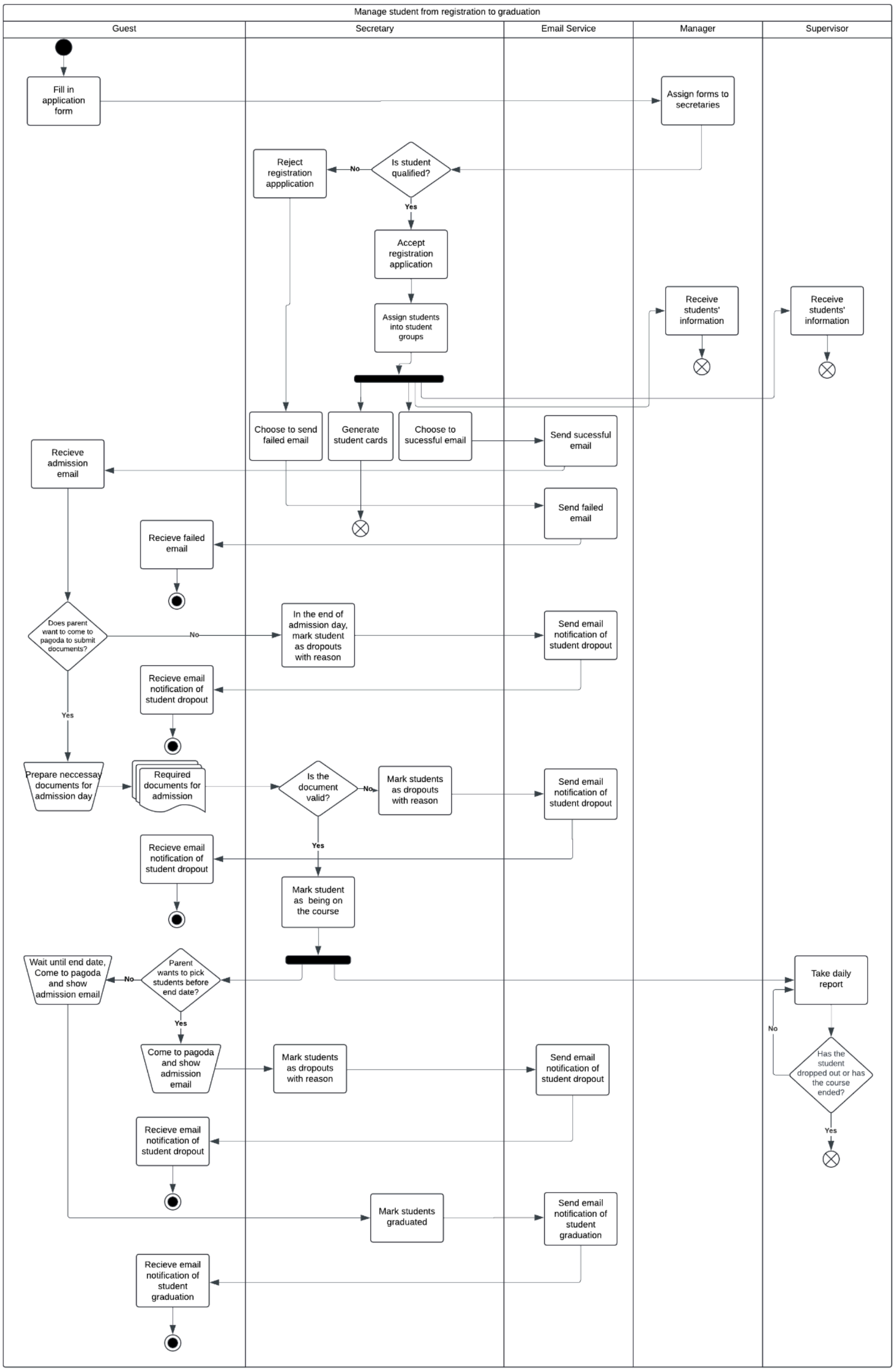


Figure 9: Swimlane diagram of managing students from registration to graduation

The link of swimlane diagram: [Swimlane - Manage student from registration to graduation]

1.2.3 Manage volunteer from registration to end date

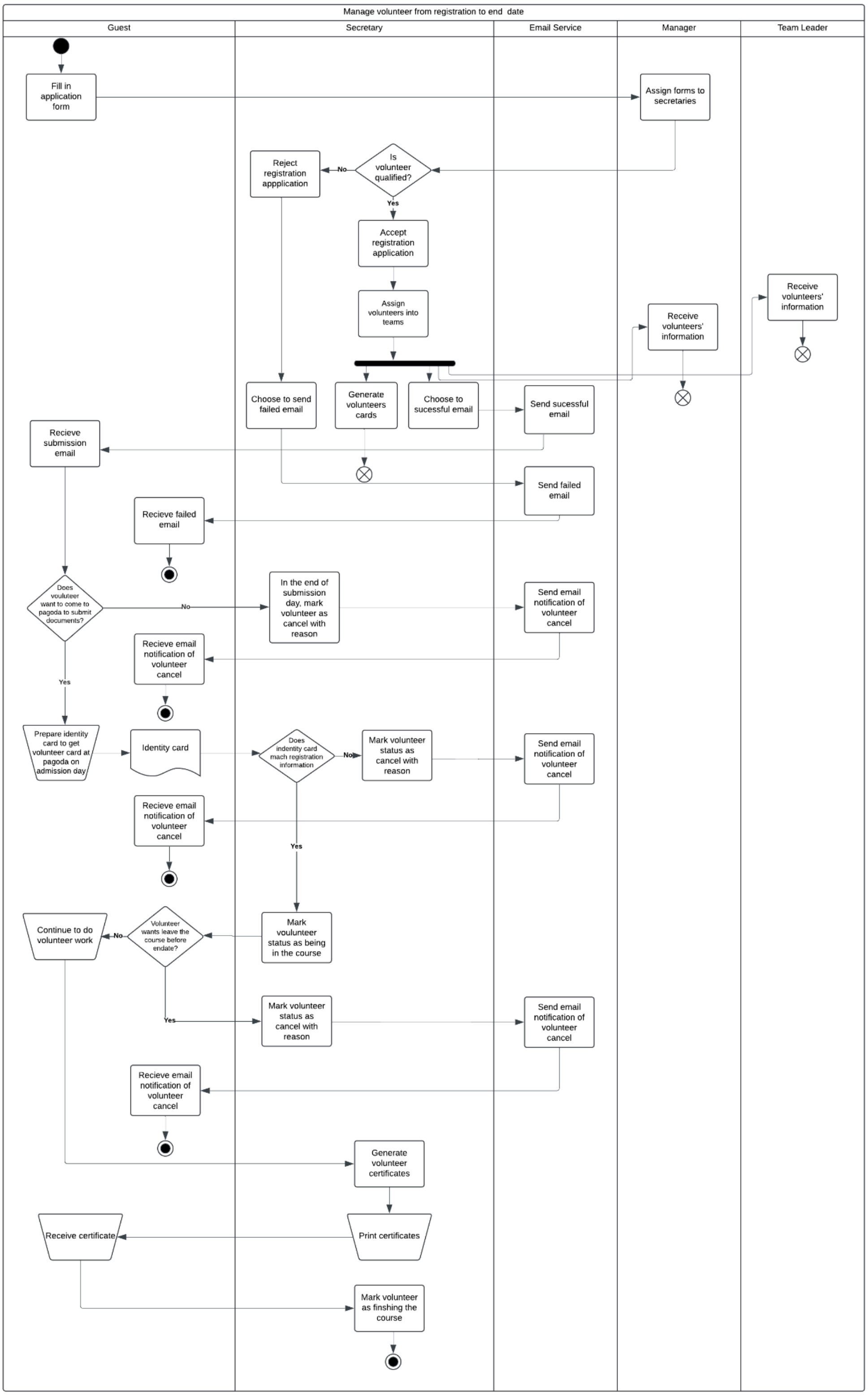


Figure 10: Swimlane diagram of managing volunteer from registration to end date

The link of swimlane diagram: [Swimlane - Manage volunteer from registration to end date]

1.2.4 Manage Night Shift

The main process for assigning the shift scheduling tasks to employees is based on the Greedy Algorithm. The goal is to ensure that each shift in the departments is allocated enough employees according to the gender and number requirements, while respecting the availability of each employee during the period from the start date to the end date of the course

Candidates: The Candidate Set is a list of employees who can be assigned to shifts. Each candidate (employee) has information about their availability and gender, which is provided by the system input.

Selection Function: Select employees whose availability matches the shift duration and meets the gender and number requirements for the department.

Feasibility Function: Checks whether a candidate can be assigned to a shift.

Objective Function: Evaluate the number of employees assigned to each room in each shift, so that the required number and gender are met. Ensure that all shifts have enough employees to meet the gender requirements.

Evaluation Function: The solution is considered complete when all shifts have been assigned enough employees according to the number and gender requirements and no employees have duplicate schedules or all employees have been scheduled.

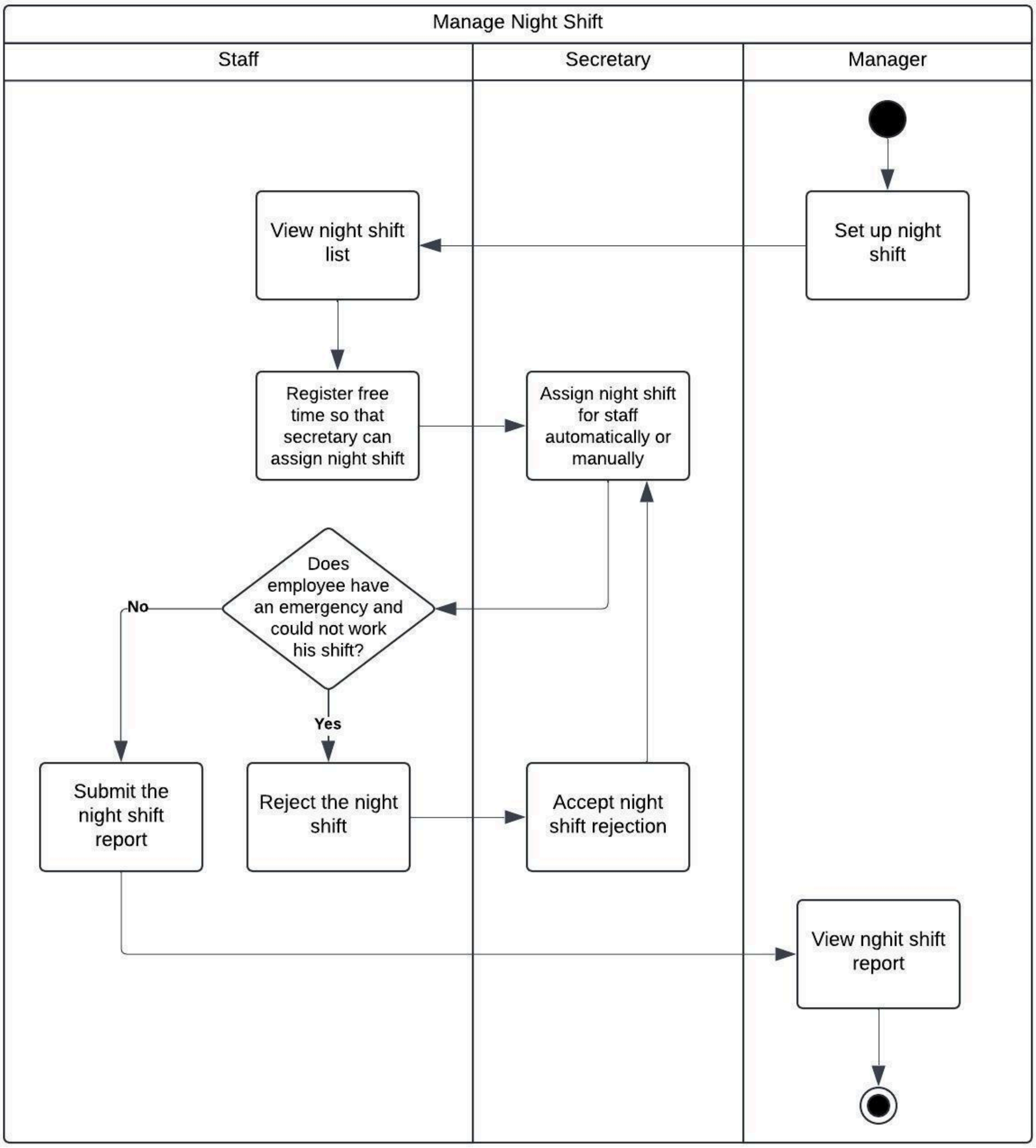


Figure 11: Swimlane diagram of managing the night shift

The link of swimlane diagram: [Swimlane - Manage Night Shift]

2. User requirements

2.1 Actors

#	Actor	Description
1	System admin	System admins manage the accounts of managers
2	SCCMS's User	Includes actors that interact with the system such as: secretary, staff, team leader, supervisor, manager, admin
3	Manager	As a monk in the pagoda, responsible for managing all cultivation courses. Managers oversee the creation and management of courses, teams, and staff. They need to be aware of night shift assignments, reports, and also manage posts and view feedback.
4	Secretary	As a Buddhist follower of the pagoda, responsible for managing personnel. The secretary will assign and arrange volunteers into teams and distribute students into groups.
5	Staff	As a Buddhist follower of the pagoda, responsible for managing volunteers, overseeing a group of students, or participating in night shifts.
6	Team leader	In the pagoda, there are usually divisions such as: mobile division, cleaning division, cooking division, etc. The team leader is the head of the division, the person who manages these divisions.
7	Supervisor	In a course that will include many classes, the Supervisor will be the one to manage the students and submit reports.
8	Guest	As someone who can view posts, submit registration forms for their child to join the cultivation course, and register to become a volunteer.
9	Email service	As a service that sends emails to parents, volunteers, and sends notification emails.

Table 15: Actor description

2.2 Use Cases

2.2.1 Diagram(s)

2.2.1.1. The relationship of actors in the system

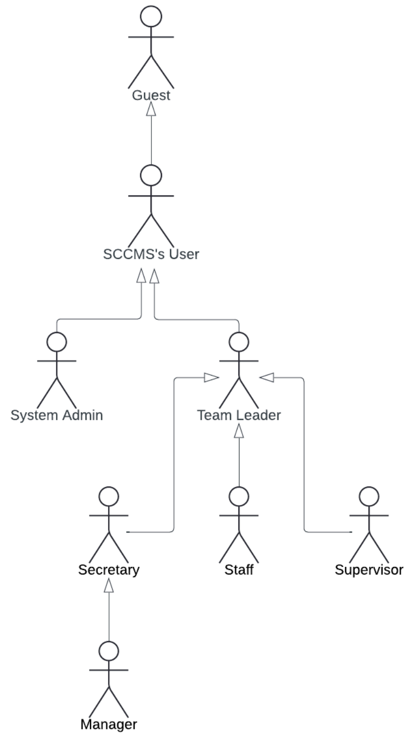


Figure 12 The relationship between the actors in the system

The link of relationship between the actors in the system: Use case diagram

2.2.1.2. Use cases for Guest

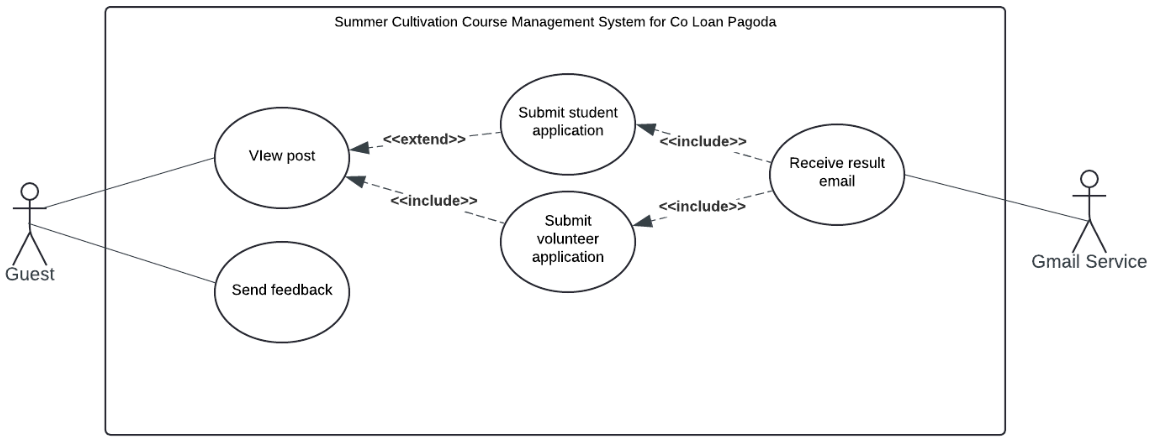


Figure 13: Use case diagram for guest

The link of Use case diagram for guest: Use case diagram

2.2.1.3. Use cases for SCCMS's User

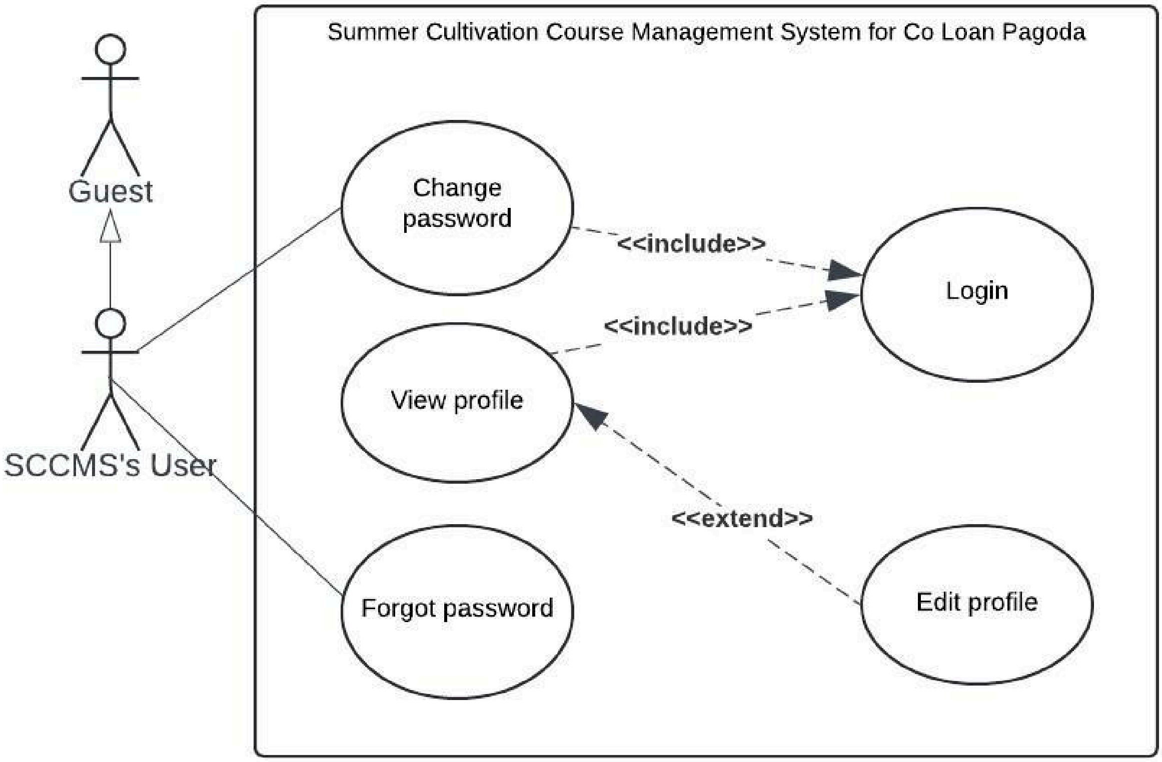


Figure 14: Use case diagram for SCCMS's user

The link of Use case diagram for user: [Use case diagram]

2.2.1.4. Use cases for Admin

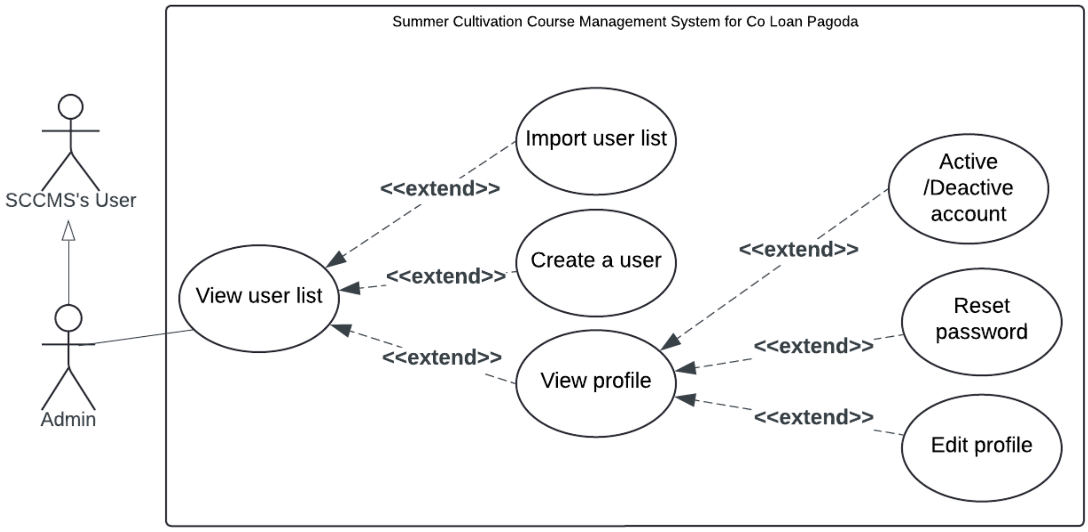


Figure 15: Use case diagram for admin

The link of Use case diagram for admin: [Use case diagram]()

2.2.1.5. Use cases for Team Leader

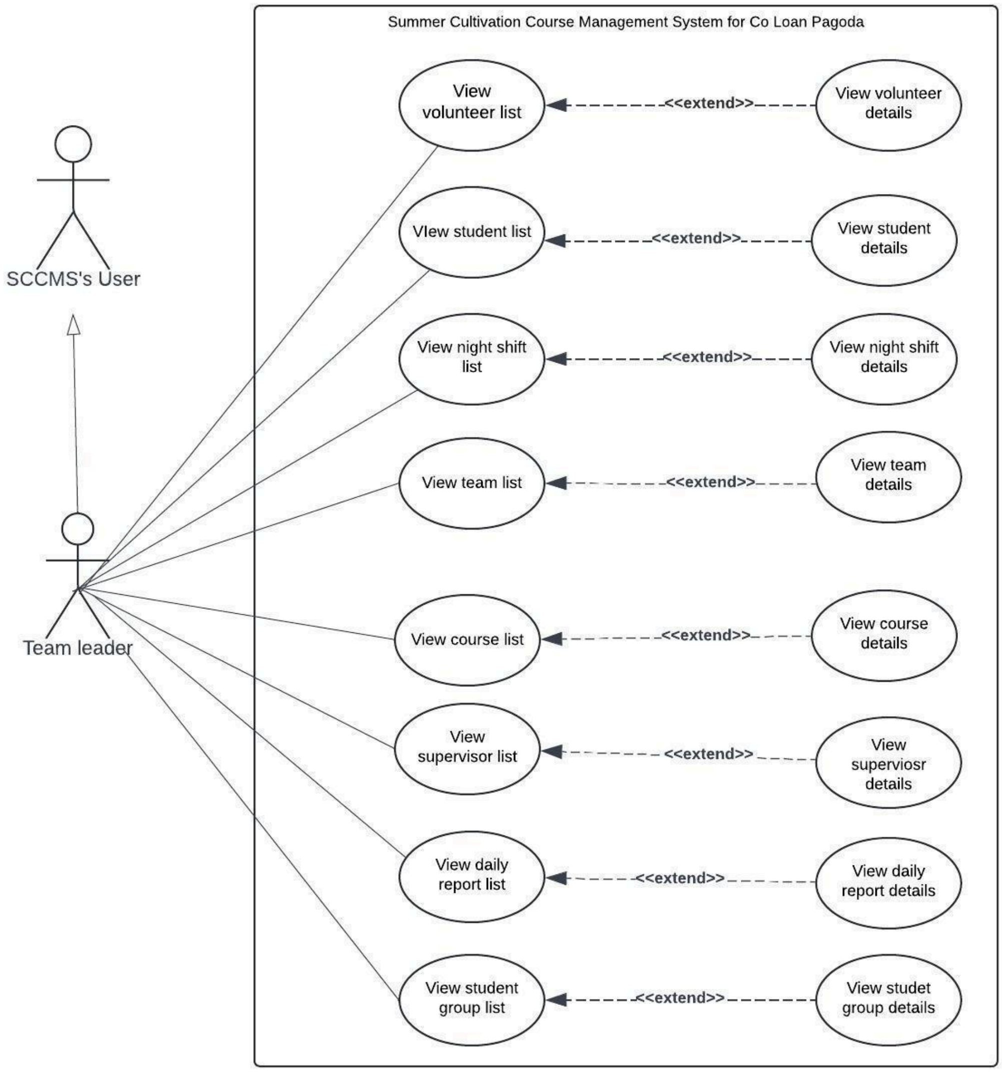


Figure 16: Use case diagram for team leader

The link of Use case diagram for team leader: [Use case diagram]()

2.2.1.6. Use cases for Staff

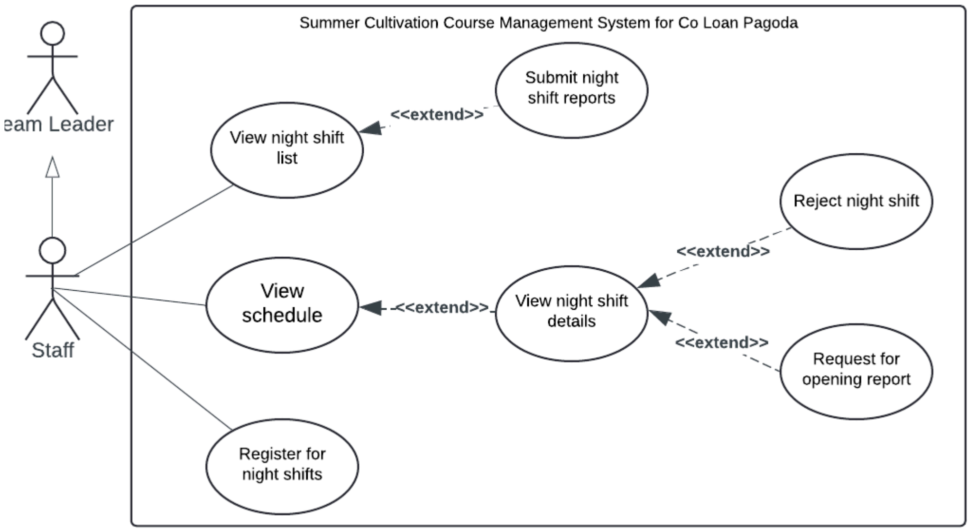


Figure 18: Use case diagram for staff

The link of Use case diagram for staff: [Use case diagram]

2.2.1.7. Use cases for Supervisor

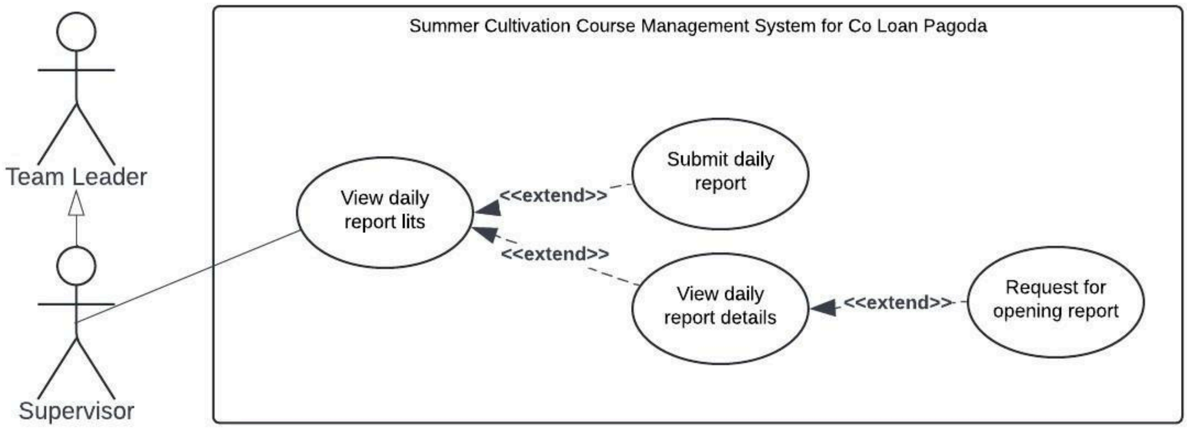


Figure 18: Use case diagram for supervisor

The link of Use case diagram for supervisor: [Use case diagram]

2.2.1.9. Use cases for Secretary

Figure 19: Use case diagram for secretary

The link of Use case diagram for secretary: [Use case diagram]

2.2.1.10. Use cases for Manager

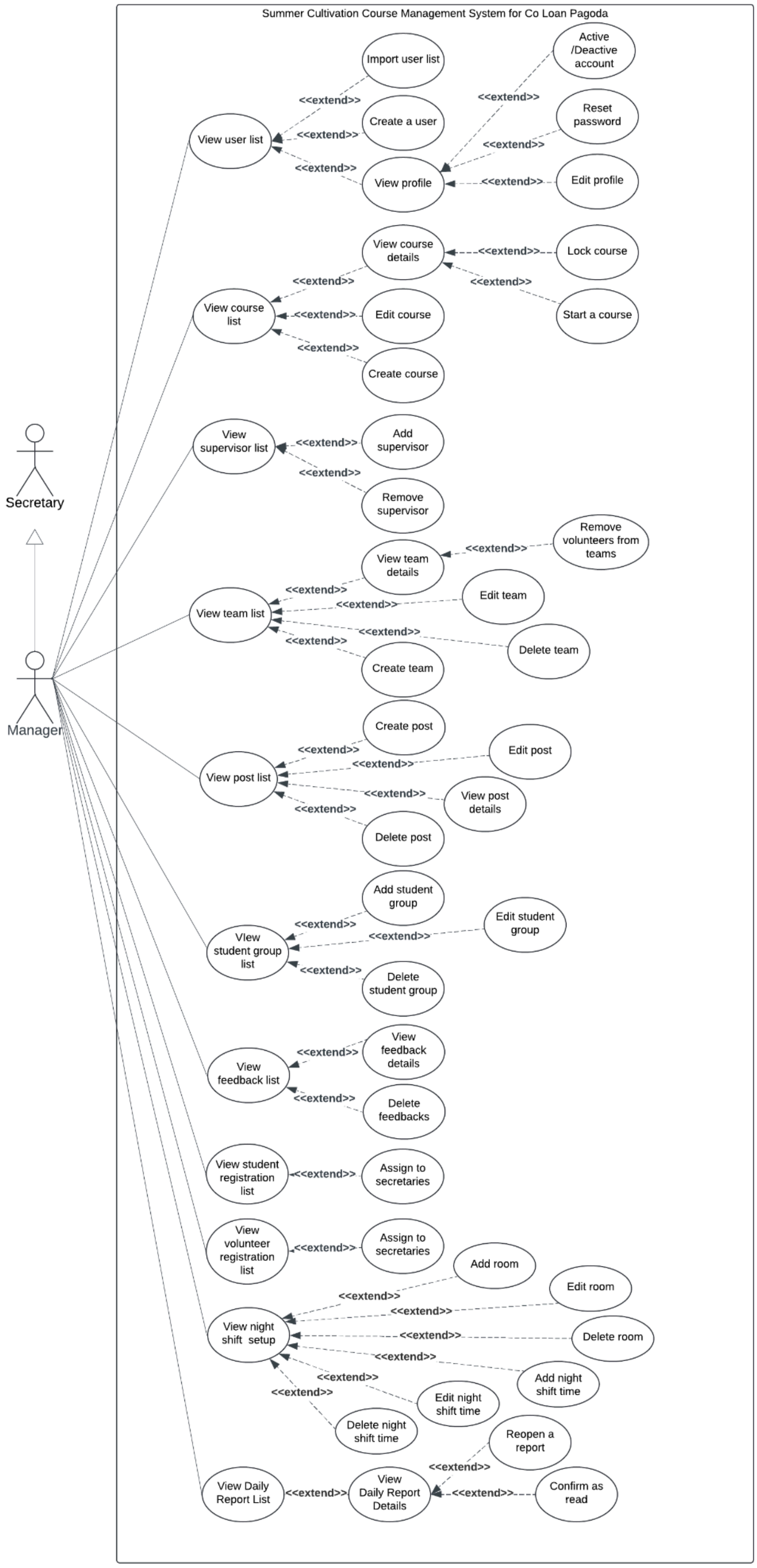


Figure 20: Use case diagram for manager
The link of Use case diagram for manager: [Use case diagram]

2.2.2 Descriptions

ID	Use Case	Actors	Use Case Description
UC-01	Login	System Admin, Manager, Secretary, Staff, Team Leader, Supervisor	Enables users to authenticate and gain access to the system by entering valid credentials.
UC-02	Logout	System Admin, Manager, Secretary, Staff, Team Leader, Supervisor	Enables users to logout from the system.
UC-03	View profile	System Admin, Manager, Secretary, Staff, Team Leader, Supervisor	Allows users to view their personal profile information, including personal details. System Admin can view profile of managers and managers can view profile of Secretary, Staff, Team Leader, Supervisor.
UC-04	Edit profile	System Admin, Manager, Secretary, Staff, Team Leader, Supervisor	Allows users to edit and update their personal profile information to keep it current and accurate. System Admin can edit profile of managers and managers can edit profile of Secretary, Staff, Team Leader, Supervisor.
UC-05	View user list	System Admin, Manager	Allows system admins and managers to view a list of all user accounts within the system for management purposes.

UC-06	Change password	System Admin, Manager, Secretary, Staff, Team Leader, Supervisor	Allows users to change their account password to maintain or enhance account security.
UC-07	Forgot password	System Admin, Manager, Secretary, Staff, Team Leader, Supervisor	Provides a mechanism for users to reset their password if forgotten, through email verification.
UC-08	Reset user password	System Admin, Manager	Enables admins to reset a manager's password and managers to reset password of secretary, staff, team leader and supervisor.
UC-09	Create a user	System Admin, Manager	Allows administrators to create a new user account individually, specifying all necessary details.
UC-10	Import user list	System Admin, Manager	Enables the importation of user account data from Excel files into the system.
UC-11	Activate/ Deactivate account	System Admin, Manager	Allows system admins and managers to activate or deactivate user accounts, controlling user access to the system. System admin has the right to activate or deactivate managers' accounts, whereas managers can activate or deactivate accounts of secretary, staff, team leader and supervisor.
UC-12	View course list	Manager, Secretary, Staff, Team Leader, Supervisor	Allows users to view a list of all courses.

UC-13	Create course	Manager	Enables the manager to create a new course by specifying all necessary information.
UC-14	View course details	Manager, Secretary, Staff, Team Leader, Supervisor	Enables users to view detailed information about a specific course.
UC-15	Edit course	Manager	Allows the manager to modify course details.
UC-16	Start a course	Manager	Allows the manager start recruiting
UC-17	Lock course	Manager	Allows the manager to lock a course, preventing further enrollments or modifications.
UC-18	Submit student application form	Guest	Enables users to submit an application form to enroll as a student in a course or program.
UC-19	View student registration list	Manager, Secretary	Allows viewing of all student registration applications for review and processing.
UC-20	View registration details	Manager, Secretary	Enables viewing detailed information of a specific student registration application.
UC-21	Accept/Reject registration	Manager, Secretary	Allows the manager or secretary to approve or deny student registration applications based on eligibility criteria.
UC-22	View student list	Manager, Secretary, Staff, Team Leader, Supervisor	Enables viewing a list of all enrolled students within the system.
UC-23	View student details	Manager, Secretary, Staff, Team Leader, Supervisor	Enables viewing detailed information about a specific student, including contact information and student status.

UC-24	Edit student information	Manager, Secretary	Allows editing of student information to keep records accurate and up-to-date.
UC-25	Edit student in course	Manager, Secretary	Enables editing of student enrollment within a specific course.
UC-26	Automatically assign registrations to secretaries	Manager, Secretary	Enables the system to automatically assign student registration applications to secretaries based on predefined criteria.
UC-27	Automatically assign students to groups	Manager, Secretary	Allows the system to automatically assign enrolled students to groups based on specific rules or criteria.
UC-28	Manually assign students to group	Manager, Secretary	Enables manual assignment of students to specific groups as needed.
UC-29	Export student list	Manager, Secretary	Allows exporting the student list to external files or formats for reporting or analysis purposes.
UC-30	Generate student cards	Manager, Secretary	Enables the creation of student identification cards containing relevant student information.
UC-31	Send email	Manager, Secretary	Allows sending emails to users for notifications, updates, or communications related to their accounts or activities within the system.
UC-32	View student group list	Manager, Secretary, Staff, Team Leader, Supervisor	Enables viewing a list of all student groups within the system.

UC-33	Add student group	Manager	Allows the manager to create new student groups for organizational purposes.
UC-34	Edit student group	Manager	Enables the manager to modify the details of existing student groups.
UC-35	View student group details	Manager, Secretary, Staff, Team Leader, Supervisor	Allows viewing detailed information about a specific student group.
UC-36	Remove students from student group	Manager, Secretary	Enables removal of students from a group as needed.
UC-37	View student's evaluation	Guest	Enables user to get evaluation about student by input the course and input student code on the homepage
UC-38	Delete student group	Manager	Enables the manager to delete a student group from the system when it's no longer needed.
UC-39	View supervisor list	Manager, Secretary, Staff, Team Leader, Supervisor	Allows viewing a list of supervisors involved in courses or student groups.
UC-40	Add supervisors	Manager	Enables the manager to assign supervisors to courses for instructional or oversight purposes.
UC-41	Delete supervisor	Manager, Secretary	Enables removal of supervisors from student groups as necessary.
UC-42	Manually assign supervisors to student group	Manager, Secretary	Enables assignment of supervisors to specific student groups manually.

UC-43	Automatically assign supervisors to student group	Manager, Secretary	Allows the system to automatically assign supervisors to student groups based on predefined criteria.
UC-44	View team list	Manager, Secretary, Staff, Team Leader, Supervisor	Enables viewing a list of teams within the organization or system.
UC-45	View team details	Manager, Secretary, Staff, Team Leader, Supervisor	Allows viewing detailed information about a specific team, including members and objectives.
UC-46	Add team	Manager	Enables the manager to form new teams.
UC-47	Edit team	Manager	Enables the manager to modify team information.
UC-48	Delete team	Manager	Allows the manager to remove a team from the system when it's no longer active.
UC-49	Remove volunteers from team	Manager, Secretary	Enables removal of volunteers from teams as needed.
UC-50	Submit volunteer application	Guest	Allows individuals to apply to become volunteers of the cultivation course.
UC-51	View volunteer registration list	Manager, Secretary	Enables viewing of all volunteer applications submitted for review.
UC-52	View volunteer registration details	Manager, Secretary	Enables viewing detailed information of a specific volunteer application.

UC-53	Accept/Reject volunteer registration	Manager, Secretary	Enables approval or rejection of volunteer applications based on eligibility and need.
UC-54	View volunteer list	Manager, Secretary, Staff, Team Leader, Supervisor	Allows viewing a list of all active volunteers within the organization.
UC-55	View volunteer details	Manager, Secretary, Staff, Team Leader, Supervisor	Enables viewing detailed information about a specific volunteer.
UC-56	Edit volunteer information	Manager, Secretary	Enables editing of volunteer information to maintain accurate records.
UC-57	Edit volunteer in course	Manager, Secretary	Allows editing of volunteer assignments within a specific course.
UC-58	Automatically assign volunteers to team	Manager, Secretary	Allows assigning approved volunteers to teams where their skills are needed automatically.
UC-59	Manually assign volunteers to team	Manager, Secretary	Allows assigning approved volunteers to teams manually.
UC-60	Automatically assign forms to secretaries	Manager, Secretary	Enables the system to automatically assign application forms to secretaries based on predefined criteria.
UC-61	Generate volunteer cards	Manager, Secretary	Allows creation of identification cards for volunteers.
UC-62	Generate volunteer certificates	Manager, Secretary	Enables the creation of certificates recognizing volunteer contributions and achievements.

UC-63	Export volunteer list	Manager, Secretary	Allows exporting the volunteer list for reporting or communication purposes.
UC-64	Register for freetime	Staff	Allows staff members to register for available free time slots.
UC-65	View night shift freetime	Manager, Secretary	Allows viewing of available free time slots for night shifts.
UC-66	View night shift list	Manager, Secretary, Staff, Team Leader, Supervisor	Enables viewing of scheduled night shifts.
UC-67	Auto assign night shift	Manager, Secretary	Allow manager or secretary automatically assign night shift to staff
UC-68	Manually assign a shift	Manager, Secretary	Allow manager or secretary assign night shift to staff
UC-69	Reject the shift	Staff	Enables staff members to decline an assigned shift due to conflicts or other reasons.
UC-70	View rejected shifts list	Manager, Secretary	Allows viewing shifts that have been declined by staff or volunteers, requiring reassignment.
UC-71	Accept a shift rejection	Manager, Secretary	Enables acknowledgment and processing of shift rejections, updating schedules accordingly.
UC-72	View night shift setup	Manager	Allows the manager to view and configure the setup for night shifts.
UC-73	Edit shift registration date	Manager	Allows managers to change staff shift registration time.
UC-74	View schedule	Staff	Allows staff to view their shifts.

UC-75	Delete a staff from shift	Manager, Secretary	Allows the manager or secretary to remove staff from the current shift.
UC-76	Add room	Manager	Enables the manager to add new rooms for accommodating shifts or activities.
UC-77	Update room	Manager	Allows the manager to update room details such as capacity, facilities, or availability.
UC-78	Delete room	Manager	Enables the manager to remove rooms from the system when they are no longer needed.
UC-79	Add night shift time	Manager	Allows the manager to define new time slots for night shifts.
UC-80	Edit night shift time	Manager	Enables the manager to modify existing night shift time slots.
UC-81	Delete night shift time	Manager	Allows the manager to remove night shift time slots that are no longer required.
UC-82	View post list	Manager	Allows viewing a list of posts such as news, updates, or announcements within the system.
UC-83	View post details	Manager	Enables viewing detailed content of a specific post.
UC-84	Add post	Manager	Allows the manager to create new posts for communication or informational purposes.
UC-85	Edit post	Manager	Enables the manager to modify existing posts.

UC-86	Delete post	Manager	Allows the manager to remove posts that are outdated or no longer relevant.
UC-87	View post list on homepage	System Admin, Manager, Secretary, Staff, Team Leader, Supervisor, Guest	Allows viewing a list of posts displayed on the homepage for users to access easily.
UC-88	View post details on homepage	System Admin, Manager, Secretary, Staff, Team Leader, Supervisor, Guest	Enables viewing detailed content of specific posts directly from the homepage.
UC-89	View report list	Manager, Secretary, Staff, Team Leader, Supervisor	Enables viewing a list of daily reports submitted by supervisors or staff.
UC-90	Submit report	Supervisor, Staff	Allows supervisors to submit daily reports for their teams or shifts.
UC-91	View report details	System Admin, Manager, Secretary, Staff, Team Leader, Supervisor	Enables viewing detailed information about specific daily reports.
UC-92	Request opening report	Staff, Supervisor	Allows users to request the opening or access to daily reports.
UC-93	Reopen report	Manager	Enables reopening of daily reports for further editing or review after they have been closed.
UC-94	Mark as read	Manager	Allow users to mark reports as read

UC-95	Submit feedback	Super Admin, Manager, Secretary, Staff, Team Leader, Supervisor, Guest	Enables users to provide feedback about the cultivation course.
UC-96	View feedback list	Manager	Allows the manager to view all feedback submissions for review and action.
UC-97	View feedback details	Manager	Enables the manager to view detailed information about specific feedback submissions.
UC-98	Delete feedback	Manager	Allows the manager to remove feedback entries from the system if necessary.

Table 16: Usecase description

3. Functional Requirements

Below is how our team defining the use case for the main functionality for the project. Other details are defined in detail in the document: Report3_Software Requirement Specification_v1.0

3.1 System Functional Overview

3.1.1 Screens Flow

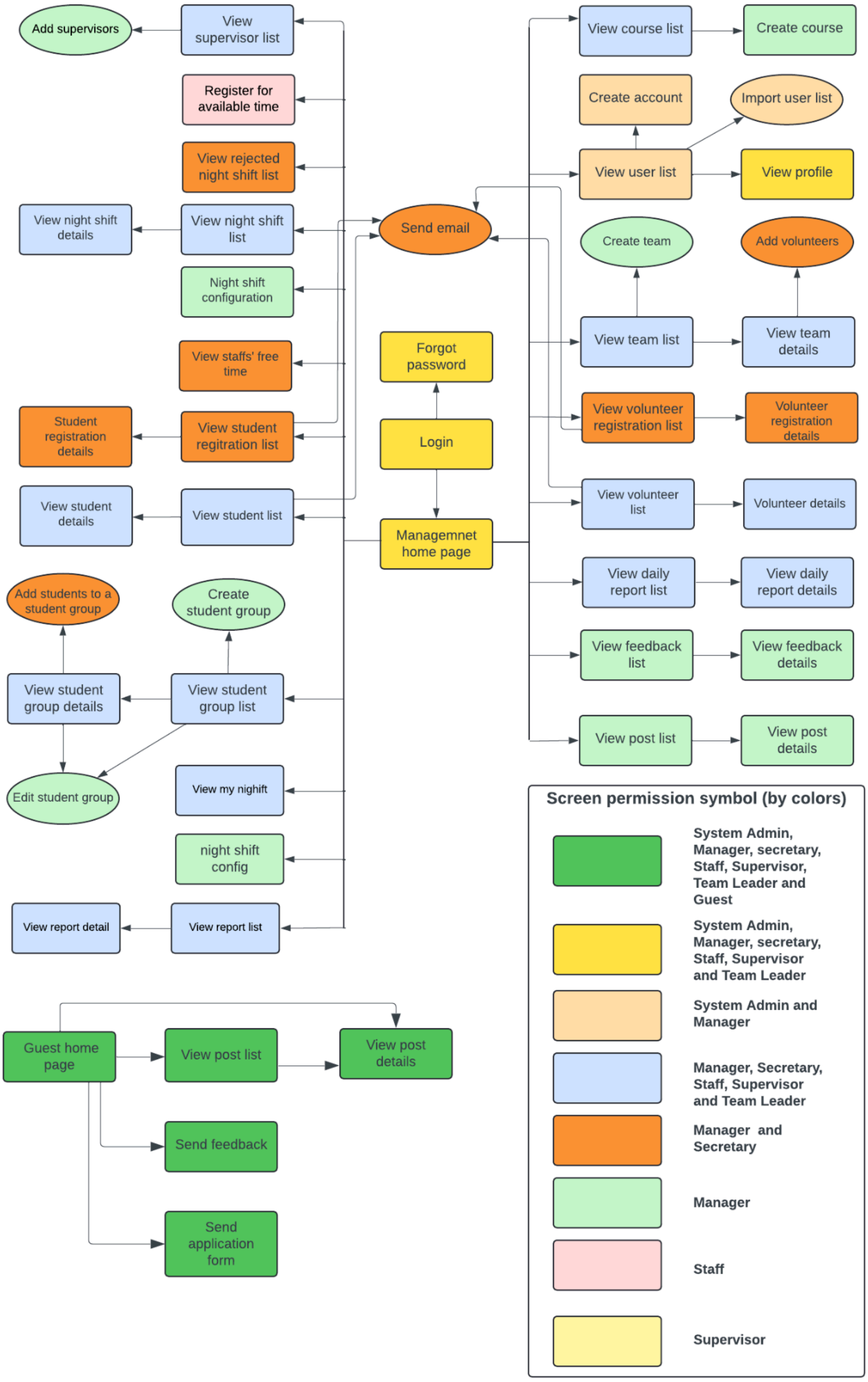


Figure 21: Screens Flow of the system

The link of screens flow: Screens Flow

3.1.2 Screen Descriptions

#	Feature	Screen	Description
1	Course Management	View course list	This screen allows viewing, searching, filtering, and sorting a list of all available courses.
2	Course Management	Create Course	This screen allows creating a new course by filling in a form with suitable data.
3	Account Management	View user list	This screen allows viewing, searching, filtering, and sorting a list of all registered users.
4	Account Management	Create account	This screen allows creating new user accounts by entering the necessary information.
5	Account Management	Import user list	This screen allows importing a list of users for bulk account creation.
6	Account Management	View profile	This screen allows viewing user profile information.
7	Account Management	Forgot password	This screen allows recovering the password by providing account details.
8	Account Management	Login	This screen allows logging into the system using credentials.
9	Team Management	View team list	This screen allows viewing, searching, filtering, and sorting a list of teams in the system.
10	Team Management	Create team	This screen allows creating a new team.
11	Team Management	View team details	This screen allows viewing detailed information about a specific team.
12	Team Management	Edit team	This screen allows editing the details of an existing team.

13	Team Management	Add volunteers to a team	This screen allows adding volunteers to a specific team.
14	Volunteer Management	View volunteer registration list	This screen allows viewing, searching, filtering, and sorting a list of volunteer registrations.
15	Volunteer Management	Volunteer registration details	This screen allows viewing detailed information about a specific volunteer registration.
16	Volunteer Management	View volunteer list	This screen allows viewing, searching, filtering, and sorting a list of volunteers.
18	Volunteer Management	Volunteer details	This screen allows viewing detailed information about a volunteer.
20	Report Management	View report list	This screen allows viewing, searching, filtering, and sorting a list of daily reports submitted for the cultivation course.
21	Report Management	View report details	This screen allows viewing detailed information about a specific daily report.
22	Supervisor Management	View supervisor list	This screen allows viewing, searching, filtering, and sorting a list of all supervisors in the system.
23	Supervisor Management	Add supervisors to a course	This screen allows adding supervisors to a specific course.
24	Night Shift Management	View night shift list	This screen allows viewing, searching, filtering, and sorting the list of night shifts assigned to volunteers or supervisors.
25	Night Shift Management	Register for available time	This screen allows registering for available night shift slots.
26	Night Shift Management	View night shift details	This screen allows viewing detailed information about a specific night shift.
27	Night Shift Management	View staffs' freetime	This screens allows viewing the registered free time of staffs

28	Night Shift Management	View list of night shift rejection	This screen allows viewing a list of rejected night shift assignments.
29	Night Shift Management	View my night shift assignment	This screens allow staff view list all their night shift assignment
30	Night Shift Management	Night Shift Config	This screens allow manager setup room, night shift for specific course
31	Student Management	View student registration list	This screen allows viewing, searching, filtering, and sorting a list of all student registrations.
32	Student Management	Student registration details	This screen allows viewing detailed information about a specific student registration.
33	Student Management	View student details	This screen allows viewing detailed information about a student.
34	Student Management	View student list	This screen allows viewing, searching, filtering, and sorting a list of all students.
35	Student Management	View student result	This screen allows guest viewing, searching feedback of students in course.
36	Student Group Management	View student group list	This screen allows viewing, searching, filtering, and sorting a list of student groups in the system.
37	Student Group Management	Create student group	This screen allows creating a new student group.
38	Student Group Management	View student group details	This screen allows viewing detailed information about a specific student group.
39	Student Group Management	Add students to a student group	This screen allows adding students to a specific group.
40	Student Group Management	Edit student group	This screen allows editing the details of a specific student group.

41	Feedback Management	View feedback list	This screen allows viewing, searching, filtering, and sorting a list of feedback received from participants.
42	Feedback Management	View feedback details	This screen allows viewing detailed feedback from participants.
43	Feedback Management	Send feedback	This screen allows participants to submit feedback.
44	Post Management	View post list	This screen allows viewing, searching, filtering, and sorting a list of posts published on the system page.
45	Post Management	View post details	This screen allows viewing detailed information about a specific post.
46	Post Management	Guest home page	This screen allows viewing the home page of the system.
47	Post Management	View post list (Guest's page)	This screen allows viewing, searching, and filtering posts published on the system.
48	Post Management	View post details (Guest's page)	This screen allows viewing detailed information about a specific post.

Table 17: Screen description

3.1.3 Screen Authorization

Screen	Super Admin	Manager	Secretary	Team Leader	Staff	Supervisor	Guest
View course list		x	x	x	x	x	
Create Course		x					
View user list	x	x					
Create account	x	x					
Import user list	x	x					
View profile		x	x	x	x	x	

Forgot password	x	x	x	x	x	x	
Login	x	x	x	x	x	x	
View team list		x	x	x	x	x	
Create team		x					
View team details		x	x	x	x	x	
Edit team		x					
Add volunteers to a team		x	x				
View volunteer registration list		x	x				
Volunteer registration details		x	x				
View volunteer list		x	x	x	x	x	
Volunteer details		x	x	x	x	x	
View daily report list		x	x	x	x	x	
View daily report details		x	x	x	x	x	
View supervisor list							
Add supervisors to a course		x					
View night shift list		x	x	x	x	x	

Register for available time					x		
View night shift details		x	x	x	x	x	
View staffs' free time		x					
View list of night shift rejection		x	x				
Night shift setup		x					
View student registration list		x	x				
Student registration details		x	x				
Edit student		x	x				
View student details		x	x	x	x	x	
View student group list		x	x	x	x	x	
Create student group		x					
View student group details		x	x	x	x	x	
Add students to a student group		x	x				
Edit student group		x					
View feedback list		x					

View feedback details		x					
Send feedback	x	x	x	x	x	x	x
Send email		x	x				
View post list		x					
Send email		x	x				
Edit post		x					
View post details		x					
Guest home page	x	x	x	x	x	x	x
View post list (Guest's page)	x	x	x	x	x	x	x
View post details (Guest's page)	x	x	x	x	x	x	x
Send application	x	x	x	x	x	x	x

Table 18: Screen Authorization

3.1.4 Non-Screen Functions

#	Feature	System Function	Description
1	Report Management	Close daily report	Every 0h every day, the system will review which reports have not been submitted and change the status of that report to late.

2	Report Management	Create daily report	Every 0h every day, the system will generate reports for that day for each department and shift with the report content being the list of students in the corresponding departments and rooms.
3	Account Management	Authorization	Controls user access and actions within the system based on their roles or groups. This function ensures that each user can only perform actions they are permitted to, enhancing security and system management efficiency.
4	Account Management	Send account email	Send emails to users to inform about account information (include username and password)
5	Account Management	Send OTP email	The system sends OTP code through email. User input that OTP in forgot password screen.
6	Account Management	Remove OTP	Every 5 minutes the system will run to see which OPT code has expired and will delete that OPT code.

Table 19: Non-screen functions

3.1.5 Entity Relationship Diagram

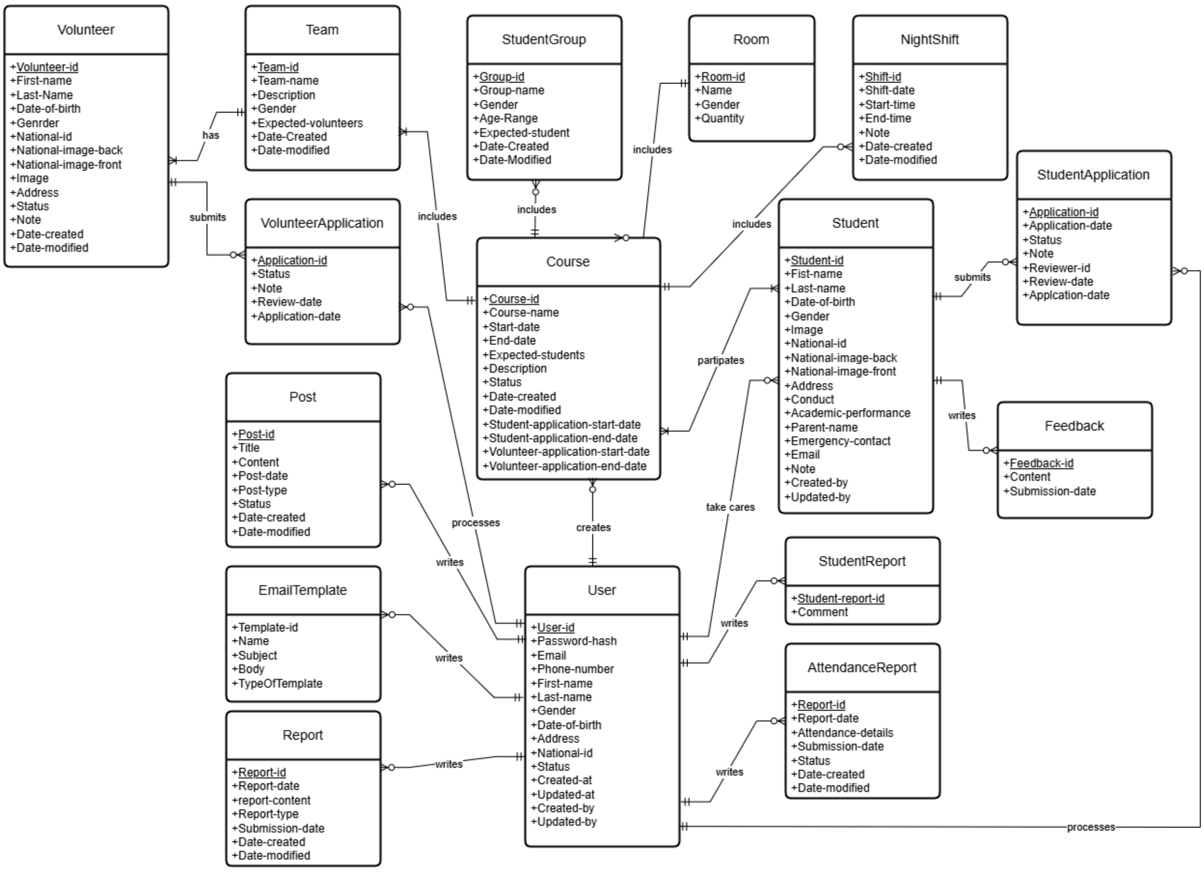


Figure 22: Entity Relationship diagram

Link of the entity relationship diagram: [Entity relationship diagram]

Entities Description

#	Entity	Description
1	User	The users of the system include admin, manager, secretary, staff, supervisor and volunteer.
2	Student	The students attended the cultivation course.
3	Volunteer	The volunteer attended the course.
4	Course	The cultivation course is a pagoda program organized by the pagoda, students participate voluntarily.
5	StudentApplication	The application form that parents submit when they want their children to join the cultivation course

6	Volunteer Application	Application form when someone wants to apply to volunteer
7	Team	A group of volunteers, taking on a certain task during the cultivation course(cooking, cleaning, ...), managed by a staff
8	StudentGroup	In a cultivation course, students will be divided into groups and managed by many staff members
9	NightShift	Every night there will be night shifts to watch the children sleep. The night shift is the temple staff.
10	Report	Report made daily by the supervisor
11	DailyReport	The report is done before the students go to bed and is done by the supervisor
12	StudentReport	Report on student status every day and done by the supervisor
13	Post	Posts are for promotion as well as providing information about cultivation courses and are done by the manager.
14	Feedback	Feedback about the cultivation courses of the pagoda.
14	EmailTemplate	An Email Template is a pre-defined structure or layout for an email that includes standard content and placeholders for dynamic data. These templates can be customised by secretaries and managers.

Table 20: Entity Description

3.2 Account Management

3.2.1 Login

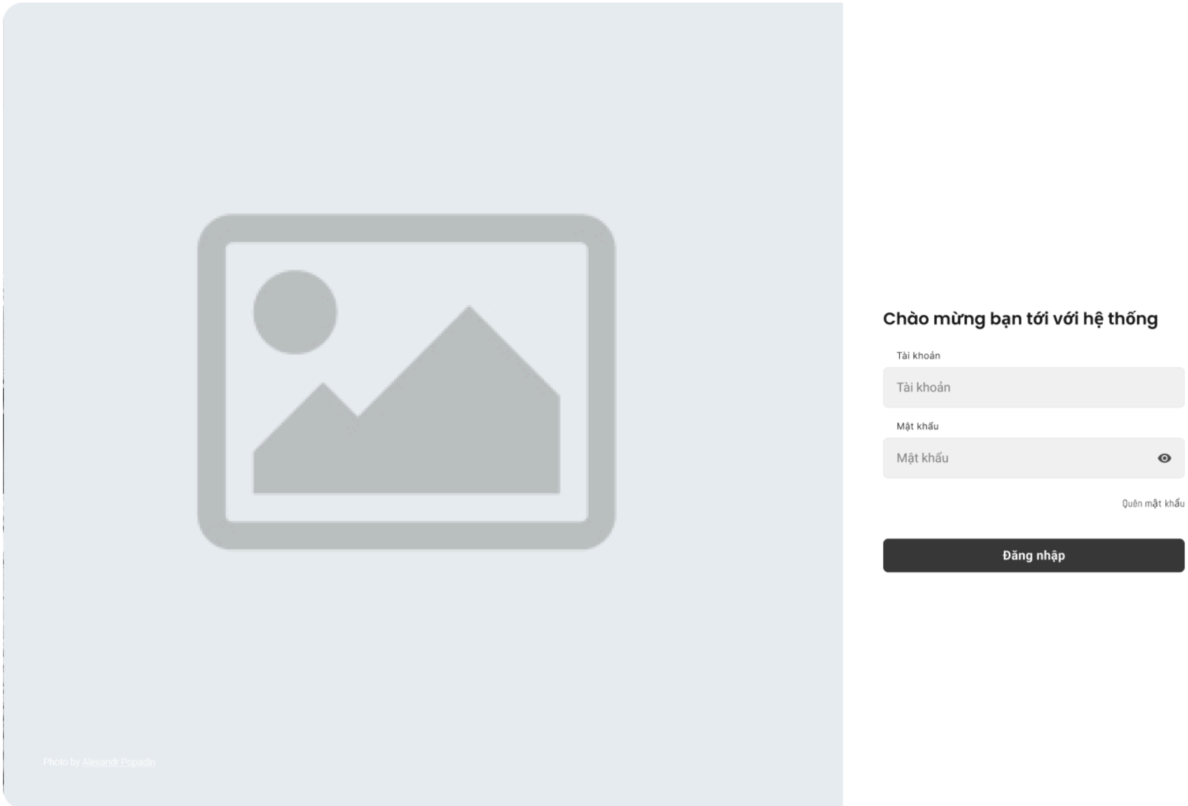


Figure 23. Login screen

UC ID and Name:	**UC-01_Login**		
Created By:	ManhDD	Date Created:	September 25, 2024
Primary Actor:	System Admin, Manager, Staff, Team Leader, Supervisor, Secretary	Secondary Actors:	N/A
Description:	The user enters login credentials to access the system. The system verifies the credentials and grants access to the appropriate system features based on the user's role.		
Trigger:	User clicks the “Đăng nhập” button on the login page		

Preconditions:	PRE-1. The user has a valid and authorised account.
Postconditions:	POST-1. The user is successfully authenticated and logged into the system.
Normal Flow:	**1.0 User Login Process** 1. The user navigates to the Login page. 2. The user enters their username and password. 3. The system verifies the entered credentials against the database. 4. The system checks the user's role and permissions. 5. Upon successful verification, the system redirects the user to the dashboard corresponding to their role.
Alternative Flows:	N/A
Exceptions:	**1.0.E1 System Unavailable** 1. If the username or password is incorrect, the system displays an error message: "Tên người dùng hoặc mật khẩu không đúng." 2. The user is prompted to re-enter the correct credentials or use the "Quên mật khẩu?" feature.
Priority:	High
Frequency of Use:	High
Business Rules:	**BR-12**, **BR-13, BR-14**
Other Information:	The system must securely transmit credentials using SSL/TLS.
Assumptions:	The system redirects to the dashboard of the most recent course

Table 21. Login function description

3.2.2 Logout

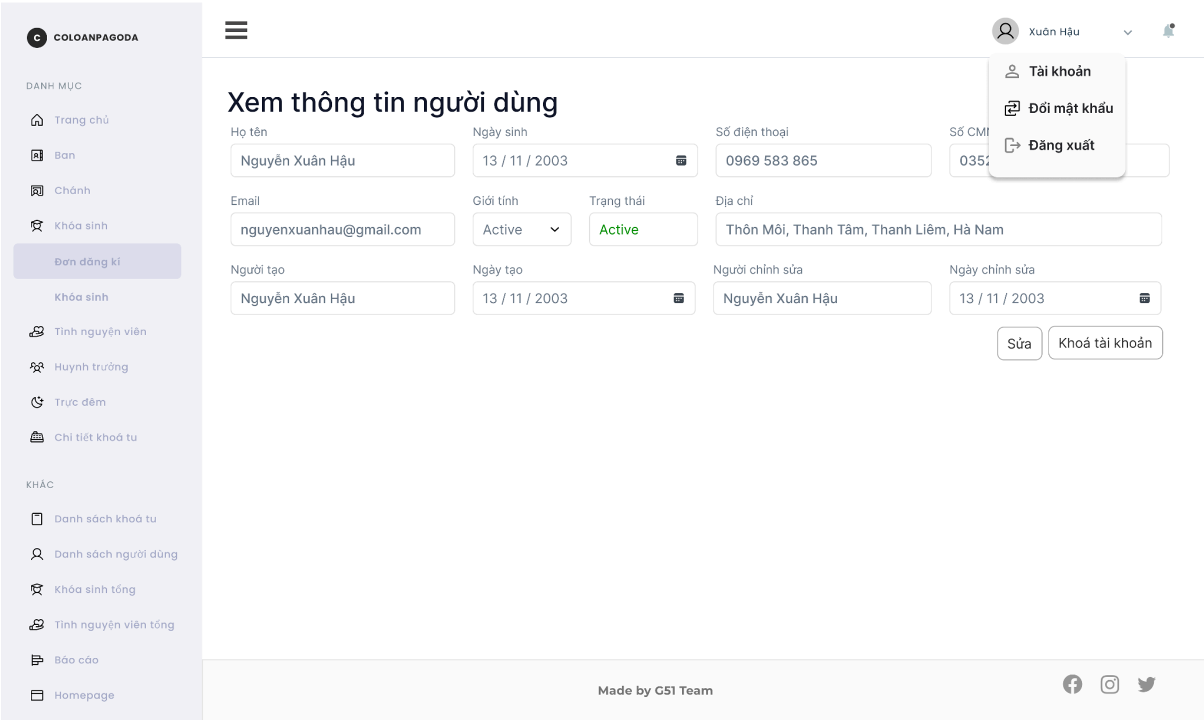


Figure 24. Logout screen

UC ID and Name:	**UC-02_Logout**		
Created By:	ManhDD	Date Created:	September 25, 2024
Primary Actor:	System Admin, Manager, Staff, Team Leader, Supervisor, Secretary	Secondary Actors:	N/A
Description:	The user chooses to log out of the system, so that the session is no longer on the system. The system terminates the session and logs the user out.		
Trigger:	The user clicks the "Đăng xuất" button on the header bar.		
Preconditions:	PRE-1: User must login to the system.		

Postconditions:	POST-1. The user is logged out and redirected to the login page.
Normal Flow:	1. The user hovers over the name of the user in the top right of the screen 2. A small menu is displayed under the name 3. The user clicks on "Đăng xuất" 4. The user is logout from the system
Alternative Flows:	N/A
Exceptions:	N/A
Priority:	Low
Frequency of Use:	Low
Business Rules:	**BR-15**
Other Information:	The system must logout in 1 second.
Assumptions:	Redirect to login screen

Table 22. Logout function description

3.2.3 View profile

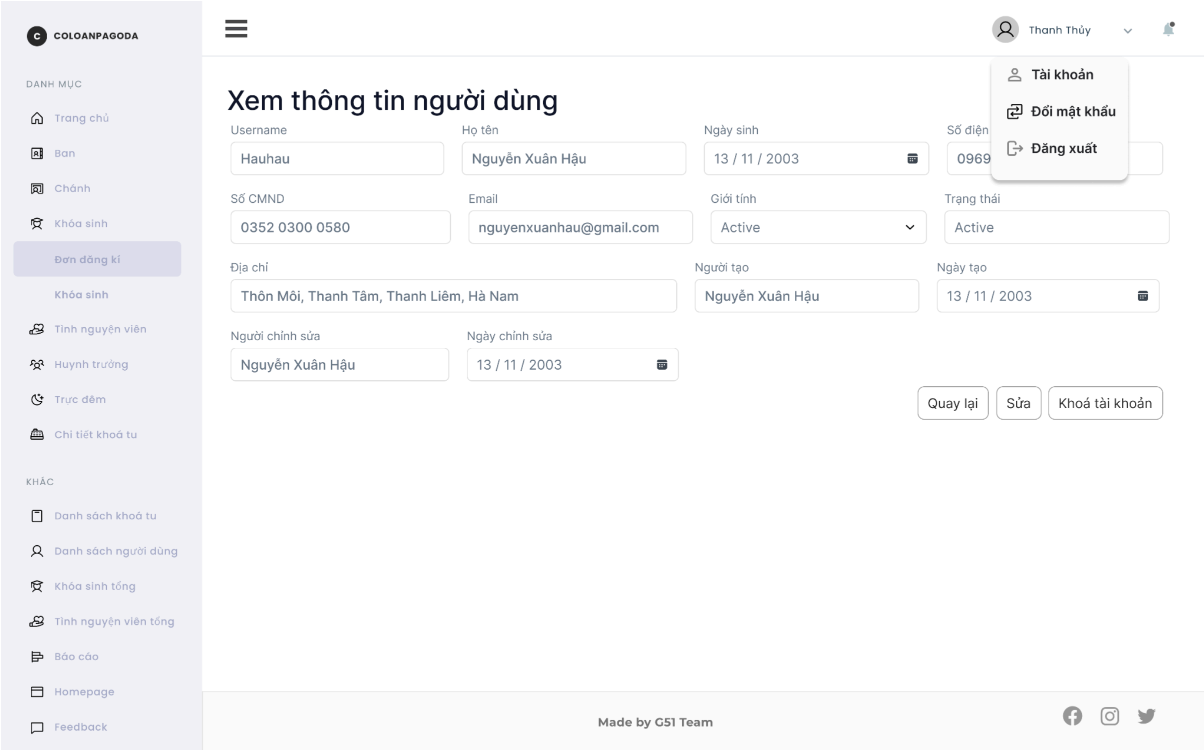


Figure 25. View profile screen

UC ID and Name:	***UC-03_View Profile***		
Created By:	ManhDD	Date Created:	September 25, 2024
Primary Actor:	System Admin, Manager, Staff, Team Leader, Supervisor, Secretary	Secondary Actors:	N/A
Description:	The actor wants to view their personal profile information in the system or other other users in the system to get detailed information from them		
Trigger:	The user wants to view the profile		
Preconditions:	PRE-1: Users must login to the system.		

	PRE-2: The user account has been authorized to perform the function.
Postconditions:	POST-1. The user’s profile information is displayed.
Normal Flow:	**3.0 View Profile by clicking in the name** 1. The user clicks the name of the user in the top right of the screen (see 3.1) 2. The system retrieves the user’s profile information from the database. 3. The system displays the user's profile information, including name, email, role, and other relevant details.
Alternative Flows:	**3.1 View Profile by clicking in the “Tài khoản”** 1. The user hovers over the name of the user in the top right of the screen 2. A small menu is displayed under the name 3. The user clicks on “Tài khoản”
Exceptions:	N/A
Priority:	Medium
Frequency of Use:	Low
Business Rules:	N/A
Other Information:	The time to load the profile is less than 3 seconds.
Assumptions:	All the actors can have the right to view their own profile. The admin can view profile of managers and manger can view profile of staff, team leader, supervisor, secretary

Table 23. View Profile function description

3.3 Course Management

3.3.1 View Course List

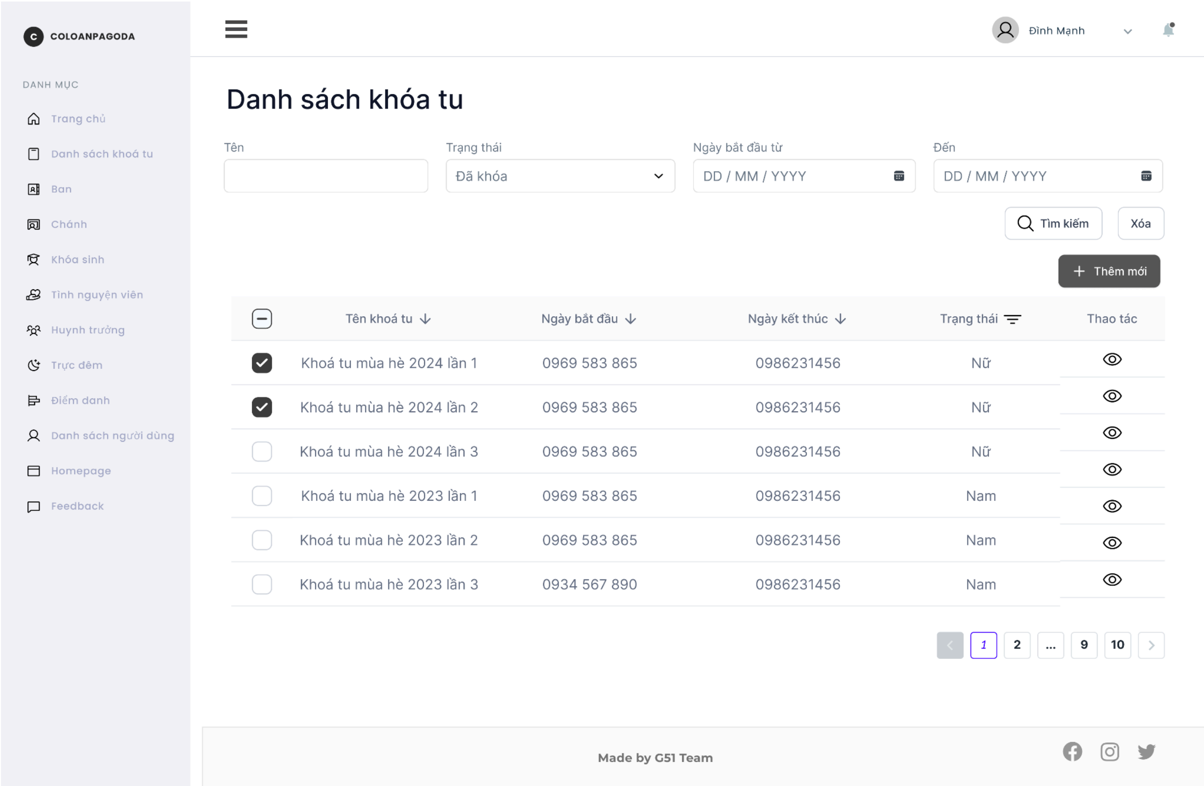


Figure 26. View course list screen

UC ID and Name:	**UC-12_View Course List**		
Created By:	NinhNT	Date Created:	25/Sep/2024
Primary Actor:	Manager, Staff, Team Leader, Supervisor, Secretary	Secondary Actors:	None
Trigger:	User wants to view list all course		
Description:	The actor views a list of courses in the system, including basic information so that the actor can know the courses existing in the		

	system. The actor can also search and filter for the users that match some specific conditions.
Preconditions:	PRE-1 Users must login to the system. PRE-2 The user account has been authorized to perform the function.
Postconditions:	PRE-2 The system displays a list of all available courses.
Normal Flow	**12.0 View course list without any search, filter or sort** 1. The actor logs into the system and navigates to the course management section. 2. The system retrieves and displays a list of all existing courses
Alternative Flows:	**12.1 View course list with search** 1. The actor fills in all necessary information in the search form. 2. The actor clicks "Tìm kiếm" button. 3. The system displays the course list that matches what the actor searched. **12.3 View course list with sort** 1. The actor clicks on the column header and selects the value they want for that column. 2. The system displays the course list that matches what the actor sorted.
Exceptions:	**12.1.E1 There is no matching result with search** 1. The system displays "Không có kết quả phù hợp"
Priority:	Normal
Frequency of Use:	Normal
Business Rules:	**BR-05**
Other Information:	The time to load the screen is less than 3 seconds.

Assumptions:	The list is sort descending by the created day of the course

Table 24. View Course List

3.3.2 Create Course

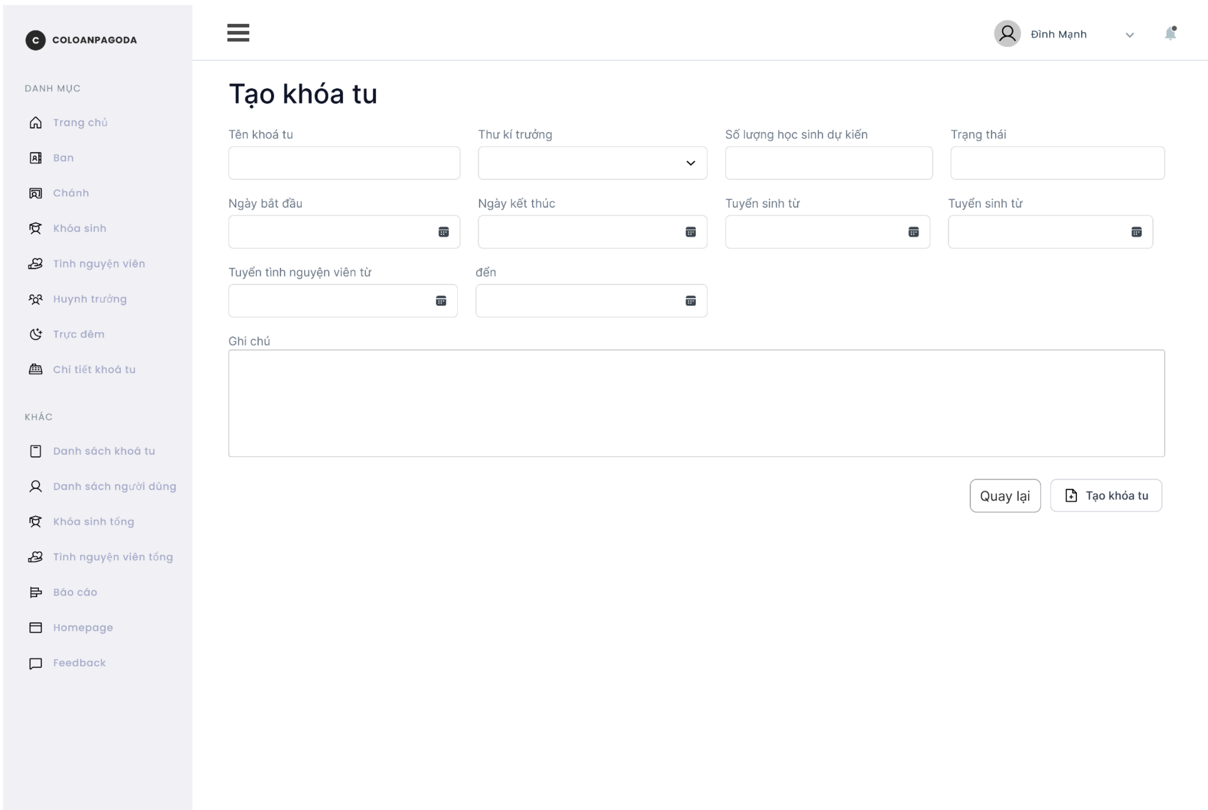


Figure 27. Create course screen

UC ID and Name:	**UC-13_Create Course**		
Created By:	NinhNT	Date Created:	25/Sep/2024
Primary Actor:	Manager	Secondary Actors:	None
Trigger:	Manager decides to create a new course in the system.		

Description:	This use case allows the Manager to create a new course with course's information.
Preconditions:	PRE-1: Users must login to the system. PRE-2: The user account has been authorized to perform the function.
Postconditions:	POST-1. A new course is created and saved in the system.
Normal Flow	**13.0 Create a new coure** 1. The Manager clicks on the "Tạo khóa tu" option on the screen. 2. The system prompts the Manager to enter course details. 3. The Manager confirms the creation of the course. 4. The system creates the new course and saves it to the database.
Alternative Flows:	N/A
Exceptions:	**13.0.E1 The information is invalid** If the entered information is invalid or incomplete, the system should display an alert message and prompt the student to retry.
Priority:	High
Frequency of Use:	Normal
Business Rules:	**BR-01, BR-02, BR-03, BR-05**
Other Information:	The system must respond and display the Create page within 2 seconds.
Assumptions:	After the user creates, the information must be inserted into the database and display a success message within a second, then back to the view course details screen.

Table 25. Create course function description

3.3.3 View Course Details

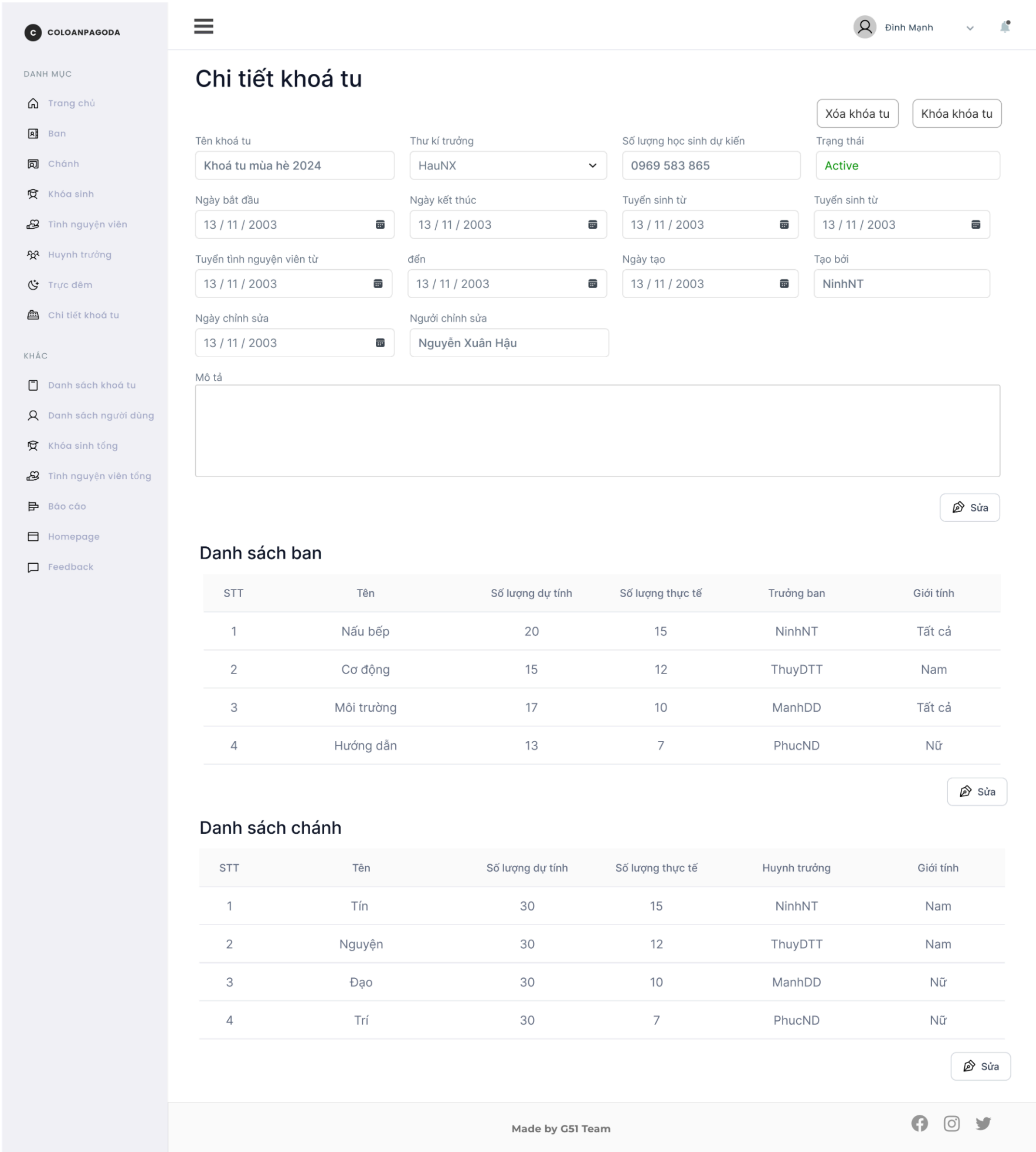


Figure 28. View Course Detail screen

UC ID and Name:	**UC-14_View Course Detail**		
Created By:	NinhNT	Date Created:	25/Sep/2024

Primary Actor:	Manager, Staff, Team Leader, Supervisor, Secretary	Secondary Actors:	None
Trigger:	User want to view course detail information		
Description:	This use case allows users to view the full details of the course, so that the users can know the detailed information of the course		
Preconditions:	PRE-1 User must login to the system. PRE-2 The user account has been authorized to perform the function. PRE-3 The course exists in the system.		
Postconditions:	POST-1. The course details are displayed to the actor.		
Normal Flow	**14.0 View Course Details** 1. The actor navigates to the course management section in the system. 2. The actor selects the "Chi tiết khóa tu" option for a specific course from the course list. 3. The system retrieves all information related to the selected course 4. The system displays the course details on the screen. 5. The actor reviews the course details.		
Alternative Flows:	N/A		
Exceptions:	N/A		
Priority:	Normal		
Frequency of Use:	Normal		
Business Rules:	**BR-05**		

Other Information:	The time to load the profile is less than 3 seconds.
Assumptions:	

Table 26. View course function description

3.3.4 Edit Course

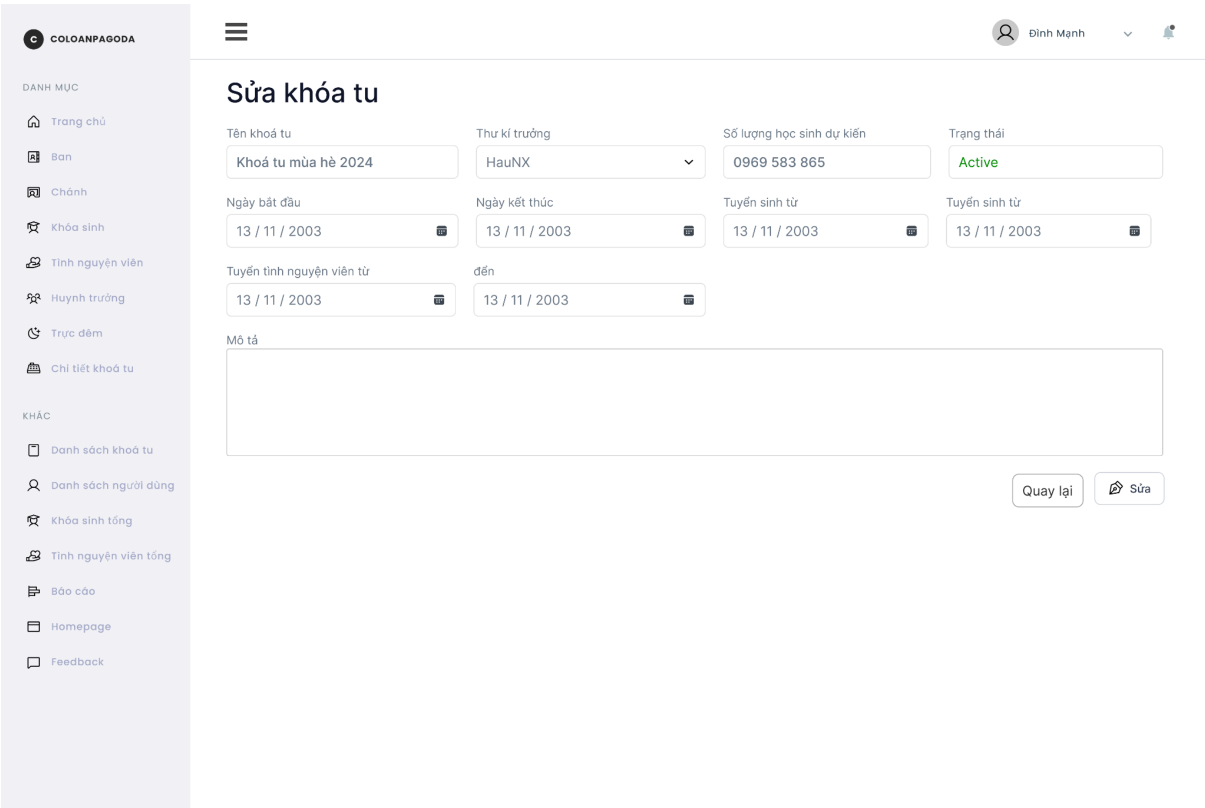


Figure 29. Update course screen

UC ID and Name:	**UC-15_Update Course**		
Created By:	NinhNT	Date Created:	25/Sep/2024
Primary Actor:	Manager	Secondary Actors:	None
Trigger:	Manager want to update course detail information		

Description:	This use case allows user to update the information of a specific course so that the user can edit when the course information change
Preconditions:	PRE-1: User must login to the system. PRE-2: The user account has been authorized to perform the function.
Postconditions:	POST-1. The course details are updated in the system.
Normal Flow	**15.0 Update course** 1. The Manager navigates to the course management section in the system. 2. The Manager selects the edit icon option for a specific course. 3. The system retrieves the current course details and displays them for editing. 4. The Manager modifies the desired course information 5. The Manager confirms the updates. (see 15.0.E1) 6. The system saves the updated course details to the database.
Alternative Flows:	**15.1. Update course after view course details** 1. The manager views course details 2. The manager clicks on the edit button 3. Return to the step 3 on the normal flow
Exceptions:	**15.0.E1 The information is invalid** 1. If the entered information is invalid or incomplete, the system should display an alert message and prompt the student to retry.
Priority:	Normal
Frequency of Use:	Normal
Business Rules:	**BR-01, BR-02, BR-03, BR-04, BR-05**

Other Information:	The system must respond and display the edit page within 2 seconds.
Assumptions:	After the user saves the information must be updated into the database and display a success message within a second, then back to the view course details screen.

Table 27. View course function description

4. Non-Functional Requirements

4.1 External Interfaces

4.1.1 User Interfaces

- All texts in the system are displayed in grammatically correct Vietnamese language.
- Whenever the user/admin performs an action that requires connection to the server, a loading indicator should appear so that the user knows what's going on and doesn't misunderstand that the service is down.
- There should be a clear alert when the app encounters a server error (e.g. offline).
- All pages should have a consistent visual theme and typeface.
- Icons and buttons should have tooltips or labels to assist users in understanding their functions.
- Users can copy and paste content using both the mouse and key combinations, enhancing the convenience of operation.
- Notifications displayed on the screen will have an easy-to-see notification frame, with clear content for users to quickly grasp.
- The interface must be compatible with popular browsers such as Chrome, Firefox, Microsoft Edge.
- The UI must be responsive on different size of screen
- The system must ask for confirmation (Y/N) for data deletion operations and bulk operations.
- The entire drop down list must be arranged in A to Z order and ascending numbers, for night shift workers, arranged in order of priority.

- When the user is in a certain function, the position of this function on the navigation bar (sidebar) will be highlighted, making it easy for users to know where they are in the system

4.1.2 Software Interfaces

- The system uses Gmail Service to send mails for users
- The system also uses SignalR to send notifications to users

4.2 Quality Attributes

4.2.1 Usability

- The system has a user-friendly interface. A non-trained new user can understand and use the system proficiently and do the tasks they want with the accuracy of 90% in 2 hours.
- Users provided with a 2-hour training session shall demonstrate proficiency by completing all tasks the user wants.
- The system must have clear labels and buttons All links and buttons must be clearly displayed and easily understood. Users should comprehend their functions within 10 seconds.
- The system uses consistent icons, colors, fonts, and terminology throughout the system to avoid user confusion.

4.2.2 Security

- The system shall require users to authenticate using a unique username and a strong password that meets complexity requirements (minimum 8 characters, including uppercase letters, lowercase letters, numbers, and special characters).
- All the password of the users are encrypted
- When a user requests a password reset, the system shall send a verification code to the user's registered email.
- The system shall validate all user inputs to prevent injection attacks.
- Ensure only authorized users can access the corresponding functions and data. The system shall implement token-based authentication (JSON Web Tokens - JWT) to manage user sessions securely and efficiently.

4.2.3 Performance

- All web pages shall load completely within 5 seconds over a standard broadband connection (10 Mbps) under normal load conditions.

- The system shall retrieve and display requested data (e.g., course lists, user profiles) within 4 seconds after a query is submitted.

5. Requirement Appendix

5.1 Business Rules

ID	Rule Definition
BR-01	The course must have a valid start and end date.
BR-02	The application opening and closing dates must be within the duration of the course.
BR-03	The estimated number of students must be a positive integer.
BR-04	Any date changes must be validated to ensure they are within a reasonable range.
BR-05	Users must log in with valid credentials (username and password) to access the application.
BR-06	A course cannot be deleted once the start date has been reached and there is at least one registration.
BR-07	When the course is locked, the course cannot accept new registrations or report until it is unlocked.
BR-08	Students must have good conduct and good academic performance to be eligible for admission to the cultivation course
BR-09	Students must be between the ages of 8-17 to be admitted to the cultivation course
BR-10	National ID must be 12 digits
BR-11	Phone number must be 10 digits, starting with 03, 03, 07, 08 or 09
BR-12	Passwords must have a minimum length of 8 characters and a maximum of 32 characters, must contain at least 1 uppercase letter, 1 lowercase letter, 1 number and 1 special character.
BR-13	All login credentials must be transmitted securely using SSL/TLS encryption to prevent interception.
BR-14	The system must grant access to features based on the user's role (System Admin, Manager, Staff, Team Leader, Supervisor, Secretary, Guest).

BR-15	When a user logs out, the system must terminate the active session immediately to prevent further access.
BR-16	Password reset links must be securely transmitted via email using SSL/TLS encryption.
BR-17	Input fields must be validated when the user updates profile information, ensuring that the email format is correct and required fields are filled.
BR-18	All passwords must be securely hashed before being saved in the database.
BR-19	Each email must be unique and not associated with any existing account. The system must verify that the email is valid and not already in use.
BR-20	The system must automatically generate a unique username for each new account to ensure no duplication.
BR-21	Only authorized users can view and change user account statuses.
BR-22	The system must display a confirmation dialog before changing a user's status. If confirmed, the status is updated; if canceled, no changes are made.
BR-23	Only authorized managers and Secretary can accept/ reject student registration.
BR-24	Only authorized managers and Secretary can accept/ reject volunteer registration.
BR-25	Only authorized Manager can assign registration for secretary
BR-26	Daily reports must be generated automatically for each class based on the course's start and end dates.
BR-27	A student cannot be marked as "Present" unless the report for that specific day and class exists in the system.
BR-28	Daily reports for a locked course can only be viewed but not edited or deleted.

BR-29	Students' default daily status in a new report is "Absent" until marked otherwise by authorized users.
BR-30	Only Managers or Secretaries can unlock a course to allow modifications to reports.
BR-31	Daily reports cannot be deleted if they are associated with a course that has already ended.
BR-32	The system must allow filters for searching daily reports by multiple criteria (e.g., date, status, student name, course name).
BR-33	Students marked as "Absent" for more than three consecutive days must trigger an alert to the assigned Supervisor.
BR-34	Each daily report must include timestamps indicating when the daily was last updated and by whom.
BR-35	Supervisors can only view daily reports for classes under their supervision.
BR-36	Posts must be sorted by the latest creation date by default.
BR-37	The system must limit the number of posts displayed per page to optimize loading time.
BR-38	Filters and search criteria must persist until reset by the user.
BR-39	All uploaded images or files must comply with system-defined size and format restrictions (e.g., max 5MB, jpg/png).
BR- 40	Posts cannot be deleted if they are referenced by other critical functions (e.g., reports or notifications).
BR-41	Supervisors must be uniquely identifiable by their name and email in the system.
BR-42	Supervisors can only be assigned to a single student group at a time.

BR-43	Supervisors cannot be deleted if they are currently assigned to an active student group.
BR–44	Only managers and secretaries have permissions to add, delete, or assign supervisors.
BR-45	The system must ensure balanced workloads when automatically assigning supervisors to student groups.
BR-46	A supervisor must not be assigned to more groups than the system's maximum group capacity per role.
BR-47	All operations (add, delete, assign) must log changes in the system for audit purposes.
BR-48	The system must display a confirmation dialog before removing a supervisor.
BR-49	Supervisors must receive notifications upon successful assignment to a student group.
BR-50	Auto-assignment logic must prioritize supervisors based on workload balance.
BR-51	Users must log in with valid credentials to access the system.
BR-52	Only Managers or Secretaries are authorized to manage student groups (view, edit, add, delete).
BR-53	Student groups must be associated with a valid course and supervisor.
BR-54	Gender selection is mandatory for creating a new student group.
BR-55	Supervisors can only be assigned to groups matching their authorized gender category.
BR-56	A student group cannot be deleted if it has assigned students or active daily reports.

BR-57	Each group name must be unique within the same course.
BR-58	Input validation must ensure the group name is non-empty and contains no special characters.
BR-59	Pagination must be applied when the number of student groups exceeds the page display limit.
BR-60	Supervisors must be pre-registered in the system to be assigned to student groups.
BR-61	Soft delete must be implemented for removing student groups, preserving their data for auditing.
BR-62	Search and filter options must persist until manually reset by the user.
BR-63	Gender-based filters must automatically exclude groups that do not match the selected criteria.
BR-64	The system must display confirmation dialogs for sensitive actions like deleting or updating groups.
BR-65	All student group operations (add, edit, delete) must be logged for auditing purposes.
BR-66	Only authorized users (Manager, Secretary) can manage teams (view, add, edit, delete).
BR-67	Team names must be unique within the same course.
BR-68	Gender selection is mandatory when creating or editing a team.
BR-65	Team leaders can only be assigned to teams matching their gender category.
BR-69	Teams must be associated with a valid course.

BR-70	A team cannot be deleted if it has assigned volunteers or is part of an active event.
BR-71	Soft delete must be implemented for removing teams, preserving their data for auditing purposes.
BR-72	Input validation must ensure the team name is non-empty and contains no special characters.
BR-73	Pagination must be applied when the number of teams exceeds the page display limit.
BR-74	Search and filter options must persist until manually reset by the user.
BR-75	Gender-based filters must automatically exclude teams that do not match the selected criteria.
BR-76	The system must display confirmation dialogs for sensitive actions like deleting or updating teams.
BR-77	All team operations (add, edit, delete) must be logged for auditing purposes.
BR-78	Feedback must be associated with a valid and existing student record in the system.
BR-79	Feedback can only be submitted for students who have completed the course.
BR-80	Once submitted, feedback cannot be modified by the parent.
BR-81	Feedback must be time stamped for tracking and sorting purposes.
BR-82	Feedback data must be securely transmitted and stored using encryption standards
BR-83	The system must display confirmation messages after feedback submission.

BR-84	Feedback cannot exceed a predefined length (e.g., 1000 characters) for comments to ensure concise data.
BR-85	The feedback form must validate student codes before submission.
BR-86	The feedback form must validate student codes before submission.
BR-87	Only authorized users (Manager, Secretary) can manage rooms.
BR-88	Room names must be unique within the same course.
BR-89	Gender selection is mandatory when creating or editing a room.
BR-90	A room cannot be deleted if it is assigned to active students or night shifts.
BR-91	Input validation must ensure the room name is non-empty and contains no special characters.
BR-92	Room capacity must be a positive integer and within predefined limits.
BR-93	The system must display confirmation dialogs for sensitive actions like deleting or updating rooms.
BR-94	Only authorized users (Manager, Secretary) can manage night shifts (view, add, edit, delete).
BR-95	Night shifts must have unique start and end times within the same course.
BR-96	Night shift assignments must be linked to valid rooms and staff members.
BR-97	Input validation must ensure all fields (e.g., start time, end time, room) are properly filled.
BR-98	Night shifts cannot overlap within the same room.
BR-99	The system must log all changes (add, edit, delete) for auditing purposes.
BR-100	Night shifts can only be edited or deleted if they haven’t started yet.

BR-101	The system must display confirmation dialogs for sensitive actions like deleting or updating shifts
BR-102	A valid reason must be provided for all decline requests.
BR-103	Managers must be notified of all decline requests.
BR-104	Decline requests must include a reason and current status.
BR-105	Managers must assign another staff member when approving a decline request.
BR-106	Filters must operate only on the selected retreat or criteria when viewing registrations.
BR-107	Notifications for decline requests must link directly to the respective request details.
BR-108	Notifications for decline requests must link directly to the respective request details.
BR-109	The shift to be declined must be at least 3 days after the current date.
BR-110	There is only one course which is active at a time.
BR-111	The report must be submitted before being marked as read.
BR-112	Student's result can only be displayed in the course course which is opened.
BR-113	The volunteers are divided according genders.
BR-114	Only team leader is divided according to genders.
BR-115	When a team is deleted, the volunteers in that team is no longer in any team.
BR-116	Volunteer can only register with the course during the application period.
BR-117	Only people who is over 18 can apply to be volunteer
BR-118	The list of the posts must be display by date

Table 28: Business Rules

5.2 Application Messages List

#	Message code	Message Type	Content
1	MSG_01	Notification	*Tên khóa tu là bắt buộc*
2	MSG_02	Notification	*Trường này là bắt buộc*
3	MSG_03	Notification	*Định dạng ngày không hợp lệ*
4	MSG_04	Notification	*Ngày không hợp lệ*
5	MSG_05	Notification	*Giá trị không hợp lệ*
6	MSG_06	Notification	*Số lượng học viên dự kiến phải lớn hơn 0*
7	MSG_07	Notification	*Ngày bắt đầu phải lớn hơn ngày hiện tại*
8	MSG_08	Notification	*Ngày kết thúc phải lớn hơn ngày bắt đầu*
9	MSG_09	Notification	*Ngày bắt đầu phải nằm trong khoảng thời gian của khóa tu*
10	MSG_10	Notification	*Ngày kết thúc phải nằm trong khoảng thời gian của khóa tu*
11	MSG_11	Notification	*Ngày bắt đầu đăng ký tình nguyện viên phải trước ngày kết thúc*
12	MSG_12	Notification	*Ngày bắt đầu đăng ký khóa sinh phải trước ngày kết thúc*
13	MSG_13	Notification	*Ngày bắt đầu phải sau ngày bắt đầu đăng ký*
14	MSG_14	Notification	*Ngày bắt đầu phải trước ngày kết thúc*
15	MSG_15	Notification	*Ngày bắt đầu đăng ký tình nguyện viên phải từ hôm nay trở đi*
16	MSG_16	Notification	*Ngày bắt đầu đăng ký khóa sinh phải từ hôm nay trở đi*
17	MSG_17	Notification	*Không có dữ liệu để hiển thị*

18	MSG_18	Notification	*Họ và tên là bắt buộc*
19	MSG_19	Notification	*Họ và tên không được vượt quá 100 ký tự*
20	MSG_20	Notification	*Họ và tên chỉ được chứa chữ cái và khoảng trắng*
21	MSG_21	Notification	*Tên cha/mẹ là bắt buộc*
22	MSG_22	Notification	*Tên cha/mẹ không được vượt quá 100 ký tự*
23	MSG_23	Notification	*Tên cha/mẹ chỉ được chứa chữ cái, dấu tiếng Việt và khoảng trắng*
24	MSG_24	Notification	*Ngày sinh là bắt buộc*
25	MSG_25	Notification	*Chỉ khóa sinh từ 8 đến 17 tuổi mới được tham gia*
26	MSG_26	Notification	*Giới tính là bắt buộc*
27	MSG_27	Notification	*Số CMND/CCCD là bắt buộc*
28	MSG_28	Notification	*Số CMND/CCCD phải có độ dài từ 9 đến 12 số và chỉ chứa ký tự số*
29	MSG_29	Notification	*Số điện thoại là bắt buộc*
30	MSG_30	Notification	*Số điện thoại phải là 10 số*
31	MSG_31	Notification	*Email không hợp lệ*
32	MSG_32	Notification	*Địa chỉ là bắt buộc*
33	MSG_33	Notification	*Học lực là bắt buộc*
34	MSG_34	Notification	*Hạnh kiểm là bắt buộc*
35	MSG_35	Notification	*Ảnh thẻ là bắt buộc*
36	MSG_36	Notification	*Kích thước ảnh tối đa là 5MB*
37	MSG_37	Notification	*Chỉ chấp nhận các định dạng .jpg, .jpeg, .png*
38	MSG_38	Notification	*Ảnh mặt trước CCCD là bắt buộc*
39	MSG_39	Notification	*Ảnh mặt sau CCCD là bắt buộc*

40	MSG_40	Notification	*Thêm nhân viên thành công!*
41	MSG_41	Notification	*Xóa ca trực thành công!*
42	MSG_42	Notification	*Cập nhật nhân viên thành công!*
43	MSG_43	Notification	*Cập nhật khóa tu thành công*
44	MSG_44	Notification	*Cập nhật trạng thái thành công*
45	MSG_45	Notification	*Kết thúc khóa tu*
46	MSG_46	Notification	*Xóa thành công*
47	MSG_47	Notification	*Cập nhật thông tin bài đăng thành công!*
48	MSG_48	Notification	*Cập nhật thông tin khóa học thành công*
49	MSG_49	Notification	*Cập nhật thông tin sinh viên thành công*
50	MSG_50	Notification	*Cập nhật thư ký thành công*
51	MSG_51	Notification	*Cập nhật nhóm thành công*
52	MSG_52	Notification	*Nộp báo cáo thành công!*
53	MSG_53	Notification	*Đã mở lại báo cáo*
54	MSG_54	Notification	*Yêu cầu mở lại đã được gửi thành công!*
55	MSG_55	Notification	*Thêm tình nguyện viên vào ban thành công!*
56	MSG_56	Notification	*Xóa tình nguyện viên thành công!*
57	MSG_57	Notification	*Cập nhật thông tin người dùng thành công!*
58	MSG_58	Notification	*Cập nhật thư ký thành công*
59	MSG_59	Notification	*Cập nhật ban thành công*
60	MSG_60	Notification	*Đổi mật khẩu thành công*
61	MSG_61	Notification	*OTP đã được gửi đến email của bạn.*
62	MSG_62	Notification	*OTP xác minh thành công.*
63	MSG_63	Notification	*Có lỗi xảy ra*

64	MSG_64	Notification	*Cập nhật trạng thái thất bại*
65	MSG_65	Notification	*Cập nhật thư ký thất bại*
66	MSG_66	Notification	*Đã có lỗi xảy ra khi cập nhật thông tin.*
67	MSG_67	Notification	*OTP không hợp lệ hoặc đã hết hạn.*
68	MSG_68	Notification	*Không thể gửi OTP.*
69	MSG_69	Notification	*Xóa phản hồi thất bại*
70	MSG_70	Notification	*Đã xảy ra lỗi khi xóa bài đăng*
71	MSG_71	Notification	*Vui lòng chọn ít nhất một khoảng thời gian để cập nhật.*
72	MSG_72	Notification	*Tạo bài đăng thất bại!*
73	MSG_73	Notification	*Đã xảy ra lỗi khi tạo bài đăng*
74	MSG_74	Notification	*Vui lòng chọn khóa sinh để in thẻ*
75	MSG_75	Notification	*Không thể tải file mẫu.*
76	MSG_76	Notification	*Đã xảy ra lỗi khi gửi email*
77	MSG_77	Notification	*Duyệt đơn thất bại*
78	MSG_78	Notification	*Không thể xóa chánh của khóa tu đã kết thúc.*

Table 29: Application Message

IV. Software Design Description

Below is how our team defined the main content for the report 4. Other details are defined in detail in the document: Report4_Software Design Document_v1.0

1. System Design

1.1 System Architecture

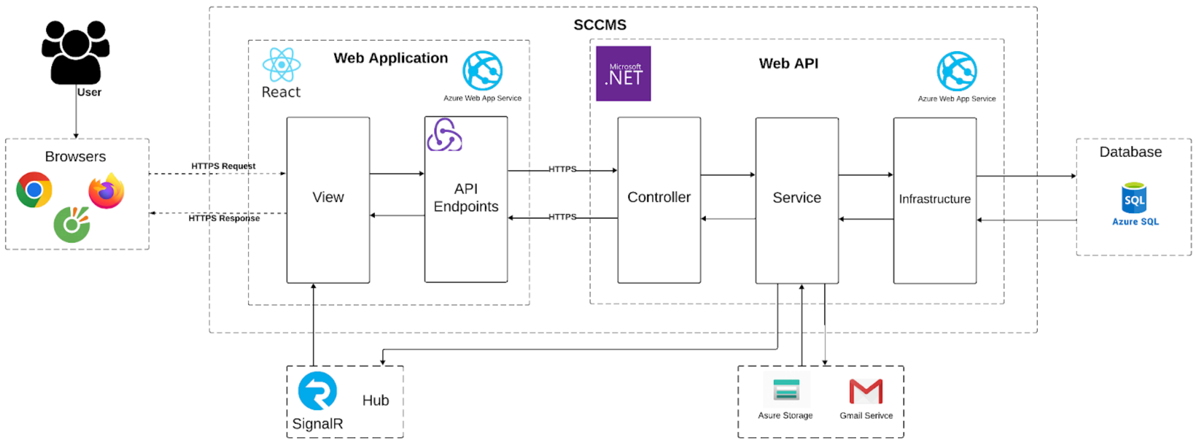


Figure 30: System architecture

The link of image: [System architecture]

The system architecture depicted in the diagram represents a comprehensive web application structure with secure, high-performance communication through HTTPS. Below is a detailed description of the system:

1. **Browers:** Users can use most browsers to access our website such as: Chrome, Firefox, Microsoft Edge
2. **Web Application (React):** The front-end application is built using React, a popular JavaScript library. React handles the user interface and sends HTTP/HTTPS requests to the Web API to retrieve data and perform various operations.
3. **Redux:** Redux is a state management library used with React to manage the application's state in a single store. It simplifies state sharing across components and enables predictable state updates through actions and reducers. Dispatching an action triggers state changes, making it easier to track and debug complex app behaviors.
4. **Web API (.NET):** The backend is developed using the .NET framework, which provides API services for the web application. All requests from React to the backend occur via HTTPS to ensure data security and privacy. The .NET API communicates with several services, including Firebase, SignalR, Azure Storage, and SQL Server.

5. **SignalR**: SignalR is integrated into the system to provide real-time bi-directional communication between the client (browser) and the server. It enables features like live updates, notifications, and collaborative tools. SignalR operates over WebSockets (or other fallback protocols like Server-Sent Events or Long Polling), ensuring efficient and low-latency communication. The integration is secured via HTTPS to maintain data confidentiality and integrity.
6. **Gmail API:** The system integrates with the Gmail API to send emails, such as notifications or email confirmations to users. All interactions between the Web API and the Gmail API are secured through HTTPS, protecting user information.
7. **Azure Services (Azure Storage Account):** The Azure Storage Account is used for storing large files or data, such as images or attachments. The Web API securely communicates with this service to store and retrieve data through HTTPS.
8. **Azure SQL:** Azure SQL is a cloud-based relational database service provided by Microsoft. It offers high availability, security, and scalability, making it ideal for applications requiring a secure and reliable SQL database. Azure SQL is fully managed, meaning Microsoft handles maintenance, updates, and scaling automatically.

9. **Azure Web App Service:** Azure Web App Service is a Platform-as-a-Service (PaaS) that allows easy deployment and management of web applications without handling the infrastructure. It supports multiple programming languages, offers automatic scaling, and manages the server-side setup, allowing developers to focus on the code rather than the underlying infrastructure.

1.2 Package Diagram

1.2.1 Front-end package diagram

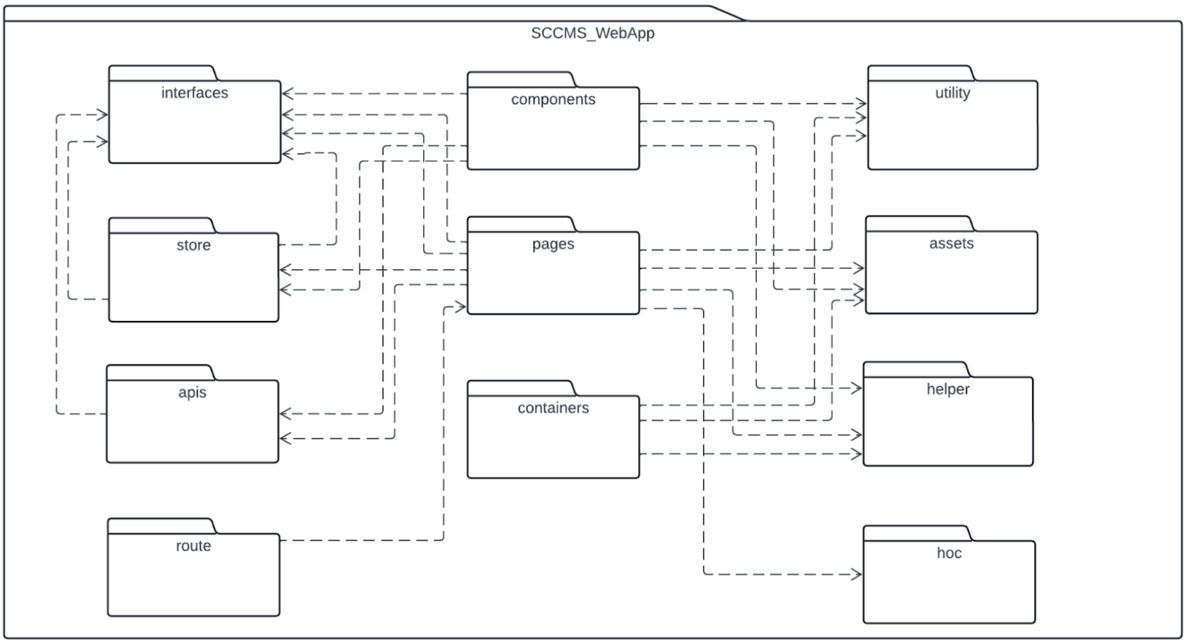


Figure 31: Front-end package diagram

The link of image: [Front-end package diagram]

Package descriptions

No	Package	Description
1	assets	Contains static resources such as images, fonts, and stylesheets. These assets are used across the application to provide styling and visual content.
2	components	This directory likely holds reusable UI components such as buttons, forms, tables, or any other visual elements used throughout the application. These components are typically independent, and they can be combined to build a larger UI.
3	pages	Pages generally represent full-screen views or routes in the application. Each page could be a collection of multiple components and containers, forming a complete section of the application.

4	containers	Contains the overall layout of the project.
5	hoc (Higher-Order Components)	This folder may contain higher-order components, which are React components that wrap other components to add additional functionality, such as authorization handling, data loading, or conditional rendering.
6	route	This package is responsible for defining the routes of the application. It probably maps different URL paths to their corresponding pages or components.
7	store	This is where state management logic resides for the Redux library. It centralizes the application's state, ensuring consistent data flow and state management.
8	helper	Contains helpers used across different parts of the application. These are not tied to any specific feature but provide essential services like formatting, validation, or shared business logic.
9	utility	Contains utility functions, helpers, or common logic used across different parts of the application. These are not tied to any specific feature but provide essential services like formatting, validation, or shared business logic.
10	apis	The APIs folder most likely handles the interaction with the backend services. This package could contain functions or classes for making HTTP requests, handling responses, and managing data fetching from the server.
11	interfaces	This package likely contains TypeScript or other interface definitions that are shared across various parts of the application. These interfaces help ensure consistent data structures throughout the app. The package also contains constant values, configuration settings, or enums in the application

Table 30: Front-end package

1.2.2 Back-end package Diagram

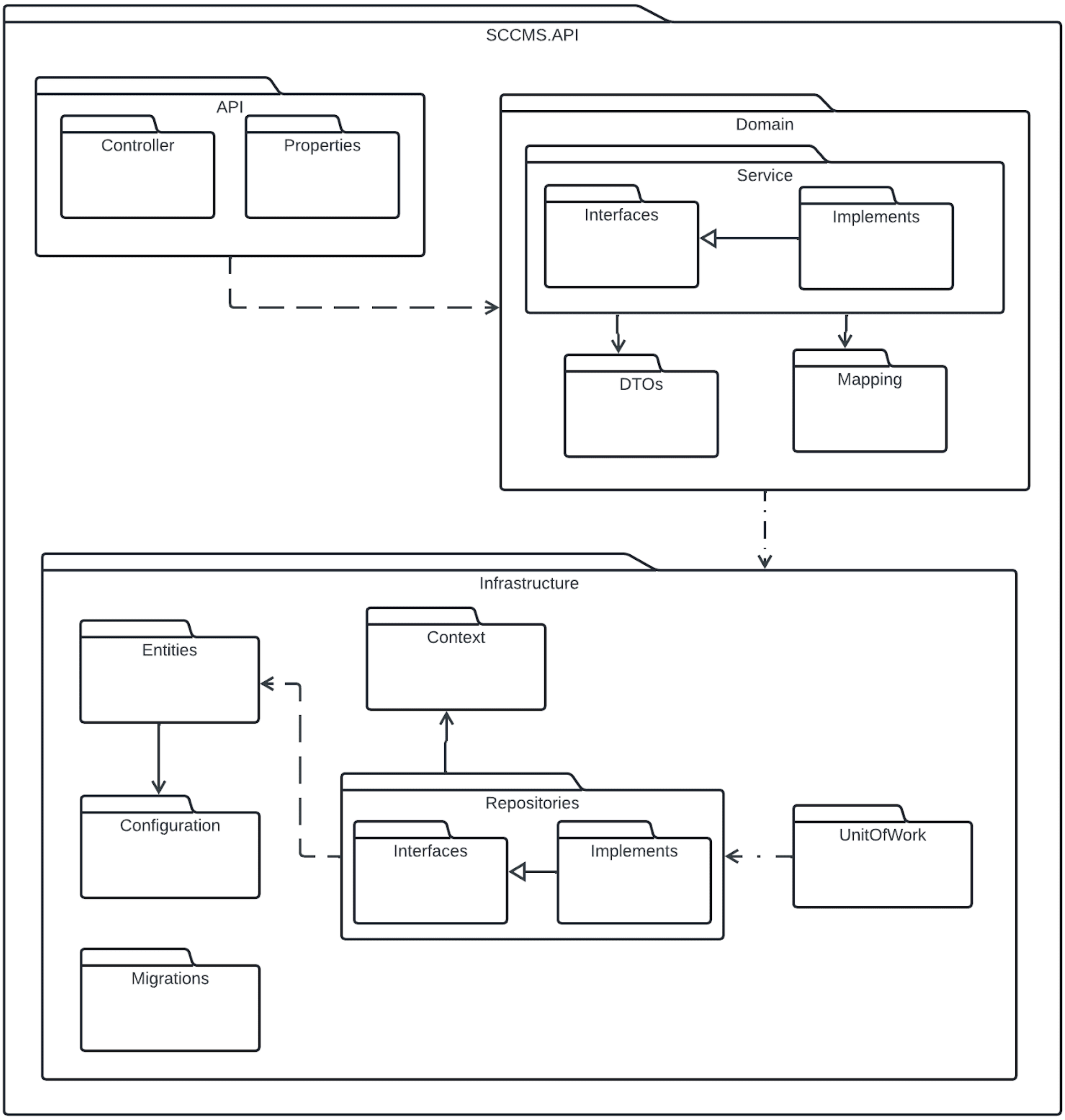


Figure 32: Back-end package diagram
The link of image: Back-end package diagram

Package descriptions

No	Package	Description
1	API	Contains the API layer, with Controller for handling HTTP requests and Properties for application settings or metadata. It serves as the main entry point for external interactions with the system.

2	Controller	Handles incoming HTTP requests, routes them to appropriate services in the Domain layer, and returns HTTP responses.
3	Properties	Contains settings and metadata related to the API layer, such as configurations or constants used in the Controllers.
4	Domain	The core business logic and domain services. It includes Interfaces to define service contracts, Implements for business logic implementations, DTOs for data exchange between layers, and Mapping for converting between entities and DTOs.
5	Interfaces	Defines contracts for services that handle business logic, ensuring a separation between implementation and usage.
6	Implements	Contains concrete implementations of the interfaces where the actual business logic and service methods are defined.
7	DTOs	Data Transfer Objects used to exchange data between layers, particularly between the API and Domain layers, to avoid exposing entities directly.
8	Mapping	Handles mapping logic between models and DTOs, ensuring smooth conversion between data structures.
9	Infrastructure	Manages data access logic and related infrastructure. It includes Entities for database table mapping, Context for interacting with the database, Data for managing data services, Repositories for encapsulating data access operations, UnitOfWork for transaction management, and Migrations for managing database schema changes.
10	Entities	Represents domain models that map directly to database tables, encapsulating the core data of the application.
11	Context	Represents the database context, use Entity Framework to interact with the database.
12	Repositories	Interfaces: Define contracts for data access, including common methods like Get, Add, Update, and Delete. Implements: Concrete implementations of these interfaces, containing the logic for data access and manipulation.

13	UnitOfWork	Manages transactions by coordinating between multiple repositories, ensuring that multiple changes can be committed as a single unit.
14	Configuration	Contains configurations related to infrastructure components, such as database connection settings or service parameters.
15	Migrations	Manages database schema changes through migrations, ensuring the database schema is kept in sync with changes in entity models.

Table 31: Back-end package

2. Database Design

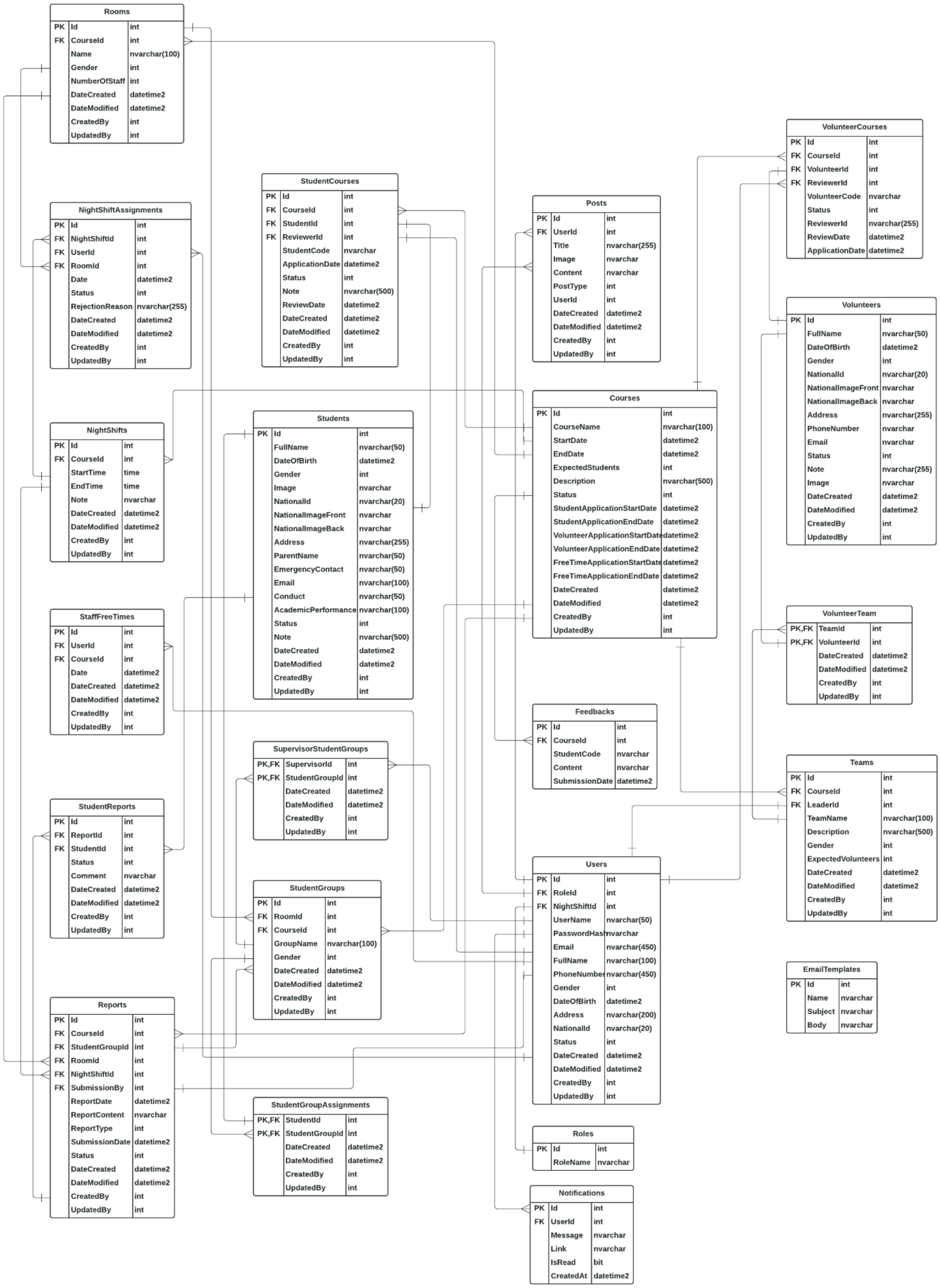


Figure 33: Database design

The link of image: Database *design*

2.1 Courses

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	CourseName	nvarchar(100)		x		Max Length: 100
03	StartDate	datetime2		x		
04	EndDate	datetime2		x		
05	ExpectedStudents	int		x		
06	Description	nvarchar(500)				Max Length: 500
07	Status	int		x		Max Length: 20
08	StudentApplicationStartDate	datetime2		x		
09	StudentApplicationEndDate	datetime2		x		
10	VolunteerApplicationStartDate	datetime2		x		
11	VolunteerApplicationEndDate	datetime2		x		
12	FreeTimeApplicationStartDate	datetime2				Nullable
13	FreeTimeApplicationEndDate	datetime2				Nullable
14	DateCreated	datetime2		x		
15	DateModified	datetime2		x		
16	CreatedBy	int		x		
17	UpdatedBy	int		x		

Table 32: Courses Table

2.2 Email Template

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	Name	nvarchar(max)		x		
03	Subject	nvarchar(max)		x		
04	Body	nvarchar(max)		x		

Table 33: Email Templates Table

2.3 Roles

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	RoleName	nvarchar(max)		x		

Table 34: Roles Table

2.4 Students

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	FullName	nvarchar(50)		x		Max Length: 50
03	DateOfBirth	datetime2		x		
04	Gender	int		x		
05	Image	nvarchar(max)		x		
06	NationalId	nvarchar(20)		x		Max Length: 20
07	NationalImageFront	nvarchar(max)		x		

08	NationalImageBack	nvarchar(max)		x		
09	Address	nvarchar(255)		x		Max Length: 255
10	ParentName	nvarchar(50)		x		Max Length: 50
11	EmergencyContact	nvarchar(50)		x		Max Length: 50
12	Email	nvarchar(100)		x		Max Length: 100
13	Conduct	nvarchar(50)				Max Length: 50 (Nullable)
14	AcademicPerformance	nvarchar(100)				Max Length: 100 (Nullable)
15	Status	int		x		
16	Note	nvarchar(500)				Max Length: 500 (Nullable)
17	DateCreated	datetime2		x		
18	DateModified	datetime2		x		
19	CreatedBy	int		x		
20	UpdatedBy	int		x		

Table 35: Students Table

2.5 Volunteers

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	FullName	nvarchar(50)		x		Max Length: 50
03	DateOfBirth	datetime2				Nullable
04	Gender	int		x		
05	NationalId	nvarchar(20)		x		Max Length: 20
06	NationalImageFront	nvarchar(max)		x		

07	NationalImageBack	nvarchar(max)		x		
08	Address	nvarchar(255)		x		Max Length: 255
09	PhoneNumber	nvarchar(max)				Nullable
10	Email	nvarchar(max)				Nullable
11	Status	int		x		
12	Note	nvarchar(255)				Max Length: 255 (Nullable)
13	Image	nvarchar(max)		x		
14	DateCreated	datetime2		x		
15	DateModified	datetime2		x		
16	CreatedBy	int		x		
17	UpdatedBy	int		x		

Table 36: Volunteers Table

2.6 Feedbacks

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	StudentCode	nvarchar(max)		x		
03	CourseId	int		x	FK	FK to Courses(Id)
04	Content	nvarchar(max)		x		
05	SubmissionDate	datetime2		x		

Table 37: Feedbacks Table

2.7 Night Shifts

No	Field Name	Type	Unique	Not Null	PK/ FK	Note

01	Id	int	x	x	PK	Identity (1,1)
02	CourseId	int		x	FK	FK to Courses(Id)
03	StartTime	time		x		
04	EndTime	time		x		
05	Note	nvarchar(max)				Nullable
06	DateCreated	datetime2		x		
07	DateModified	datetime2		x		
08	CreatedBy	int		x		
09	UpdatedBy	int		x		

Table 38: Night Shifts Table

2.8 Rooms

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	CourseId	int		x	FK	FK to Courses(Id)
03	Name	nvarchar(100)		x		Max Length: 100
04	Gender	int		x		
05	NumberOfStaff	int		x		
06	DateCreated	datetime2		x		
07	DateModified	datetime2		x		
08	CreatedBy	int		x		
09	UpdatedBy	int		x		

Table 39: Rooms Table

2.9 Users

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	UserName	nvarchar(50)	x	x		Max Length: 50
03	PasswordHash	nvarchar(max)		x		
04	Email	nvarchar(450)	x	x		
05	FullName	nvarchar(100)		x		Max Length: 100
06	PhoneNumber	nvarchar(450)	x			Nullable
07	Gender	int				Nullable
08	DateOfBirth	datetime2				Nullable
09	Address	nvarchar(200)				Max Length: 200, Nullable
10	NationalId	nvarchar(20)	x	x		Max Length: 20
11	Status	int				Nullable
12	RoleId	int		x	FK	FK to Roles(Id)
13	NightShiftId	int			FK	FK to NightShifts(Id), Nullable
14	DateCreated	datetime2		x		
15	DateModified	datetime2		x		
16	CreatedBy	int		x		
17	UpdatedBy	int		x		

Table 40: Users Table

2.10 Student Groups

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	CourseId	int		x	FK	FK to Courses(Id)
03	GroupName	nvarchar(100)		x		Max Length: 100
04	Gender	int		x		
05	RoomId	int			FK	FK to Rooms(Id), Nullable
06	DateCreated	datetime2		x		
07	DateModified	datetime2		x		
08	CreatedBy	int		x		
09	UpdatedBy	int		x		

Table 41: Student Groups Table

2.11 Student Courses

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	CourseId	int		x	FK	FK to Courses(Id)
03	StudentId	int		x	FK	FK to Students(Id)
04	StudentCode	nvarchar(max)				Nullable
05	ApplicationDate	datetime2				Nullable
06	Status	int		x		
07	Note	nvarchar(500)				Max Length: 500, Nullable
08	ReviewerId	int			FK	FK to Users(Id), Nullable

09	ReviewDate	datetime2				Nullable
10	DateCreated	datetime2		x		
11	DateModified	datetime2		x		
12	CreatedBy	int		x		
13	UpdatedBy	int		x		

Table 42: Student Courses Table

2.12 Staff Free Times

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	UserId	int		x	FK	FK to Users(Id)
03	CourseId	int		x	FK	FK to Courses(Id)
04	Date	datetime2		x		
05	DateCreated	datetime2		x		
06	DateModified	datetime2		x		
07	CreatedBy	int		x		
08	UpdatedBy	int		x		

Table 43: Staff Free Times Table

2.13 Reports

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	CourseId	int		x	FK	FK to Courses(Id)
03	UserId	int		x	FK	FK to Users(Id)

04	StudentGroupId	int			FK	FK to StudentGroups(Id), Nullable
05	ReportDate	datetime2		x		
06	ReportContent	nvarchar(max)				Nullable
07	ReportType	int		x		Max Length: 50
08	SubmissionDate	datetime2		x		
09	Status	int		x		
10	DateCreated	datetime2		x		
11	DateModified	datetime2		x		
12	CreatedBy	int		x		
13	UpdatedBy	int		x		

Table 44: Reports Table

2.14 Student Group Assignments

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	StudentId	int		x	PK, FK	PK: Composite Key (StudentId, StudentGroupId)
02	StudentGroupId	int		x	FK	FK to StudentGroups(Id)
03	DateCreated	datetime2		x		
04	DateModified	datetime2		x		
05	CreatedBy	int		x		
06	UpdatedBy	int		x		

Table 45: Student Group Assignments Table

2.15 Supervisor Student Groups

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	SupervisorId	int		x	PK, FK	PK: Composite Key (SupervisorId, StudentGroupId)
02	StudentGroupId	int		x	FK	FK to StudentGroups(Id)
03	DateCreated	datetime2		x		
04	DateModified	datetime2		x		
05	CreatedBy	int		x		
06	UpdatedBy	int		x		

Table 46: Supervisor Student Groups Table

2.16 Volunteer Courses

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	CourseId	int		x	FK	FK to Courses(Id)
03	VolunteerId	int		x	FK	FK to Volunteers(Id)
04	VolunteerCode	nvarchar(max)				Nullable
05	Status	int				Max Length: 20, Nullable
06	Note	nvarchar(255)				Max Length: 255, Nullable
07	ReviewerId	int			FK	FK to Users(Id), Nullable
08	ReviewDate	datetime2				Nullable

09	ApplicationDate	datetime2		x		
10	DateCreated	datetime2		x		
11	DateModified	datetime2		x		
12	CreatedBy	int		x		
13	UpdatedBy	int		x		

Table 47: Volunteer Courses Table

2.17 Volunteer Team

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	TeamId	int		x	PK, FK	PK: Composite Key (TeamId, VolunteerId)
02	VolunteerId	int		x	FK	FK to Volunteers(Id)
03	DateCreated	datetime2		x		
04	DateModified	datetime2		x		
05	CreatedBy	int		x		
06	UpdatedBy	int		x		

Table 48: Volunteer Team Table

2.18 Posts

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	Title	nvarchar(255)		x		Max Length: 255
03	Image	nvarchar(max)		x		
04	Content	nvarchar(max)		x		
05	PostType	int		x		

06	Status	int		x		
07	UserId	int			FK	FK to Users(Id), Nullable
08	DateCreated	datetime2		x		
09	DateModified	datetime2		x		
10	CreatedBy	int		x		
11	UpdatedBy	int		x		

Table 49: Post Table

2.19 Student Reports

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	ReportId	int		x	FK	FK to Reports(Id)
03	StudentId	int		x	FK	FK to Students(Id)
04	Status	int		x		
05	Comment	nvarchar(max)				Nullable
06	DateCreated	datetime2		x		
07	DateModified	datetime2		x		
08	CreatedBy	int		x		
09	UpdatedBy	int		x		

Table 50: Student Reports Table

2.20 Night Shift Assignments

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)

02	NightShiftId	int		x	FK	FK to NightShifts(Id)
03	UserId	int			FK	FK to Users(Id), Nullable
04	Date	datetime2		x		
05	RoomId	int			FK	FK to Rooms(Id), Nullable
06	Status	int		x		Max Length: 50
07	RejectionReason	nvarchar(255)				Max Length: 255, Nullable
08	DateCreated	datetime2		x		
09	DateModified	datetime2		x		
10	CreatedBy	int		x		
11	UpdatedBy	int		x		

Table 51: Night Shift Assignments Table

2.21 Notifications

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	UserId	int		x	FK	FK to Users(Id)
03	Message	nvarchar(max)		x		
04	Link	nvarchar(max)		x		
05	IsRead	bit		x		
06	CreatedAt	datetime2		x		

Table 52: Notifications Table

2.22 Teams

No	Field Name	Type	Unique	Not Null	PK/ FK	Note
01	Id	int	x	x	PK	Identity (1,1)
02	CourseId	int			FK	FK to Courses(Id), Nullable
03	LeaderId	int		x	FK	FK to Users(Id)
04	TeamName	nvarchar(100)		x		Max Length: 100
05	Description	nvarchar(500)				Max Length: 500, Nullable
06	Gender	int				Nullable
07	ExpectedVolunteers	int		x		
08	DateCreated	datetime2		x		
09	DateModified	datetime2		x		
10	CreatedBy	int		x		
11	UpdatedBy	int		x		

Table 53: Teams Table

3. Detailed Design

3.1 Class Diagram

3.1.1 Account Management

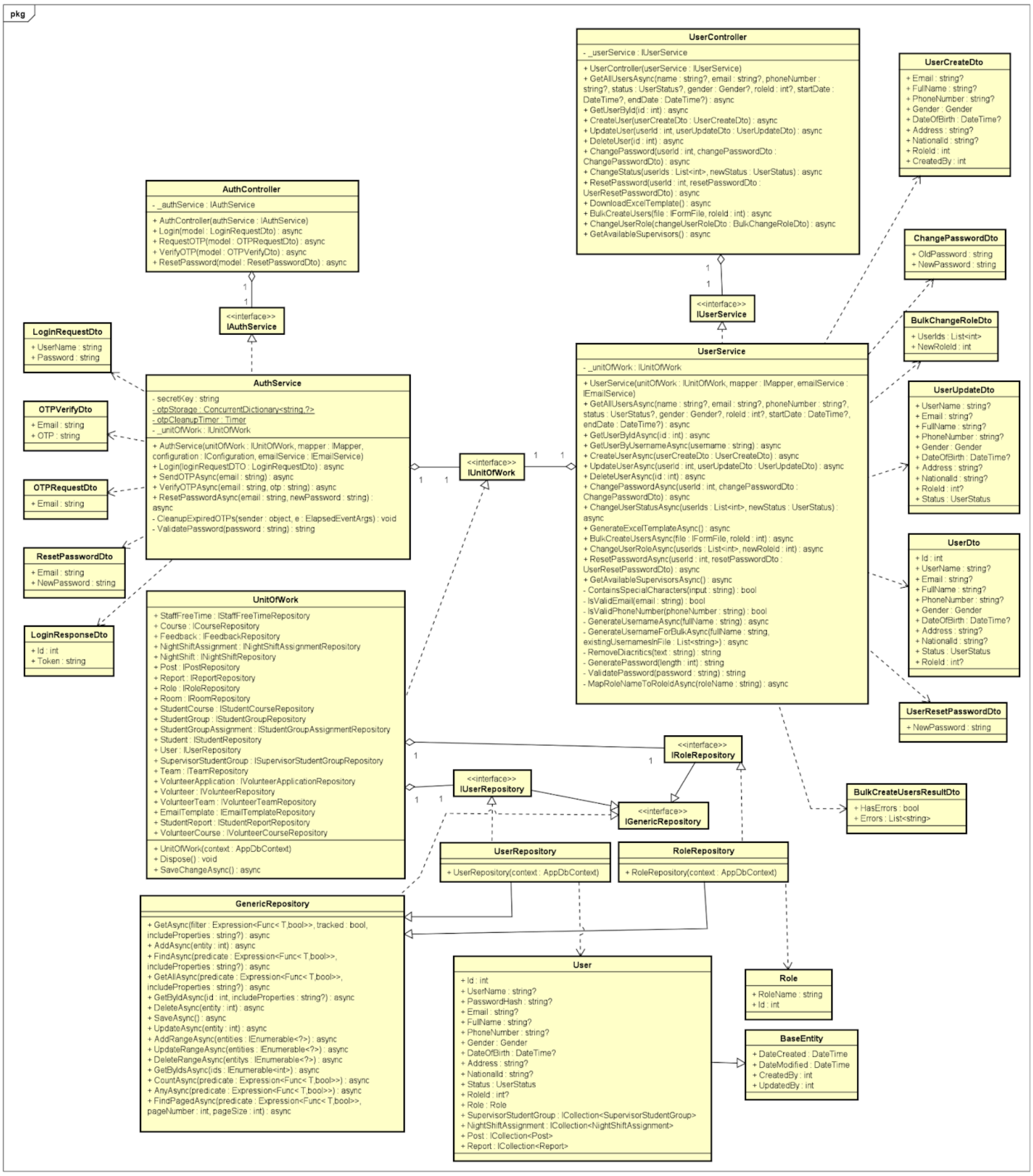


Figure 34: Account management class diagram

The link of image: [Account Management]()

3.1.2 Course Management

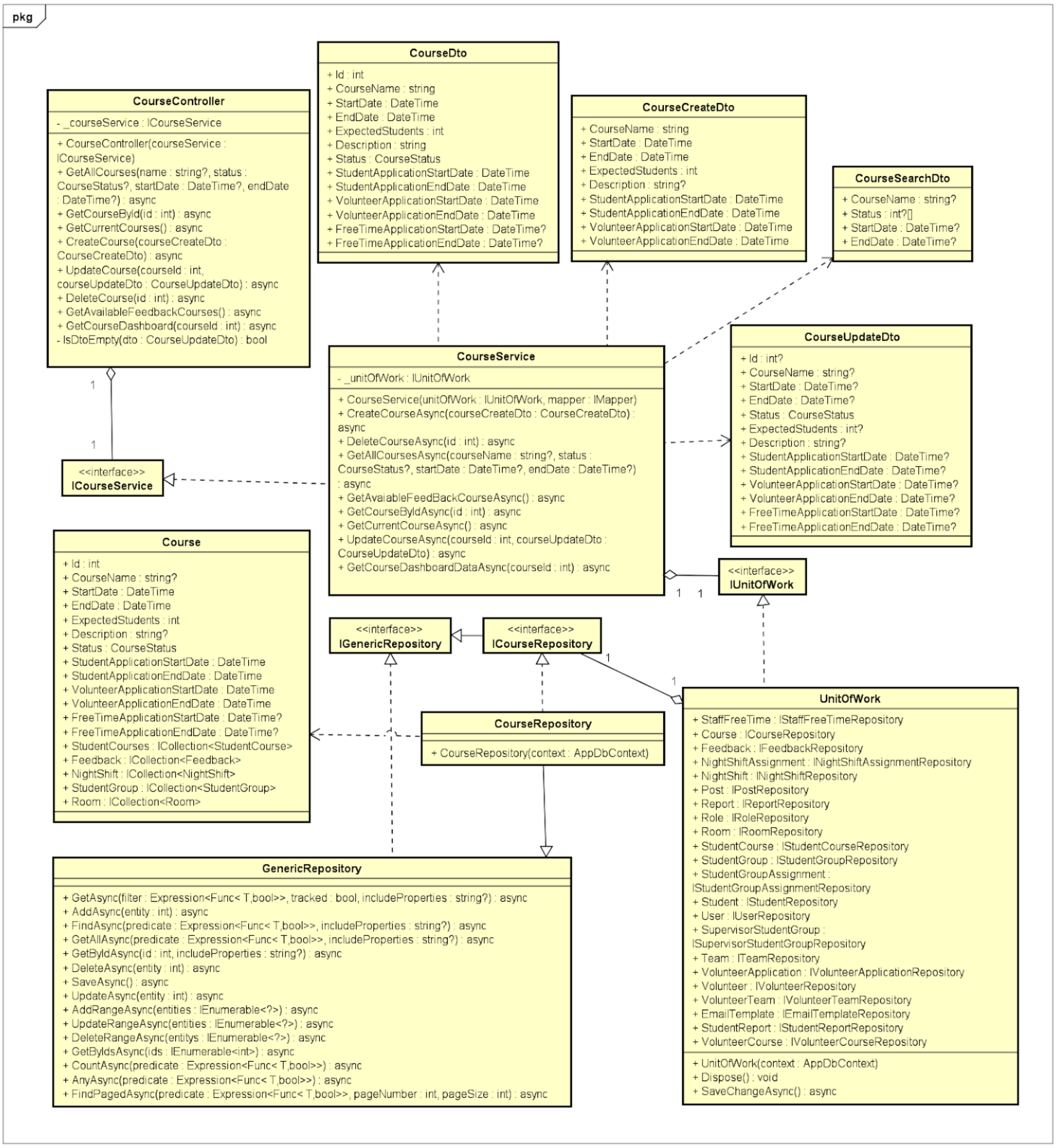


Figure 35: Account management class diagram

The link of image: [Course Management]

3.1.3 Student Management

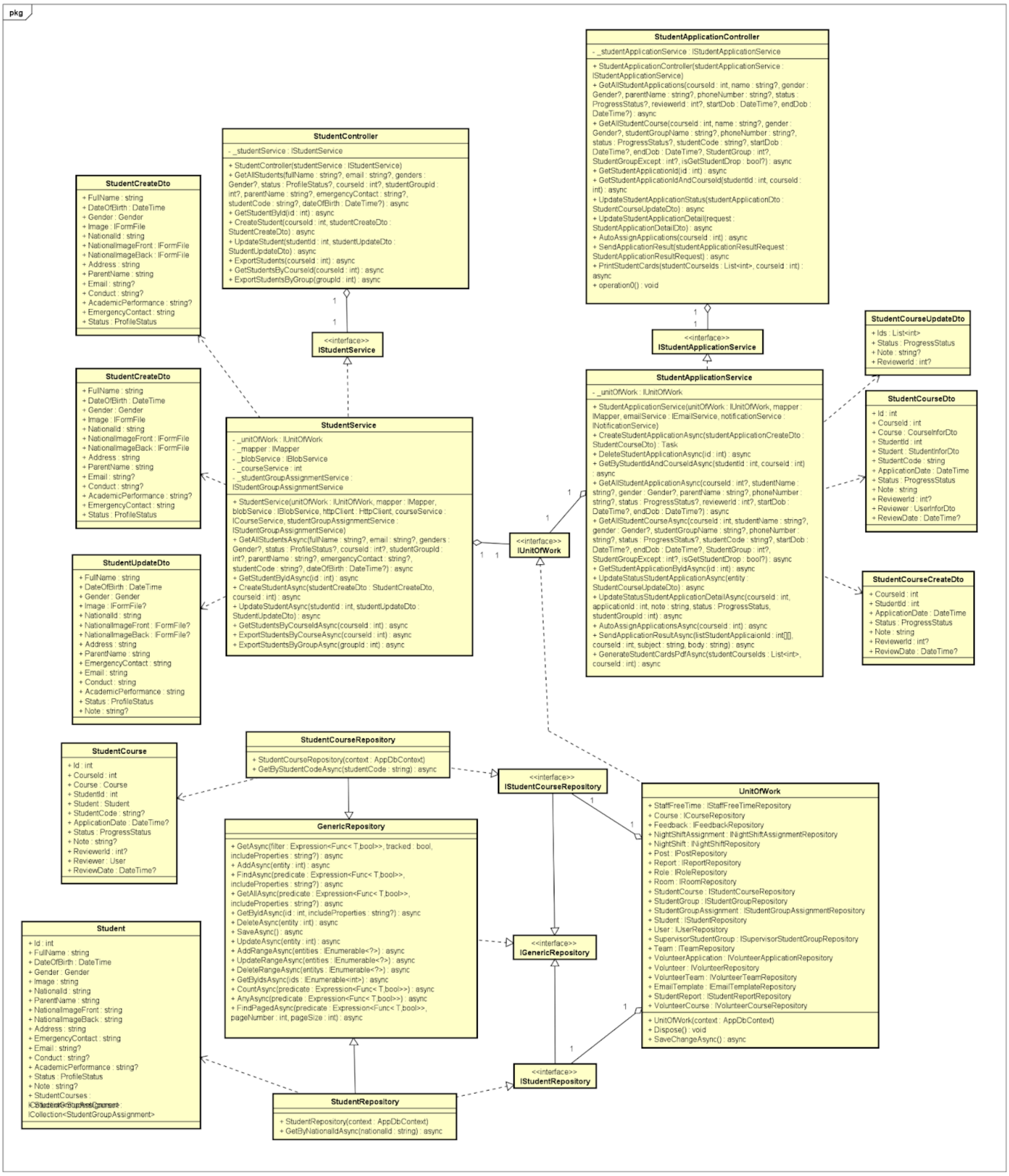


Figure 36: Student management class diagram

The link of image: [Student Management]

3.2 Sequence Diagram

3.2.1 Login

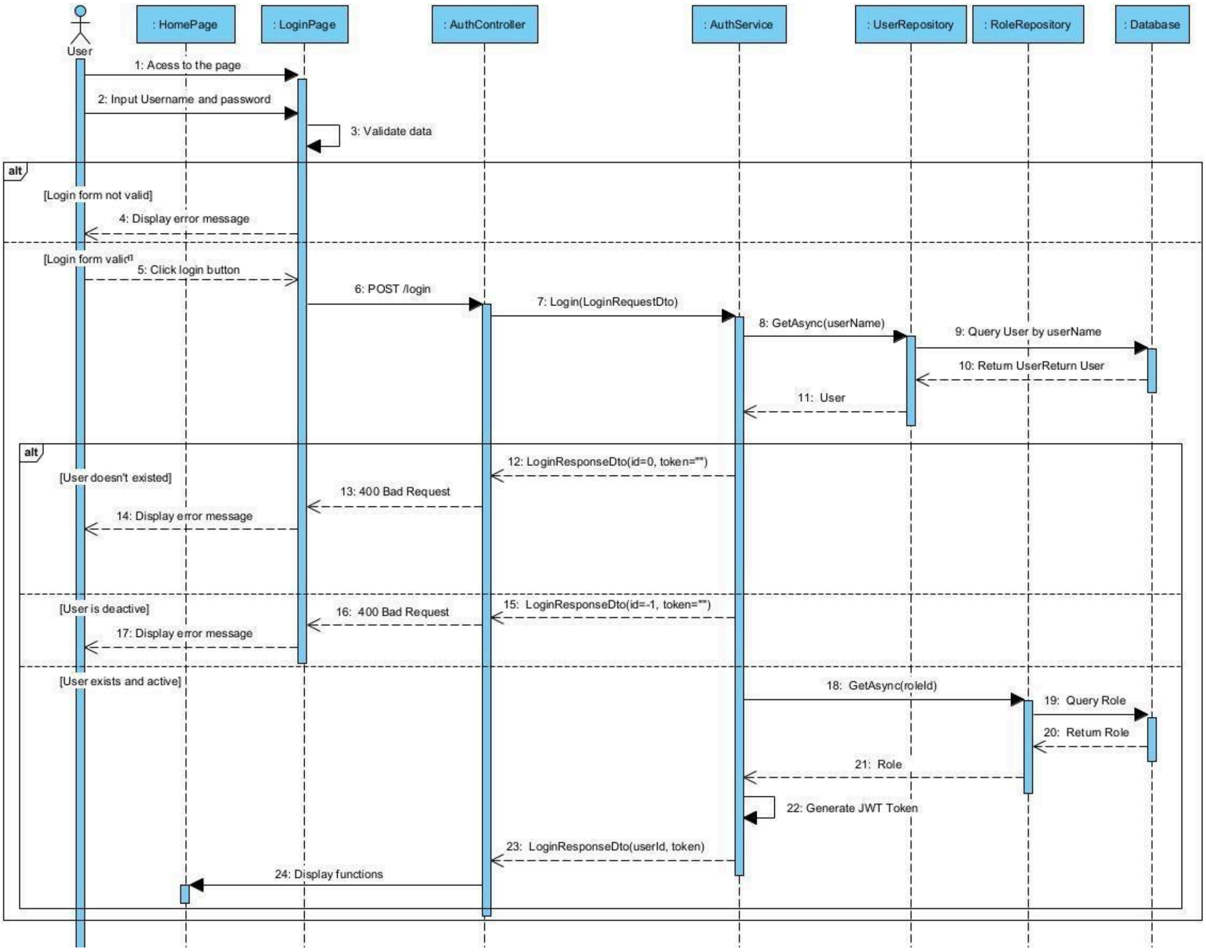


Figure 37: Login sequence diagram

The link of image: [Login]

3.2.2 Create course

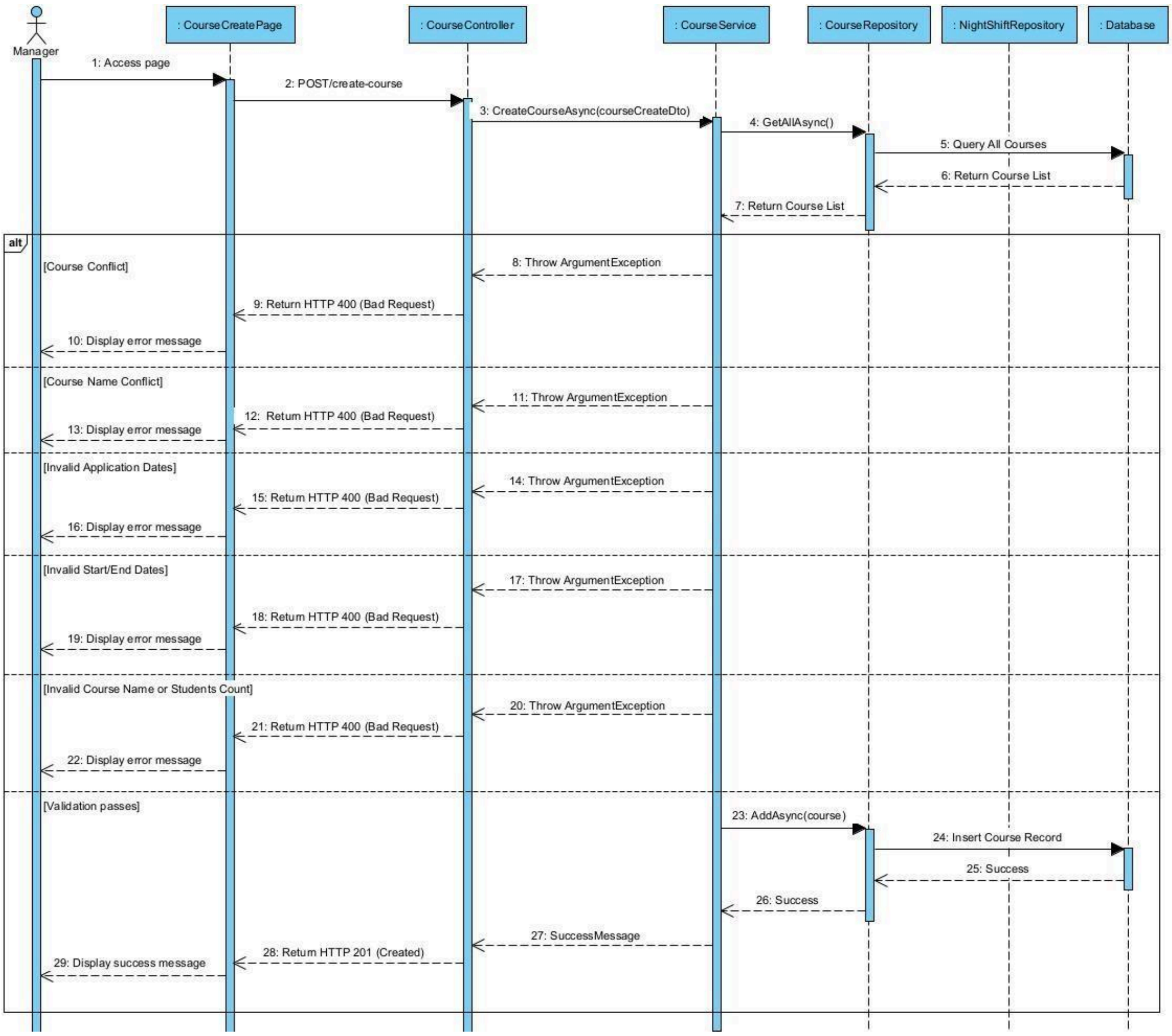


Figure 38: Create course sequence diagram

The link of image: Create course

3.2.3 Add student group

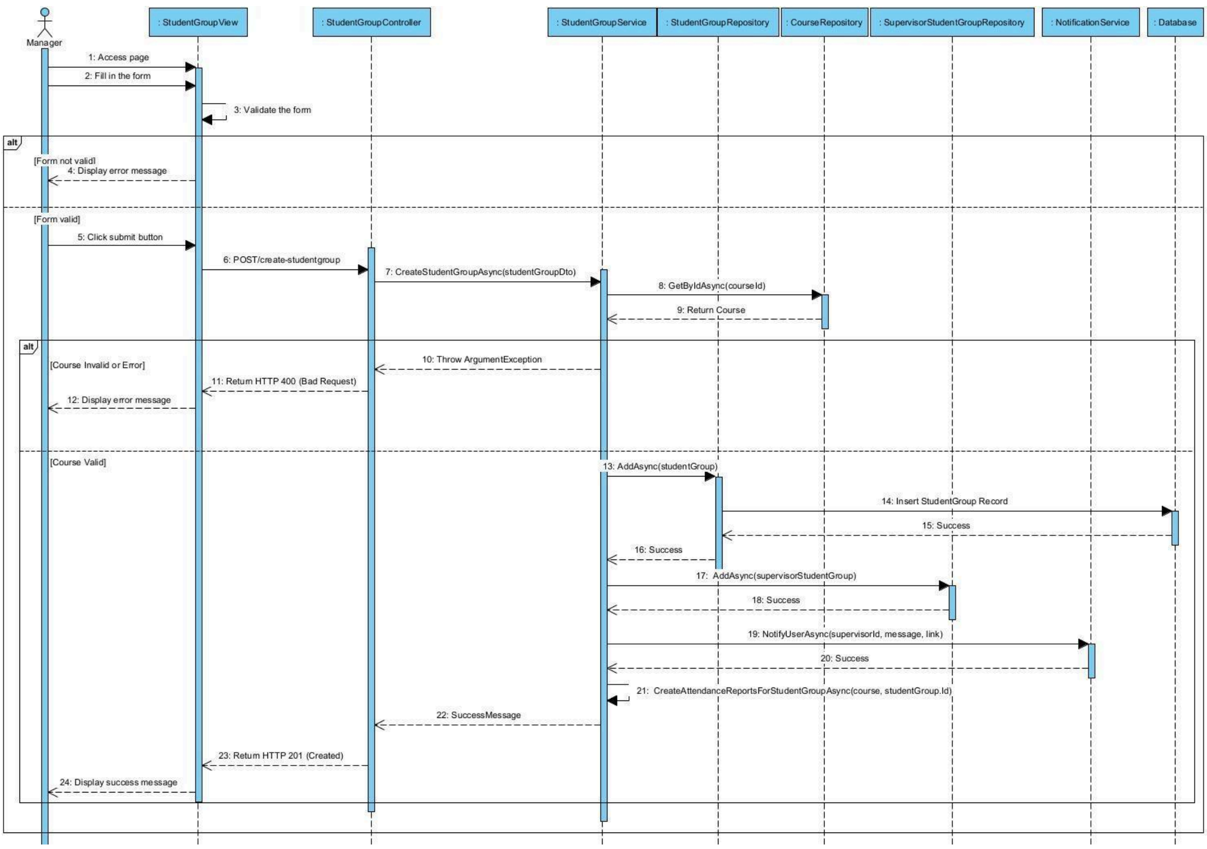


Figure 39: Add student group sequence diagram

The link of image: [Add student group]

V. Software Testing Documentation

1. Scope of Testing

1.1.Target of test

1.1.1. Feature, Functional

The test scope of the project includes all features – functions defined in [SEP490_G51_Report 1_Project Introduction]

1.1.2. Non - Functional

A. External Interfaces

1. User Interface

- All texts in the system are displayed in grammatically correct Vietnamese language.

- Whenever the user/admin performs an action that requires connection to the server, a loading indicator should appear so that the user knows what's going on and doesn't misunderstand that the service is down.
- There should be a clear alert when the app encounters a server error (e.g. offline).
- All pages should have a consistent visual theme and typeface.
- Icons and buttons should have tooltips or labels to assist users in understanding their functions.
- Users can copy and paste content using both the mouse and key combinations, enhancing the convenience of operation.
- Notifications displayed on the screen will have an easy-to-see notification frame, with clear content for users to quickly grasp.
- The interface must be compatible with popular browsers such as Chrome, Firefox, Microsoft Edge.
- The UI must be responsive on different size of screen
- The system must ask for confirmation (Y/N) for data deletion operations and bulk operations.
- The entire drop down list must be arranged in A to Z order and ascending numbers, for night shift workers, arranged in order of priority.
- When the user is in a certain function, the position of this function on the navigation bar (sidebar) will be highlighted, making it easy for users to know where they are in the system

2. Software Interfaces

Software interfaces that our framework will associated with amid operation:

- The system uses Gmail Service to send mails for users
- The system also uses SignalR to send notifications to users

B. Quality Attributes

1. Usability

- Implement solid design principles, such as clear icons, button types, tooltips, and logical menu structure. Users should comprehend their functions within 10 seconds.
- Ensure that the app's user interface is intuitive and easy to use, even for non-IT users with less technical expertise. A non-trained new user can understand and use the system proficiently and do the tasks they want with the accuracy of 90% in 2 hours.

- Include in-app guides to assist users in understanding how to use a specific functionality effectively.
- Ensure a consistent UX across.
- The Front-end web application should support Chrome, Edge browsers.

2. Security

To ensure system security, strong security measures such as data encryption, user authentication and access control are required.

- The system shall require users to authenticate using a unique username and a strong password that meets complexity requirements (minimum 8 characters, including uppercase letters, lowercase letters, numbers, and special characters).
- All the password of the users are encrypted
- When a user requests a password reset, the system shall send a verification code to the user's registered email.
- The system shall validate all user inputs to prevent injection attacks.
- Ensure only authorized users can access the corresponding functions and data. The system shall implement token-based authentication (JSON Web Tokens - JWT) to manage user sessions securely and efficiently.

3. Performance

- All web pages shall load completely within 5 seconds over a standard broadband connection (10 Mbps) under normal load conditions.
- The system shall retrieve and display requested data (e.g., course lists, user profiles) within 4 seconds after a query is submitted.

1.2. Testing Levels

We have 4 levels of testing: Unit Test, Integration Test, System Test, Acceptance test

No.	Test Level	In charge	Time	Focuses	Acceptance criteria
1	Unit Test	Developers	Implementing	Checks the functionality of software components that may be tested separately and looks for errors in them.	The entire source code must be entirely covered by branch conditions and correctly executed without unexpected exceptions.
2	Integration Test	Testers and Developer	After accepting UT	Make sure all the parts come together smoothly	Ensure accurate data flow between the controller layer and service layer, the repository, and then the database.
3	System Test	Testers	After accepting IT	Integrate various modules and test the interaction between them to ensure the accuracy of the data.	Each logic flow operates exactly as intended. System interfaces and non-functional requirements operate correctly during production.

4	Acceptance Test	End-users: Parents of student, manager, secretary, staff	Before the product is officially launched	Pure functional testing to check system behaviour with real data.	The system meets customer business needs,product availability.

Table 54. Description of Testing Types

2. Test Strategy

2.1 Testing Types

The test team has to test the following type on Postman, Google Chrome browser, Edge browser:

- Unit Testing
- GUI Testing
- API Testing
- Functional Testing
- Non-Functional Testing

Types	Objective	Technique	Completion Criteria
Unit Test	Verify that all method code in the program functions as intended with no unexpected exceptions for the function for which it was written.	Black box testing: A method for testing the user interface, input, and output. White box testing is performed to test the behaviour of each of the functions.	The entire set of source code's branch conditions must be covered and correctly performed without any unexpected exceptions.

GUI Testing	The primary goal of the GUI testing is to validate the features of the software, or the application performs as per the given requirement / specifications.	A graphical user interface includes all the elements such as menus, checkbox, buttons, colours, fonts, sizes, icons, content, and images. GUI testing is done to check the functionality and usability of design elements as a user for an application under test.	All GUI components render correctly. The intended functionality of the application can be executed using the GUI. Error messages are displayed correctly.
API Testing	Make sure the parts that are connected to one another function effectively. Verify the program logic all the way through from the controller layer to the service layer to the repository layer.	In order to track predicted and actual outcomes, testers develop test cases in accordance with the flow scenario that moves from the controller layer to the service, repository, and finally the database.	Ensure accurate data flow between the controller layer and the service layer, the repository, and the database.

Functional Testing	By engaging with the application through the Graphical User Interface (GUI) and examining the outputs or results, you may confirm the application and its internal workings.	Testing professionals develop test scenarios based on functional requirements. Black-box testing techniques will be used to construct test scenarios. A better checklist will be created by gathering common flaws.	To verify adequate data acceptance, processing, retrieval, and the appropriate application of the business rules, all functional test cases have been conducted. Address boundary, abnormal, and normal cases.
Non-functional Testing	Non-functional testing should improve the product's usability, effectiveness, maintainability, and portability. Reduces the production risk and expense linked to the product's non-functional features. The knowledge of current technology and product behaviour should be improved.	Testers design test scenarios using non-functional requirements as a basis. Testers write reports and run tests according to test scenarios.	The non-functional requirements listed in the software requirement specification must be complied with by all non-functional requirements.

Table 55. Description of testing types

2.2 Test Levels

Type of Tests	Test Level			
	Unit	Integration	System	Acceptance
Unit Test	X			
GUI Test		X	X	X
API Test		X		
Functional Testing	X	X	X	X
Non-functional Testing			X	X

Table 56. Test levels

2.3 Supporting Tools

Purpose	Tool	Vendor/In-house	Version
Create test plan	Google Docs	Google	Online
Create test report	Google Sheet	Google	Online
Manage test case	Google Sheet	Google	Online
Mange test result	Google Sheet	Google	Online
Test APIs	Postman	Postman Inc.	v11
Manage bugs	Gitlab issue	Gitlab	Online

Table 57. Tools test table

3. Test Plan

3.1 Human Resources

Worker/Doer	Role	Specific Responsibilities/Comments
ThuyDTT	Team member	• Create test cases and perform integration tests. • Create test cases and perform system tests. • Create test cases, checklists and perform acceptance tests.

ThuyDTT, NinhNT, ManhDD, PhucND	Team member	● Create test cases and perform unit tests. ● Summaries test reports.
NinhNT, HauNX, ManhDD	Team member	● Fix bugs front-end ● Log bugs front-end
NinhNT, ThuyDTT, PhucND, ManhDD	Team member	● Fix bugs back-end ● Log bugs back-end
ThuyDTT, NinhNT, ManhDD, HauNX, PhucND	Team member	● Review Test Plan. ● Execute Test (IT, ST) ● Review Test Report.

Table 58. Human Resources

3.2 Test Environment

Purpose	Tool	Provider
Write unit test documents	Google sheet	Google
Write test report documents	Google sheet, google doc	Google
Run UT	Visual studio code	Microsoft
Run IT	Postman, Chrome	Postman Inc.
Run ST, AT	Chrome, Microsoft Edge	Google, Microsoft
Tracking defects, issues and Q&A	Gitlab issue	Gitlab

Table 59. Test Environment

3.3 Test Milestones

Milestone Task		Start Date	End Date
Project Initiating	Create test plan	07/09/2024	09/09/2024
Iteration 1	Create test cases	09/09/2024	06/10/2024
	Execute unit test & integration test		
	Create test report		
Iteration 2	Create test cases	07/10/2024	27/10/2024
	Execute unit test & integration test & system test		
	Create test report		
Iteration 3	Create test cases	28/10/2024	17/11/2024
	Execute unit test & integration test & system test		
	Create test report		
Iteration 4	Create test cases	18/11/2024	08/12/2024
	Execute unit test & integration test & system test & acceptance test		
	Create test report		

Table 60. Test Milestones

4. Test Cases

4.1. Unit test

Functional testing will be done on the server development side by the backend developer. Development teams use unit testing to achieve the following advantages:

- Detect errors as early as possible, develop and test faster.
- Save development time, detecting errors early means fewer late changes and easier to detect problems than when done at a later stage.

● Reduce the level of errors in production code.
● Documentation can be easily generated from tests.
● Make code changes and refactoring easier by improving code design.

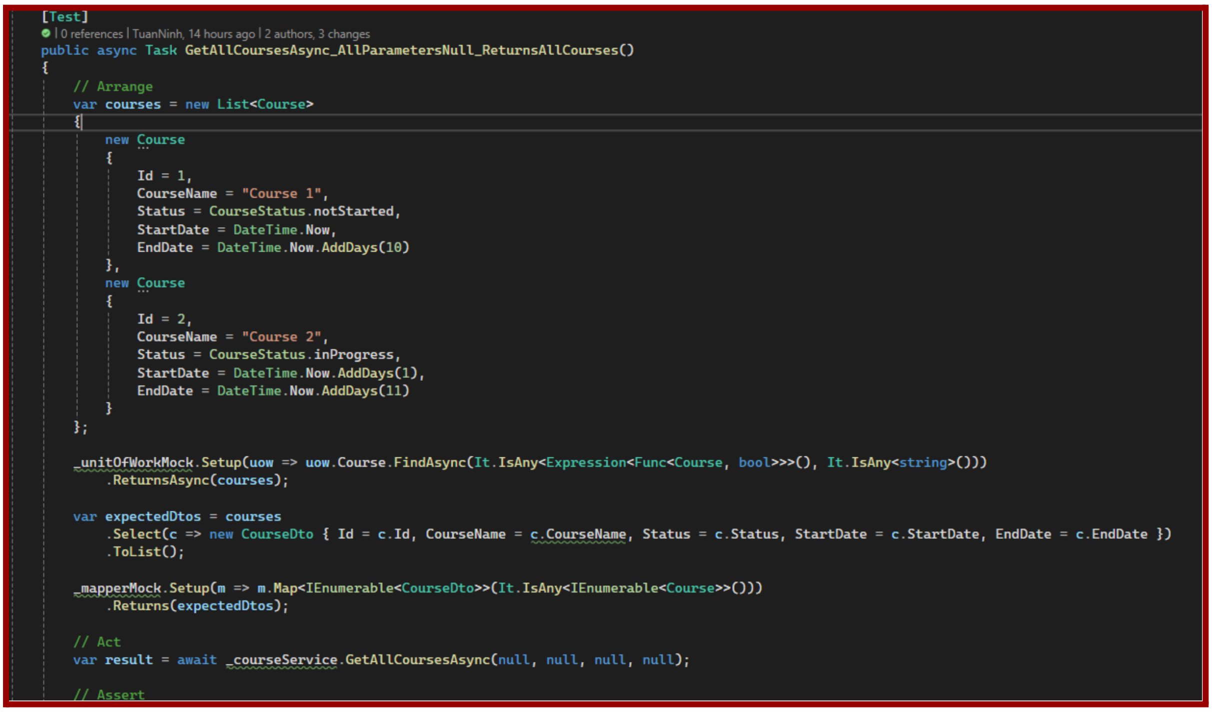


Figure 40: Unit Test example

4.2. Integration test.

Integration testing will involve incorporating the API into the application's interface to evaluate the data representation for the user. The tester will select test cases from the Test Report where actions are performed between functions or data is exchanged between modules. Then, Postman will be used to test the input-output data. If any errors occur during testing, especially related to unexpected data, the tester will log the errors or bugs on git issue and track the failed test case until it is resolved by the developers.

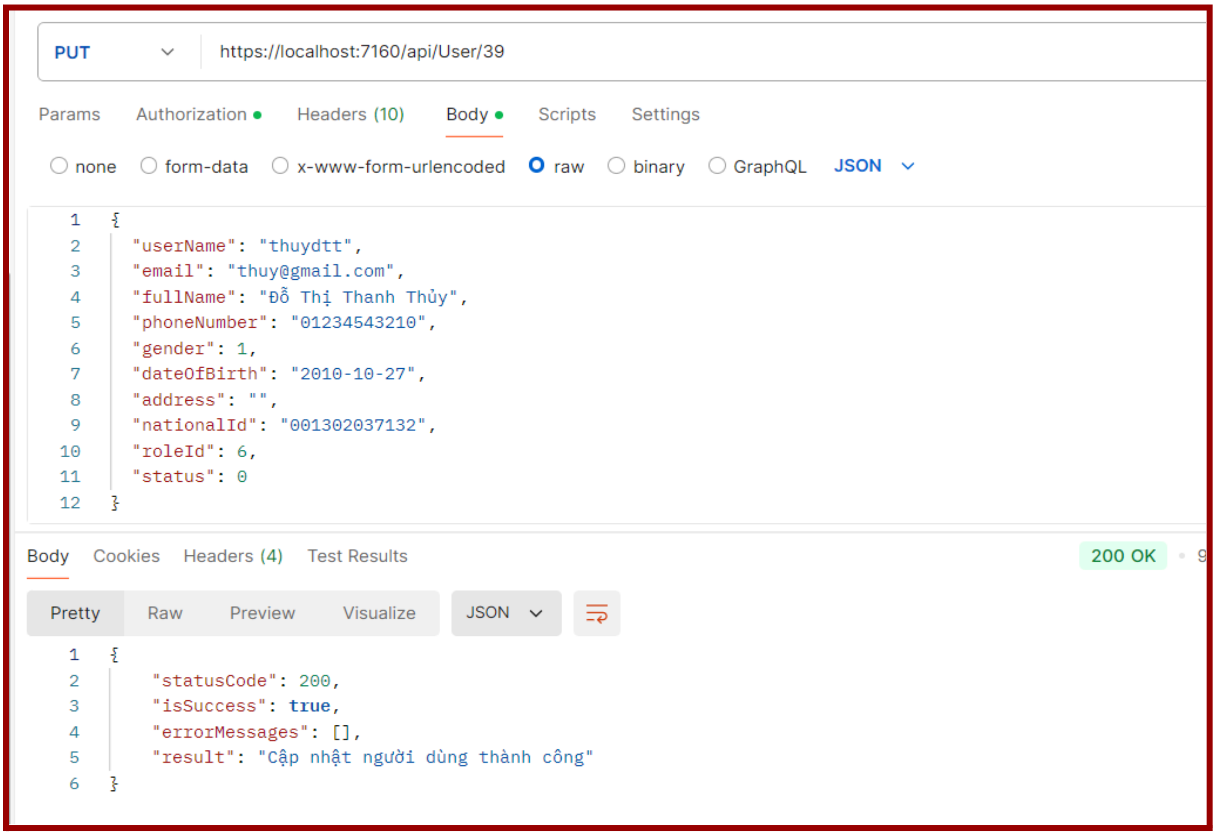


Figure 41: Integration Test example

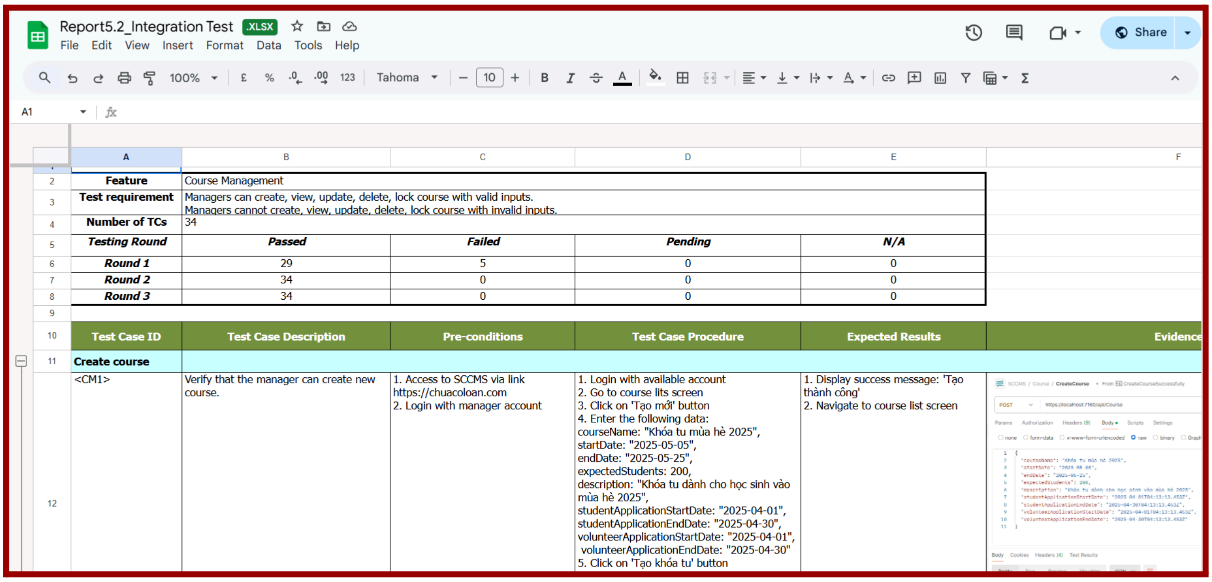
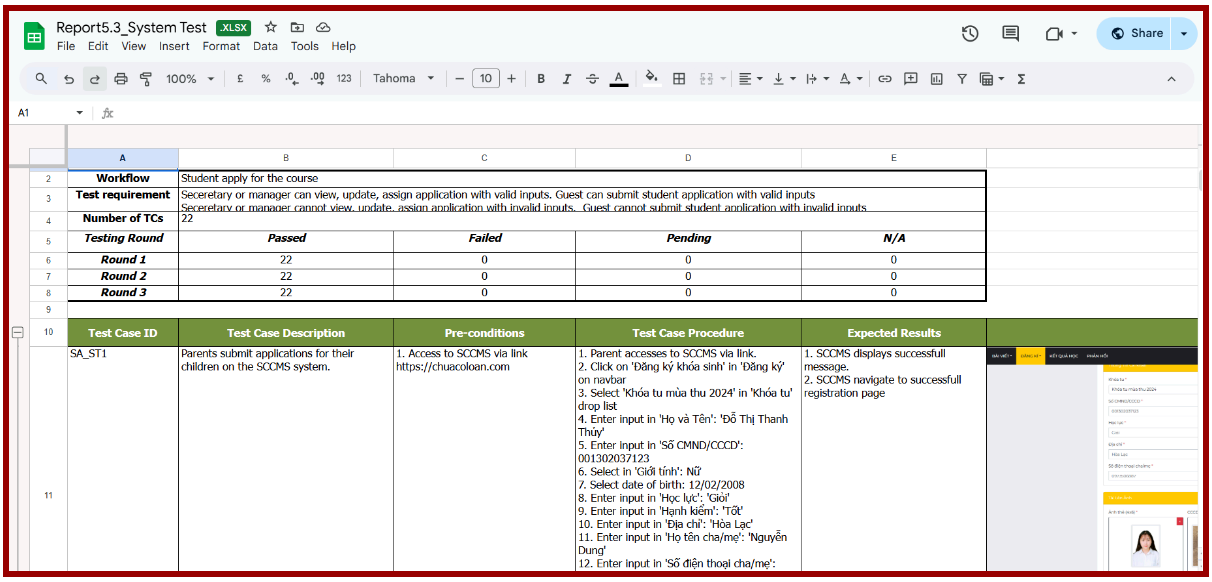


Figure 42: Integration Test report

4.3. System test

This testing phase takes place after the functionality outlined in the requirements has been fully implemented. We will conduct testing using Edge, Chrome browser. The main goal of this testing phase is to evaluate the performance of the system when the user operates it on the device, evaluating how the user interacts with the application in different scenarios.



Workflow	Student apply for the course			
Test requirement	Seceretary or manager can view, update, assign application with valid inputs. Guest can submit student application with valid inputs Seceretary or manager cannot view, update, assign application with invalid inputs. Guest cannot submit student application with invalid inputs			
Number of TCs	22			
Testing Round	***Passed***	***Failed***	***Pending***	***N/A***
Round 1	22	0	0	0
Round 2	22	0	0	0
Round 3	22	0	0	0

Test Case ID	Test Case Description	Pre-conditions	Test Case Procedure	Expected Results
SA_ST1	Parents submit applications for their children on the SCCMS system.	1. Access to SCCMS via link https://chuacoloan.com	1. Parent accesses to SCCMS via link. 2. Click on 'Đăng ký khóa sinh' in 'Đăng ký' on navbar 3. Select 'Khóa tu mùa thu 2024' in 'Khóa tu' drop list 4. Enter input in 'Họ và Tên': 'Đỗ Thị Thanh Thủy' 5. Enter input in 'Số CMND/CCCD': 001302037123 6. Select in 'Giới tính': Nữ 7. Select date of birth: 12/02/2008 8. Enter input in 'Học lực': 'Giỏi' 9. Enter input in 'Hạnh kiểm': 'Tốt' 10. Enter input in 'Địa chỉ': 'Hòa Lạc' 11. Enter input in 'Họ tên cha/mẹ': 'Nguyễn Dung' 12. Enter input in 'Số điện thoại cha/mẹ':	1. SCCMS displays successfull message. 2. SCCMS navigate to successfull registration page

Figure 43: System Test report

4.4. Acceptance test

Acceptance Testing is a level of software testing where a system is tested for acceptability. The purpose of this test is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery.

ID	Checklist	Test Result	
		Pass	Fail
	GUI and Usability		
1	The website design takes inspiration from the following design: Figma Design	☑	☐
2	The screen is well-organized, providing a user-friendly interface.	☑	☐
3	The steps are short and easy to understand so that the user can interact with the application without training	☑	☐
4	The main functions are organized into tabs, facilitating easier access and navigation for users.	☑	☐
5	Links, buttons, and checkboxes are easily clickable.	☑	☐
6	Important commands are displayed as buttons with distinct background colors.	☑	☐
7	The displayed text is in Vietnamese for the convenience of most employees	☑	☐
8	The static text is clear, concise, and meaningful.	☑	☐
9	Pop-up menus are provided for the user to access information about an object's properties or perform specific tasks on the object.	☑	☐
10	System display notification message when meet trouble, error.	☑	☐

Feature						
	Function	Test case description	Test Case Proceduce	Expected Result	Pass	Fail
13	Login	Users can login with username and password with authorized account	1. Access to SCCMS via link 2. Navigate to Login screen 3. Login with authorized account	1. Display login page 2. Navigate to homepage of authorized account 3. If the password and username is invalid or incorrect, the system will display an error message.	☑	☐
14	Import user list	Users can create multiple accounts by importing a user list file.	1. Login with manager account 2. Click on 'Danh sách người dùng' in 'Người dùng' on navbar 3. Click on 'Nhập từ tệp' button 5. Click on 'Chọn tệp' button 6. Choose file 7. Click on 'Xác nhận' button	1. SCCMS will display success message 2. Navigate to user list screen 3. Send account information via email to users 4. If the input fields are invalid, the system will report an error and cannot create successfully.	☑	☐
15	Create course	Users login with manager account and create course	1. Login with manager account 2. Click on 'Khóa tu' on navbar 3. Click on 'Tạo mới' button 5. Input following data: 'Tên khóa tu': 'Khóa tu mùa hè 2025' 'Thời gian bắt đầu khóa tu' :05/01/2025 'Thời gian kết thúc khóa tu':	1. SCCMS displays successfull message. 2. SCCMS redirects to "Danh sách khóa tu" screen. 3.The course selection dropdowns on other screens will have the newly added course added. 4. If the input fields are invalid, the	☑	☐
	Create student group	Users login with manager account	1. Login with manager account	1. Display success message: 'Tạo thành		

Figure 44: Acceptance Test report

5. Test Reports

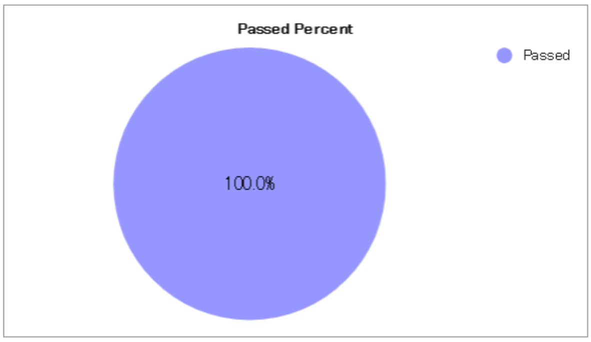
5.1 Unit Test

No	Function code	Passed	Failed	Untested	N	A	B	Total Test Cases
1	GetAllStudentApplication	19	0	0	18	0	1	19
2	GetStudentApplicationById	4	0	0	2	1	1	4
3	UpdateStatusStudentApplication	8	0	0	2	5	1	8
4	AutoAssignApplication	3	0	0	1	2	0	3
5	GetAllCourse	11	0	0	9	0	2	11
6	GetCourseById	4	0	0	1	2	1	4
7	CreateCourse	12	0	0	1	10	1	12
8	Updatecourse	12	0	0	1	10	1	12
9	DeleteCourse	4	0	0	1	2	1	4
10	GetAllUsers	21	0	0	13	5	3	21
11	ValidatePassword	9	0	0	1	7	1	9
12	GetAllStudent	24	0	0	10	12	2	24
13	GetStudentById	4	0	0	1	2	1	4
14	CreateStudent	23	0	0	10	11	2	23
15	Login	8	0	0	2	6	0	8
16	VerifyOTP	9	0	0	1	8	0	9
17	ResetPassword	11	0	0	1	9	1	11
18	IsValidEmail	7	0	0	2	5	0	7
19	CreateTeam	14	0	0	5	9	0	14
20	GetAllTeamsByCourseId	4	0	0	1	2	1	4
21	GetTeamById	4	0	0	1	2	1	4
22	UpdateTeam	14	0	0	5	9	0	14
23	DeleteTeam	4	0	0	2	2	0	4
24	CreateFeedback	9	0	0	1	7	1	9
25	GetFeedbackById	4	0	0	2	1	1	4
26	GetAllFeedback	7	0	0	6	0	1	7
27	DeleteFeedback	4	0	0	1	2	1	4

28	CreateRoom	5	0	0	2	2	1	5
29	GetRoomById	3	0	0	1	1	1	3
30	DeleteRoom	3	0	0	1	2	0	3
31	UpdateRoom	7	0	0	2	3	2	7
32	GetAllFreeTime	4	0	0	2	0	2	4
33	GetFreeTimeById	3	0	0	1	1	1	3
34	DeleleFreeTime	3	0	0	1	1	1	3
35	AssignStaffToShift	6	0	0	1	3	2	6
36	UpdateAssignment	7	0	0	1	4	2	7
37	UpdateAssignmentStatus	5	0	0	2	1	2	5
38	CreateUser	19	0	0	1	18	0	19
39	ChangeUserRoleAsync	8	0	0	1	7	0	8
40	MapRoleNameToRoleIdAsync	6	0	0	2	4	0	6
41	ResetPassword	11	0	0	1	9	1	11
42	GeneratePassword	3	0	0	2	1	0	3
43	GetUserById	4	0	0	1	3	0	4
44	ChangePassword	14	0	0	1	12	1	14
45	ChangeUserStatusAsync	5	0	0	1	4	0	5
46	CreatePost	7	0	0	1	6	0	7
47	UpdatePost	8	0	0	1	7	0	8
48	GetPostByIdAsync	3	0	0	3	0	0	3
49	GetAllPostsAsync	12	0	0	8	4	0	12
50	GettAllVolunteerCourse	17	0	0	9	7	1	17
51	UpdateVolunteerApplication	8	0	0	2	5	1	8
52	GetVolunteerApplicationById	4	0	0	1	2	1	4
53	GetVolunteerByCourseId	4	0	0	1	2	1	4
54	GetVolunteerById	4	0	0	1	2	1	4

50	GettAllVolunteerCourse	17	0	0	9	7	1	17
51	UpdateVolunteerApplication	8	0	0	2	5	1	8
52	GetVolunteerApplicationById	4	0	0	1	2	1	4
53	GetVolunteerByCourseId	4	0	0	1	2	1	4
54	GetVolunteerById	4	0	0	1	2	1	4
	Sub total	440	0	0	152	242	46	440

Test coverage	**100.00**	%
Test successful coverage	**100.00**	%
Normal case	**34.55**	%
Abnormal case	**55.00**	%
Boundary case	**10.45**	%



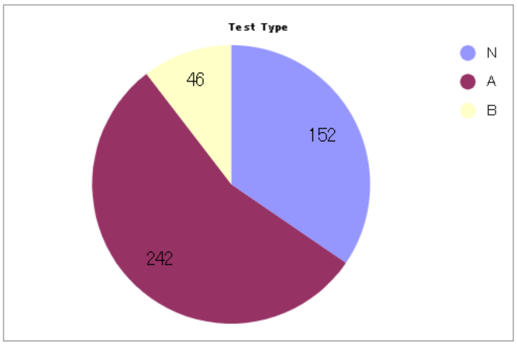


Figure 45 Unit Test Result

The reports of the Unit test procedure are shown details via: Report 5.1

5.2 Integration Test

No	Module code	Passed	Failed	Pending	N/A	Number of test cases
1	Authentication	38	0	0	0	38
2	Student Management	52	0	0	0	52
3	Course Management	34	0	0	0	34
4	Student Application Management	21	0	0	0	21
5	User Management	44	0	0	0	44
6	Student Group Management	24	0	0	0	24
7	Supervisor Management	5	0	0	0	5
8	Volunteer Management	51	0	0	0	51
9	Volunteer Application Management	18	0	0	0	18
10	Team Management	24	0	0	0	24
11	Feedback Management	15	0	0	0	15
12	Night Shift Management	36	0	0	0	36
13	Post Management	19	0	0	0	19
14	Report Management	21	0	0	0	21
	Sub total	402	0	0	0	402

Test coverage **100.00** %
Test successful coverage **100.00** %

Figure 46 Unit Test Result

The reports of the Integration test procedure are shown details via: Report 5.2

5.3 System Test

No	Module code	Passed	Failed	Pending	N/A	Number of test cases
1	User Authorization	33	0	0	0	33
2	Course management	18	0	0	0	18
3	Student group managerment	19	0	0	0	19
4	Team management	20	0	0	0	20
5	Post management	8	0	0	0	8
6	Student apply for the course	22	0	0	0	22
7	Student management	12	0	0	0	12
8	Volunteer apply for the course	22	0	0	0	22
9	Volunteer management	13	0	0	0	13
10	Nightshift management	41	0	0	0	41
11	Report management	7	0	0	0	7
12	Feedback management	12	0	0	0	12
	Sub total	227	0	0	0	227

Test coverage **100.00** %
Test successful coverage **100.00** %

Figure 47 Unit Test Result

The reports of the System test are shown details via: Report 5.3

5.4 Acceptance test

No	Yes	No	N/A	Number of test cases
1	45	0	0	**45**
Summary	**45**	**0**	0	**45**

Test coverage **100%**
Test successful coverage **100%**

Figure 48 Acceptance Test Result

The reports of the System test are shown details via: Report 5.4

VI. Release Package & User Guides

1. Deliverable Package

No	Deliverable Item	Sub-items	Type	Version
Code Package				
1	Front-end	FA24_SEP490-G51_FrontEnd.zip	zip	1.0
2	Back-end	FA24_SEP490-G51_BackEnd.zip	zip	1.0
Database				
3	Database script	SCCMS	mdf	1.0
Documents				
4	Project Introduction	FA24_SEP490-G51_Report1_Project Introduction	pdf	1.0
5	Project management plan	FA24_SEP490-G51_Report2_Project Management Plan	pdf	1.0
6	Software Requirement Specification	FA24_SEP490-G51_Report3_Software Requirement Specification	pdf	1.0
7	Software Design Document	FA24_SEP490-G51_Report4_Software Design Document	pdf	1.0
8	Software User Guide	FA24_SEP490-G51_Report6_Software User Guide	pdf	1.0
9	Final Report	FA24_SEP490-G51_Report7_FinalReport	pdf	1.0
Testing				
10	Testing Plan	FA24_SEP490-G51_Test-Plan	pdf	1.0
11	Unit Test	FA24_SEP490-G51_Report5.1_UnitTest	xlsx	1.0
	Integration Test	FA24_SEP490-G51_Report5.2_Itegration Test	xlsx	1.0

	System Test	FA24_SEP490-G51_Report5.3_SystemTest	xlsx	1.0
	Acceptance Test	FA24_SEP490-G51_Report5.4_AcceptanceTest	xlsx	1.0
Slide				
12	Presentation Slide	FA24_SEP490-G51_Slides	pdf	1.0

Table 61: Deliverable Package

2. Installation Guides

2.1 System Requirements

For Front-end application:

- **Hardware:**
  - CPU: At least 4 cores.
  - RAM: At least 8 GB.
- **Software:**
  - Operating System: Windows 10/11.
  - Required Tools: Visual Studio Code, Node.js, Git.

For Back-end application:

- **Hardware:**
  - CPU: At least 8 cores.
  - RAM: At least 8 GB.
- **Software:**
  - Operating System: Windows 10/11.
  - Required Tools: Visual Studio 2022, .NET 8.0 SDK, SQL Server.

For Production environment:

- **Hardware:**
  - CPU: At least 12 cores.

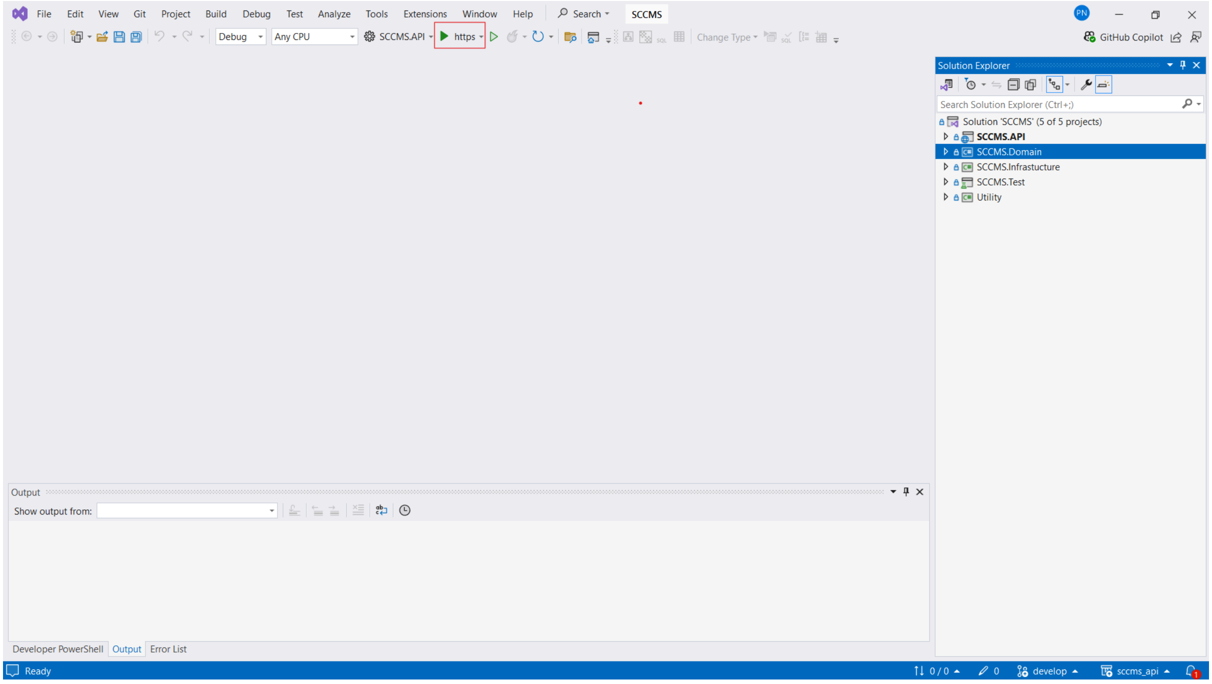
- RAM: At least 90 GB.
- Storage: 500 GB SSD.

2.2 Installation Instruction

2.2.1 Install Back-end Application

Dev Environment:

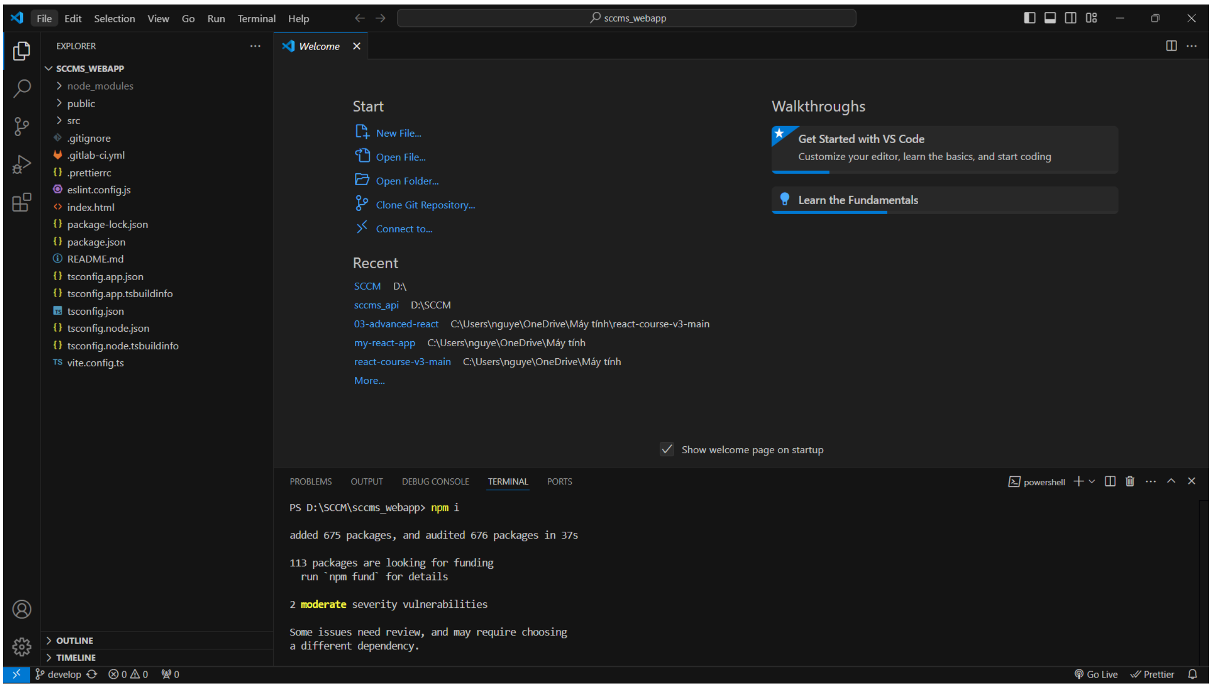
1. **Install Visual Studio:**
   - Download the latest version of Visual Studio 2022 from: https://visualstudio.microsoft.com/downloads/
   - During setup, ensure you select the **ASP.NET and web development** workload.
2. **Install .NET SDK:**
   - Download the .NET 8.0 SDK from: https://dotnet.microsoft.com/download/dotnet
3. **Install SQL Server:**
   - Download SQL Server 2019 or higher from: https://www.microsoft.com/en-us/sql-server/sql-server-downloads
   - Optionally, install SQL Server Management Studio (SSMS) for easier database management: https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms
4. **Start Project:**
   - Clone the back-end project repository from your version control system (e.g., GitLab).
   - Open the project in Visual Studio and configure the required settings (e.g., database connection string).
   - Run the project using F5 or the **dotnet run** command.



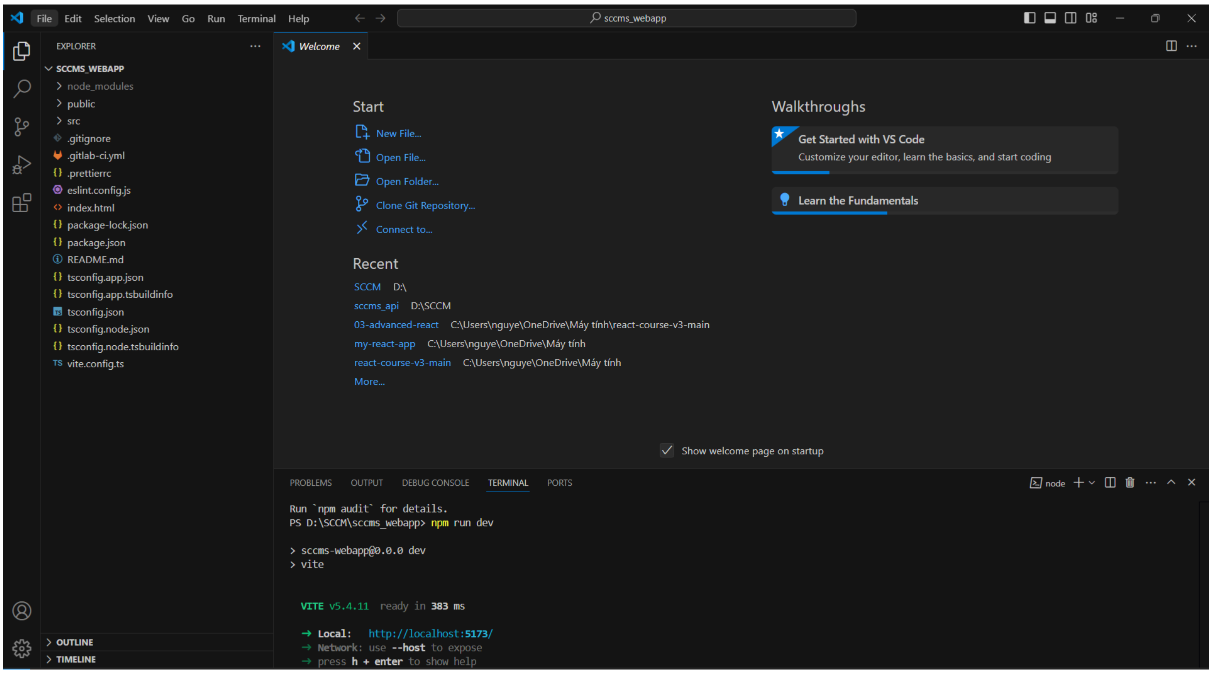
2.2.2 Install Front-end Application

Dev Environment:

1. **Install Visual Studio Code:**
   - Download the latest version of Visual Studio Code from: [https://code.visualstudio.com/.](https://code.visualstudio.com/)
2. **Install Node.js and NPM:**
   - Download the latest stable version of Node.js (which includes NPM) from: [https://nodejs.org/.](https://nodejs.org/)
3. **Start Project:**
   - Clone the back-end project repository from your version control system (e.g., GitLab).
   - Open the project in Visual Studio Code and run the “npm i” command to install all the necessary node modules.



- Run the project by “npm run dev” command.



Production Environment:

1. **Deployment:**
   - Use deployment tools like **Azure App Services**, to deploy the application.
   - Configure the production database and application settings.

2. **Repository:**
    - GitLab link:
    - **For Front-end application:**
      https://gitlab.com/nguyentuanninh02/sccms_webapp.git
    - **For Back-end application:**
      https://gitlab.com/nguyentuanninh02/sccms_api.git

3. User Manual

3.1 Overview

SSCMS is a web-based platform designed to optimize the registration, management, and communication for the summer cultivation courses at Co Loan Pagoda in Ninh Binh. The system aims to support over 500 students and volunteers each summer by automating key processes, which currently rely heavily on manual procedures.

The system has five key roles, each with specific functionalities as shown in the context diagram:

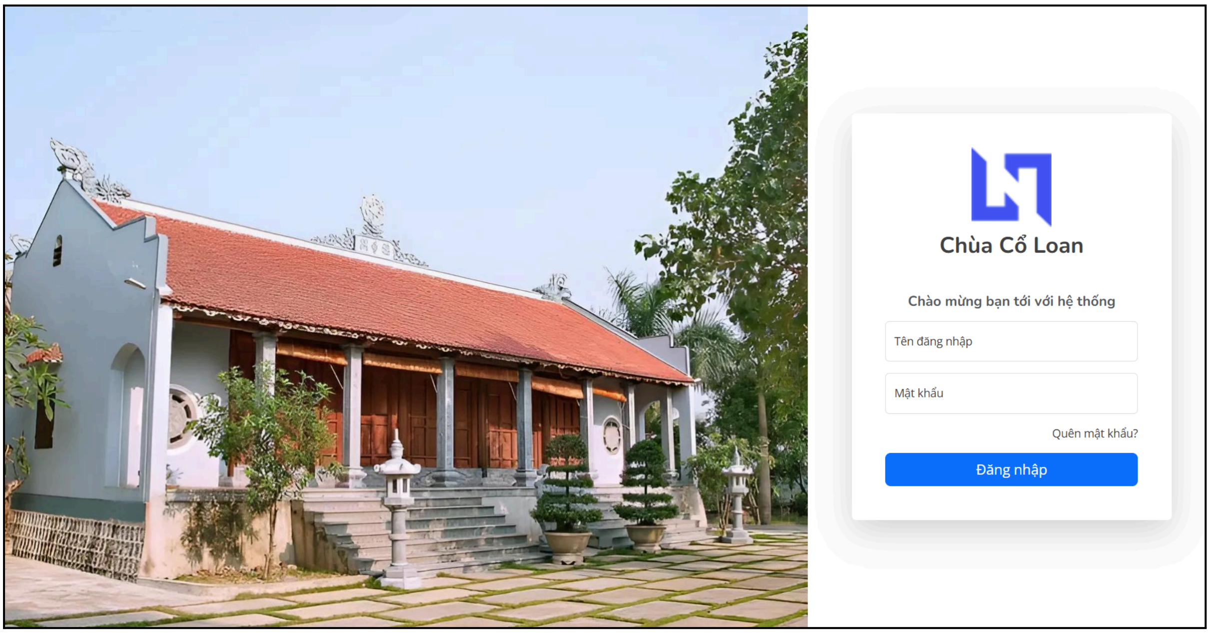
1. **Parents**:
    - **Role**: Prepare and submit the necessary documentation for their children before enrollment.
    - **Responsibilities**: Parents are responsible for ensuring all required forms and information are complete and submitted prior to the course. They can also receive updates or notifications about the course schedule, requirements, and their child's progress.
2. **Students**:
    - **Role**: Participate in the summer cultivation course.
    - **Responsibilities**: Students will arrive on the enrollment day to complete the registration process, receive uniforms and ID cards, and are guided to their accommodations. They also participate in various activities such as meditation, and educational programs, while their progress is tracked by the system.
3. **Staff**
    - **Role**: Supervise and manage the daily activities of the students.

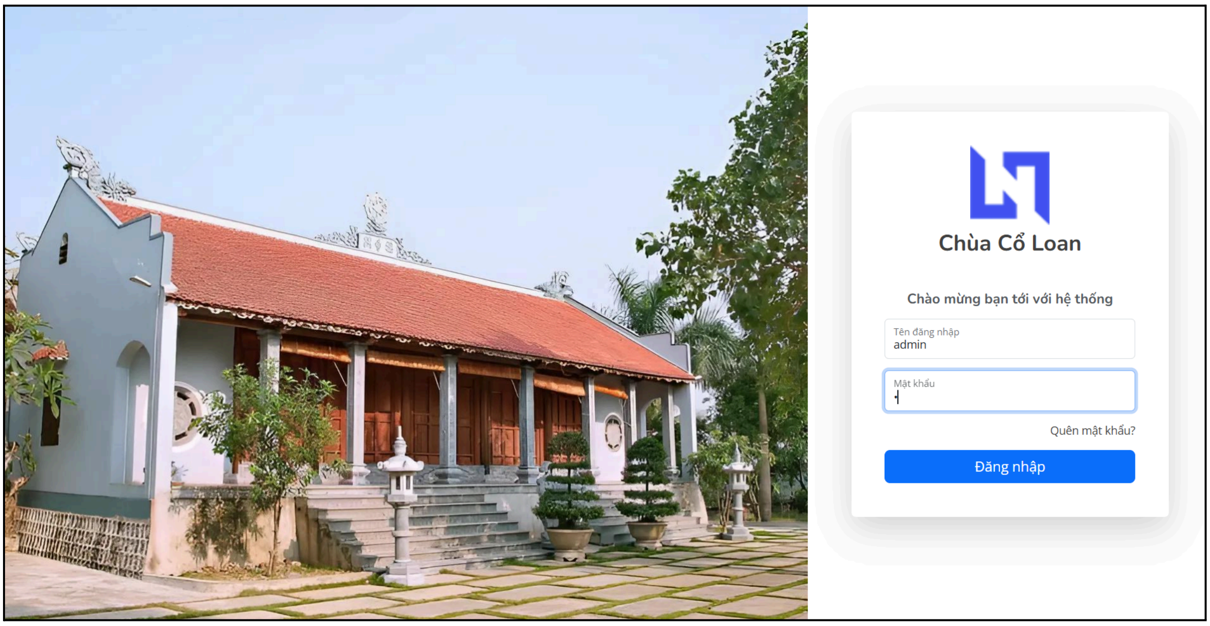
    - **Responsibilities**: Staff are responsible for coordinating students during the course, guiding them to their accommodations, overseeing activities, and ensuring students comply with the course schedule. They also play a key role in helping students adapt to the program.
4. **Secretary:**
    - **Role**: Handle administrative tasks and communication.
    - **Responsibilities**: The secretary is in charge of processing registrations, sending out notifications to students and parents, and ensuring all documentation is accurate and complete. The secretary also acts as the main point of contact for any administrative queries related to the course.
5. **Administrator:**
    - **Role**: Oversee and manage the overall system operations.
    - **Responsibilities**: The administrator manages user roles and access permissions, ensuring proper coordination between all parties involved (parents, students, leaders, and secretary). They monitor the system's functionality, manage data entry, and ensure the smooth operation of all backend processes related to the course management.

Below is how our team defining the user guides for the function managing trips, the most important function of our website. Other user guides are defined in the document Report6_ Software User Guide_v1.0.

3.2 User authorization

3.2.1. Log in

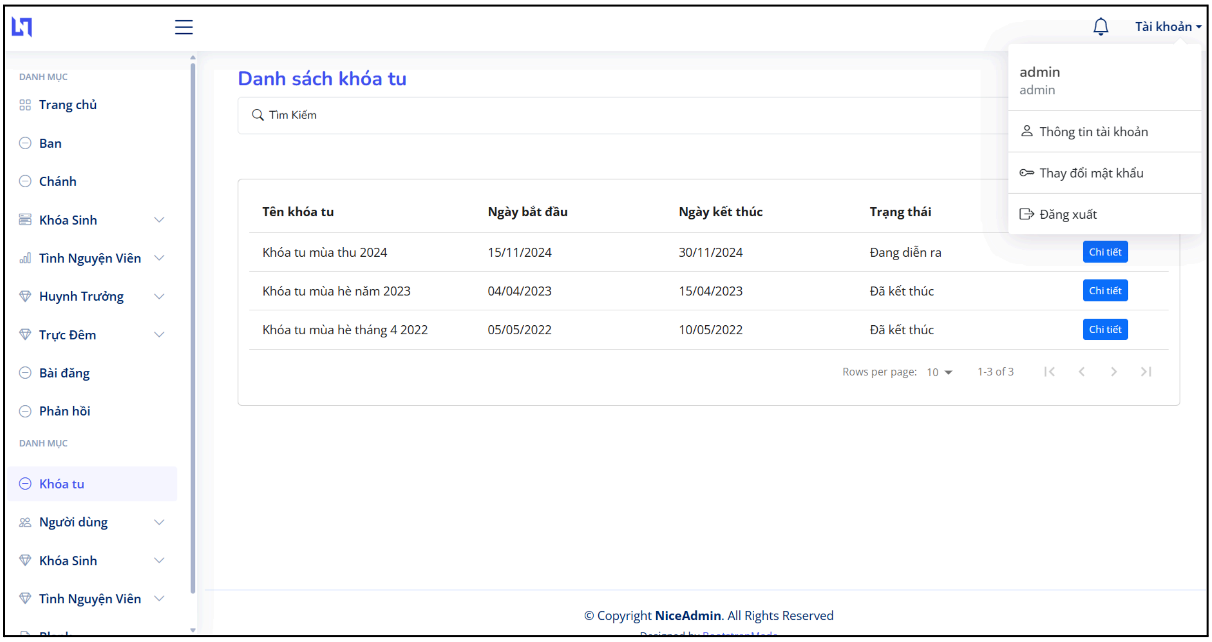




To log in, follow these steps:

1. User is redirected to the “Login” Screen.
2. User enters the correct username and password, then presses the “Đăng nhập” button.
3. After successful login, the user is redirected to the dashboard.

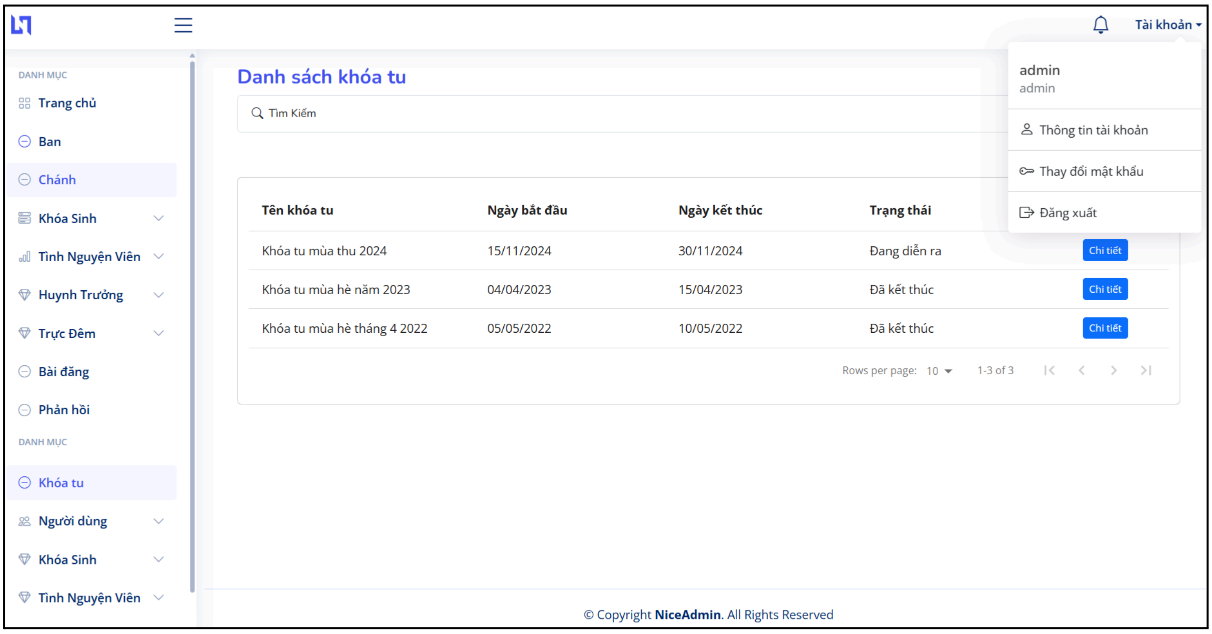
3.2.2. Log out

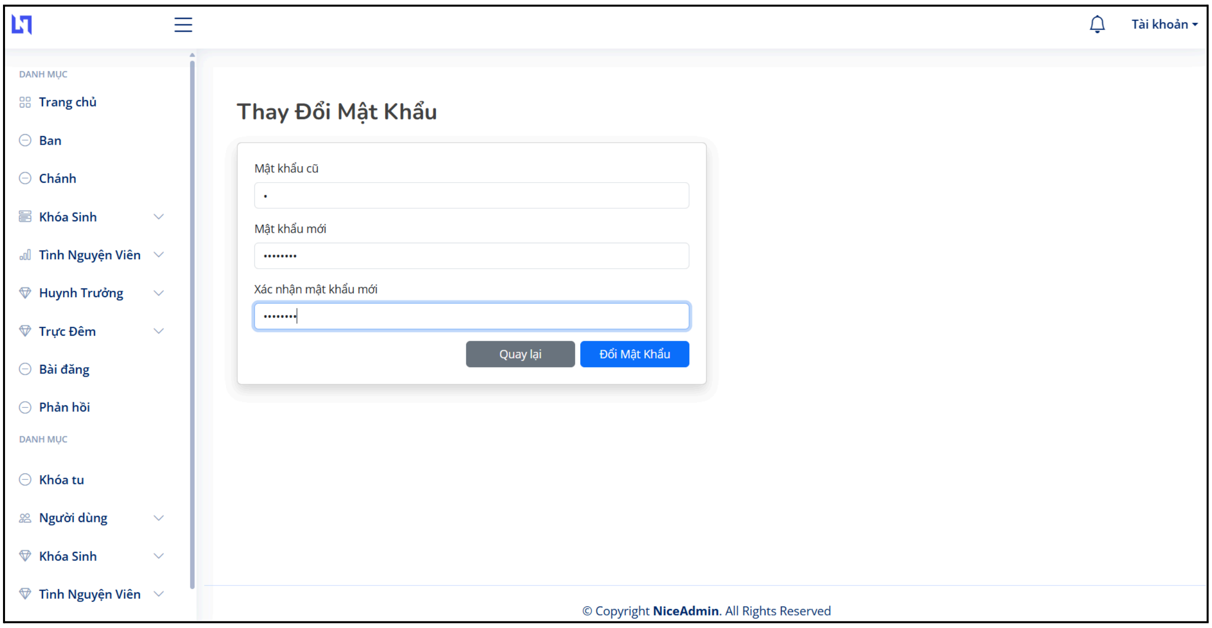


To log out, follow these steps:

1. The user logs into the system, clicks "Tài khoản" on the navigation bar and then has a dropdown that appears.
2. The user clicks the "Đăng xuất" button.

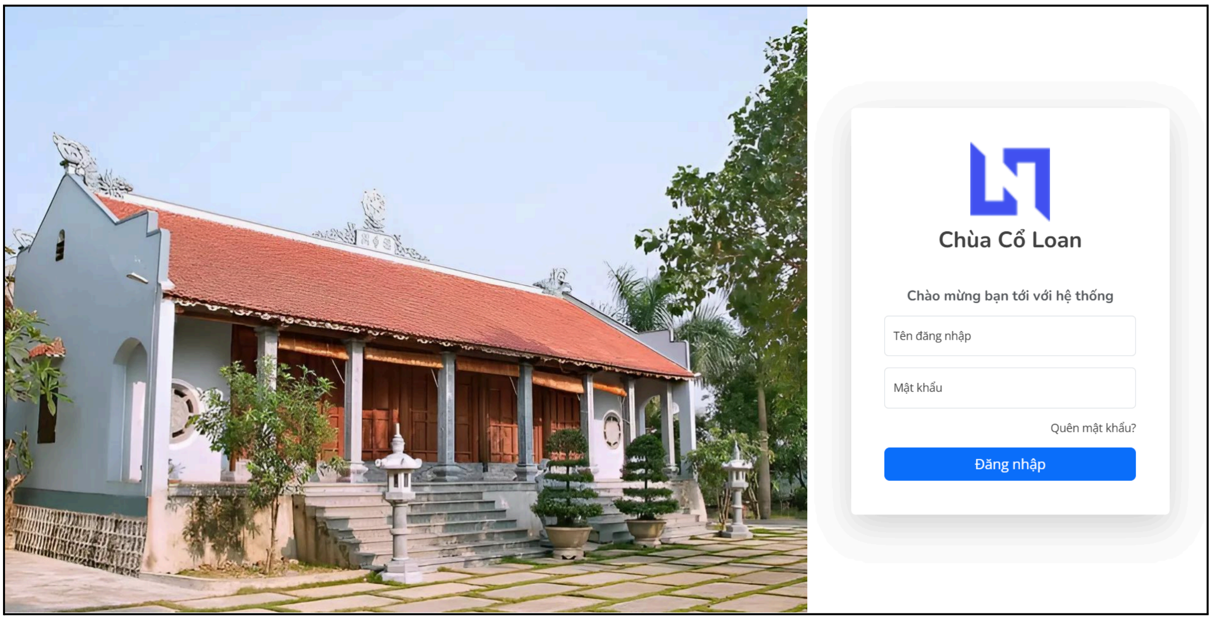
3.2.3. Change password

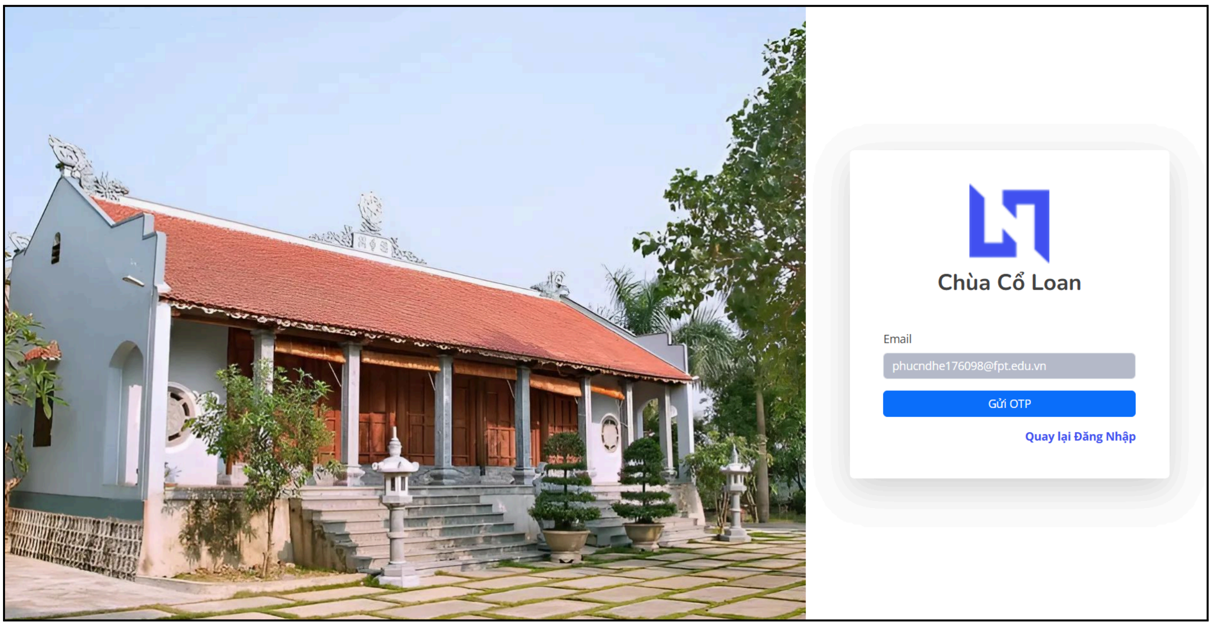


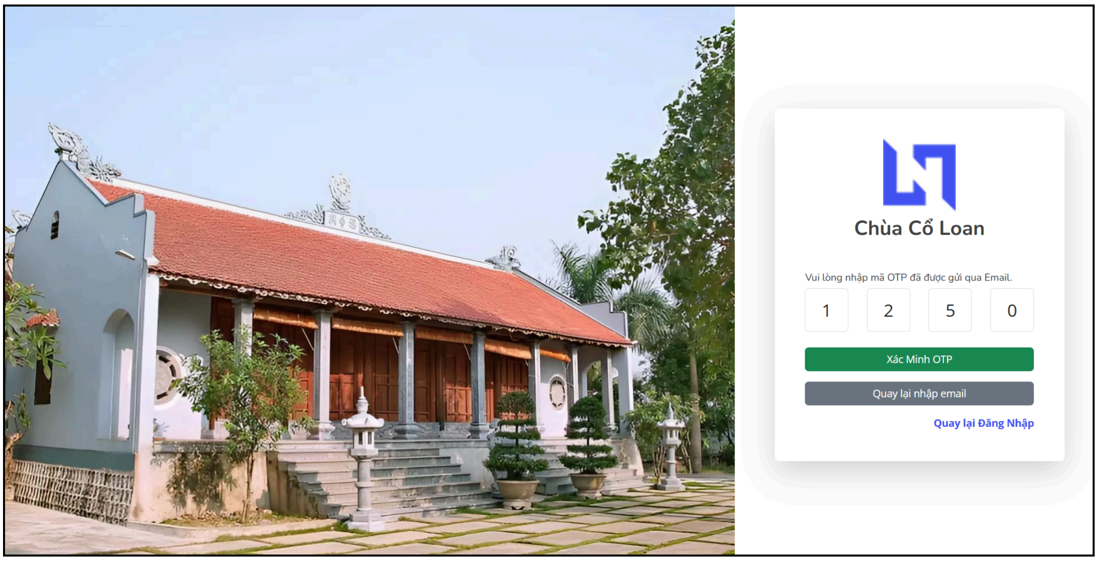


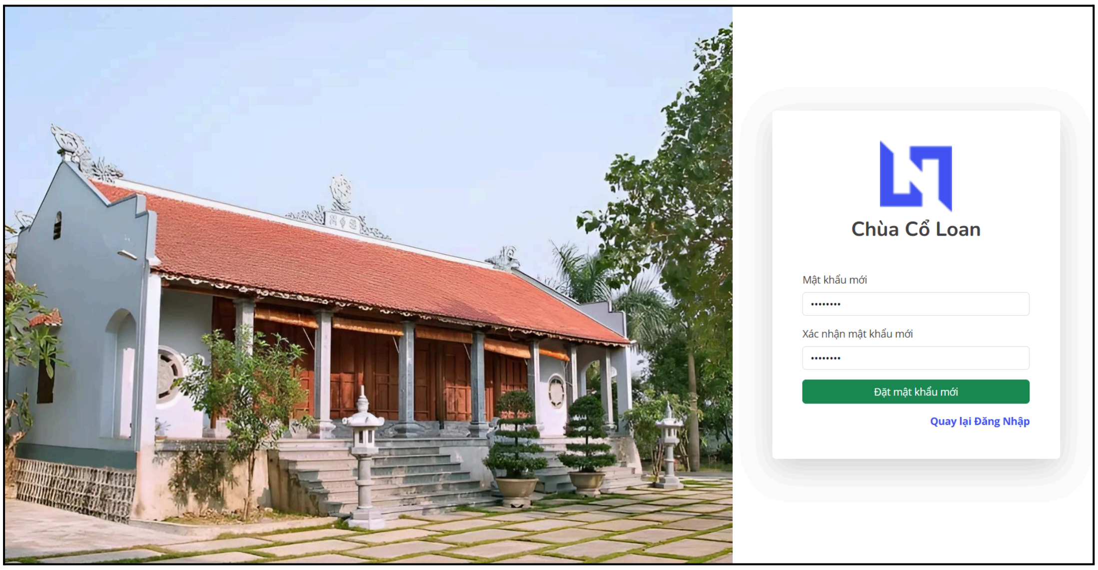
1. The user logs into the system, clicks "Tài khoản" on the navigation bar and then has a dropdown that appears. The user clicks the "Thay đổi mật khẩu" button.
2. User is redirected to Change Password Screen then enter Old Password and New Password to change the password.

3.2.4. Forgot password







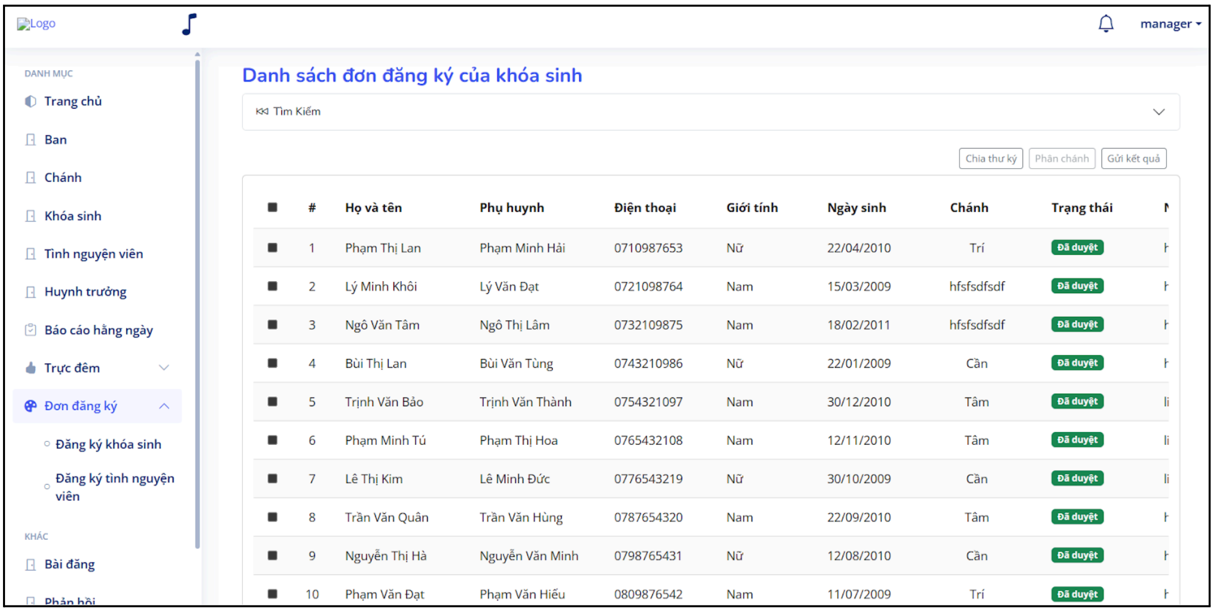


If users forgot the password, follow these steps:

1. User access into "Đăng nhập" Screen then click "Quên mật khẩu" option.
2. User is redirected to "Quên mật khẩu" Screen then enter email to receive an OTP to reset password.
3. Enter received OTP.
4. Enter a new password and confirm the new password.

3.3 Feature Student Registration Management

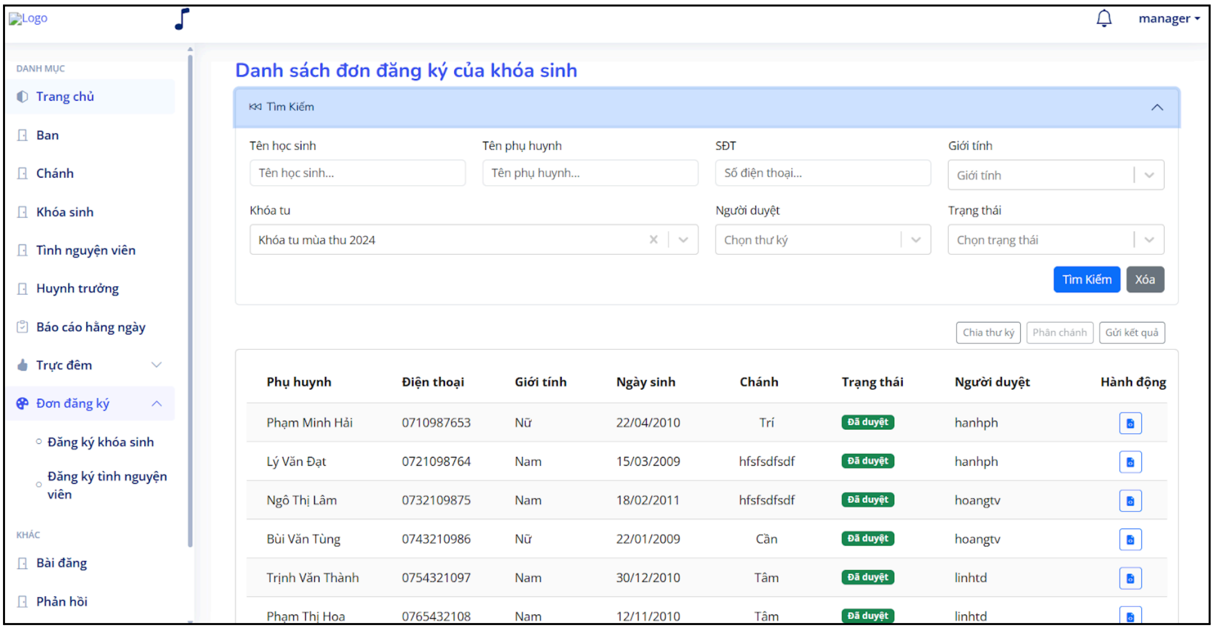
3.3.1 View all student registration feature



To view a list of all student registration, follow these steps:

1. Navigate to the student registration list interface through “Đăng ký khóa sinh” under “Đơn đăng ký” button .
2. The system will display all features with essential information such ”Họ và tên”, “Ngày sinh”, “Giới tính”,”Trạng thái”, “Số điện thoại”,“Người duyệt”, “Chánh”, “Phụ huynh”.

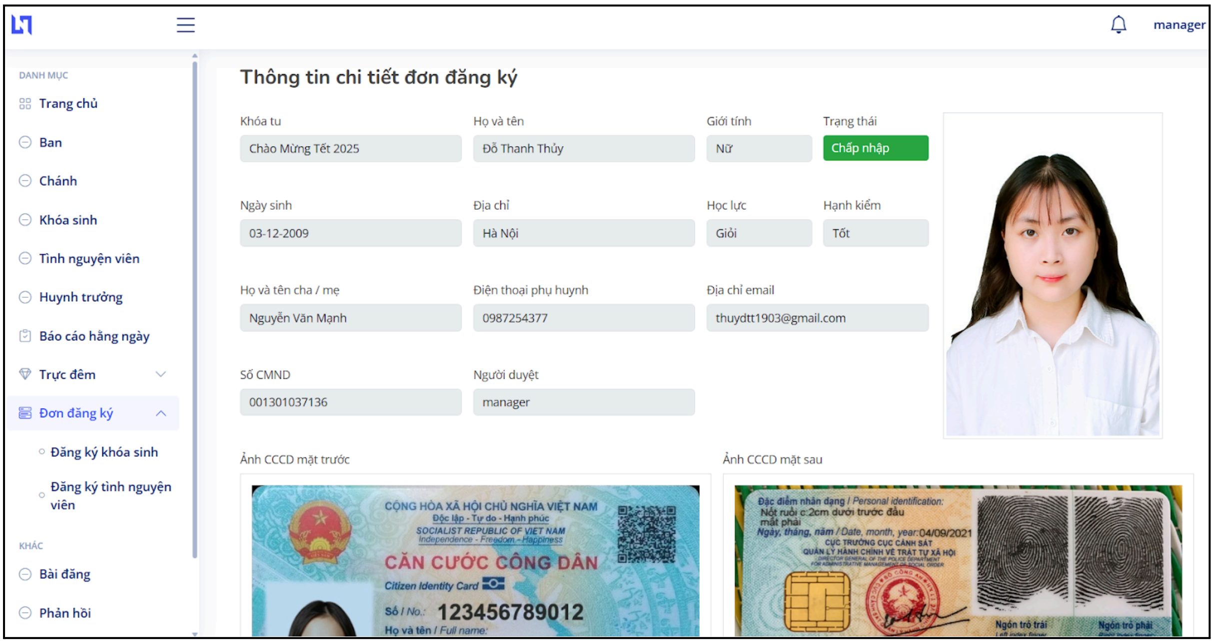
3.3.2 Search a student registration within list of student registration feature



To search for a student registration within a student registration list, follow these steps:

1. Select the fields that you want to search for a student registration.
2. Enter the search criteria (e.g., "Họ và tên", "Ngày sinh","Giới tính", 'Trạng thái", "Số điện thoại","Người duyệt", "Chánh", "Phụ huynh", etc.).
3. The system will search for student registrations that match the criteria within the student registration list and display the search results.

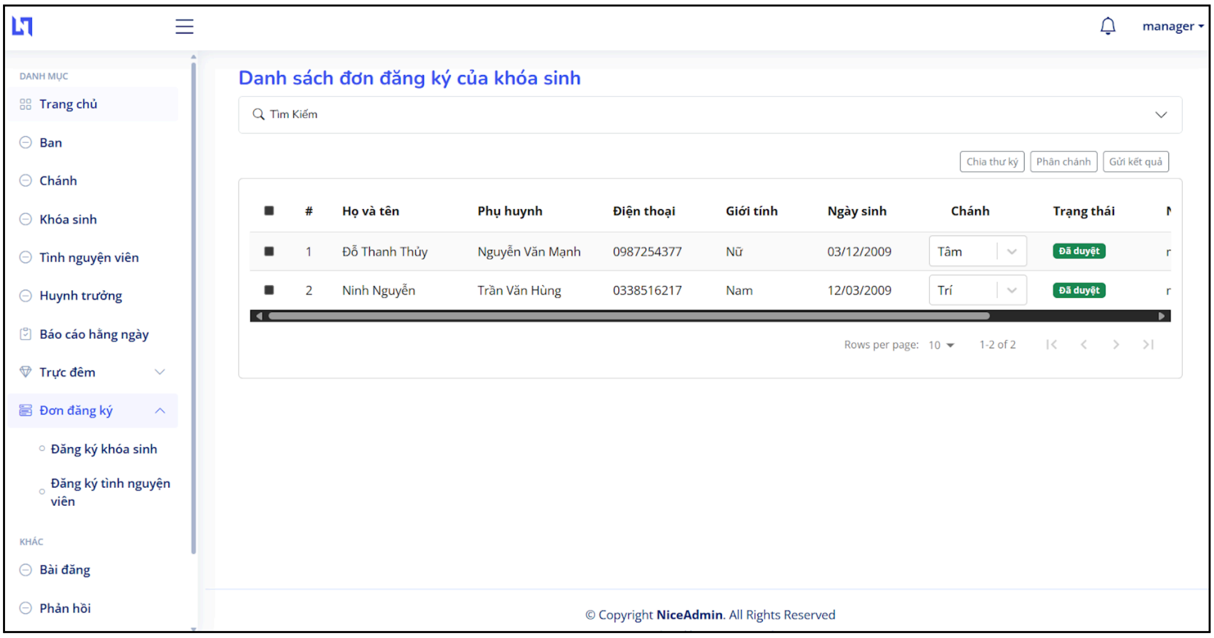
3.3.3 View student registration detail.

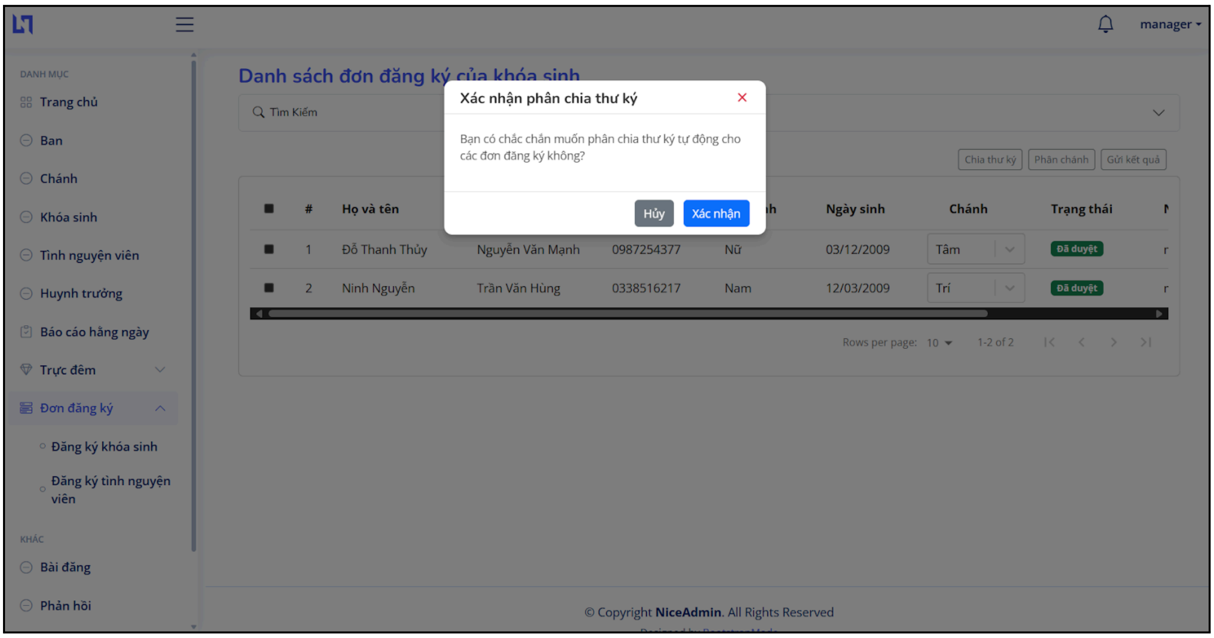


To view a student registration detail within a student registration list, follow these steps:

1. Navigate to the student registration management interface.
2. Select a student registration and click on a student to view their registration details.

3.3.4 Auto assign student registrations for secretary.



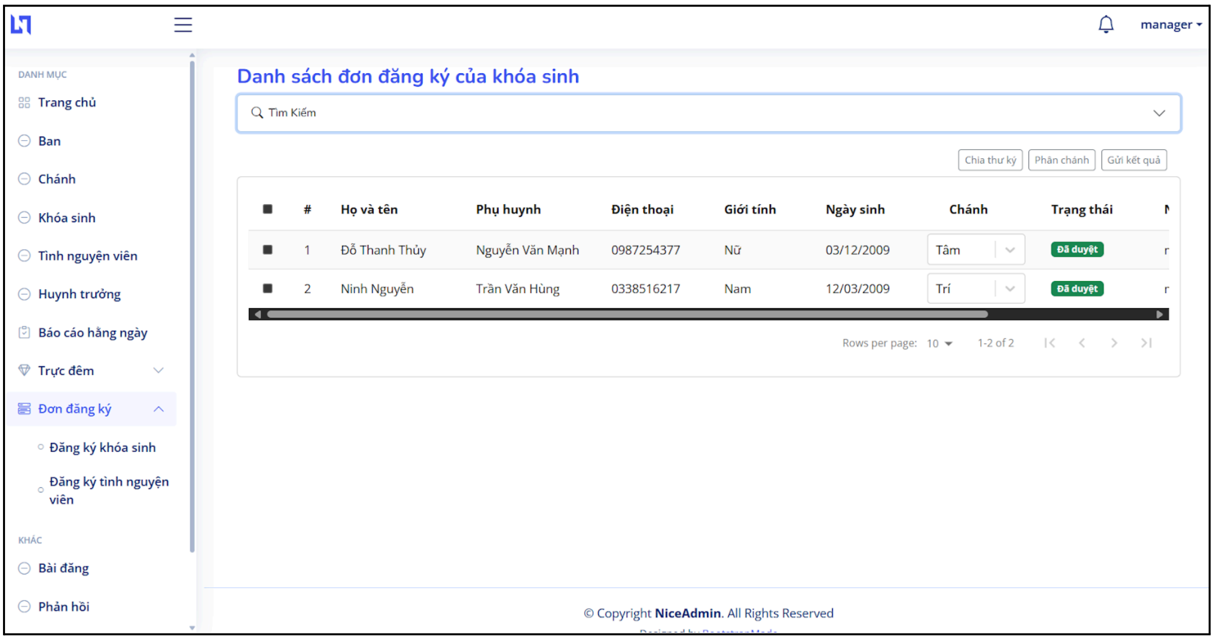


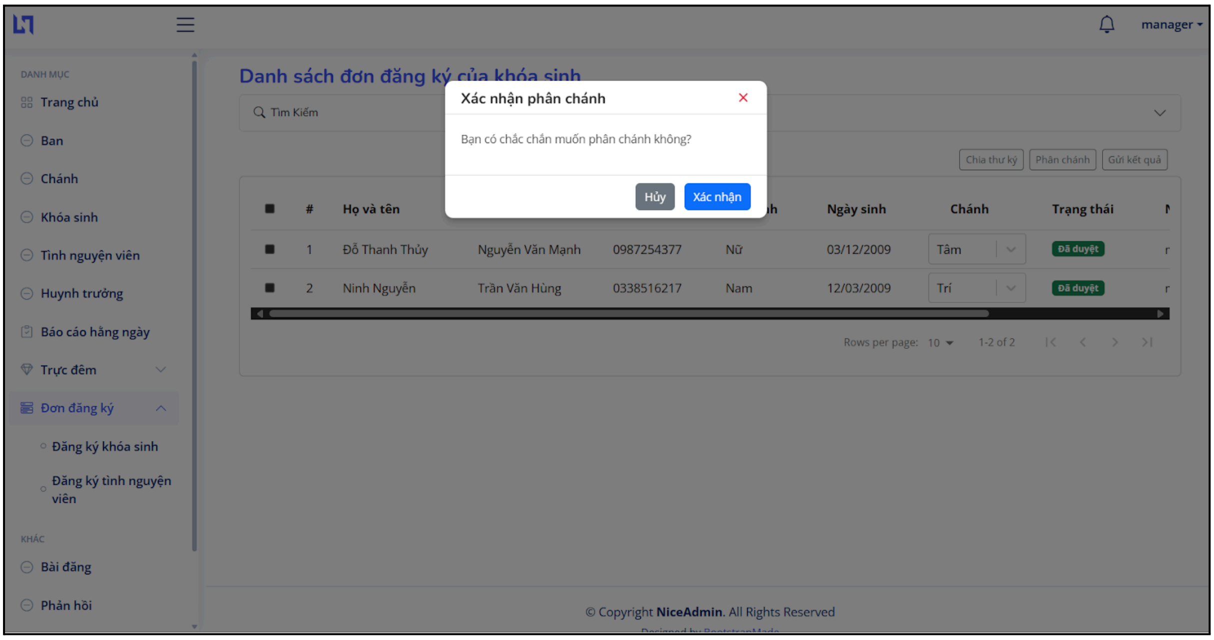
To auto assign student registrations for secretary, follow these steps:

1. Users access the student registration list page through the "Đơn đăng kí khóa sinh" button in the dashboard area on the left side of the page.
2. Users select the "Chia thư ký" button.
3. The system will once again confirm whether users want to assign the secretaries or not.

4. If the user clicks on the "Xác nhận" button the system updates the secretaries assignment status for each registration.

3.3.5 Automatically assign students for student groups.



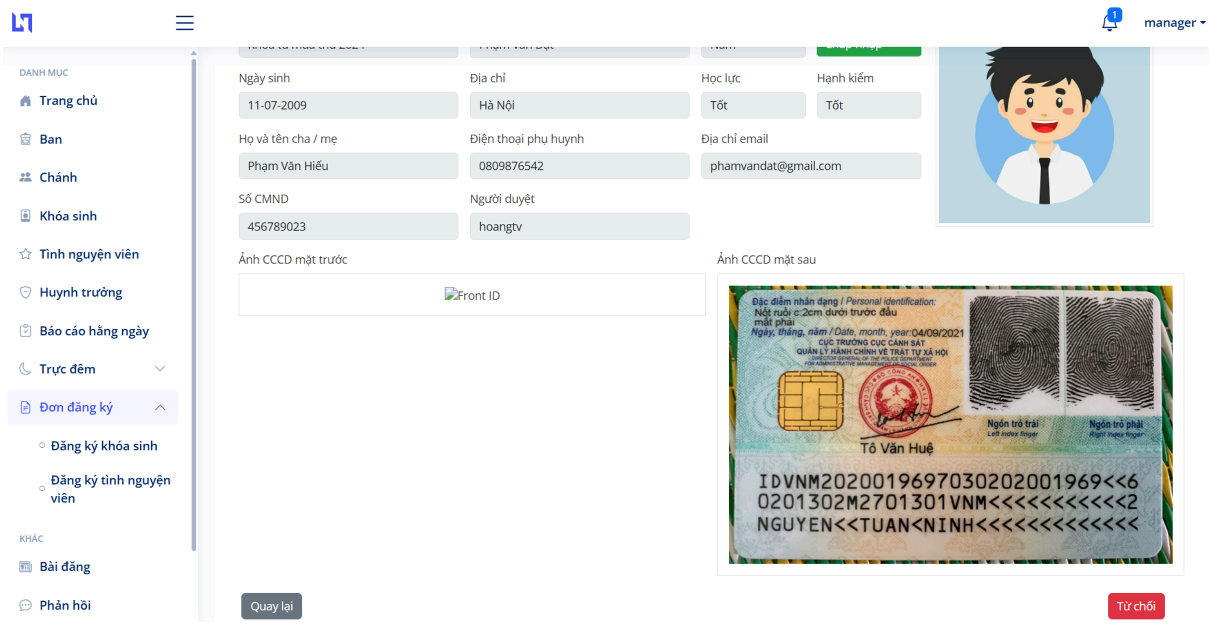


To automatically assign students for student groups, follow these steps:

1. Users access this page through the "Đơn đăng ký khóa sinh" button under "Đơn đăng ký" in the dashboard area on the left side of the page.

2. Users select the “Phân chánh” button.
3. The system will once again confirm whether users want to assign the students to student groups or not.
4. If the user clicks on the “Xác nhận” button the system updates the students assignment status for each student group.

3.3.6 Accept/Reject Student Registration.

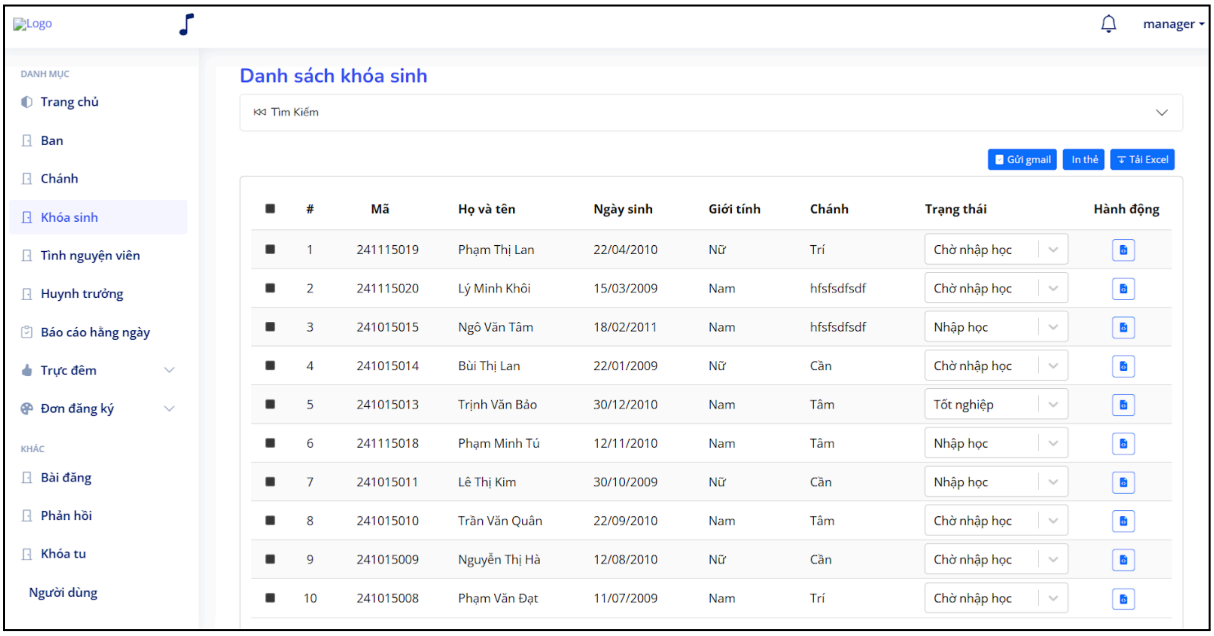


To accept/reject volunteer registration, follow these steps:

1. View student registration detail: do the same as: [3.8.3 View student registration details]
2. The user accept or reject the registration by clicking “Duyệt” or “Từ chối” button at the bottom of the form.

3.4 Feature Student Management

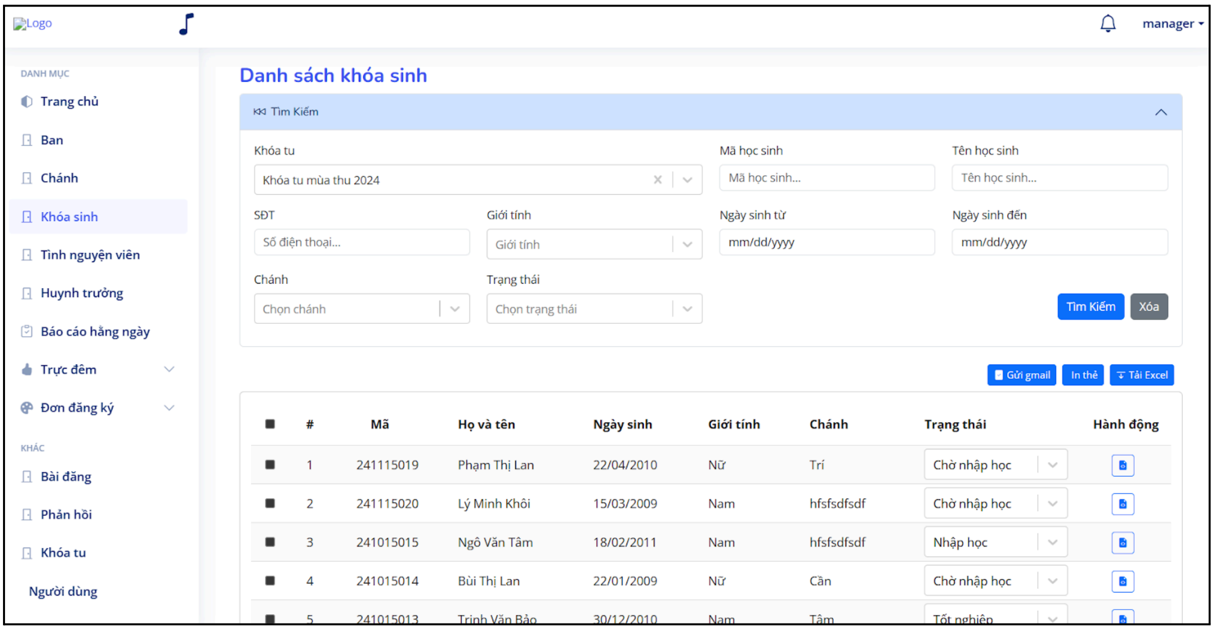
3.4.1 View list of student feature



To view a list of student, follow these steps:

1. Navigate to the student list interface through the “Khóa sinh” button on the sidebar.
2. The system will display all features with essential information such as ”Mã học sinh”, “Họ và tên”, “Ngày sinh”, “Giới tính”, “Trạng thái”.

3.4.2 Search a student within list of student feature



To search for a student task within a project, follow these steps:

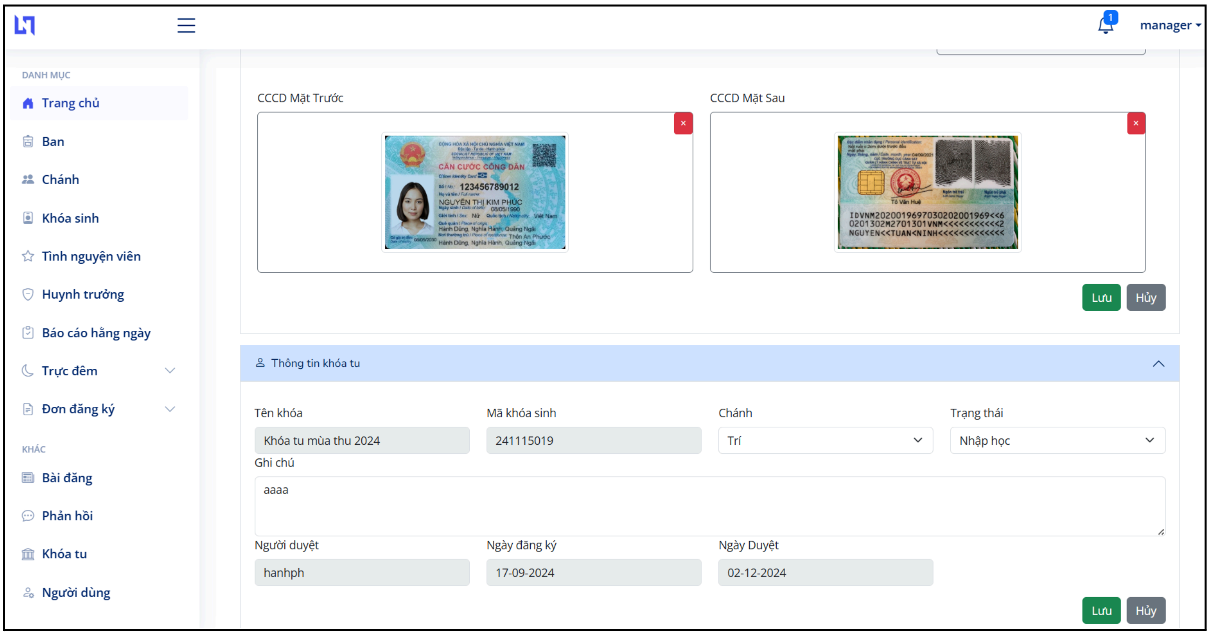
1. Select the fields that you want to search for a student.
2. Enter the search criteria (e.g., “Mã học sinh”, “Họ và tên”, “Ngày sinh”, “Giới tính”, etc.).
3. The system will search for students that match the criteria within the student list and display the search results.

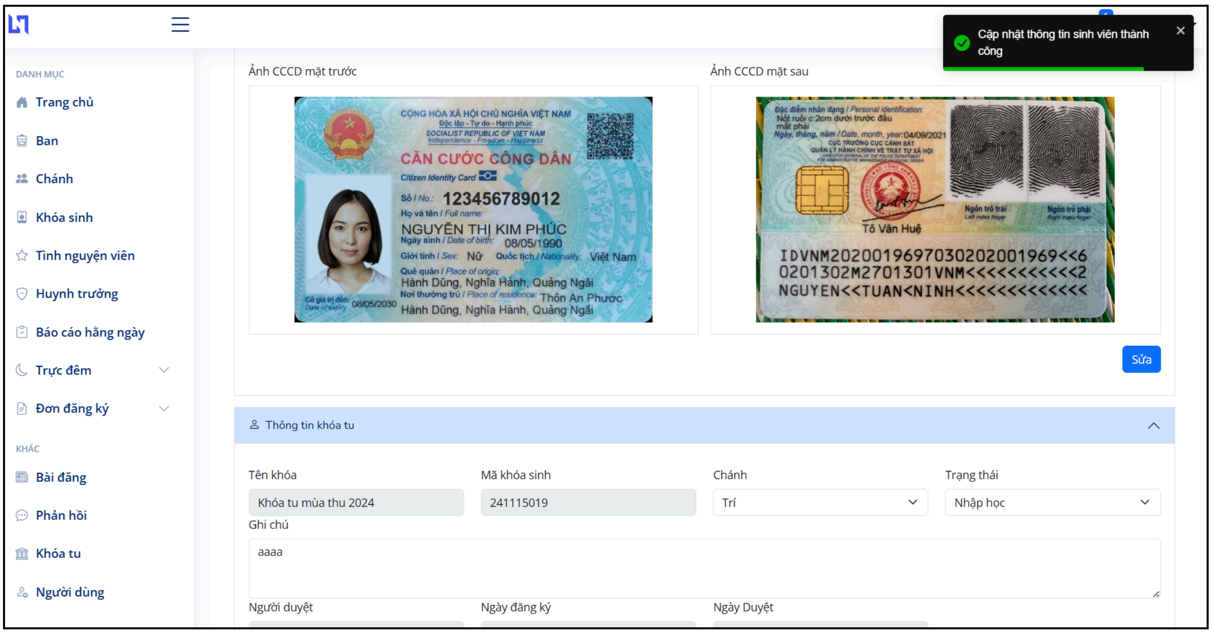
3.4.3 View student detail

To view student details, follow these steps:

1. Navigate to the student management interface.
2. Select a student and click on a student to view their details.

3.4.3 Edit student information

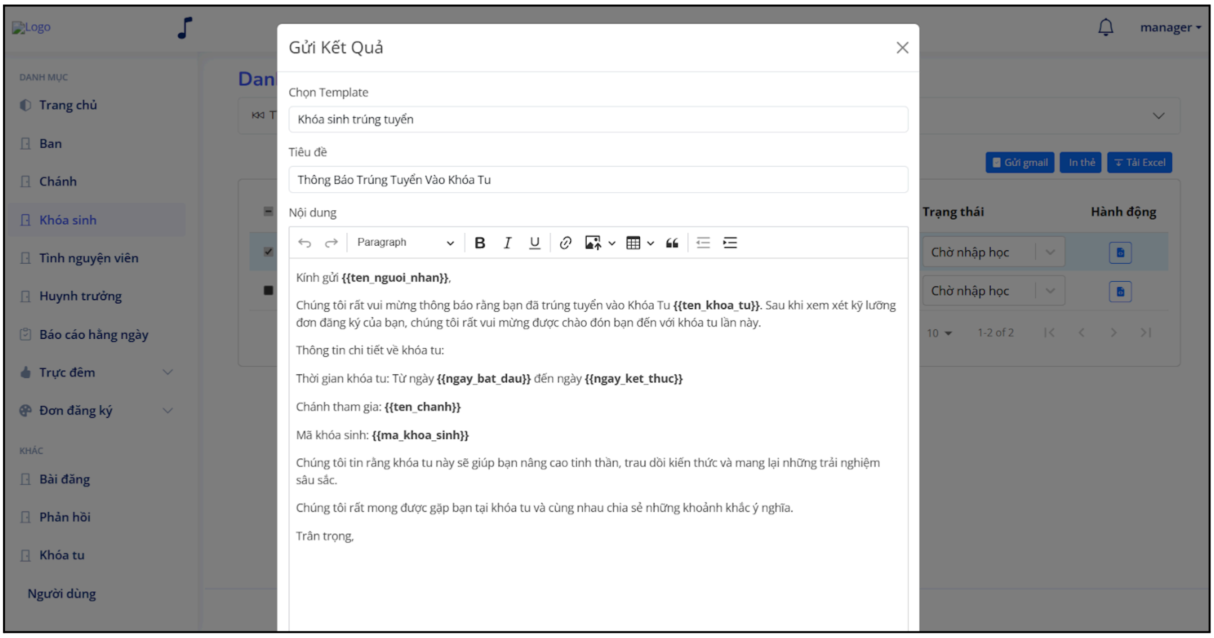




To edit student information, follow these steps:

1. View student detail: do the same as [3.6.3 View student detail](#).
2. The user can edit student information and student’s course information by clicking “sửa” button under the student’s information form.
3. The user updates the student information and clicks the “Lưu” button to save the information.

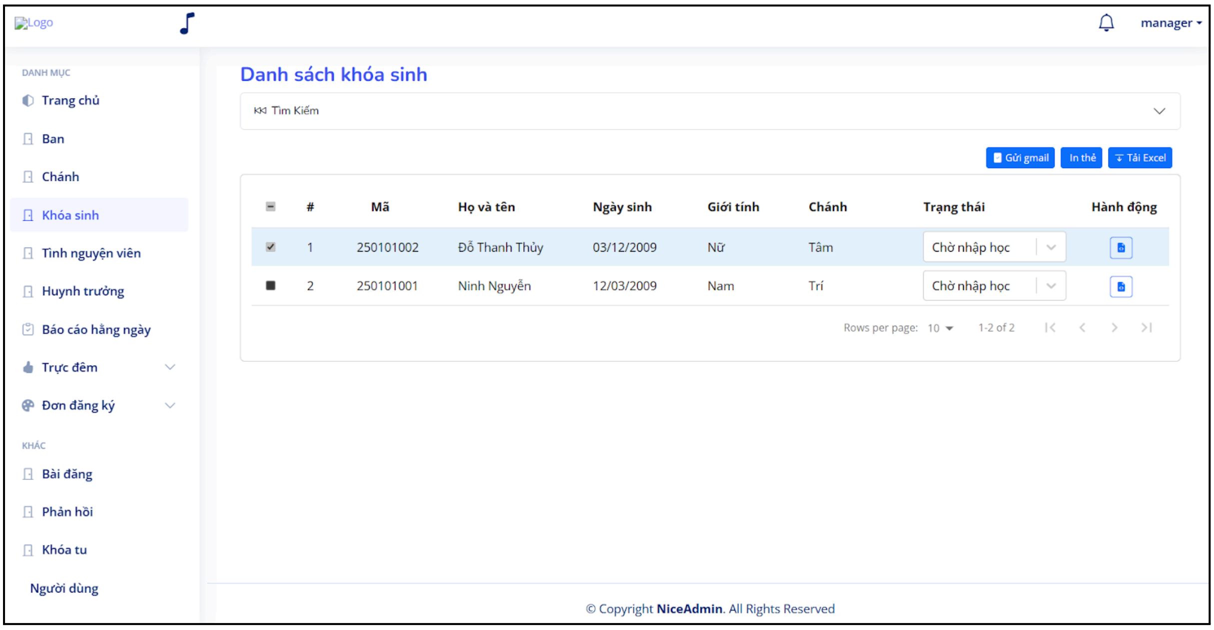
3.4.4 Send an email to students in student list



To send an email to students in student list, follow these steps:

1. Navigate to the student management interface.
2. Click the “Gửi mail” button.
3. Choose the email template that you want to send to the student in student list and click the “Gửi” button to confirm your option.

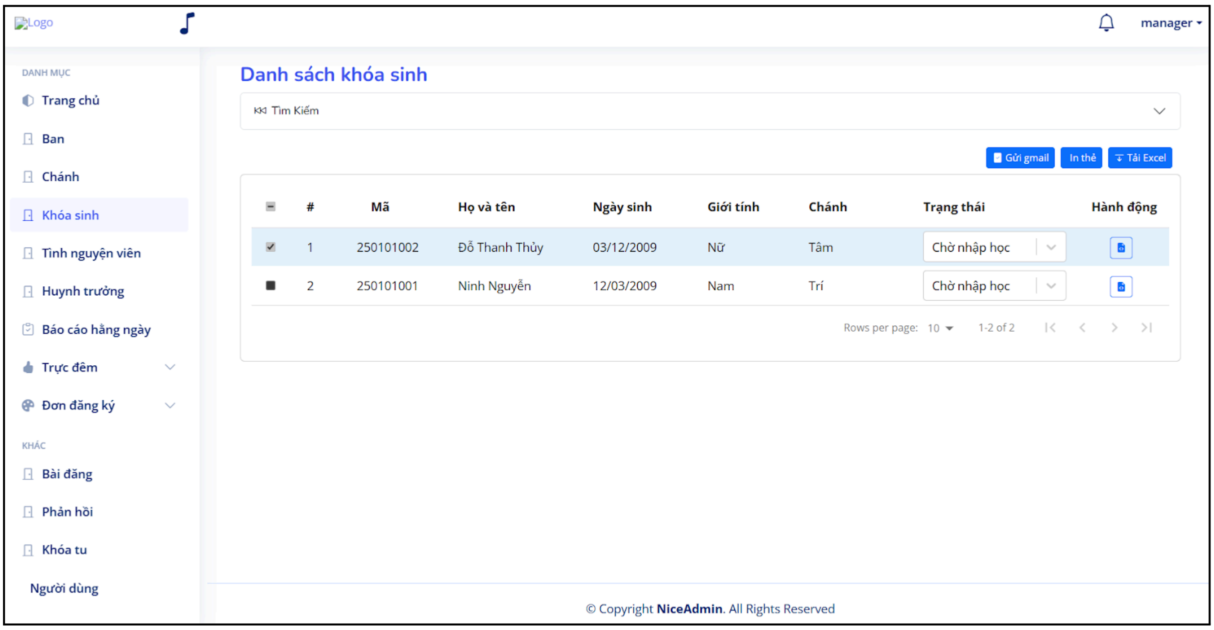
3.4.5 Auto generate student cards of students in student list



To send an email to students in student group, follow these steps:

1. Navigate to the student management interface.
2. In the student management interface click the button “In thẻ”.
3. The system auto generates the student cards of the students in each corresponding student.

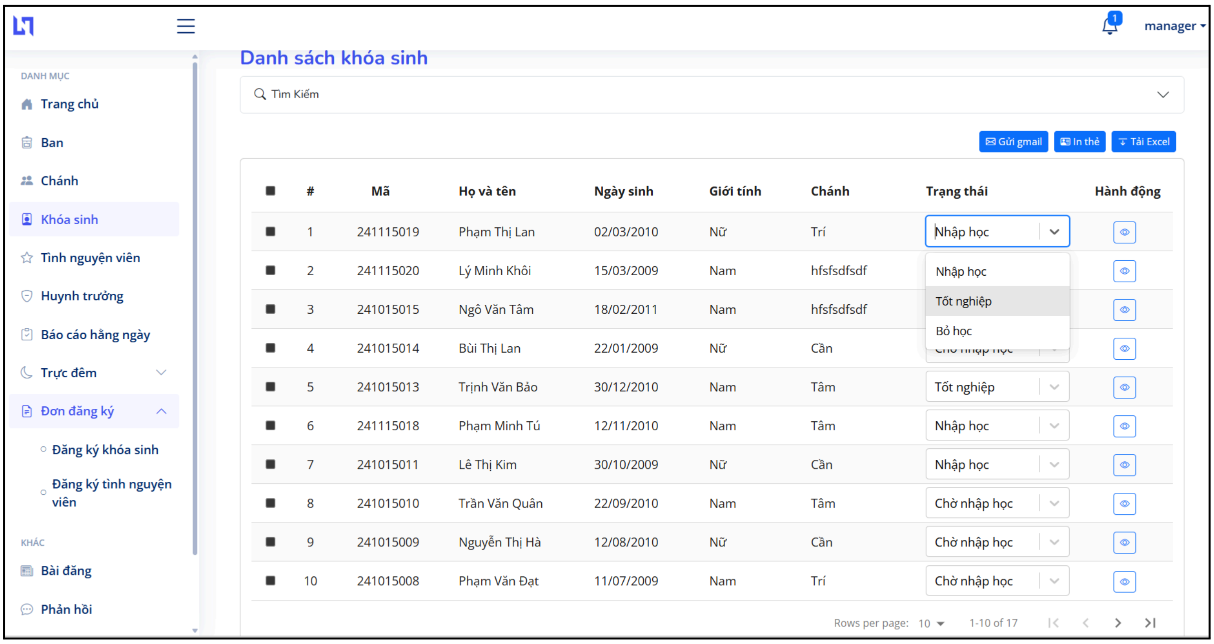
3.4.6 Download excel file list of students in student list



To send an email to students in student group, follow these steps:

1. Navigate to the student group management interface.
2. Select the option to view the student group in detail.
3. In the student group management interface click the button “Xuất excel”.
4. The system downloads the student excel file of the student list in each corresponding group.

3.4.7 Update Student Status

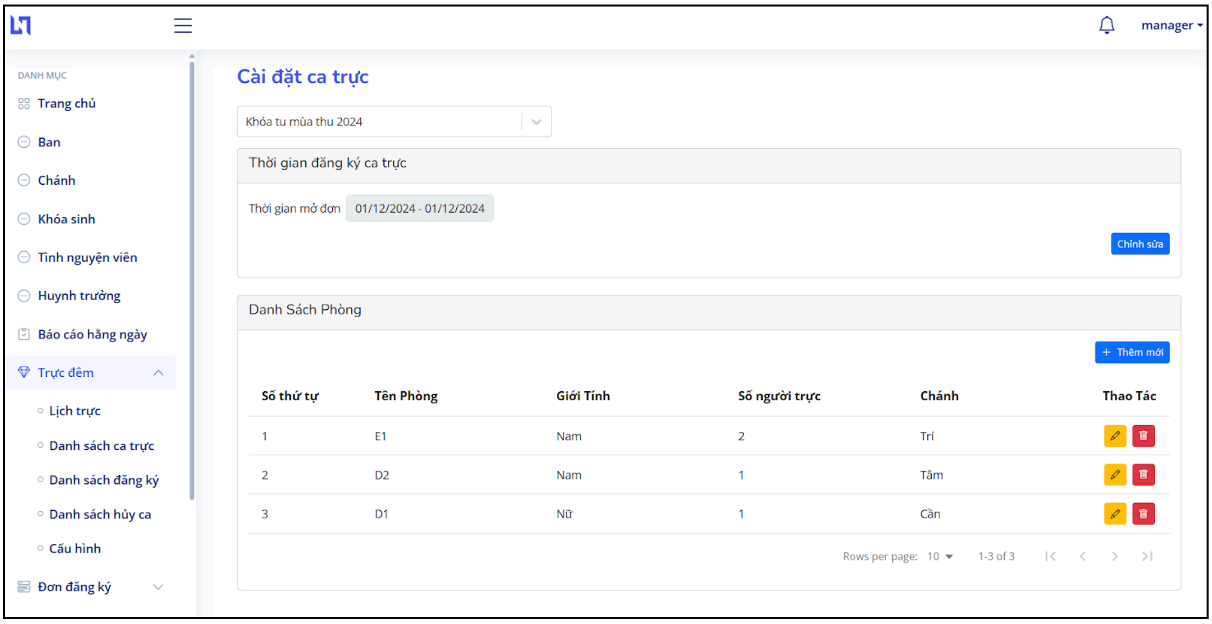


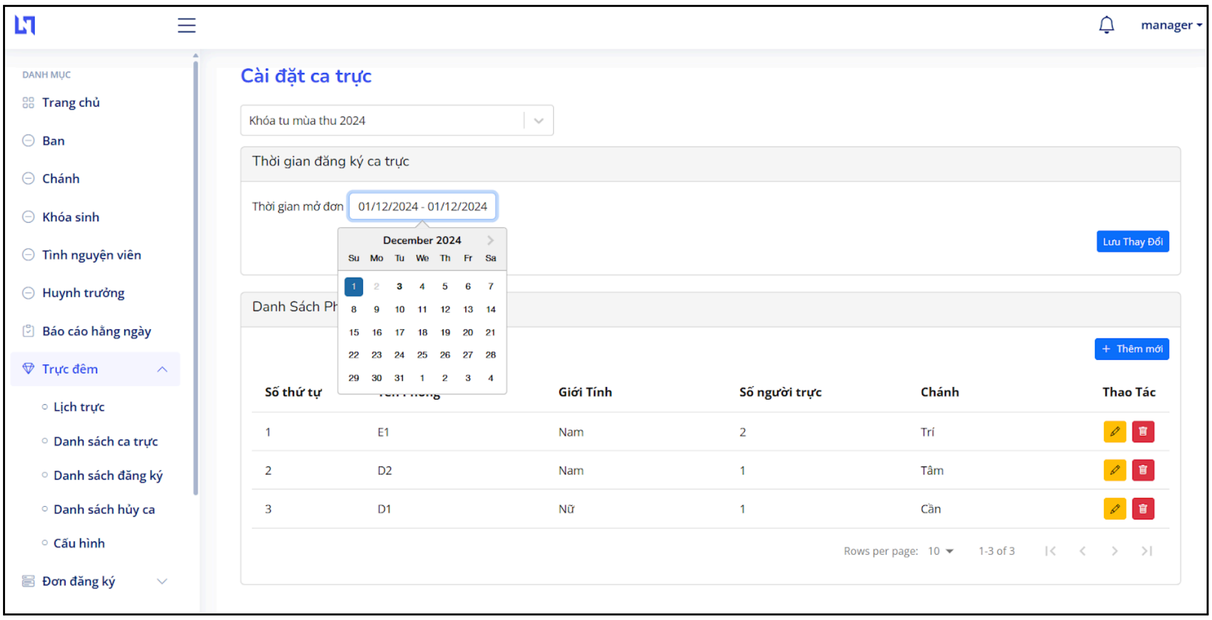
To update student status, follow these steps:

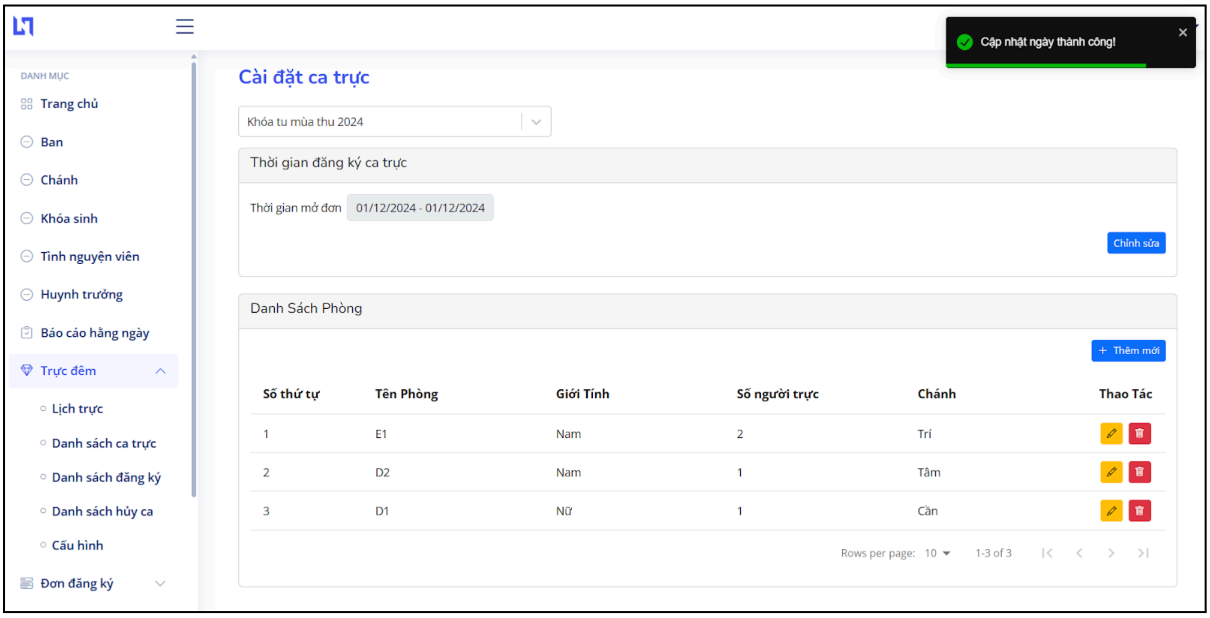
1. Navigate to the student management interface.
2. Select the status to update for students at the “Trạng thái” column.

3.5 Night Shift Management

3.5.1 Config open registration times for night shifts


manager
DANH MỤC
Trang chủ
Ban
Chánh
Khóa sinh
Tình nguyện viên
Huynh trưởng
Báo cáo hằng ngày
Trực đêm
Lịch trực
Danh sách ca trực
Danh sách đăng ký
Danh sách hủy ca
Cấu hình
Đơn đăng ký
Cài đặt ca trực
Khóa tu mùa thu 2024
Thời gian đăng ký ca trực
Thời gian mở đơn
01/12/2024 - 01/12/2024
Chỉnh sửa
Danh Sách Phòng
+ Thêm mới
Số thứ tự
Tên Phòng
Giới Tính
Số người trực
Chánh
Thao Tác
1
E1
Nam
2
Trí
2
D2
Nam
1
Tâm
3
D1
Nữ
1
Cần
Rows per page: 10
1-3 of 3


manager
DANH MỤC
Trang chủ
Ban
Chánh
Khóa sinh
Tình nguyện viên
Huynh trưởng
Báo cáo hằng ngày
Trực đêm
Lịch trực
Danh sách ca trực
Danh sách đăng ký
Danh sách hủy ca
Cấu hình
Đơn đăng ký
Cài đặt ca trực
Khóa tu mùa thu 2024
Thời gian đăng ký ca trực
Thời gian mở đơn
01/12/2024 - 01/12/2024
December 2024
Su Mo Tu We Th Fr Sa
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31 1 2 3 4
Lưu Thay Đổi
+ Thêm mới
Số thứ tự
Giới Tính
Số người trực
Chánh
Thao Tác
1
E1
Nam
2
Trí
2
D2
Nam
1
Tâm
3
D1
Nữ
1
Cần
Rows per page: 10
1-3 of 3



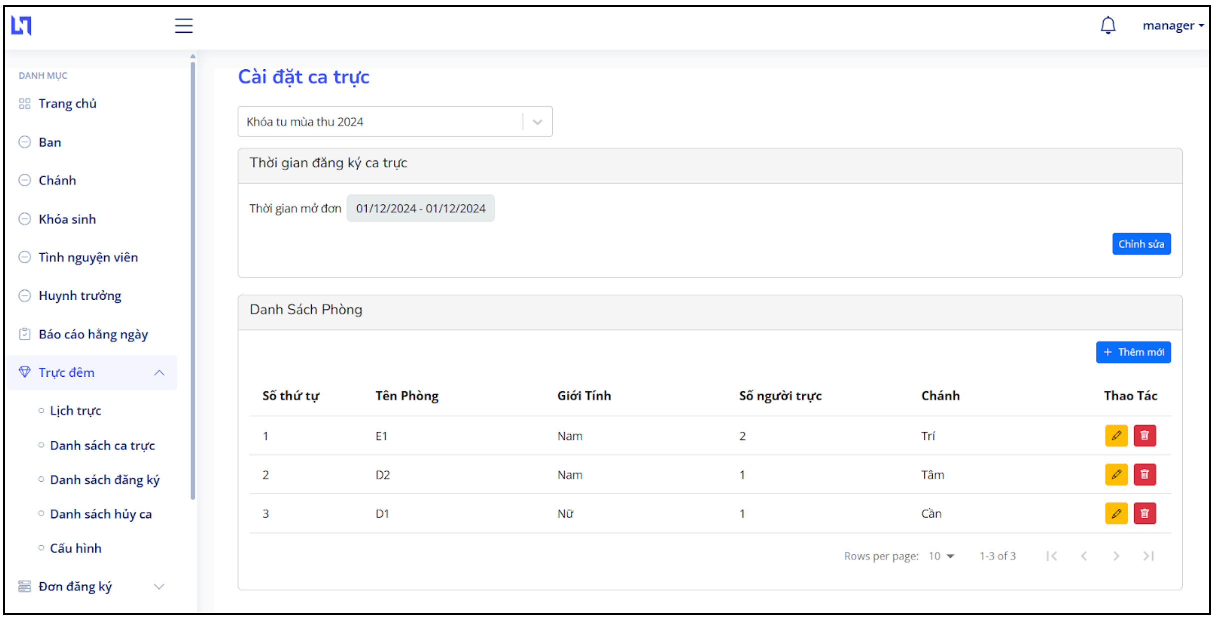
To config open registration times for night shifts, follow these steps:

1. The user logs in with a manager account and selects the "Cấu hình" under the "Trực đêm" button at the sidebar.
2. The user can config open registration times for night shifts by clicking the "Chỉnh sửa" button of "Thời gian đăng ký ca trực" form.
3. Then choose the time user want to change and click "Lưu thay đổi" button

3.5.2 Config room for night shifts

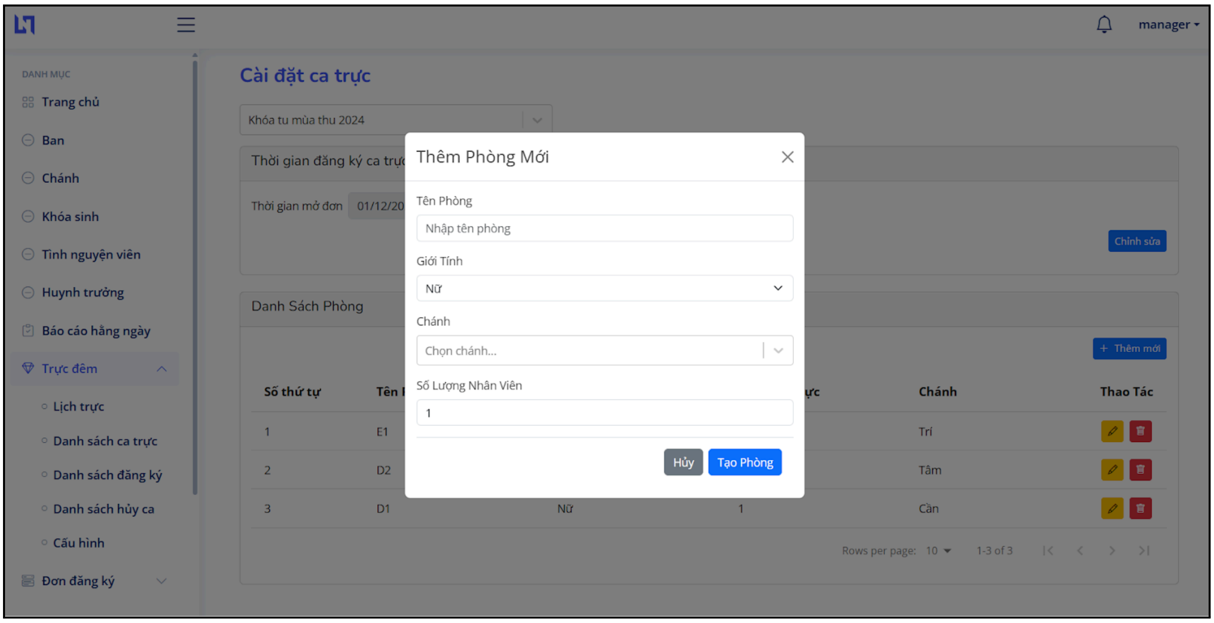
To config room for night shifts, follow these steps:

1. The user logs in with a manager account and selects the "Cấu hình" under the "Trực đêm" button at the sidebar.

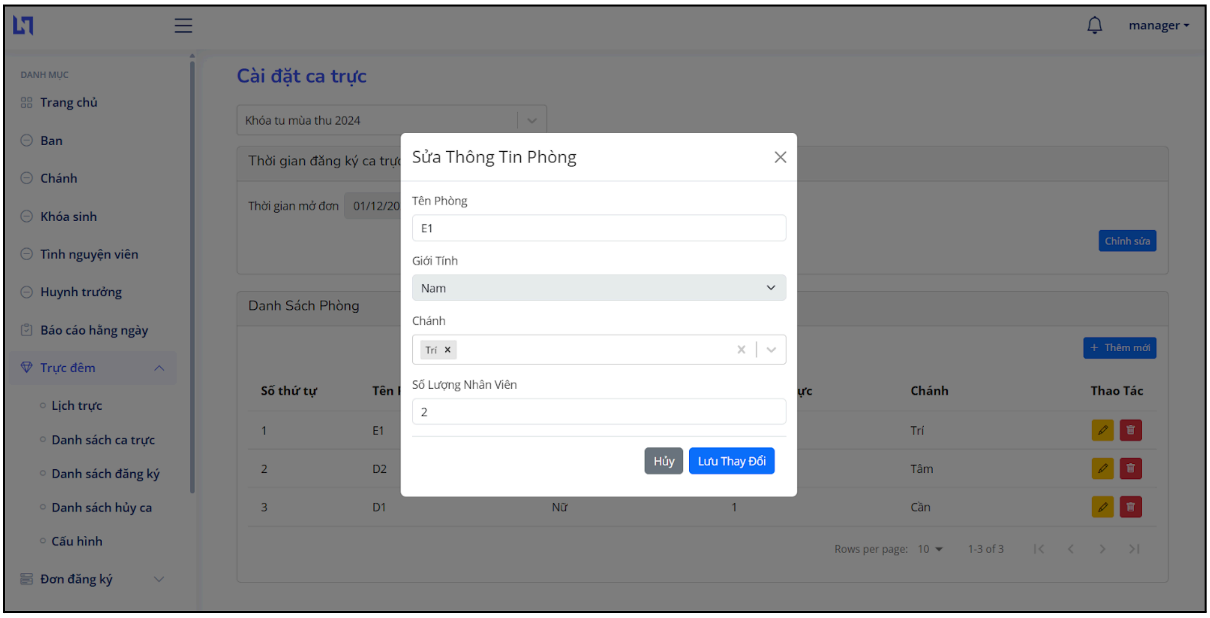


2. The user can configure rooms for night shifts by choosing the option below in "Danh sách phòng" form.

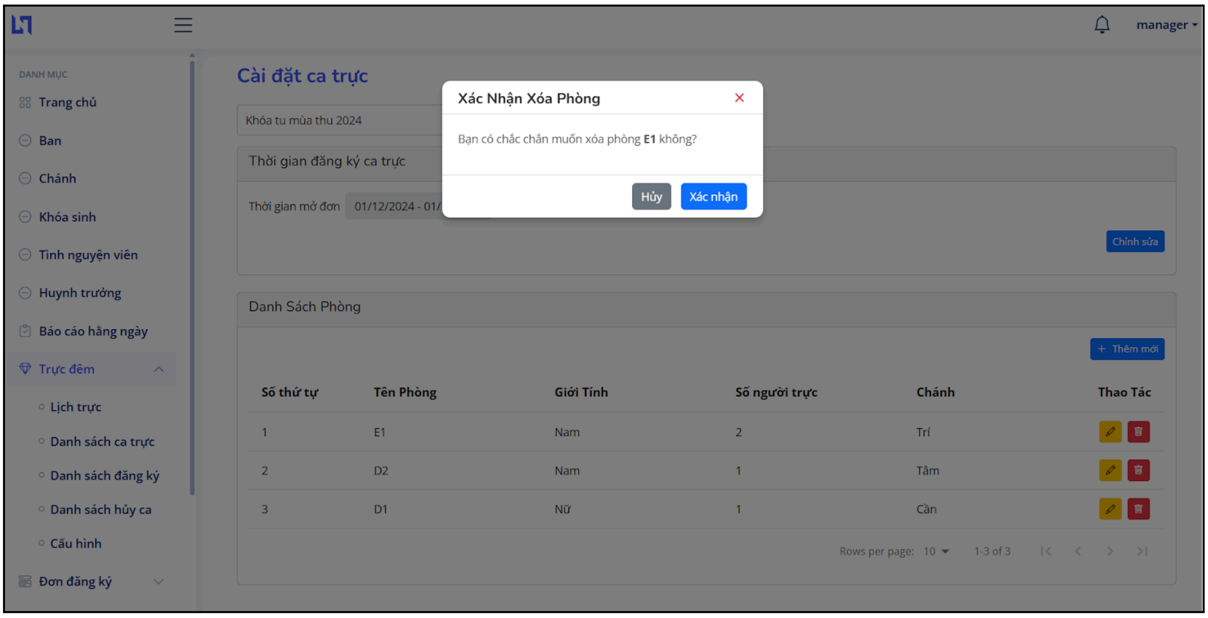
   - "Thêm" button to add a new room.



   - Pen icon at "Thao tác" column to edit the room's information.



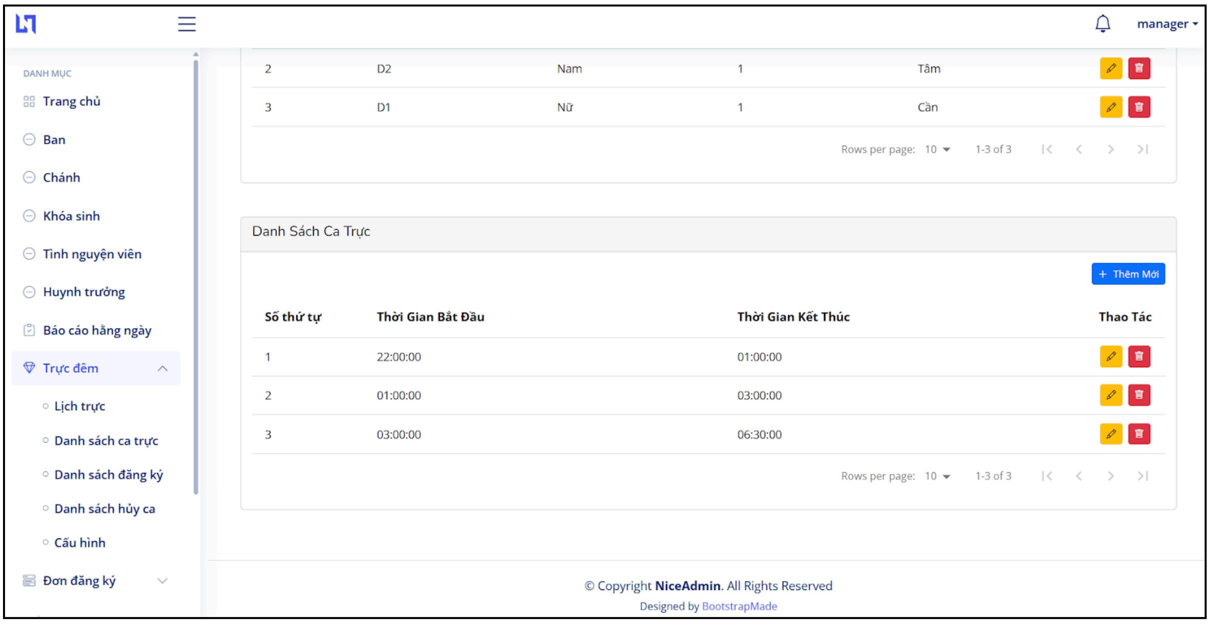
- Bin icon at "Thao tác" column to delete a room.



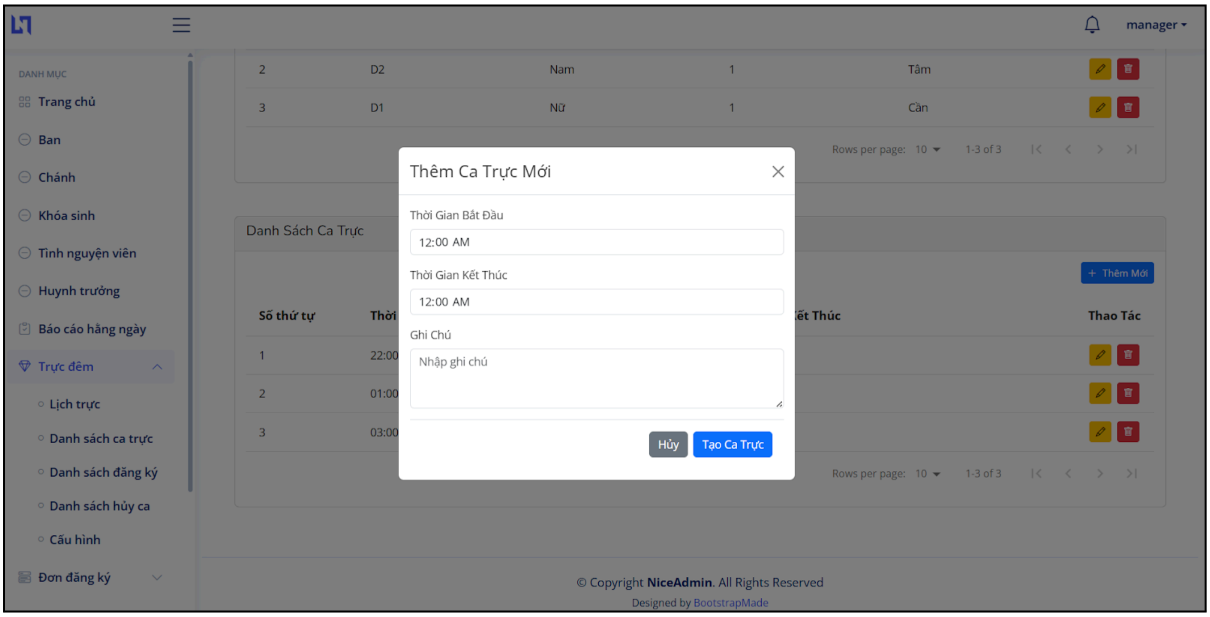
3.5.3 Config night shifts.

To config night shifts, follow these steps:

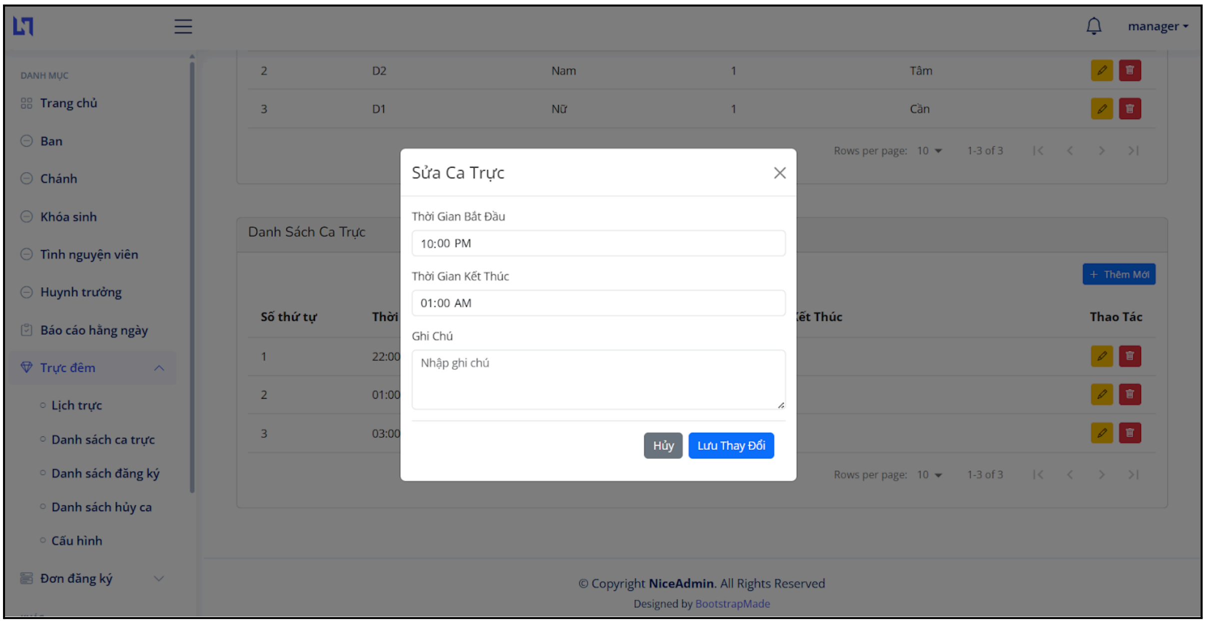
1. The user logs in with a manager account and selects the "Cấu hình" under the "Trực đêm" button at the sidebar.



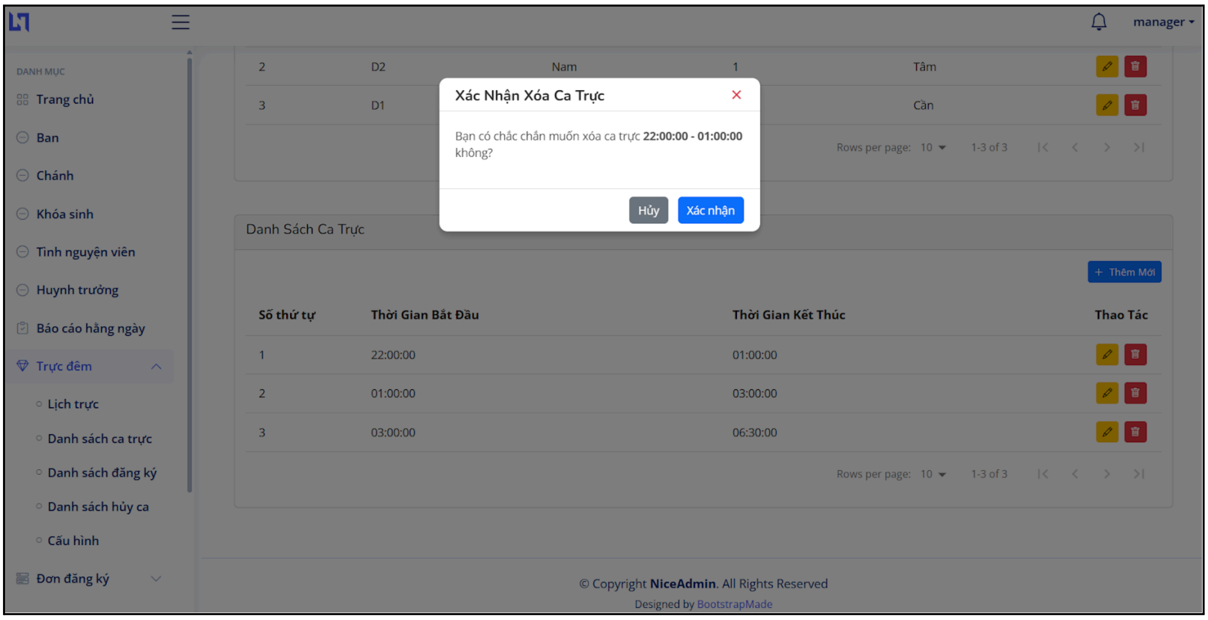
3. The user can configure night shifts by choosing the option below in "Danh sách ca trực" form.
    - "Thêm mới" button to add a new night shift.



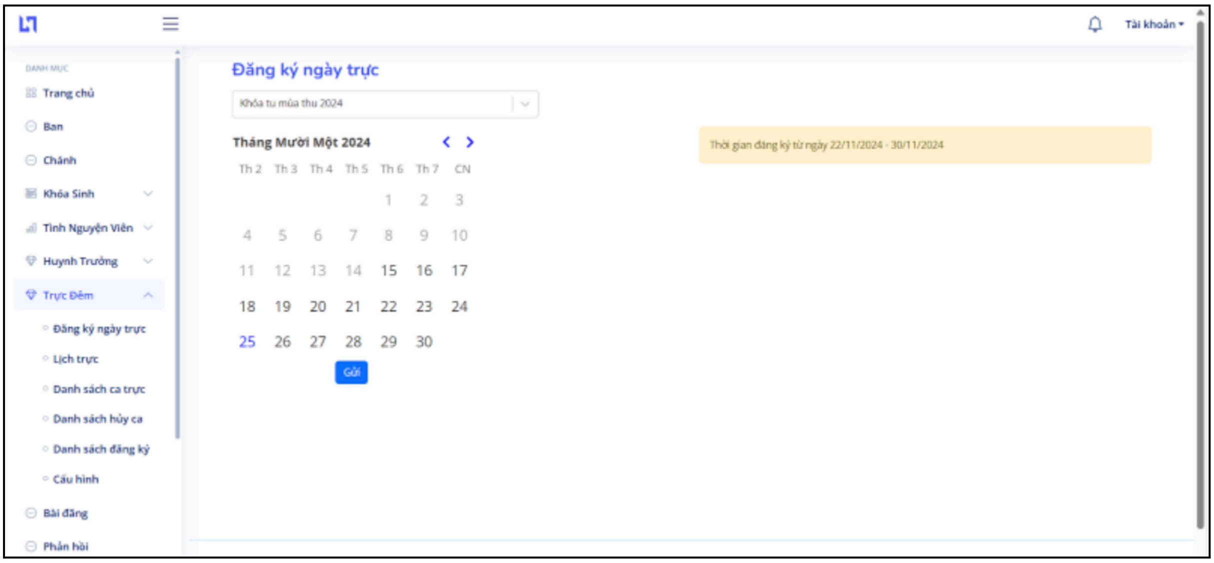
    - Pen icon at "Thao tác" column to edit the night shift information.



- Bin icon at "Thao tác" column to delete a night shift.



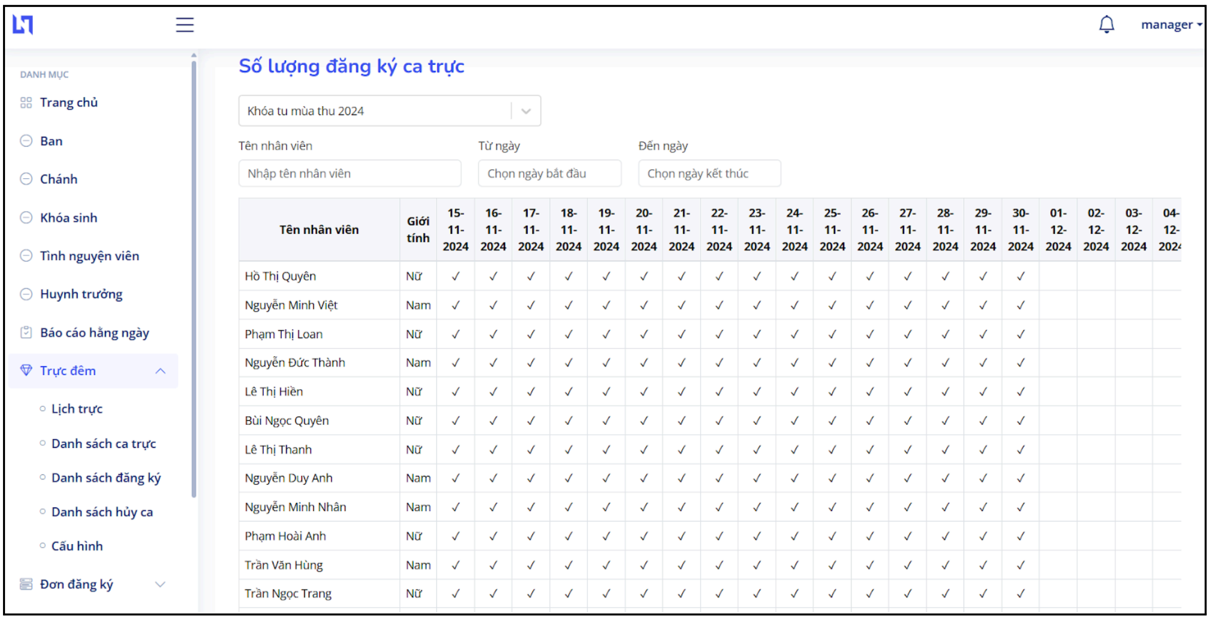
3.5.4 Register free time for night shifts



To register free time for night shifts, follow these steps:

1. The user logs in with a supervisor account and selects the "Đăng kí ngày trực" under the "Trực đêm" button at the sidebar.
2. The system displays the calendar and the supervisor chooses the time that they are free for a night shift and submit them (they can filter it by course).

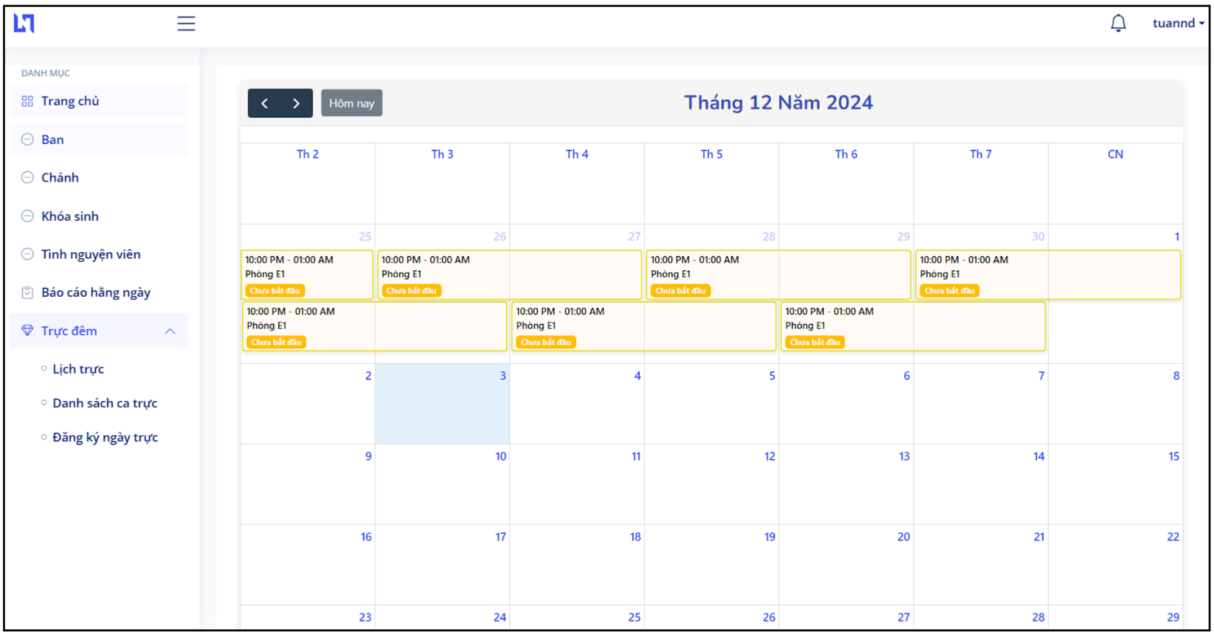
3.5.5 View the number of shift registrations for night shifts



To view the number of shift registrations for night shifts, follow these steps:

1. The user logs in with a supervisor account and selects the “Danh sách đăng kí” under the “Trực đêm” button at the sidebar.
2. The system displays the list of the number of shift registrations for night shifts (user can filter by supervisor’s name, time span and course).

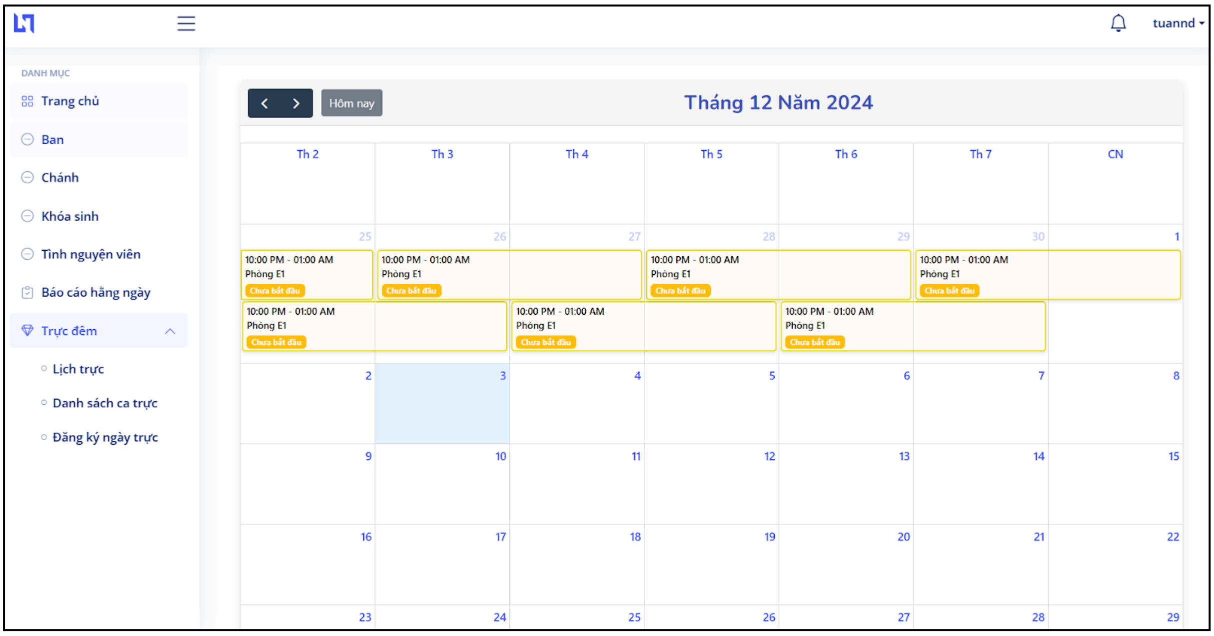
3.5.6 View user’s night shift.

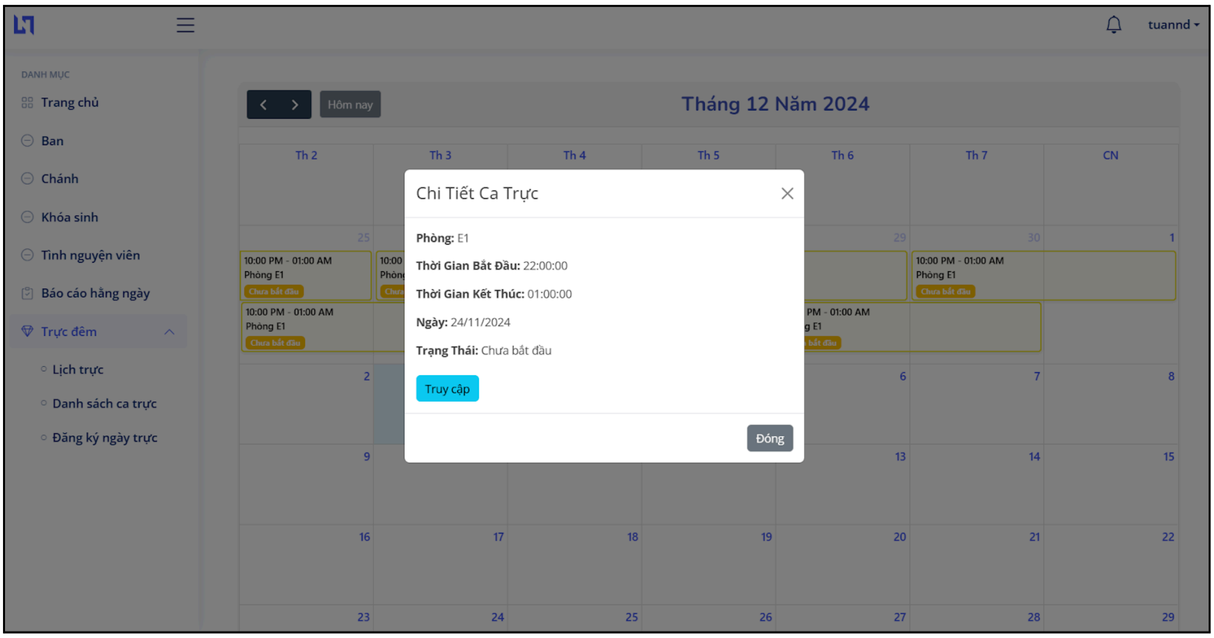


To view night shifts, follow these steps:

1. The user logs in with a manager account and selects the “Lịch trực” under the “Trực đêm” button at the sidebar.
2. The system will display the user’s night shift in calendar format.

3.5.7 View user’s night shift detail.

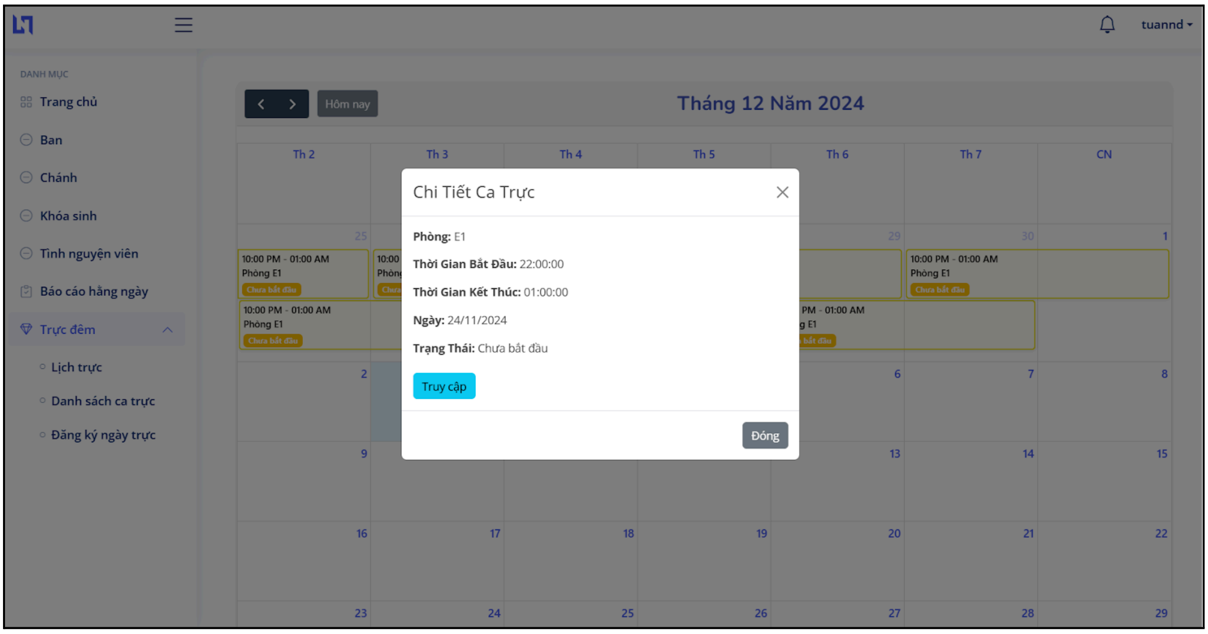


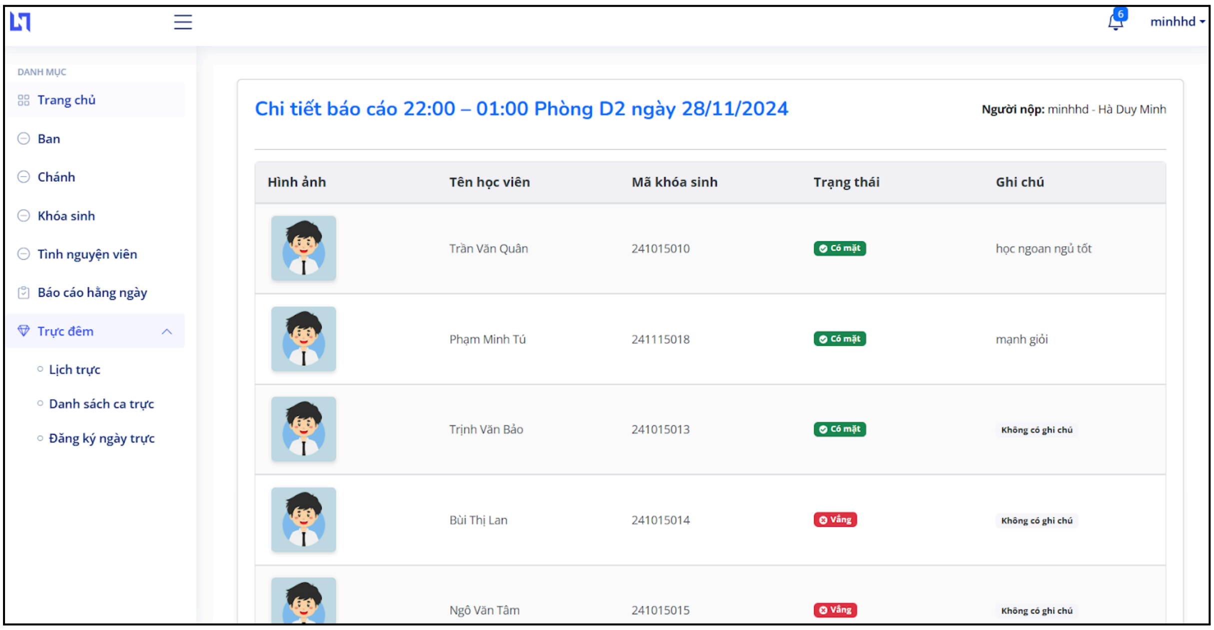


To view night shifts detail, follow these steps:

1. The user logs in with a manager account and selects the "Lịch trực" under the "Trực đêm" button at the sidebar.
2. The system will display the user's night shift in calendar format.
3. The user clicks on the shift that they want to see in detail.
4. The system displays the details of the shift.

3.5.8 View night shift report.

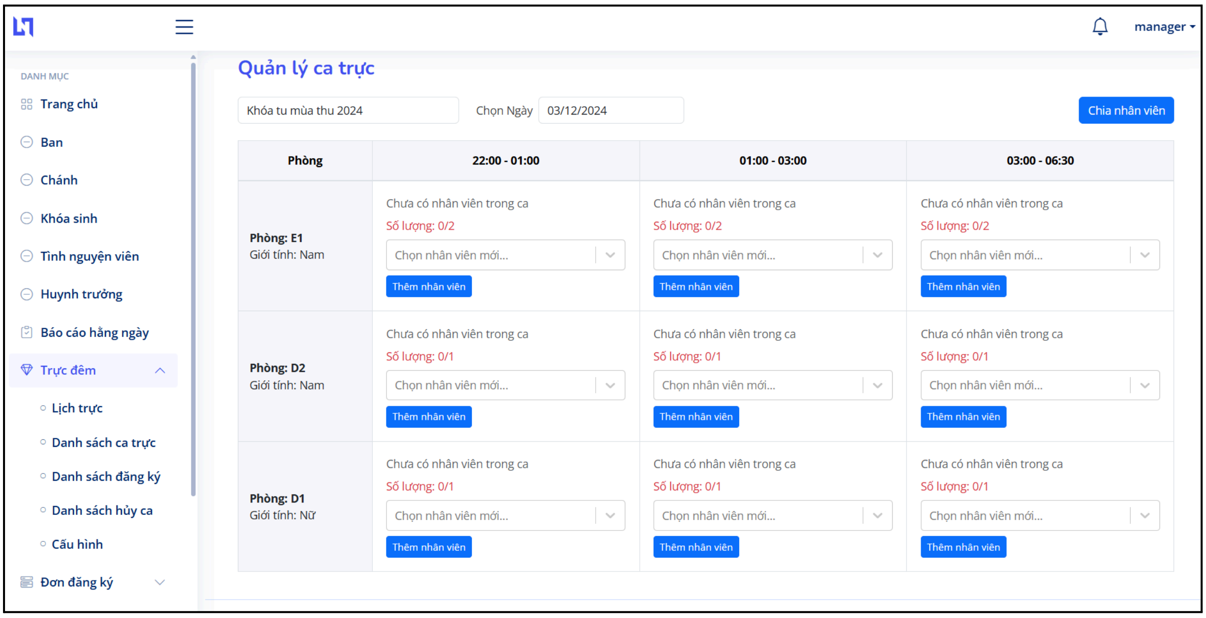




To view night shifts detail, follow these steps:

1. The user logs in with a manager account and selects the "Lịch trực" under the "Trực đêm" button at the sidebar.
2. The system will display the user's night shift in calendar format.
3. The user clicks on the shift that they want to see in detail.
4. The user clicks the "Truy cập" button to access the report.

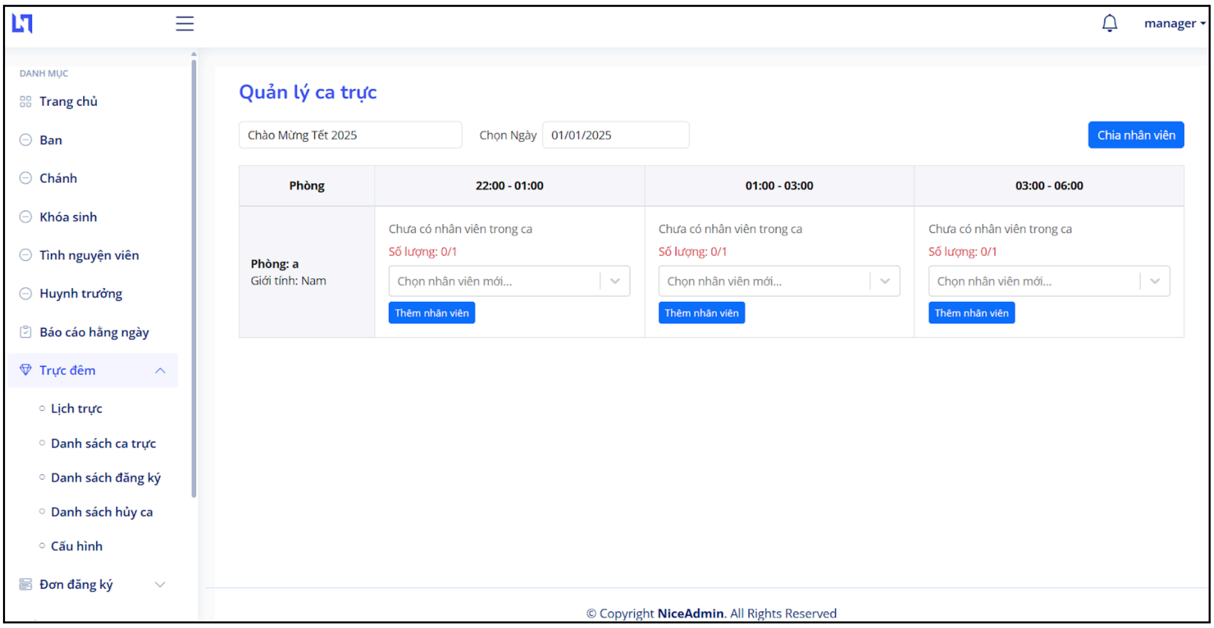
3.5.9 View list of night shift.



To view list of night shifts, follow these steps:

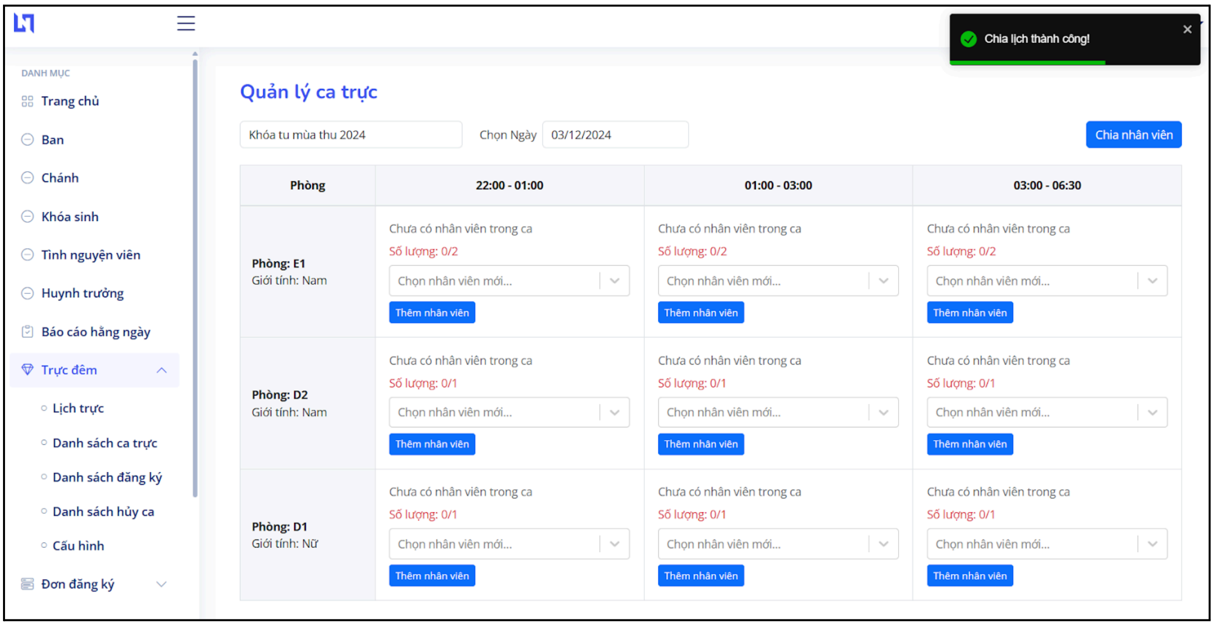
1. The user logs in with a manager account and selects the “Danh sách ca trực” under the “Trực đêm” button at the sidebar they can filter the shift by course and specific date.
2. The system displays the list of night shifts according to room and the number of people per room.

3.5.10 Assign supervisor for night shifts.

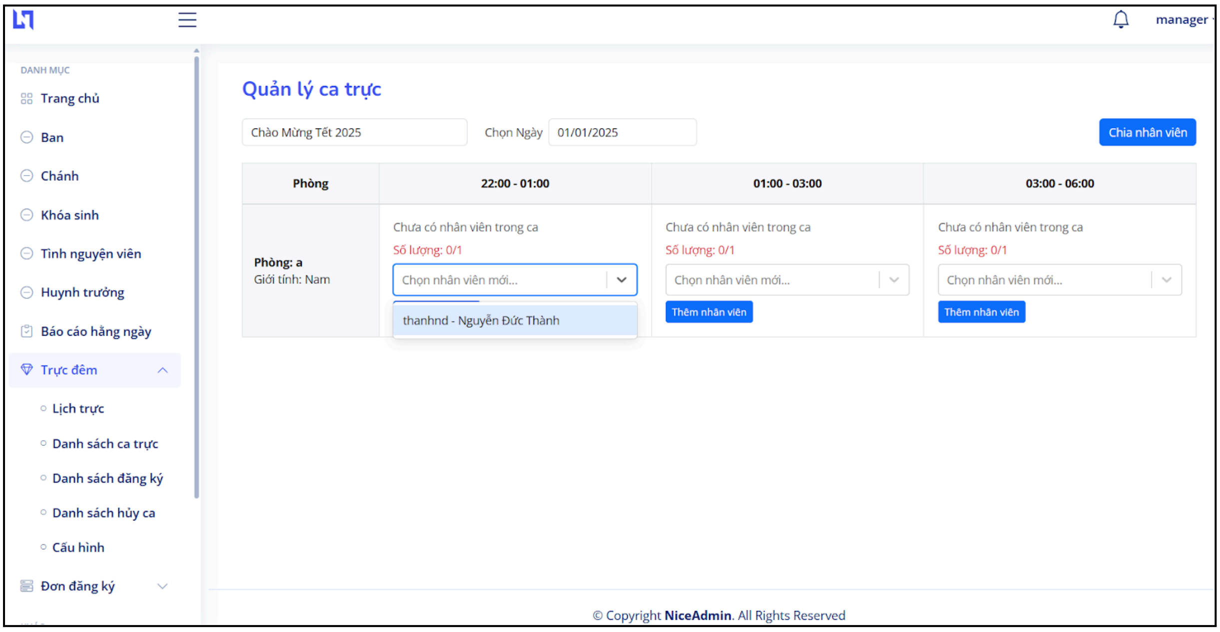


To assign supervisor for night shifts, follow these steps:

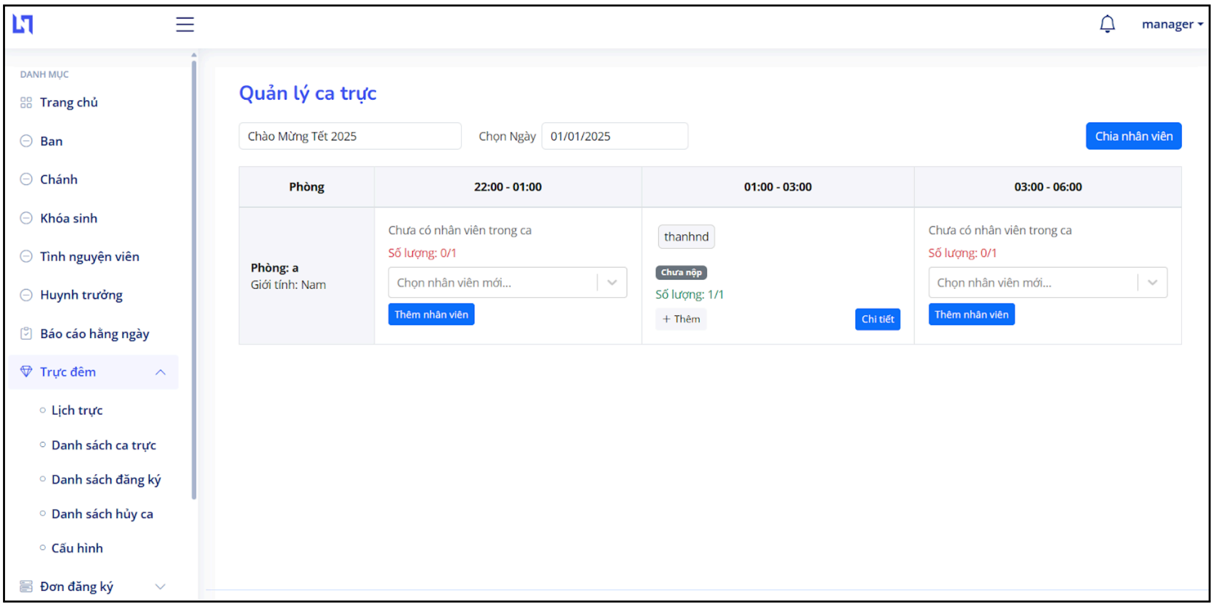
1. The user logs in with a manager account and selects the “Danh sách ca trực” under the “Trực đêm” button at the sidebar they can filter the shift by course and specific date.
2. The system displays the list of night shifts according to room and the number of people per room.
3. The user can divide the supervisor equally into student group in two ways below:
    - The user can click the “Chia nhân viên” button to auto assign a supervisor for the night shift.



- Or user can manually assign supervisor to a night shift by select a person who is available in the dropdown list in each night shift



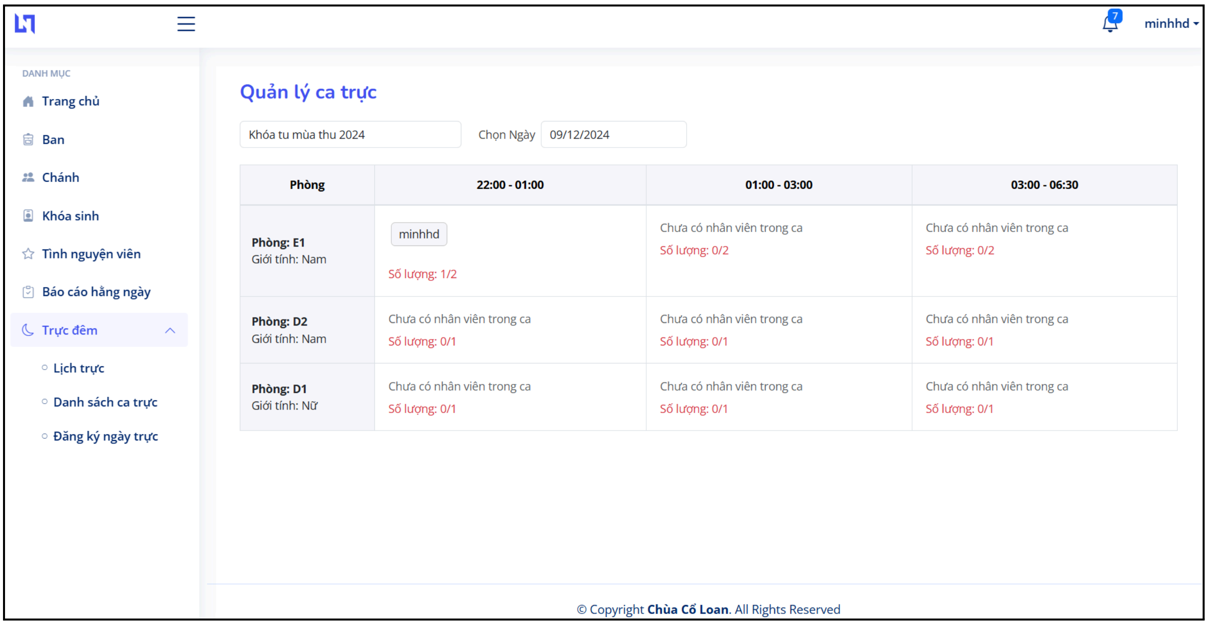
- If a user makes a mistake when assigning a night shift to a supervisor user can delete them from night shift by clicking on their name in each night shift then clicking the "Xác nhận" button to confirm.



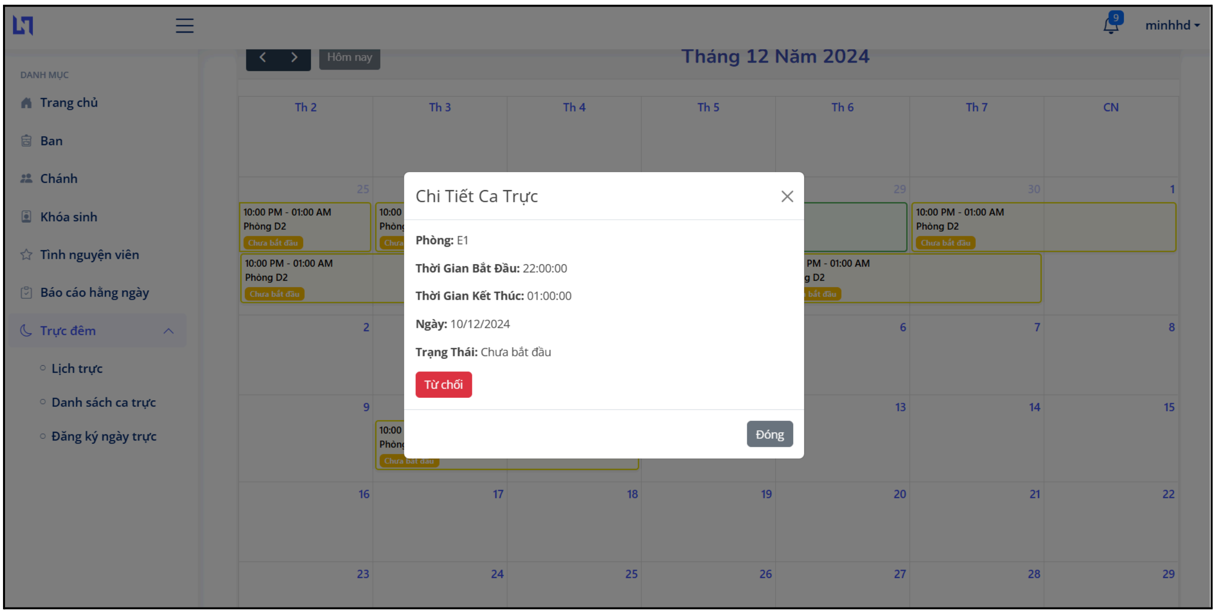
3.5.11 Request to reject night shifts.

To request to reject night shifts, follow these steps:

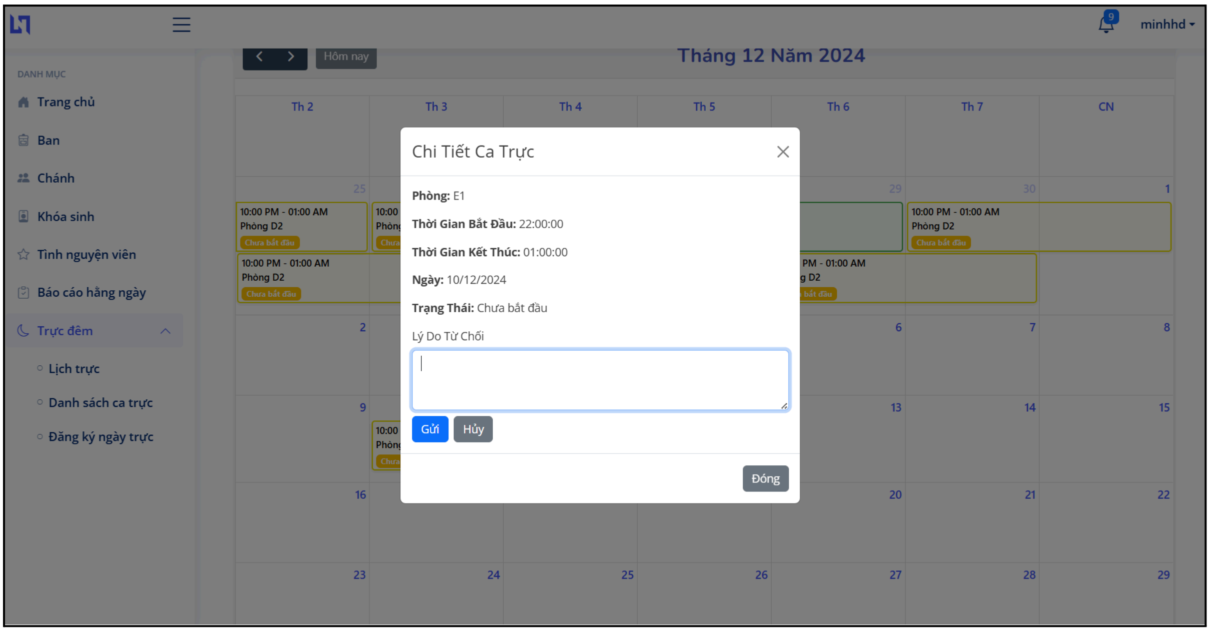
1. The user logs in with a supervisor account and selects the "Danh sách ca trực" under the "Trực đêm" button at the sidebar they can filter the shift by course and specific date.
2. The system displays the list of night shifts according to room and the number of people per room.



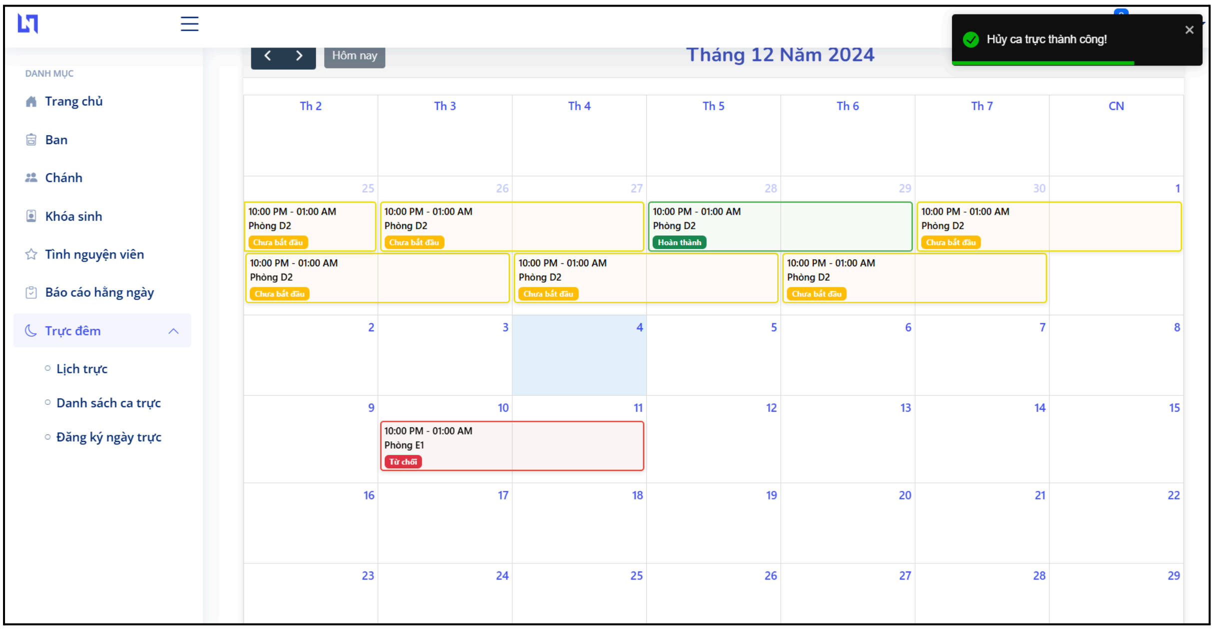
3. The user click on their name of their assigned shift and click "Từ chối" button to reject a night shift



4. The system displays the screen for users to submit the reason why they reject the shift.



5. The user fill in the form and click "Gửi" button to submit the request

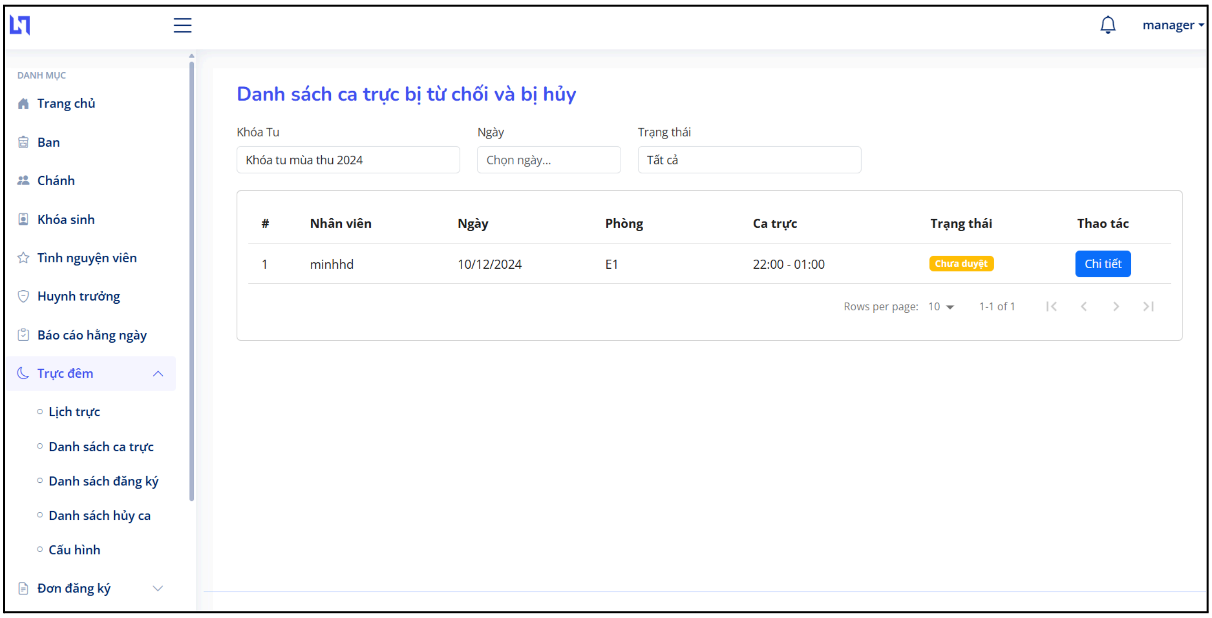


3.5.12 Accept request to reject night shifts.

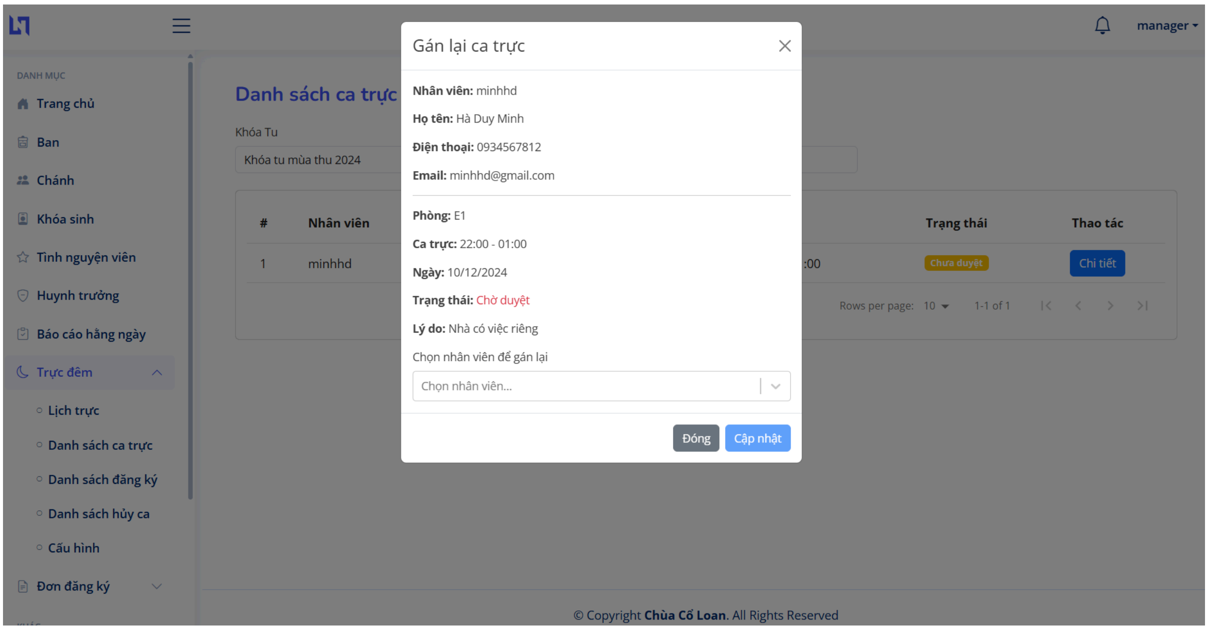
To accept the request to reject night shifts, follow these steps:

1. The user access click the button "Danh sách hủy ca" under the "Trực đêm" button

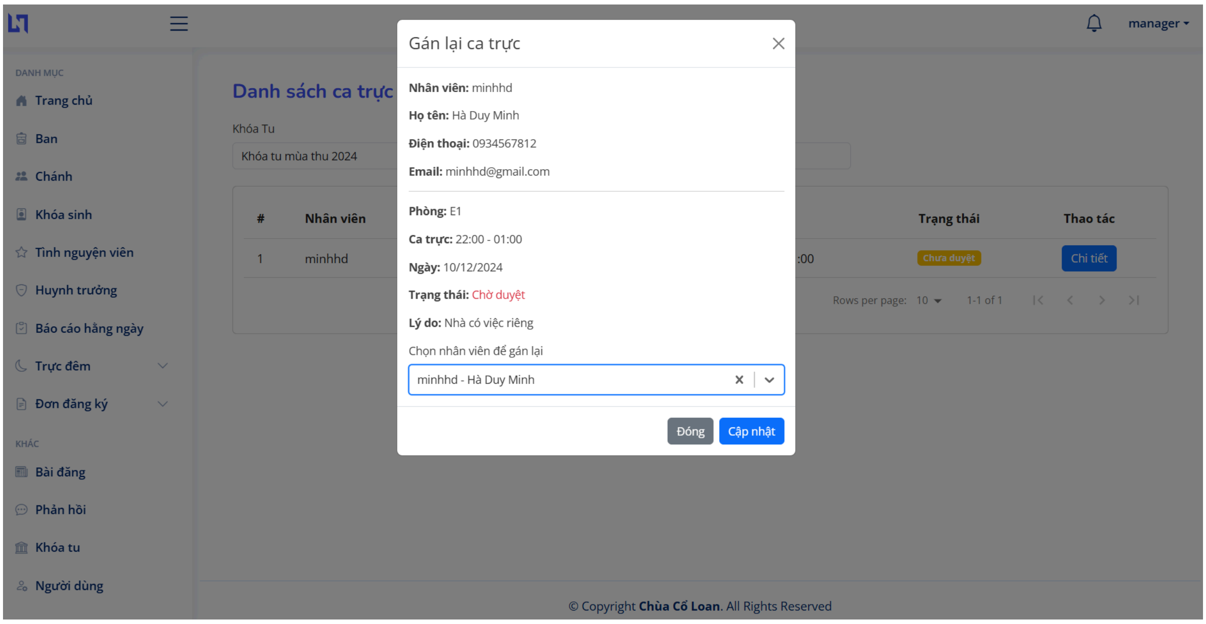
2. The system display the list of people who reject the night shift



3. Users click the "Chi tiết" button to view details of the request.



4. The users select available supervisors to reassign for the shift and click the "Cập nhật" button.



VII. Appendix

Appendix A. Report1_Project Introduction

Appendix B. Report2_Project Management Plan

Appendix C. Report3_Software Requirement Specification

Appendix D. Report4_Software Design Document

Appendix E. Report5_Software Test Documentation

Appendix F. Report5.1_Unit Test

Appendix G. Report5.2_Integration Test

Appendix H. Report5.3_System Test

Appendix I. Report5.4_Acceptance Test

Appendix J. Report6_Software User Guide