

**VIETNAM NATIONAL UNIVERSITY HOCHIMINH CITY
UNIVERSITY OF INFORMATION TECHNOLOGY
ADVANCED PROGRAM IN INFORMATION SYSTEMS**

NGUYEN THANH LOC – CAO HOANG KHANG

STUDENT DORMITORY MANAGEMENT SYSTEM

INFORMATION SYSTEMS ENGINEER

HO CHI MINH CITY, 2023

**NATIONAL UNIVERSITY HOCHIMINH CITY
UNIVERSITY OF INFORMATION TECHNOLOGY
ADVANCED PROGRAM IN INFORMATION SYSTEMS**

**Nguyen Thanh Loc – 19521763
Cao Hoang Khang – 19521660**

STUDENT DORMITORY MANAGEMENT SYSTEM

INFORMATION SYSTEMS ENGINEER

**THESIS ADVISOR
Do Duy Thanh**

HO CHI MINH CITY, 2023

The Assessment Committee is established under the Decision, dateby Rector of the University of Information Technology.

1 - Chairman.

2 - Secretary.

3 - Member

4 - Member.

ACKNOWLEDGMENTS

After a period dedicated to completing the graduation project, all tasks related to the project have now been successfully concluded. Throughout this time, we have received an immense amount of assistance. In the initial section of the thesis, I would like to express my deep gratitude to those who have been immensely supportive.

I sincerely thank the Faculty of Information Systems and the University of Information Technology (VNUHCM) for providing us with favorable conditions to study and carry out this graduation project.

I would like to extend my heartfelt appreciation to lecturer Do Duy Thanh for wholeheartedly guiding and instructing us throughout the process of this project. I am sincerely grateful to all the professors in the Faculty of Information Systems for their dedicated teaching and for imparting invaluable knowledge to us.

Once again, I would like to express my gratitude to all the teachers at the University of Information Technology (VNUHCM) who have lectured and supported me during this time.

Ho Chi Minh, December 2023

Students perform

Nguyen Thanh Loc

Cao Hoang Khang

TABLE OF CONTENTS



ACKNOWLEDGMENTS	i
LIST OF FIGURES	v
LIST OF TABLES	viii
ABSTRACT	ix
Chapter 1 Introduction.....	1
1.1 Reason for choosing the topic	1
1.2 Research problems	2
1.3 Objectives of project	2
1.3.1 Main objectives:	2
1.3.2 Specific objectives:.....	3
1.4 Scope of project	3
Chapter 2 Survey for similar applications.....	4
2.1 Dormitory management at the University of Technical Education, Student implementation: Do Ngoc Quang Huy (2022), Instructor: Master Do Phu Huy	4
2.1.1 The core of the system.....	4
2.1.2 Advantages of the system.....	4
2.1.3 Disadvantages of the system	6
2.1.4 Technology used.....	6
2.2 Dormitory management at Vinh University, Student implementation: Tran Thi Huong (2006), Instructor: Master Tran Thi Kim Oanh	8
2.2.1 The core of the system.....	8
2.2.2 Advantages of the system.....	8
2.2.3 Disadvantages of the system	9
2.2.4 Technology used.....	9
2.3 Dormitory management at Hanoi University of Industry, Student implementation: Pham Dinh Chien and Tran Van Huan (2014), Instructor: Lecturer Nguyen Duc Luu.....	10
2.3.1 The core of the system.....	10
2.3.2 Advantages of the system.....	10
2.3.3 Disadvantages of the system	11
2.3.4 Technology used.....	11
2.4 iDorm SV and iDorm QL	12

2.4.1	The core of the system.....	12
2.4.2	Advantages of the system.....	12
2.4.3	Disadvantages of the system	13
2.4.4	Technology used.....	13
Chapter 3 Development tools and Technology	18	
3.1	HTML.....	18
3.2	CSS	19
3.3	Javascript.....	20
3.3.1	Introduction to Javascript	20
3.3.2	Advantages of Javascript.....	21
3.3.3	Disadvantages of Javascript	22
3.4	React	22
3.4.1	Introduction to React	22
3.4.2	Advantages of React.....	24
3.4.3	Disadvantages of React	24
3.5	MongoDB.....	25
3.5.1	Introduction to MongoDB	25
3.5.2	Advantages of MongoDB	26
3.5.3	Disadvantages of MongoDB	26
Chapter 4 Analysis and Design System.....	28	
4.1	Description Student Dormitory Management System.....	28
4.2	Use Case diagram	30
4.3	Usecase specification	33
4.4	Activity Diagram.....	54
4.5	Sequence Diagram and Class Diagram	75
4.6	Entity Relationship Diagram (ERD)	104
4.7	Description Database	104
Chapter 5 Implementation	109	
5.1	Architecture diagram	109
5.2	Application.....	109
5.2.1	Student role	109
5.2.2	Admin role	119
Chapter 6 Conclusion	129	
6.1	Results	129
6.1.1	Advantage	129
6.1.2	Disadvantage	129

6.2 Future development direction	129
6.3 Conclusion.....	129
REFERENCES	131

LIST OF FIGURES



Figure 4.1 Use case Diagram for system.....	31
Figure 4.2 Use case Diagram for function of system	32
Figure 4.3 Activity Diagram for Register.....	54
Figure 4.4 Activity Diagram for Login.....	55
Figure 4.5 Activity Diagram for View student information	56
Figure 4.6 Activity Diagram for change account password.....	57
Figure 4.7 Activity Diagram for Change bank account number.....	58
Figure 4.8 Activity Diagram for Submit room transfer request.....	59
Figure 4.9 Activity Diagram for Submit room repair request	60
Figure 4.10 Activity Diagram for Submit room check-out request.....	61
Figure 4.11 Activity Diagram for Search room	62
Figure 4.12 Activity Diagram for Edit room.....	63
Figure 4.13 Activity Diagram for Delete room.....	64
Figure 4.14 Activity Diagram for View room information	65
Figure 4.15 Activity Diagram for Search student	66
Figure 4.16 Activity Diagram for Edit student.....	67
Figure 4.17 Activity Diagram for Edit student.....	68
Figure 4.18 Activity Diagram for Approve room transfer request	69
Figure 4.19 Activity Diagram for Approve room repair request	70
Figure 4.20 Activity Diagram for Approve Room check-out request.....	71
Figure 4.21 Activity Diagram for Create invoice.....	72
Figure 4.22 Activity Diagram for Approve invoice	73
Figure 4.23 Activity Diagram for Logout	74
Figure 4.24 Sequence Diagram for Login	75
Figure 4.25 Class Diagram for Login	75
Figure 4.26 Sequence Diagram for View account information	76
Figure 4.27 Class Diagram for View account information	77
Figure 4.28 Sequence Diagram for View student information	77
Figure 4.29 Sequence Diagram for Change account password	78

Figure 4.30 Sequence Diagram for Change bank account number	79
Figure 4.31 Sequence Diagram for View invoice	80
Figure 4.32 Class Diagram for View invoice	81
Figure 4.33 Sequence Diagram for View fee invoice	81
Figure 4.34 Sequence Diagram for View electronic and water invoice	82
Figure 4.35 Sequence Diagram for View unpaid invoice	82
Figure 4.36 Sequence Diagram for Service.....	83
Figure 4.37 Class Diagram for Service.....	83
Figure 4.38 Sequence Diagram for Room transfer	84
Figure 4.39 Sequence Diagram for Room check-out.....	85
Figure 4.40 Sequence Diagram for Room repair	86
Figure 4.41 Sequence Diagram for Manage student	87
Figure 4.42 Class Diagram for Manage student.....	88
Figure 4.43 Sequence Diagram for Insert student	88
Figure 4.44 Sequence Diagram for Search student	89
Figure 4.45 Sequence Diagram for Edit student	90
Figure 4.46 Sequence Diagram for Delete student.....	91
Figure 4.47 Sequence Diagram for Manage room	92
Figure 4.48 Class Diagram for Manage room.....	93
Figure 4.49 Sequence Diagram for Search room	93
Figure 4.50 Sequence Diagram for Edit room.....	94
Figure 4.51 Sequence Diagram for Delete room.....	95
Figure 4.52 Sequence Diagram for Manage invoice.....	96
Figure 4.53 Class Diagram for Manage invoice.....	96
Figure 4.54 Sequence Diagram for Create invoice	97
Figure 4.55 Sequence Diagram for Approve invoice.....	98
Figure 4.56 Sequence Diagram for Manage service	99
Figure 4.57 Class Diagram for Manage service	100
Figure 4.58 Sequence Diagram for Approve room transfer	100
Figure 4.59 Sequence Diagram for Approve room check-out.....	101
Figure 4.60 Sequence Diagram for Approve room repair	102
Figure 4.61 Sequence Diagram for Log out	103

Figure 4.62 Class Diagram for Log out	103
Figure 4.63 Entity Relationship Diagram	104
Figure 5.64 Architecture Diagram	109

LIST OF TABLES



Table 4.1 List of actors.....	30
Table 4.2 List of Use Case	30
Table 4.3 Table Invoice.....	104
Table 4.4 Table InvoiceDetail.....	105
Table 4.5 Table Room.....	105
Table 4.6 Table RoomDetail.....	105
Table 4.7 Table StyleRoom	106
Table 4.8 Table Account	106
Table 4.9 Table RoleDetail	106
Table 4.10 Table Role	106
Table 4.11 Table Student	107
Table 4.12 Table Admin	107
Table 4.13 Table Service.....	108

ABSTRACT

The "Student Dormitory Management System" aims to modernize and optimize university dormitory management through innovative solutions and seamless integration of diverse functionalities.

- [1] Efficient Room Rental Services: The system offers a user-friendly platform for students to search for available rooms and complete the rental process with ease, enhancing student satisfaction.
- [2] Room Allocation and Return Management: It provides a robust framework for tracking room assignments, ensuring smooth check-in and check-out procedures, and maintaining an accurate record of room occupancy, optimizing resource allocation.
- [3] Utility Bill Payments: The system facilitates hassle-free payments for electricity and water bills.
- [4] Internal Maintenance Tracking: To ensure the safety and comfort of students, the system tracks and manages repair and maintenance requests within the dormitory premises.
- [5] Online Interaction: With an intuitive online portal, students can interact directly with the system to perform tasks such as room reservations, bill payments, and submitting service requests.
- [6] Rating and Feedback: Users can log in and provide ratings and feedback on rooms, services, and their overall dormitory experience, building trust for potential renters, while keeping the character count within limits.

The "Student Dormitory Management System" enhances dormitory management by promoting transparency, accountability, and efficiency, ultimately leading to an improved student experience. It also aligns with the trend of digitalization in educational institutions, providing a modern and convenient way to manage student dormitories.

Chapter 1 Introduction

1.1 Reason for choosing the topic

When arriving in a new land, a new place to live will be the top concern of students as well as parents when their children have to study away from home. Choosing a suitable accommodation will greatly determine the quality of your studies. Compared to choosing to stay in a boarding house or apartment, staying in a dormitory will provide an environment with more strengths. The dormitory will be the right choice for you to start your new journey to get used to an independent life. Today, the number of students registering to study at universities and colleges is increasing, creating great pressure to provide and manage student dormitories effectively, helping to optimize management. Manage information about students, staff, accommodation, and related activities.

However, the traditional registration method for students to register has many inconveniences. For example, students have to come in person to learn how to register, and the specific registration time is not clear, so it is very difficult for students living far away; the number of students registering is too high, leading to a loss of control; many students arrive but have no available rooms to stay in. This creates an urgent need for the development and implementation of effective management systems to meet the requirements for managing student information, accommodation, and related dormitory activities. In addition, students have many diverse choices, know the prices of room types in advance, and are updated with the latest dormitory announcements through the system quickly. Therefore, choosing to register and manage online students is an inevitable and necessary development for the present and the future.

The Student Dormitory Management System website is where students can quickly register for accommodation, no matter where they are. Building a dormitory management information system for students is not simply a technological improvement but also has a deeper meaning in providing the best living and learning environment for students. With the following criteria:

Modernization in education. Research and build a dormitory management system to contribute to modernizing the education sector. This is not only about

improving technology but also about creating the best living and learning environment for students.

Optimize management. This system helps improve management processes and optimize resources. The management of student, room, and dormitory operations becomes more flexible and effective.

Improve the student experience. The ultimate goal of this management system is to bring the best living experience to students. From improving services to ensuring security and comfort in dormitories, the system helps create a livable living environment for students.

1.2 Research problems

The traditional dorm room registration process requires students to go to the dormitory to obtain all registration information. This will lead to many inadequacies for students, such as:

- Students will not know room information in advance.
- Students can know the room price in advance.
- Students can get information about accommodation facilities. For example, what is in the room,...
- Students can prepare your own registration documents in advance.
- The management board can process information more quickly when students already know almost all information about the dormitory through the system.

Instead of students spending extra time going to the dormitory to find out all the information about their new place, they can quickly grasp all the information at home by accessing the dormitory's management website. Therefore, the student dormitory management system has eliminated the shortcomings and risks of traditional student dormitory management.

1.3 Objectives of project

1.3.1 Main objectives:

The main goal of this project is to create a student dormitory management

website that allows management to optimize resource management effectively. At the same time, it helps students grasp information during their time in the dormitory effectively and quickly.

1.3.2 Specific objectives:

Building a student dormitory management system. It helps manage students and rooms more effectively and scientifically. It helps students grasp information about dormitories and rooms more clearly. Provide specific solutions to help students become more satisfied with service quality. With the above requirements, it is necessary to create a system to overcome the above problems, bring satisfaction to students, and increase responsibility for the dormitory management board.

To design an effective online management system, it is necessary to meet the following requirements:

- A communication channel for students to give feedback, suggestions, and student services.
- Information is stored in a database, making it easy to retrieve, backup, and restore synchronously, quickly, and reliably.
- Manage student information and transactions.
- Fully manage information about dormitories and students.
- Manage the process of CRUD information for students quickly and easily.

1.4 Scope of project

We cannot think about the existence of an individual department only, for everything is connected to everything else. But due to time constraint, focusing overall would be unwise, therefore we're only focus on the main features such as:

- Student management.
- Dormitory room management.
- Manage the services and activities of the dormitory.
- Manage outstanding and paid student bills.

Chapter 2 Survey for similar applications

2.1 Dormitory management at the University of Technical Education, Student implementation: Do Ngoc Quang Huy (2022), Instructor: Master Do Phu Huy

2.1.1 The core of the system

Successfully built and tested the dormitory management system at the University of Technical Education. Allows students to view available rooms, register online, and view electricity and water bills for each room directly. On this system, manage students, rooms, zones, equipment bills, and dormitory receipts.

2.1.2 Advantages of the system

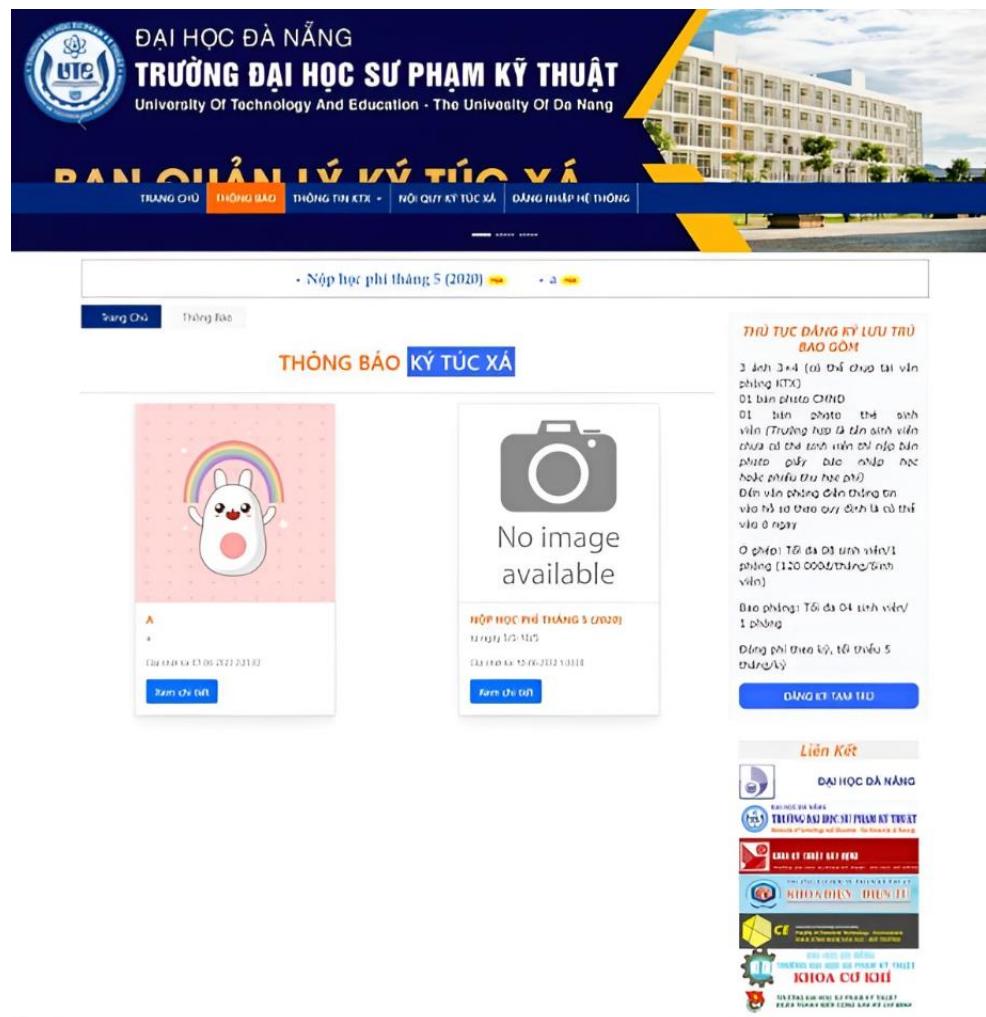
Has all the basic functions:

- Login
- Manage staff
- Manage students
- Manage notices
- Manage rooms
- Manage electricity and water invoices
- Manage payment receipts

Use new programming languages (Angular, Node.js, and MongoDB).

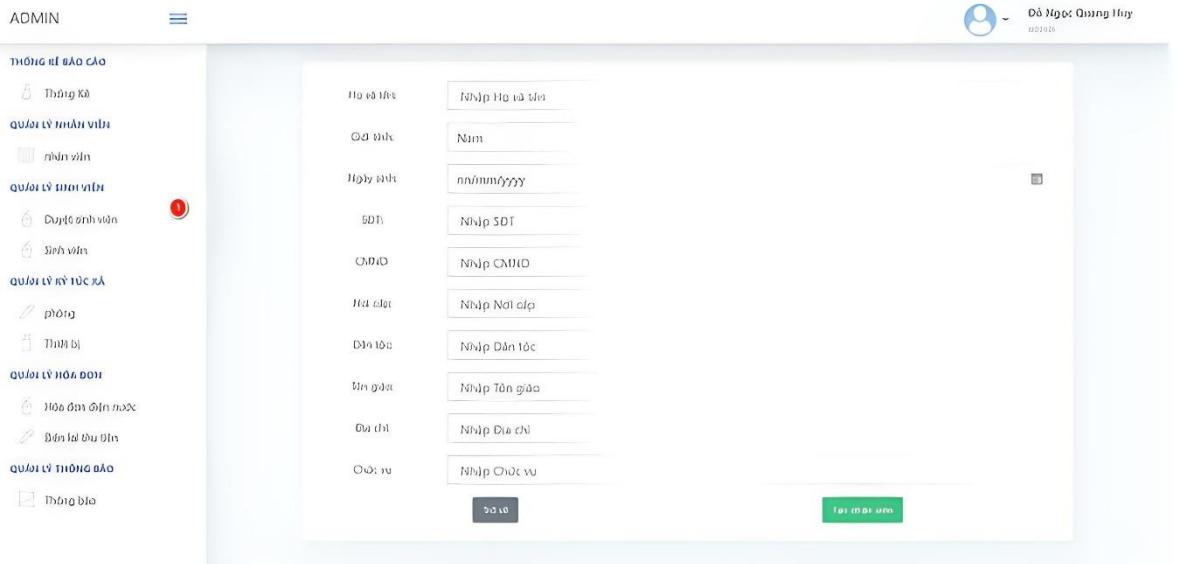
Beautiful interface, suitable for users.

Here are a few pictures that show the system:



Picture 2.1 Notification management function

Picture 2.2 Employee management function



Picture 2.3 Employee registration function

2.1.3 Disadvantages of the system

Software security is not high.

Processing data is a bit complicated.

There are not many advanced functions yet.

2.1.4 Technology used

Angular

Angular is open-source code written in TypeScript and used to design web interfaces. It is considered a good and specialized framework for advanced HTML users.

Advantages of Angular:

Angular is highly appreciated by experts; this source code helps single-page applications work easily and quickly.

Thanks to the ability to bind data onto HTML platforms, front-end code is often very user-friendly.

You can conveniently unit test. Components can be reused more easily.

Disadvantages of Angular:

Security is not high: Angular is a front-end framework, so the security of the front-end is often not as high as the back-end. Therefore, it is necessary to build a data-checking system so that the return is optimal when using the API.

Not secure: The website may become insecure if you use some browsers that have the Disable JavaScript feature.

Node.js

Node is the runtime environment for JavaScript. While JS typically needs a browser to work, Node.js creates the appropriate settings for JS to run outside of the browser. It is built on the V8 JavaScript engine in Chrome.

Advantage of Node.js:

Powerful technology for web application development.

Use only one language for the entire project.

Build microservices simply and quickly.

Develop scalable web applications.

Control flow control feature.

V8 Engine.

Powerful data processing.

Disadvantages of Node.js:

Wastes resources and time.

Suitable for new application development.

MongoDB

MongoDB is a document-oriented database, a type of NoSQL database. Therefore, MongoDB will avoid the table-based structure of relational databases to accommodate documents such as JSON with a very flexible schema called BSON (Binary JSON).

Advantages of MongoDB:

It is not allowed to store more than 16MB of data in a document.

MongoDB is considered by many users to be memory-consuming because it stores data in the form of key - value, collection. On this platform, the data only

differs in value, so the key will still be repeated many times. Because Join is not supported, it easily leads to data redundancy.

There is a risk of data loss when the backup is not completed. This can completely happen because MongoDB's process of inserting, updating, or removing records does not immediately update to the hard drive. It takes about 60 seconds for this platform to write all the changed data from RAM to the hard drive. If a problem such as a power failure occurs in this case, data loss may occur.

Disadvantages of MongoDB:

It is not allowed to store more than 16MB of data in a document.

MongoDB is considered by many users to be memory-consuming because it stores data in the form of key - value, collection. On this platform, the data only differs in value, so the key will still be repeated many times. Because Join is not supported, it easily leads to data redundancy.

There is a risk of data loss when the backup is not completed. This can completely happen because MongoDB's process of inserting, updating, or removing records does not immediately update to the hard drive. It takes about 60 seconds for this platform to write all the changed data from RAM to the hard drive. If a problem such as a power failure occurs in this case, data loss may occur.

2.2 Dormitory management at Vinh University, Student implementation: Tran Thi Huong (2006), Instructor: Master Tran Thi Kim Oanh

2.2.1 The core of the system

All the required functions for a basic dormitory management model (student management, rooms, areas, unit prices, receipts, operations, etc.) Beautiful interface and easy to use.

2.2.2 Advantages of the system

Meets basic software requirements.

Interface suitable for new users.

Easy to operate and precise.

2.2.3 Disadvantages of the system

Using old technology to design the interface is not beautiful.

It will be difficult to develop on the web.

2.2.4 Technology used

Visual Basic

Visual Basic is a programming language that provides general ease of use combined with the ease of implementing a graphical user interface. It is relatively simple and therefore limited in functionality compared to more advanced, cross-platform languages such as Java.

Advantages of Visual Basic:

The ability to support interaction with users is very good and is very professional and user-friendly.

This programming language has a simple structure and is easy to use.

Executable code is the highlight of Visual Basic.

Allows programmers to write code, check for bugs, or fix bugs quickly.

Disadvantages of Visual Basic:

Visual Basic may not achieve the same performance as other programming languages, such as C++ or C#.

Visual Basic can be limited in flexibility compared to other programming languages.

Visual Basic is often tightly coupled with the Windows operating system, which means that VB applications often only run on Windows systems and are incompatible or difficult to port to other platforms.

Compared to some other programming languages, Visual Basic does not receive as widespread support and active development from the programming community as some other languages, such as Python or JavaScript.

Microsoft Access

Microsoft Access, often abbreviated as MS Access or simply Access, is a relational database management software copyrighted by Microsoft.

Advantages of Microsoft Access:

Access provides an intuitive and easy-to-use environment for users without programming or database expertise.

Access integrates tightly with other applications in the Microsoft Office suite, such as Excel, Word, and Outlook, making it easy for users to share data and integrate data from different sources.

Access allows users to simply create forms, reports, queries, and user interfaces.

Access databases can be easily moved and shared between users. This helps in effective teamwork and data management.

Disadvantages of Microsoft Access:

Because it is a Microsoft proprietary language, its compatibility with non-Microsoft systems is limited.

Limited web development applications.

2.3 Dormitory management at Hanoi University of Industry, Student implementation: Pham Dinh Chien and Tran Van Huan (2014), Instructor: Lecturer Nguyen Duc Luu

2.3.1 The core of the system

The system uses a simple interface, suitable for the simplest operations for users, reasonable business order, ensuring high accuracy.

2.3.2 Advantages of the system

Beautiful font, easy to see.

Simple and clear operations.

High accuracy.

Forms are available to quickly fill out information.

2.3.3 Disadvantages of the system

The interface is a bit old.

If later data is large, it can slow down the software.

There are no automatic functions yet.

2.3.4 Technology used

C Sharp (C#)

C# (C Sharp) is a powerful and general-purpose programming language, widely used in many different types of applications and fields.

Advantages of C#:

Is a common and popular programming language similar to languages such as C++, Pascal, Python, and C.

For beginners, it will be easy to develop.

Inherit the advantages of powerful languages because C# is built on their foundation.

Improve C/C++ defects such as pointers, side effects, etc.

Disadvantages of C#:

Only runs on Windows platforms and has the .NET Framework installed.

Hardware manipulation is weaker than in other languages; most have to rely on Windows.

Limited to special applications: C# is commonly used for developing Windows applications, games, mobile applications, and ASP.NET web applications.

Consumes compilation time: Because C# is a static language, the source code needs to be compiled into machine code before running, which can increase development time compared to just-in-time compiled languages.

SQL Server

Structured Query Language (SQL) is a structured query language popular in the field of storing, processing, and retrieving data in relational databases.

Advantages of SQL Server:

Has high query and retrieval speeds. Highly flexible.

Multiple MS SQL versions can be installed on a single server.

Maintain separate environments (production, development, and testing).

Separate security privileges.

Disadvantages of SQL Server:

SQL Server can only operate on Windows operating systems.

You must pay a license fee if you want to run multiple databases.

High cost.

Complex interface.

2.4 iDorm SV and iDorm QL

2.4.1 The core of the system

iDormSV: The iDormSV application, developed by the Dormitory Management Center (TTQLKTX) under Ho Chi Minh City National University (HCMC-NU), was introduced in 2021 to provide utilities for on-campus residents at HCMC-NU dormitories.

iDormQL: The iDormQL application, also developed by TTQLKTX, was launched in 2022 and is specifically designed for dormitory management staff, aiding in the efficient administration of the dormitory system.

2.4.2 Advantages of the system

iDorm SV:

Convenience: Students can easily access information, register for services, and submit support requests directly through the application.

Time and Effort Savings: Students no longer need to visit the dormitory in person for procedures such as registration and support.

Transparency: Provides clear information about dormitory regulations and services.

iDorm QL:

Efficient Management: Assists dormitory staff in managing student information, room conditions, and repair statuses effectively.

Swift Request Handling: Receives, assigns, and tracks the timely processing of student requests.

Statistical Reporting: Provides statistical data on the dormitory situation, aiding in management decision-making.

2.4.3 Disadvantages of the system

iDorm SV:

Quality: Some features are not fully developed, impacting the user experience. For example, The service registration feature does not allow students to view registration history, the support request sending feature does not allow students to track request processing progress.

Stability: The application encounters occasional glitches, hindering access or feature utilization. For example, Stability: The application sometimes crashes, making it difficult to access or use features.

iDorm QL:

User Interface: The interface is not user-friendly and requires improvement for ease of use. For example, the interface layout is not scientific, the colors are not harmonious, and the fonts are not suitable for users.

Feature Completeness: Lacks some essential features for effective management. For example, there is no data analysis feature to support management decisions, nor is there a feature to link with other systems to exchange data.

2.4.4 Technology used

Both iDorm SV and iDorm QL leverage the following technologies:

Flutter

An open-source framework developed by Google, allowing cross-platform application development (Android, iOS).

Advantages of Flutter:

Flutter also applies to the web and provides proper documentation that allows you to test how native controls work.

Flutter decouples the UI from native controls, eliminating (though rare) incompatibility errors from the manufacturer. The distinct UI also automatically provides uniformity across all system versions.

Flutter runs fast, saving time, effort and money.

Disadvantages of Flutter:

Flutter is not yet perfect. Because it is a new application, Flutter is not yet perfect.

Flutter applications are quite "heavy". They take up a lot of space and take a long time to download or update.

Frameworks change rapidly, making it difficult to maintain code.

Firebase

Google's mobile app development platform, offering features such as data storage, databases, and data analytics.

Advantages of Flutter:

Easy to use and integrate, Firebase has an intuitive and user-friendly interface.

Continuously updated and cross-platform.

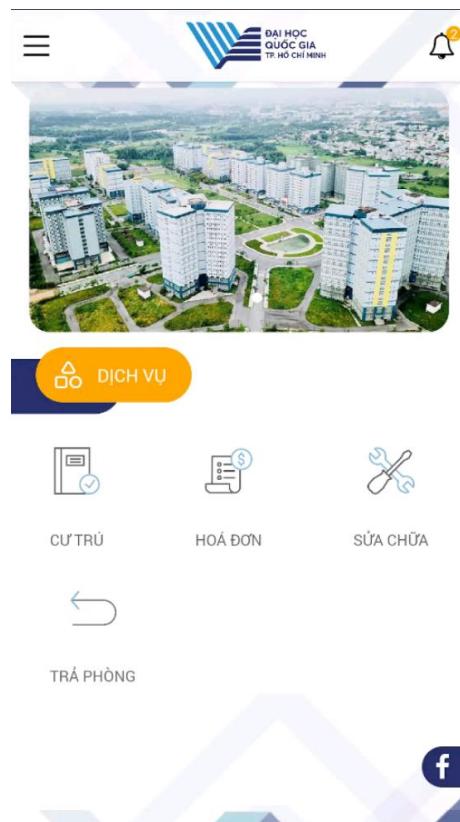
Disadvantages of Flutter:

Limits on application size.

Limited customization capabilities.

Security is weak.

Below are some illustrations of iDorm SV and iDorm QL



Picture 2.4 Home Screen of iDorm SV

Thông tin sinh viên

Nguyễn Văn Hiếu
ĐH Công nghệ Thông tin, ĐHQG
TP.HCM

Thay đổi PIN

Thông tin chung Thông tin liên hệ

CMND/Thẻ căn cước

Mặt trước Mặt sau

Ngày sinh: 09/11/1996

Giới tính: Nam

Năm sinh viên:

Picture 2.5 Personal Information Page of iDorm SV



Picture 2.6 Home Screen of iDorm QL

The image shows the room list screen of the iDorm QL application. The top navigation bar is dark blue with the text 'VIETTEL' and signal strength, the time '13:25', and a back arrow icon. The title 'Khu' is centered above the content area. The content area has two dropdown menus: 'Chọn khu' (Select Area) set to 'Khu A' and 'Chọn cụm' (Select Cluster) set to 'CM 1'. Below these are four small icons representing room types: A01, A02, A03, and A04. The main content is a table titled 'Tầng 1' (Floor 1) showing room details. The table has two rows, each corresponding to a room unit P101 and P102. Each row contains a room icon, the room number, the name 'Khu A - A03 - P. 101' or 'Khu A - A03 - P. 102', room type 'Phòng 8 giường', price information ('Đơn giá' 140.000, 'Đơn giá hè' 140.000), and the number of beds ('Số giường'). The table has scroll bars at the top and bottom.

Tầng 1	
P101	Khu A - A03 - P. 101 Loại Phòng 8 giường Đơn giá 140.000 Đơn giá hè 140.000 Số giường
P102	Khu A - A03 - P. 102 Loại Phòng 8 giường Đơn giá 140.000 Đơn giá hè 140.000 Số giường

Picture 2.7 Room List Screen of iDorm QL

In general, these projects focus on developing core functions for the student dormitory management system, such as managing student information, rooms, and creating receipts and invoices. The common goal is to manage student dormitories as effectively and optimally as possible. The interface must be as user-friendly and easy-to-use as possible. And our team also has the same goal, aiming to build a student dormitory management system to help manage it as effectively as possible.

There are a few distinctions, though. The first is in the technology that was employed to construct the system; we will go into more detail about this in the upcoming chapter (Chapter 3 Developer tools and Technology). Next, in terms of features, our team differs in that we require students to register in person rather than allowing them to do so online. This is so because dorms are reserved for students from particular schools, and students can only register to stay in a dorm after completing the admissions process at their home institution. We then add a few more elements to the stay process, like room transfers, check-out, room repair requests,... In the upcoming chapters, these characteristics will be outlined in detail.

Chapter 3 Development tools and Technology

3.1 HTML

HTML (HyperText Markup Language) is a markup language designed to create web pages presenting information on the World Wide Web. It is defined as a simple application of SGML and is used in organizations requiring complex publishing requirements. HTML has become an internet standard maintained by the World Wide Web Consortium (W3C). The latest official version of HTML is HTML 4.01 (1999). Subsequently, developers replaced it with XHTML. Currently, HTML is undergoing further development with the HTML5 version, promising a new look for the web.

Using dynamic HTML or Ajax, programmers can create and handle it using a multitude of tools, from a simple text editor (where one can start typing right from the first lines) to complex WYSIWYG publishing tools.

HTML is not a programming language; it is a markup language used for presentation purposes.

Types of HTML Tags:

- HTML Tags: These are used to define HTML elements.
- HTML Tags are Enclosed: They are enclosed by the greater than < and less than > signs.
- Paired Tags: HTML tags typically come in pairs, such as and .
- First Tag - Opening Tag, Second Tag - Closing Tag: The first tag marks the beginning, and the second tag denotes the end.
- Content Between Tags: The text or content between the opening and closing tags is the information displayed.
- Case Insensitivity: HTML tags are not case-sensitive; for instance, both and are considered the same.

HTML Tag Attributes:

- All HTML tags come with their respective attributes. These attributes provide

information about the HTML elements of a webpage. This tag defines the body of the HTML page: <body>. By adding an attribute like bgcolor, you can inform the browser that the background color of this page is red, for instance: <body bgcolor="red"> or <body bgcolor="#E6E6E6"> (#E6E6E6 is the hex value of the color).

- This tag defines an HTML table: <table>. With an attribute for border, you can inform the browser that the table will have no border: <table border="0">.
- Attributes always come in pairs in the format name="value". Attributes are always added to the opening tag of the HTML element.

In summary, HTML is the most basic and important foundation for developing websites. It provides the basic structure for web content, and HTML is an indispensable element in building a website or a modern web application.

3.2 CSS

Cascading Style Sheets (CSS) is a style sheet language used to describe the presentation of a document written in HTML. It complements HTML by defining how the elements should be displayed on a webpage. CSS enables the separation of document content (HTML structure) from document presentation (styling and layout).

Key aspects of CSS:

- Styling HTML Elements: CSS allows the styling of individual HTML elements or groups of elements. It defines properties such as color, size, font, spacing, and positioning.
- Simplifying Design: By applying CSS rules, a developer can easily control the layout and appearance of multiple web pages, ensuring a consistent and professional look across an entire website.
- Selectors and Properties: CSS uses selectors to target specific HTML elements and then applies various properties to those elements. For example, a selector might target all <p> (paragraph) elements and set their font size or color.
- Cascading Nature: CSS rules can be applied in multiple ways. The

"cascading" aspect of CSS means that styles can be inherited or overridden based on their specificity, source, and order in the code.

- External Stylesheets: CSS can be written within an HTML file using internal styles or can be linked externally to a separate CSS file. External stylesheets allow for easier management and consistency across a website.
- Responsive Design: CSS enables the creation of responsive designs by using media queries and other techniques to adapt the layout of a webpage based on the device or screen size. This is crucial for ensuring a good user experience on various devices.

CSS plays a crucial role in modern web design, allowing for greater flexibility, control, and aesthetic appeal in presenting web content. It works hand-in-hand with HTML, enabling web developers to create visually engaging and user-friendly websites.

3.3 Javascript

3.3.1 Introduction to Javascript

JavaScript is a high-level, versatile programming language primarily used to create dynamic and interactive content on web pages. It is a core technology in web development, allowing for the enhancement of the user experience through various functionalities.

Key Aspects of JavaScript:

- Client-Side Scripting: JavaScript is primarily executed on the client side (in the user's web browser), enabling dynamic changes and interactions without having to communicate with the server constantly. This leads to faster and more responsive web applications.
- Enhancing user interactivity: It's extensively used to create interactive elements like forms, animations, games, and responsive design features. This enables user engagement and interaction with web content.
- Manipulating HTML and CSS: JavaScript can dynamically manipulate HTML and CSS, allowing for modifications in the content and styling of web

pages in real-time based on user actions or other events.

- Asynchronous operations: It supports asynchronous operations through callbacks and promises, allowing for actions to be executed independently from the main program flow. This is crucial for tasks such as fetching data from servers without blocking other operations.
- Event handling: JavaScript handles various events like user actions (clicks, mouse movements, keystrokes), document loading, or timers. It responds to these events, triggering specific actions or functions.
- Cross-Browser Compatibility: JavaScript is supported by all major web browsers, making it a universal choice for web development. However, developers need to ensure their code is cross-browser compatible due to slight variations in browser implementations.
- Server-Side development: With Node.js, JavaScript can also be used for server-side development, enabling full-stack development using a single programming language.

3.3.2 Advantages of Javascript

- Quick response: Users accessing the website do not have to wait too long for the page to reload because of JavaScript support.
- Rich development interface: With more than 90% of websites using JavaScript, source code libraries supporting JS are also diversely built. Some JS source libraries: Algolia Places, AOS (Animate On Scroll), Chart.js, React.
- Easy to learn for beginners: JavaScript is an accessible language with a syntax similar to other languages such as C++ and Java, so learning and getting used to it can happen quickly.
- Asynchronous Programming: JavaScript supports asynchronous programming methods, allowing tasks to be performed without slowing down the main application process.
- Large Community and Strong Support: JavaScript has a large and widespread development community. There are many libraries and frameworks such as

React, Vue.js, or Angular developed to help enhance the capabilities and power of JavaScript.

3.3.3 Disadvantages of Javascript

- Restrictions on some browsers: Although supported on most popular browsers today, during operation there are also some browsers that do not support JavaScript or on some websites that use JS to prevent some actions from the user, causing difficulty in browsing the web.
- Poor Memory Management: During execution, JavaScript does not manage memory as clearly as other languages, leading to memory leak errors if not managed strictly.
- Security: JavaScript can have security issues due to its ability to run code from untrusted sources, leading to the risk of Cross-Site Scripting (XSS) attacks.
- Performance optimization: Sometimes optimizing the performance of a JavaScript application can become complicated, especially when working with applications with large amounts of data or high processing requirements.

Javascript is a flexible and easy-to-use language for beginners. The reason this project uses Javascript instead of other languages is because of the following reasons: Javascript can run in the browser without going through any external devices, which reduces the complexity of deploying and maintaining the application. Next, Javascript can be used on both the frontend and backend, and it also runs on many different platforms, including the web, mobile, and desktop. Javascript is a popular language and has a strong development community. It has a rich documentation repository and many frameworks such as React, Vue.js, Express.js,... In short, with projects medium and small, Javascript is a very good choice.

3.4 React

3.4.1 Introduction to React

React is an open-source JavaScript library mainly used for building user interfaces, especially for single-page applications where user experience is a crucial

aspect of web development. Developed and maintained by Facebook, React has gained immense popularity for its efficiency in creating interactive and responsive UI components.

Key Aspects of React:

- Component-Based Architecture: React uses a component-based architecture, where the UI is broken down into reusable components. This modular approach allows developers to create complex interfaces by composing smaller, independent parts, which makes development more manageable and scalable.
- Virtual DOM (Document Object Model): React employs a virtual DOM, a lightweight copy of the actual DOM. When changes occur, React compares the virtual DOM with the real DOM and then selectively updates only the necessary parts. This minimizes the number of manipulations in the actual DOM, enhancing performance.
- JSX (JavaScript XML): JSX is a syntax extension for JavaScript that enables the mixing of HTML-like code within JavaScript. This makes writing React components more intuitive and readable.
- Unidirectional data flow: React follows a unidirectional data flow. Changes in the data flow from parent components to child components, maintaining a clear and predictable data lifecycle.
- Reusable components: The ability to create reusable and composable components allows for efficient development and maintenance of complex UIs. Developers can use these components in various parts of the application.
- Community and Ecosystem: React has a vast and active community along with an extensive ecosystem of tools, libraries, and resources, contributing to its growth and making it easier for developers to find solutions and support.
- Support for mobile development: With React Native, a framework based on React, developers can build mobile applications for iOS and Android platforms using JavaScript and React principles.

3.4.2 Advantages of React

- Component-Based Architecture: React uses a component-based architecture, allowing the UI to be divided into independent, reusable components. This facilitates easier management, maintenance, and scalability of the application.
- Virtual DOM (Document Object Model): React utilizes the Virtual DOM to optimize application performance. Instead of updating the entire DOM when changes occur, React updates only the necessary DOM elements, speeding up the rendering process and improving the user experience.
- JSX (JavaScript XML): JSX enables HTML to be written in JavaScript, creating more readable and writable source code. This makes interface creation clearer and more straightforward.
- Unidirectional Data Flow: The unidirectional data model eases control and management of data, avoiding unnecessary complexity when handling data within the application.
- Large Community Support: React has a large and active community along with extensive documentation, libraries, and tools, aiding in issue resolution and efficient application development.
- React Native: React Native allows for mobile application development for both iOS and Android using JavaScript and the React architecture. This makes mobile app development flexible and time-efficient.

3.4.3 Disadvantages of React

- Difficult for beginners: Learning React can be quite complicated at first for beginners, especially when it comes to understanding concepts like JSX, Virtual DOM, unidirectional data models, etc.
- Expansion in large projects: In large and complex projects, state management can become more complex, requiring care to avoid state management becoming chaotic and confusing.
- Use third-party libraries: Sometimes, to handle specific functions, you need

to use third-party libraries or have to learn, implement, and build solutions yourself.

As mentioned, React is a Javascript framework, so using React in the application development process brings many benefits to the application building process. In addition, React is a framework with a very large user and development community, so it helps a lot in the process of solving arising problems effectively and quickly. Therefore, choosing React to build applications is an effective choice.

3.5 MongoDB

3.5.1 Introduction to MongoDB

MongoDB is a document-oriented database, a type of NoSQL database. Therefore, MongoDB will avoid the table-based structure of relational databases to accommodate documents such as JSON with a very flexible schema called BSON (Binary JSON). MongoDB uses data storage in the form of Document JSON, so each collection will have different sizes and documents. The data is stored in a JSON document, so queries will be very fast.

** NoSql is an open-source database. These are the abbreviations for the phrase None-Relational SQL, also known as Not-Only SQL.*

Key Aspects of MongoDB:

- Document-Oriented: MongoDB is designed to store data in the form of documents instead of tables, like in a relational database management system (RDBMS). Each document in MongoDB is stored as an independent record consisting of fields (key-value pairs) and corresponding values.
- Backup and recovery: MongoDB provides flexible data backup and recovery, allowing users to store copies of data and restore data in case of failure.
- Security: MongoDB supports many security features, including user authentication, data encryption, and access control.
- Indexing: MongoDB creates indexes to speed up querying and searching for data in the database. When creating an index on a field or set of fields, MongoDB will create a corresponding data structure, sorting and storing the

values of that field or set of fields in a certain order. When querying data, MongoDB uses indexes to quickly search for and return documents that match the query criteria. Using indexes helps reduce query time to search for data while also increasing MongoDB's performance and scalability.

3.5.2 Advantages of MongoDB

- Ease of use: MongoDB provides intuitive and easy-to-use data management tools that help users optimize performance and manage databases with ease.
- Easily integrates with Big Data Hadoop.
- Flexibility: MongoDB is a non-relational database system that provides the ability to store data anytime, anywhere, without having to adhere to a specific relational model.
- Scalability: MongoDB is easily scalable, thanks to the sharding feature that allows data to be divided into multiple parts and stored on multiple servers.
- Fast access speed: MongoDB can respond to data query requests in less time than traditional relational database systems.
- High availability: MongoDB provides data backup and recovery features, helping users protect their data from risks.

3.5.3 Disadvantages of MongoDB

- It is not allowed to store more than 16MB of data in a document.
- MongoDB is considered by many users to be memory-consuming because it stores data in the form of key - value, collection. On this platform, the data only differs in value, so the key will still be repeated many times. Because Join is not supported, it easily leads to data redundancy.
- There is a risk of data loss when the backup is not completed. This can completely happen because MongoDB's process of inserting, updating, or removing records does not immediately update to the hard drive. It takes about 60 seconds for this platform to write all the changed data from RAM to the hard drive. If a problem such as a power failure occurs in this case, data loss may occur.

To meet many factors and each specific requirement that the project brings, our team wants to mention the use of MongoDB to fit the above, to choose MongoDB for reasons such as MongoDB for It allows users to use it for free; in addition, MongoDB is easy to use and very suitable for web application development.

As discussed in Chapter 2 (Related Applications and Technologies) of the previous chapter. There will be certain technological variations in the system our team built for this student dormitory management project. We believe that every language and technology has its own advantages and disadvantages, thus we did not select these technologies based solely on their effectiveness and performance. We selected the aforementioned languages and technologies because they are well-known and free, enabling us to solve issues as they arise with speed and efficiency. We have experience with these technologies from working on class projects during our university education, so we will be more at ease with them than we would be with unfamiliar technology.

Chapter 4 Analysis and Design System

After surveying dissertations on related topics and available similar systems, we focused on identifying important core functions that the system needs to support. These systems will be suitable for student dormitory models or small and medium-sized room models. Below are the core functions that we have identified to build our system.

As Student role: Login, Register, View account information, Service, Logout.

As Admin role: Login, Manage students , Manage rooms, Manage service, Manage invoices, Logout.

These are the essential features that our team has decided to include in our system. The functional requirements of the system and a detailed presentation of the system design analysis are provided just below.

4.1 Description Student Dormitory Management System

3.1.1 Functions that do not require login:

Provides basic information about the dormitory, such as information about room types (number of residents, air conditioning or not), room prices, and general descriptions of room types.

3.1.2 Common functions for Admin and Students:

Login

Description: Allows users to authenticate and access the system using student or admin accounts to use specific features.

Log out

Description: Allows users to exit the session, delete login information, and end all activities on the website.

3.1.3 Functions requiring login:

3.1.3.1 With a Student Account:

View account information

Description: Students can view their name, date of birth, citizen identification number, school name, student code, email, address, school year, and bank account

number (Except bank account number Students can only view other information but are not allowed to edit it. When students want to edit, they need to contact the Admin). In addition, students can change their account password themselves.

Service

Description: Includes room transfer requests, check-out requests, and room repair requests. Students fill out and submit the request form and wait for admin to confirm.

View invoice

Description: Students can view fee invoices, monthly utility invoices and unpaid invoices.

3.1.3.2 With an admin account:

Manage students

Description: Allows Admin to add students to the room, delete, edit, and search for students.

Create student

Description: Admin can create new accounts for students when they register directly at the dormitory.

Manage room

Description: Allows Admin to find rooms, delete rooms, and edit room information.

Manage service

Description: Admin can approve room transfer, check-out, and room repair requests from students (agree or decline).

Manage invoices

Description: Create and approve monthly utility invoices, approve invoices after students pay directly at the dormitory.

4.2 Use Case diagram

Table 4.1 List of actors

No	Actor name	Define
1	Admin	Responsible for management (student management, dormitory room management, invoice management, dormitory room maintenance, and repair management)
2	Student	View registration information, log in, view room information, view accommodation information, view invoices, and manage personal information.

Table 4.2 List of Use Case

No	Use Case name	Define
1	Register	Register a user account in the system
2	Login	Log in to your account in the system
3	View account information	View student details
4	Service	Room repair, check-out and room transfer services
5	View invoice	View invoices (fee invoice, electricity and water invoice, unpaid invoice)
6	View room information	View basic information of the room
7	Manage student	Add, search, delete and edit student information
8	Manage rooms	Search, delete and edit room information
9	Manage service	Approve or decline service application forms from students
10	Manage invoice	Create and approve paid student invoices
11	Log out	Log out of your account from the system

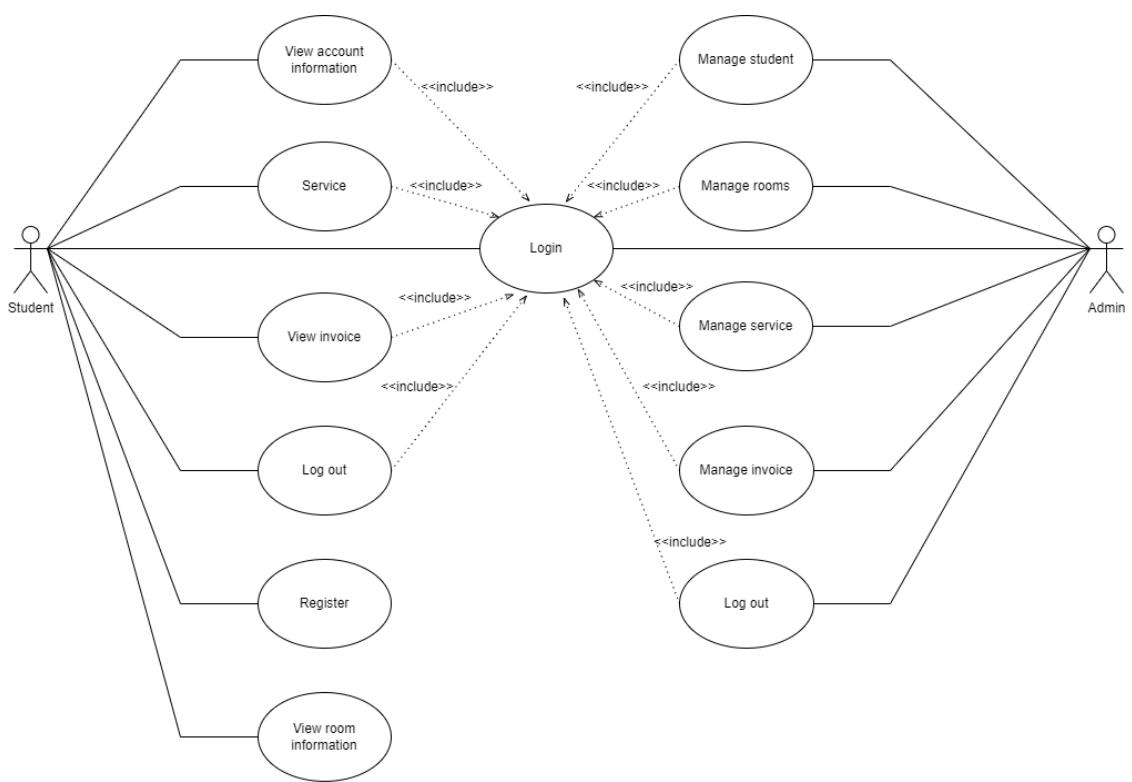


Figure 4.1 Use case Diagram for system



Figure 4.2 Use case Diagram for function of system

4.3 Usecase specification

4.3.1 Register

Use – Case	Content
Use-case name	Register
Description	Allows students to view detailed information about the types of documents that need to be prepared when registering directly at the dormitory. See detailed information about room types.
Actor	Student
Main scenario	<p>Step 1: The system displays the home page.</p> <p>Step 2: Students click on the registration button.</p> <p>Step 3: The system will display information about the types of documents students need to prepare to go to the dormitory to register directly. Besides, the system will display detailed information about room types so students can consider and choose before arriving at the dormitory.</p>
Precondition	Students must click on the registration button
Postcondition	None

4.3.2 Login

Use – Case	Content
Use-case name	Login
Description	Allows user to log in to their account when starting to use it
Actor	Student/Admin
Main scenario	<p>Step 1: The system displays the home page.</p> <p>Step 2: User click on the account login button.</p> <p>Step 3: The system will display login information, including the CMND/CCCD and password.</p> <p>Step 4: Enter CMND/CCCD and password.</p>

	Step 5: After successfully logging in, the system will display the home page for logged-in user.
Precondition	The user must click on the login button
Postcondition	If the user enters the wrong account/password: The system will ask the user to re-enter

4.3.3 View account information

A. View student information

Use – Case	Content
Use-case name	View Information
Description	Allows students to view personal information (which can only be viewed, not edited)
Actor	Student.Main scenario
Main scenario	<p>Step 1: The system displays the home page for logged in students.</p> <p>Step 2: Students click on the student information button.</p> <p>Step 3: The system will display detailed personal information about the student</p>
Precondition	Students must successfully log in and click on the student information button
Postcondition	None

B. Edit bank account number

Use – Case	Content
Use-case name	Edit bank account number
Description	Allows students to view and change bank account numbers
Actor	Student
Main scenario	<p>Step 1: The system displays the home page for logged in students.</p> <p>Step 2: Students click on the student information button.</p>

	<p>Step 3: The system will display detailed personal information about the student.</p> <p>Step 4: Students click on the edit bank account number button.</p> <p>Step 5: The system will display the bank account number for editing.</p> <p>Step 6: Students proceed to edit the bank account number.</p> <p>Step 7: After editing, click the save button.</p>
Precondition	Students must successfully log in and click on the student information button, then click on the edit bank account number button
Postcondition	<ol style="list-style-type: none"> If the change is successful, the system will update and display the new account number that the student just entered. If the change fails, the system will report an error and ask the student to re-enter

C. Edit account password

Use – Case	Content
Use-case name	Edit account password
Description	Allows students to change their password
Actor	Student
Main scenario	<p>Step 1: The system displays the home page for logged in students.</p> <p>Step 2: Students click on the student information button.</p> <p>Step 3: The system will display detailed personal information about the student.</p>

	<p>Step 4: Students click on the edit password button.</p> <p>Step 5: The system will display an information panel for students to fill in, including their current password, new password, and re-enter a new password.</p> <p>Step 6: Students proceed to fill in the information fields.</p> <p>Step 7: After editing, click the confirm button.</p>
Precondition	Students must log in successfully and click on the student information button, then click on the edit password button
Postcondition	<ol style="list-style-type: none"> 1. If the change is successful, the system will notify you of its success. 2. If fails, the system will report an error and ask the student to re-enter.

4.3.4 Service

A. Submit a room transfer request

Use – Case	Content
Use-case name	Submit a room transfer request
Description	Allows students to send a room transfer request to the admin
Actor	Student
Main scenario	<p>Step 1: The system displays the home page for logged-in students.</p> <p>Step 2: Students click on the service button.</p> <p>Step 3: Students click on the room change button.</p> <p>Step 4: The system will display a room transfer registration form.</p> <p>Step 5: Students complete the form.</p>

	<p>Step 6: Students click the submit form button.</p> <p>Step 7: After successfully submitting the form, the system will notify you that the form has been submitted successfully</p>
Precondition	Students must successfully log in and click on the service button, then click on the room change button
Postcondition	<ol style="list-style-type: none"> 1. If the form is successfully submitted, the system will notify you of its success. 2. If the form submission fails, the system will display an error

B. Submit a check-out request

Use – Case	Content
Use-case name	Submit a check-out request
Description	Allows students to send check-out requests to admin
Actor	Student
Main scenario	<p>Step 1: The system displays the home page for logged-in students.</p> <p>Step 2: Students click on the service button.</p> <p>Step 3: Students click the check-out button.</p> <p>Step 4: The system will display a check-out registration form.</p> <p>Step 5: Students complete the form.</p> <p>Step 6: Students click the submit form button.</p> <p>Step 7: After successfully submitting the form, the system will notify you that the form has been submitted successfully</p>
Precondition	Students must successfully log in and click on the service button, then click on the check-out button

Postcondition	<ol style="list-style-type: none"> 1. If the form is successfully submitted, the system will notify you of its success. 2. If the form submission fails, the system will display an error
---------------	---

C. Submit a room repair request

Use – Case	Content
Use-case name	Submit a room repair request
Description	Allows students to send room repair requests to admin
Actor	Student
Main scenario	<p>Step 1: The system displays the home page for logged-in students.</p> <p>Step 2: Students click on the service button.</p> <p>Step 3: Students click on the room repair button.</p> <p>Step 4: The system will display a room repair registration form.</p> <p>Step 5: Students complete the form.</p> <p>Step 6: Students click the submit form button.</p> <p>Step 7: After successfully submitting the form, the system will notify you that the form has been submitted successfully</p>
Precondition	Students must successfully log in and click on the service button, then click on the room repair button
Postcondition	<ol style="list-style-type: none"> 1. If the form is successfully submitted, the system will notify you of its success. 2. If the form submission fails, the system will display an error

4.3.5 View invoice

A. View fee invoice

Use – Case	Content
Use-case name	View fee invoice
Description	Allows students to view fee invoices
Actor	Student
Main scenario	<p>Step 1: The system displays the home page for logged-in students.</p> <p>Step 2: Students click on the invoice button.</p> <p>Step 3: Students click on the fee invoice button.</p> <p>Step 4: The system will display a list of student fee invoices in order from the most recent date to the top of the list.</p> <p>Step 5: Students click on any invoice.</p> <p>Step 6: The system will display details of that invoice</p>
Precondition	Students must successfully log in and click on the Bills button, then click on the fee invoice button
Postcondition	None

B. View electricity and water invoice

Use – Case	Content
Use-case name	View electricity and water invoice
Description	Allows students to view utility invoice
Actor	Student
Main scenario	<p>Step 1: The system displays the home page for logged-in students.</p> <p>Step 2: Students click on the invoice button.</p> <p>Step 3: Students click on the utility bill button.</p> <p>Step 4: The system will display a list of students'</p>

	<p>electricity and water invoice in order from the most recent date to the top of the list.</p> <p>Step 5: Students click on any invoice.</p> <p>Step 6: The system will display details of that invoice</p>
Precondition	Students must successfully log in and click on the Bills button, then click on the utility bills button
Postcondition	None

C. View unpaid invoice

Use – Case	Content
Use-case name	View unpaid bills
Description	Allows students to view unpaid invoices
Actor	Student
Main scenario	<p>Step 1: The system displays the home page for logged-in students.</p> <p>Step 2: Students click on the Bills button.</p> <p>Step 3: Students click on the unpaid invoices button.</p> <p>Step 4: The system will display a list of students' unpaid bills in order from the most recent date to the top of the list.</p> <p>Step 5: Students click on any invoice.</p> <p>Step 6: The system will display details of that invoice</p>
Precondition	Students must successfully log in and click on the Bills button, then click on the unpaid bills button
Postcondition	None

4.3.6 View room information

Use – Case	Content
Use-case name	View room information
Description	Allows students to view room information when not logged in
Actor	Student
Main scenario	<p>Step 1: The system displays the home page for students who have not logged in.</p> <p>Step 2: Students click on the room type button.</p> <p>Step 3: The system will display the available room types in the dormitory.</p> <p>Step 4: Students click on a certain type of room that they want to see information about.</p> <p>Step 5: The system will display detailed information of that room type</p>
Precondition	Students must click on the room type button and click on the room type they want to view information
Postcondition	None

4.3.7 Manage student

A. Insert student

Use – Case	Content
Use-case name	Insert student
Description	Allows admin to insert student to room
Actor	Admin
Main scenario	<p>Step 1: The system displays the home page for admins.</p> <p>Step 2: Admin click on the room management button.</p> <p>Step 3: Admin click on the search room button.</p> <p>Step 4: The system will display the search box.</p>

	<p>Step 5: Admin enters information about the room to search.</p> <p>Step 6: Admin click search or press the enter key.</p> <p>Step 7: The system will display the room the admin searched for.</p> <p>Step 8: Admin selects the room displayed on the screen.</p> <p>Step 9: The system will display detailed information of the room you just selected.</p> <p>Step 10: Admin clicks the add student button.</p> <p>Step 11: Admin selects the students to add.</p> <p>Step 12: Admin click confirm.</p> <p>Step 13: After adding successfully, the system will notify you that it was successful</p>
Precondition	Admin must click on the room management button, then search for a room, then select the room and click add student
Postcondition	<ol style="list-style-type: none"> 1. If the insert is successful, the system will display a success message. 2. If the insert fails, the system will report an error

B. Search student

Use – Case	Content
Use-case name	Search student
Description	Allows admin to find students
Actor	Admin
Main scenario	Step 1: The system displays the home page for admins.

	<p>Step 2: Admin click on the search student button.</p> <p>Step 3: The system will display the search box.</p> <p>Step 4: Admin enters information in the search box.</p> <p>Step 5: The system will display the student the admin is looking for</p>
Precondition	Admin must click on the search student button
Postcondition	<ol style="list-style-type: none"> 1. If the search is successful, the system will display the student information that the admin is looking for. 2. If the search fails, the system will report an error

C. Edit student

Use – Case	Content
Use-case name	Edit student
Description	Allows admin to edit student information
Actor	Admin
Main scenario	<p>Step 1: The system displays the home page for admins.</p> <p>Step 2: Admin click on the edit student information button.</p> <p>Step 3: The system will display student information that the admin can edit.</p> <p>Step 4: Admin selects the information that needs to be edited.</p> <p>Step 5: Admin enters the information that needs to be edited.</p> <p>Step 6: Admin click confirm.</p>

	Step 7: After successfully editing, the system will display the student's information after editing
Precondition	Admin must click on the edit student information button
Postcondition	<ol style="list-style-type: none"> 1. If the editing is successful, the system will display the student information that the admin has edited. 2. If editing is not successful, the system will report an error

D. Delete student

Use – Case	Content
Use-case name	Delete student
Description	Allows admin to delete students
Actor	Admin
Main scenario	<p>Step 1: The system displays the home page for admins.</p> <p>Step 2: Admin click on the search student button.</p> <p>Step 3: The system will display the search box.</p> <p>Step 4: Admin enters information in the search box.</p> <p>Step 5: The system will display the student the admin is looking for.</p> <p>Step 6: Admin clicks on the student information displayed on the screen.</p> <p>Step 7: Admin press the delete button.</p> <p>Step 8: Admin click confirm.</p> <p>Step 9: After successful deletion, the system will notify success</p>

Precondition	Admin must click on the delete student button
Postcondition	<ol style="list-style-type: none"> 1. If the deletion is successful, the system will display a success message. 2. If deletion is not successful, the system will report an error

4.3.8 Manage rooms

A. Edit room

Use – Case	Content
Use-case name	Edit room
Description	Allows admin to edit room information
Actor	Admin
Main scenario	<p>Step 1: The system displays the home page for admins.</p> <p>Step 2: Admin click on the room management button.</p> <p>Step 3: Admin click on the edit room button.</p> <p>Step 4: The system will display a list of all rooms and arrange them by floor.</p> <p>Step 5: Admin selects the room to edit.</p> <p>Step 6: Admin click on the edit room information button.</p> <p>Step 7: The system will display all information about that room.</p> <p>Step 8: Admin selects the information that needs to be edited.</p> <p>Step 9: Admin enters new information.</p> <p>Step 10: Admin click save.</p> <p>Step 11: After successfully saving, the system will</p>

	display new information after the admin edits
Precondition	Admin must click on the room management button, then click on the edit room button and select the room whose information needs to be edited
Postcondition	<ol style="list-style-type: none"> 1. If editing is successful, the system will display a success message. 2. If editing is not successful, the system will report an error

B. Search room

Use – Case	Content
Use-case name	Search room
Description	Allows admin to edit room information
Actor	Admin
Main scenario	<p>Step 1: The system displays the home page for admins.</p> <p>Step 2: Admin click on the room management button.</p> <p>Step 3: Admin click on the search room button.</p> <p>Step 4: The system will display the search box.</p> <p>Step 5: Admin enters information about the room to search.</p> <p>Step 6: Admin click search or press the enter key.</p> <p>Step 7: The system will display the room the admin searched for</p>
Precondition	Admin must click on the room management button, then search for a room.
Postcondition	<ol style="list-style-type: none"> 1. If the search is successful, the system will display a success message.

	2. If the search fails, the system will report an error
--	---

C. Delete room

Use – Case	Content
Use-case name	Delete room
Description	Allows admin to delete room
Actor	Admin
Main scenario	<p>Step 1: The system displays the home page for admins.</p> <p>Step 2: Admin click on the room management button.</p> <p>Step 3: Admin click on the search room button.</p> <p>Step 4: The system will display the search box.</p> <p>Step 5: Admin enters information about the room to search.</p> <p>Step 6: Admin click search or press the enter key.</p> <p>Step 7: The system will display the room the admin searched for.</p> <p>Step 8: Admin click on the room displayed on the screen.</p> <p>Step 9: The system will display detailed information of the room.</p> <p>Step 10: Admin presses the delete room button.</p> <p>Step 11: Admin clicks the confirmation button.</p> <p>Step 12 After successful deletion, the system will notify you of success</p>

Precondition	Admin must click on the room management button, then click search for rooms, then select the room you want to delete and click delete room
Postcondition	<ol style="list-style-type: none"> 1. If the delete is successful, the system will display a success message. 2. If the delete fails, the system will report an error

4.3.9 Manage service

A. Approve room transfer request

Use – Case	Content
Use-case name	Approve room transfer request
Description	Allows admin to approve students' room transfer applications
Actor	Admin
Main scenario	<p>Step 1: The system displays the home page for administrators.</p> <p>Step 2: Admin click on the service management button.</p> <p>Step 3: Admin clicks on the room transfer request button.</p> <p>Step 4: The system will display a list of all student room transfer requests.</p> <p>Step 5: Admin selects the application that needs approval.</p> <p>Step 6: Admin click to accept or reject (if rejected, the reason will be added).</p> <p>Step 7: Admin click confirm.</p> <p>Step 8: After successfully recording, the system will notify success.</p>

Precondition	Admin must click on the service management button, then click on the room transfer request button, and select the form needs approval
Postcondition	<ol style="list-style-type: none"> 1. If approved successfully, the system will display a success message. 2. If approval is not successful, the system will report an error

B. Approve room repair requests

Use – Case	Content
Use-case name	Approve room repair requests
Description	Allows admin to approve students' room repair applications
Actor	Admin
Main scenario	<p>Step 1: The system displays the home page for administrators.</p> <p>Step 2: Admin click on the service management button.</p> <p>Step 3: Admin clicks on the room repair request button.</p> <p>Step 4: The system will display a list of all student room repair requests.</p> <p>Step 5: Admin selects the application that needs approval.</p> <p>Step 6: Admin click to accept or reject (if rejected, the reason will be added).</p> <p>Step 7: Admin click confirm.</p> <p>Step 8: After successfully recording, the system will notify success</p>

Precondition	Admin must click on the service management button, then click on the room repair request button, and select the form needs approval
Postcondition	<ol style="list-style-type: none"> 1. If approved successfully, the system will display a success message. 2. If approval is not successful, the system will report an error

C. Approve room check-out request

Use – Case	Content
Use-case name	Approve room check-out request
Description	Allows admin to approve students' room check-out applications
Actor	Admin
Main scenario	<p>Step 1: The system displays the home page for administrators.</p> <p>Step 2: Admin click on the service management button.</p> <p>Step 3: Admin clicks on the room check-out request button.</p> <p>Step 4: The system will display a list of all student room check-out requests.</p> <p>Step 5: Admin selects the application that needs approval.</p> <p>Step 6: Admin click to accept or reject (if rejected, the reason will be added).</p> <p>Step 7: Admin click confirm.</p> <p>Step 8: After successfully recording, the system will notify success</p>

Precondition	Admin must click on the service management button, then click on the room check-out request button, and select the form needs approval
Postcondition	<ol style="list-style-type: none"> 1. If approved successfully, the system will display a success message. 2. If approval is not successful, the system will report an error

4.3.10 Manage invoice

A. Create invoice

Use – Case	Content
Use-case name	Create invoice
Description	Allows creating invoices
Actor	Admin
Main scenario	<p>Step 1: The system displays the home page for administrators.</p> <p>Step 2: Admin click on the manage invoice button.</p> <p>Step 3: Admin click create invoice.</p> <p>Step 4: The system will display the information table that needs to be filled in when creating the bill.</p> <p>Step 5: Admin enters the necessary information.</p> <p>Step 6: Admin clicks the create button.</p> <p>Step 7: After successfully creating, the system will notify success</p>
Precondition	Admin must click on the invoice management button, then click on the create invoice button
Postcondition	<ol style="list-style-type: none"> 1. If the invoice is successfully created, the system will display a success message. 2. If creating an invoice fails, the system will

	report an error
--	-----------------

B. Approve bills (when students have paid)

Use – Case	Content
Use-case name	Approve bills
Description	Allows approval of bills
Actor	Admin
Main scenario	<p>Step 1: The system displays the home page for administrators.</p> <p>Step 2: Admin click on the manage bills button.</p> <p>Step 3: Admin click approve bill.</p> <p>Step 4: The system will display a list of student bills.</p> <p>Step 5: Admin selects the paid student bill.</p> <p>Step 6: Admin click paid.</p> <p>Step 7: After successfully updating the invoice status, the system will notify you of success and change the status of the invoice</p>
Precondition	Admin must click on the bill management button, then click on the approve bill button
Postcondition	<ol style="list-style-type: none"> If the bills are approved successfully, the system will display a success notification and update the bill's status. If the bills are not approved successfully, the system will report an error
Precondition	Admin must click on the add student button
Postcondition	<ol style="list-style-type: none"> If the student is successfully added, the system will display a success message. If adding fails, the system will display an error

	message
--	---------

4.3.11 Logout

Use – Case	Content
Use-case name	Login
Description	Allows users to log out
Actor	Admin/Student
Main scenario	<p>Step 1: The system displays the home page for logged-in users.</p> <p>Step 2: The user presses the logout button.</p> <p>Step 3: After successfully logging out, the system will return the user to the home page without logging in</p>
Precondition	The user must click on the logout button
Postcondition	None

4.4 Activity Diagram

Activity Diagram for Register

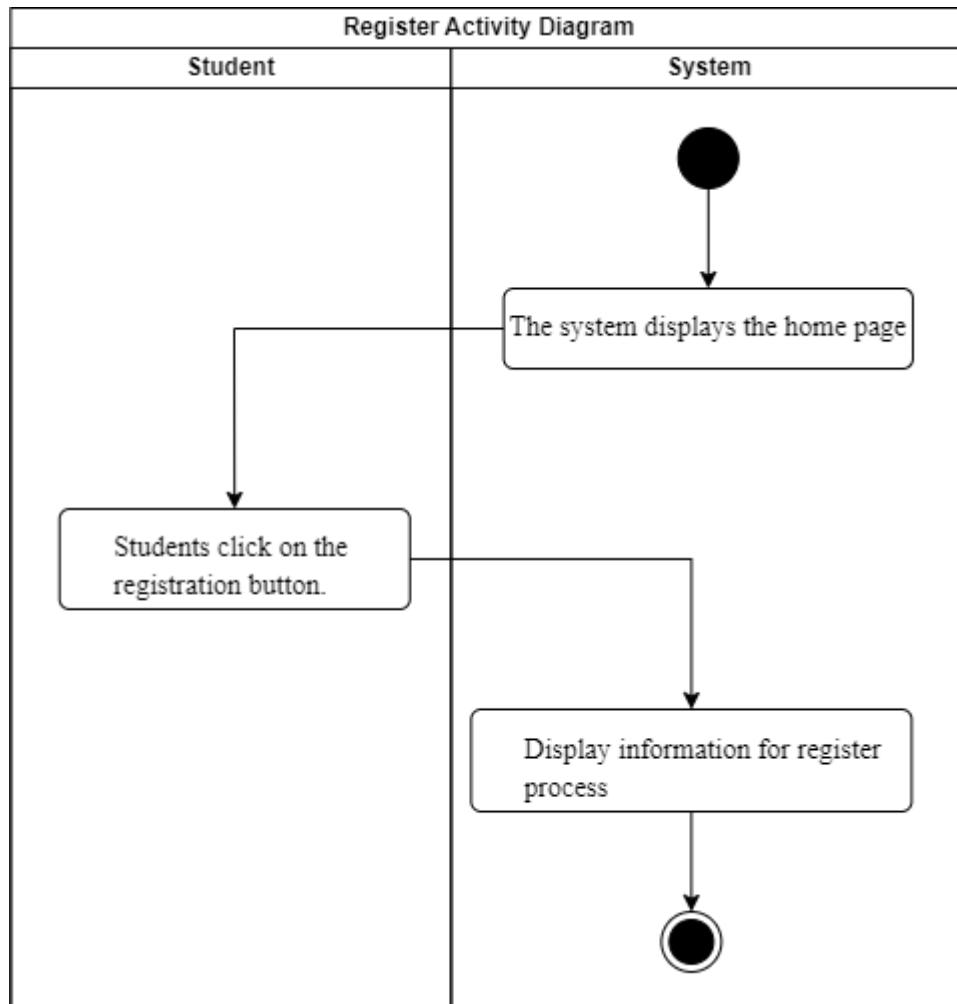


Figure 4.3 Activity Diagram for Register

Activity Diagram for Login

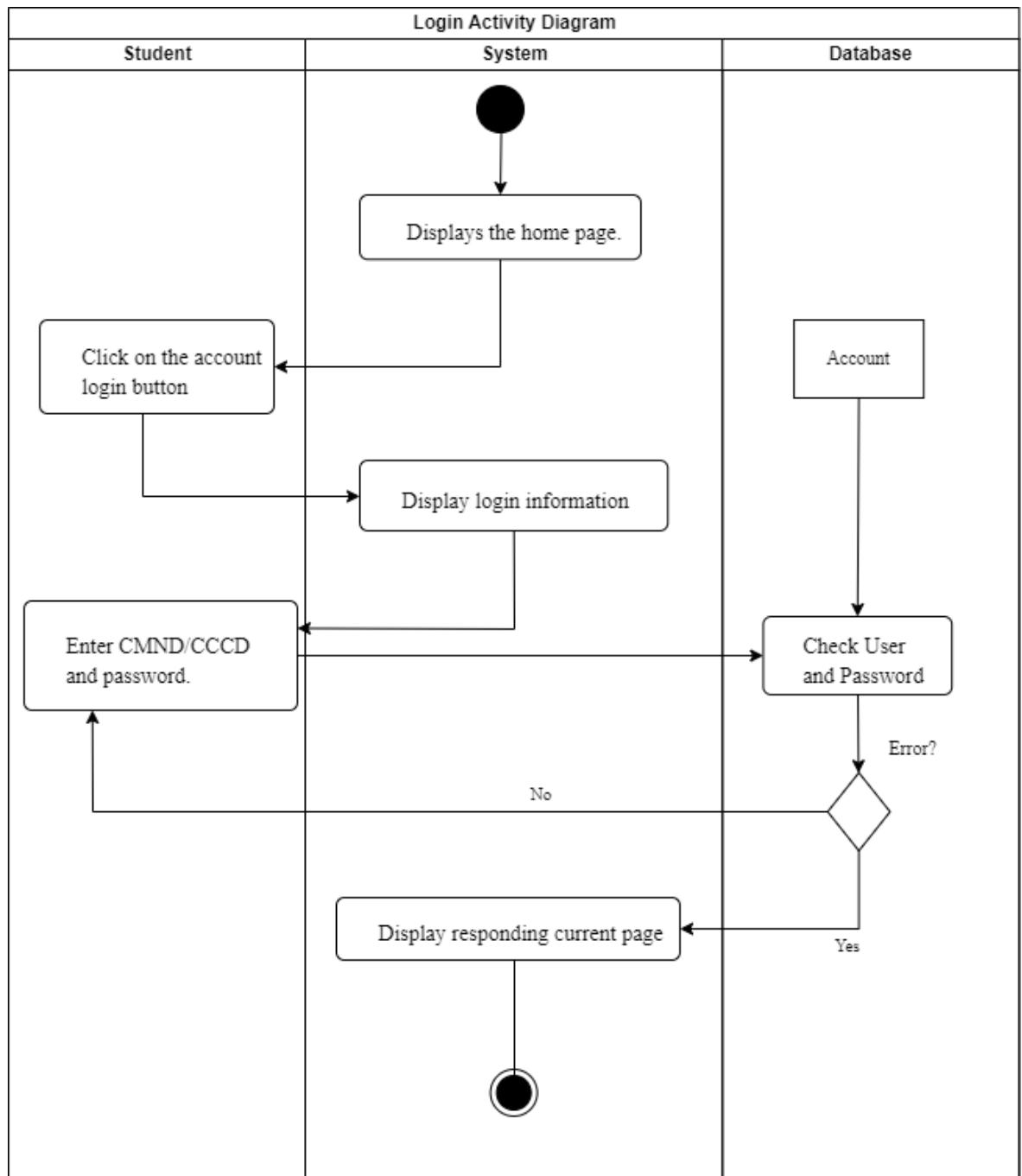


Figure 4.4 Activity Diagram for Login

Activity Diagram for View student information

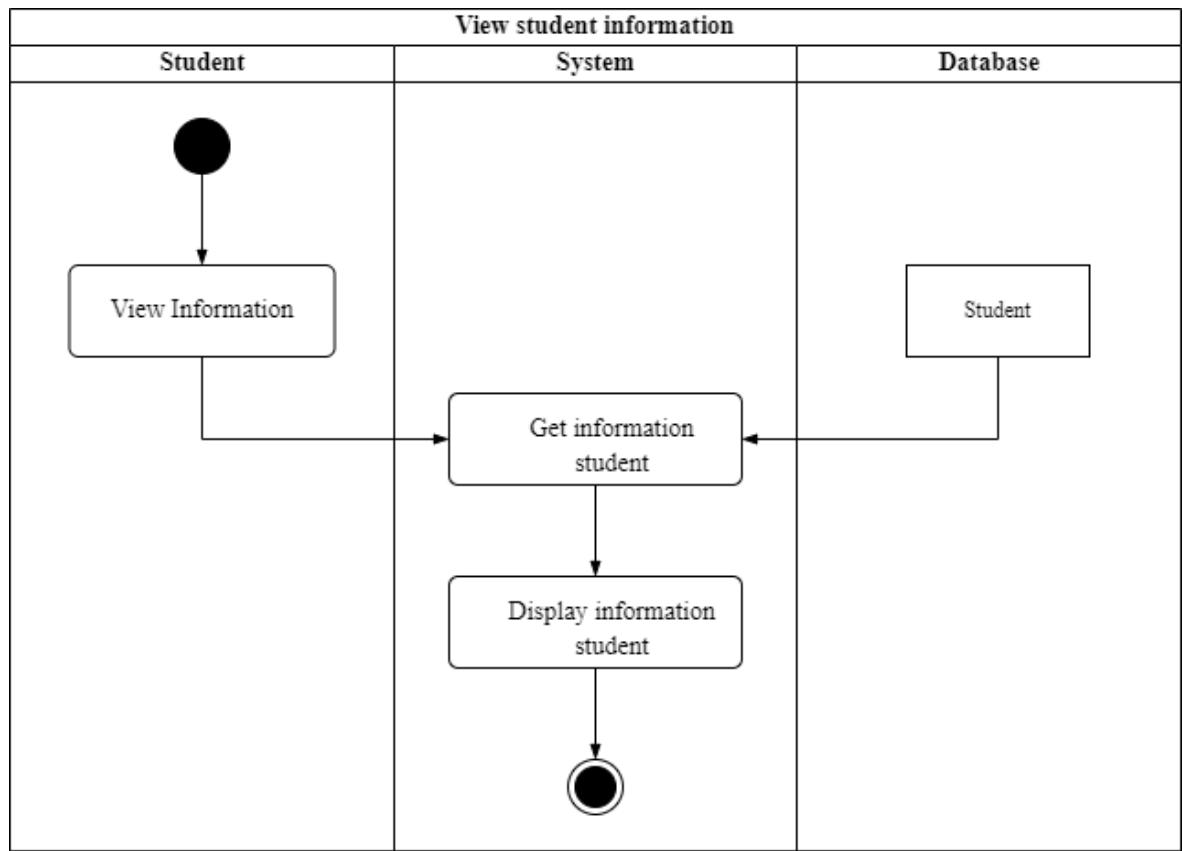


Figure 4.5 Activity Diagram for View student information

Activity Diagram for Change account password

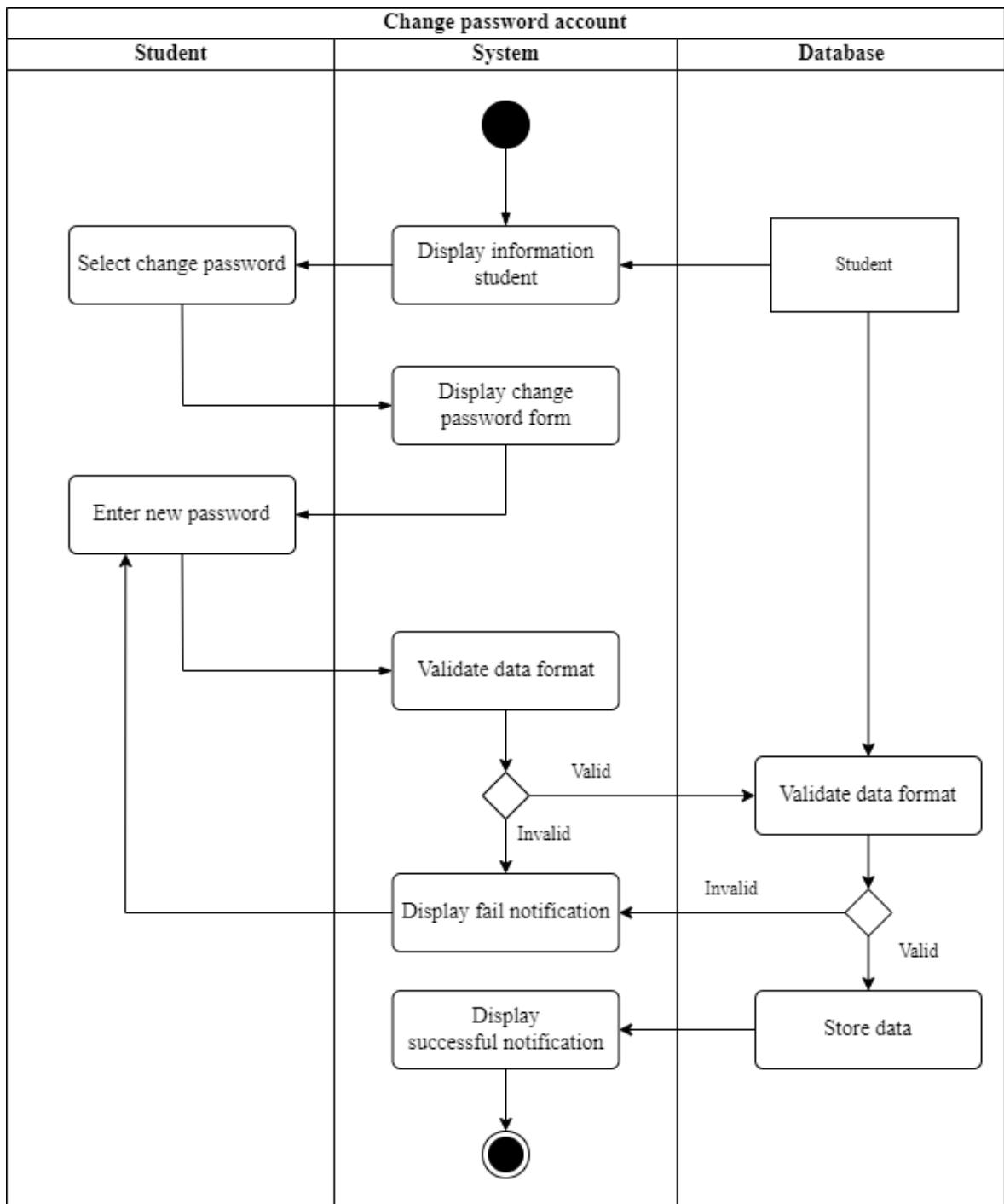


Figure 4.6 Activity Diagram for change account password

Activity Diagram for Change bank account number

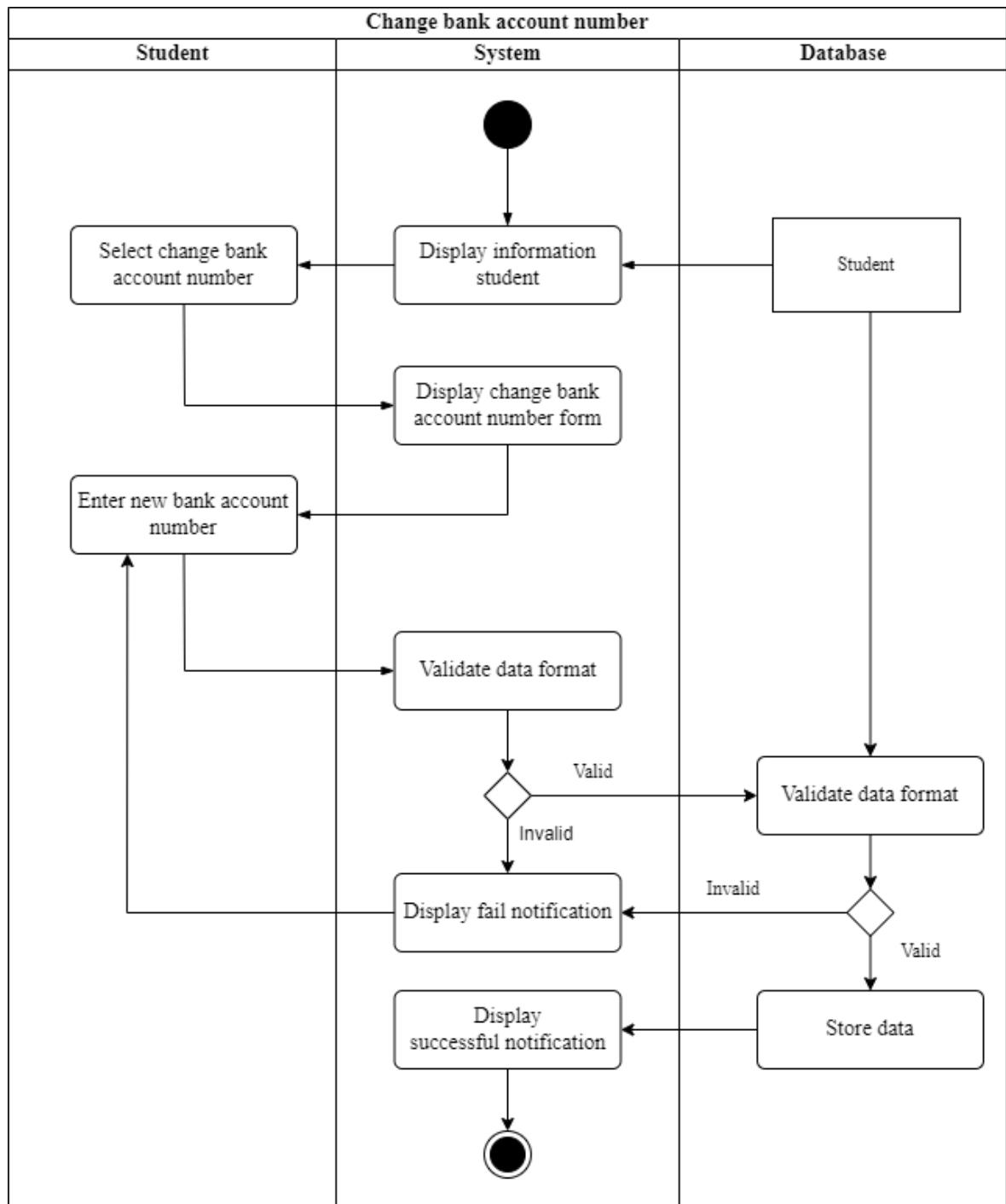


Figure 4.7 Activity Diagram for Change bank account number

Activity Diagram for Submit room transfer request

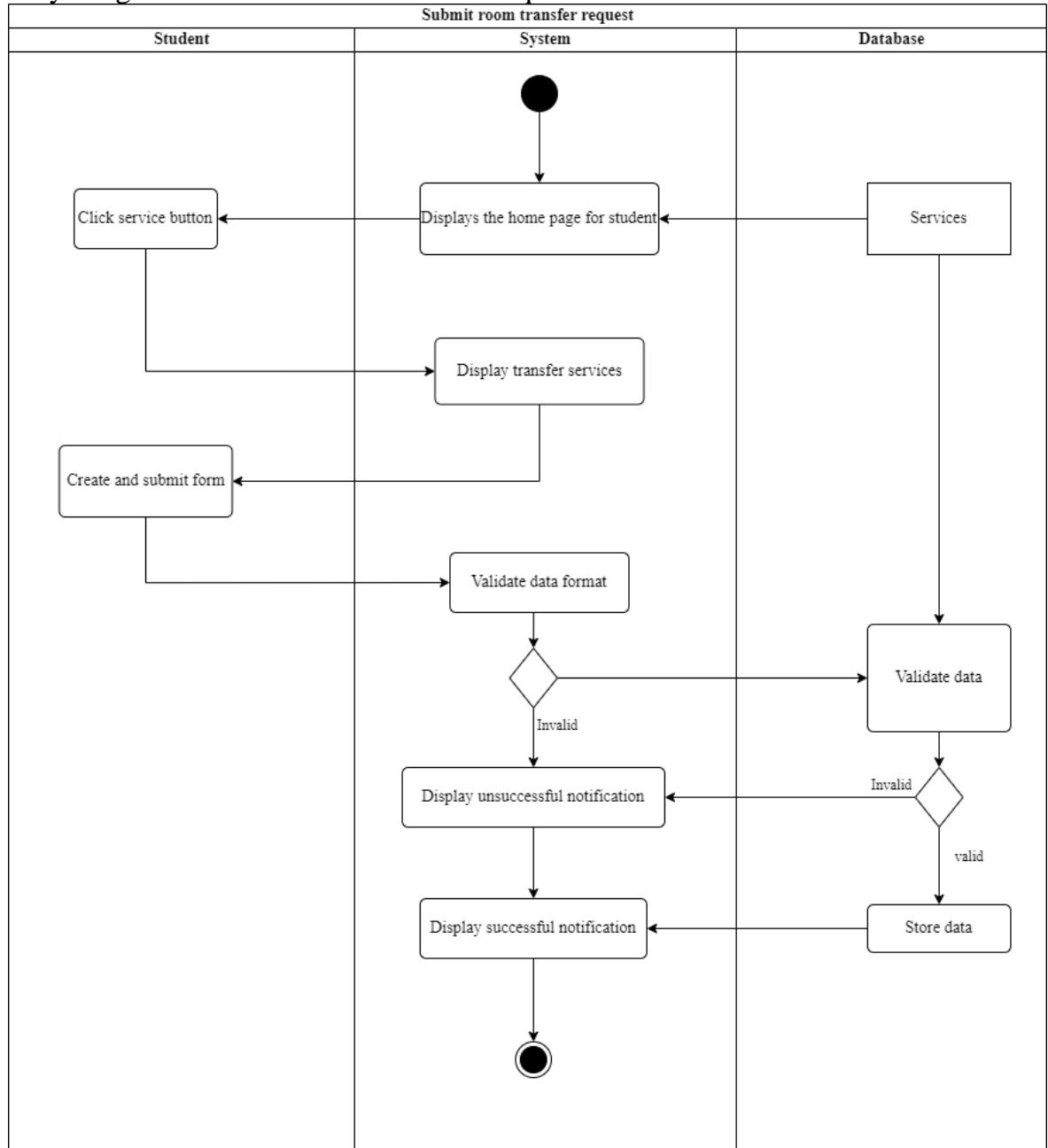


Figure 4.8 Activity Diagram for Submit room transfer request

Activity Diagram for Submit room repair request

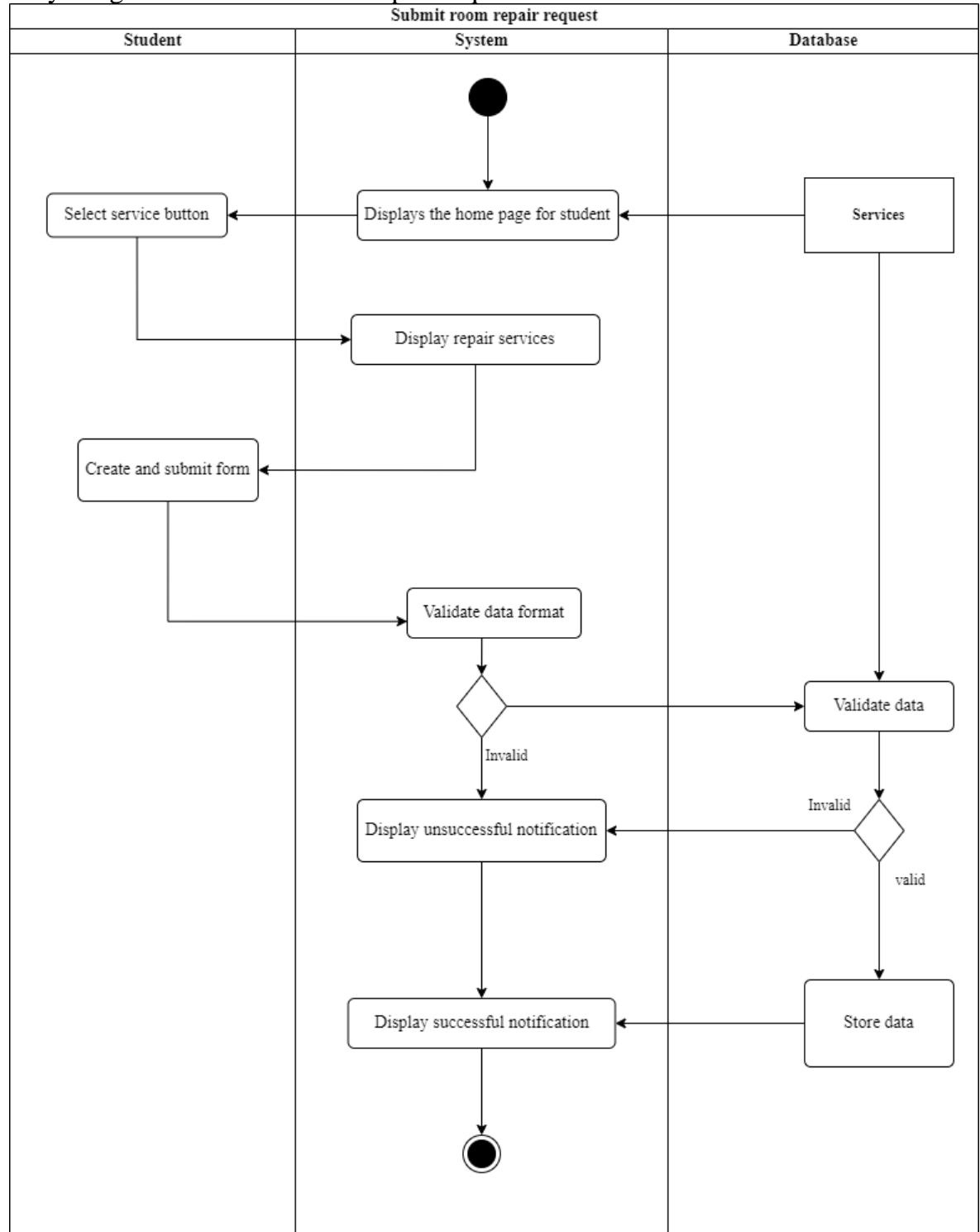


Figure 4.9 Activity Diagram for Submit room repair request

Activity Diagram for Submit room check-out request

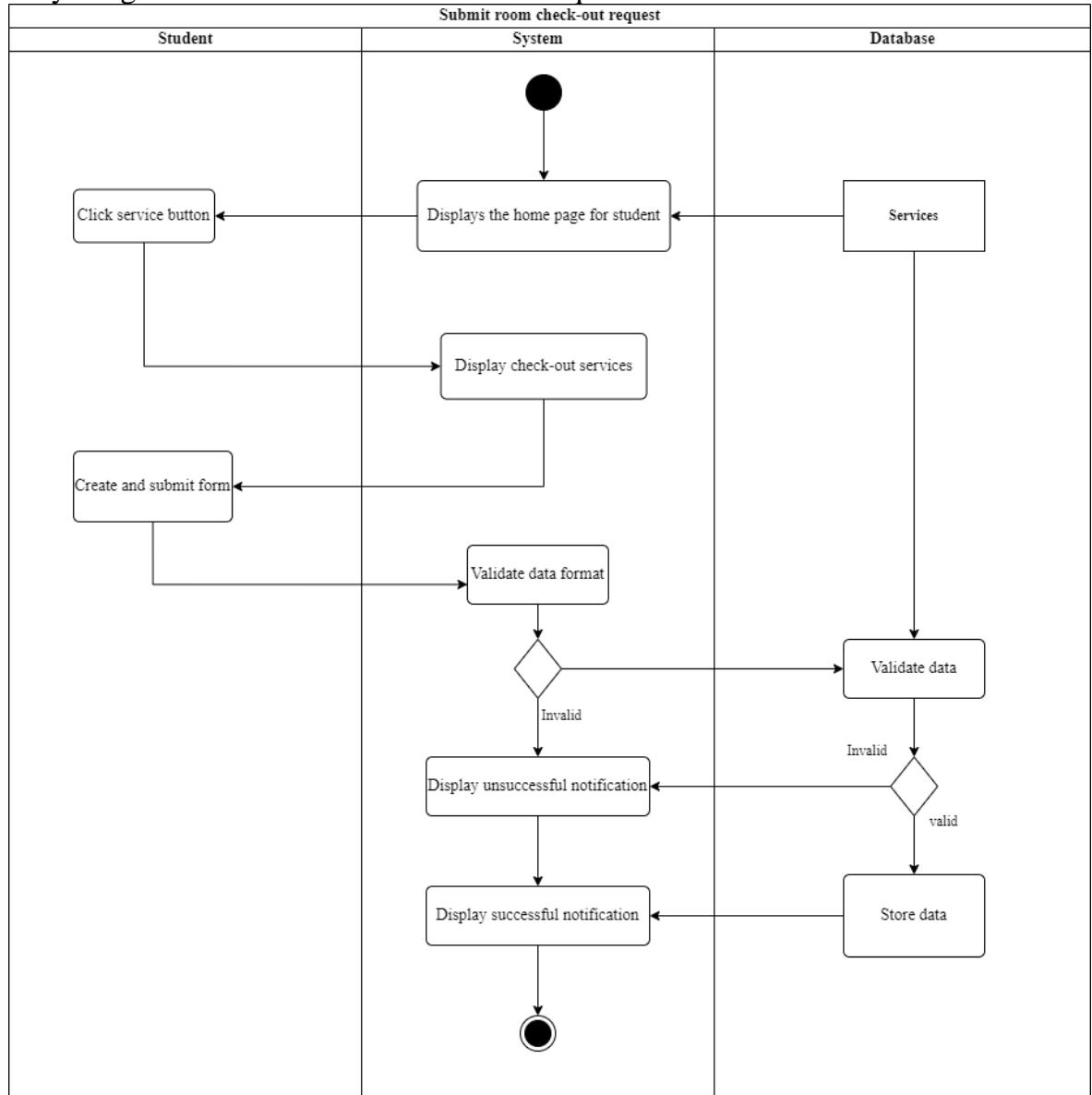


Figure 4.10 Activity Diagram for Submit room check-out request

Activity Diagram for Search room

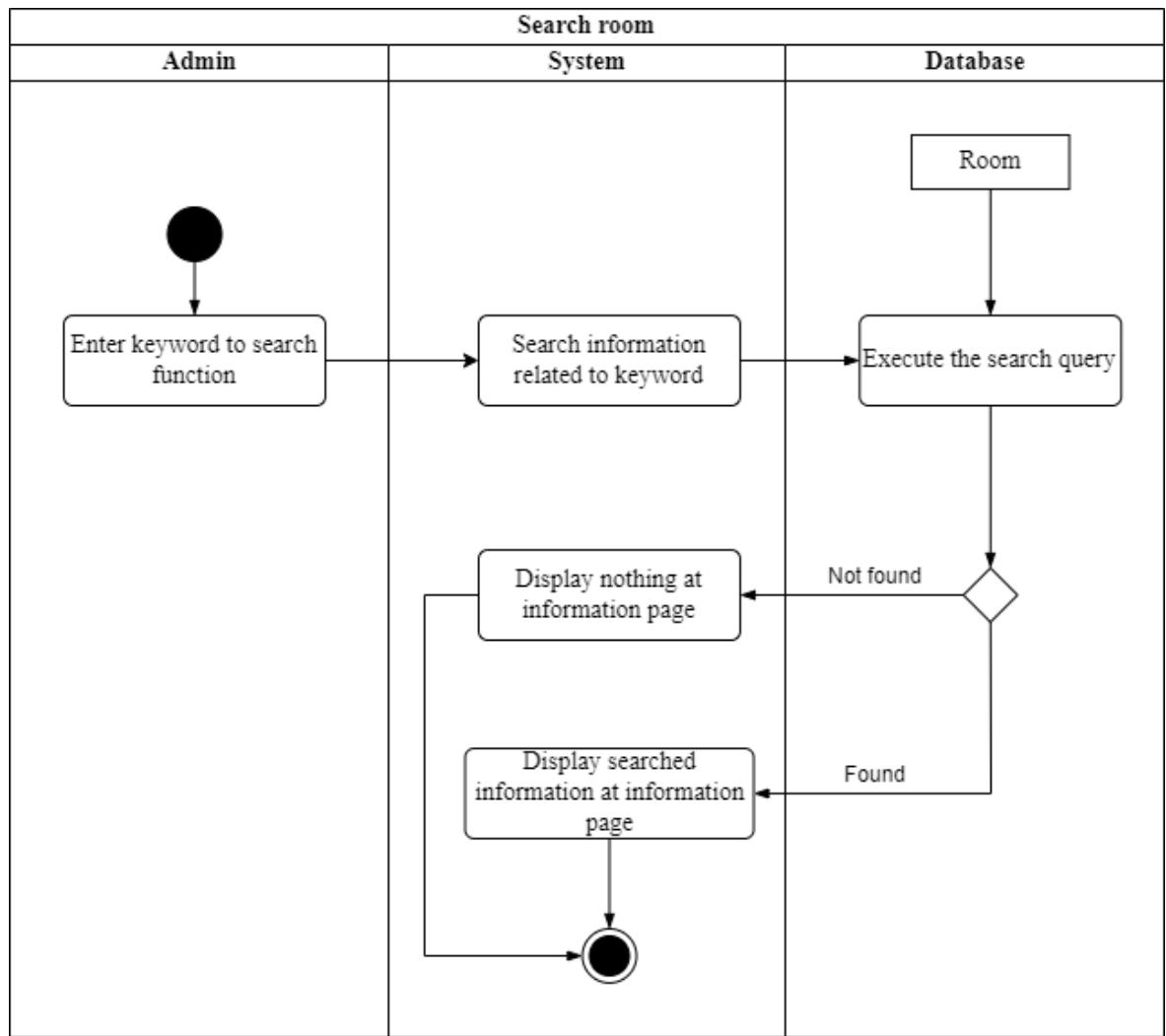


Figure 4.11 Activity Diagram for Search room

Activity Diagram for Edit room

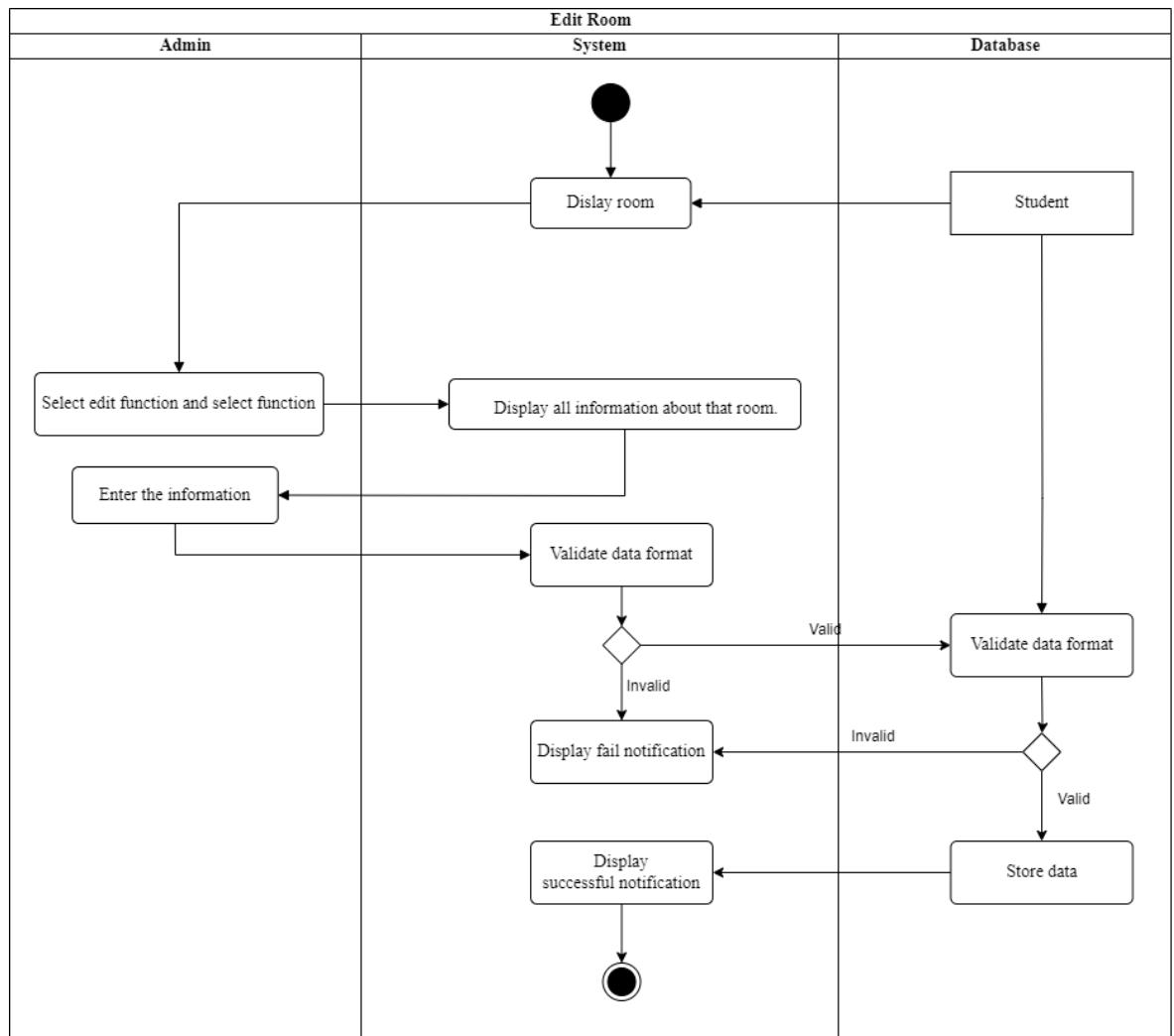


Figure 4.12 Activity Diagram for Edit room

Activity Diagram for Delete room

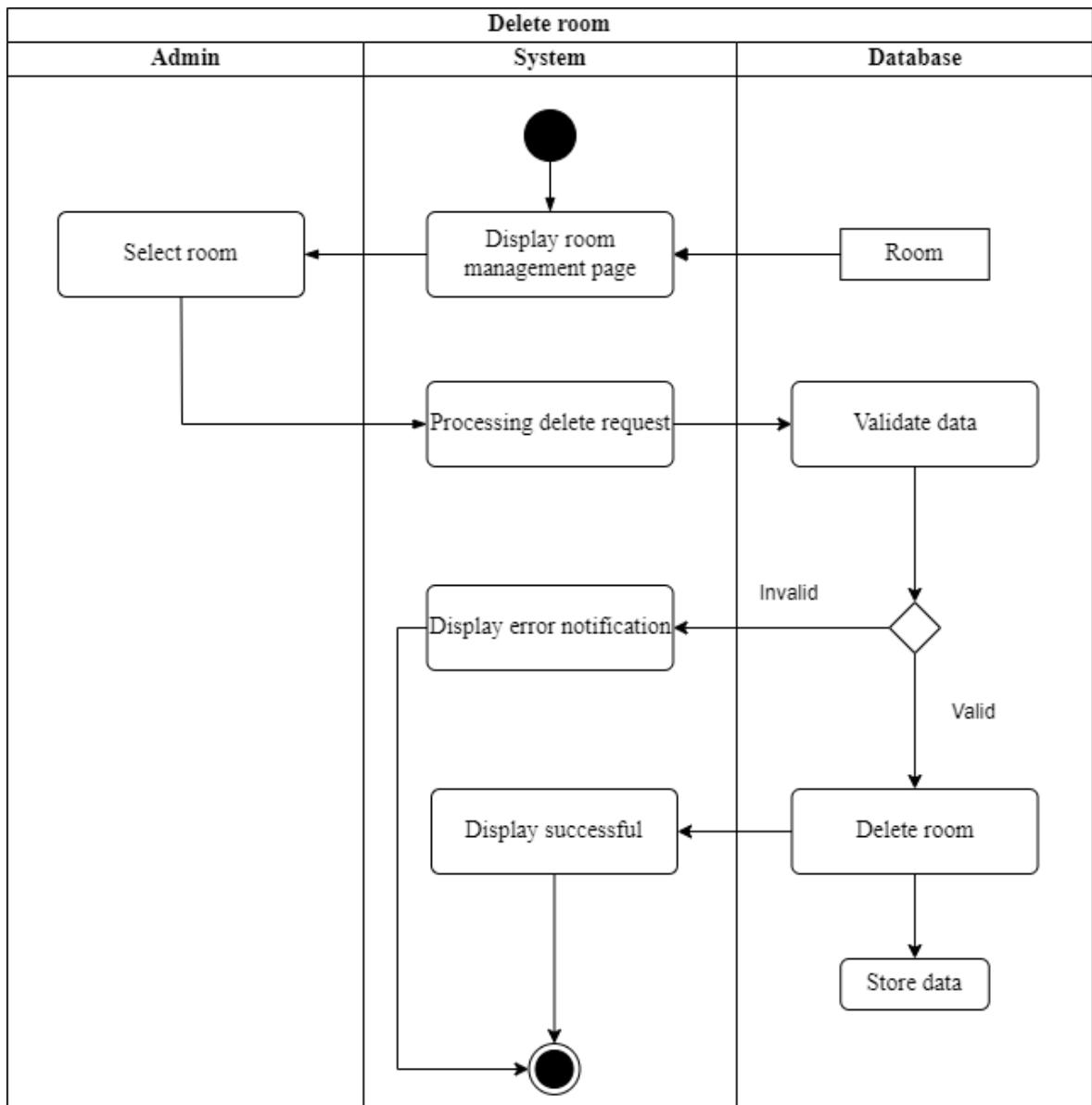


Figure 4.13 Activity Diagram for Delete room

Activity Diagram for Insert student

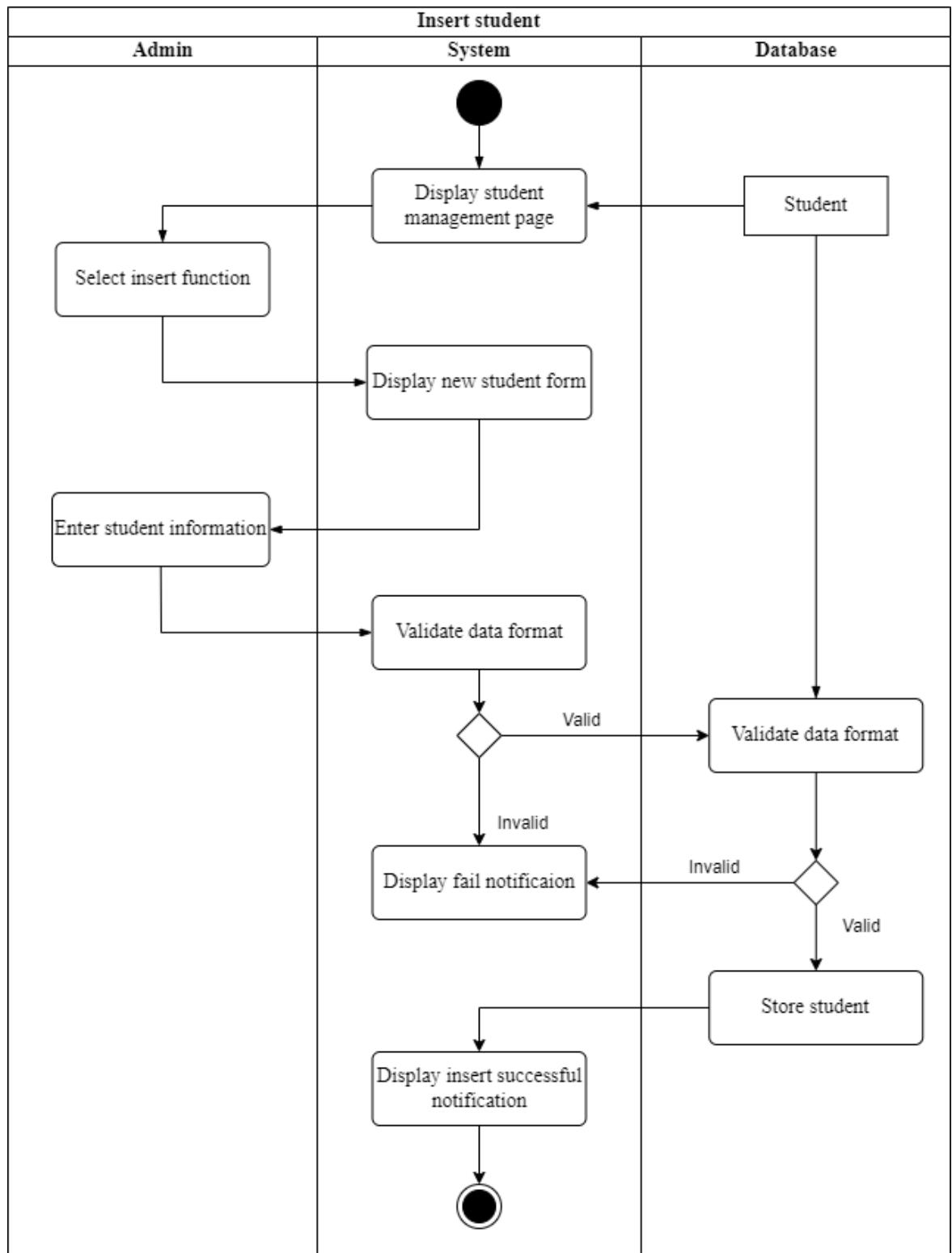


Figure 4.14 Activity Diagram for View room information

Activity Diagram for Search student

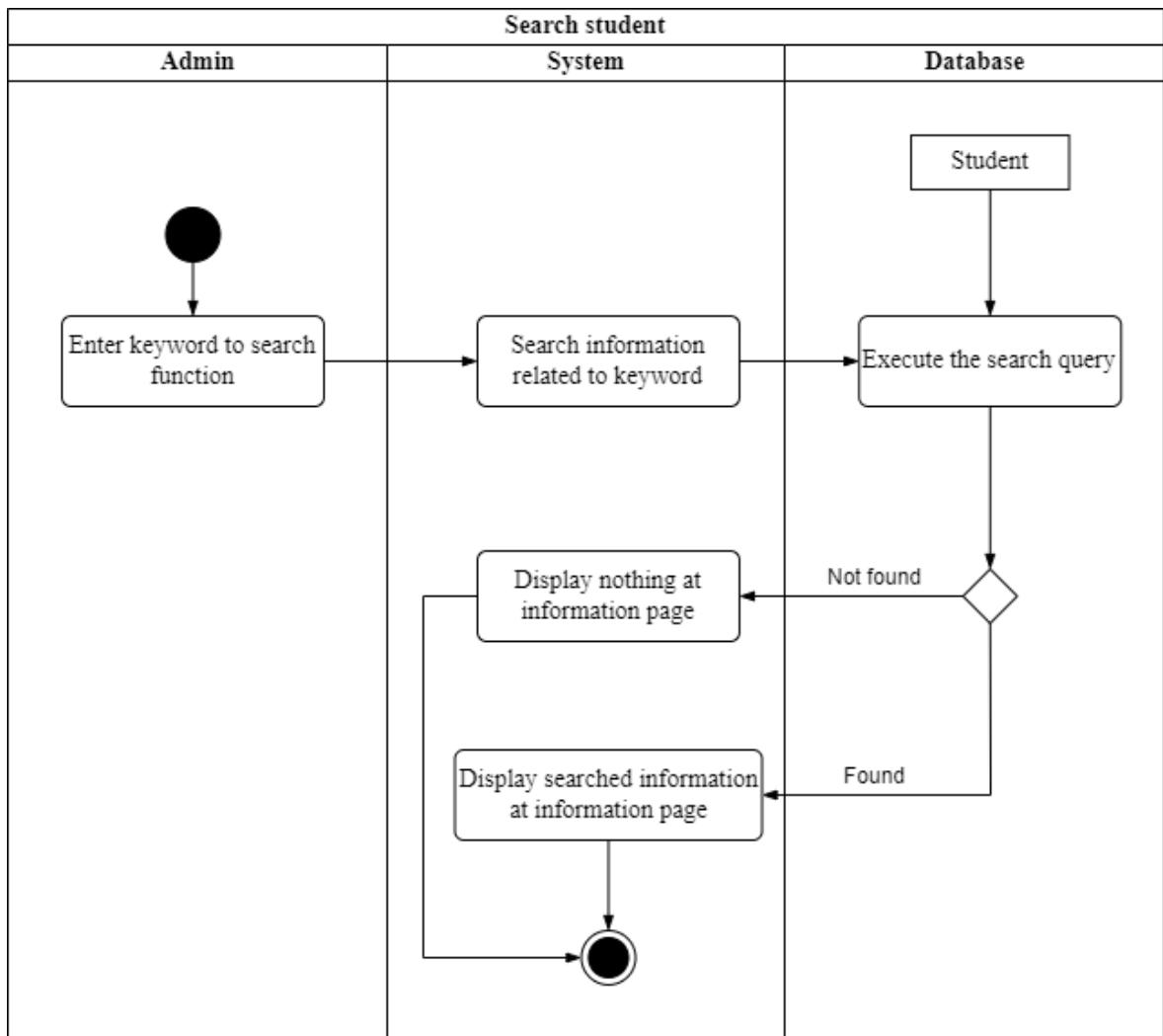


Figure 4.15 Activity Diagram for Search student

Activity Diagram for Edit student

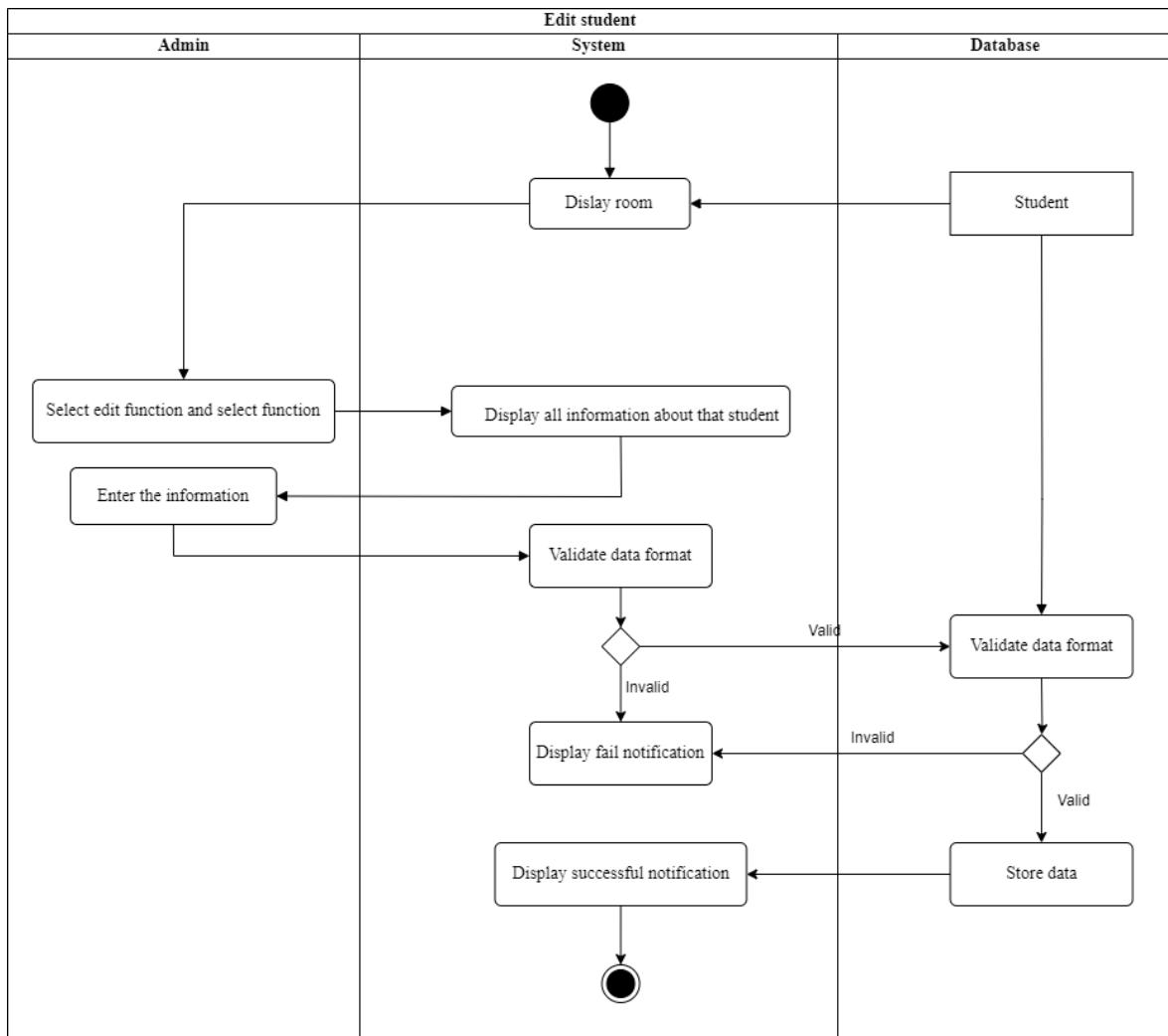


Figure 4.16 Activity Diagram for Edit student

Activity Diagram for Delete student

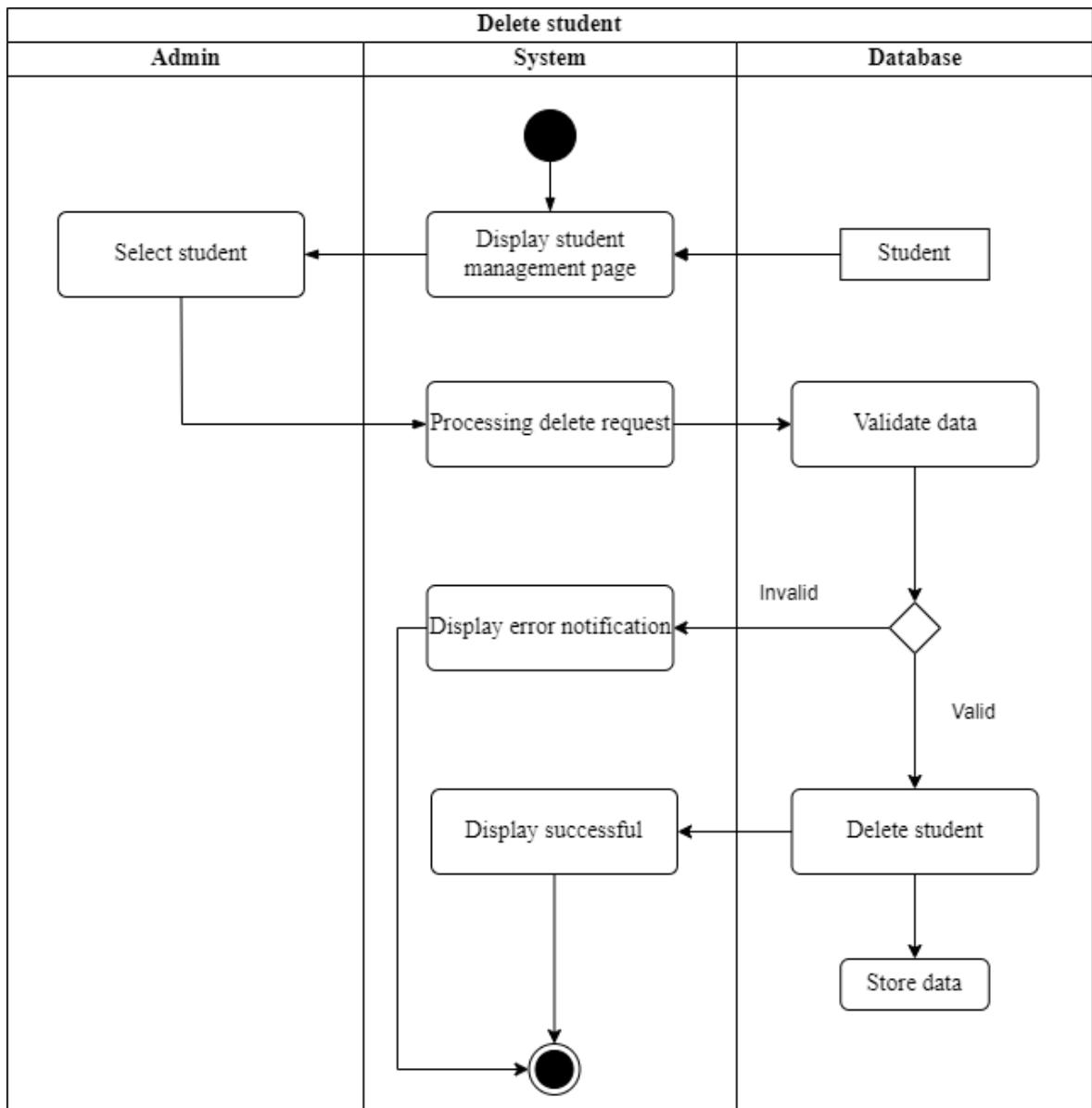


Figure 4.17 Activity Diagram for Edit student

Activity Diagram for Approve room transfer request

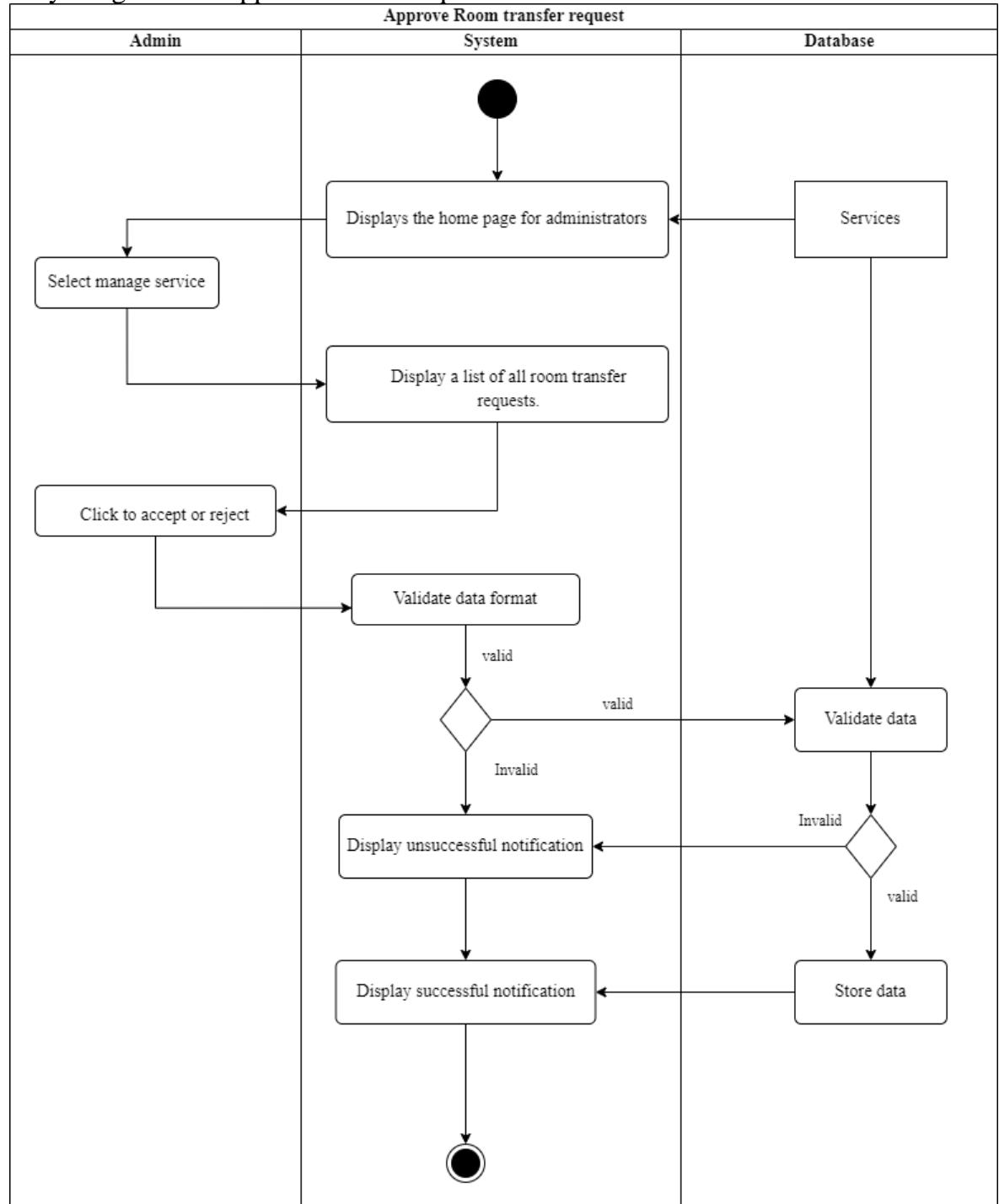


Figure 4.18 Activity Diagram for Approve room transfer request

Activity Diagram for Approve repair request

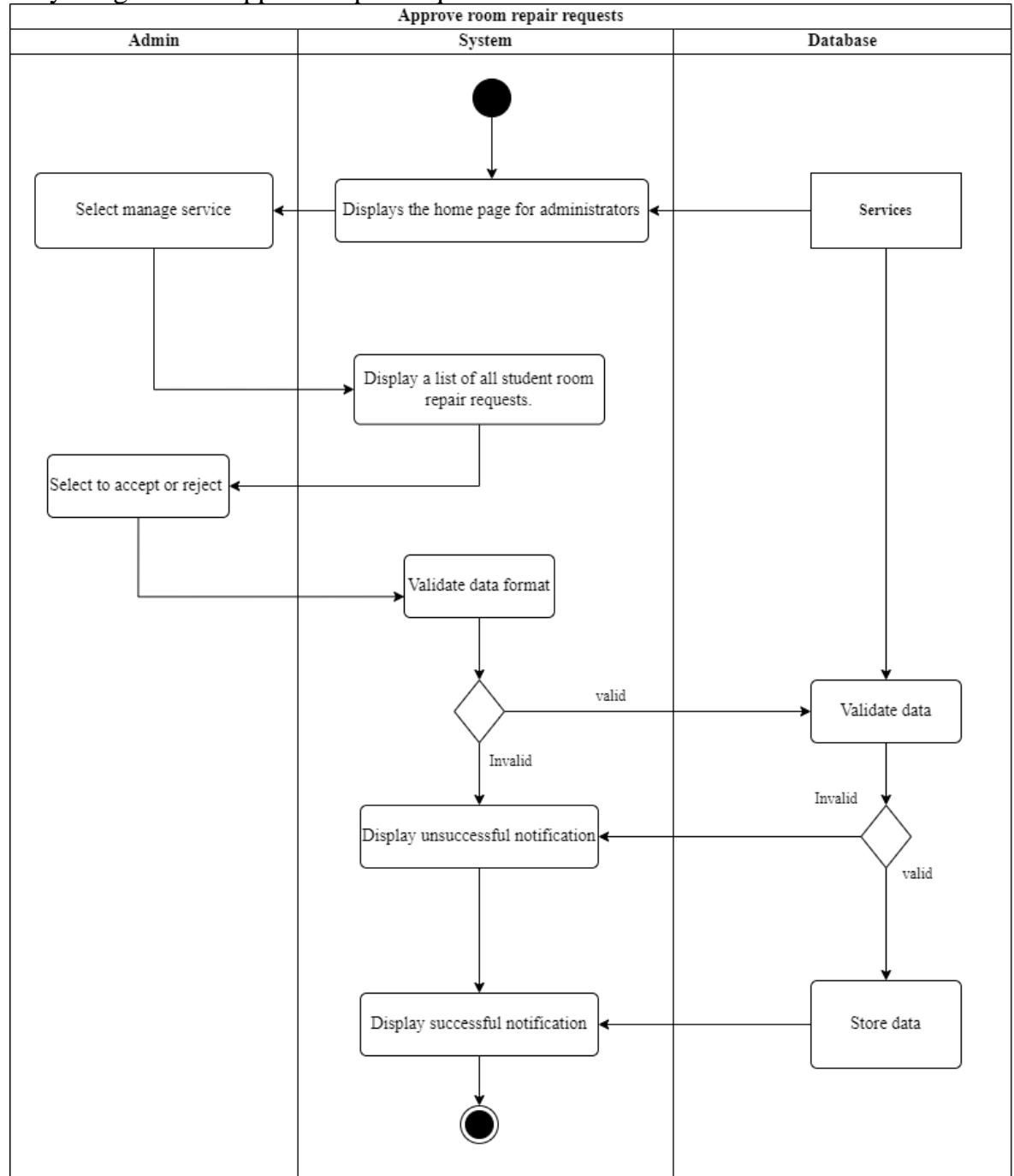


Figure 4.19 Activity Diagram for Approve room repair request

Activity Diagram for Approve check-out request

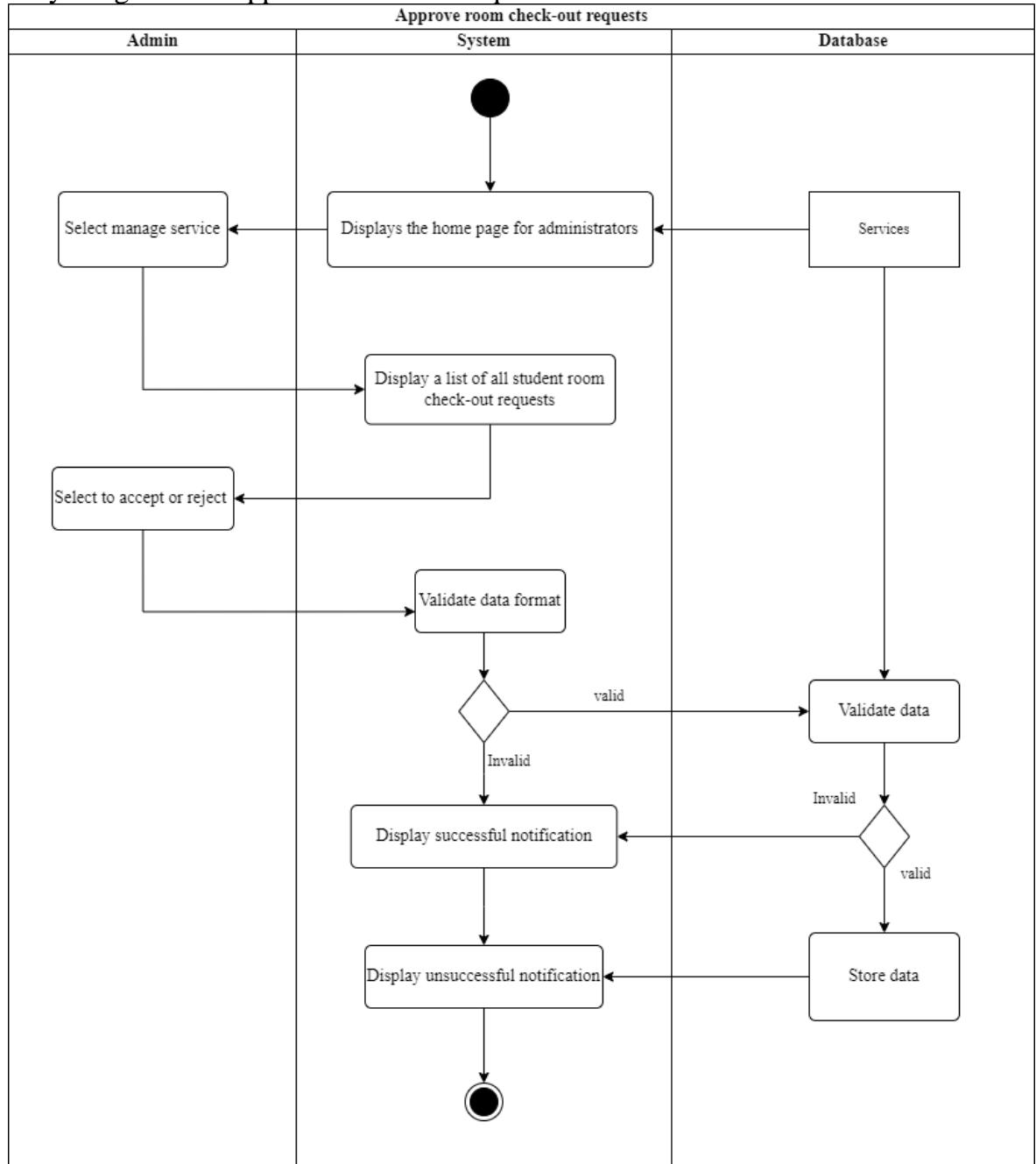


Figure 4.20 Activity Diagram for Approve Room check-out request

Activity Diagram for Create invoice

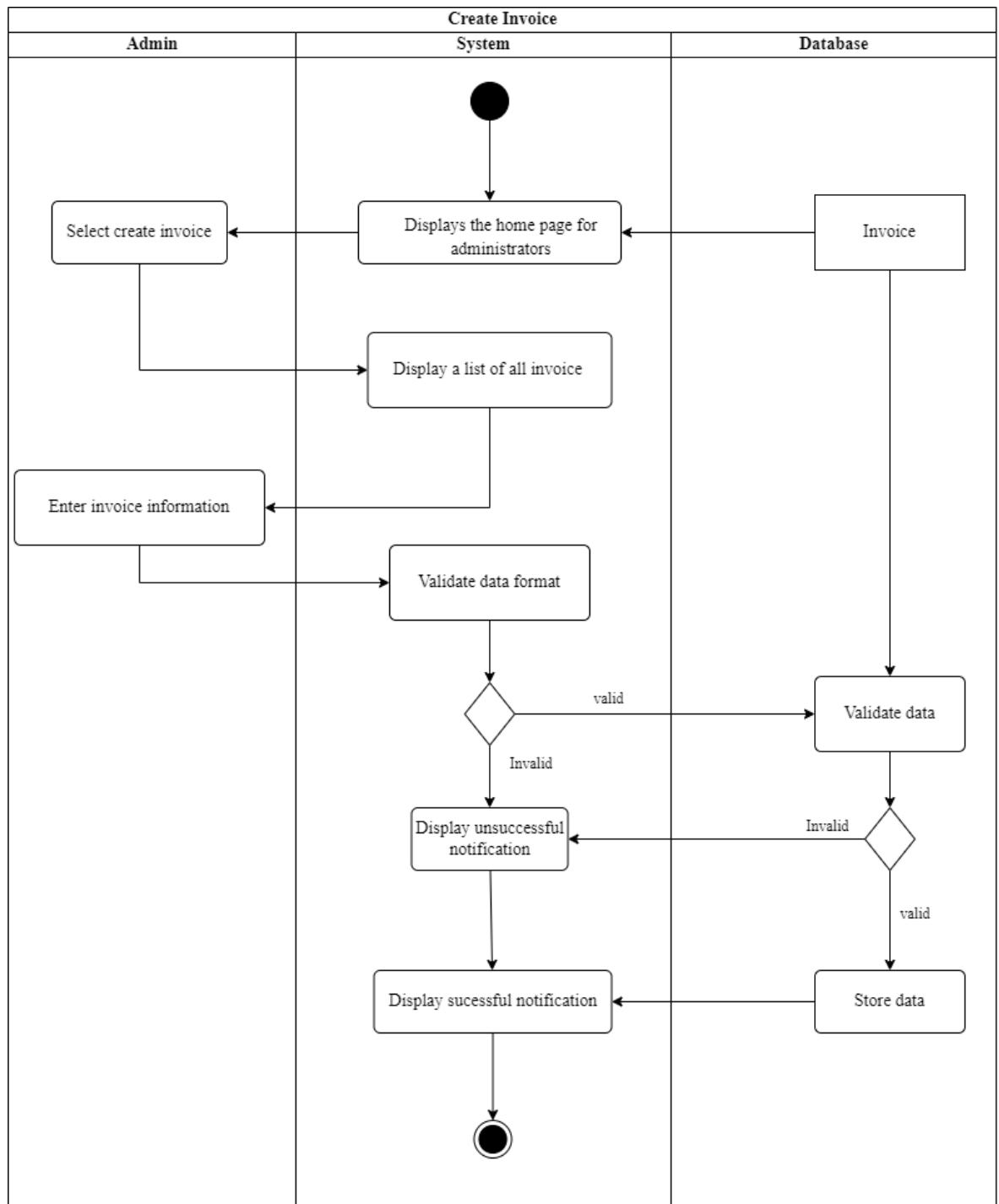


Figure 4.21 Activity Diagram for Create invoice

Activity Diagram for Approve invoice

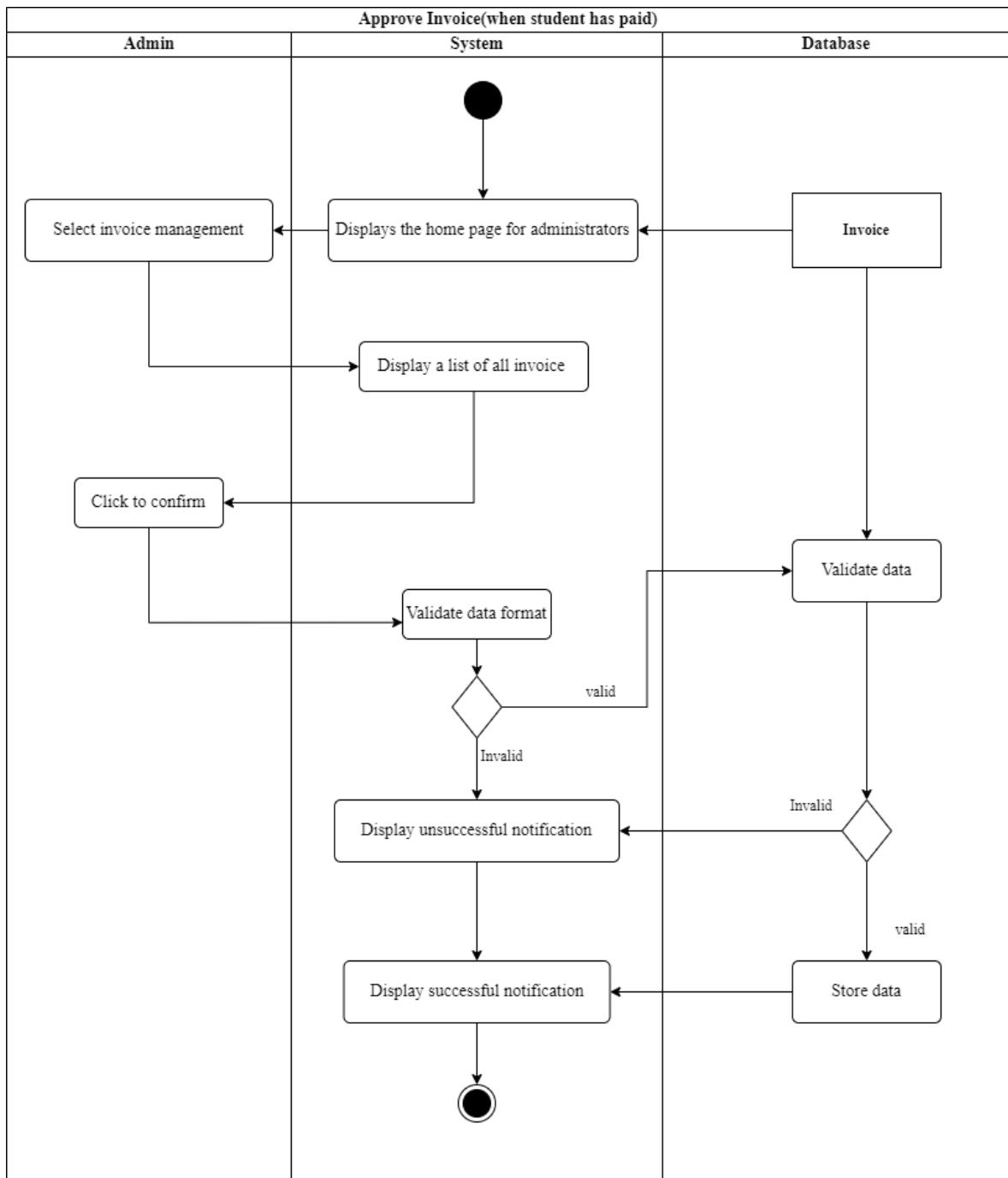


Figure 4.22 Activity Diagram for Approve invoice

Activity Diagram for Logout

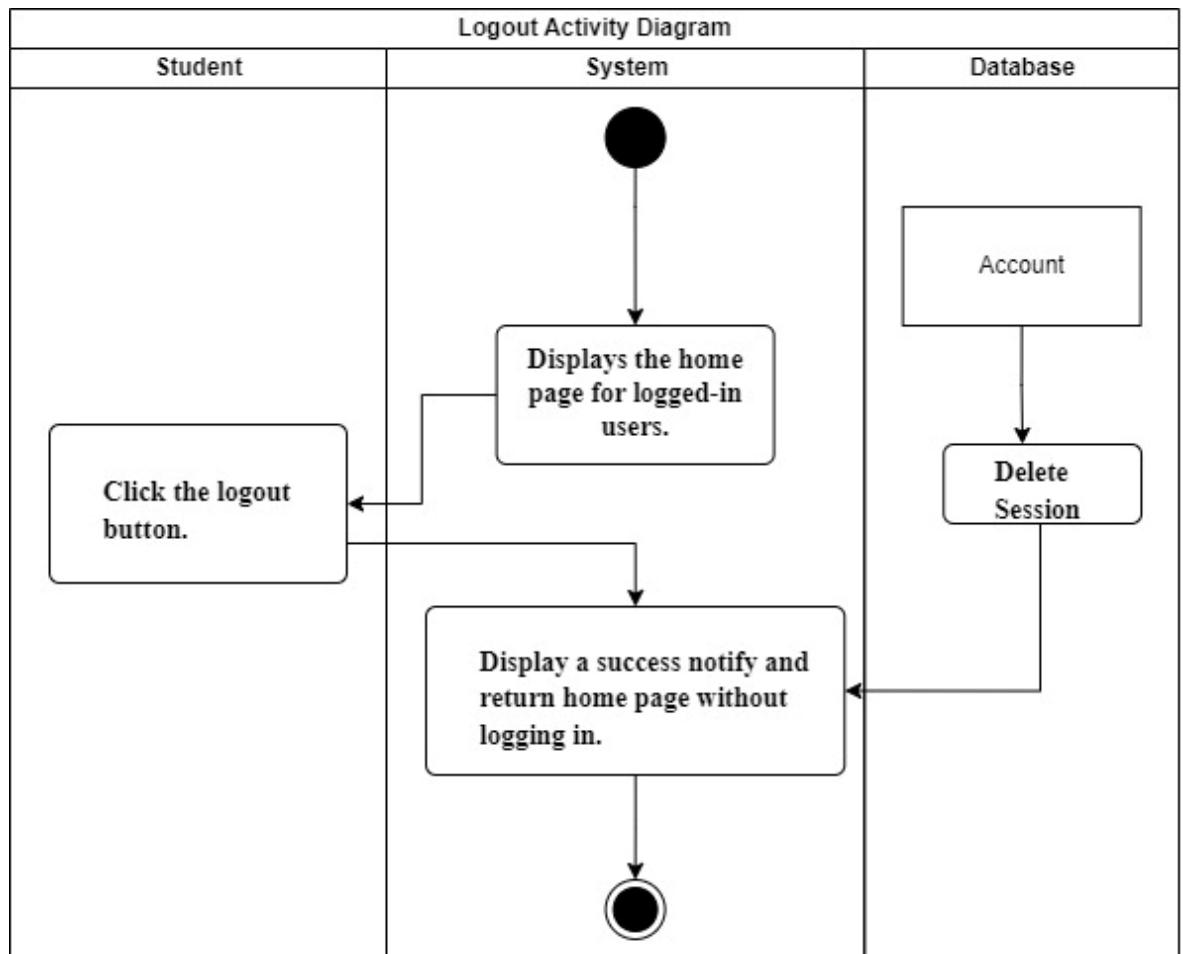


Figure 4.23 Activity Diagram for Logout

4.5 Sequence Diagram and Class Diagram

Login

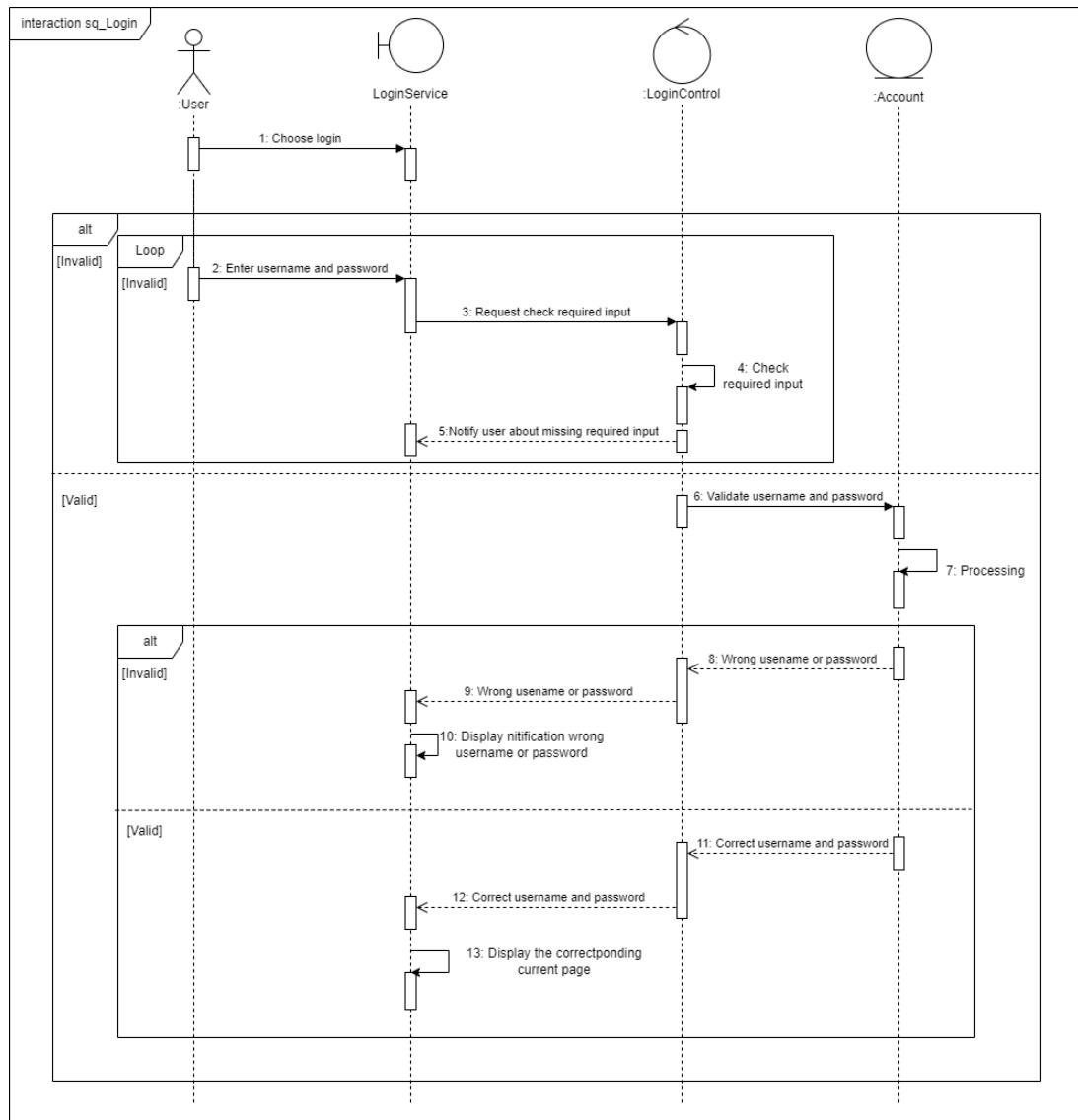


Figure 4.24 Sequence Diagram for Login

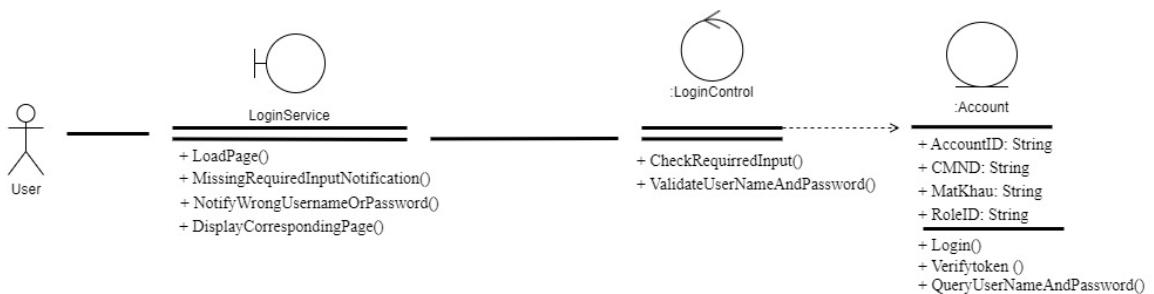


Figure 4.25 Class Diagram for Login

View account information

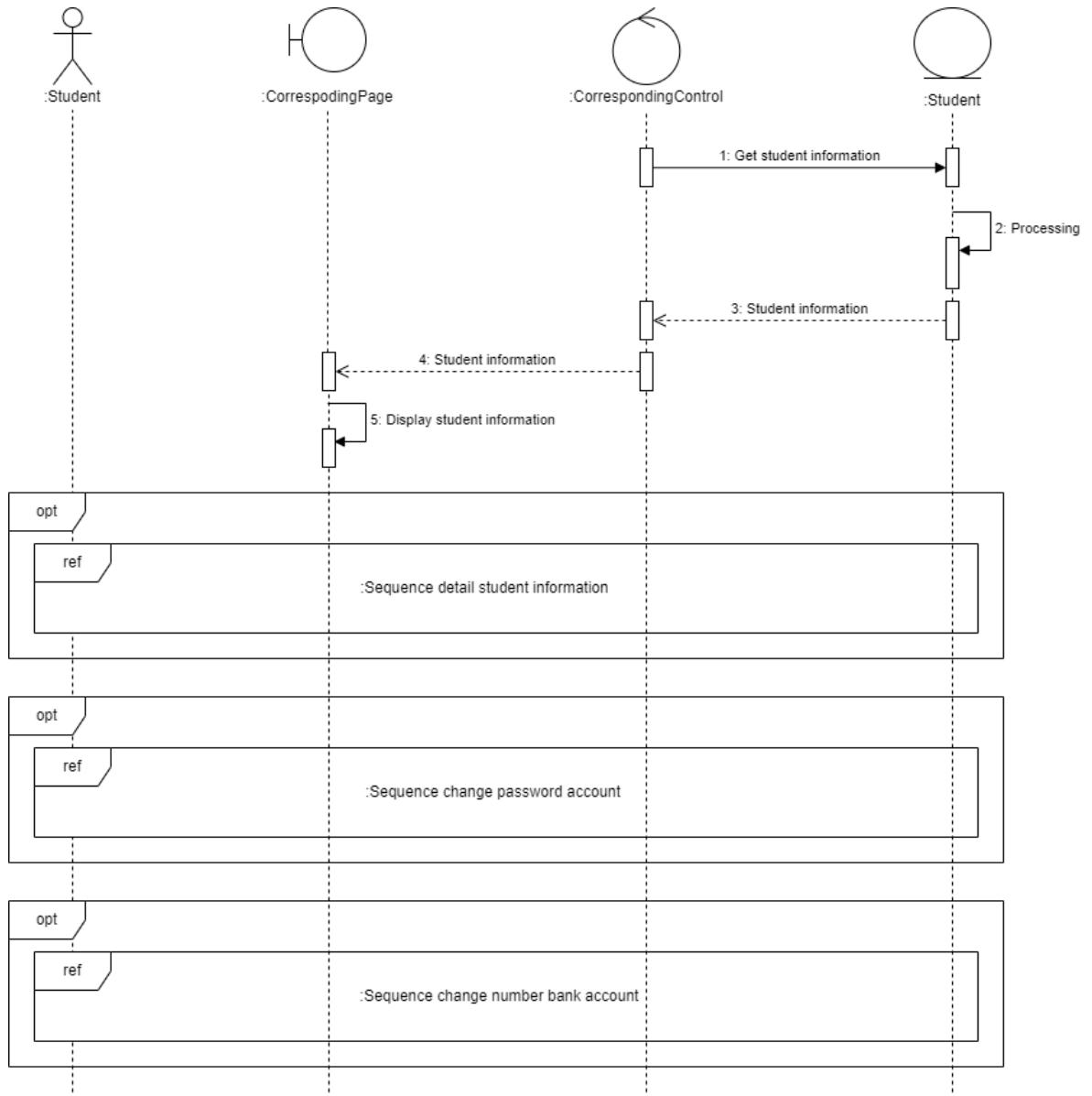


Figure 4.26 Sequence Diagram for View account information

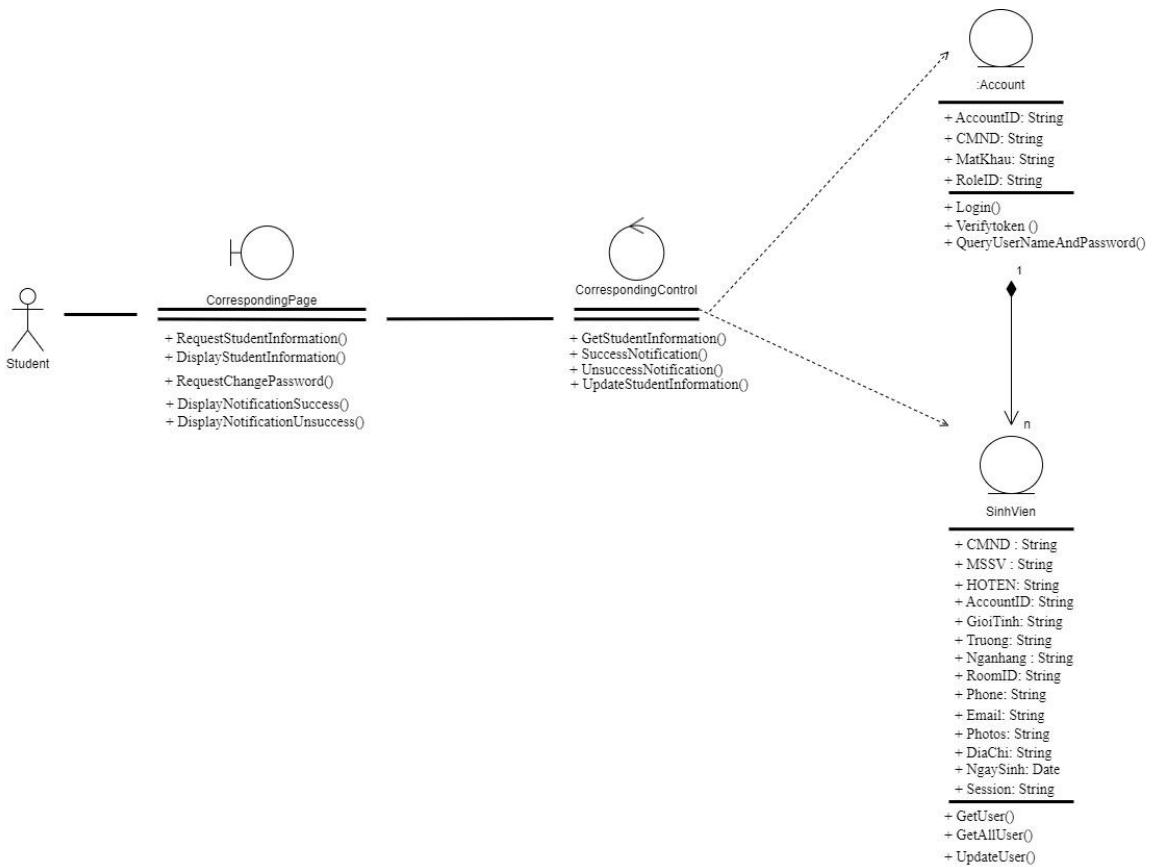


Figure 4.27 Class Diagram for View account information

View student information

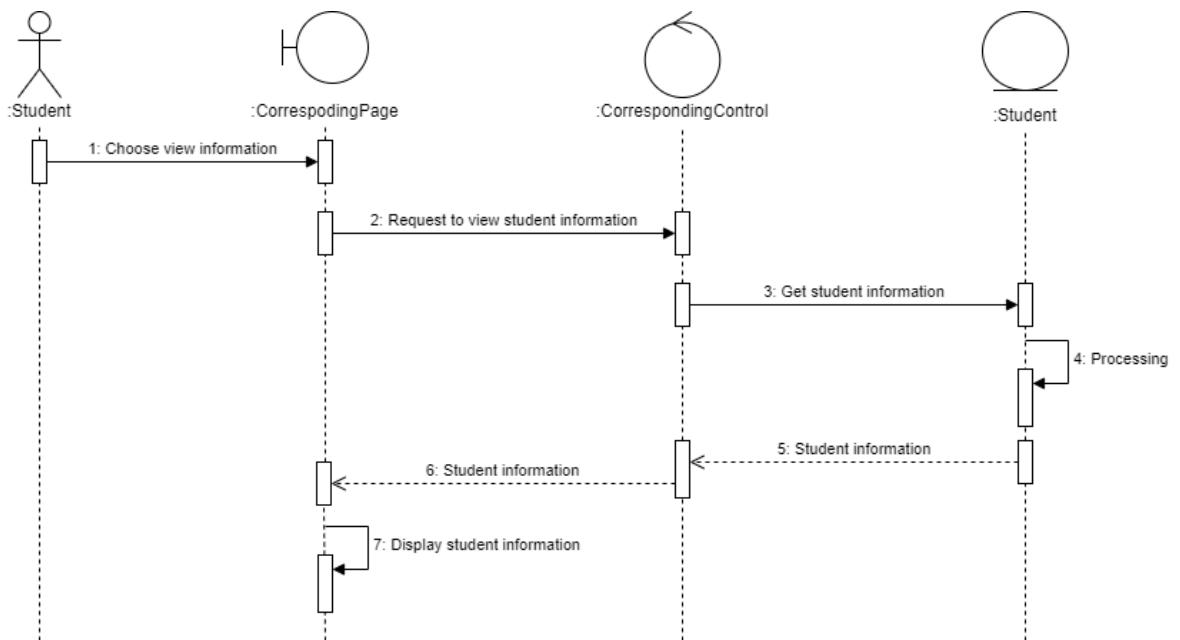


Figure 4.28 Sequence Diagram for View student information

Change account password

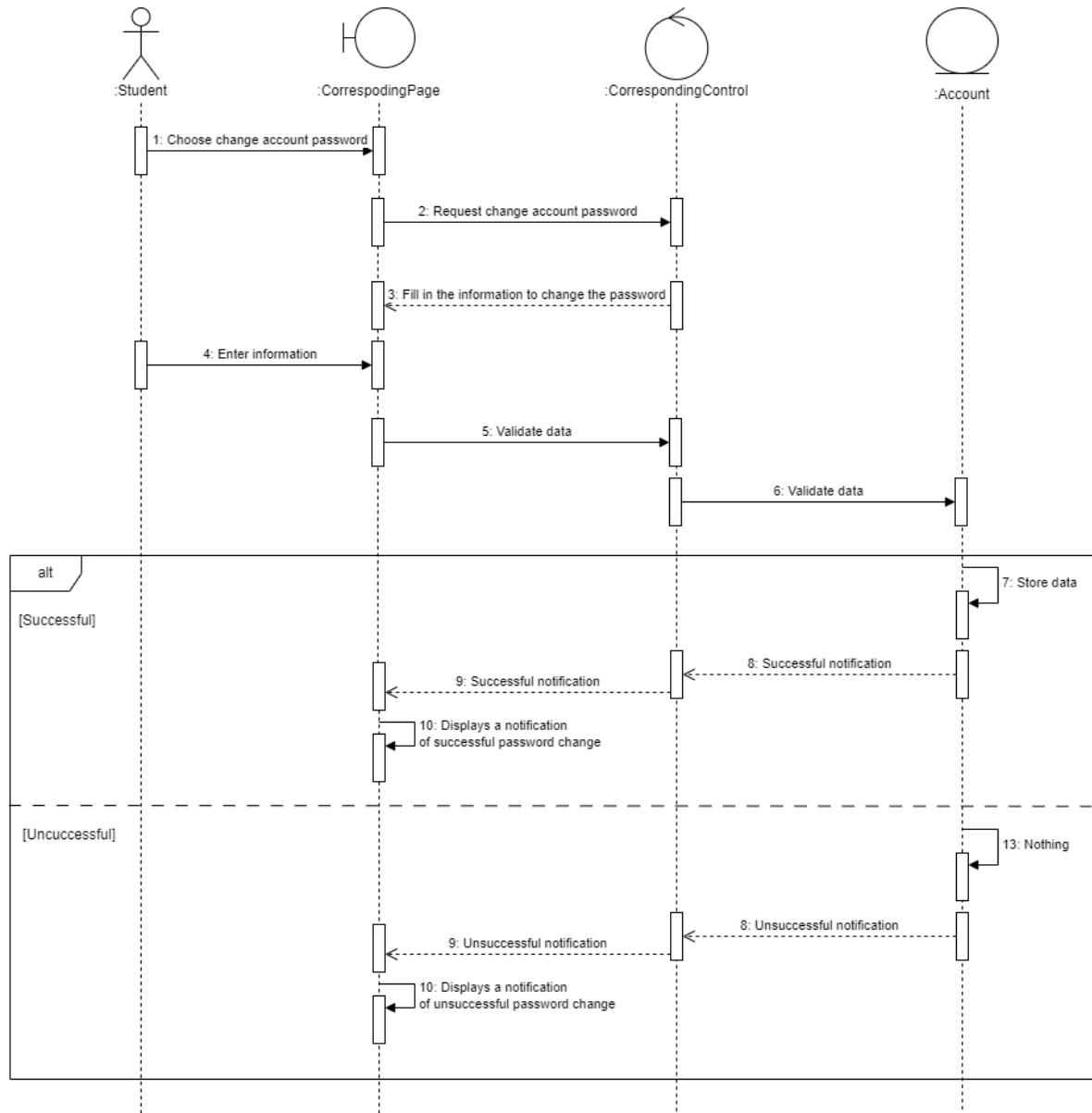


Figure 4.29 Sequence Diagram for Change account password

Change bank account number

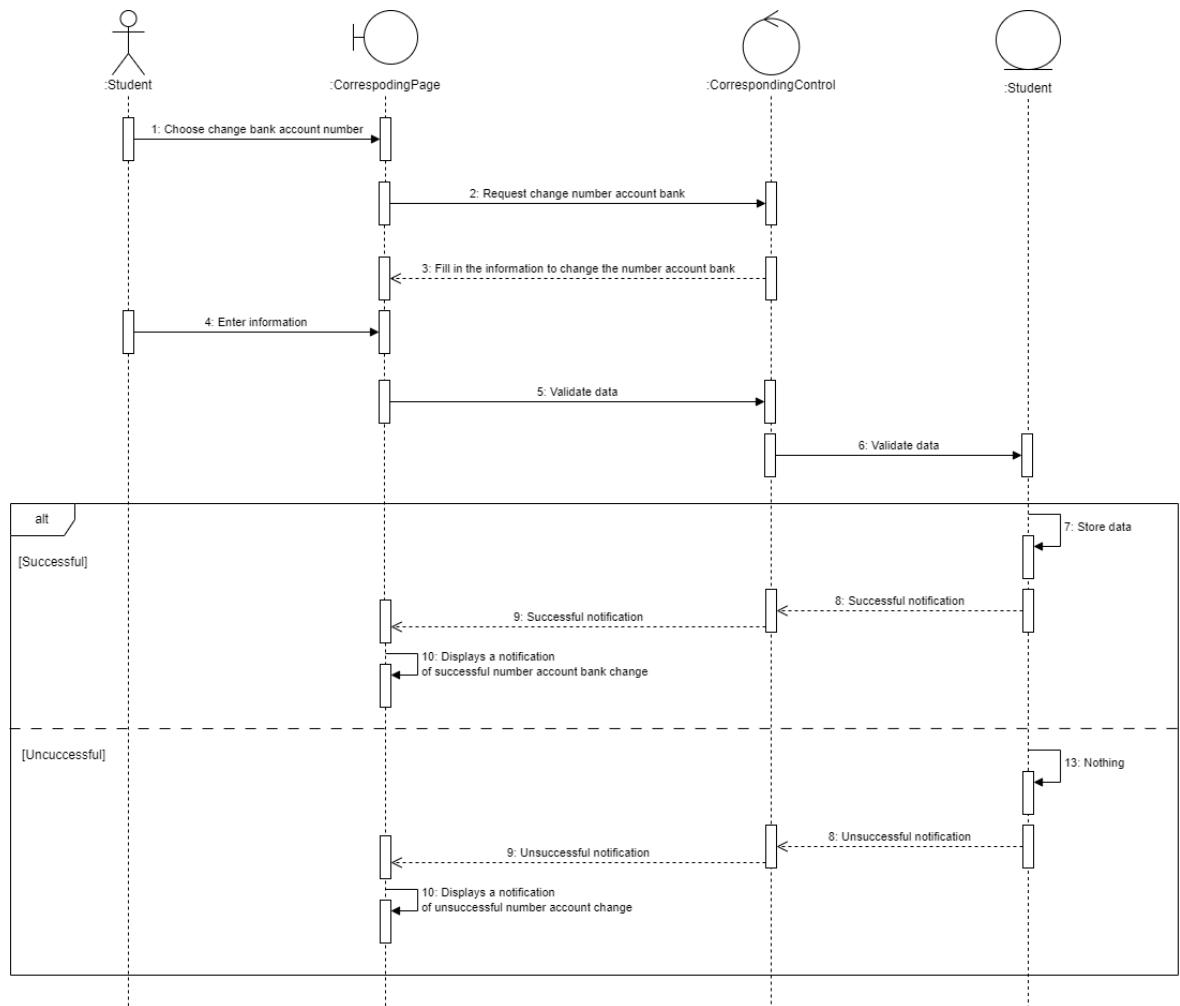


Figure 4.30 Sequence Diagram for Change bank account number

View invoice

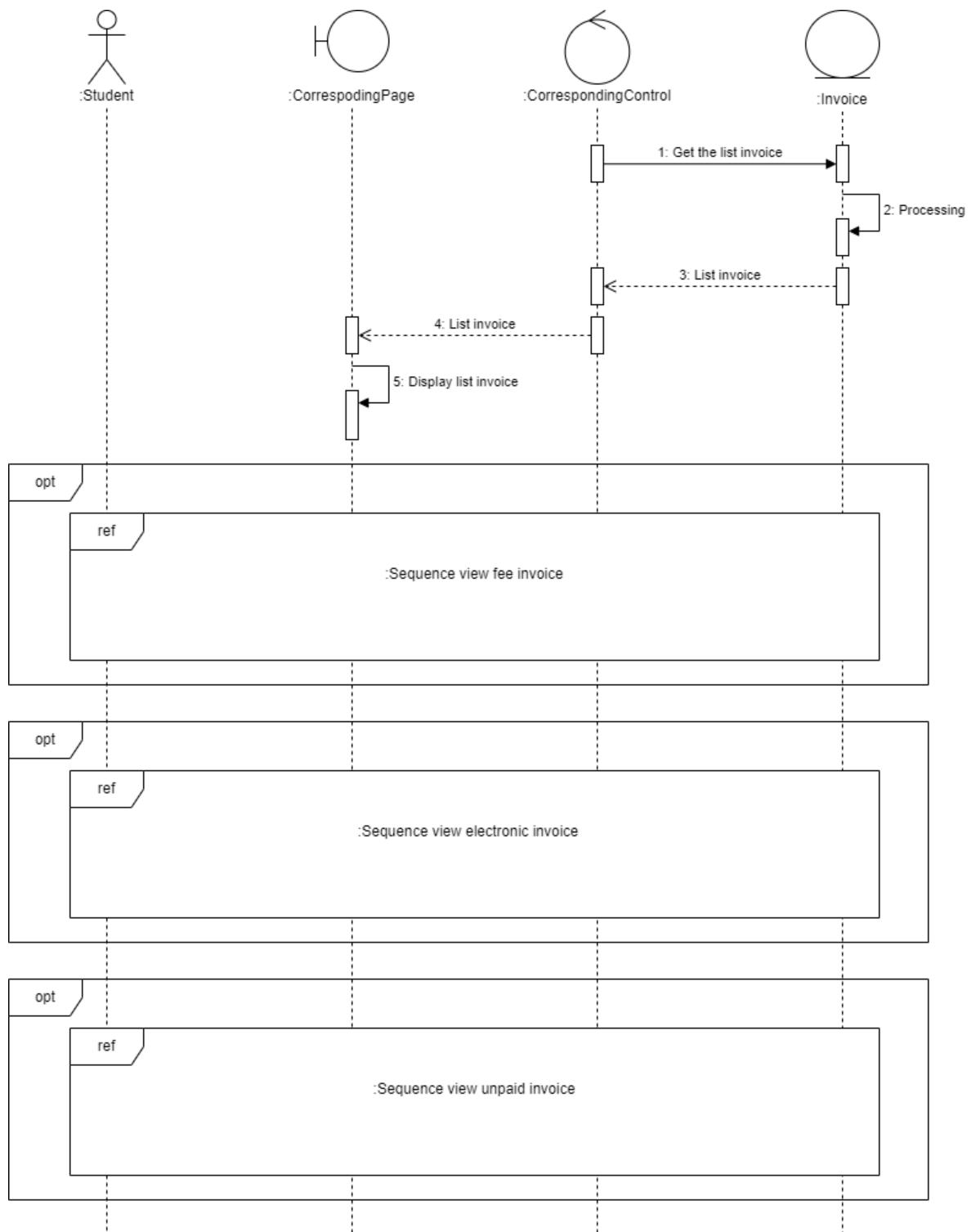


Figure 4.31 Sequence Diagram for View invoice

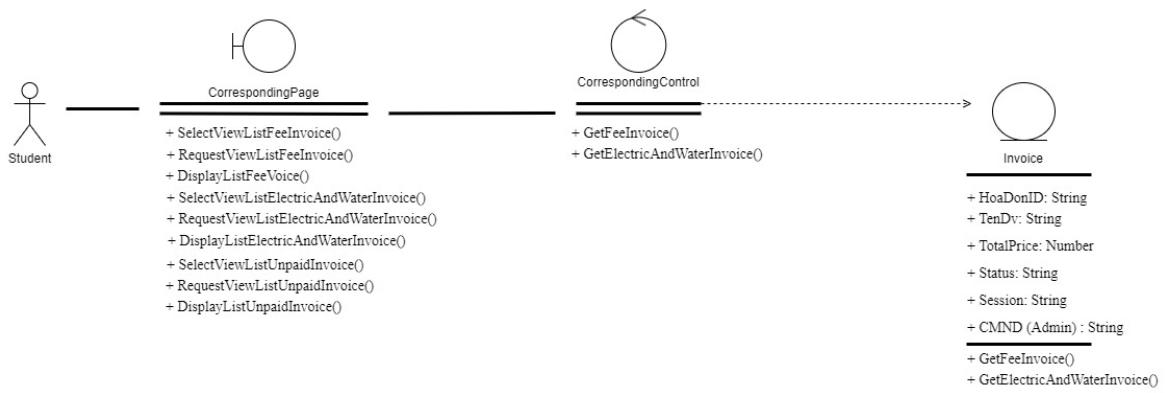


Figure 4.32 Class Diagram for View invoice

View fee invoice

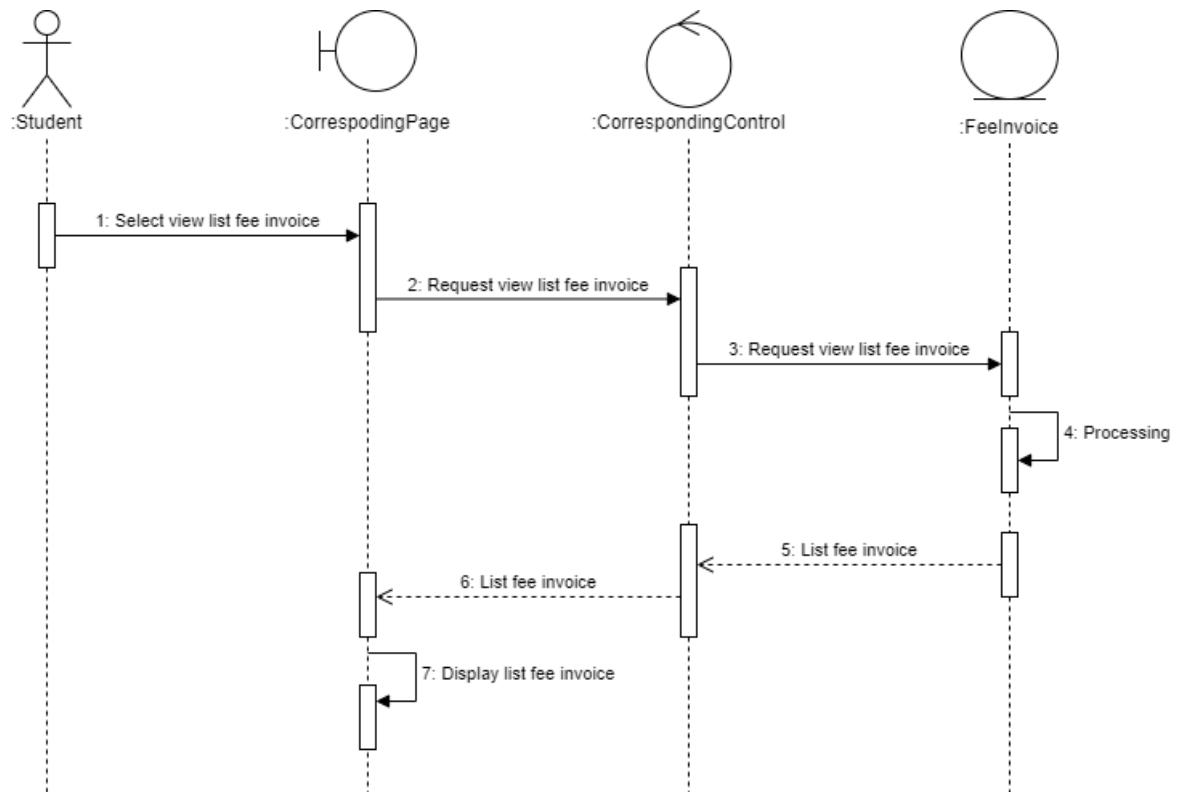


Figure 4.33 Sequence Diagram for View fee invoice

View electronic and water invoice

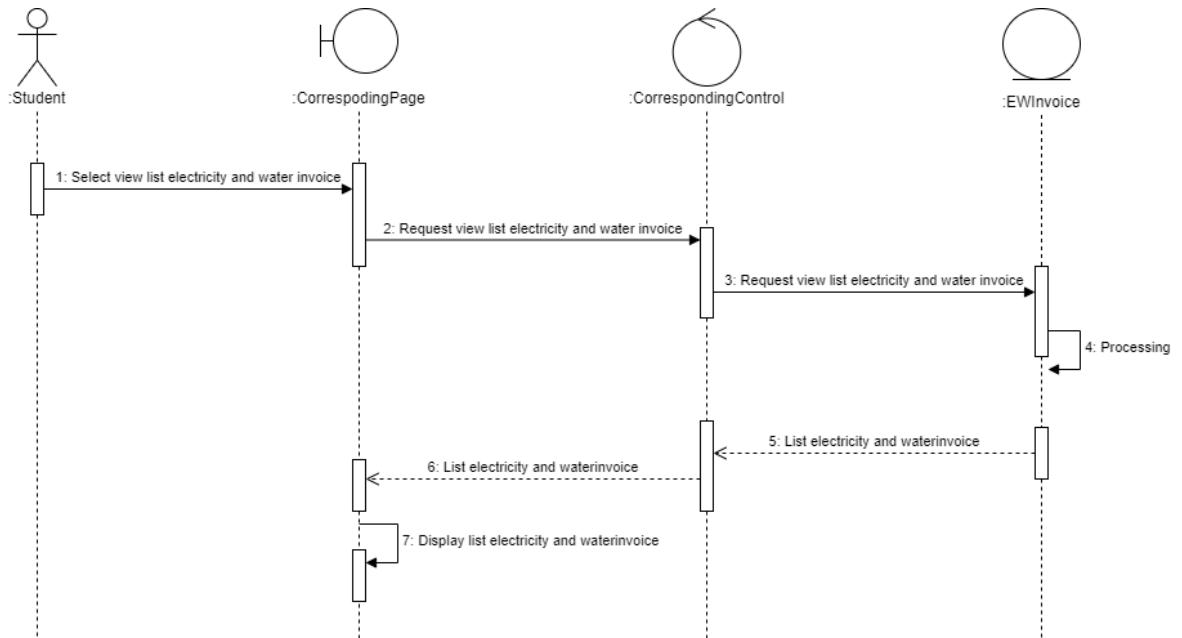


Figure 4.34 Sequence Diagram for View electronic and water invoice

View unpaid invoice

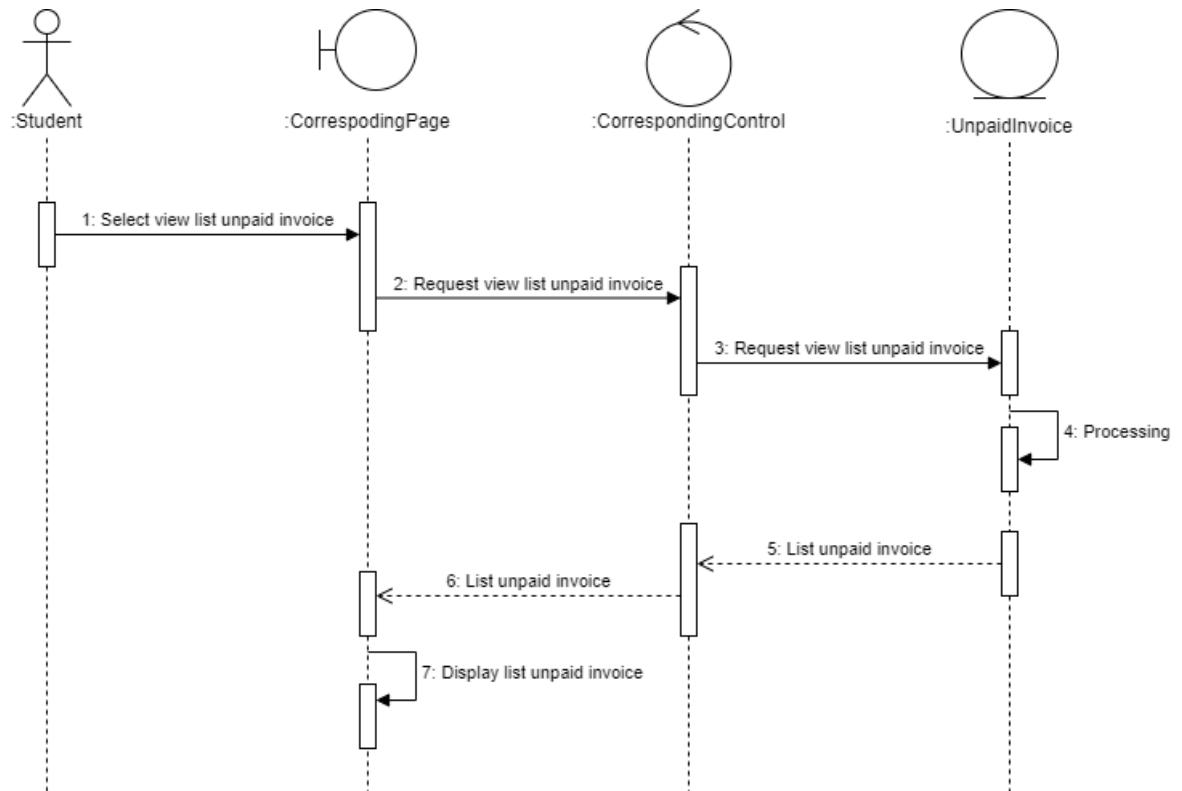


Figure 4.35 Sequence Diagram for View unpaid invoice

Service

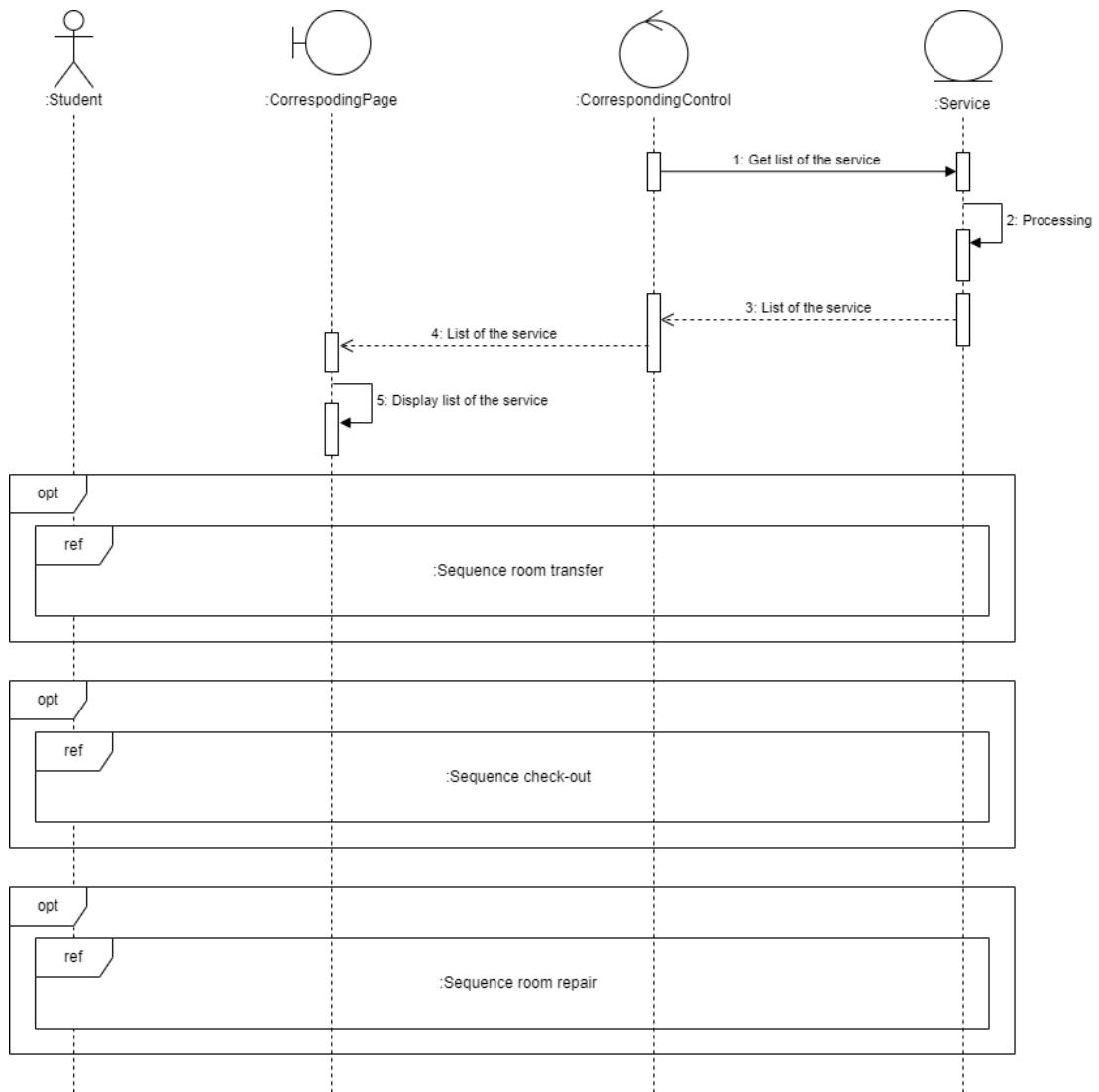


Figure 4.36 Sequence Diagram for Service

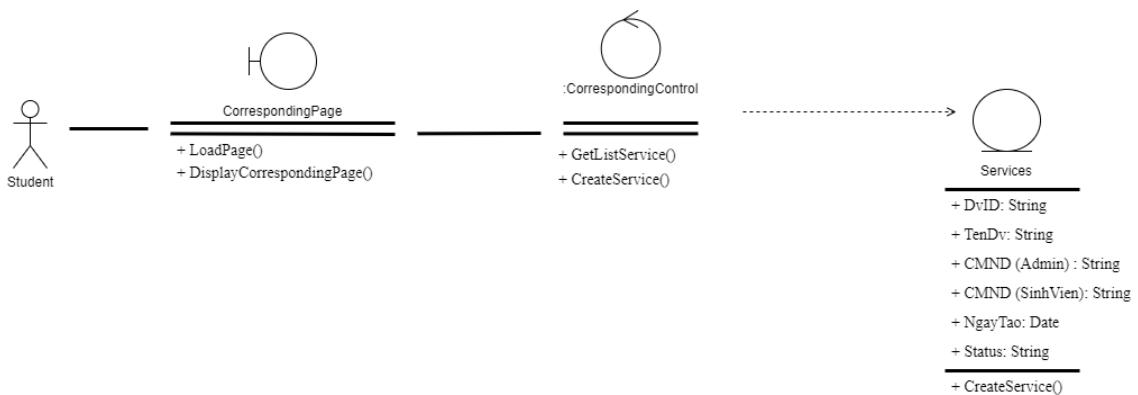


Figure 4.37 Class Diagram for Service

Room transfer

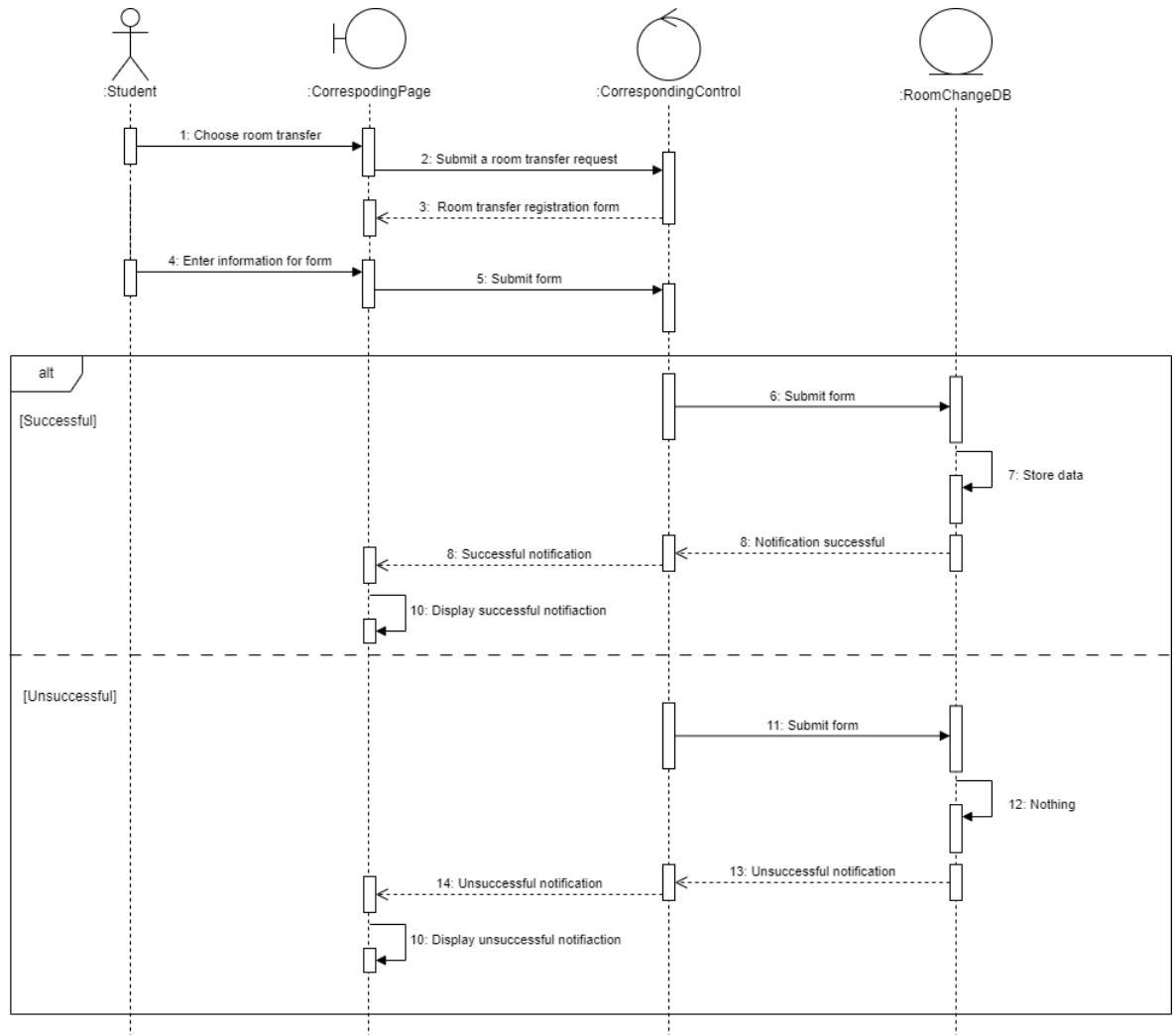


Figure 4.38 Sequence Diagram for Room transfer

Room check-out

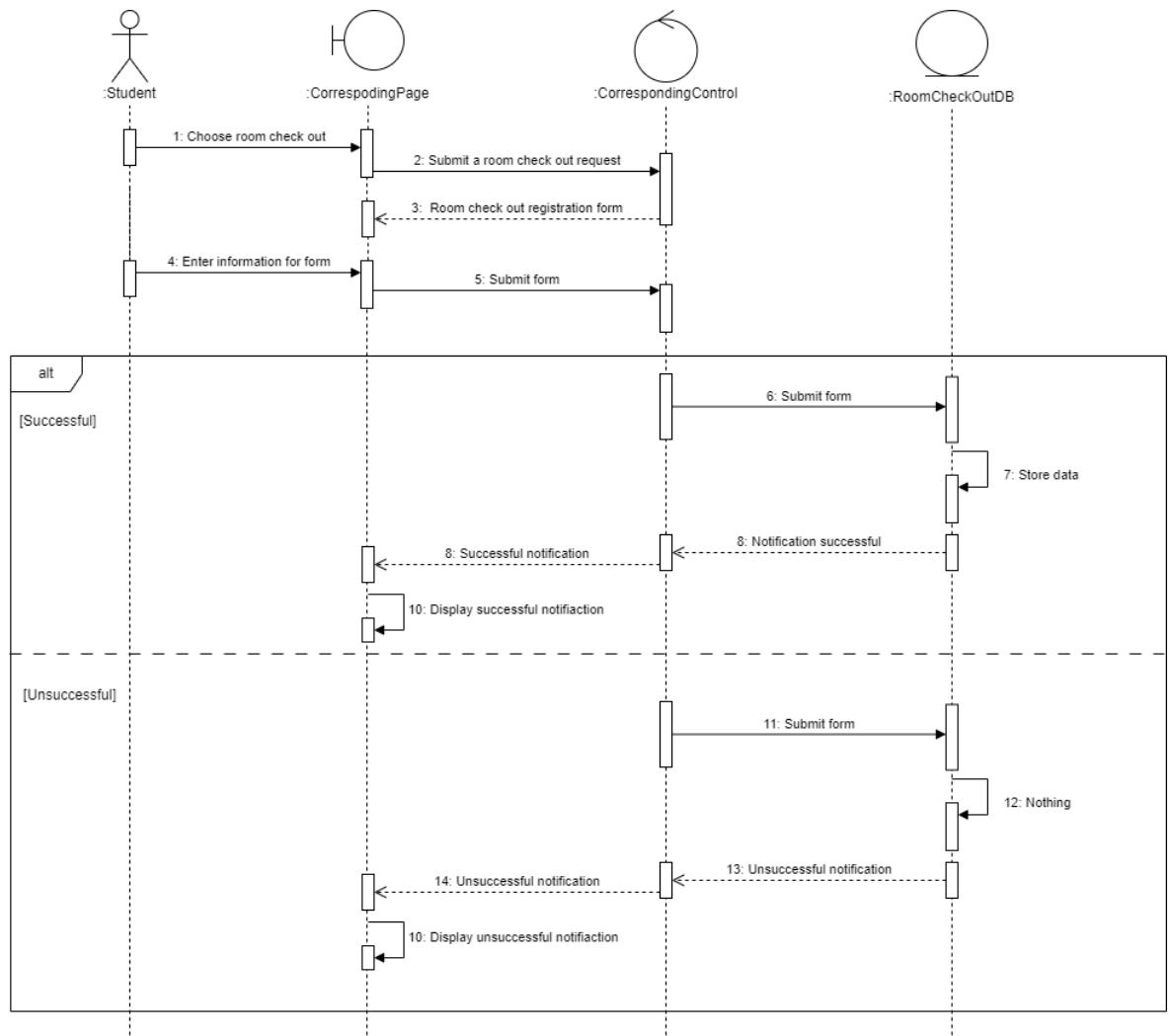


Figure 4.39 Sequence Diagram for Room check-out

Room repair

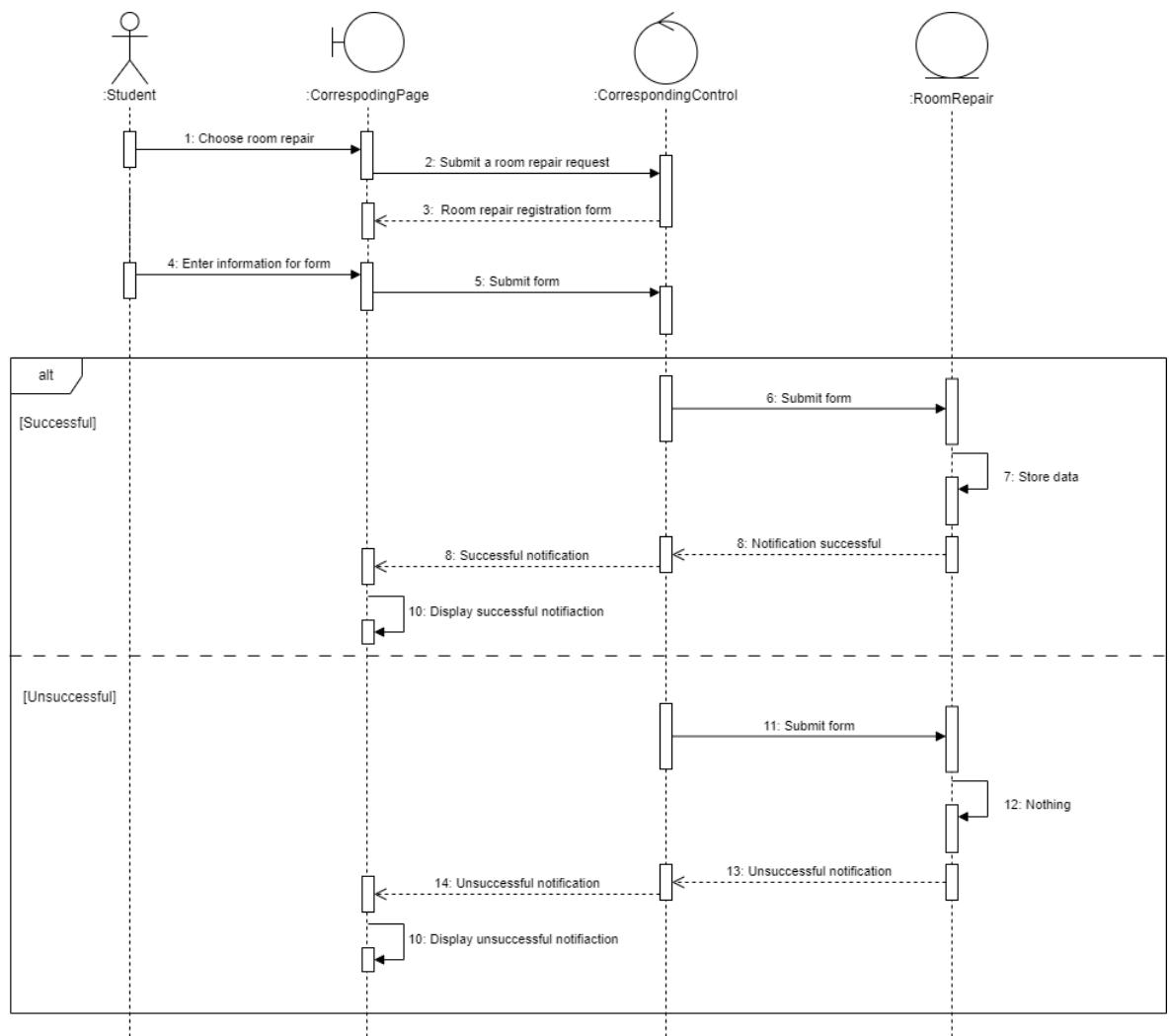


Figure 4.40 Sequence Diagram for Room repair

Manage student

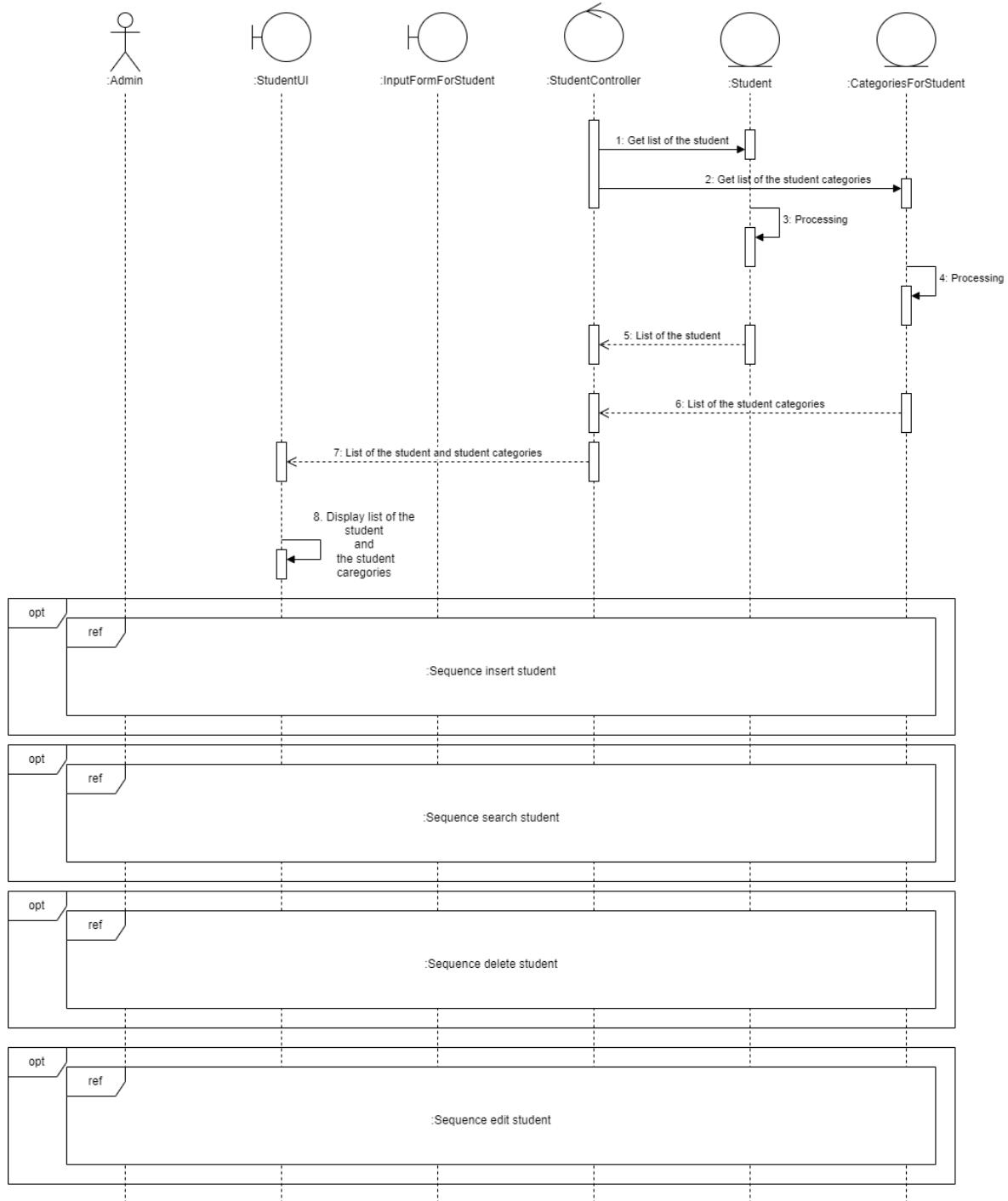


Figure 4.41 Sequence Diagram for Manage student

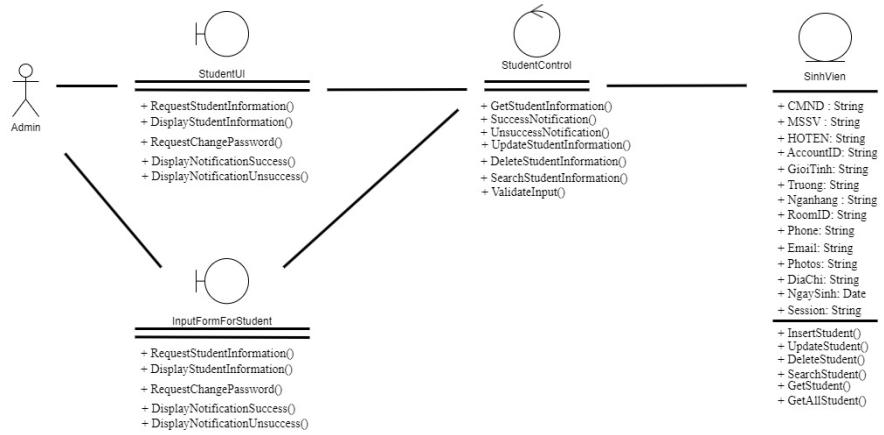


Figure 4.42 Class Diagram for Manage student

Insert student

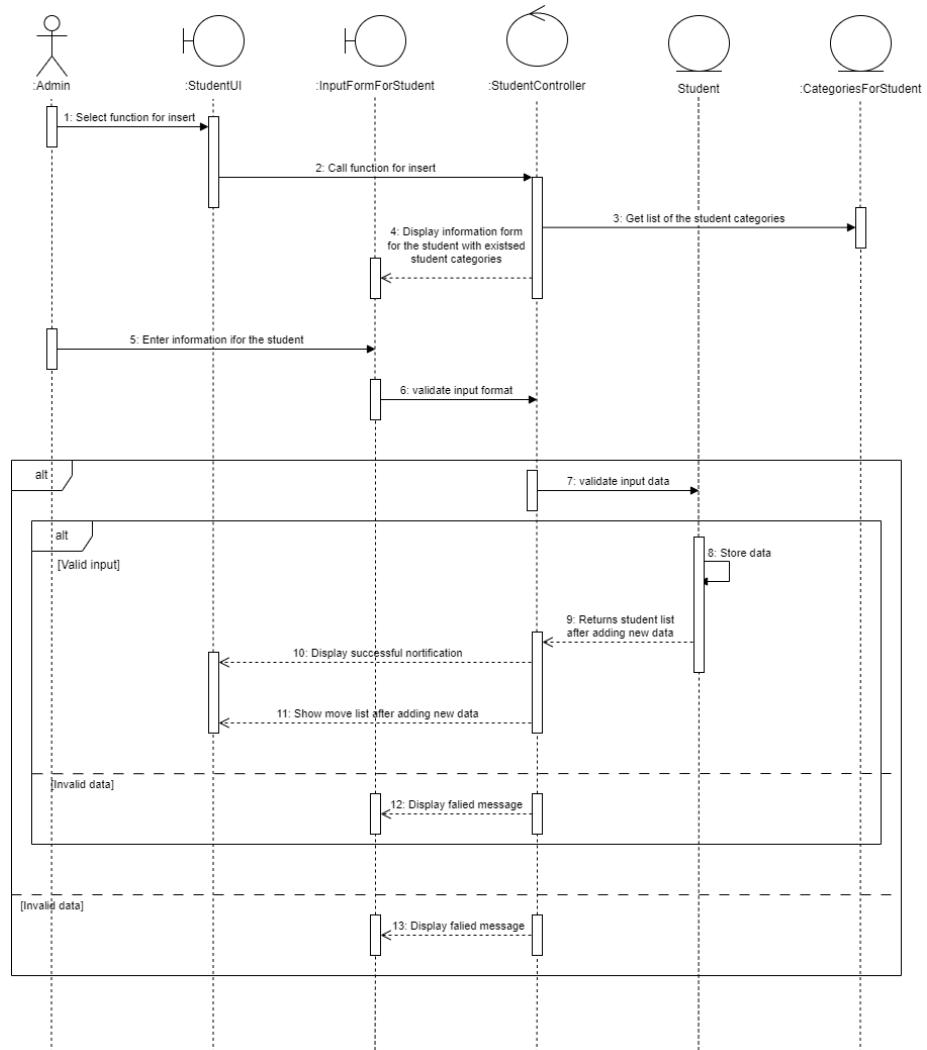


Figure 4.43 Sequence Diagram for Insert student

Search student

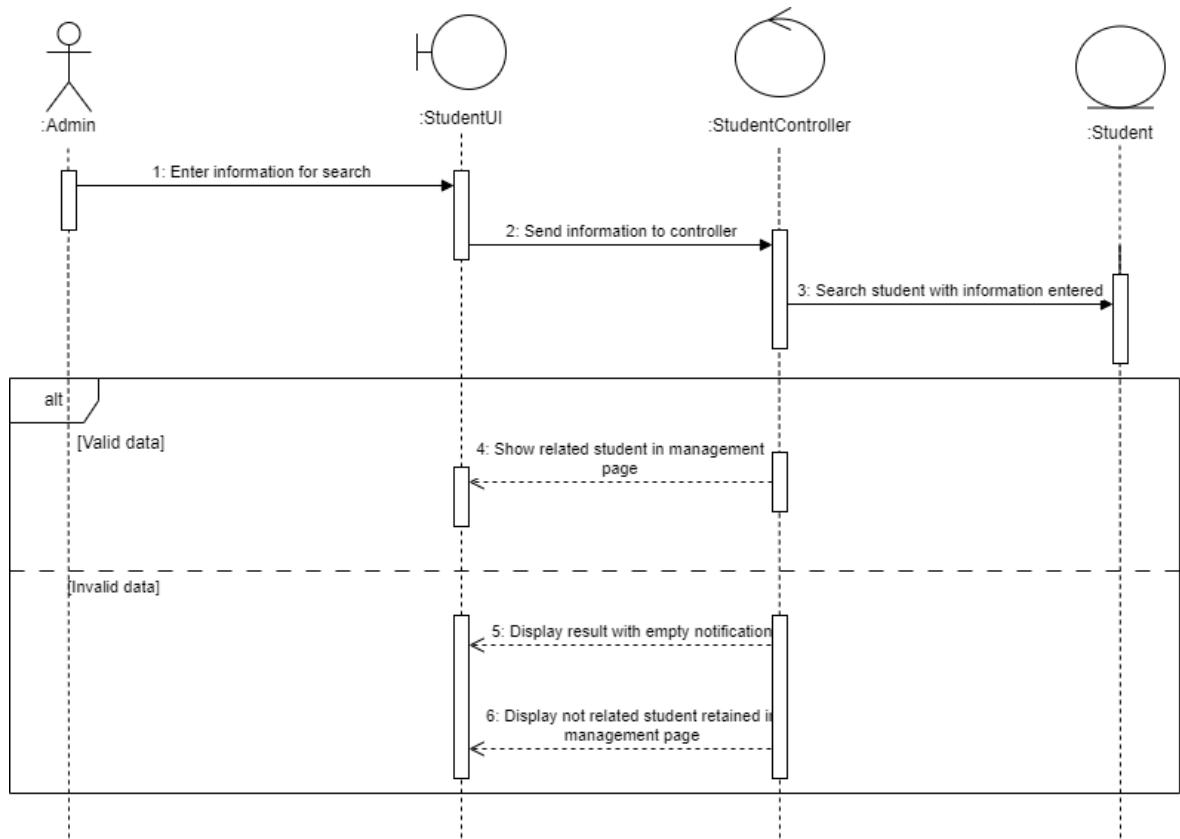


Figure 4.44 Sequence Diagram for Search student

Edit student

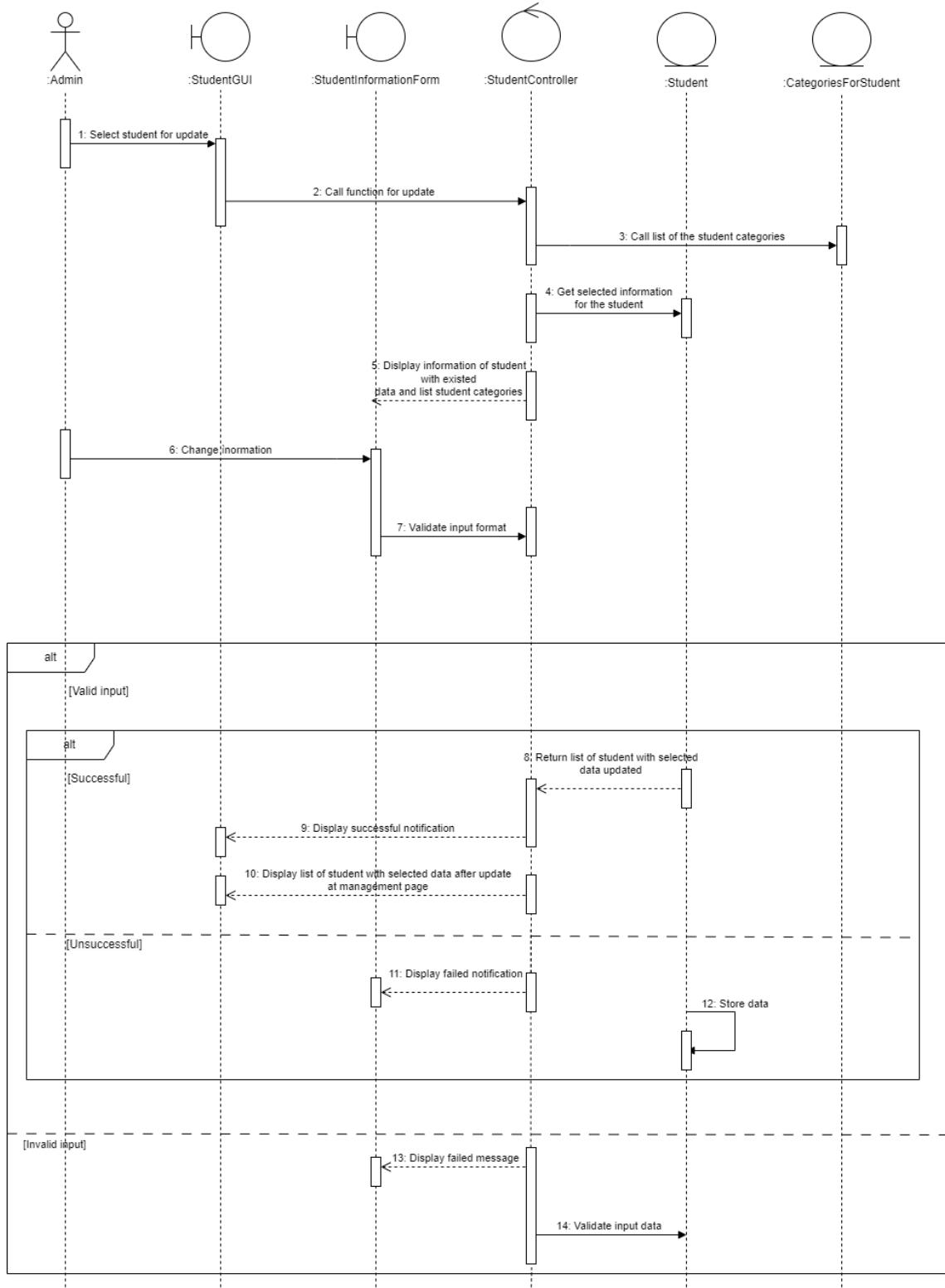


Figure 4.45 Sequence Diagram for Edit student

Delete student

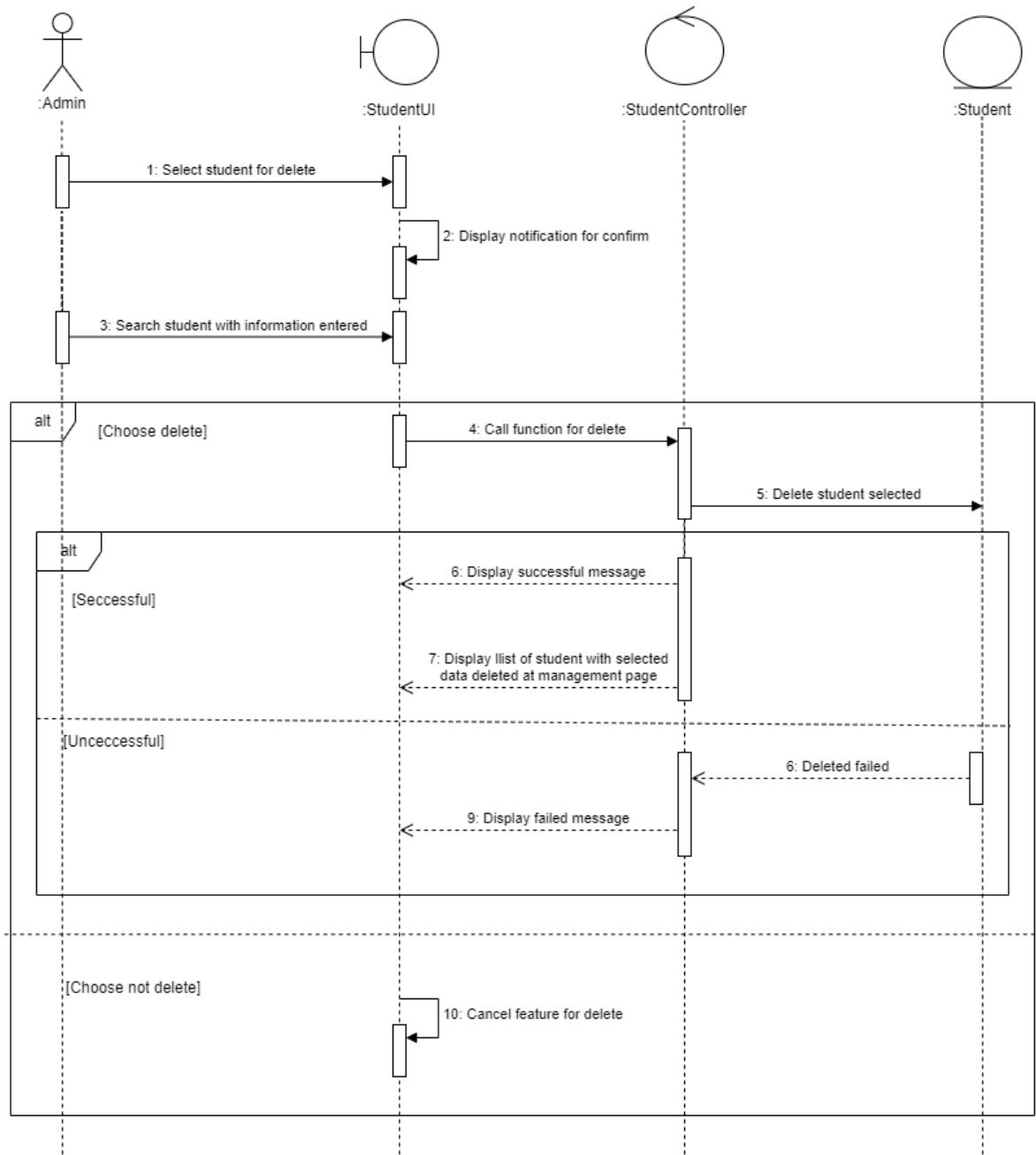


Figure 4.46 Sequence Diagram for Delete student

Manage room

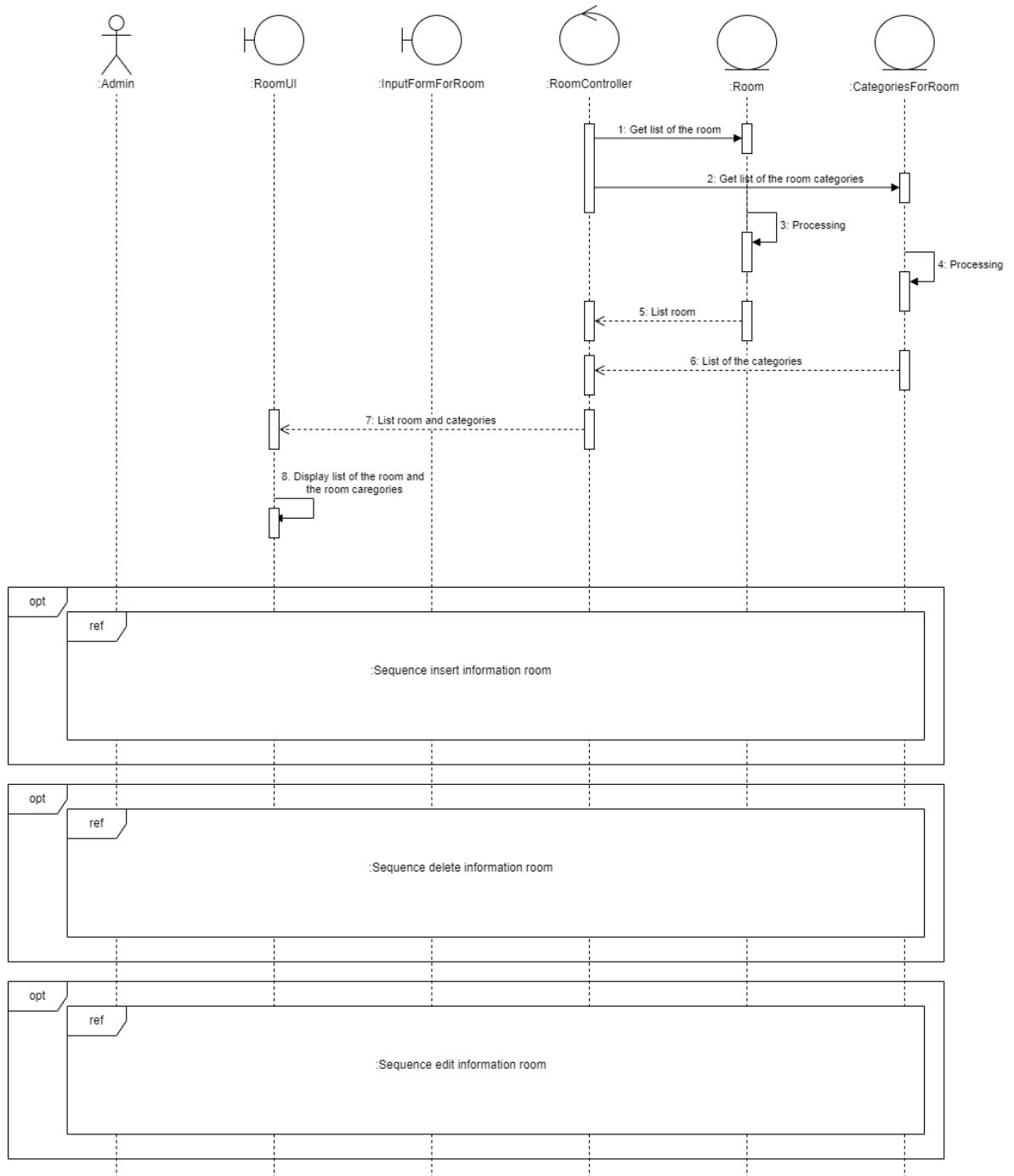


Figure 4.47 Sequence Diagram for Manage room

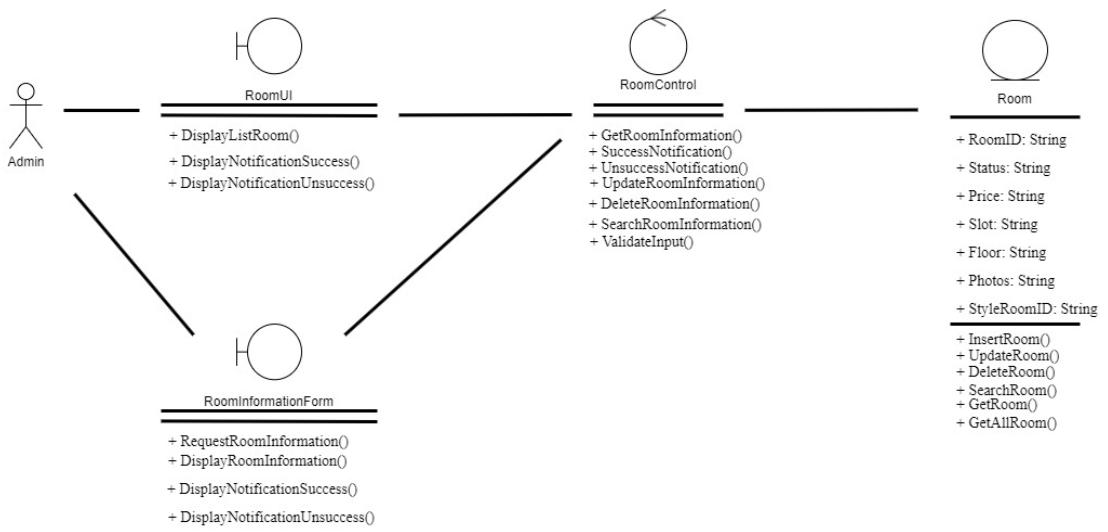


Figure 4.48 Class Diagram for Manage room

Search room

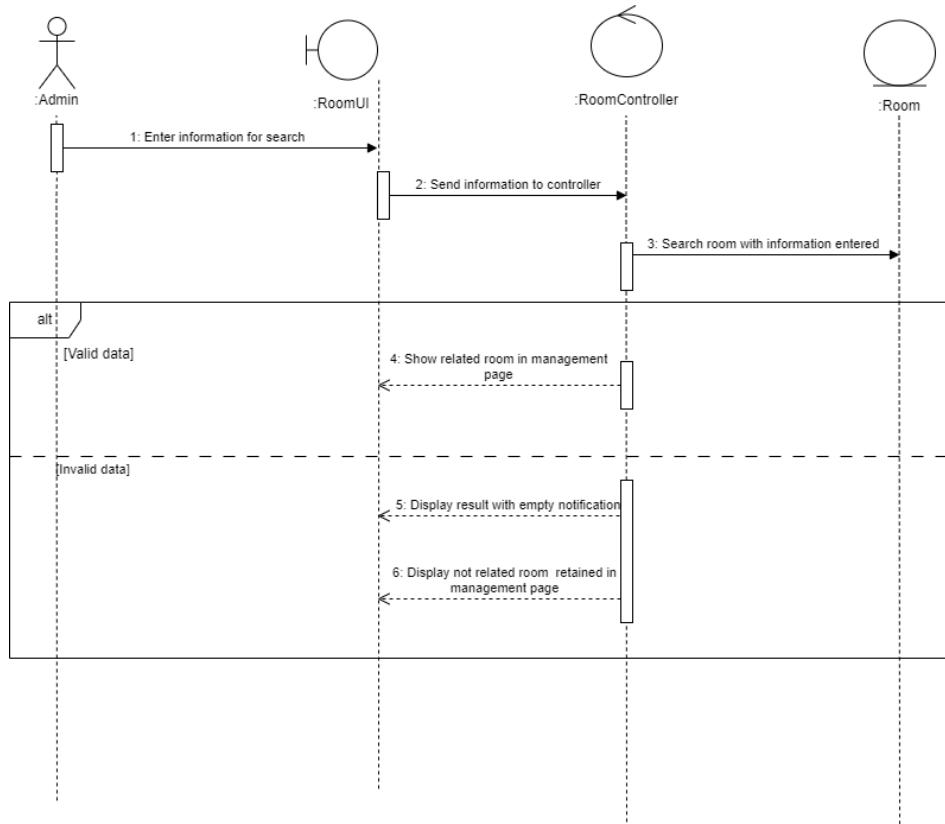


Figure 4.49 Sequence Diagram for Search room

Edit room

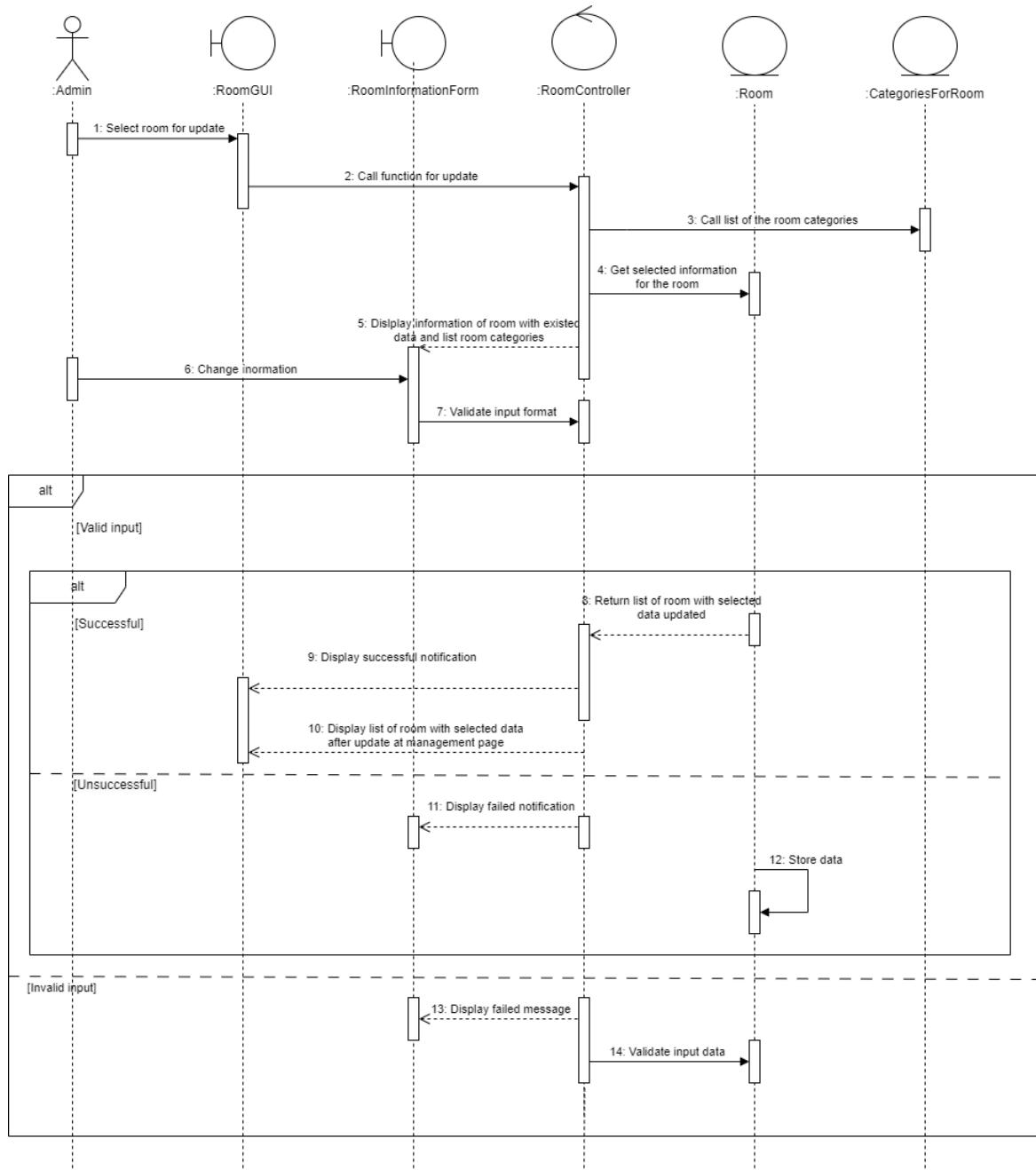


Figure 4.50 Sequence Diagram for Edit room

Delete room

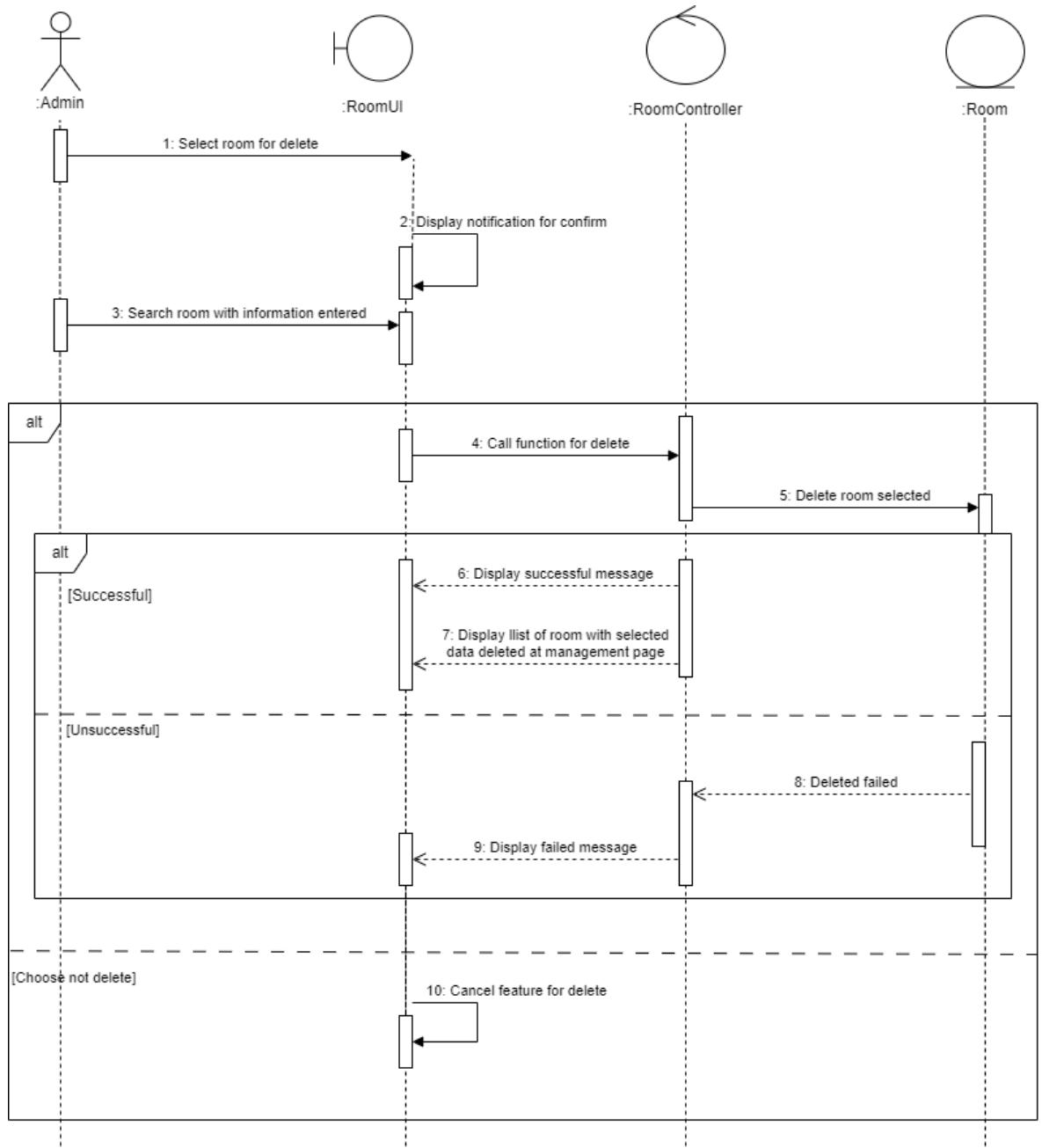


Figure 4.51 Sequence Diagram for Delete room

Manage invoice

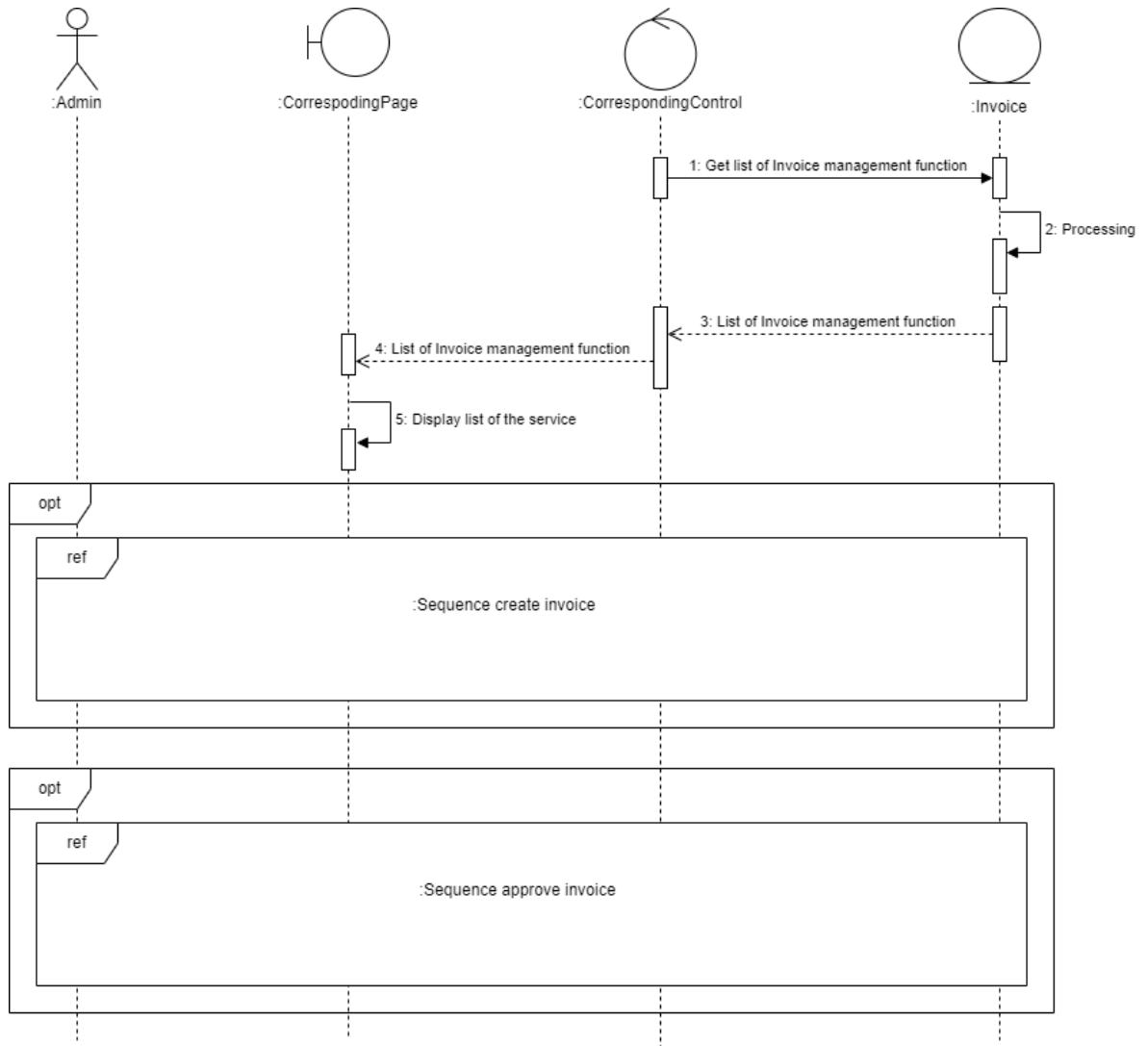


Figure 4.52 Sequence Diagram for Manage invoice

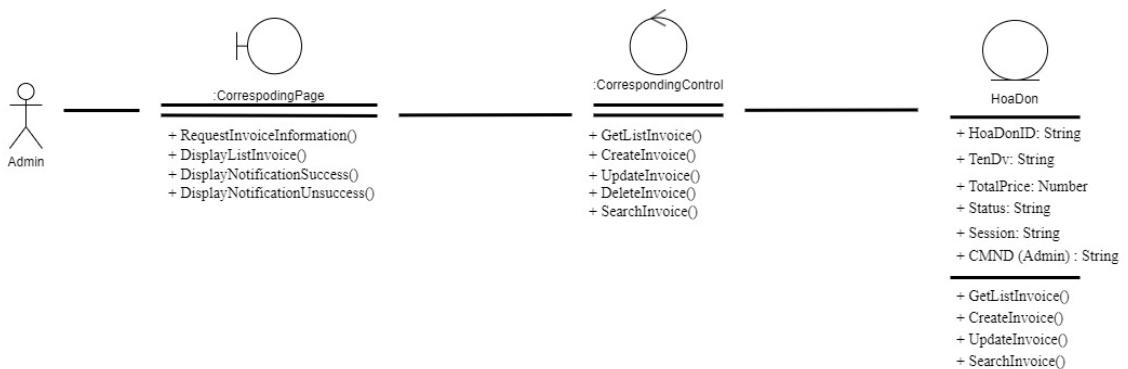


Figure 4.53 Class Diagram for Manage invoice

Create invoice

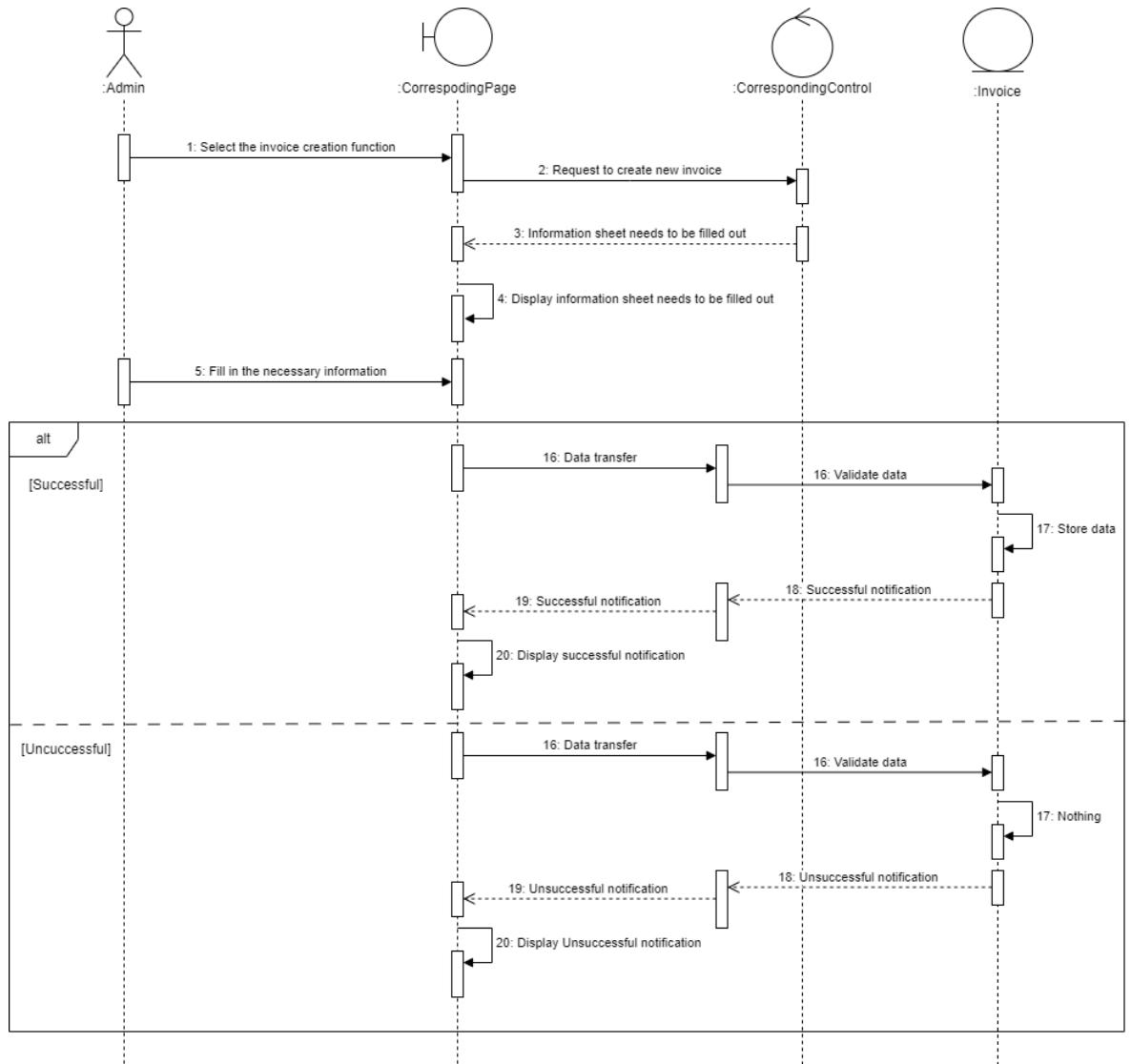


Figure 4.54 Sequence Diagram for Create invoice

Approve invoice

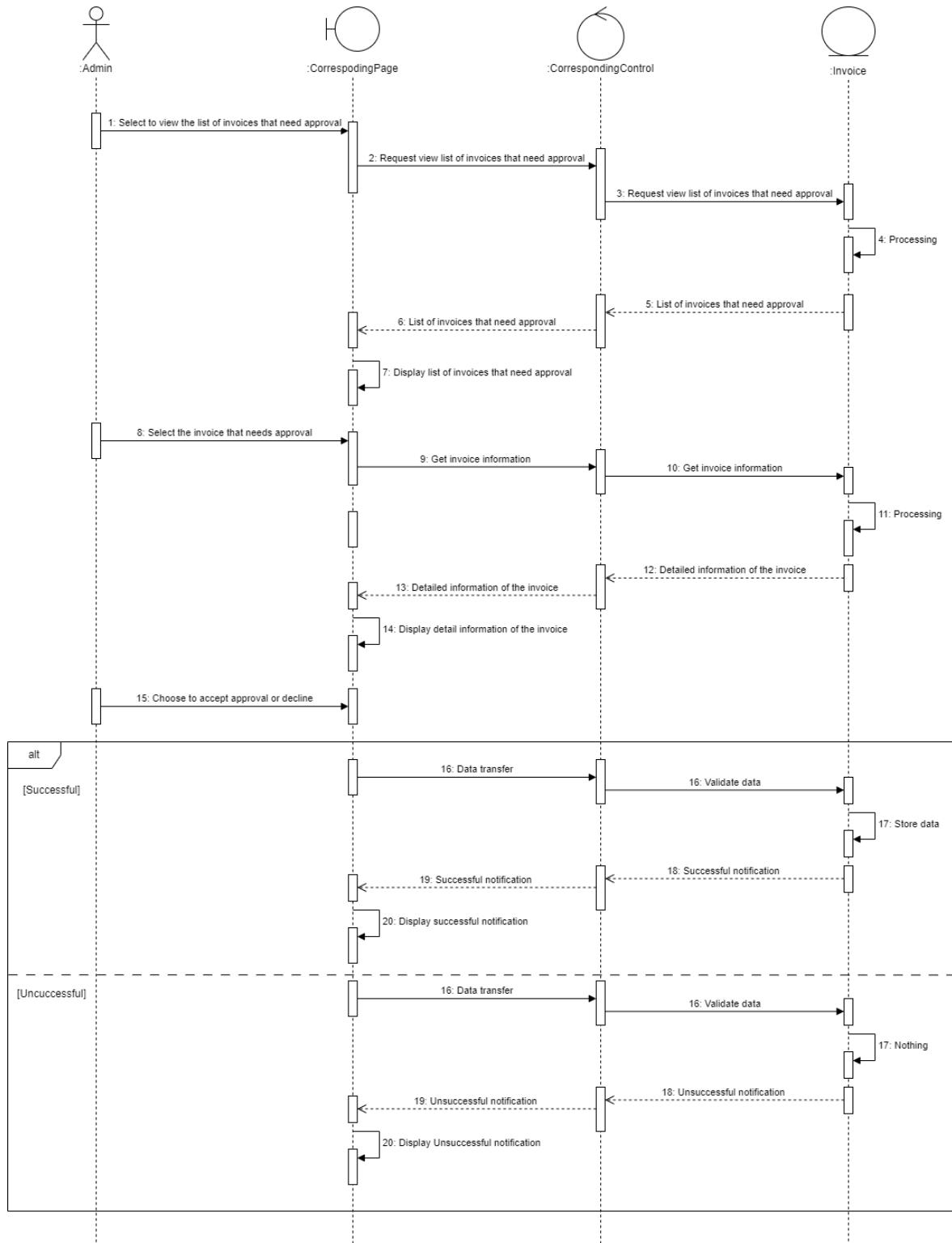


Figure 4.55 Sequence Diagram for Approve invoice

Manage service

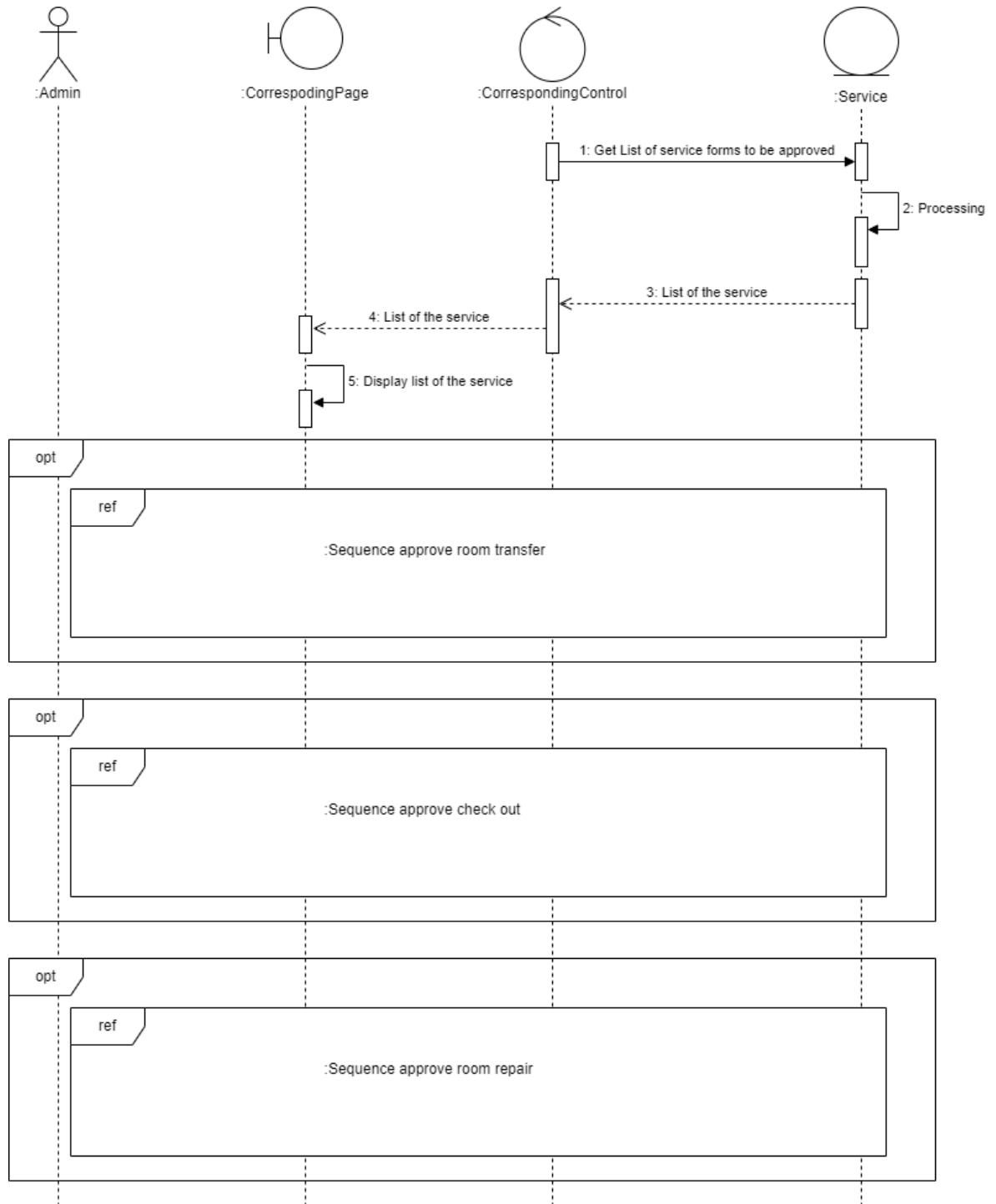


Figure 4.56 Sequence Diagram for Manage service

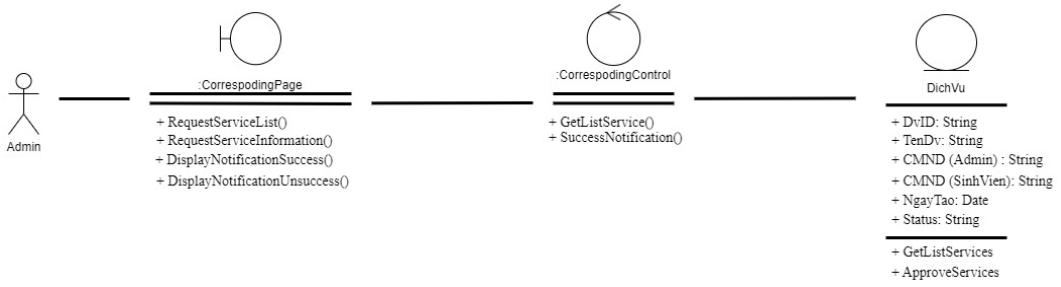


Figure 4.57 Class Diagram for Manage service

Approve room transfer

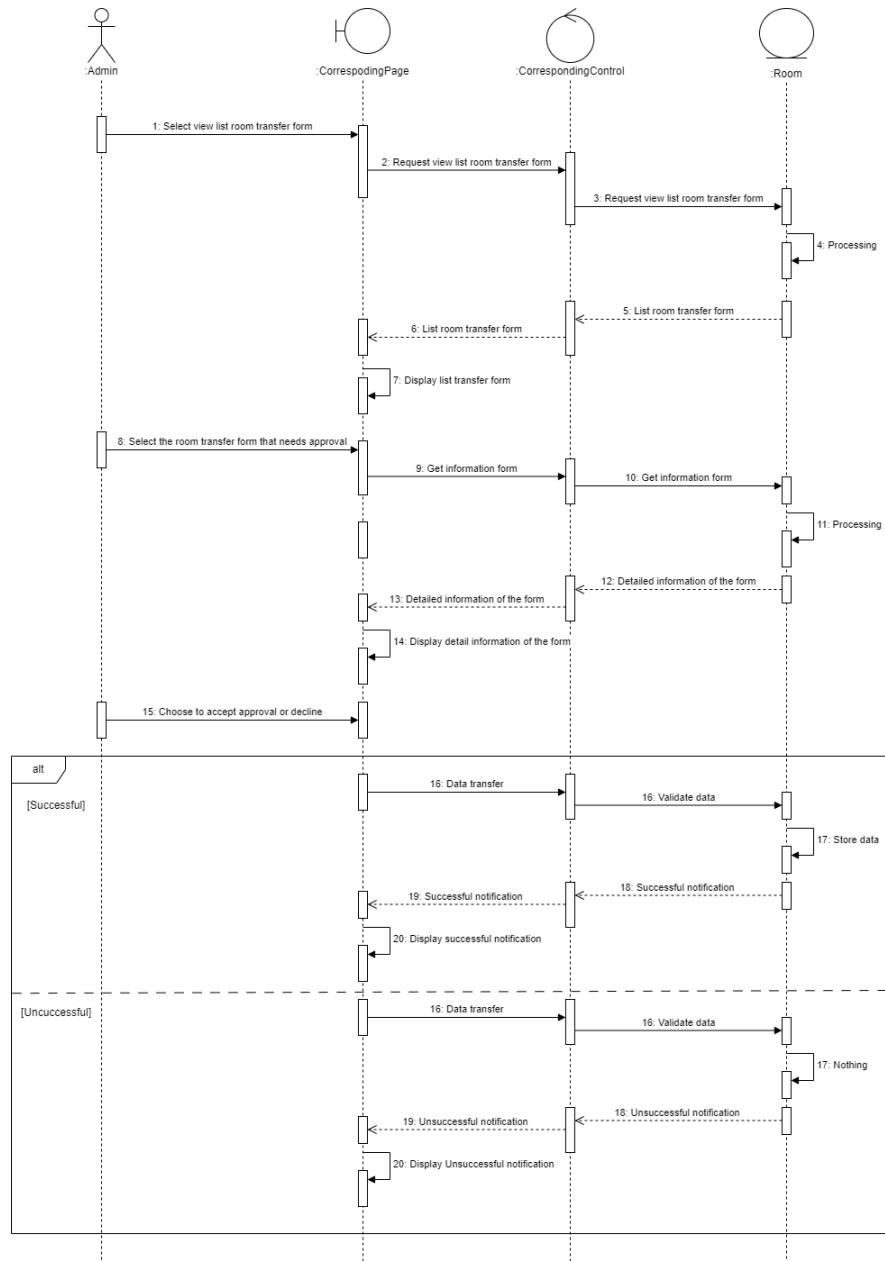


Figure 4.58 Sequence Diagram for Approve room transfer

Approve room check-out

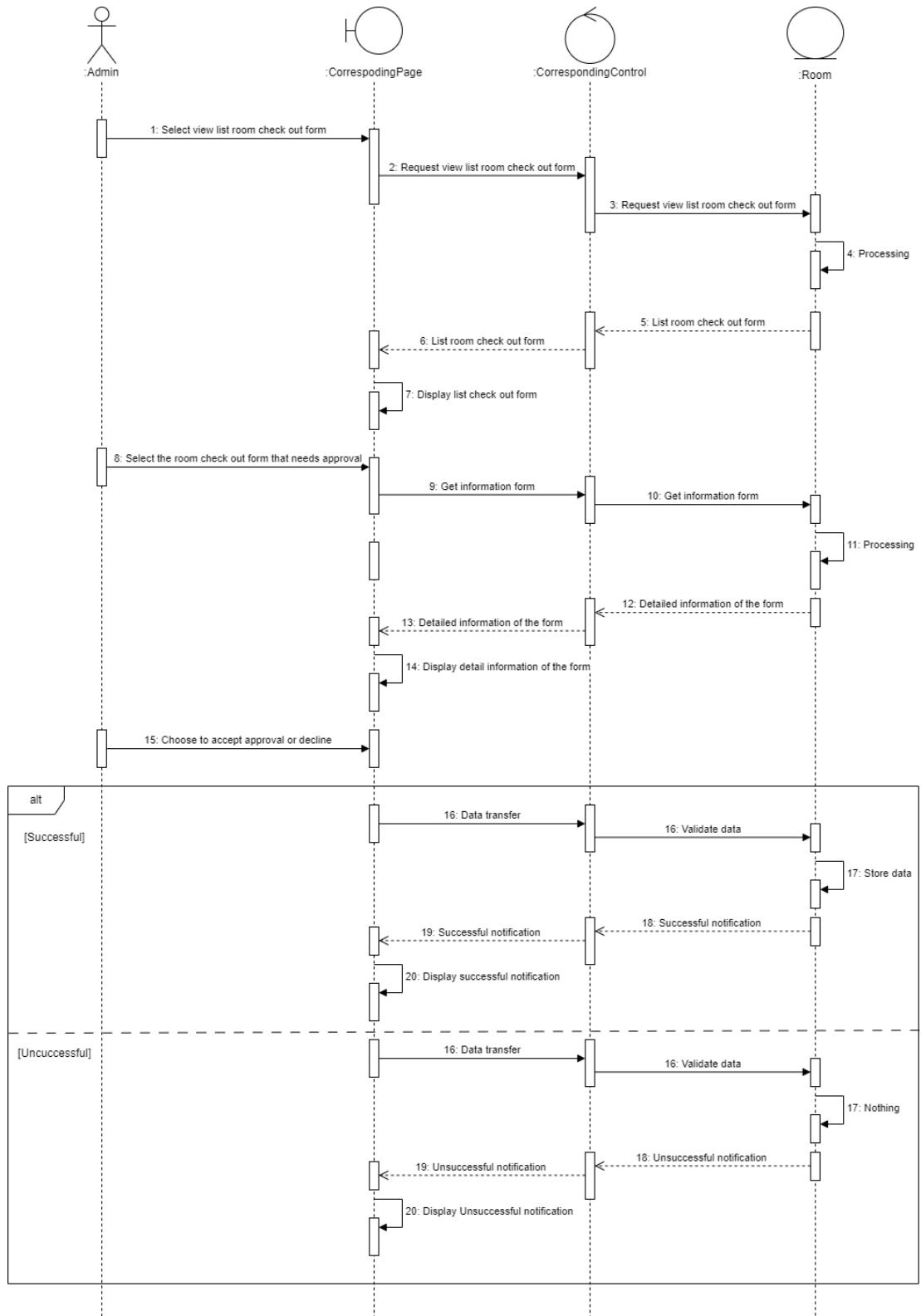


Figure 4.59 Sequence Diagram for Approve room check-out

Approve room repair

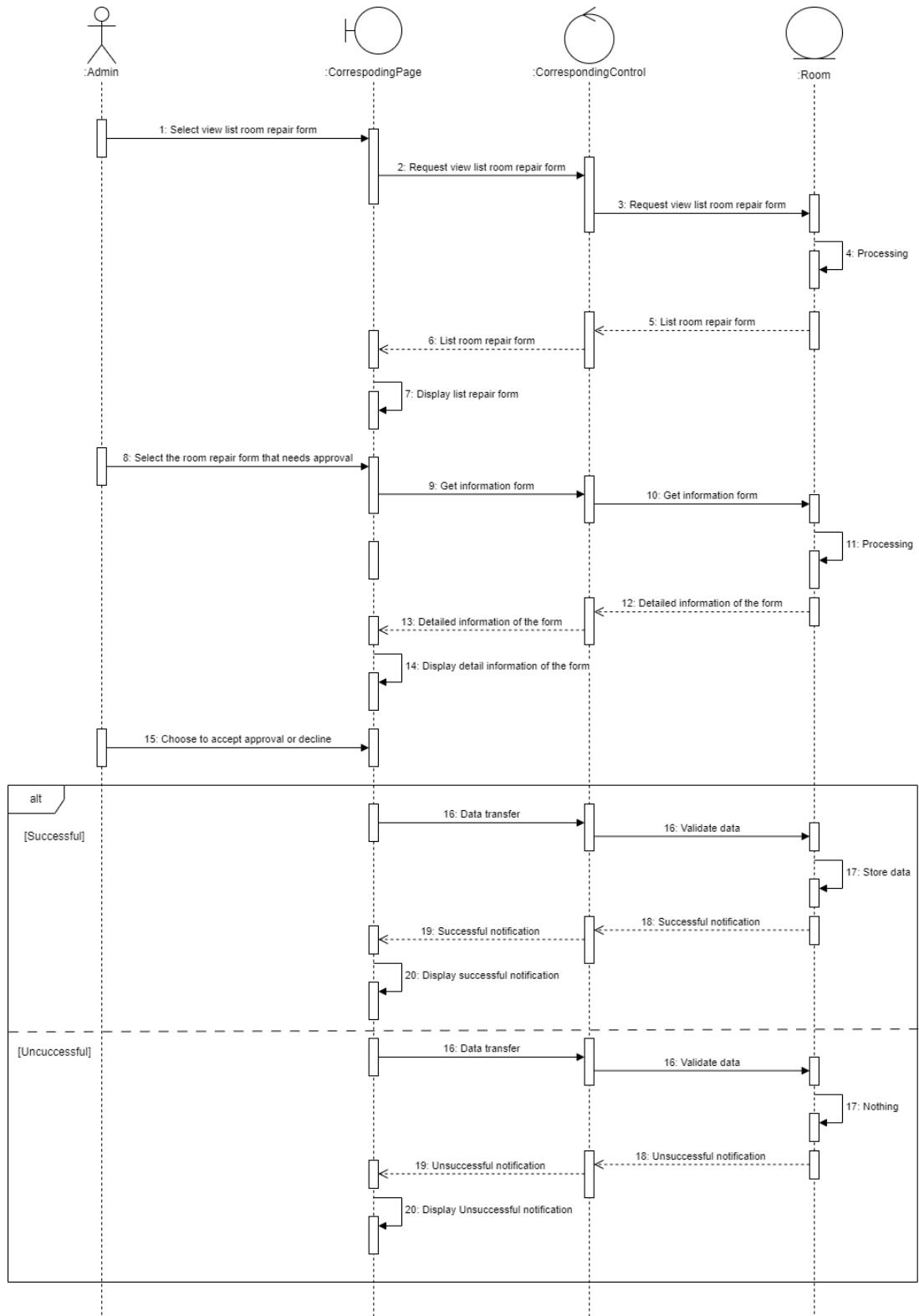


Figure 4.60 Sequence Diagram for Approve room repair

Log out

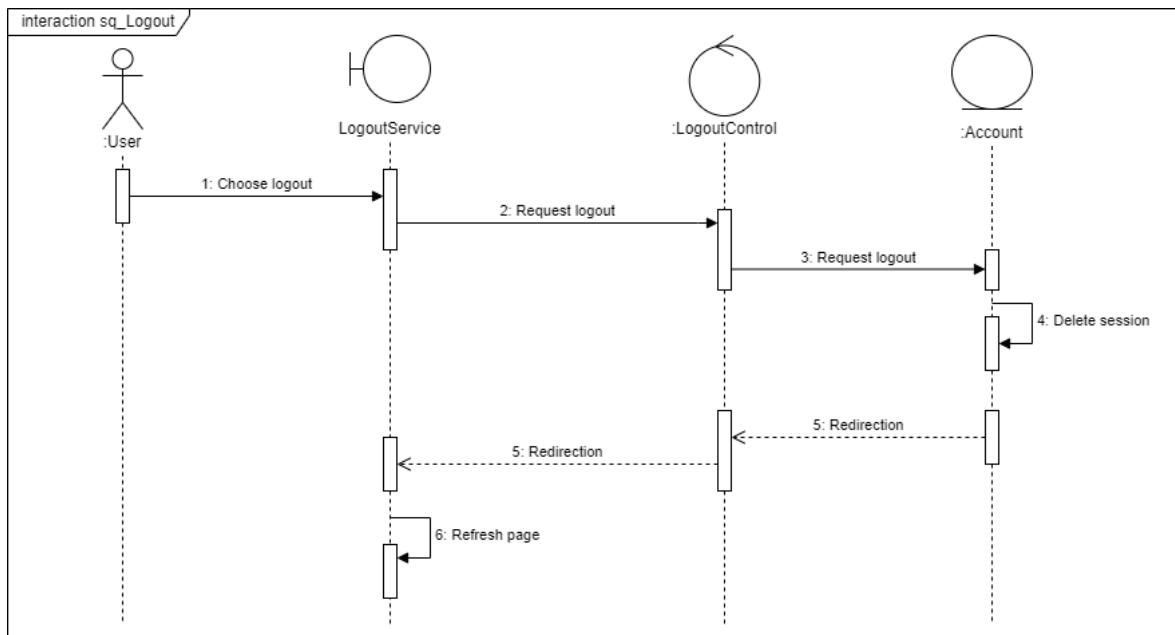


Figure 4.61 Sequence Diagram for Log out

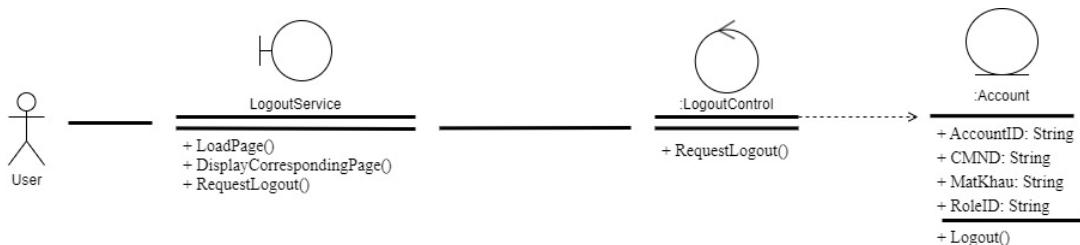


Figure 4.62 Class Diagram for Log out

4.6 Entity Relationship Diagram (ERD)

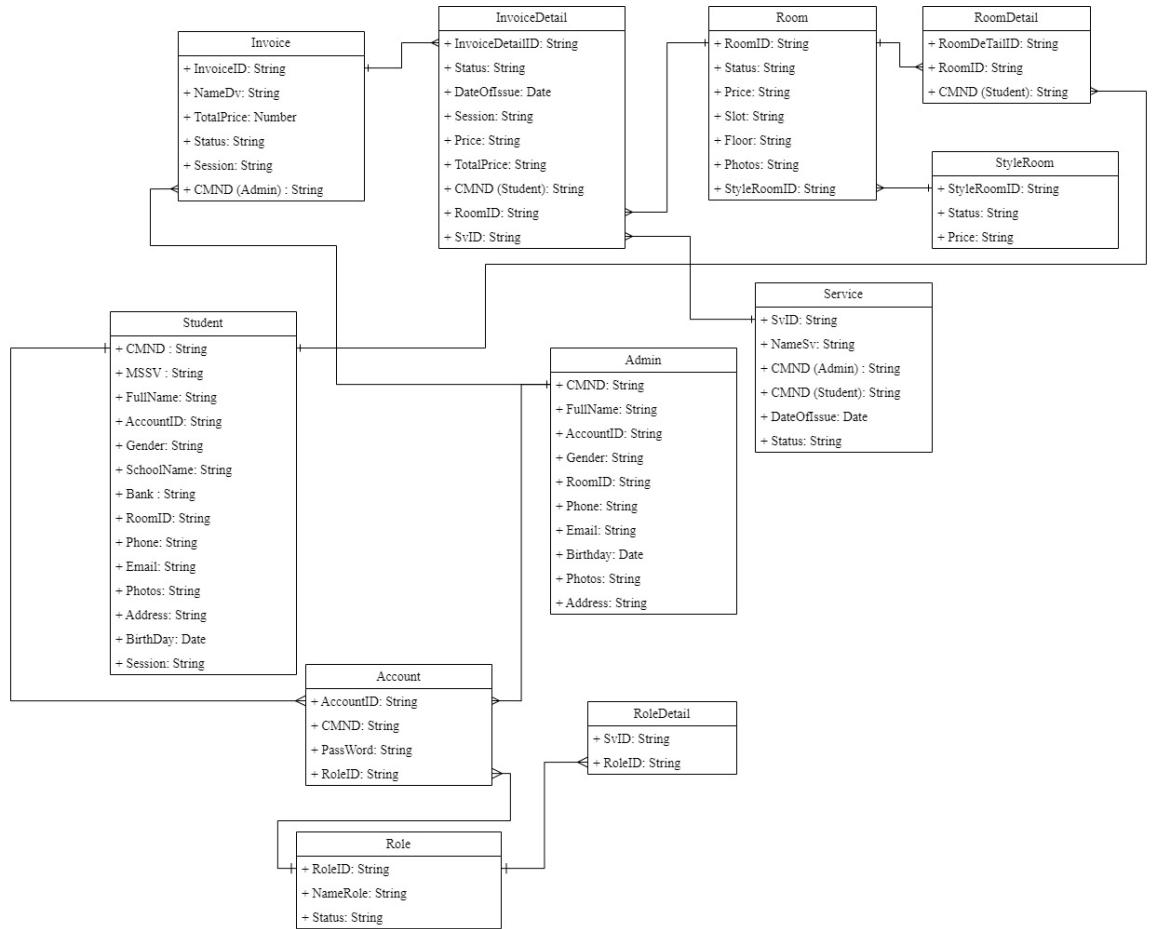


Figure 4.63 Entity Relationship Diagram

4.7 Description Database

Table 4.3 Table Invoice

Item	Type	Description
InvoiceID	String	The ID of invoice
NameSv	String	The name of service
TotalPrice	Number	Total costs to be paid
Status	String	Invoice status
Session	String	The school year is displayed on the invoice
CMND(Admin)	String	Admin's citizen identification number

Table 4.4 Table InvoiceDetail

Item	Type	Description
InvoiceDetailID	String	The ID of invoice detail
Status	String	Invoice status
DateOfIssue	Date	Invoice printing date
Session	String	The school year is displayed on the invoice
Price	String	Costs appear on invoice
TotalPrice	String	Total all costs
CMND(Student)	String	Student's citizen identification number
RoomID	String	The ID of room
SvID	String	The ID of service

Table 4.5 Table Room

Item	Type	Description
RoomID	String	The ID of room
Status	String	Room status
Price	String	Cost of room
Slot	String	Maximum number of students that can be in the room
Floor	String	The serial number of the floor in the building
Photos	String	Pictures of rooms
StyleRoomID	String	Room type

Table 4.6 Table RoomDetail

Item	Type	Description
RoomDetailID	String	The ID of room detail
RoomID	String	The ID of room

CMND(Student)	String	The student's citizen identification number is in the room
---------------	--------	--

Table 4.7 Table StyleRoom

Item	Type	Description
StyleRoomID	String	The ID of style room
Status	String	
Price	String	Cost per room type

Table 4.8 Table Account

Item	Type	Description
AccountID	String	The ID of account
CMND	String	Citizen identification number of the account holder
Password	String	Account password
RoleID	String	The ID of role

Table 4.9 Table RoleDetail

Item	Type	Description
SvID	String	The ID of service
RoleID	String	The ID of role

Table 4.10 Table Role

Item	Type	Description
RoleID	String	The ID of role
NameRole	String	Account password Account rights (divide student or admin)
Status	String	role status

Table 4.11 Table Student

Item	Type	Description
CMND	String	Citizen identification number of the student
MSSV	String	Student's student number
FullName	String	Student's full name
AccountID	String	The ID of account
Gender	String	Student's gender
SchoolName	String	Name of the student's school
Bank	String	Bank account number
RoomID	String	The ID of room
Phone	String	Student phone number
Email	String	Student emails
Photos	String	Student's photo
Address	String	Student's permanent address
BirthDay	Date	Student's date of birth
Session	String	School year

Table 4.12 Table Admin

Item	Type	Description
CMND	String	Citizen identification number of the admin
FullName	String	Full name of admin
AccountID	String	The ID of account
Gender	String	Admin's gender
RoomID	String	The ID of room
Phone	String	Admin's phone number
Email	String	Admin emails
BirthDay	Date	Admin's date of birth

Photos	String	Admin's photo
Address	String	Admin's permanent address

Table 4.13 Table Service

Item	Type	Description
SvID	String	The Id of service
NameSv	String	Name of service
CMND(admin)	String	Admin's citizen identification number
CMND(student)	String	Student's citizen identification number
DateOfIssue	Date	The day students use the service
Status	String	Service status (approved or not)

Chapter 5 Implementation

5.1 Architecture diagram

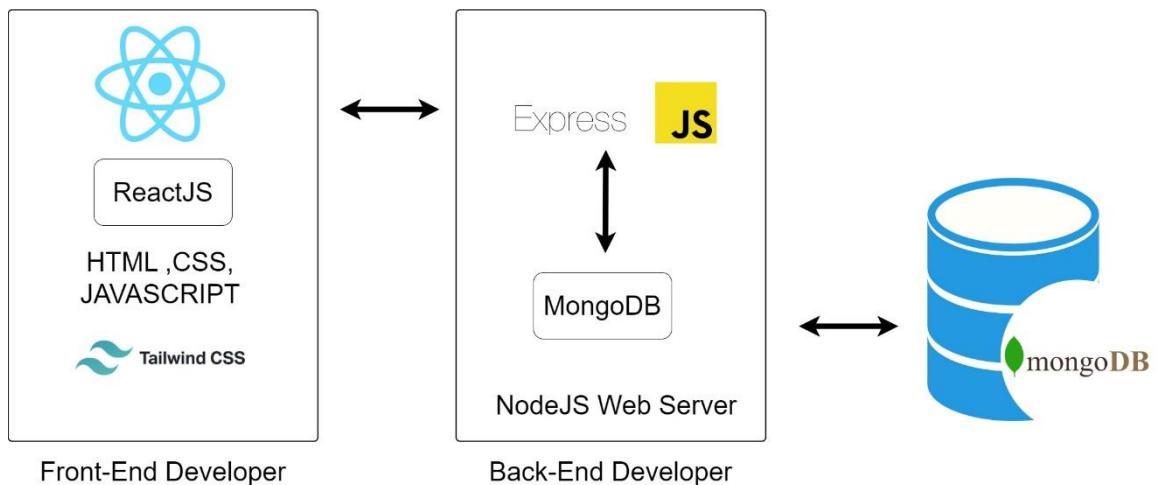


Figure 5.64 Architecture Diagram

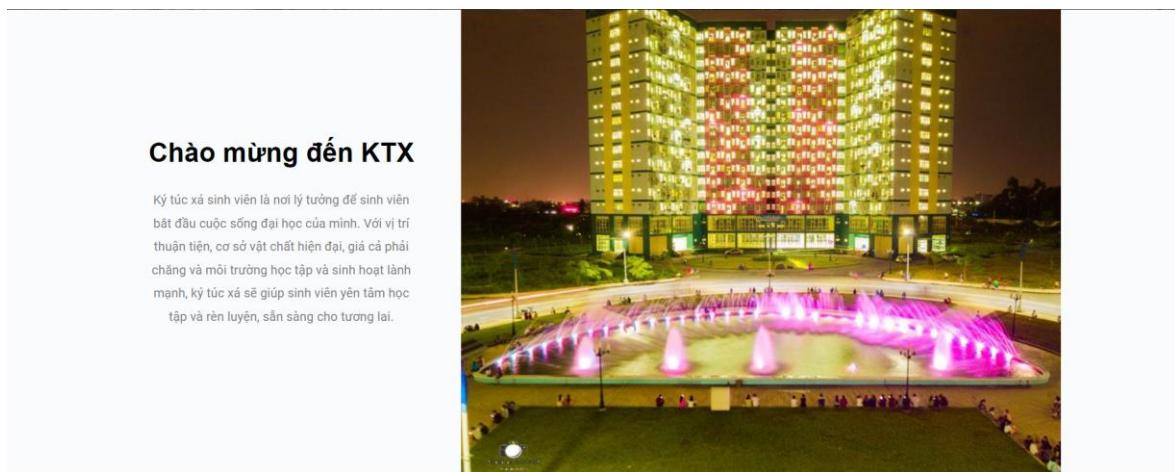
5.2 Application

5.2.1 Student role

Even without an account, students can still use the system to view some basic information, such as room details (such as the type of accommodation, its fee, and the paperwork required to get ready for work). enroll, etc.). Here's an example of how students see the system's fundamental information: they glance at the room details and system interface without logging in.



Picture 5.1 Home page interface



Chào mừng đến KTX

Ký túc xá sinh viên là nơi lý tưởng để sinh viên bắt đầu cuộc sống đại học của mình. Với vị trí thuận tiện, cơ sở vật chất hiện đại, giá cả phải chăng và môi trường học tập và sinh hoạt lành mạnh, ký túc xá sẽ giúp sinh viên yên tâm học tập và rèn luyện, sẵn sàng cho tương lai.

Picture 5.2 Home page interface

Các loại phòng

Phòng 2 <ul style="list-style-type: none"> Máy lạnh Phòng có máy lạnh Đặc điểm Không gian rộng rãi, thoáng mát Giá phòng 1,250,000đ/Tháng 	Phòng 4 <ul style="list-style-type: none"> Máy lạnh Không có máy lạnh Đặc điểm Không gian rộng rãi, thoáng mát Giá phòng 650,000đ/Tháng 	Phòng 6 <ul style="list-style-type: none"> Máy lạnh Không có máy lạnh Đặc điểm Không gian rộng rãi, thoáng mát Giá phòng 240,000đ/Tháng
--	--	--

[XEM THÊM](#)

Picture 5.3 Home page interface

The page with registration details appears next. This page simply shows information because direct registration is only permitted at the dorm and does not let students to register online, according to the laws of the dorm. Details regarding room costs, amenities, and the paperwork students need to prepare before checking into the dorm. Here is one example of that.

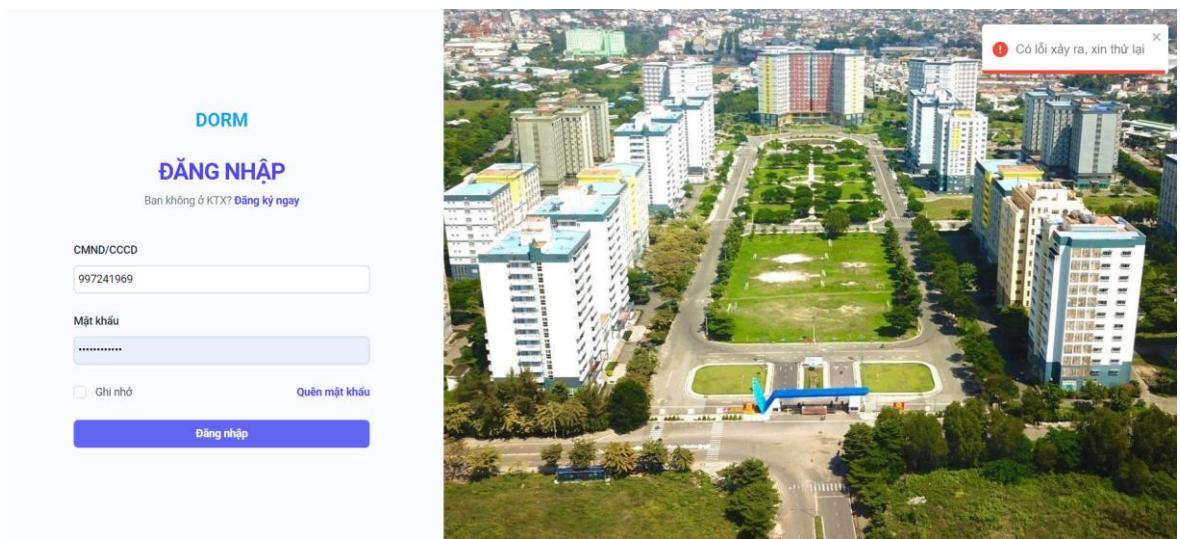
DORM	Trang chủ	Loại phòng	Dịch vụ	Đăng ký	Đăng nhập →																						
<p>THÔNG TIN SINH VIÊN ĐĂNG KÝ Ở KTX ĐHQG-HCM NH 2023-2024</p> <p>I. HƯỚNG DẪN ĐĂNG KÝ</p> <p>Để đảm bảo thời gian đăng ký ở Ký túc xá trực tuyến, sinh viên chuẩn bị đầy đủ file hình sau:</p> <ul style="list-style-type: none"> - Chứng minh nhân dân/thẻ căn cước công dân (mặt trước và mặt sau), mã số định danh cá nhân, trường hợp chưa có mã số định danh. - Thẻ Bảo hiểm Y tế. - Đổi với sinh viên: Giấy tờ minh chứng đã làm thủ tục nhập học tại CSDT (theo phụ lục VII của Thông báo số 479/TB-TTQLKTX)(*) - Đổi với sinh viên năm 2 trở lên: Thủ tục xác nhận sinh viên/thời khóa biểu học tập(*) - Hình thẻ kích thước 4 x 6 (áo sơ mi, chụp rõ khuôn mặt). <p>Sinh viên tìm hiểu kỹ thông tin sau:</p> <ul style="list-style-type: none"> - Thời gian đăng ký tối đa 20 phút. - Thời gian trả kết quả: trong 36 giờ tính từ khi hệ thống xác nhận đăng ký thành công. <p>CÁC KHOẢN THU VÀ MỨC THU</p> <p>(Ban hành kèm theo Thông báo số 479/TB-TTQLKTX ngày 15 tháng 08 năm 2023 của Trung tâm Quản lý Ký túc xá)</p> <ol style="list-style-type: none"> 1. Tiền hồ sơ: 60.000 đồng/sinh viên. 2. Tiền thuê chăn tài sản-cơ sở vật chất (TCTS-CSVC): 100.000 đồng/sinh viên. 3. Tiền BHYT: 850.500 đồng/sinh viên/15 tháng (dành cho tân sinh viên đóng BHYT tại KTX); 680.400 đồng/ sinh viên/12 tháng (dành cho tân sinh viên trường Đại học Công nghệ Thông tin). 4. Bảo hiểm tai nạn: 30.000đ/sinh viên/12 tháng. 5. Mức giá lệ phí phòng ở: Căn cứ Công văn số 1593/DHQG-KHTC ngày 09/8/2022 của ĐHQG-HCM về "Quy định mức giá lệ phí phòng ở KTX từ năm học 2022-2023 đến năm học 2025-2026 và đơn giá dịch vụ", Trung tâm thông báo mức giá lệ phí phòng ở và đơn giá dịch vụ tăng thêm như sau <p>Đơn giá từ 01/8/2023-31/8/2023 và đơn giá từ 01/9/2023-31/8/2024:</p> <p>(Đơn giá tính: đồng)</p> <table border="1"> <thead> <tr> <th rowspan="2">STT</th> <th rowspan="2">Loại phòng</th> <th colspan="3">Từ 01/8/2023 - 31/8/2023</th> <th colspan="3">Từ 01/9/2023 - 31/8/2024</th> </tr> <tr> <th>Đơn giá phòng /sinh viên/tháng</th> <th>Đơn giá dịch vụ tăng thêm/sinh viên/tháng</th> <th>Tổng đơn giá</th> <th>Đơn giá phòng /sinh viên/tháng</th> <th>Đơn giá dịch vụ tăng thêm/sinh viên/tháng</th> <th>Tổng đơn giá</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td>1</td> <td>2</td> <td>3=1+2</td> <td>4</td> <td>5</td> <td>6=4+5</td> </tr> </tbody> </table>						STT	Loại phòng	Từ 01/8/2023 - 31/8/2023			Từ 01/9/2023 - 31/8/2024			Đơn giá phòng /sinh viên/tháng	Đơn giá dịch vụ tăng thêm/sinh viên/tháng	Tổng đơn giá	Đơn giá phòng /sinh viên/tháng	Đơn giá dịch vụ tăng thêm/sinh viên/tháng	Tổng đơn giá			1	2	3=1+2	4	5	6=4+5
STT	Loại phòng	Từ 01/8/2023 - 31/8/2023			Từ 01/9/2023 - 31/8/2024																						
		Đơn giá phòng /sinh viên/tháng	Đơn giá dịch vụ tăng thêm/sinh viên/tháng	Tổng đơn giá	Đơn giá phòng /sinh viên/tháng	Đơn giá dịch vụ tăng thêm/sinh viên/tháng	Tổng đơn giá																				
		1	2	3=1+2	4	5	6=4+5																				

Picture 5.4 Register page interface

Tùy chọn trước khi sau																																									
<p>Đơn giá từ 01/8/2023-31/8/2023 và đơn giá từ 01/9/2023-31/8/2024:</p> <p>(Đơn giá tính: đồng)</p> <table border="1"> <thead> <tr> <th rowspan="2">STT</th> <th rowspan="2">Loại phòng</th> <th colspan="3">Từ 01/8/2023 - 31/8/2023</th> <th colspan="3">Từ 01/9/2023 - 31/8/2024</th> </tr> <tr> <th>Đơn giá phòng /sinh viên/tháng</th> <th>Đơn giá dịch vụ tăng thêm/sinh viên/tháng</th> <th>Tổng đơn giá</th> <th>Đơn giá phòng /sinh viên/tháng</th> <th>Đơn giá dịch vụ tăng thêm/sinh viên/tháng</th> <th>Tổng đơn giá</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Phòng 2 sinh viên có máy lạnh</td> <td>925,000</td> <td>420,000</td> <td>1,345,000</td> <td>1,250,000</td> <td>420,000 1,670,000</td> </tr> <tr> <td>2</td> <td>Phòng 4 sinh viên</td> <td>500,000</td> <td>-</td> <td>500,000</td> <td>500,000</td> <td>- 500,000</td> </tr> <tr> <td>3</td> <td>Phòng 6 sinh viên</td> <td>215,000</td> <td>-</td> <td>215,000</td> <td>240,000</td> <td>- 240,000</td> </tr> </tbody> </table> <p>Ghi chú:</p> <ul style="list-style-type: none"> - Mức đóng cho các loại phòng trên chưa bao gồm tiền sử dụng điện, nước và các dịch vụ khác. - Sinh viên được nhận lại tiền TCTS-CSVC đã đóng khi rời khỏi KTX. - Mức lệ phí phòng ở từ 01/9/2024-31/8/2026: Sinh viên xem chi tiết phụ lục IV - Loại phòng này sẽ được hiển thị trên phần mềm để sinh viên lựa chọn khi đăng ký. Sinh viên 							STT	Loại phòng	Từ 01/8/2023 - 31/8/2023			Từ 01/9/2023 - 31/8/2024			Đơn giá phòng /sinh viên/tháng	Đơn giá dịch vụ tăng thêm/sinh viên/tháng	Tổng đơn giá	Đơn giá phòng /sinh viên/tháng	Đơn giá dịch vụ tăng thêm/sinh viên/tháng	Tổng đơn giá	1	Phòng 2 sinh viên có máy lạnh	925,000	420,000	1,345,000	1,250,000	420,000 1,670,000	2	Phòng 4 sinh viên	500,000	-	500,000	500,000	- 500,000	3	Phòng 6 sinh viên	215,000	-	215,000	240,000	- 240,000
STT	Loại phòng	Từ 01/8/2023 - 31/8/2023			Từ 01/9/2023 - 31/8/2024																																				
		Đơn giá phòng /sinh viên/tháng	Đơn giá dịch vụ tăng thêm/sinh viên/tháng	Tổng đơn giá	Đơn giá phòng /sinh viên/tháng	Đơn giá dịch vụ tăng thêm/sinh viên/tháng	Tổng đơn giá																																		
1	Phòng 2 sinh viên có máy lạnh	925,000	420,000	1,345,000	1,250,000	420,000 1,670,000																																			
2	Phòng 4 sinh viên	500,000	-	500,000	500,000	- 500,000																																			
3	Phòng 6 sinh viên	215,000	-	215,000	240,000	- 240,000																																			

Picture 5.5 Register page interface

The student will receive an account from the dorm management once they have properly registered and completed the required paperwork. Students can then log into the system after that. An example of logging into the system in both successful and unsuccessful scenarios can be found below. The first scenario is when a student enters erroneous account information and is unable to log in.



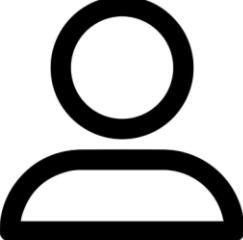
Picture 5.6 Login page interface

The system will show a notification on the right side of the screen if an attempt is made to log in. Additionally, in order for students to access the system, they will need to reenter their username and password. This is an example of how the system will reroute to the student page following a successful login.



Picture 5.7 Page for students after logging in

We can see it is similar to the website when not logged in, but here there will be a few differences. The function to view student account information has replaced the old login button. Additionally, full student account information is displayed when we click on it. Here is one example of that.



DORM Trang chủ Loại phòng Dịch vụ Đăng ký 

Thông tin sinh viên [Đổi mật khẩu](#) [Đổi tài khoản ngân hàng](#) [Đổi số điện thoại](#)

Họ tên	Số điện thoại
Phan Huy Vũ	0902369741
CMND/CCCD	MSSV
077396411635	19521896
Email	Ngân hàng
vuph@gmail.com	Sacombank - 060298763201
Địa chỉ	
20/5 Phan Văn Trị, Phường 5, Quận Gò Vấp, TP HCM	
Loại	Niên khóa

Picture 5.8 View student information page

As you can see, the avatar that appears when you first create an account is the default one, however you can modify it. Furthermore, student are unable to alter personal data (per the rules of the dorm; to make changes, speak with the manager of the dorm). Only certain items' information, such as the profile image, password, bank account number, and phone number, can be altered. The first step is to alter the avatar picture; an example is provided below.



DORM Trang chủ Loại phòng Dịch vụ Đăng ký 

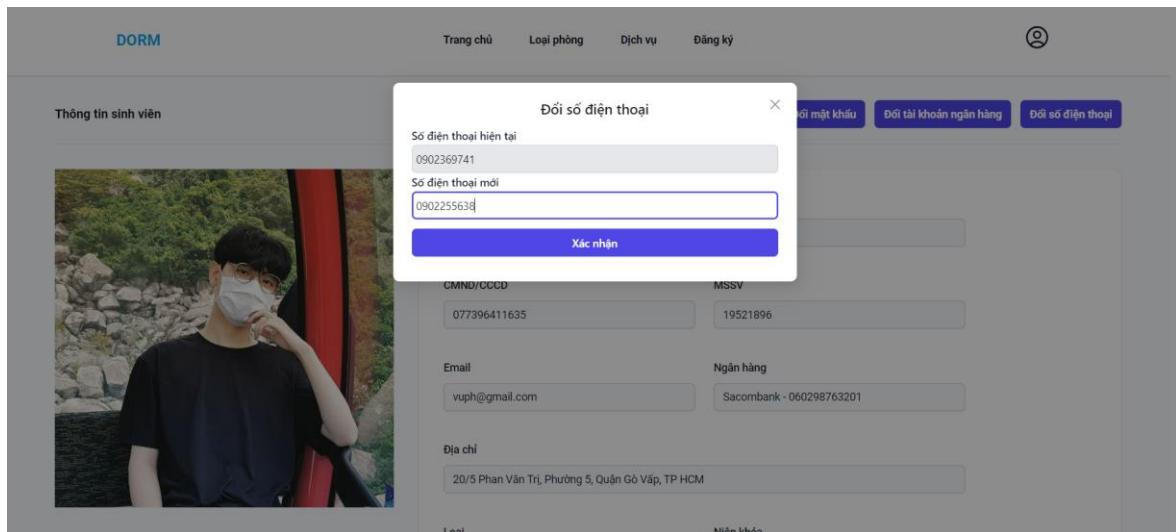
Thông tin sinh viên [Đổi mật khẩu](#) [Đổi tài khoản ngân hàng](#) [Đổi số điện thoại](#)

Họ tên	Số điện thoại
Phan Huy Vũ	0902369741
CMND/CCCD	MSSV
077396411635	19521896
Email	Ngân hàng
vuph@gmail.com	Sacombank - 060298763201
Địa chỉ	
20/5 Phan Văn Trị, Phường 5, Quận Gò Vấp, TP HCM	
Loại	Niên khóa

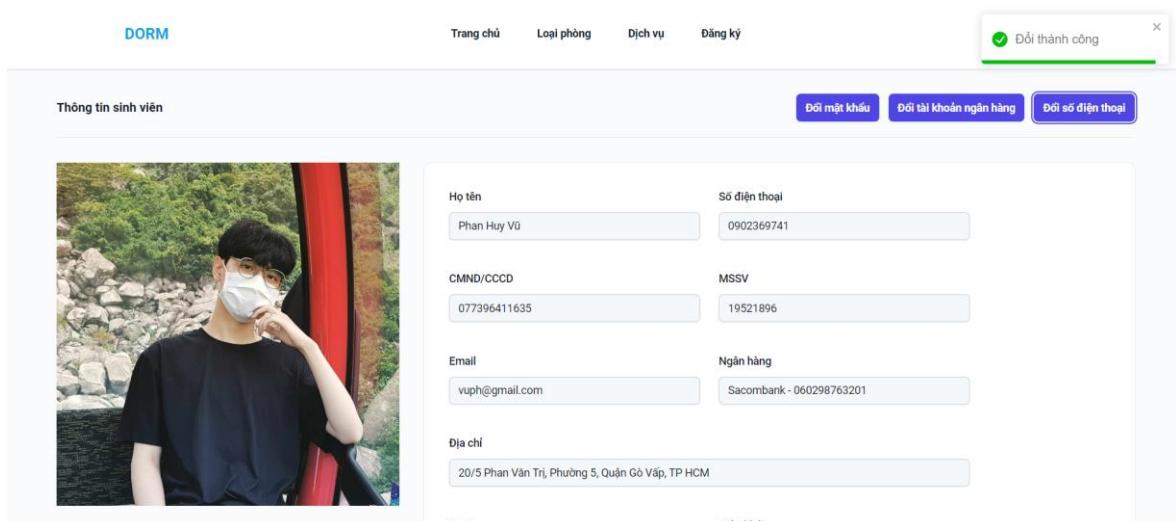
Picture 5.9 View student information page

Next is to change the phone number. The system will show details about the previous phone number and a field to enter the new phone number when we choose the

phone number changing option. The system will update the new phone number for the account after we click "confirm" after entering the new number.

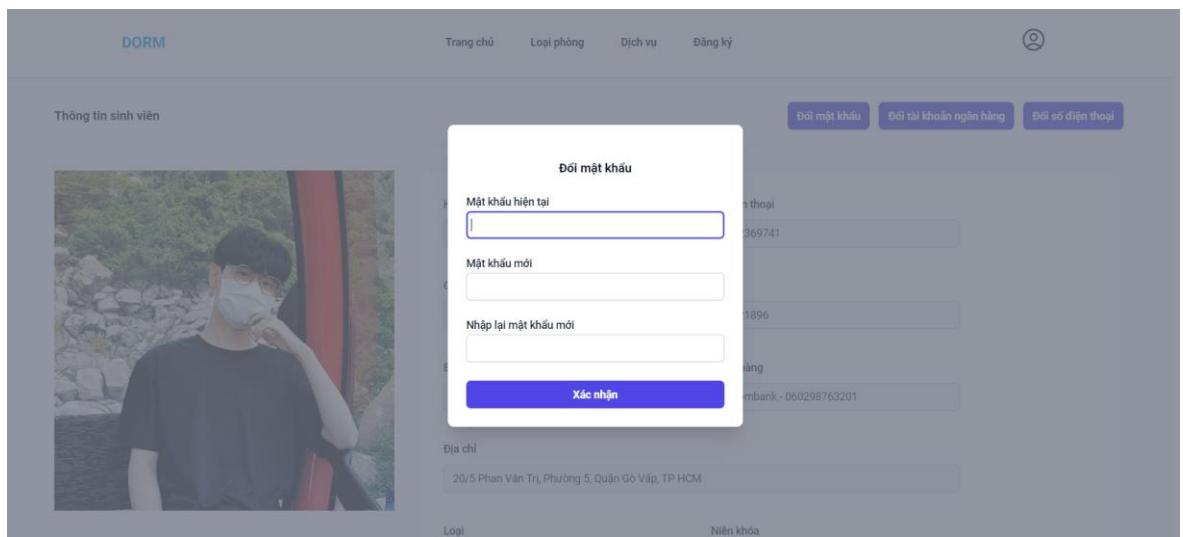


Picture 5.10 Illustration of the phone number change function

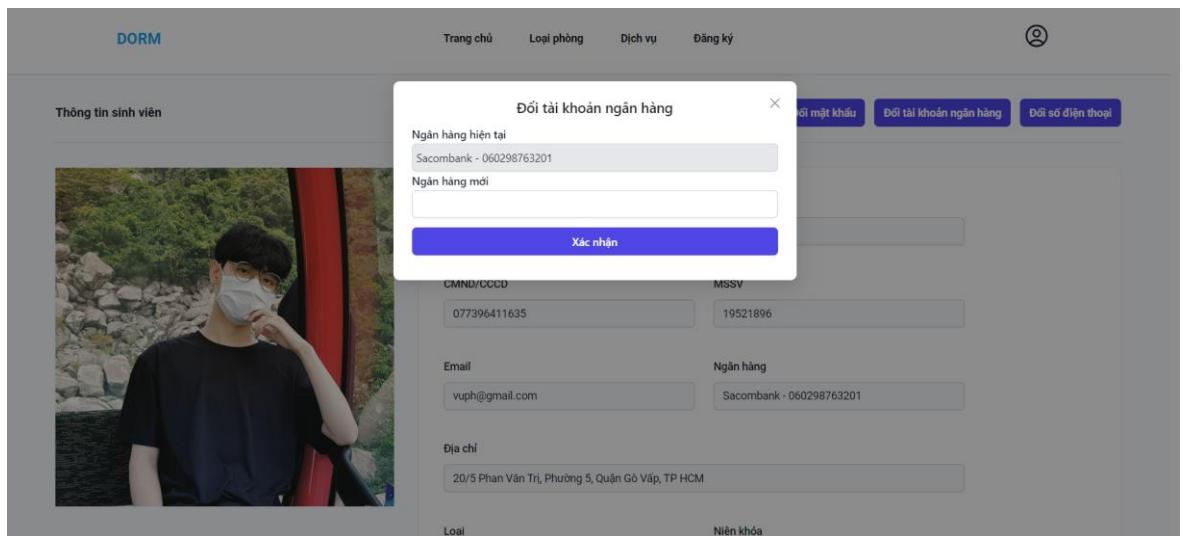


Picture 5.11 Illustration of the phone number change function (after change)

Changes to the password and bank account number are made in a manner akin to those of phone number changes. Below is an illustration of updating a bank account number and changing an account password.

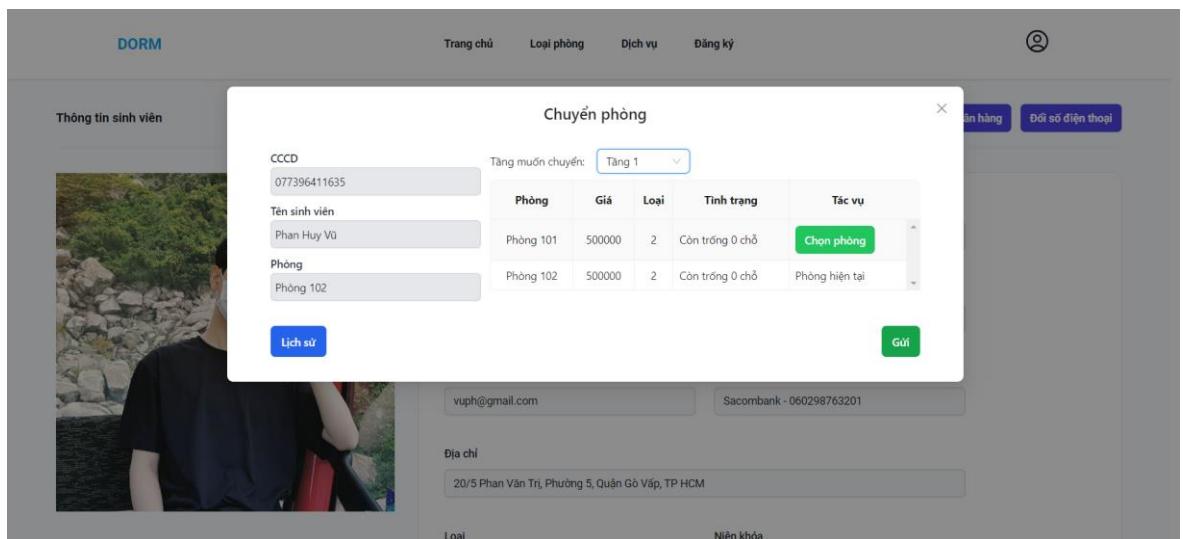


Picture 5.12 Illustration of the password change function



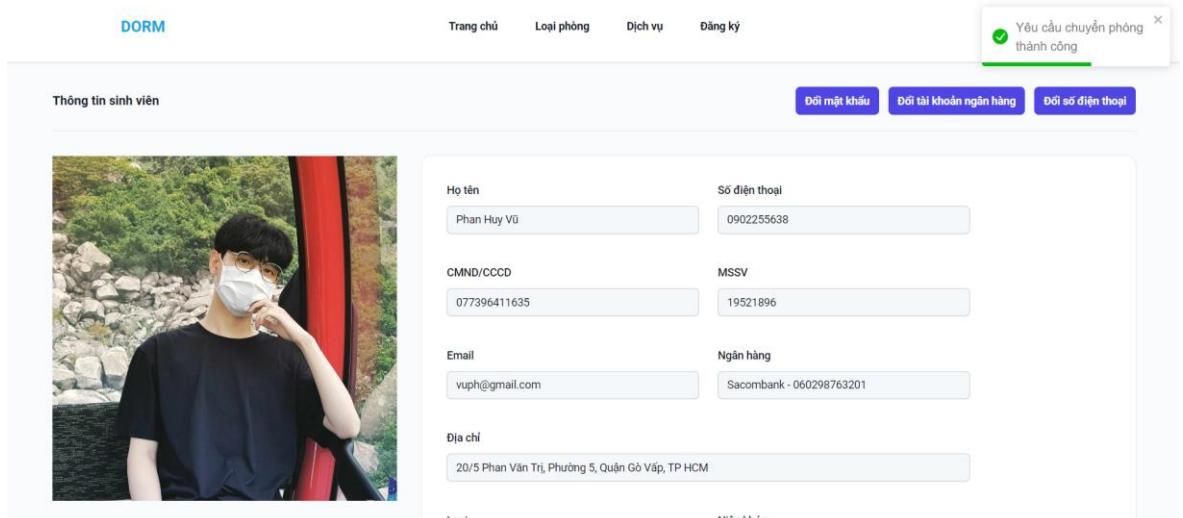
Picture 5.13 Illustration of the bank account number change function

Next, there are other features available in the student account for students who are living in the dorm, including sending requests for check-out, transfer rooms, fixing rooms when there are issues with the amenities, and viewing invoices. The first will provide an example of how to submit a request for a room change.



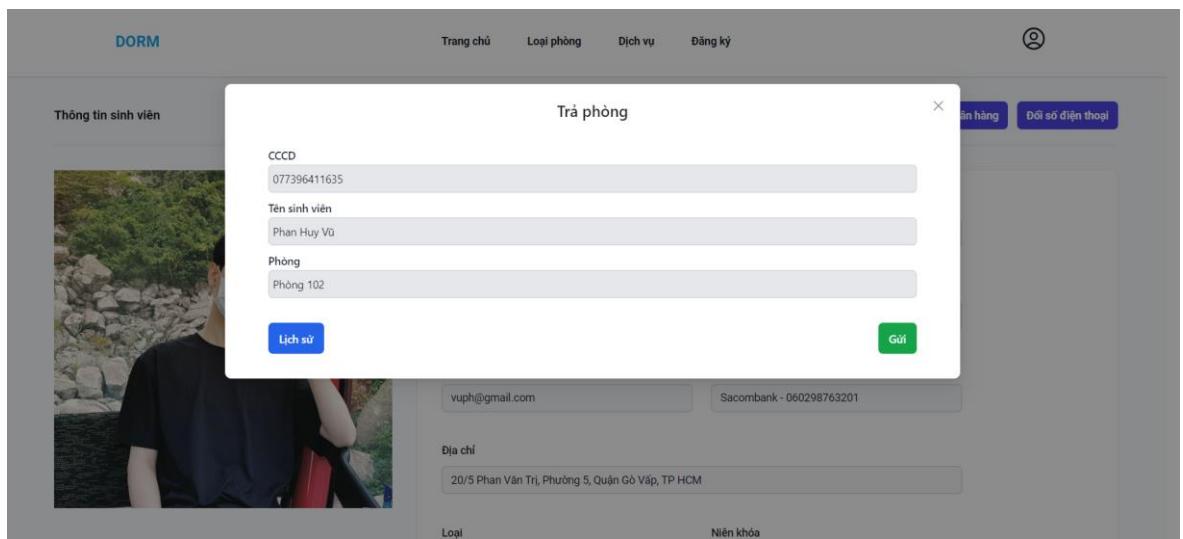
Picture 5.14 Illustration of the room transfer function

Following our floor selection, a list of the rooms and the number of seats available in each will appear. When we click Submit after choosing the room, the data is saved by the system. Here's an example of that as well.



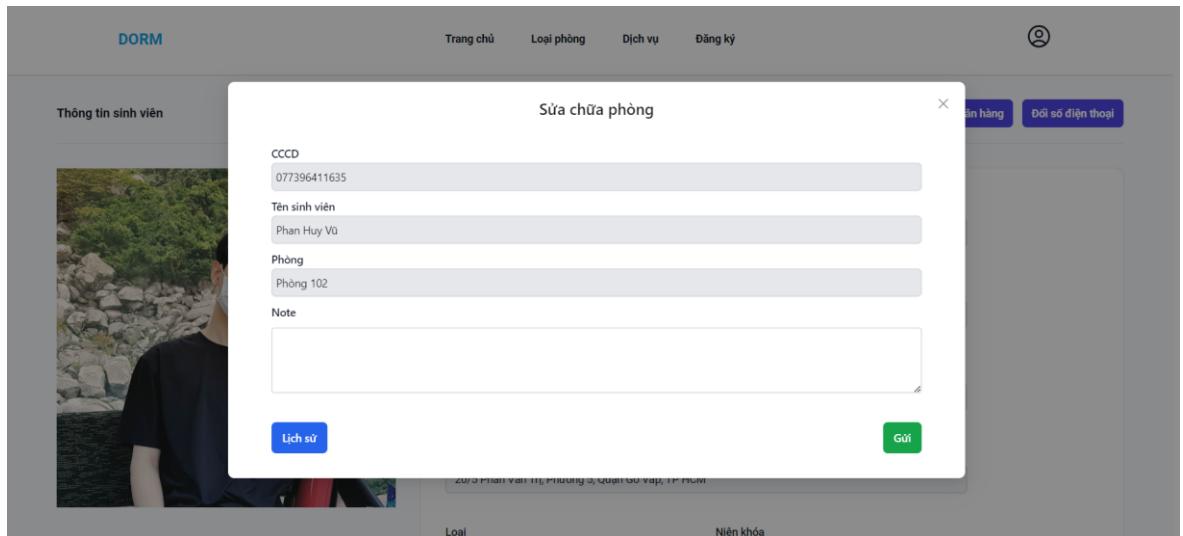
Picture 5.15 Illustration of the room transfer request function (after send request)

Similar to this, the system will show us some basic account information when we click to Check out, and we can click Submit if we agree.



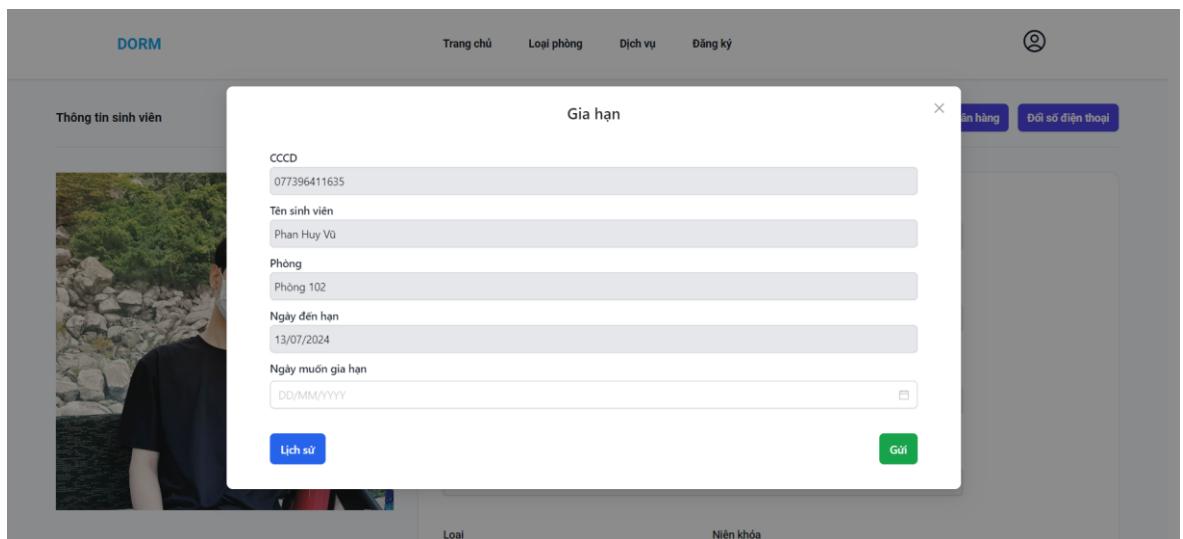
Picture 5.16 Illustration of the room check-out request function

Next, in the room repair part, students can fill in the facilities that need repair and any extra information if needed in the Note filed, which will also display basic information about the student making the request. An example of submitting a request for room repairs is provided below.



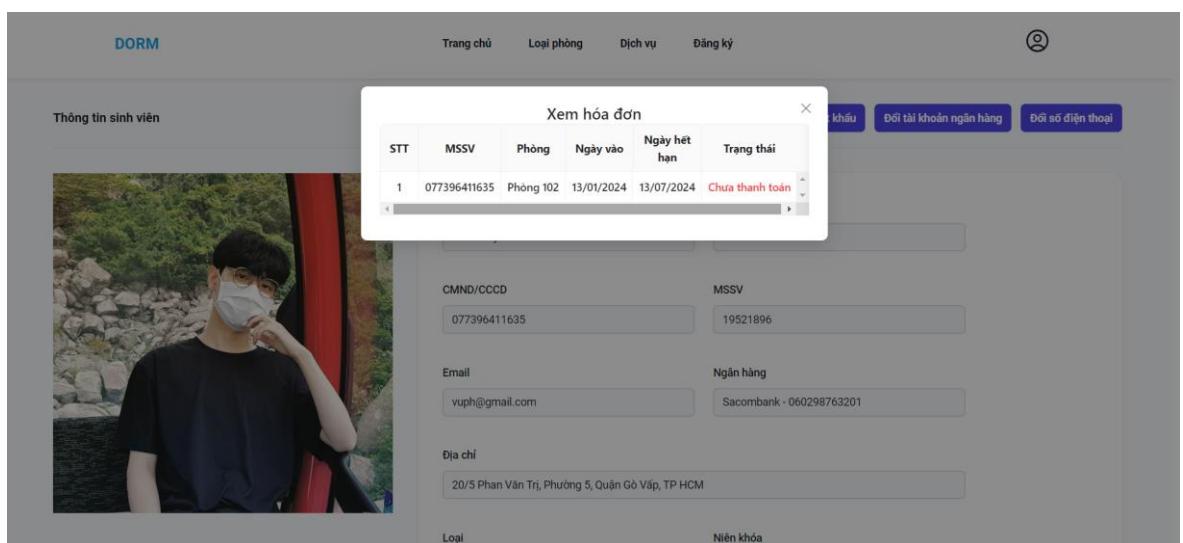
Picture 5.17 Illustration of the room repair request function

The renewals of stay at the dormitory comes next. Here, the system will present some student data and let the student select the date of renewal when the student selects this function.



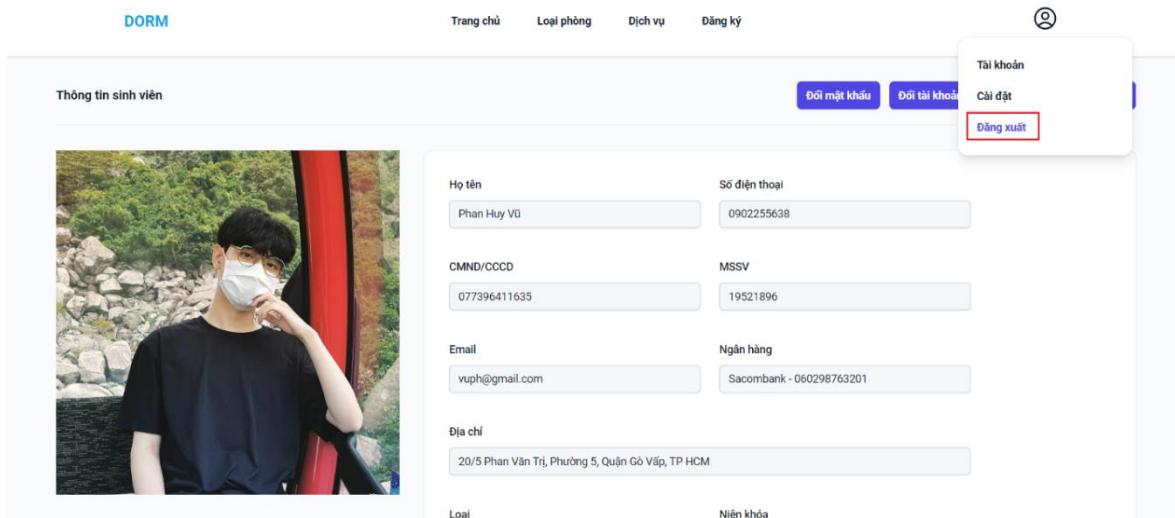
Picture 5.18 Illustration of the renewals request function

The invoice view part is the last one. The system will show student invoices in this function.



Picture 5.19 Illustration of the view invoice function

Students can click to log out of the system at any time, and the system will redirect students to the homepage without logging into their account. Here is one example of that.



Picture 5.20 Illustration of the log out function



Picture 5.21 Illustrate the system after logging out

The functions that a student account can perform are listed above; the actions that a dorm management can perform on the system will be covered next.

5.2.2 Admin role

First, the admin must log in to the account, just like with the birth account. If the login attempt is unsuccessful because of inaccurate account information, a failure warning will also appear. An example of an admin accessing the admin's management page successfully is shown below.



Picture 5.22 Admin page after logging in

Upon selecting the Admin website, the user will be redirected to the website where information about the dormitory, including rooms and students, is managed. The student management example comes first.

STT	Họ tên	CCCD/CMND	Phòng	Số điện thoại	Email	Trường	Ngân hàng	Tác vụ
1	Admin 1	999888777	0111222334	Test@gmail.com	UIT	Sacombank - 0000000000000000		
2	Nguyen Thanh Loc	997241969423	0902789654	9972419694@gmail.com	Đại học Bách khoa	Sacombank - 0000000000000000		
3	Thanh Loc	99990999	999099999	0999@gmail.com	Đại học Bách khoa	BIDV		
4	Cao Hoàng Khang	79201025910	778748306	tinlun0386@gmail.com	Đại học Công nghệ Thông tin	MB-0778748301		
5	Ngô Đức Hiền	79219227251	913825719	hien123@gmail.com	Đại học Khoa học Tự nhiên	MB-0913825719		
6	Nguyễn Minh Đức	79281724812	903134398	ducnguyen123@gmail.com	Đại học Bách khoa	MB-0903134398		
7	Nguyễn Đức Khải	79201925812	912018212	khai123@gmail.com	Đại học Công nghệ Thông tin	MB-0912018212		

Picture 5.23 The page displays the student list

We can change student information, remove students, and search for students at the admin line. The administrator has the ability to search by phone number, bank account, name, school, and citizen identification card. An example of looking up students by name may be found below.

Picture 5.24 Illustration of the student search function

The next step involves changing student data; in this case, the administrator can make changes to data that, as stated in the Student role section, students are not permitted to make. The system will show student information when the Update option is selected. The administrator can then add new information and click "Confirm."

Picture 5.25 Illustration of the student update function

The system will notify the administrator that the update was successful.

STT	Họ tên	CCCD/CMND	Phòng	Số điện thoại	Email	Trường	Ngân hàng	Tác vụ
1	Admin 1	999888777		0111222334	Test@gmail.com	UIT	Sacombank - 06	<button>Cập nhật</button> <button>Xóa</button>
2	Nguyen Thanh Loc	997241969423		0902789654	9972419694@gmail.com	Đại học Bách khoa	Sacombank - 06	<button>Cập nhật</button> <button>Xóa</button>
3	Thanh Loc	999909999		999099999	0999@gmail.com	Đại học Bách khoa	BIDV	<button>Cập nhật</button> <button>Xóa</button>
4	Cao Hoàng Khang	79201025910		778748306	tinlun0386@gmail.com	Đại học Công nghệ Thông tin	MB-0778748306	<button>Cập nhật</button> <button>Xóa</button>
5	Ngô Đức Hiển	79219227251		913825719	hien123@gmail.com	Đại học Khoa học Tự nhiên	MB-0913825719	<button>Cập nhật</button> <button>Xóa</button>
6	Nguyễn Minh Đức	79281724812		903134398	ducnguyen123@gmail.com	Đại học Bách khoa	MB-0903134398	<button>Cập nhật</button> <button>Xóa</button>
7	Nguyễn Đức Khải	79201925812		912018212	khai123@gmail.com	Đại học Công nghệ Thông tin	MB-0912018212	<button>Cập nhật</button> <button>Xóa</button>

Picture 5.26 Illustration of the student update function (after update)

Subsequently, if the administrator has to remove a student who has left the dorm, they can pick the delete option next to the student and click confirm. Here's an example of removing pupils.

STT	Họ tên	Trường	Ngân hàng	Tác vụ
14	Minh Hiếu	Đại học Công nghệ Thông tin	BIDV - 8770541	<button>Cập nhật</button> <button>Xóa</button>
15	Nguyễn Quốc Thủ	Đại học Công nghệ Thông tin	MB - 0903928111	<button>Cập nhật</button> <button>Xóa</button>
16	Nguyễn Văn Minh	Đại học Công nghệ Thông tin	MB - 0903827111	<button>Cập nhật</button> <button>Xóa</button>
17	Minh Xuân	Đại học Công nghệ Thông tin	MB - 0902938111	<button>Cập nhật</button> <button>Xóa</button>
18	Nguyễn Thành Lộc	Đại học Công nghệ Thông tin	Sacombank - 06	<button>Cập nhật</button> <button>Xóa</button>
19	Phan Huy Vũ	Đại học Bách khoa	Sacombank - 06	<button>Cập nhật</button> <button>Xóa</button>

Picture 5.27 Illustration of the student delete function

Next, students are only permitted to register in person at the dorm, as was stated in the section on the student role. Here's an example of how an administrator sets up a student's account when they check into the dorm.

The screenshot shows the DORM application interface. On the left is a sidebar with navigation options: Dashboard, Sinh viên (selected), Danh sách sinh viên, Thêm sinh viên (highlighted in red), Quản lý phòng, Quản lý hóa đơn, and Quản lý đơn. The main area displays a table titled "Danh sách Sinh viên" with columns: STT, Họ tên, CCCD/CMND, Phòng, Số điện thoại, Email, Trường, Ngân hàng, and Tác vụ (Edit and Delete buttons). There are 7 rows of student data. An inset window titled "Thông tin cá nhân" is open, showing a placeholder user icon and input fields for: Họ tên (Nguyễn Văn Minh Đức), Số điện thoại (0903827185), CMND/CCCD (079273829175), MSSV (20520481), Email (minhduc123@gmail.com), Ngân hàng (MB - 0903827185), Địa chỉ (82 đường bình tri đồng A), Loại (Sinh viên), Niên khóa (2020-2024), and Trường (Đại học Công nghệ Thông tin).

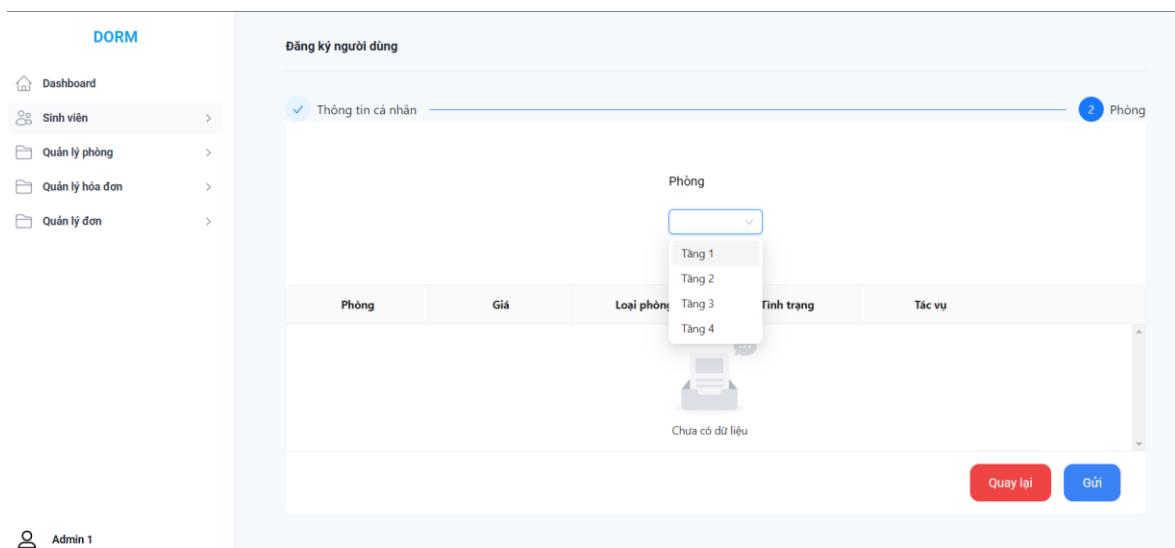
Picture 5.28 Illustration of the student insert function

The system will show student information when the admin chooses the "add students" feature, and the admin will then input student data into the system. An example of how to add recently registered students to the system is shown below. First, complete the student's required information.

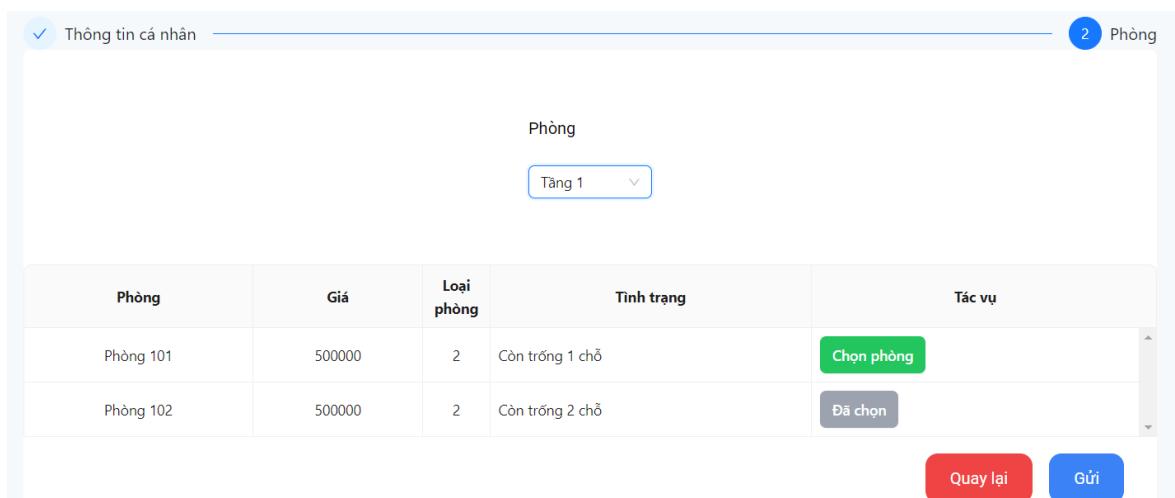
The screenshot shows the DORM application interface. On the left is a sidebar with navigation options: Sinh viên (selected), Quản lý phòng, Quản lý hóa đơn, and Quản lý đơn. The main area displays a form titled "Thông tin cá nhân" with fields for: Họ tên (Nguyễn Văn Minh Đức), Số điện thoại (0903827185), CMND/CCCD (079273829175), MSSV (20520481), Email (minhduc123@gmail.com), Ngân hàng (MB - 0903827185), Địa chỉ (82 đường bình tri đồng A), Loại (Sinh viên), Niên khóa (2020-2024), and Trường (Đại học Công nghệ Thông tin).

Picture 5.29 Illustration of the student insert function

Students should then be added to the room they have already chosen, provided that room is available.



Picture 5.30 Select floor when adding students



Picture 5.31 Select room when adding students

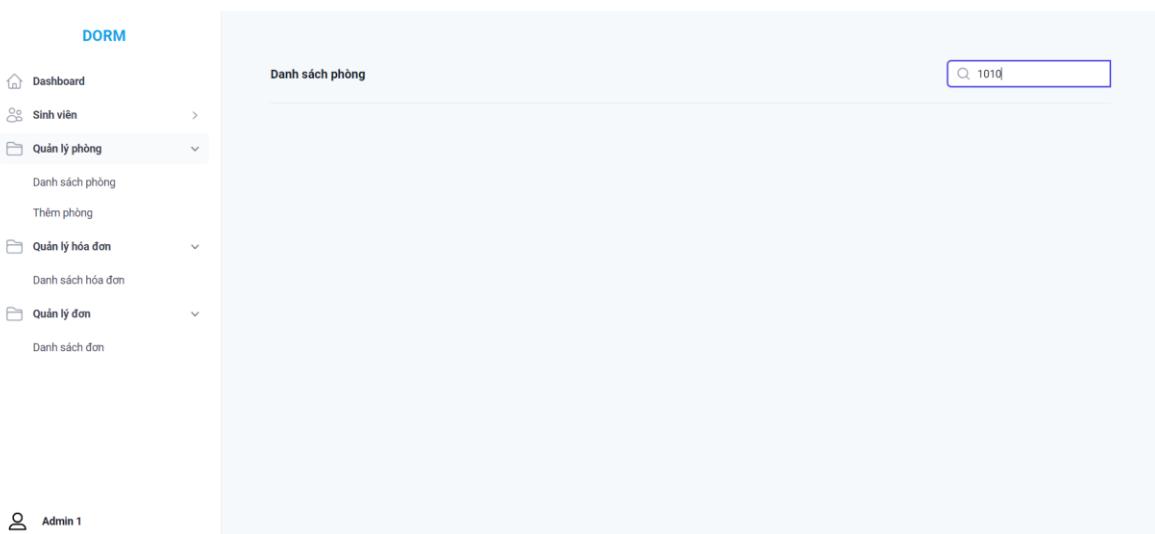
The management of dorm rooms comes next. Upon choosing Room Management and subsequently Room List, the system will present a list of every dorm room, arranged according to floor, along with the room's status (available or reserved sufficient number of student in room). This is an example of that below.

Picture 5.32 The page displays the room list

Here, we can make use of the room search feature. Just type the room number into the search box, and the system will show the results of the relevant search.

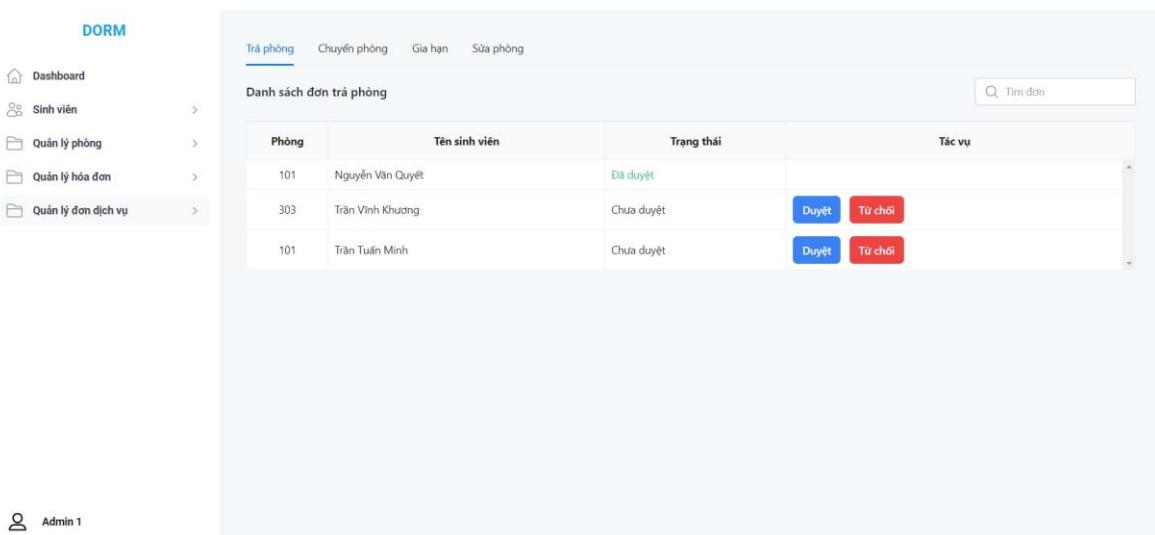
Picture 5.33 Illustration of the room search function

The system won't show any information about any room if it isn't included among the dorm rooms. An example of the room not being discovered is shown below.



Picture 5.34 Illustration of the student insert function (No information found)

Next, the admin can handle student requests for room transfers, check-outs, repairs, and extensions of dorm stays using the admin account. Depending on each situation, the administrator may decide to approve or reject this. An example of an administrator handling student queries is seen below. The check-out request form is the first.



Picture 5.35 Illustration of the check-out request approval function

The second is a room transfer request.

Picture 5.36 Illustration of the room transfer request approval function

Next is the Renewal application.

Picture 5.37 Illustration of the renewals request approval function

Finally, there is the room repair request.

Danh sách đơn sửa phòng

Phòng	Tên sinh viên	Tin nhắn	Trạng thái	Tác vụ
102	Ngô Tuấn Nhì	Quạt trần bị hư	Chưa duyệt	<button>Đọc</button> <button>Reject</button>
204	Nguyễn Minh Bảo	Nhà vệ sinh bị tắc	Chưa duyệt	<button>Đọc</button> <button>Reject</button>
303	Nguyễn Tiến Vinh	Đèn bị hỏng	Đã duyệt	

meet.google.com đang chia sẻ màn hình của bạn. Đang chia sẻ X

Picture 5.38 Illustration of the room repair request approval function

When the administrator has finished her work, they can click to log out, just like with student accounts. The administrator will then be redirected by the system to the Home page if the user is not logged in.

Chapter 6 Conclusion

6.1 Results

6.1.1 Advantage

Website can be deployed on any computer with many different platforms.

Security: clear decentralization system for each type of account, corresponding to the authority of different positions.

Easy to use: The program is designed on a graphical interface with easy-to-use windows and page.

Improve student and room management effectively and quickly.

Helps students easily grasp information about dormitories easily.

6.1.2 Disadvantage

The system is still primitive and does not have many features.

Most of the features are only at the basic level and not yet advanced.

The level of perfection is not high yet.

There may be many errors.

6.2 Future development direction

With the goal of long-term development and widespread development, The Student Dormitory Management System product has a number of development directions to improve the quality of experience for both administrators and students in the future, such as:

Build more mobile applications to help students more easily manage information.

Develop some additional features such as allowing online payments, invoice statistics,...

6.3 Conclusion

During the process of researching and implementing the student dormitory management project, we became aware of the importance of improving and optimizing

the dormitory management system to meet the increasing demand. of students. Our team has focused on analyzing system design to improve dormitory performance while providing the best living experience for students.

Compared to the initially set goals, most of the projects have been completed, but there are still a few goals that the project has not been able to complete, such as: allowing students to pay bills online, collecting collect and analyze information about student housing trends, preferences, and facility usage to inform future choices and improve the campus living experience. Although the project still needs to be improved in some points, through this project our group has gained many benefits, such as reviewing the knowledge and skills learned throughout the study process. at school and develop them further to be able to complete this project. In addition, our group also gained a lot of experience and new knowledge, for example, we were able to understand the actual operating procedures of student dormitories, and learned more practical Professional knowledge about the regulations. Student management procedure, understanding the process of creating ideas and completing a product. All of the above are the knowledge and experience needed when working in real environments at businesses, and this project helps us gain them.

In summary, through this The Student Dormitory Management System project, we hope that the suggestions and insights from our research will contribute to the improvement of student dormitory management, create the best learning environment, and promote the comprehensive development of students.

REFERENCES

- [1] Bytesoft, “Lập trình C# và những điều có thể bạn chưa biết (C# programming and some unknown facts)”, January 12, 2024. [Online]. Available: <https://bytesoft.vn/lap-trinh-c-va-nhung-dieu-co-the-ban-chua-biet> [Accessed October 11, 2023].
- [2] FPT Software Academy, “Angular là gì? Ưu và Nhược điểm (What Is Angular? Positives and negatives)”, November 18, 2021. [Online]. Available: <https://fsoft-academy.edu.vn/tin-tuc-su-kien/angular-la-gi-uu-va-nhuoc-diem/> [Accessed October 11, 2023].
- [3] Got It Vietnam, “Tìm hiểu về Nodejs và những ưu nhược điểm của Nodejs (Learn about Node.js, including its benefits and drawbacks)”, February 03, 2021. [Online]. Available: <https://vn.got-it.ai/blog/tim-hieu-ve-nodejs-va-nhung-uu-nhuoc-diem-cua-nodejs> [Accessed October 11, 2023].
- [4] Hybrid Technologies, “Ưu và nhược điểm của MongoDB (Pros and cons of MongoDB)”, September 28, 2022. [Online]. Available: <https://jobs.hybrid-technologies.vn/blog/uu-nhuoc-diem-mongodb/> [Accessed October 11, 2023].
- [5] TopDev, “MongoDB là gì? Định nghĩa đầy đủ và chi tiết nhất về MongoDB (What is MongoDB? The most comprehensive and in-depth explanation of MongoDB)”, February 8, 2023. [Online]. Available: <https://topdev.vn/blog/mongodb-la-gi/> [Accessed October 11, 2023].
- [6] FPT Smart Cloud, “Microsoft Access là gì? Ưu điểm và cách sử dụng phần mềm Access (What is Microsoft Access? Advantages and how to use Access software)”, October, 2023. [Online]. Available: <https://microsoft.fptcloud.com/kien-thuc/microsoft-access-la-gi/#:~:text=%C4%90i%E1%BB%83m%20m%E1%BA%A1nh%20c%E1%BB%A7a%20Microsoft%20Access,th%C3%B4ng%20qua%20l%E1%BA%ADp%20tr%C3%ACnh%20VBA> [Accessed October 11, 2023].
- [7] LANIT JSC, “Visual basic là gì? Những điều cơ bản nhất bạn cần biết (What is Visual Basic? The essential knowledge that you should possess)”, September 29, 2022. [Online]. Available: <https://lanit.com.vn/visual-basic-la-gi.html> [Accessed October 11, 2023].
- [8] Team Việt Dev, “SQL Server là gì? Lịch sử hình thành SQL Server như thế nào? Tại sao sử dụng SQL Server? Những ưu điểm và nhược điểm của SQL Server? (What is the SQL Server model? What is the background of SQL Server creation? The purpose of SQL Server? What benefits and drawbacks does SQL Server have?)”, August 19, 2023. [Online]. Available: <https://teamvietdev.com/sql-server-la-gi/> [Accessed October 11, 2023].
- [9] GIANG (Bizfly Cloud), “HTML là gì? Ưu nhược điểm và vai trò của HTML trong lập trình web? (What is HTML? What are the benefits, drawbacks, and functions of HTML in web programming?)”, November 19, 2021. [Online]. Available: <https://bizflycloud.vn/tin-tuc/html-la-gi-20211119162558126.htm> [Accessed October 12, 2023].
- [10] W3schools, “HTML Tutorial”, October 11, 2023. [Online]. Available: <https://www.w3schools.com/html/default.asp> [Accessed October 12, 2023].

- [11] Web4s, “CSS là gì? Vai trò của CSS đối với website(Just what is CSS? The function of CSS in relation to websites)”, April 12, 2021. [Online]. Available: <https://web4s.vn/css-la-gi.html> [Accessed October 12, 2023].
- [12] W3schools, “CSS Tutorial”, October 11, 2023. [Online]. Available: <https://www.w3schools.com/css/default.asp> [Accessed October 12, 2023].
- [13] GREEN ACADEMY, “JAVASCRIPT LÀ GÌ? ÚNG DỤNG CỦA NGÔN NGỮ JAVASCRIPT (Just what is JavaScript? use of the JavaScript language)”, June 20, 2022. [Online]. Available: <https://www.greenacademy.edu.vn/kien-thuc-lap-trinh/javascript-la-gi-ung-dung-cua-ngon-ngu-javascript> [Accessed October 12, 2023].
- [14] W3schools, “JavaScript Tutorial”, October 11, 2023. [Online]. Available: <https://www.w3schools.com/js/default.asp> [Accessed October 12, 2023].
- [15] Nguyễn Thị Huyền (Vnskills Academy), “Học React JS có ưu, nhược điểm gì? Có nên học React JS không? (What benefits and drawbacks come with learning React JS? Must one learn React JS?)”, January 3, 2023. [Online]. Available: <https://vnskills.edu.vn/hoc-react-js/> [Accessed October 12, 2023].
- [16] W3schools, “React Tutorial”, October 12, 2023. [Online]. Available: <https://www.w3schools.com/react/default.asp> [Accessed October 12, 2023].
- [17] Đỗ Ngọc Quang Huy, *Xây dựng hệ thống website quản lý ký túc xá Trường Đại học Sư phạm Kỹ thuật (Constructing a website system for the University of Technical Education's dorm management)*, Graduation thesis, University of Technical Education - University of Danang, Da Nang city, 2022.
- [18] Trần Thị Huong, *Quản lý ký túc xá (Dormitory management)*, Graduation thesis, Vinh University, Vinh city, 2006.
- [19] Phan Dinh Chien and Tran Van Huan, *Xây dựng phần mềm quản lý ký túc xá (developing software for dormitory management)*, Graduation thesis, Hanoi University of Industry, Ha Noi, 2014.
- [20] joboko, “Flutter là gì? Ưu, nhược điểm của Flutter (What is Flutter? Advantages and disadvantages of Flutter)”, January 5, 2022. [Online]. Available: <https://vn.joboko.com/blog/flutter-la-gi-nwi4800> [Accessed October 11, 2023].
- [21] FPT CLOUD, “Firebase là gì? Ưu nhược điểm và Các dịch vụ của Firebase (What is Firebase? Firebase Pros and Cons and Services)”, December 31, 2021. [Online]. Available: <https://fptcloud.com/firebase-la-gi/> [Accessed October 11, 2023].

