

VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY
UNIVERSITY OF INFORMATION TECHNOLOGY
FACULTY OF INFORMATION SYSTEM



PROJECT REPORT
INFORMATION SYSTEM ANALYSIS AND DESIGN
COFFEE SHOP MANAGEMENT SYSTEM

Members:

Nguyen Khanh Van – 21522781 (Leader)

Le Thi Kieu Lam – 21522275

Duong Truc Uyen – 21522780

Dang Quang Nhat – 21522413

Class: IS201.N23.HTCL

Instructors: Ph.D. Cao Thi Nhan, MSc. Duong Phi Long

HO CHI MINH CITY, 2023

ACKNOWLEDGMENTS

First of all, we would like to express our sincere thanks to all of the teachers and students of the University of Information Technology - Vietnam National University Ho Chi Minh City and the teachers of the Information System Faculty for helping us to have the basic knowledge that is the basis for implementing this topic.

In particular, the authors would like to send our deepest gratitude and gratitude to Ph.D. Cao Thi Nhan – theoretical instructor of Information Systems Analysis and Design, and MSc. Duong Phi Long – practical instructor of Information Systems Analysis and Design. They have guided directly, corrected and contributed many valuable comments to help the authors successfully complete the course report.

During the project implementation period, the authors have applied the accumulated knowledge and combined it with learning and researching new knowledge. Since then, the authors make the most of what has been collected to complete a report on the best project. However, during the implementation process, the author group cannot avoid shortcomings. Therefore, the author group is looking forward to receiving suggestions from instructors to improve the knowledge that the authors have learned and is the luggage for the author group to carry out other topics in the future.

Sincerely thank you!

Ho Chi Minh City, 2023

Nguyen Khanh Van

Le Thi Kieu Lam

Duong Truc Uyen

Dang Quang Nhat

TABLE OF CONTENTS

Contents

CHAPTER 1:	5
INTRODUCTION	5
1.1. Introduction	5
1.2. Research problems	5
1.3. Research Objectives	6
1.4. Limitations of the study	6
CHAPTER 2:	7
REQUIREMENTS SPECIFICATIONS	7
2.1. Situation survey/ Survey of the current situation.....	7
2.1.1 Interview	7
2.1.2. Survey of the organization/ Current organizational situation.....	12
2.1.3. Survey of business processes, major activities.....	12
2.1.4. Survey of the existing system.....	14
2.2. Software requirements	15
2.2.1. Functional requirements	15
2.2.2. Non-functional requirements	16
CHAPTER 3:	17
ANALYSIS AND DESIGN	17
3.2.1.2 Manage cart	24
3.2.2 Use-case specification and activity diagram of Employee	34
CHAPTER 4:	122
IMPLEMENTATION AND TESTING	122
4.1 Introduction to the implementation environment.....	122
4.1.1 Development environment	122
4.1.2 Deployment environment	122
4.2 User Interface	123

4.2.1 Customer interface.....	123
4.2.1 Manager interface	125
CHAPTER 5:	129
CONCLUSION	129
5.1 Project achievement	129
5.2 Limitations	129
5.3 Development direction	130
SELF-ASSESSMENT	Error! Bookmark not defined.

CHAPTER 1:

INTRODUCTION

1.1. Introduction

In today's fast-paced world, convenience is key. With busy schedules and limited time, customers are always looking for ways to save time and streamline their daily routines. This is particularly true when it comes to getting their morning coffee fix. That's where our innovative ordering application for coffee shops comes in. Our app offers customers a quick and easy way to order and pay for their favorite coffee drinks from the comfort of their own devices. With just a few clicks, customers can customize their orders, skip the line, and have their drinks ready for pickup when they arrive at the coffee shop. Our app is designed to make the ordering process as seamless and efficient as possible, so that customers can get their caffeine fix and get on with their day without any hassle.

For coffee shop owners, the app provides a powerful tool for managing orders and streamlining operations. Orders are sent directly to the shop's point-of-sale system, which helps to reduce errors and speed up the fulfillment process. This means that orders can be prepared and ready for pickup by the time the customer arrives, minimizing wait times and improving overall customer satisfaction.

1.2. Research problems

Ingredient and price problems are common challenges faced by coffee shops. So that we wish to create an ordering application for coffee shops that can help to address ingredient and price problems in several ways.

1. By automating the ordering process: An ordering application can streamline the ordering process, eliminating the need for manual order-taking and reducing the risk of errors. This can help to ensure that orders are accurate and that ingredients are used in a timely manner.
2. By offering price transparency: An ordering application can provide customers with transparent pricing, allowing them to see the prices of different menu items and any promotions or discounts that may be available. This can help to build trust with customers and reduce confusion over pricing.

1.3. Research Objectives

In the process of surveying and researching the objectives for the coffee shop ordering application, we have identified some general and optimistic objectives below:

- Determine the feasibility of developing an ordering application for a coffee shop.
- Identify the key features and functionalities that customers would like to see.
- Evaluate the user experience of existing ordering applications for coffee shops and identify areas for improvement.
- Assess the impact of an ordering application on customer satisfaction, loyalty, and retention rates.
- Analyze the cost-benefit ratio of developing and implementing an ordering application for a coffee shop.
- Develop a business plan for the implementation of an ordering application for a coffee shop, including cost estimates, revenue projections, and marketing strategies.

1.4. Limitations of the study

Because of limitations about time, humans, our teams focus on some main features such as:

- Manage Customer
- Manage Employee
- Manage Customer's order cart
- Drinks order

CHAPTER 2:

REQUIREMENTS SPECIFICATIONS

2.1. Situation survey/ Survey of the current situation

2.1.1 Interview

General survey				
System: Highlands, Phuc Long, The Coffee House Creator: Le Thi Kieu Lam Created Date: 18/05/2023				
Ord	Topic	Requirement	Start	End
1	Cafe management	To get detailed information about beverage information storage, menu, distribution, bill valuation processes, cleaning's time, ..	18/05/2023	18/05/2023
2	News, promotions management	To get detailed information about proposed promotions, applying for promotions.	18/05/2023	18/05/2023

3	Customers management	To get detailed information about customers for better services and experience	18/05/2023	18/05/2023
4	Analysis	Process of making a report on beverage sales based on database.	18/05/2023	18/05/2023
5	Physical status	Physical equipment such as tables and chairs, dispensing equipment, cash registers,...	18/05/2023	18/05/2023

Table 1. Survey Strategy

Survey Forms	
System: IT's Coffee	
Interviewee: Nguyen Quynh Nhu	Interviewer: Le Thi Kieu Lam
Place: Ngon's Coffee	Start: 13h30 am 22/05/2023 End: 11:30 am 22/05/2023
Target: Collect and understand the working specifically process of the system in the café.	
Details of the interview: <ul style="list-style-type: none"> • Introduction • Overview about the current system • Survey plan: • Topic 1: Cafe management • Topic 2: News, promotions management 	Estimated Time: <ul style="list-style-type: none"> • 5 mins • 15 mins • 1 hours

<ul style="list-style-type: none"> • Topic 3: Customers management • Topic 4: Analysis and reports on sales • Topic 5: Physical Status 	
General Observe	
Unexpected occurrence	

Table 2. Survey form

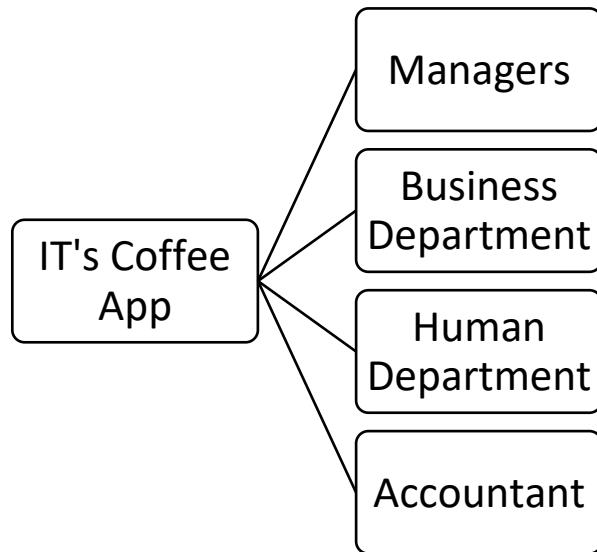
Interviewee: Nguyen Quynh Nhu	Date:
Questions	Records
Topic 1	In case the number of customers ordering drinks is too much, how will the cafe handle it?
	How is the cleaning time of the cafe divided?
	How does the price for best-seller drinks change?
	How to design a diverse and attractive menu?
Topic 2	Aside from cost discounts, are there any additional offers?

		loyal clients on key occasions to assist take better care of them.
	How should news be updated?	News must be updated regularly so that customers can easily be up to date.
Topic 3	How is the client's score computed, and how does the score improve as the client progresses through the categories?	According to the point categories, customers will be split into a variety of levels. The rise in points will be determined by the invoice that the consumer pays for the beverage purchase. Customers earn one point for every 1000 VND they spend.
	Will the information be preserved if a member client does not buy drinks in the cafe for an extended period?	Customers are the most important aspect in the cafe's success. As a result, whether consumers are frequently interested in cafe, maintaining client information is vital.
	Do various levels of customers have different priorities?	Different degrees and priorities will be assigned to member customers: <ul style="list-style-type: none"> • New users can only collect points and convert them into gifts from 0 to 1499. • Loyal customers receive a 5% discount on their bill between 1500 and 2999. • Over 3000: VIP members receive a 10% discount on their bill.
	How will the consumer's information be saved?	Customer information will be saved in the database. When buying drinks, staff will ask guests to provide phone numbers or customer codes to accumulate points and apply

		discounts according to customers' points.
Topic 4	How long does it take to figure out how many bills and drinks are sold on a regular basis?	The statistics of drink sales and bills will be collected on a weekly basis and for a term specified by the management if data reporting is required.
	What statistics does the system need?	<p>The system includes the following statistics:</p> <ul style="list-style-type: none"> • Statistics of bills. Statistics of beverages sold by each product. • Statistics of cafe revenues over a specified period of time.
Topic 5	Do different branches of the café have different forms of management (fares, promotions ...)?	Some cafes may share the same promotion campaign, but in detail, the promotions of cafes differ a bit. With an assumption that a cafe get low revenue in the previous quarter must get promotions frequently to pull customer back.
	How often is the inspection of the cafe's equipment (chairs, tables, dispensing equipment,...) carried out?	<p>The examination of chairs and tables is performed</p> <p>Once a month to guarantee that it is always in good operating condition.</p> <p>Employees will check dispensing equipment twice a week.</p>
	When the equipment fails, how will it be handled?	If the equipment fails, backup devices will be used to replace and fixed as soon as possible.

Table 3. Question & records

2.1.2. Survey of the organization/ Current organizational situation



2.1.3. Survey of business processes, major activities

2.1.3.1 Business Department

The **beverage menu management business** is the set of activities and responsibilities involved in managing, developing, and maintaining the beverage menu in a coffee shop. It includes identifying beverages, classifying and characterizing them, researching and developing new beverages, managing ingredients, pricing and profitability, training staff in beverage processing, and evaluating prices and menu improvements.

Market research and analysis:

Perform research and analysis to determine what drinks should be on the menu, create combo packages or attractive offers adjust menus to meet customer needs.

Formulate menus and categorize drinks:

- Determine what kind of drink the cafe will offer(ingredients, flavors, sizes for each drink).
- Classified into different categories.
- Store in the database.

Business efficiency analysis:

Monitor and analyze the business performance of drinks on the menu.

By collecting data on sales, profits, and feedback from customers, evaluate the performance of products and recommend necessary adjustments or improvements.

Pricing and Promotion Strategy:

- Set prices and suggest selling prices for each item on the menu.
- Propose suitable pricing strategies to ensure profits and attract customers.
- Develop and manage promotions, offers or combo packages to increase sales.

Marketing business: propose new strategies to promote the buying or selling of a product, service, or good. Including promotions, customer services and search engine optimization.

Promotion process:

- Propose various strategies to attract customers.
- Inform to the technical department for implementation.

Customer services process: not only create a close connection between customer and system, but also provide information to customers in time.

- Manage customers' information.
- Provide any questions to customers as soon as possible.
- Execute strategies from the Marketing department.

Sales operations: serve customers who come to buy drinks.

Beverage sales process:

- Customers go to the counter to buy drinks according to the set menu
- available. The staff takes drinks, charges and bills customers.

2.1.3.2 Human Department

Human resource management business: monitor the activities of all employees in the cafe, including employee information management process, employee activity management process and other processes.

Employee information management process:

- Recruitment.
- Enter employee information.
- Update employee information.

Employee activity management process:

- Employee working hours and activity history.
- Periodic reports on employee activities.

2.1.3.3 Accountant

Analysis business: is the implementation of data-driven reports to assist the Sales Department in monitoring the performance system. Including the process of calculating wages and paying employees' salaries, and reports.

Salaries process:

- Collect data about working hours at Human Department.
- Calculate salary and bonus for each employee at end of the month.
- Store data into database.

Reporting process:

- Get data from databases.
- Create report mainly on total number of tickets, promotions, customer traffic.
- Evaluate the report and transfer to manager.

2.1.4. Survey of the existing system

2.1.4.1. Business Department

- Coffee shop management business: This department is responsible for importing information and pricing.
 - Importing information process: The importing information process is about entering information about products, prices, stock availability, and other relevant data into the coffee shop management system. This information can be entered manually or imported from a spreadsheet or other data source.
 - Pricing process: This process typically involves considering factors such as the cost of goods sold, the competition, and the desired profit margin.
- Marketing business: This department is responsible for promoting the coffee shop and providing customer service.
 - Promotion process: The promotion process involves developing and executing marketing campaigns. This can include advertising, public relations, and social media marketing.

-
- Customer services process: The customer services process involves helping customers with their orders, complaints, and other issues.

2.1.4.2. Human Department

- Employee information management process: Collecting and storing information about employees. This information includes the employee's name, address, phone number, job title, and salary.
- Employee activity management process: This process involves tracking the activities of employees. This can include tracking their work hours, their productivity, and their performance.

2.1.4.3. Technology Department

- System management process: Managing the coffee shop's computer systems. This includes tasks such as installing and updating software, configuring hardware, and troubleshooting problems.
- System development process: Developing better software for the coffee shop. Adding new features to the existing system or developing a completely new system.

2.1.4.4. Accountant

- Salaries process: Calculating and paying employees' salaries and customers' bonuses. This includes tasks such as tracking employee hours, calculating overtime pay then storing data into the database.
- Reporting process: This process involves generating reports on the coffee shop's financial performance. Evaluate the results and transfer to the board of management.

2.2. Software requirements

2.2.1. Functional requirements

2.2.1.1 Storage function

The storage feature is one of the most important features of the system, which is the basis for implementing system's features. To ensure effective management, the system needs to store the following information:

- Supplier information: name, address, phone, email, website, supplier's representative, payment methods, product supplied
- Menu information: beverage's name, cost, beverage's type, size, special information(sugar, ice)

-
- User information: username, gender, date of birth, address, phone, email, id, accumulated points, password, user type.
 - Promotion information: promotion name, image, content, start date, end date.

2.2.1.2 Feature function

In addition, the system must also meet the following requirements:

- Query function: able to respond query requests from customers/manager such as: Find menu's information, find customer information, promotion information, beverage information, news information...
- Statistical, analysis, reporting functions: help management staff monitor the general situation of the cafe.
- Update function: the system must ensure that the update function is convenient and avoid errors due to the user's negligence.
- Automatic functions: help users more convenient in operation such as automatic bill , automatic generation of bill id ...

2.2.2. Non-functional requirements

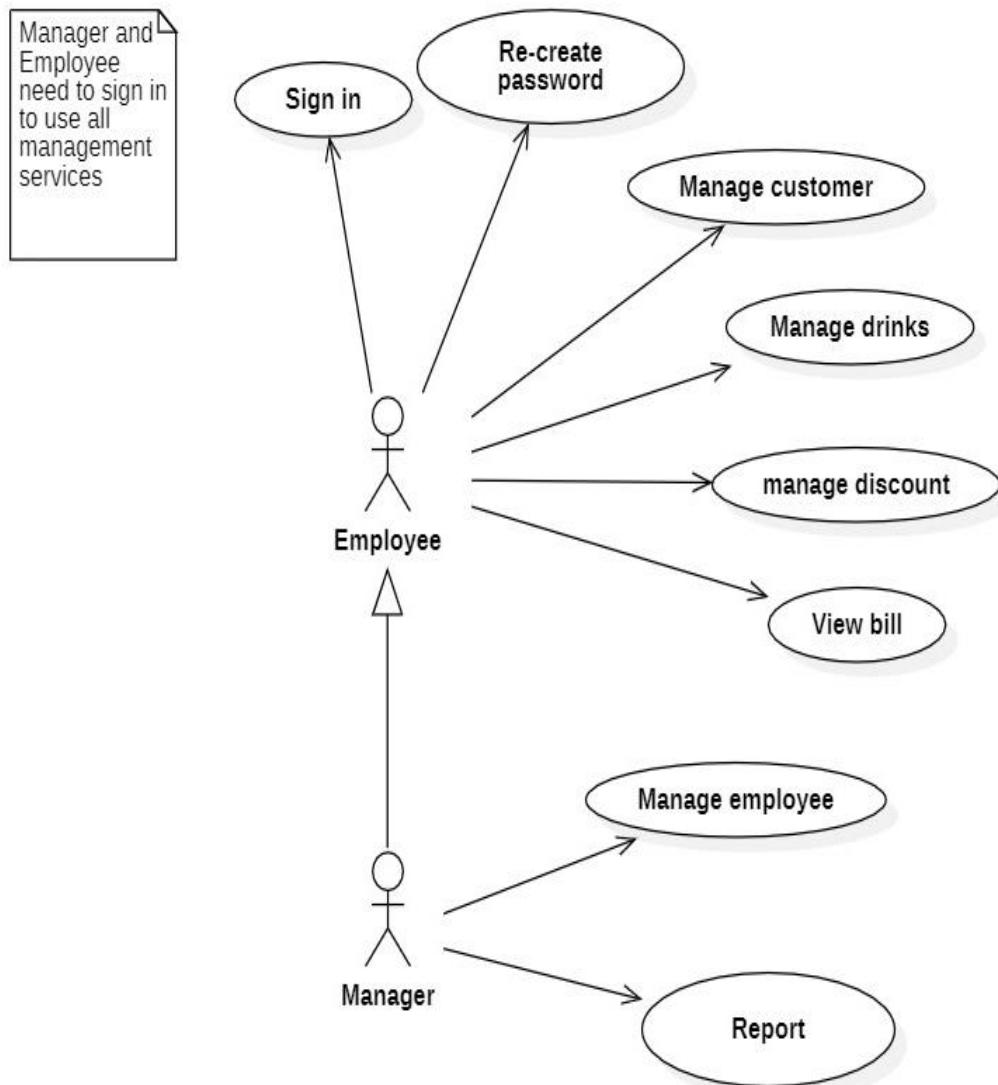
- Customer GUI requirements: multilingual, friendly interface, highly interactive.
- The management GUI: simple, close as report as possible, easy to query.
- Quality requirements:
 - + Evolution:
 - Easy to upgrade or increase utility modules.
 - + Convenience:
 - The system has an intuitive, user-friendly and easy-to-use interface.
 - Clear specifications and instructions for use.
 - + Efficiency:
 - The app shows the information about the beverage well.
 - The app works stably and reliably.
 - + Compatibility:
 - Widely compatible with different operating systems and computer configurations
 - + Security:
 - Applications should have security measures in place to protect users' personal data and important information.

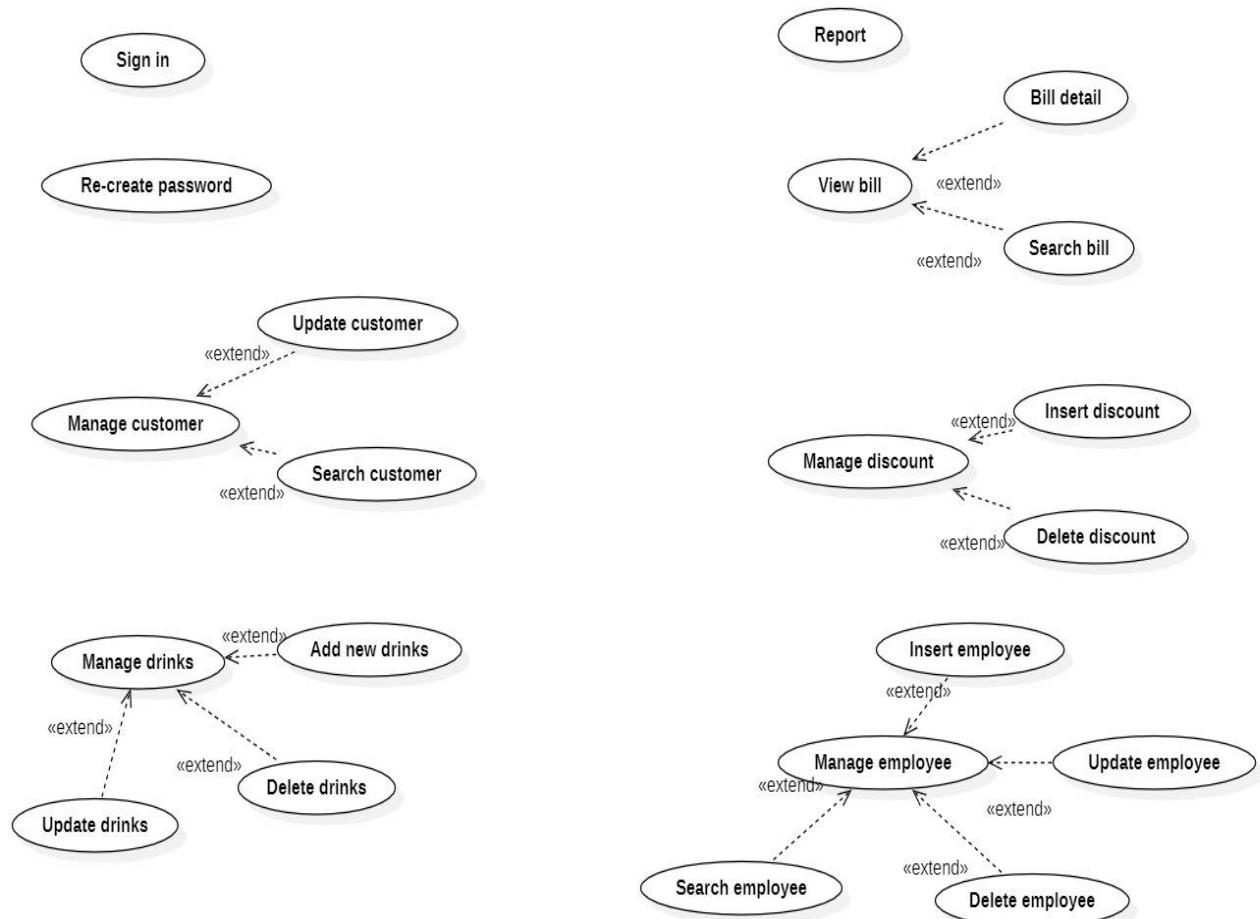
CHAPTER 3:

ANALYSIS AND DESIGN

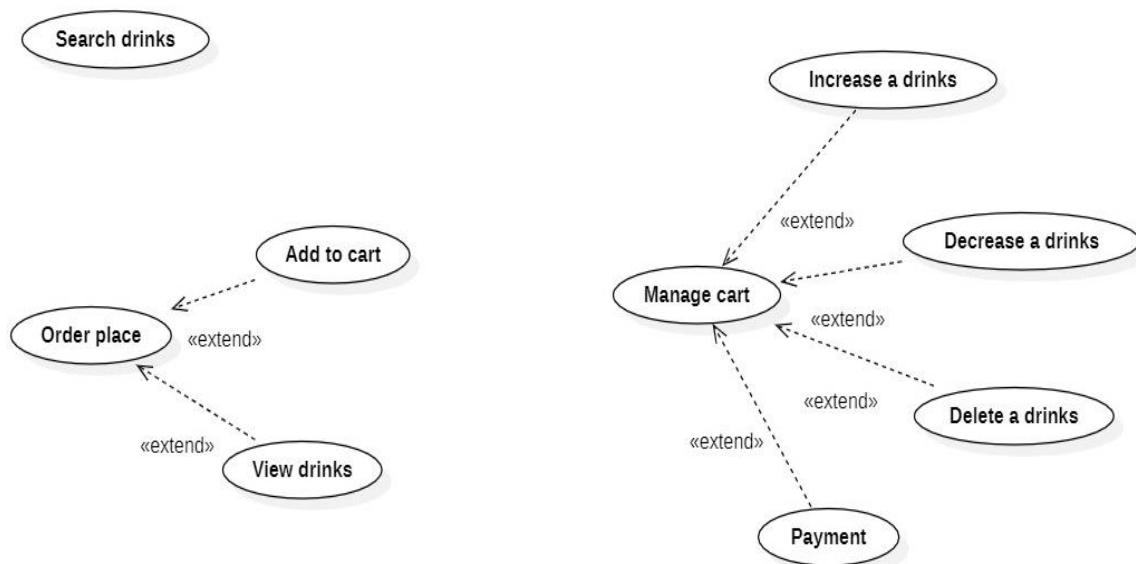
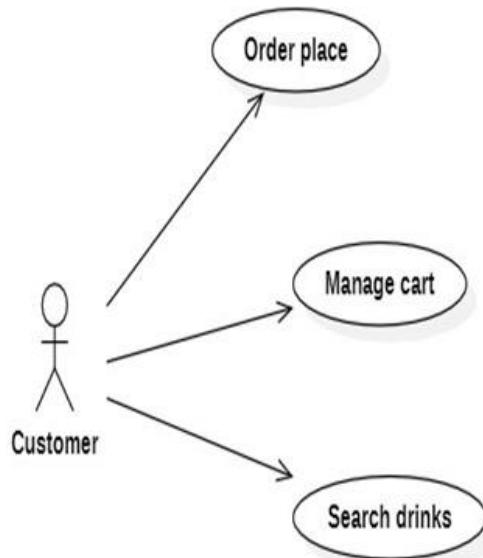
3.1 Use-case Diagram

3.1.1 Use-case Diagram of Manager and employee





3.1.2. Use-case Diagram of Customer



3.1.3 List of actors

Index	Actor	Meaning
1	Manager	People manage all system's services
2	Employee	People manage partial system's services
3	Customer	People can use all system's services

3.1.4 List of use-case

Role		Use Case	Meaning
Customer's role		Order place	See information about drinks or add drinks to the cart.
		Manage cart	Increase/Decrease/Delete a drinks/Payment.
		Search drinks	Search drinks information related keywords.
Manager's role	Employee's role	Sign in	Sign in to play the role
		Re-create password	Re-create the password if forget it.
		Manage drinks	Insert/Update/Delete/Search drinks.
		Manage customer	Update/Search customer
		Manage discount	Insert/Delete discount.
		Manage employee	Insert/Update/Delete/Search employee.
		Report	Reporting revenue

3.2 Use-case specification and activity diagram

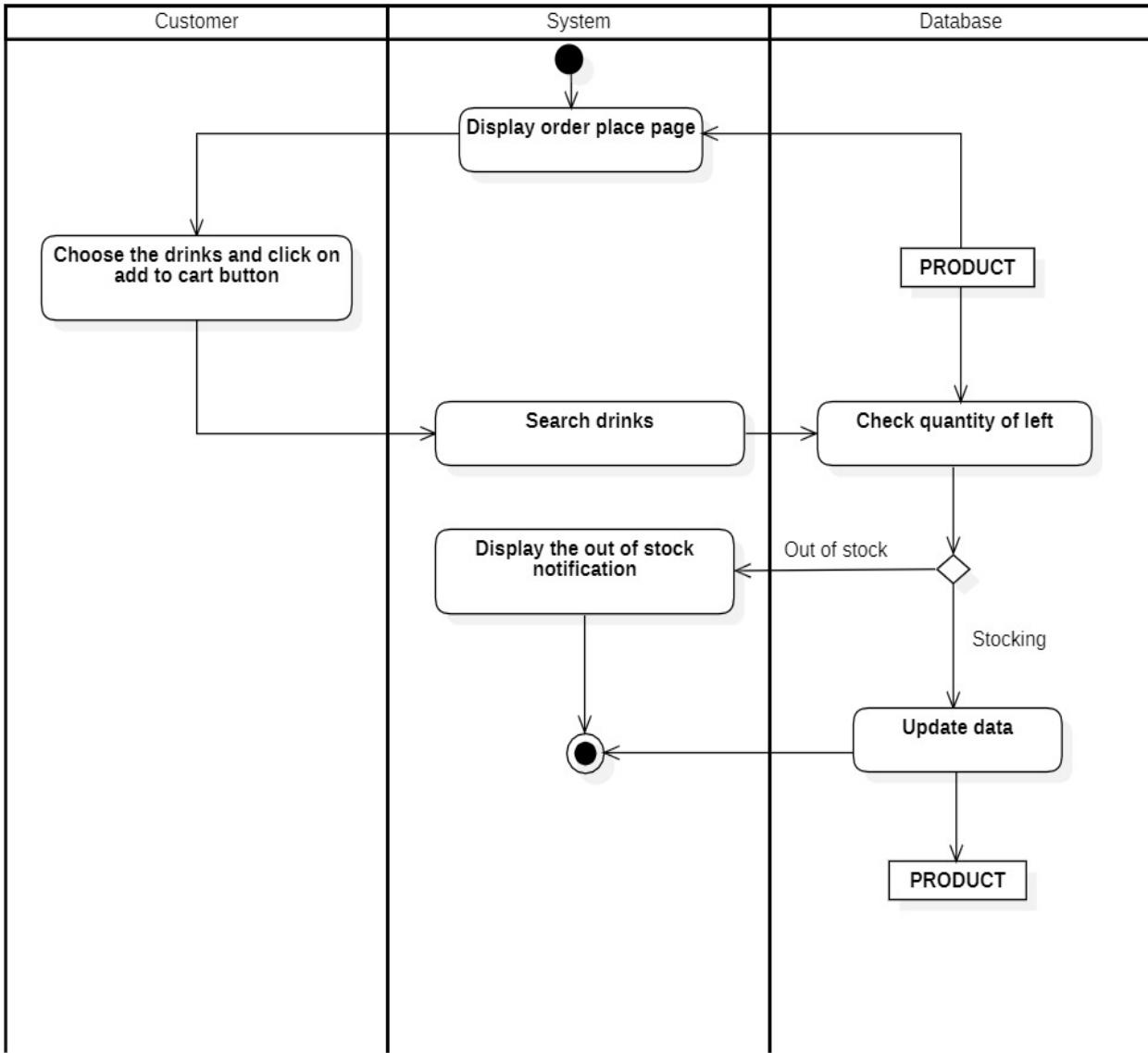
3.2.1 Use-case Specification and activity diagram of Customer

3.2.1.1 Order place

3.2.1.1a Add to cart

Use-case name	Add to cart
Description	Customers can add drinks to the cart

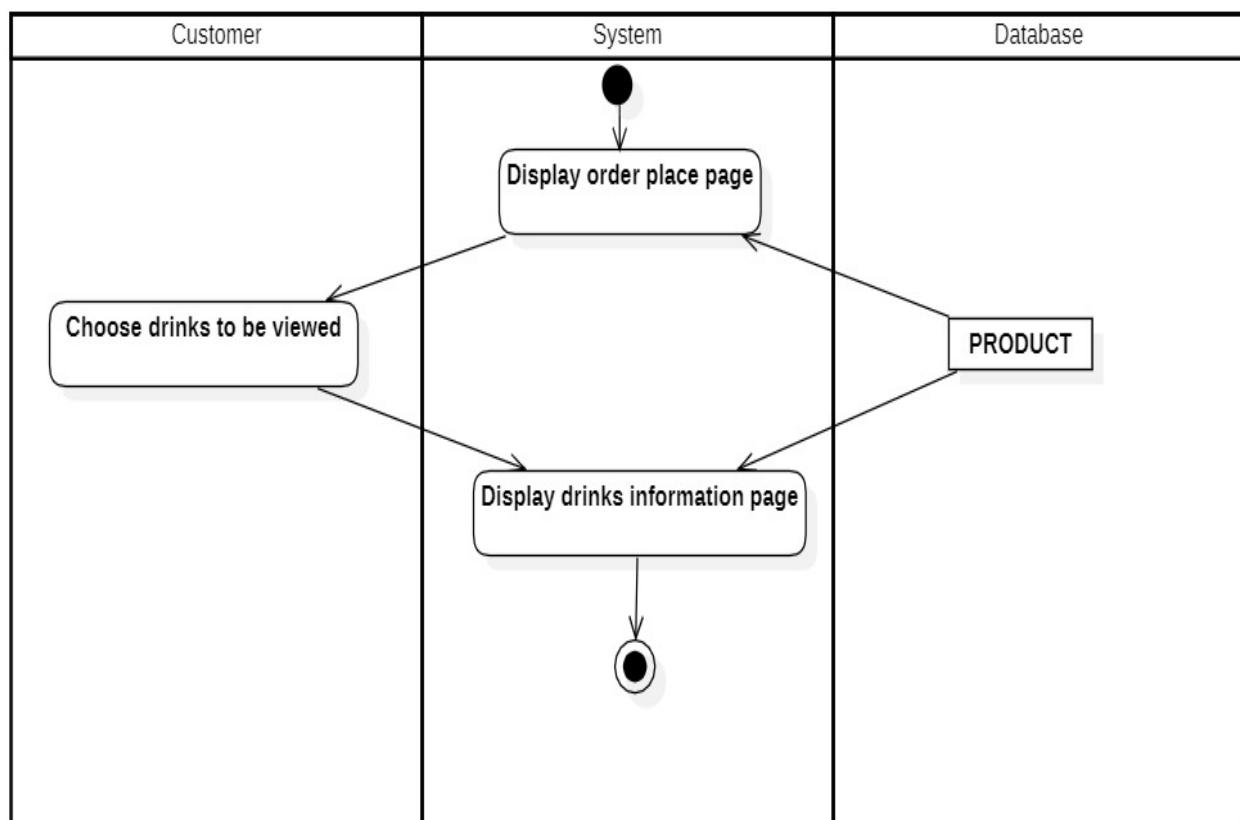
Trigger	Customers click to button “Add to cart” on the order place page
Pre-condition	Customers’ devices must connect to the Internet. Customer’s status is signed-in.
Post-condition	Customer cart information is updated in the database.
Basic flow	<ol style="list-style-type: none"> 1. System displays order place page. 2. Customer chooses drinks to be added to the cart and clicks on the “Add to cart” button. 3. System search drinks in the database. 4. System checks the quantity of drinks left in the database and stocking. 5. System update the quantity in database.
Alternative flow	
Exception flow	4a. System displays the out of stock notification. Use-case stop.



3.2.1.1b View drinks

Use-case name	View Drinks
Description	Customers can see information about drinks
Trigger	Customers click on drinks to be viewed.
Pre-condition	Customers' devices must connect to the Internet.

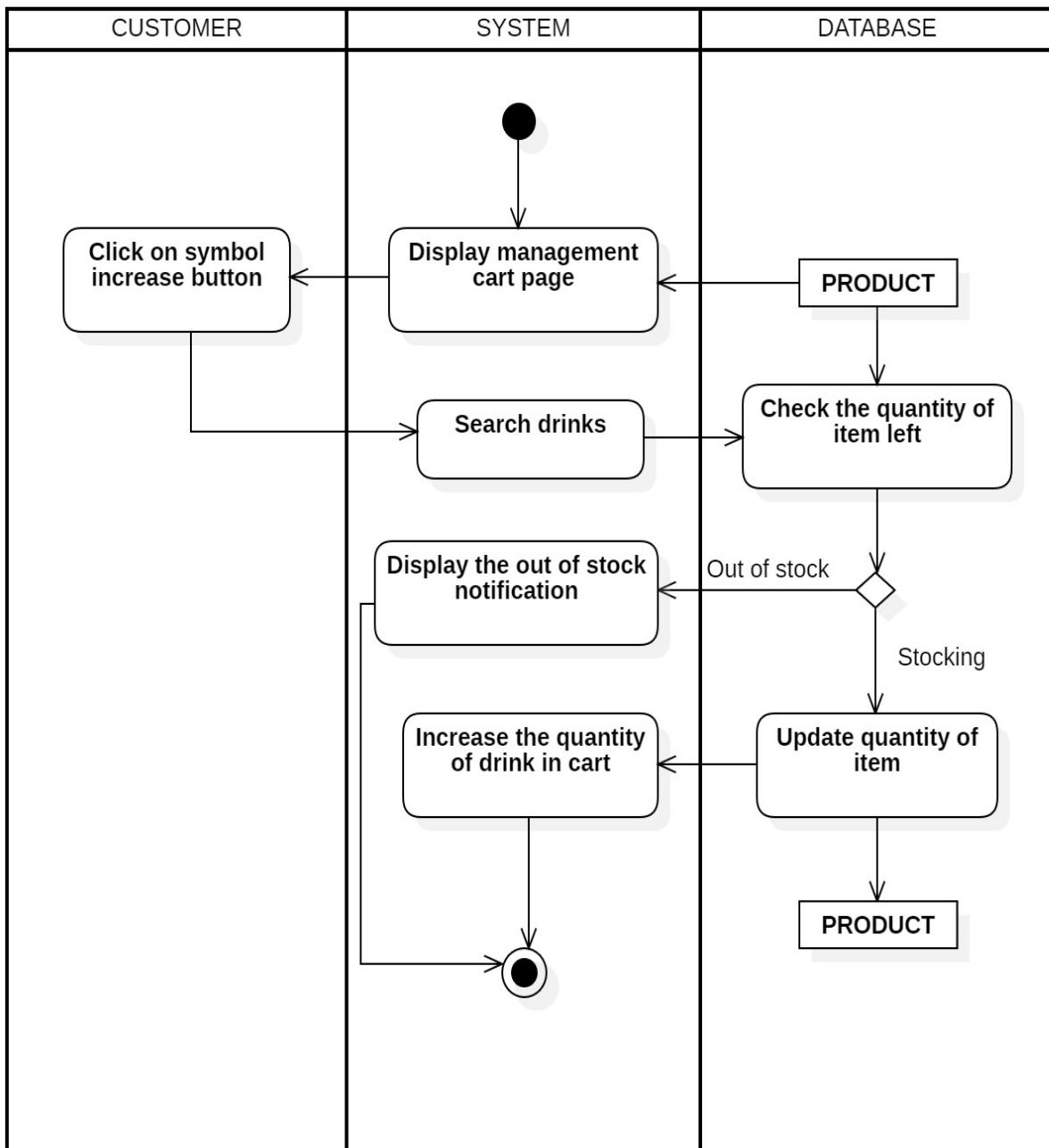
Post-condition	Customers get drinks with detailed information.
Basic flow	<ol style="list-style-type: none"> 1. System displays the order place page with data. 2. Customer chooses drinks to be viewed. 3. System displays drinks information page with data in the database.
Alternative flow	
Exception flow	



3.2.1.2 Manage cart

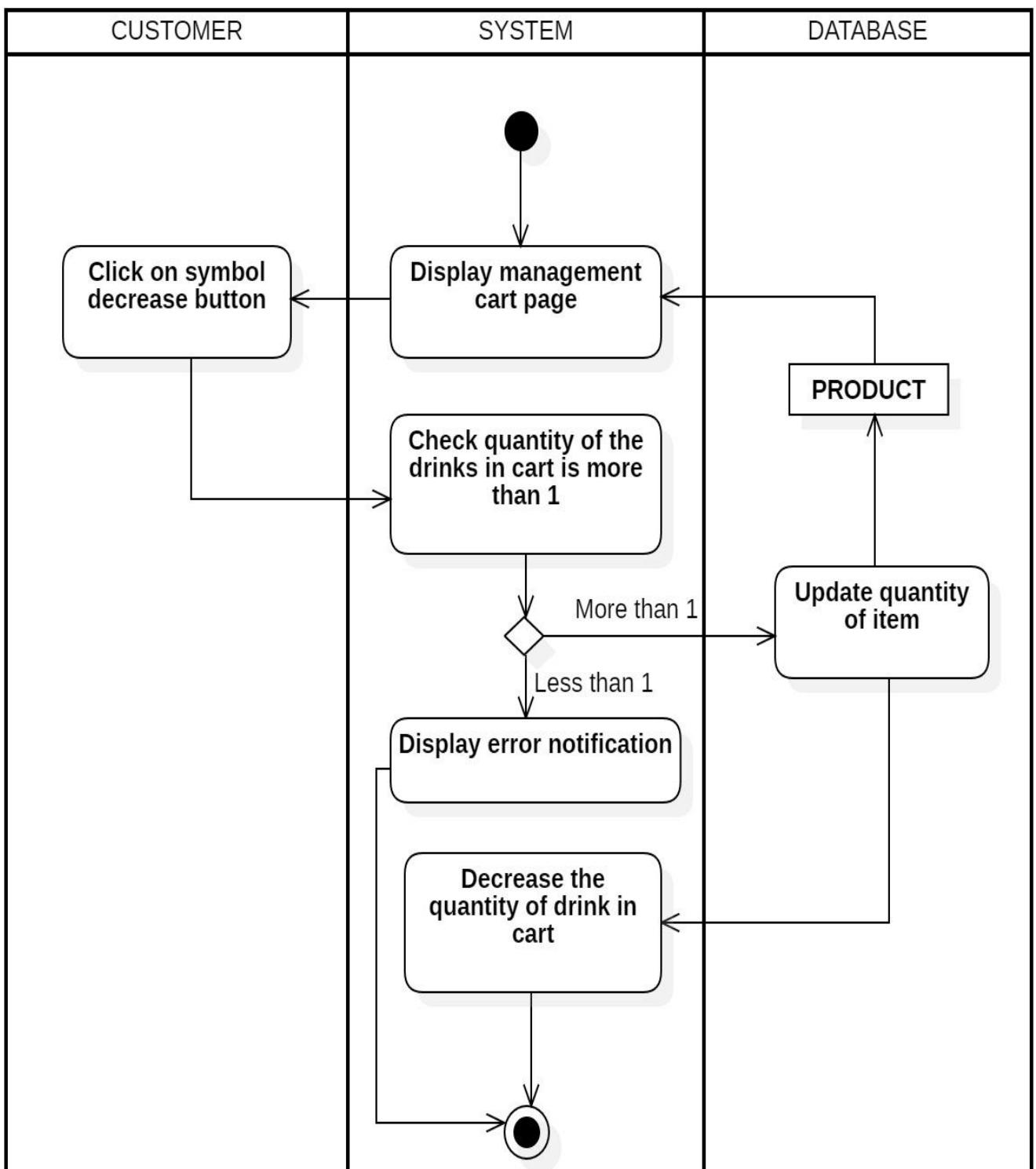
3.2.1.2a Increase a drinks

Use-case name	Increase a drinks
Description	Customer can increase a product in the cart.
Trigger	Customer choose a drinks and click to the symbol “^” to increase.
Pre-condition	Customer's devices must connect to Internet and access to the website. There is at least one product in the cart.
Post-condition	The products which customer chooses are increase.
Basic flow	<ol style="list-style-type: none">1. Customer choose a drinks and click to the symbol “^”.2. System query and check the quantity of item left.3. System update data in database.4. System increase a drink in the cart.
Alternative flow	
Exception flow	2a. System display out of stock notification. Use-case stop.



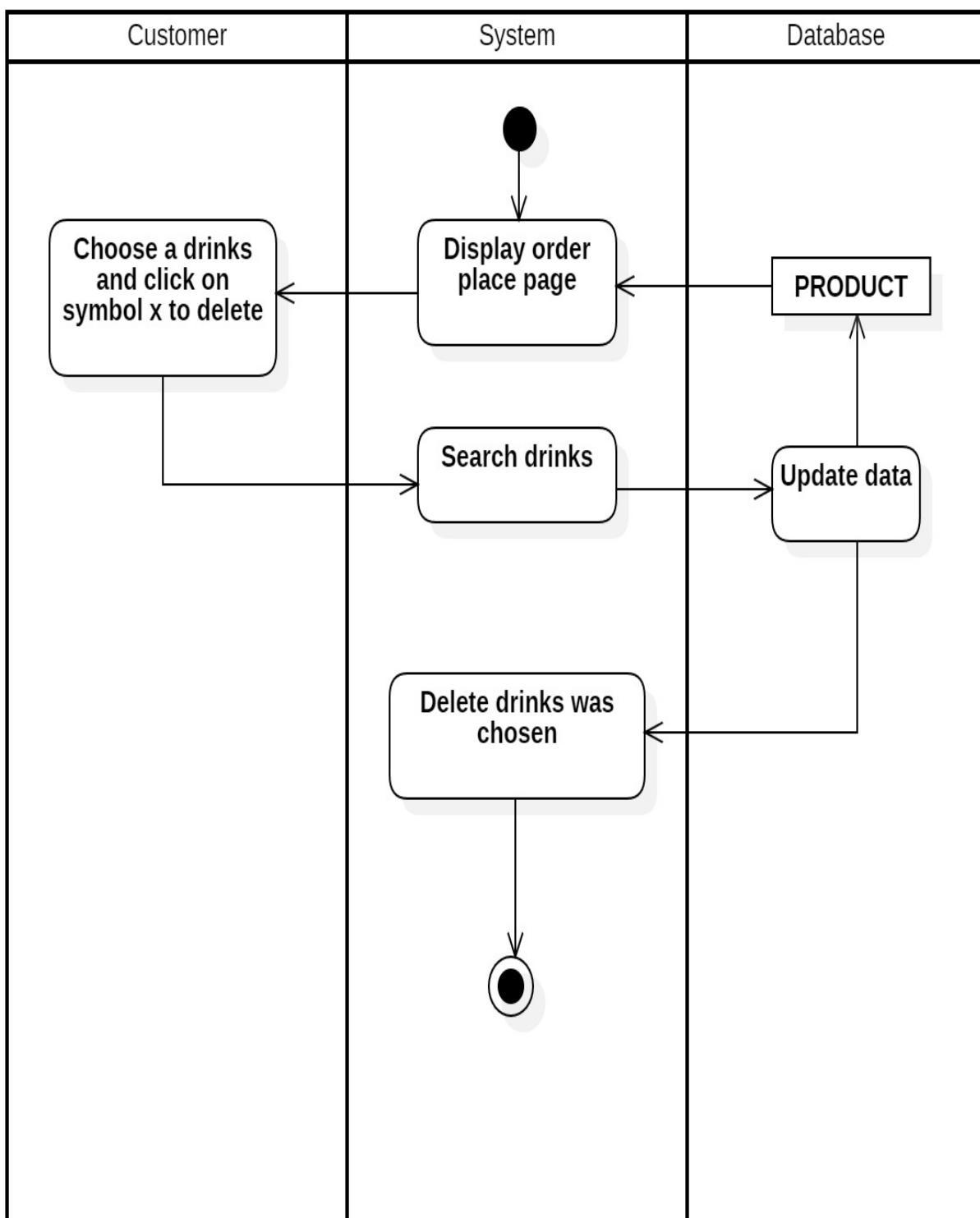
3.2.1.2b Decrease a drinks

Use-case name	Decrease a drinks
Description	Customer can decrease a product in the cart.
Trigger	Customer choose a drinks and click to the symbol “v” to decrease.
Pre-condition	Customer’s devices must connect to Internet and access to the website. There is at least one product in the cart.
Post-condition	The products which customer chooses are decrease.
Basic flow	<ol style="list-style-type: none">1. Customer choose a drinks and click to the symbol “v”.2. Check quantity of the drinks in cart is more than 1.3. System update data in database.4. System decrease a drink in the cart.
Alternative flow	
Exception flow	2a. System display error notification. Use-case stop.



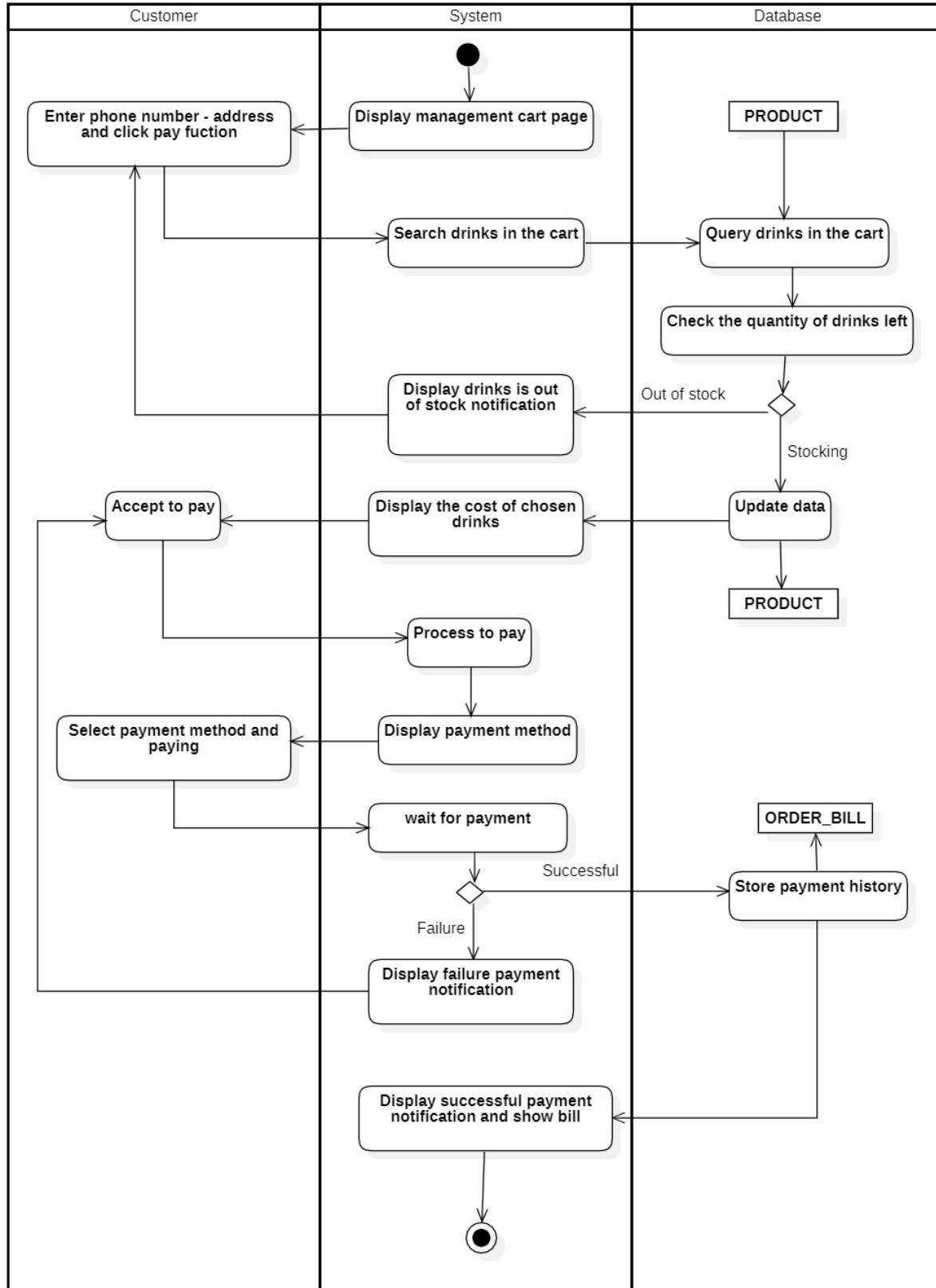
3.2.1.2c Delete a drinks

Use-case name	Delete a drinks
Description	Customer can delete the product in the cart.
Trigger	Customer choose a drinks and click to the symbol “x” to delete.
Pre-condition	Customer’s devices must connect to Internet and access to the website. There is at least one product in the cart.
Post-condition	The products which customer chooses are deleted in the cart
Basic flow	<ol style="list-style-type: none">1. Customer choose a drinks and click to the symbol “x”.2. System update data in database.3. System reload the cart and remove the deleted product.
Alternative flow	
Exception flow	



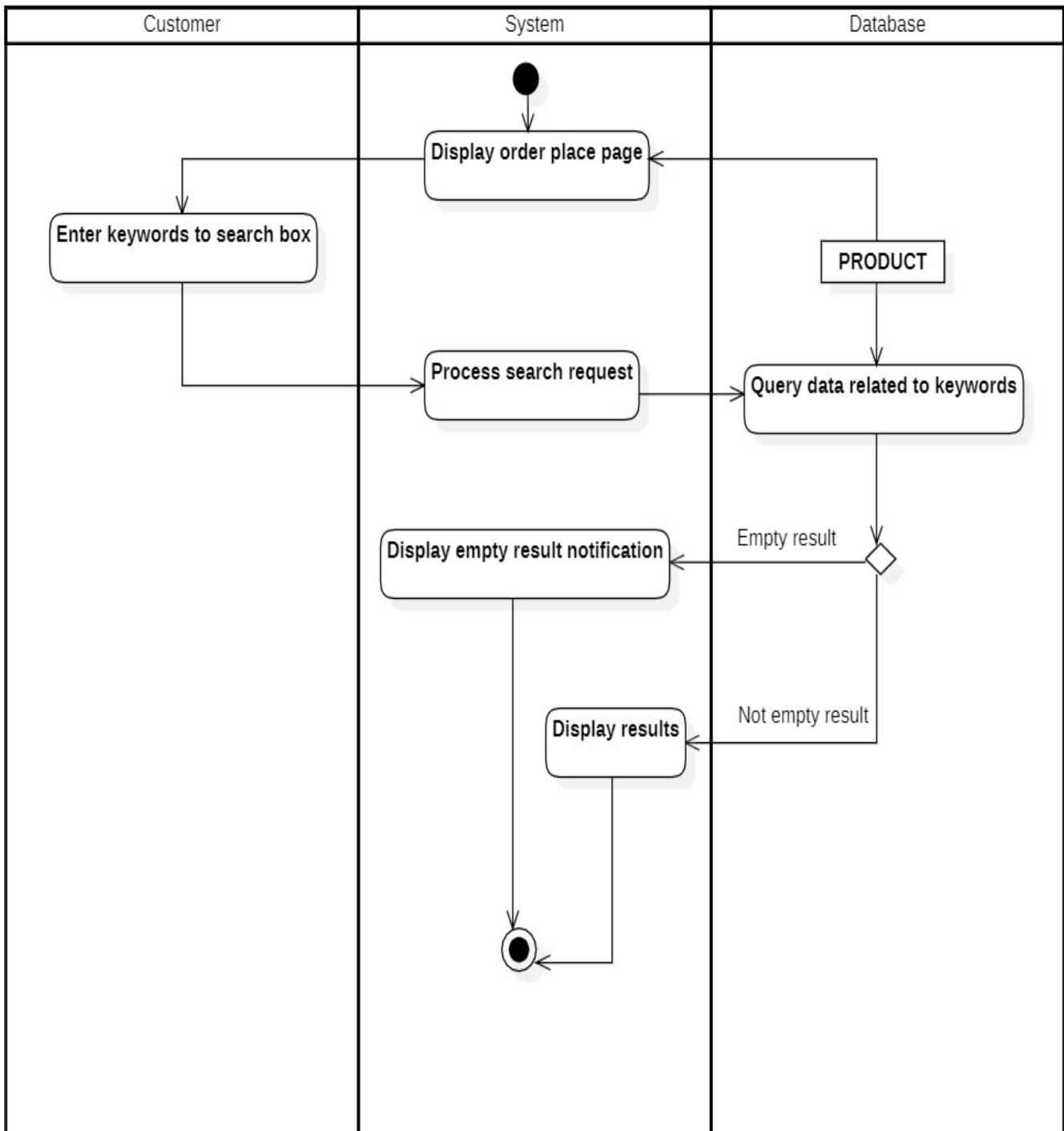
3.2.1.2d Payment

Use-case name	Payment
Description	Customers pay for Coffee bill.
Trigger	Customers click on “Pay” button.
Pre-condition	Customer’s devices must connect to Internet. Customer’s status is signed-in.
Post-condition	Customers pay successfully for Coffee bill. Customer’s payment informations is stored into database.
Basic flow	<ol style="list-style-type: none">1. System display management cart page.2. Customers enter phone number, address and click pay function3. System search list of drinks to query in database4. System check the quantity left.4. System update the quantity of drinks into database.5. System display cost of customer’s drinks chosen.6. Customer accept to pay (by click on “Continue” button).7. System processes to pay.8. System display available payment methods.9. Customer select suitable payment method and paying (by click on “Pay” button).10. System wait for payment process.11. System stores payment information into database.12. System display successful payment notification and show invoice information.
Alternative flow	<ol style="list-style-type: none">3a. System display the drink is out of stock notification. Use-case continue at step 2.10a. System display failure payment notification. Use-case continue at step 6.
Exception flow	



3.2.1.3 Search drinks

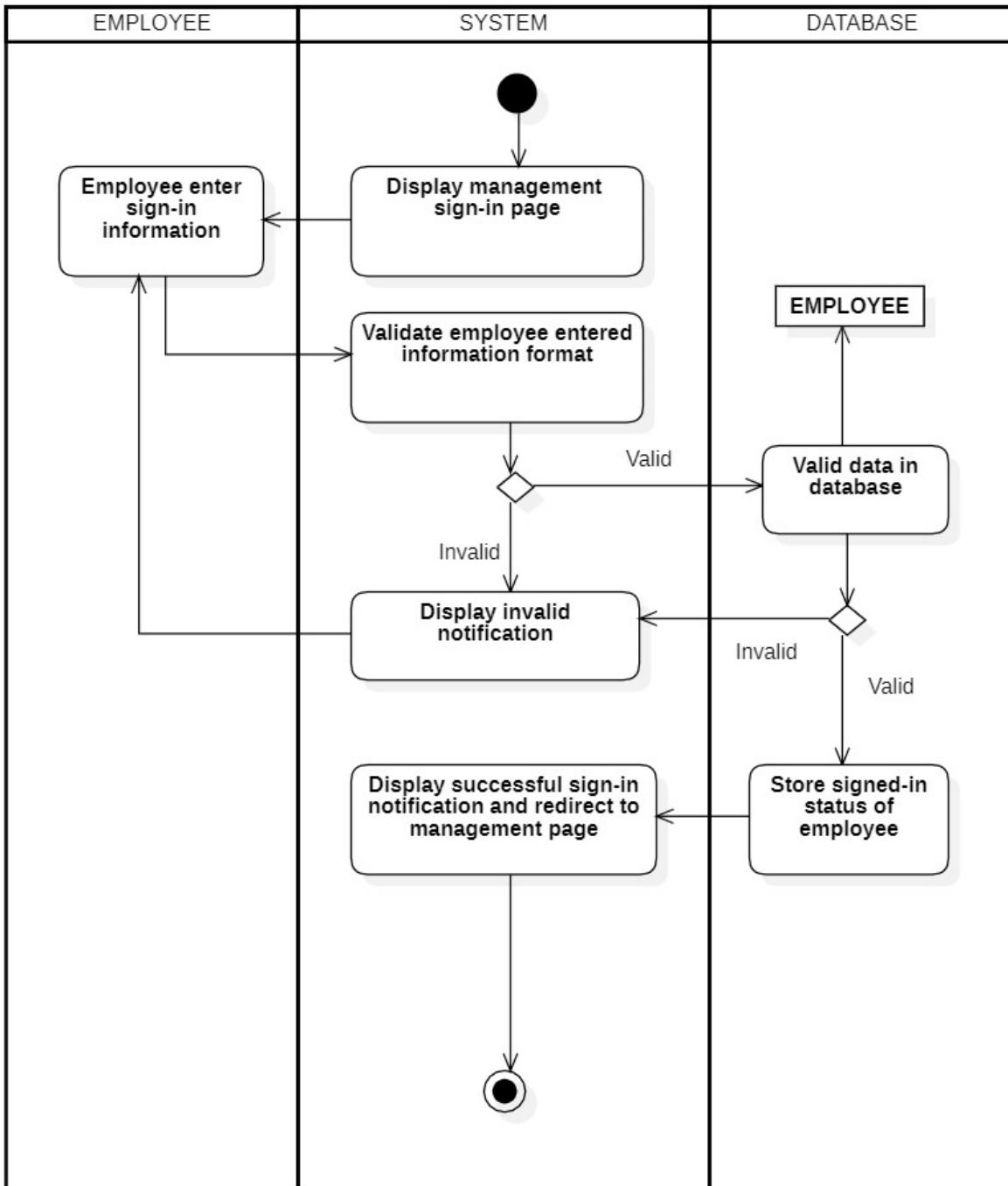
Use-case name	Search drinks
Description	Customers search drinks including some words
Trigger	Customers click to button “Search” button on the order place page.
Pre-condition	Customer’s devices must connect to the Internet.
Post-condition	Customer gets a list of drinks with detailed information
Basic flow	<ol style="list-style-type: none">1. System displays the order place page with data.2. Customers enter keywords on the text field and then click the “Search” button3. System processes searching data.4. System queries data based on keywords and not empty results.5. System displays the result.
Alternative flow	
Exception flow	4a. System displays empty result notifications. Use-case stop.



3.2.2 Use-case specification and activity diagram of Employee

3.2.2.1 Sign-in

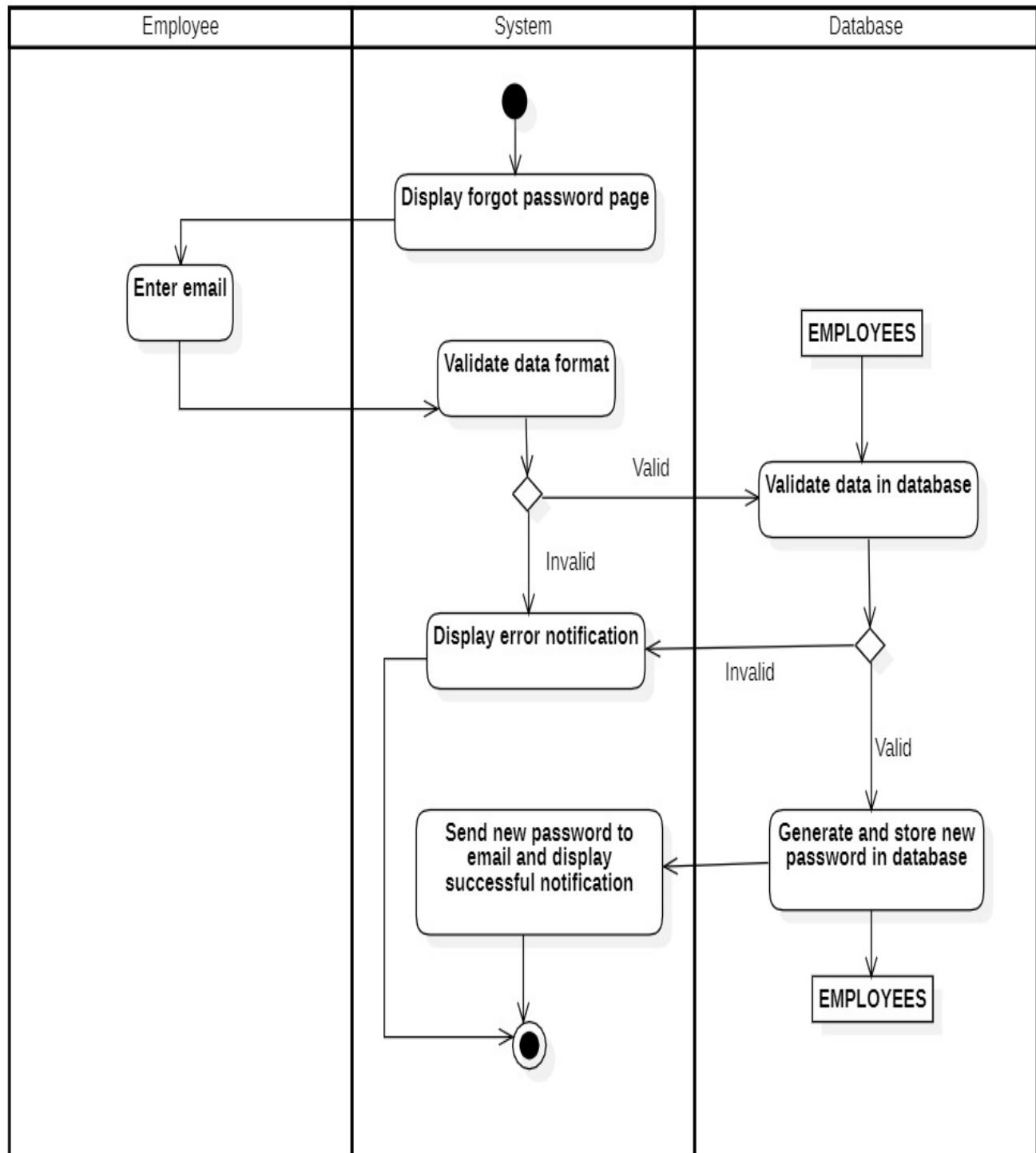
Use-case name	Sign in
Description	Employee use existing management account to sign-in to system for all monitoring.
Trigger	Employees clicks to button “Sign in” at management page
Pre-condition	Employees’ device must connect to Internet Employees’ account existed
Post-condition	Employees get a notification about signed-in successful. Employees is redirected to management page
Basic flow	<ol style="list-style-type: none">1. Employees clicks “Sign in” button at management page2. System display management sign-in page3. Employees enter email & password4. System validate Employees entered email & password in correct format5. System validates Employees’ data with existed Employee account in database.6. System store signed-in status of Employees using sessions7. System displays successful signed-in and redirect to management page.
Alternative flow	<p>3a. System display invalid information due to wrong format. Use-case continue at step 2.</p> <p>4a. System display invalid information due to Employees’ account information does not match with existed account information in database.</p> <p>Use-case continue at step 2.</p>
Exception flow	



3.2.2.2 Re-create password

Use-case name	Re-create password
Description	Employees forgot account password so cannot sign-in to system for all services.
Trigger	Employees click to button “Forgot password”

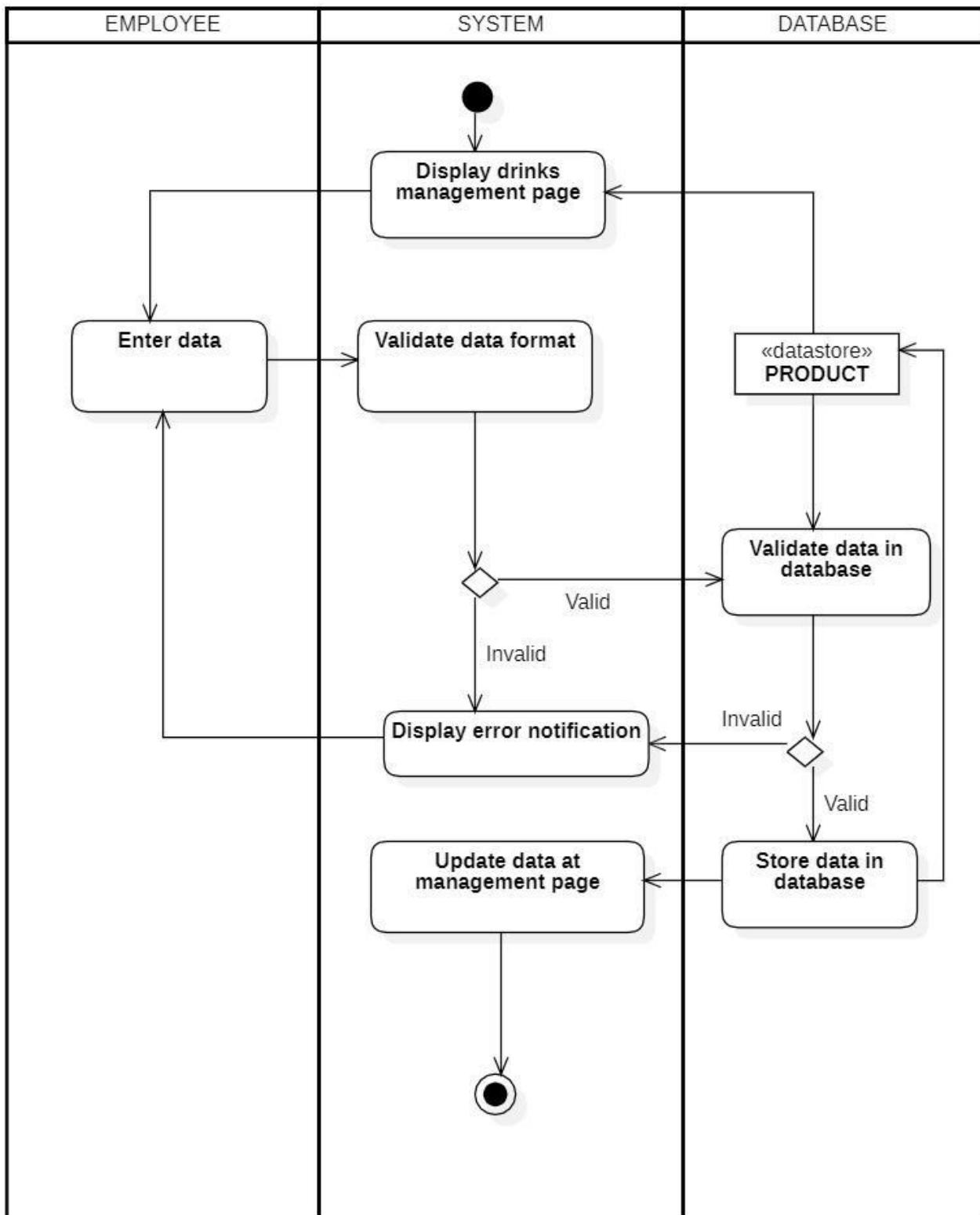
Pre-condition	employee's devices must connect to Internet Employee's accounts existed Employees forgot password and cannot sign-in Employees must remember sign-up email
Post-condition	Employees get an email containing new password notification about sign in successful. Employee's password will be update in the database
Basic flow	<ol style="list-style-type: none"> 1. System display forgot password page 2. Employees enter email 3. System validates employee's email in correct format 4. System validates email of employee with existed in database. 5. System generates new password and update data in database. 6. System send email to customers containing new generated password and display successful message.
Alternative flow	
Exception flow	<p>3a. System display invalid information due to wrong format Use-case stop.</p> <p>4a. System display invalid information due to no existed account been signed with customers' entered email. Use-case stop.</p>



3.2.2.3 Manage drinks

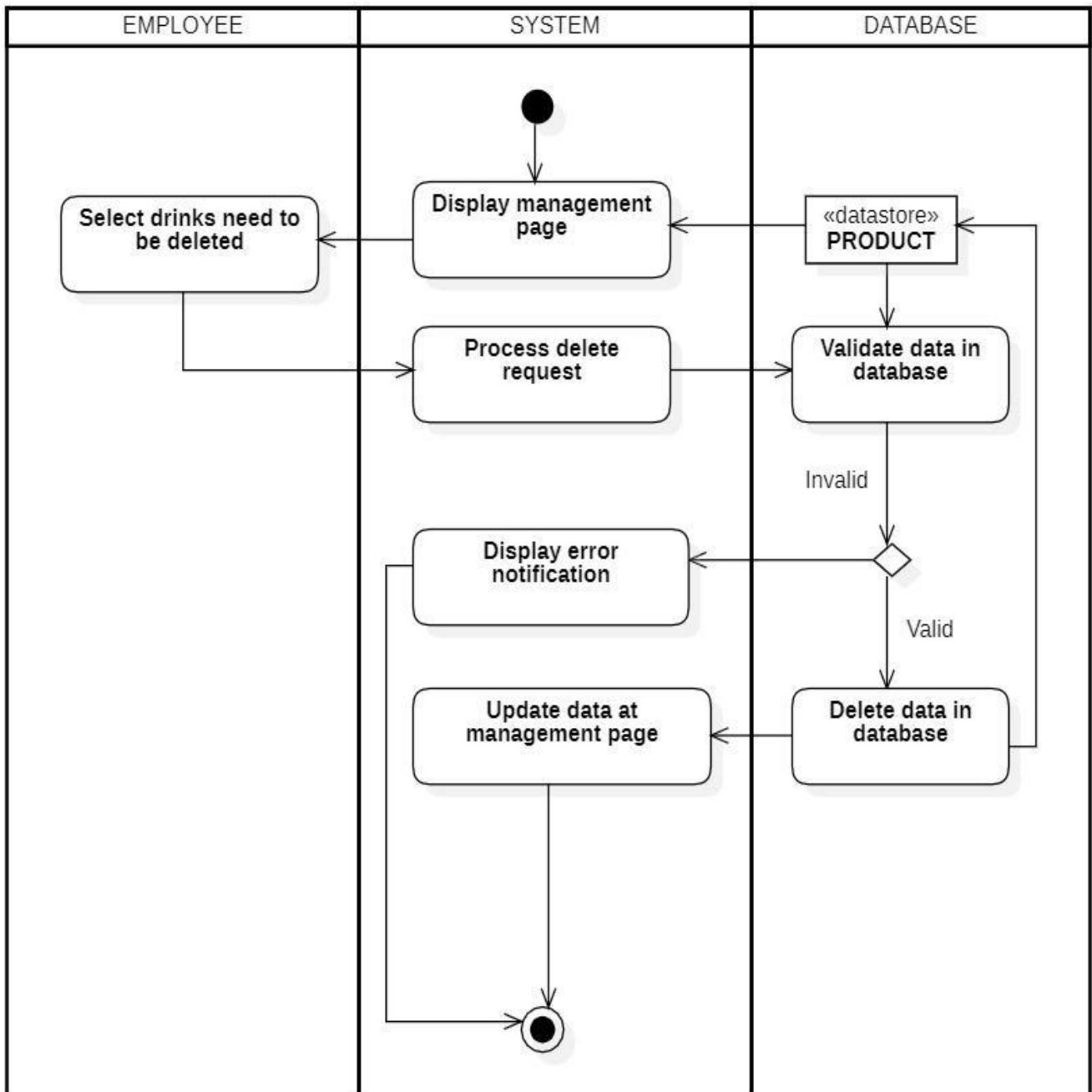
3.2.2.3a Add new drinks

Use-case name	Add new drinks
Description	Employees can add new drinks to the database
Trigger	Employees click to button “Add new” at the drinks management page
Pre-condition	Employees’ devices must connect to Internet Employees’ accounts existed Employee’s status is signed-in.
Post-condition	Drinks information successful insert to the database Data at the management page is updated (with new drinks)
Basic flow	<ol style="list-style-type: none">1. System display drinks management page with data loaded from the database (existing drinks)2. Employees select insert function3. Employees enter drinks information4. System validates drinks information in the correct format5. System validates drinks information with existed data in the database.6. System stores drinks information in the database7. System updates data at the drinks management page with inserted data (new data).
Alternative flow	<p>4a. System display invalid information due to wrong format. Use-case continue at step 3.</p> <p>5b. System display invalid information because data had been existed in the database. Use-case continue at step 3.</p>
Exception flow	



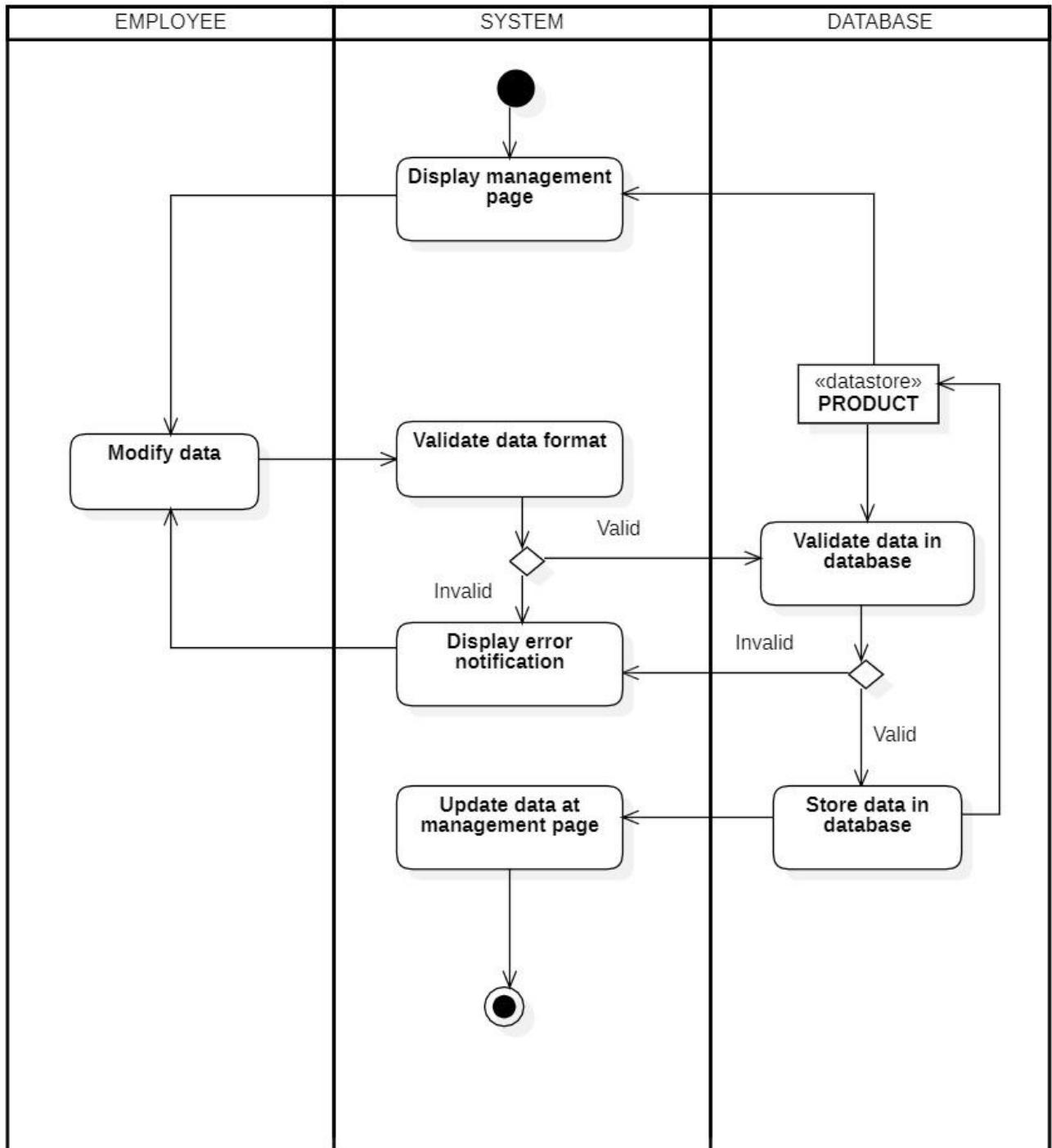
3.2.2.3b Delete drinks

Use-case name	Delete drinks
Description	Employee can delete an existed drinks in the database
Trigger	Employees clicks to button “Delete” at drinks management page
Pre-condition	Employees’ device must connect to Internet Employees’ account existed Employee’s status is signed-in. Drinks information must be existed in database before deleting No external data involved.
Post-condition	Data at management page is updated (selected drinks had been deleted)
Basic flow	<ol style="list-style-type: none">1. System display drinks management page with data loaded from database (existed drinks)2. Employees select drink to be deleted3. System processes delete request4. System validates no external data required drinks information and selected item existed in database5. System deletes drinks information in the database6. System updates data at drinks management page with selected item deleted.
Alternative flow	
Exception flow	5a. System display error notification because external data had required that drinks information or no data existed Use-case stop.



3.2.2.3c Update drinks

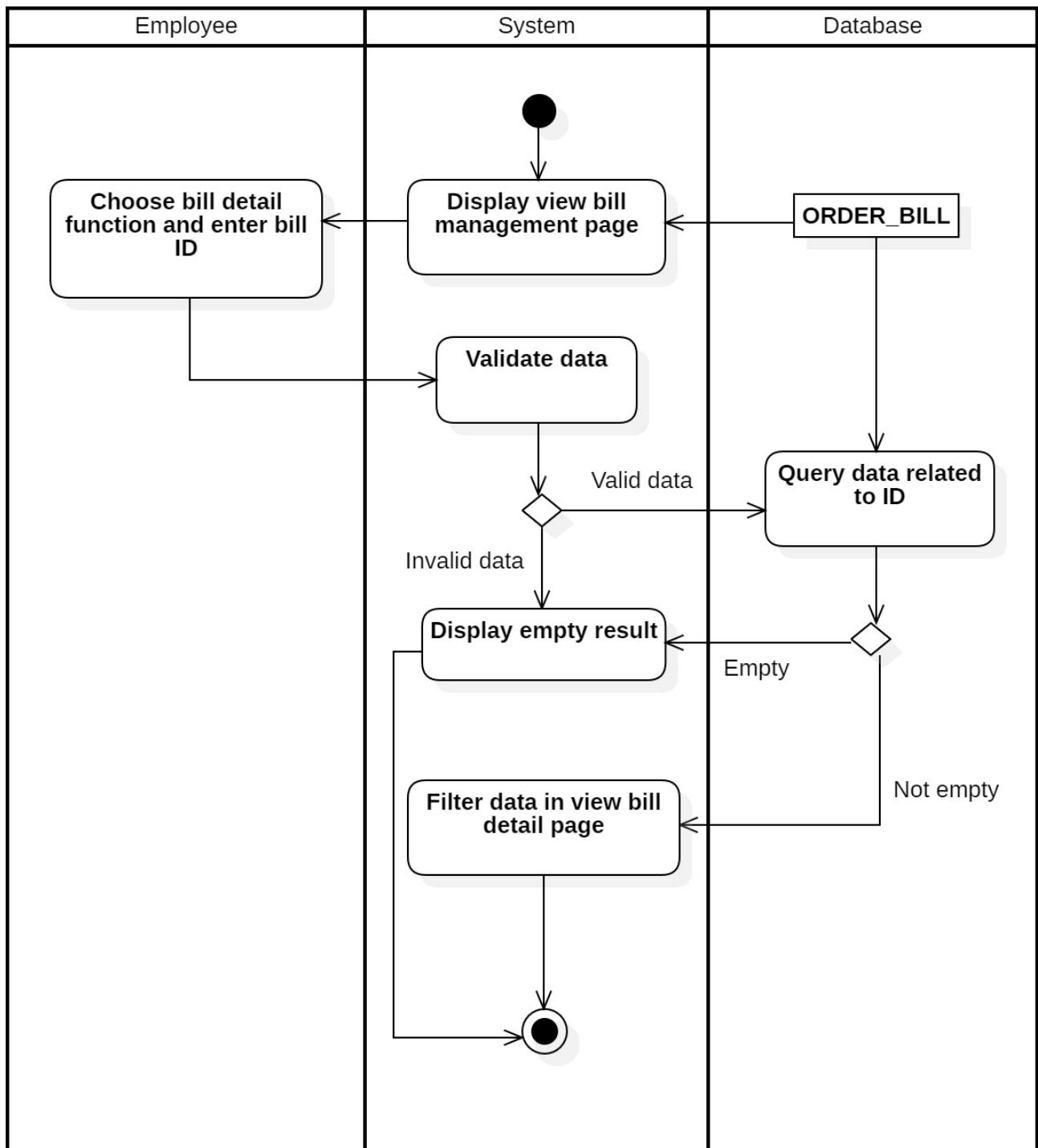
Use-case name	Update drinks
Description	Employee can update an existed drinks in the database
Trigger	Employees clicks to button “Update” at drinks management page
Pre-condition	Employees’ device must connect to Internet Employees’ account existed Employee’s status is signed-in. Drinks information must be existed in database before updating
Post-condition	Data at management is updated (with updated information at selected drinks)
Basic flow	<ol style="list-style-type: none">1. System display drinks management page with data loaded from database (existed drinks)2. Employees select update function and item to be updated3. System display drinks information form with existed data4. Employees modify drinks information5. System validate drinks information in correct format6. System validates drinks information with existed data in database.7. System updates drinks information in the database8. System updates data at drinks management page with updated data at selected item.
Alternative flow	5a. System display invalid information due to wrong format. Use-case continue at step 4. 6b. System display invalid information because selected data hadn’t been existed in the database. Use-case continue at step 4.
Exception flow	



3.2.2.4 View bill

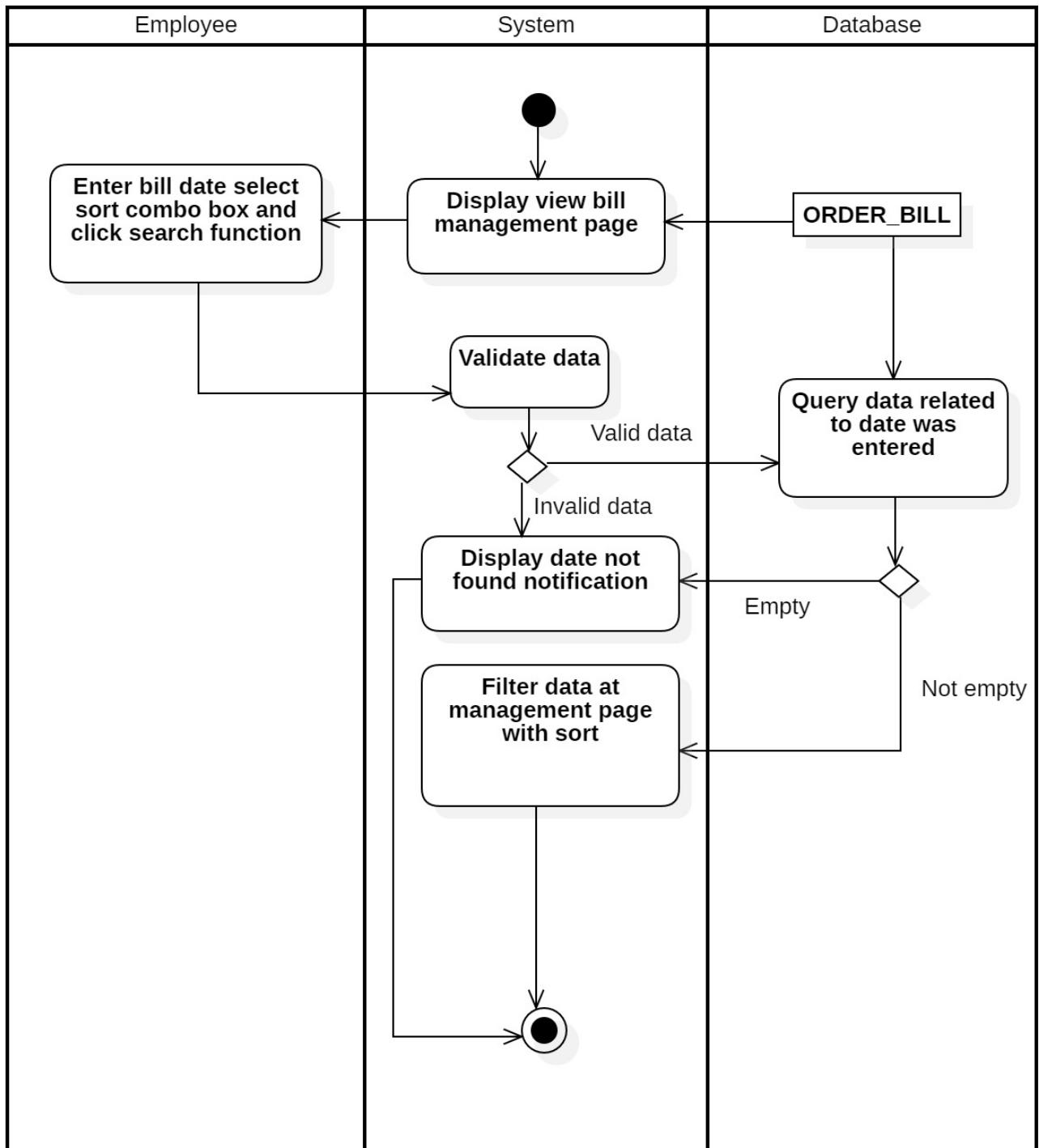
3.2.2.4a Bill Detail

Use-case name	Bill Detail
Description	Employee can view bill detail by ID.
Trigger	Employee choose bill detail function.
Pre-condition	Customer's devices must connect to Internet and access to the website. Employee-status is signed-in.
Post-condition	The bill that employee search filter in view bill detail page.
Basic flow	<ol style="list-style-type: none">1. Employee choose bill detail function.2. System validate data.3. System query data related to ID was entered.4. System filter data at view bill detail page.
Alternative flow	
Exception flow	2a. System display empty result. Use-case stop. 3a. System display empty result.. Use-case stop.



3.2.2.4b Search bill

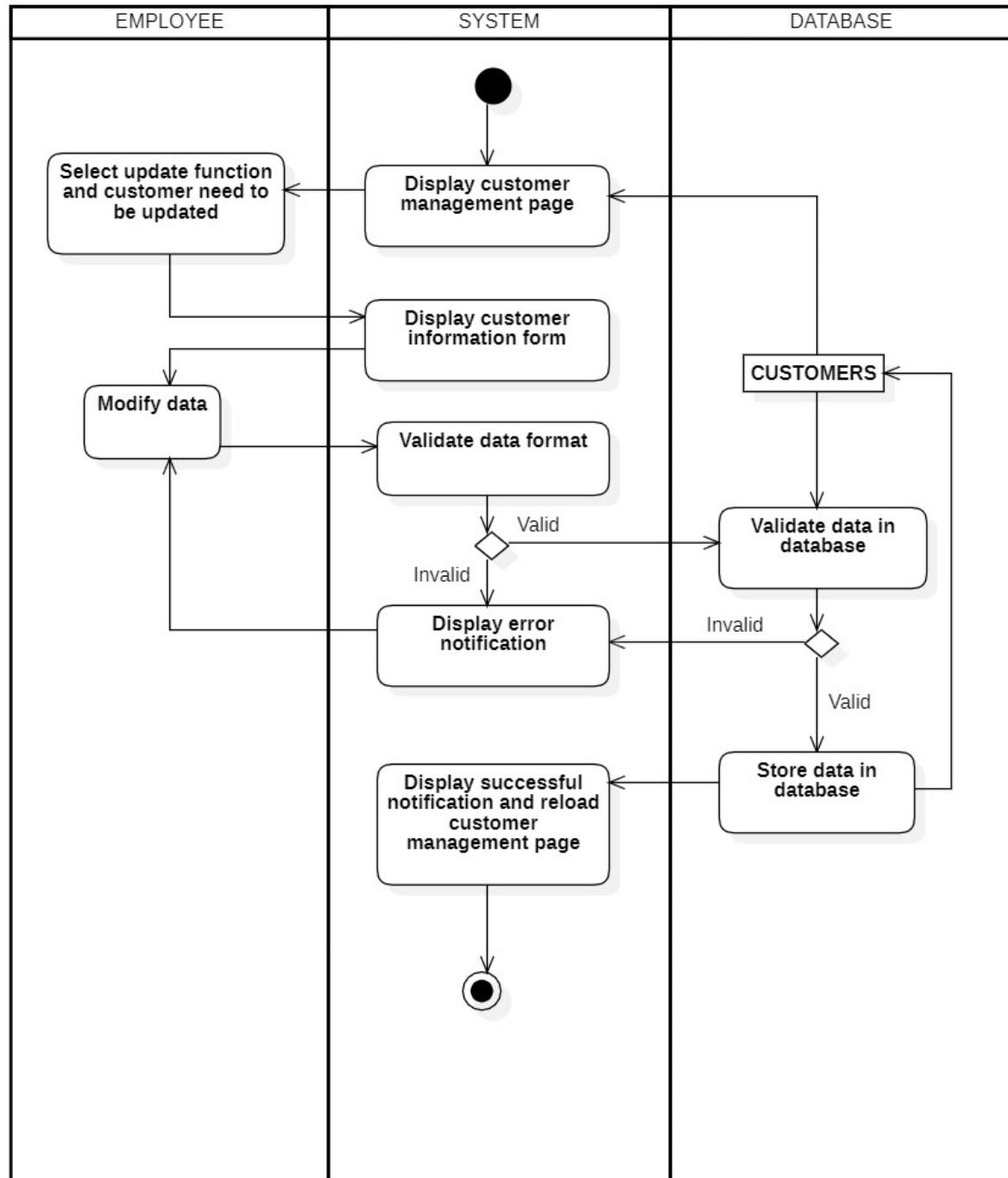
Use-case name	Search Bill
Description	Employee can search bills by date and sort by total cost.
Trigger	Employee enter bill date, select sort in combo box and choose search function.
Pre-condition	Customer's devices must connect to Internet and access to the website. Employee-status is signed-in.
Post-condition	The bill that employee search filter in view bill page.
Basic flow	<ol style="list-style-type: none">1. Employee enter bill date, select sort in combo box and choose search function.2. System validate data.3. System query data related to date was entered.4. System filter data at management page with sort.
Alternative flow	
Exception flow	2a. System display date is not found notification. Use-case stop. 3a. System display date is not found notification. Use-case stop.



3.2.2.5 Manage customer

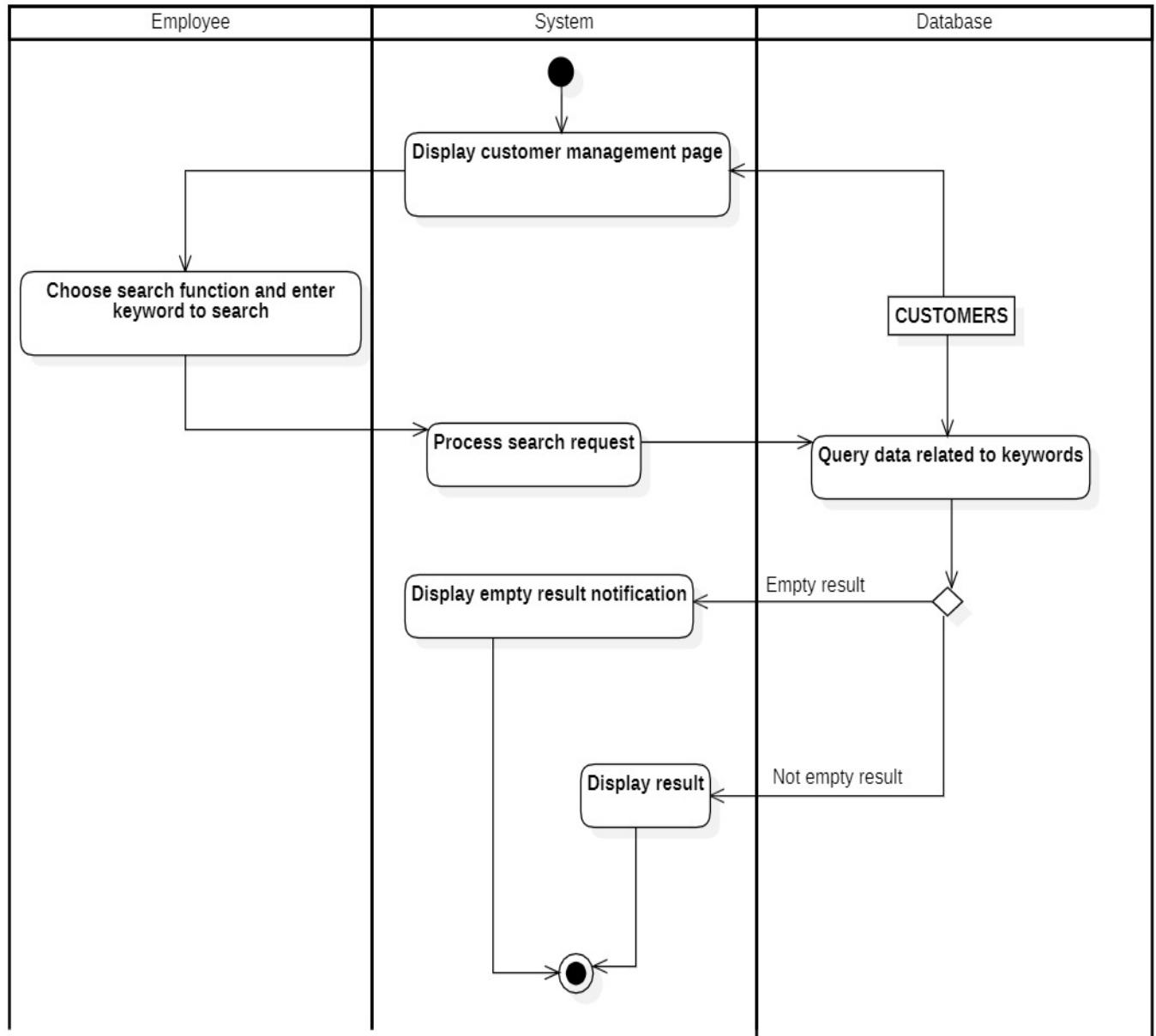
3.2.2.5a Update customer

Use-case name	Update customer
Description	Employees can update employee information
Trigger	Employees click to button “Update” on the customer management page
Pre-condition	Employees’ devices must connect to the Internet. Employees’ accounts existed Employee’s status is signed-in.
Post-condition	Data on the employee information page is updated.
Basic flow	<ol style="list-style-type: none">1. System displays the customer management page with data.2. Employee chooses to update function and the customer to be updated.3. System displays customer information form.4. Employee modifies customer information and clicks on the “submit” button.5. System validates customer information in the correct format.6. System validates customer information with existing data in the database.14. System stores customer information in the database.15. System displays the successful notification and reloads the customer management page.
Alternative flow	<ol style="list-style-type: none">5a. System displays invalid information due to the wrong format. Use-case continues to step 4.6a. System displays invalid information because data had existed in the database. Use-case continues to step 4.
Exception flow	



3.2.2.5b Search customer

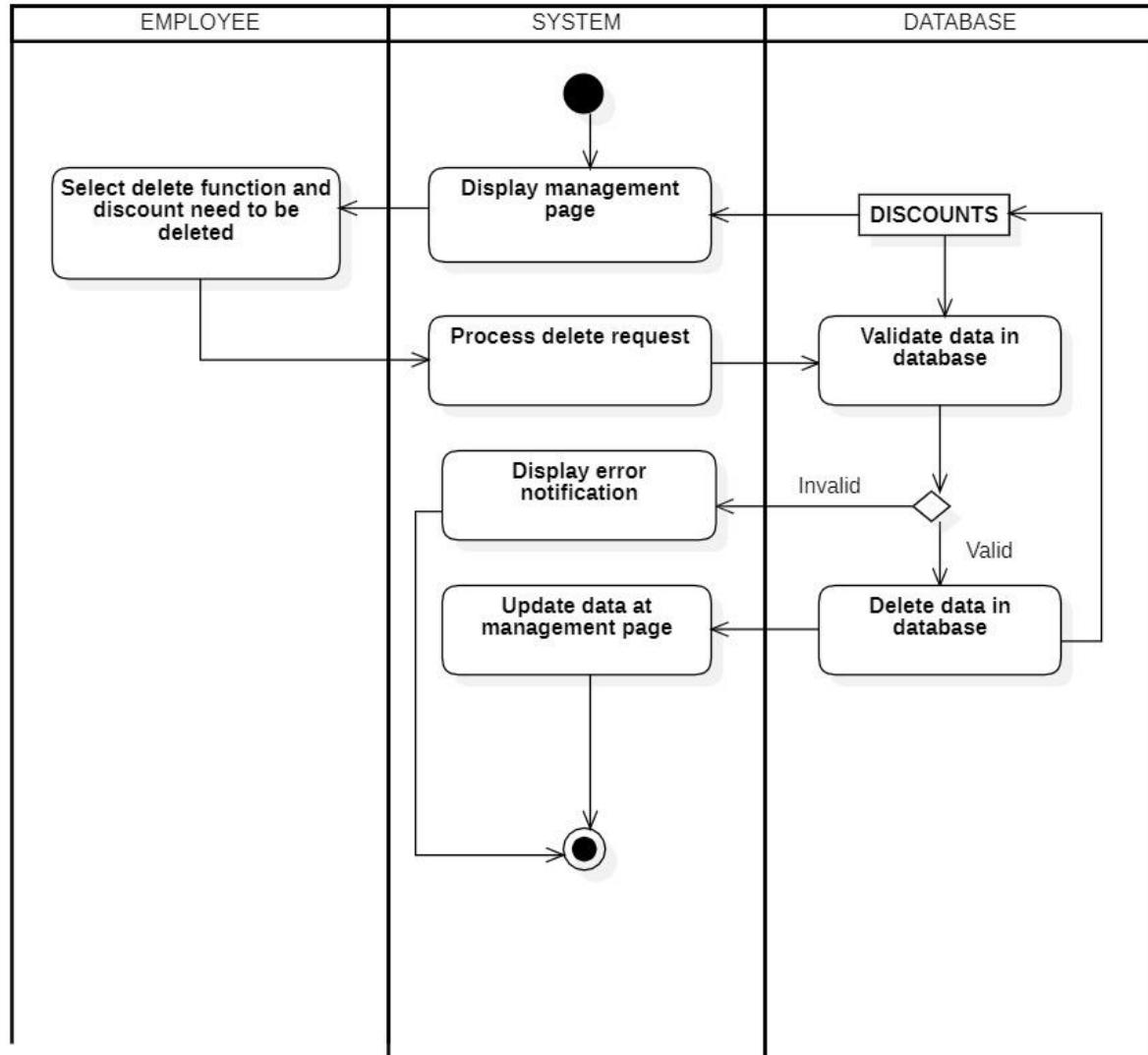
Use-case name	Search customer
Description	Employees can search customer information related to keyword
Trigger	Employees click to button “Search” on the customer management page
Pre-condition	Employees’ devices must connect to the Internet. Employee’s account existed. Employee’s status is signed-in.
Post-condition	Data on the customer information page is filtered by keyword.
Basic flow	1. System displays the customer management page with data. 2. Employee chooses search function and enters keyword to search. 3. System processes searching data. 4. System queries data based on keywords and not empty results. 5. System displays the result.
Alternative flow	
Exception flow	4a. System displays empty result notification. Use-case stop.



3.2.2.6 Manage discount

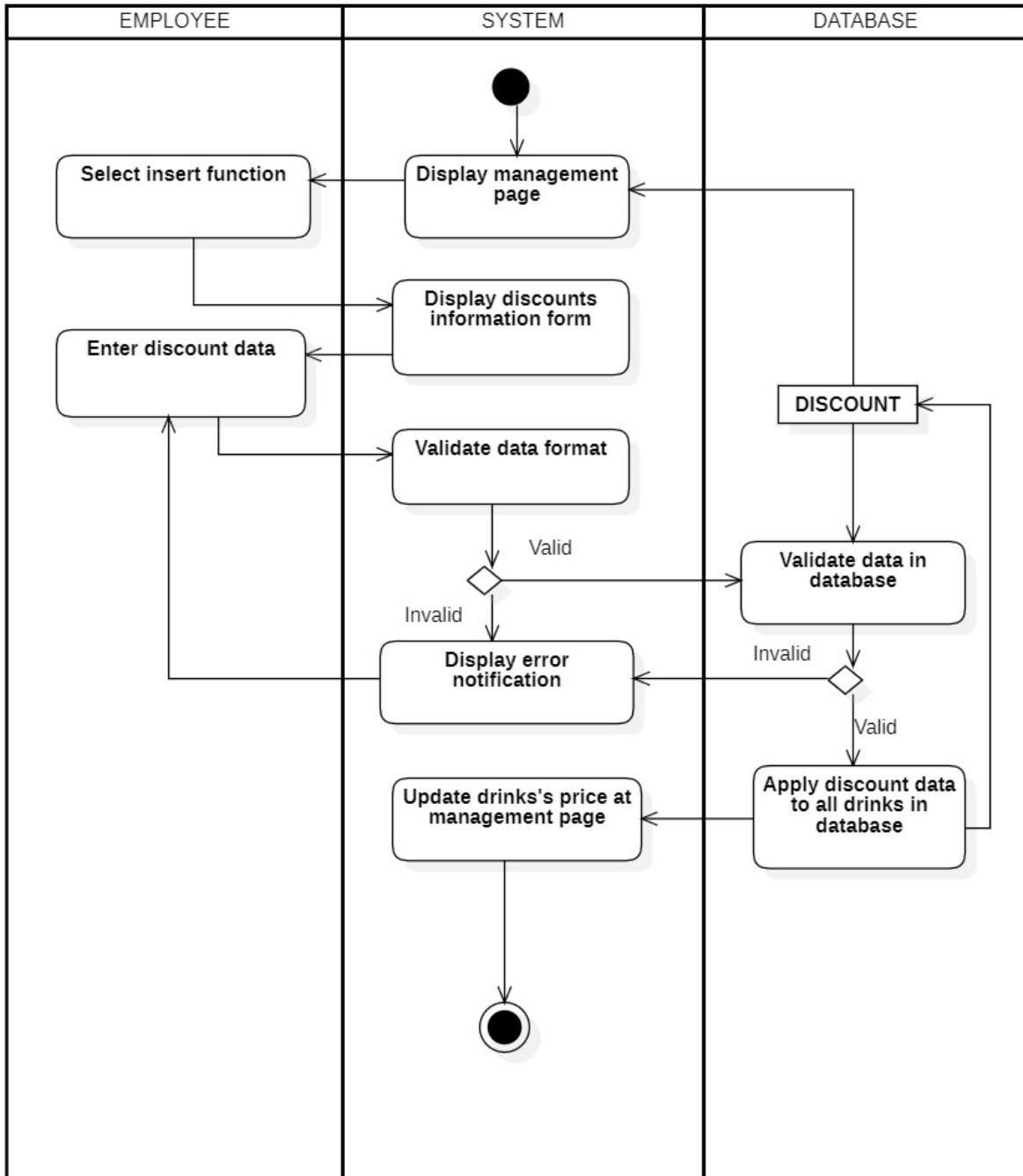
3.2.2.6a Delete discount

Use-case name	Delete discount (for Employee)
Description	Employee can delete an existed discount in the database
Trigger	Employees clicks to button “Delete” at discount management page
Pre-condition	Employees’ device must connect to Internet Employees’ account existed Employee’s status is signed-in. Discount information must be existed in database before deleting No external data involved.
Post-condition	Data at management page is updated (selected drinks had been deleted)
Basic flow	<ol style="list-style-type: none">1. System display discount management page with data loaded from database (existed discount)2. Employees select discount to be deleted3. System processes delete request4. System validates no external data required discount information and selected discount existed in database5. System deletes discount information in the database6. System updates data at discount management page with selected discount deleted.
Alternative flow	
Exception flow	5a. System display error notification because external data had required that drinks information or no data existed Use-case stop.



3.2.2.6b Insert discount

Use-case name	Insert discount
Description	Employee can add new discount to database
Trigger	Employees clicks to button “Add new” at discount management page
Pre-condition	Employees’ device must connect to Internet Employees’ account existed Employee’s status is signed-in.
Post-condition	Discount information successful insert to database Data at management page is updated (with new discount)
Basic flow	<ol style="list-style-type: none">1. System display discount management page with data loaded from database (existed drinks)2. Employees select insert function3. Employees enter discount information4. System validate discount information in correct format5. System validates discount information with existed data in database.6. System stores discount information in database7. System updates data at discount management page with inserted data (new data).
Alternative flow	<p>4a. System display invalid information due to wrong format. Use-case continue at step 3.</p> <p>5b. System display invalid information because data had been existed in the database. Use-case continue at step 3.</p>
Exception flow	

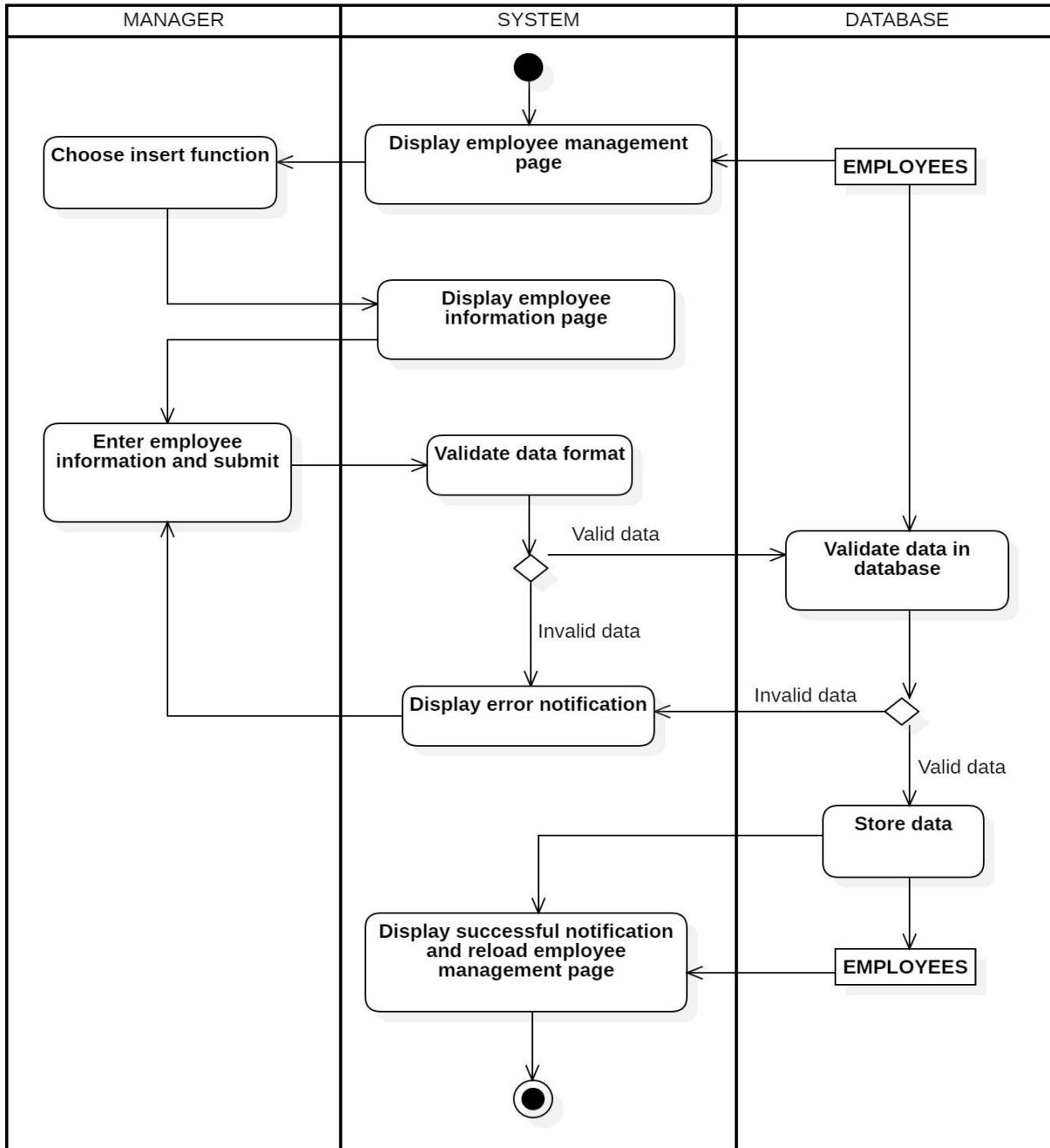


3.2.3 Use-case specification and activity diagram of Manager

3.2.3.1 Manage employee

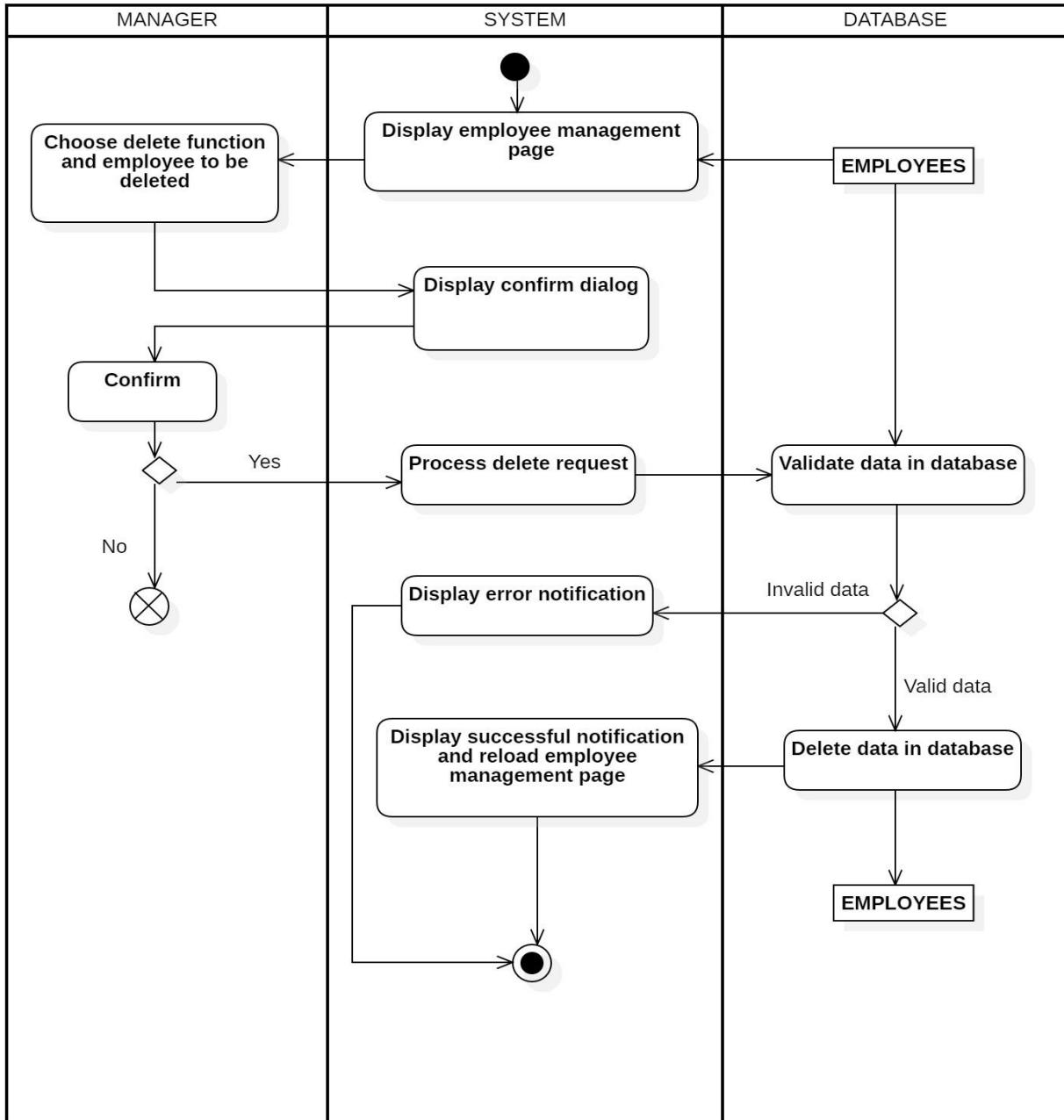
3.2.3.1a Insert Employee

Use-case name	Insert employee
Description	Managers can insert employee information
Trigger	Managers click to button “Insert” on the employee management page.
Pre-condition	Managers’ devices must connect to the Internet. Manager’s account existed. Manager’s status is signed-in.
Post-condition	Data on the employee information page is updated.
Basic flow	<ol style="list-style-type: none">1. System displays the employee management page with data.2. Manager chooses to insert function.3. System displays employee information page.4. Manager enters employee information and clicks on the “Submit” button.5. System validates employee information in the correct format.6. System validates employee information with existing data in the database.7. System stores employee information in the database.8. System displays the successful notification and reloads the employee management page.
Alternative flow	<ol style="list-style-type: none">5a. System displays invalid information due to the wrong format. Use-case continues to step 4.6a. System displays invalid information because data had existed in the database. Use-case continues to step 4.
Exception flow	



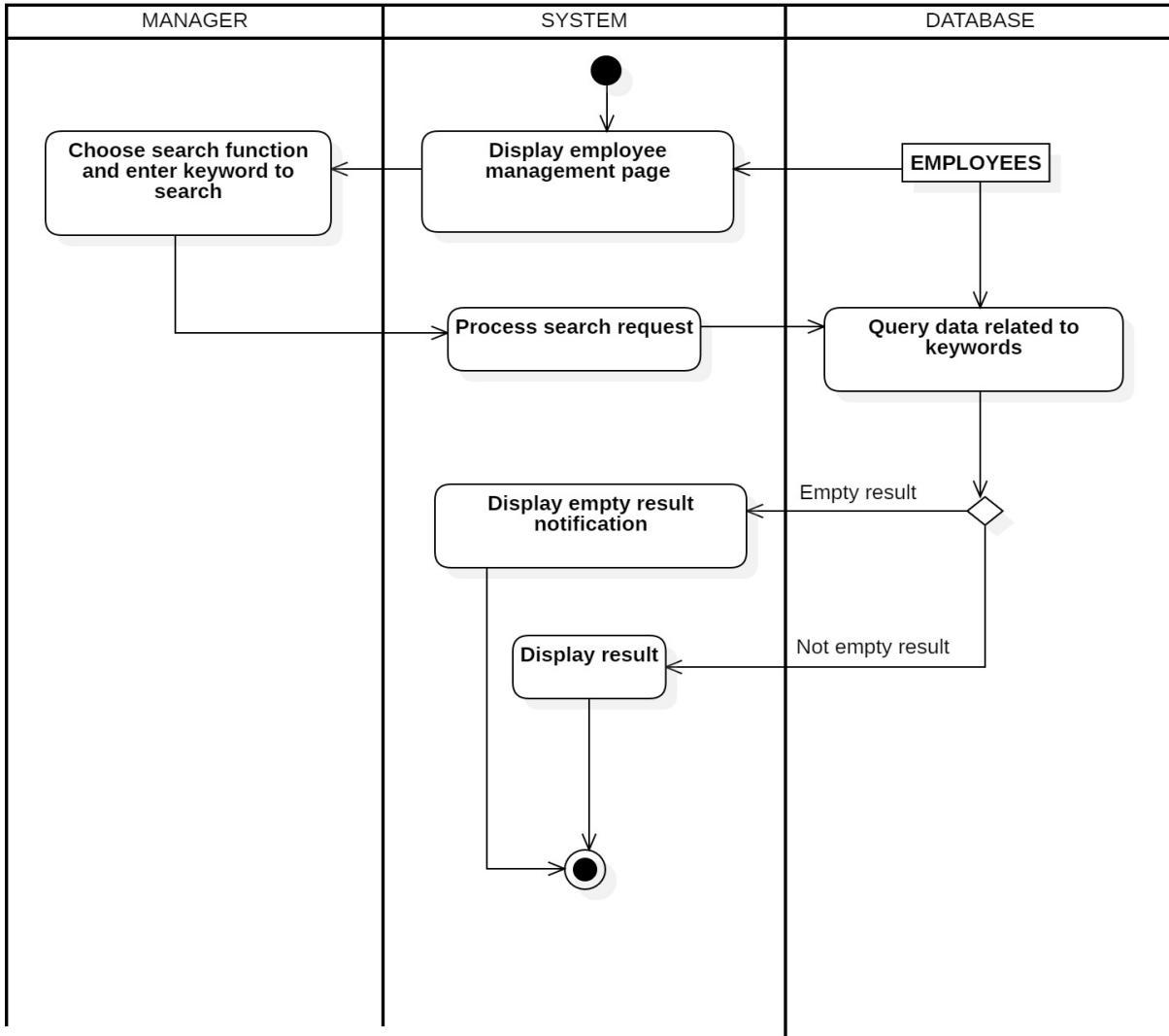
3.2.3.1b Delete Employee

Use-case name	Delete employee
Description	Managers can delete employee information
Trigger	Managers click to button “Delete” on the employee management page
Pre-condition	Managers’ devices must connect to the Internet. Manager’s account existed. Manager’s status is signed-in.
Post-condition	Data on the employee information page is updated.
Basic flow	1. System displays the employee management page with data. 2. Manager chooses to delete function and the employee to be deleted. 3. System displays confirm dialog. 4. Manager chooses the “yes” button to confirm the deletion. 5. System processes delete requests. 6. System validates no external data required employee information and selected employee existed in the database. 7. System deletes employee information in the database. 8. System displays the successful notification and reloads the customer management page
Alternative flow	
Exception flow	4a. Use-case stop 6a. System display error notification because external data had required that employee information or no data existed. Use-case stop.



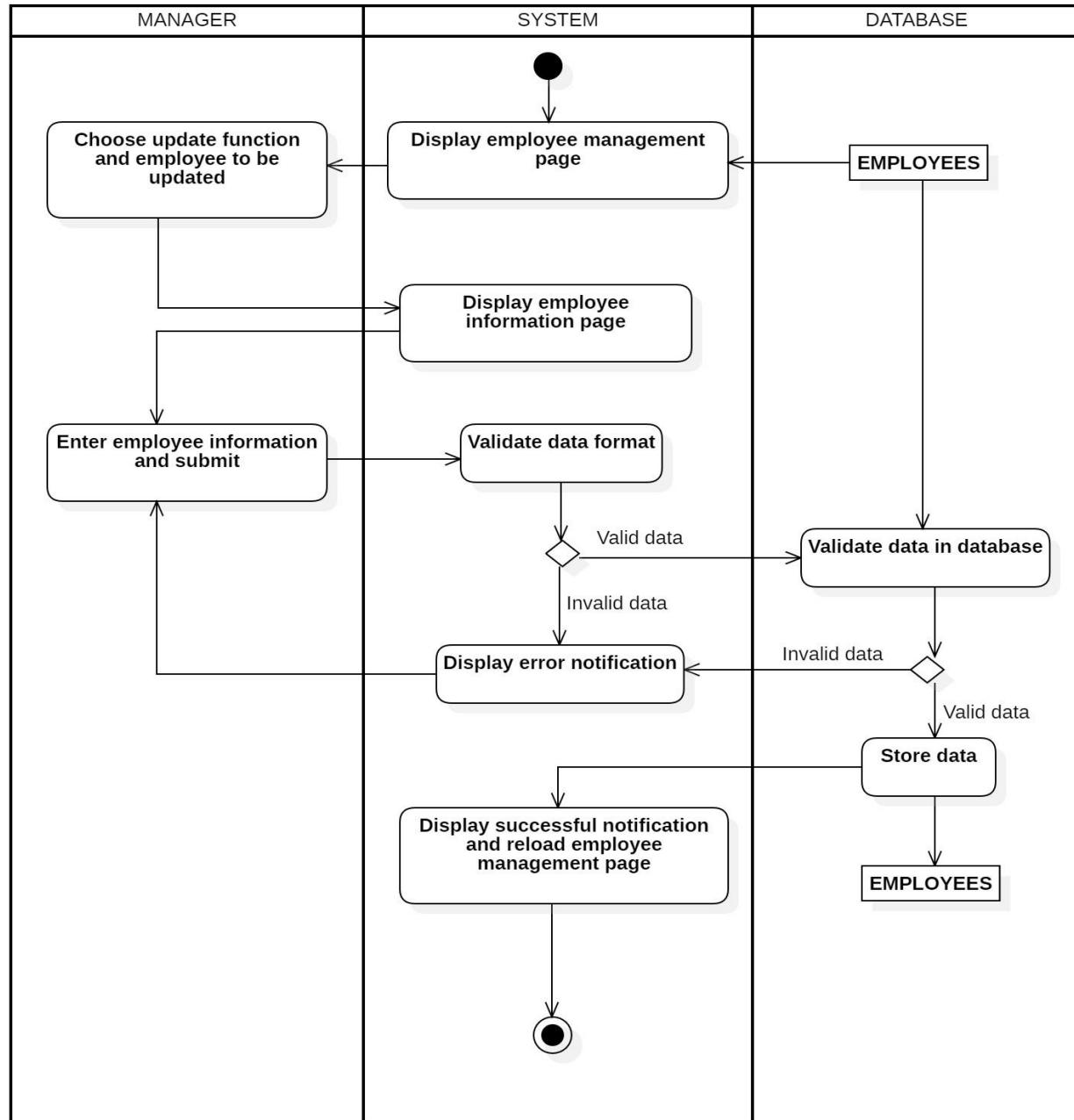
3.2.3.1c Search Employee

Use-case name	Search employee
Description	Managers can search employee information related to keyword
Trigger	Managers click to button “Search” on the employee management page
Pre-condition	Managers’ devices must connect to the Internet. Manager’s account existed. Manager’s status is signed-in.
Post-condition	Data on the employee information page is filtered by keyword.
Basic flow	1. System displays the employee management page with data. 2. Manager chooses search function and enters keyword to search. 3. System processes searching data. 4. System queries data based on keywords and not empty results. 5. System displays the result.
Alternative flow	
Exception flow	4a. System displays empty result notification. Use-case stop.



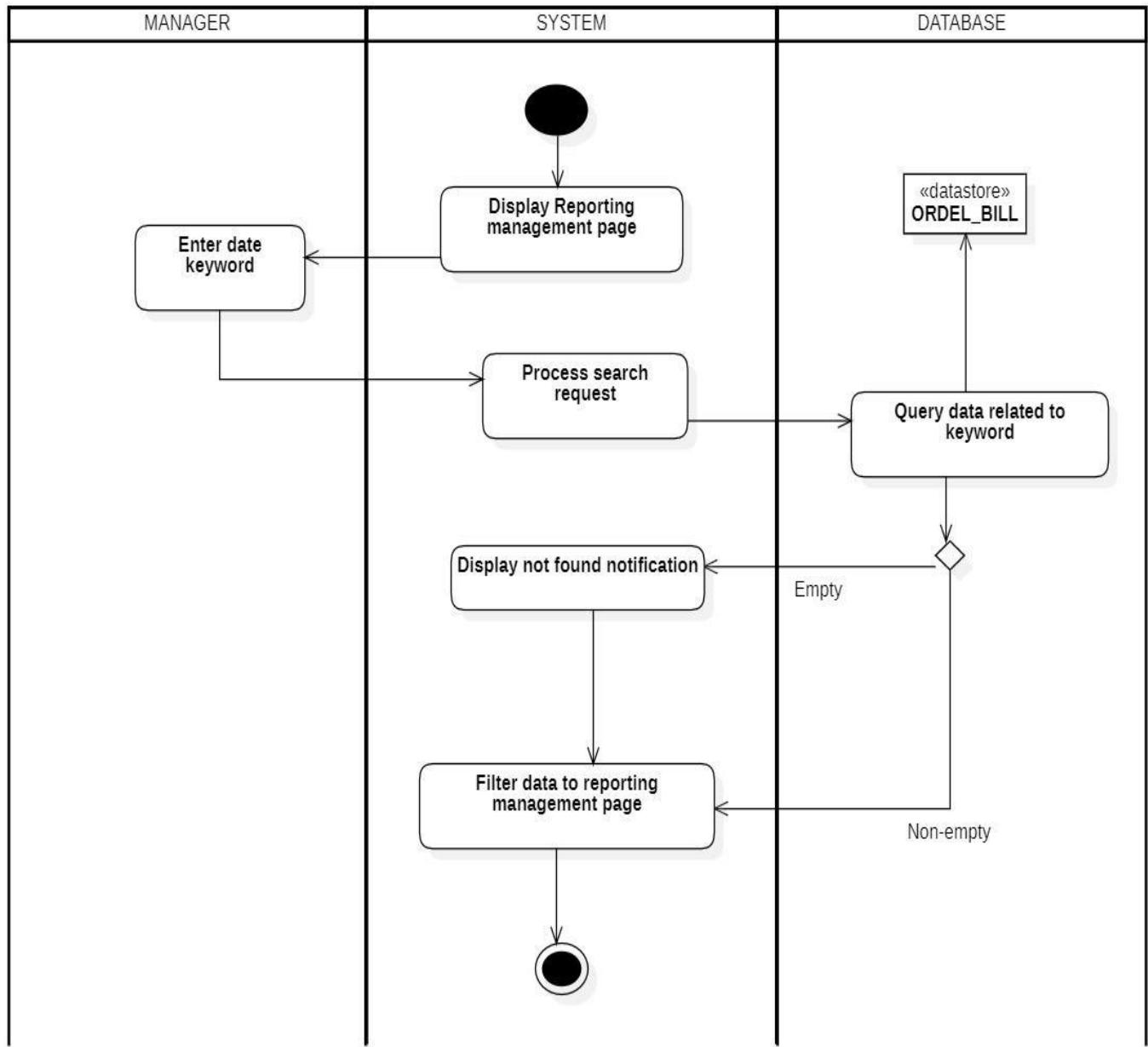
3.2.3.1d Update Employee

Use-case name	Update employee
Description	Managers can update employee information
Trigger	Managers click to button “Update” on the employee management page
Pre-condition	Managers’ devices must connect to the Internet. Manager’s account existed. Manager’s status is signed-in.
Post-condition	Data on the employee information page is updated.
Basic flow	<ol style="list-style-type: none">1. System displays the employee management page with data.2. Manager chooses to update function and the employee to be updated.3. System displays employee information page.4. Manager modifies employee information and clicks on the “submit” button.5. System validates employee information in the correct format.6. System validates employee information with existing data in the database.14. System stores employee information in the database.15. System displays the successful notification and reloads the employee management page.
Alternative flow	<ol style="list-style-type: none">5a. System display invalid information due to the wrong format. Use-case continues to step 4.6a. System displays invalid information because data had existed in the database. Use-case continues to step 4.
Exception flow	



3.2.3.2 Reporting

Use-case name	Reporting
Description	Manager can search bill in the database
Trigger	Manager clicks to button “Search” at View Bill management page
Pre-condition	Manager’ device must connect to Internet Manager’ account existed Manager’s status is signed-in.
Post-condition	Data at management page is filtered by keywords
Basic flow	<ol style="list-style-type: none">1. System display report management page with data loaded from database (existed bills)2. Manager enter data keywords3. System process searching data4. System queries data based on keywords5. System filters data at Report management page (data related to keyword is retained)
Alternative flow	4a. System display empty result notification. Use-case continue at step 5.(no data retained)
Exception flow	

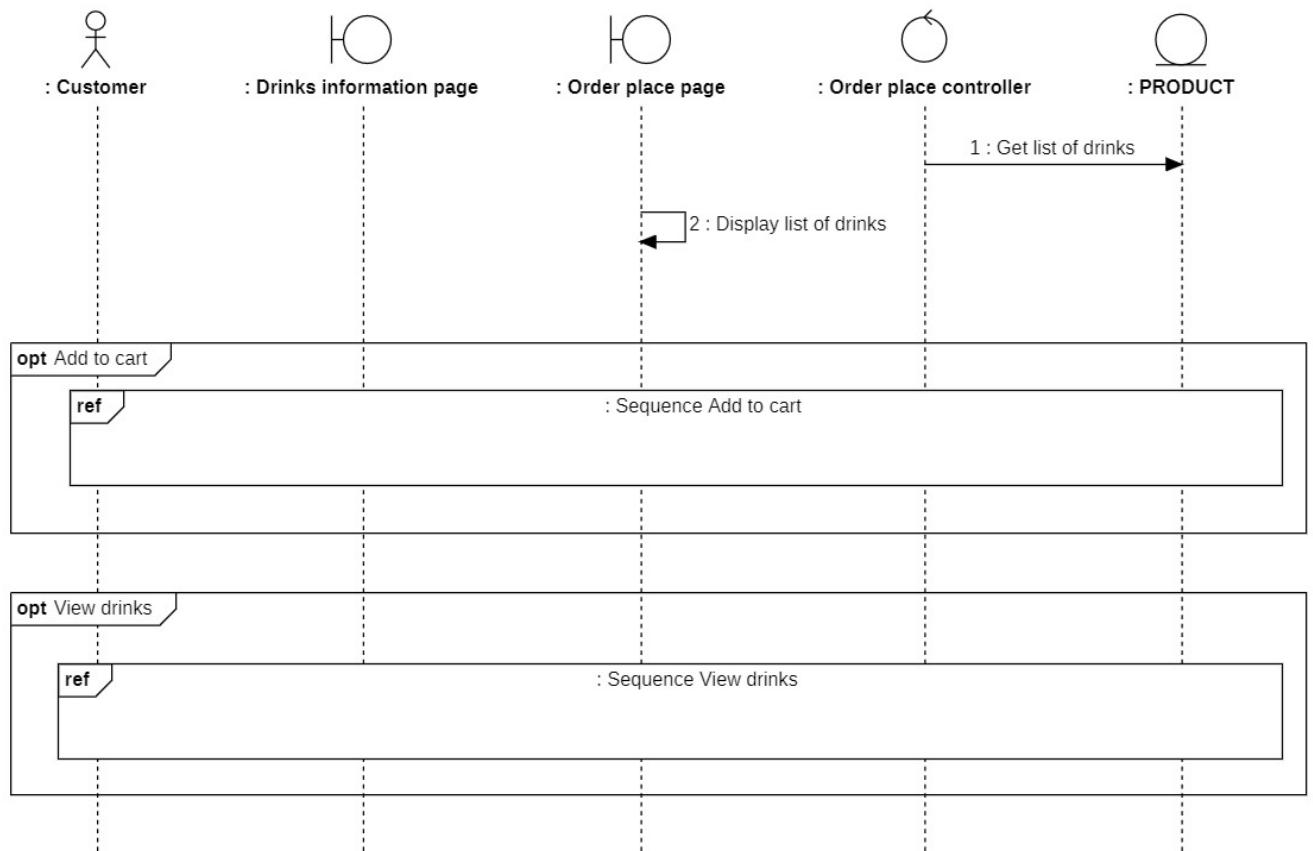


3.3 Sequence diagram and class diagram

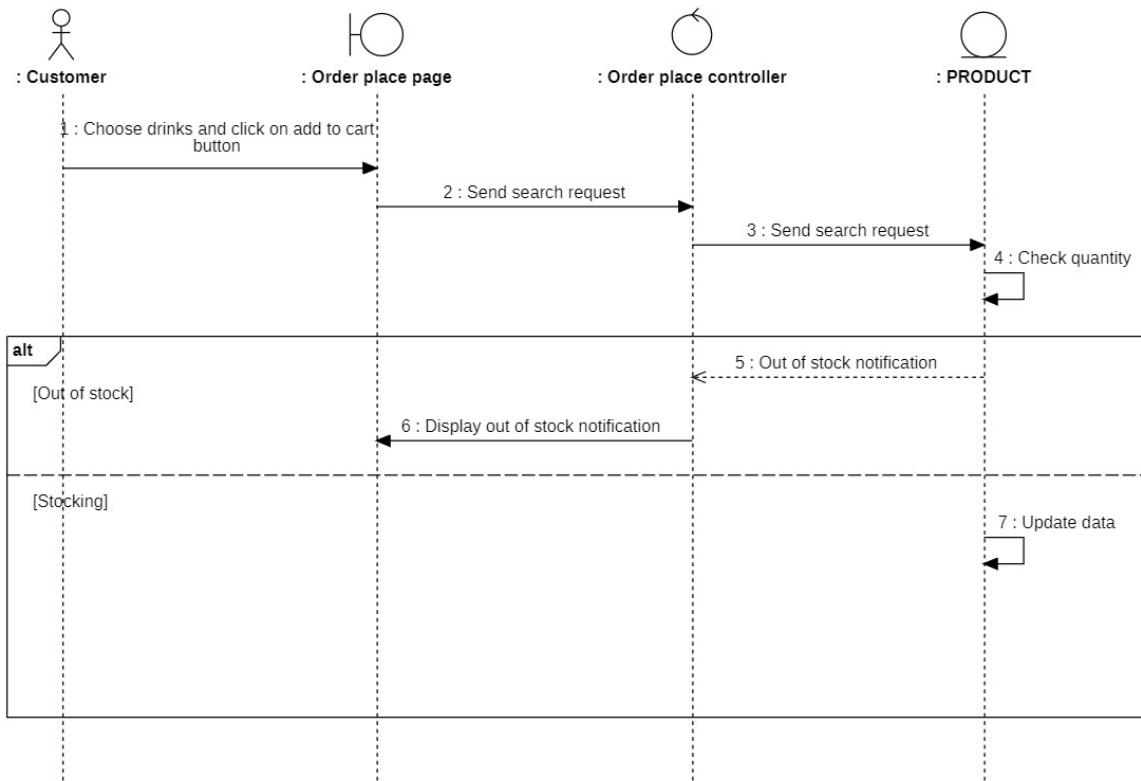
3.3.1 Sequence diagram and class diagram of Customer

3.3.1.1 Order place

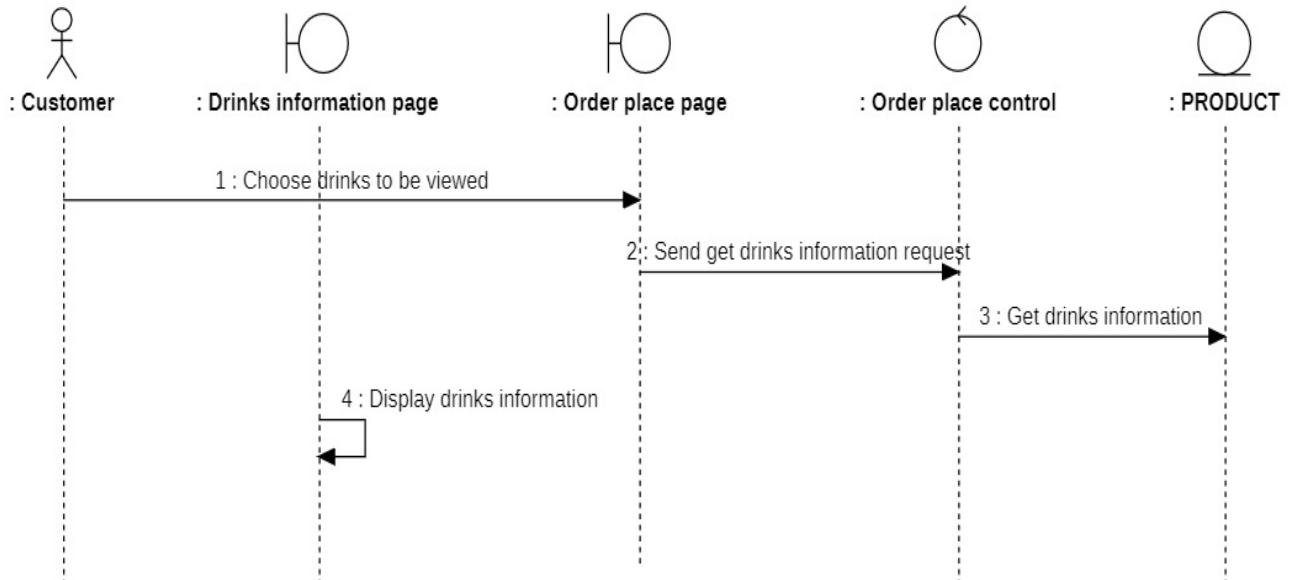
3.3.1.1a Sequence diagram



1. Add to cart



2. View drinks



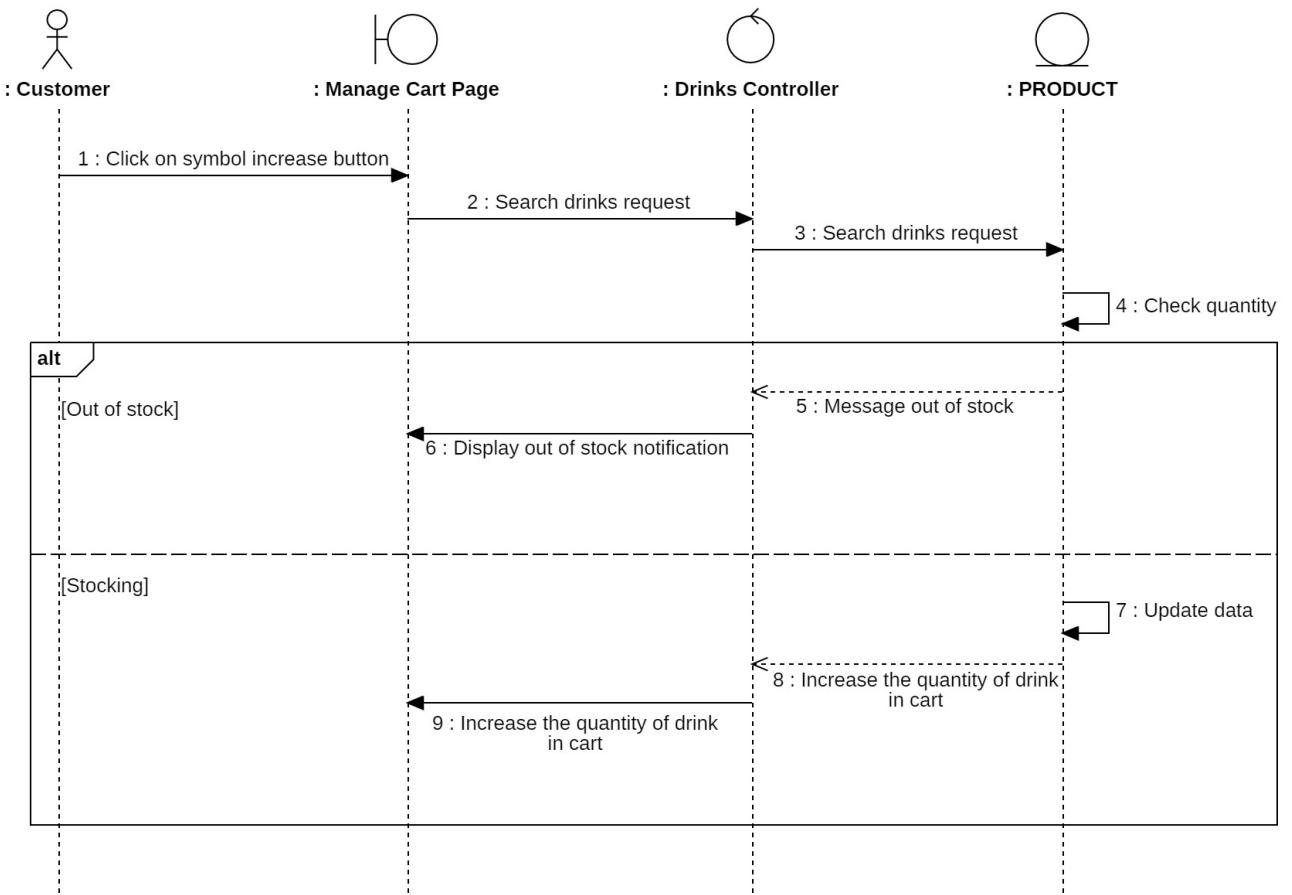
3.3.1.1b Class diagram



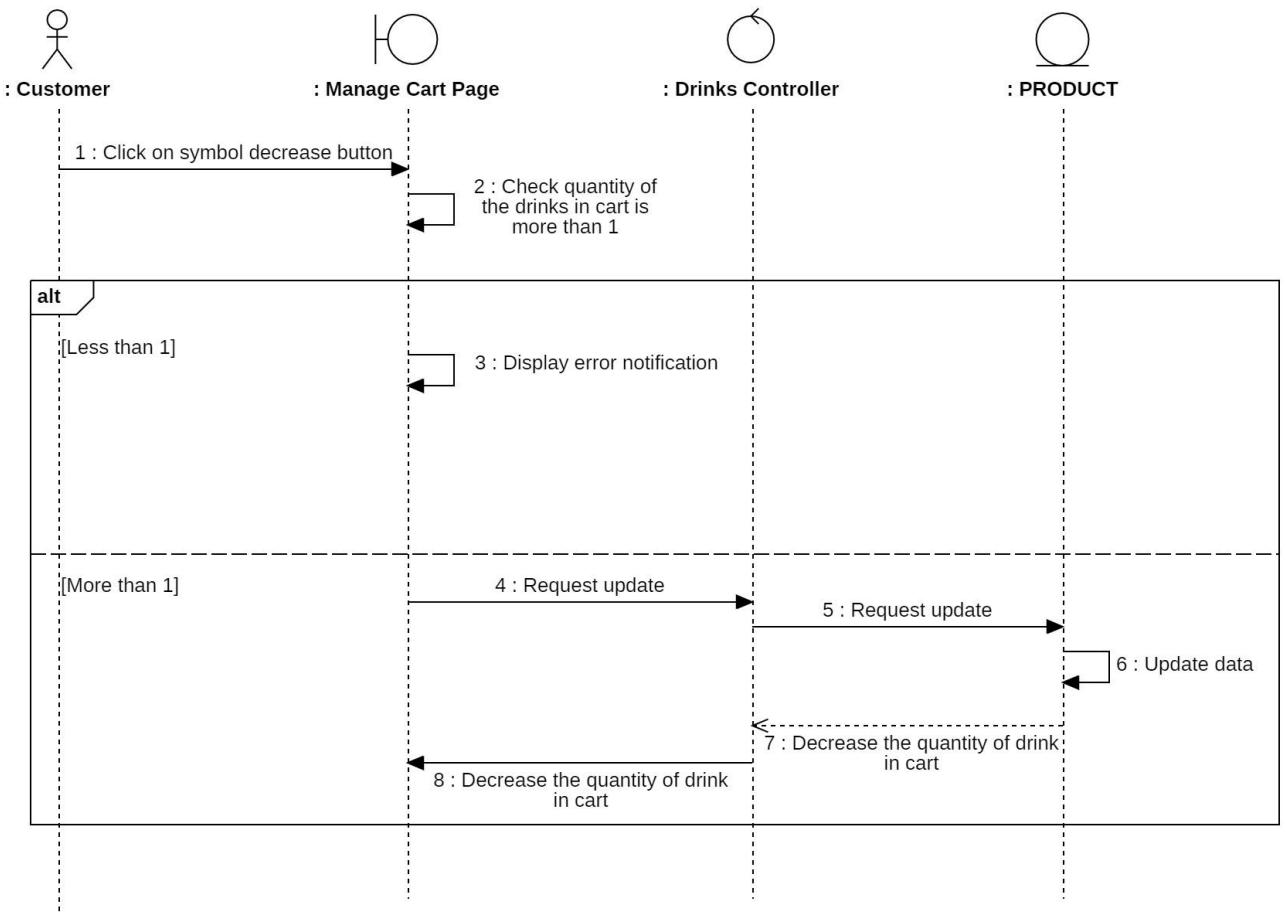
3.3.1.2 Manage cart

3.3.1.2a Sequence diagram

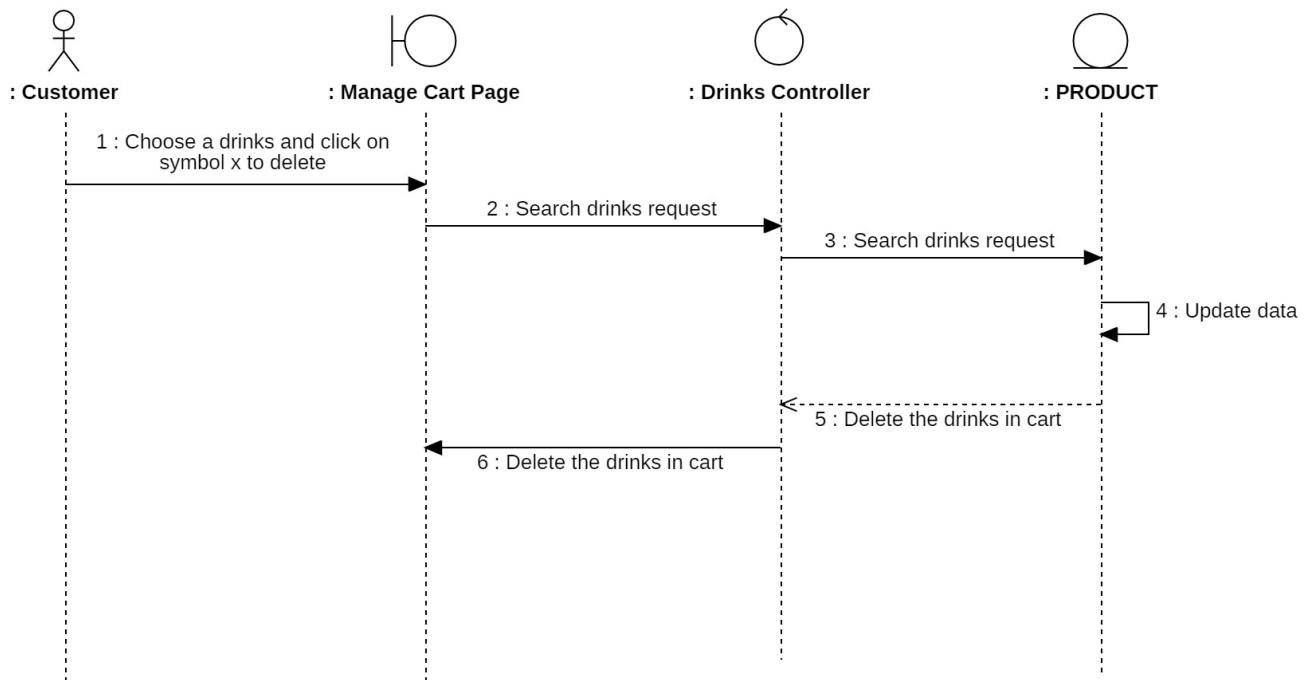
1. Increase a drink



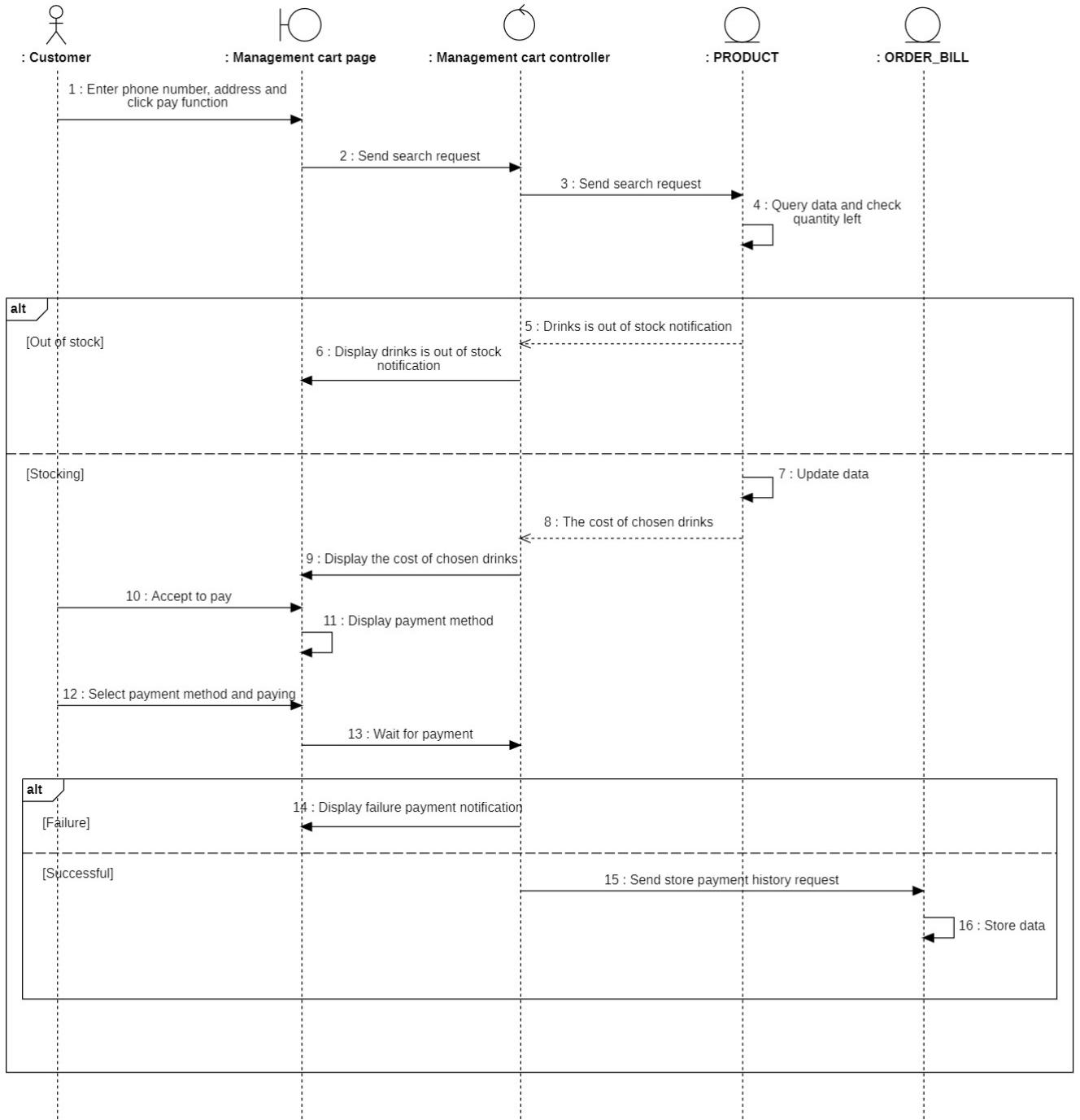
2. Decrease a drink



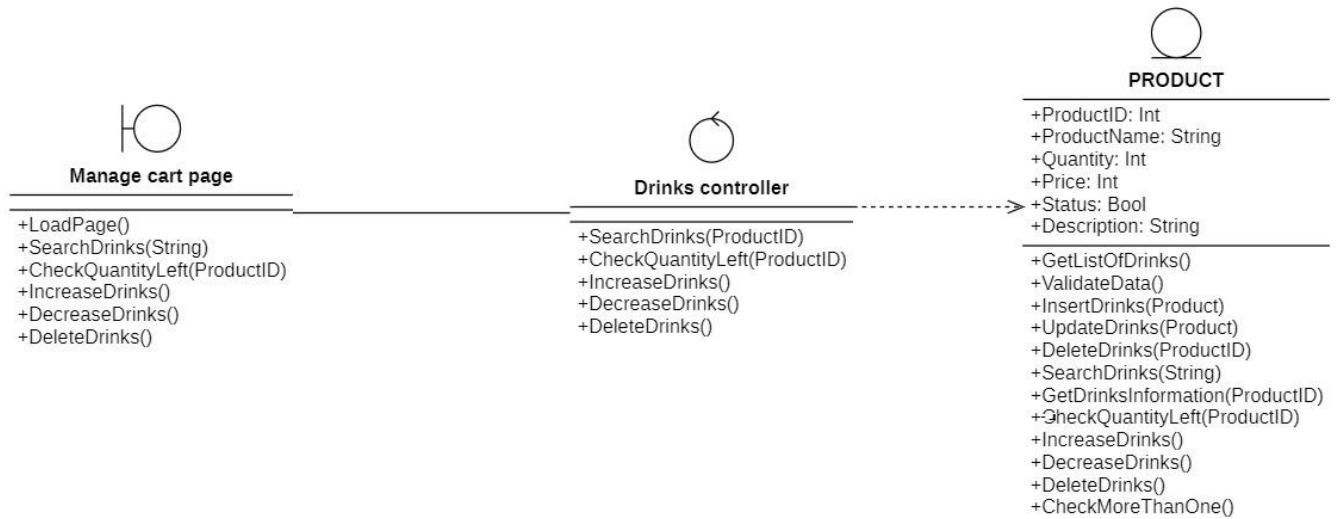
3. Delete a drinks



4. Payment

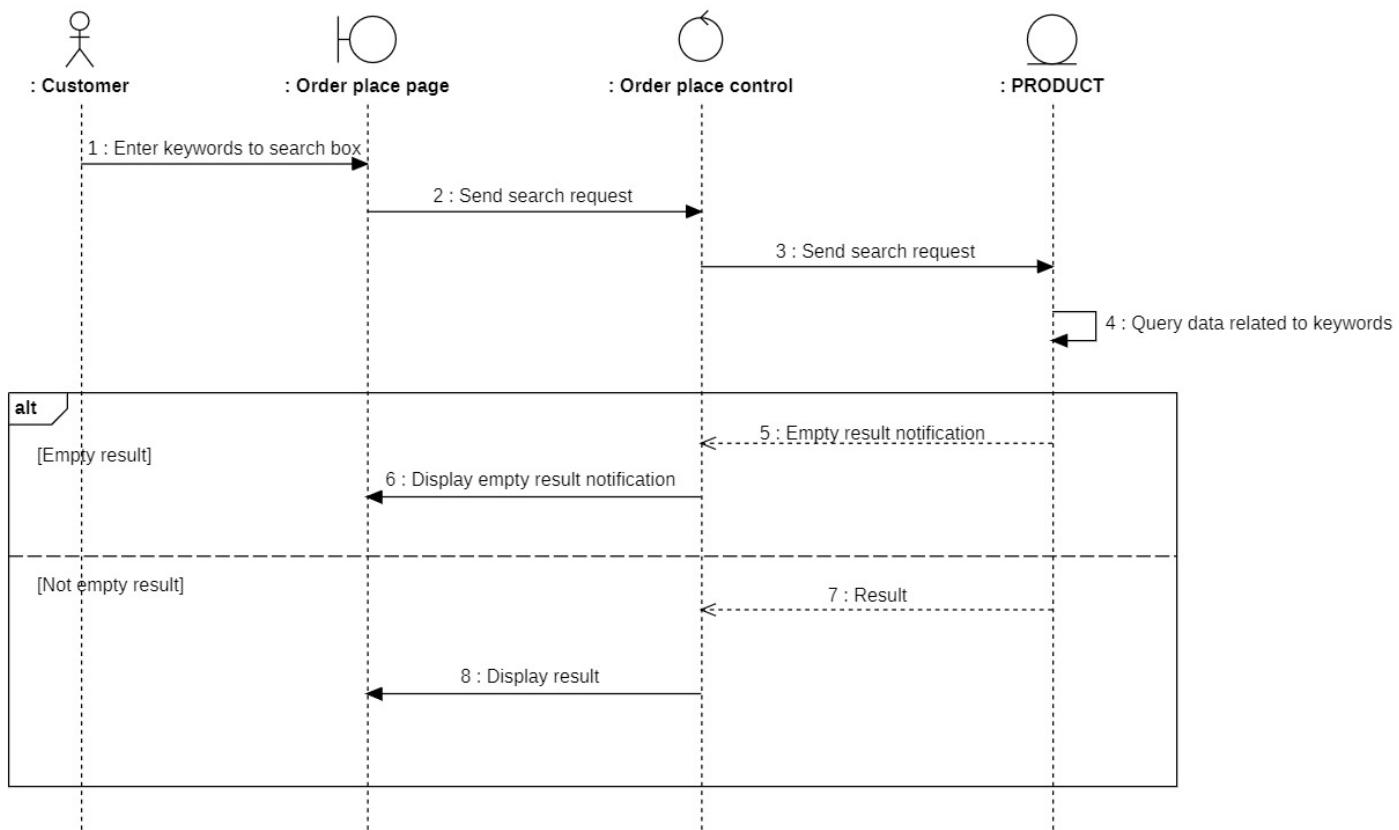


3.3.1.2b Class diagram

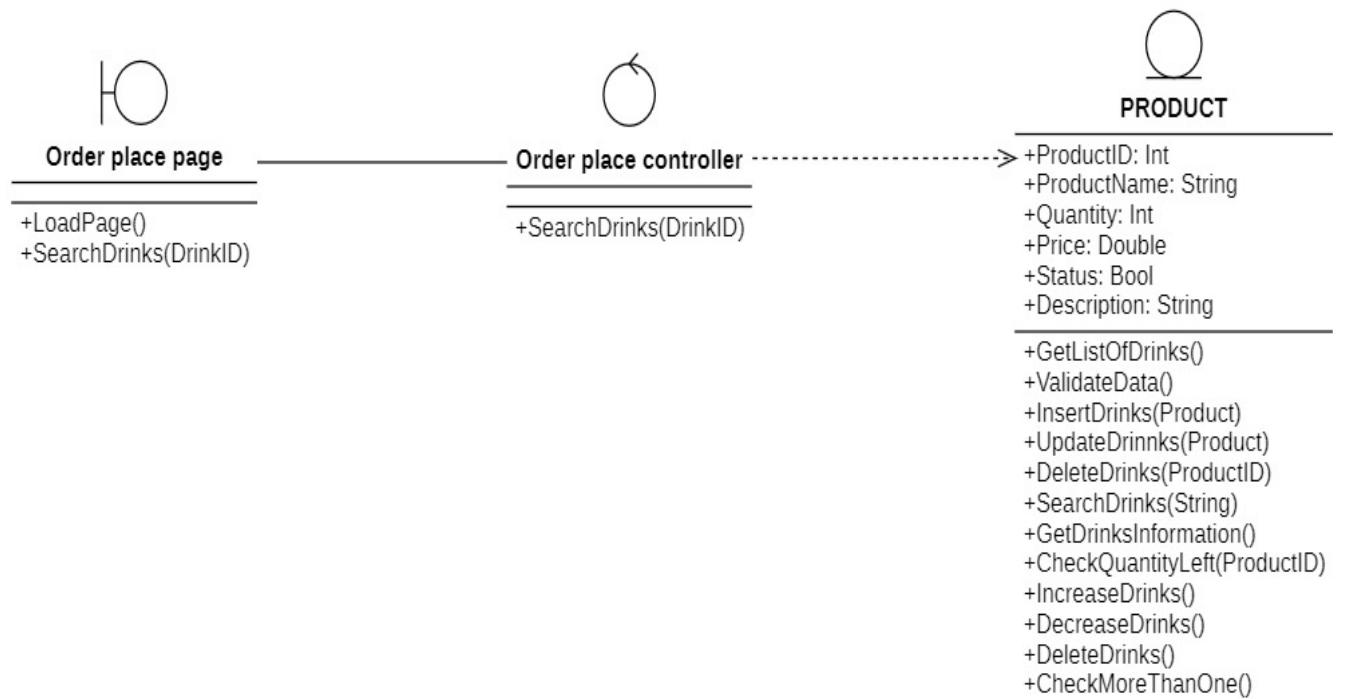


3.3.1.3 Search drinks

3.3.1.3a Sequence diagram



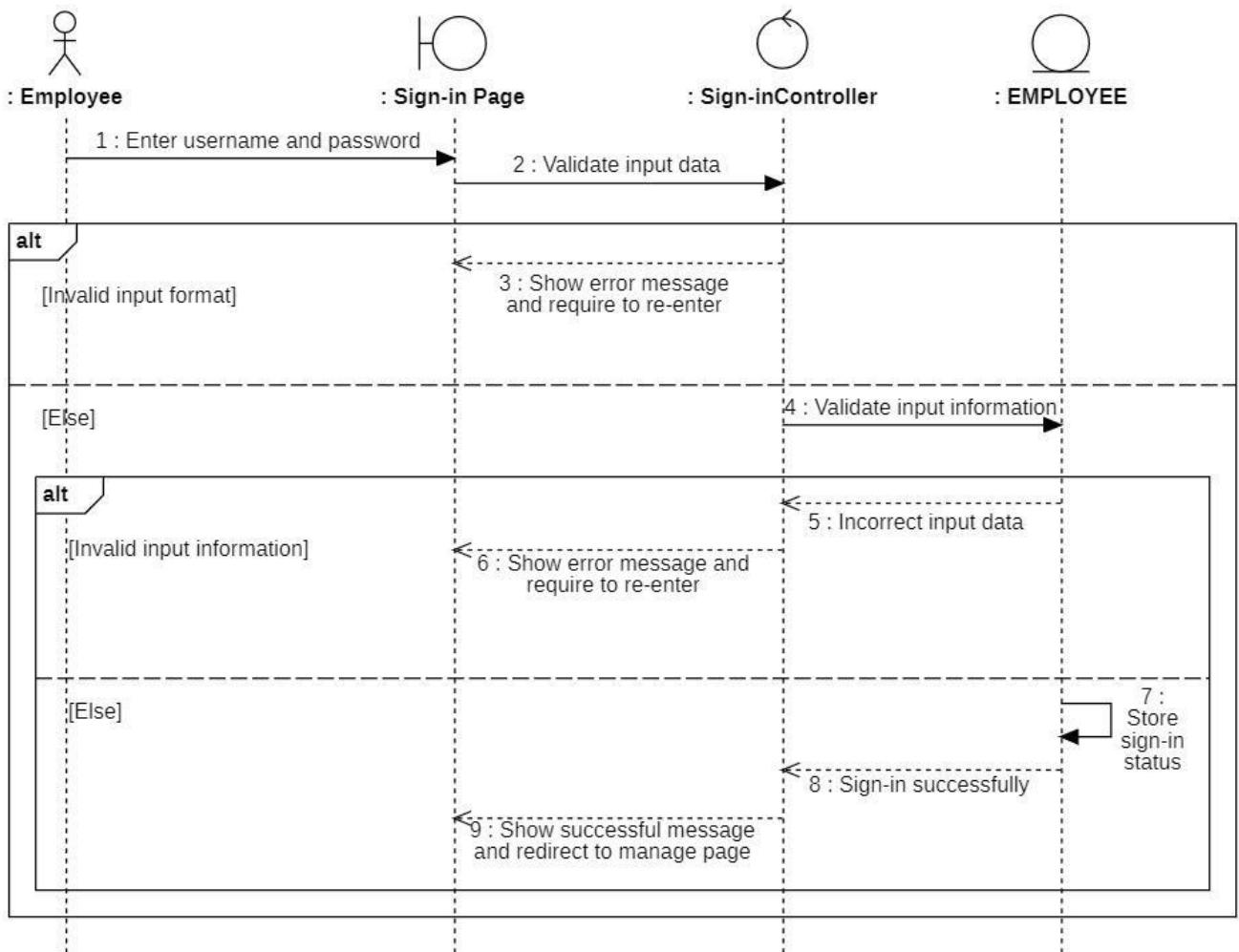
3.3.1.3b Class diagram



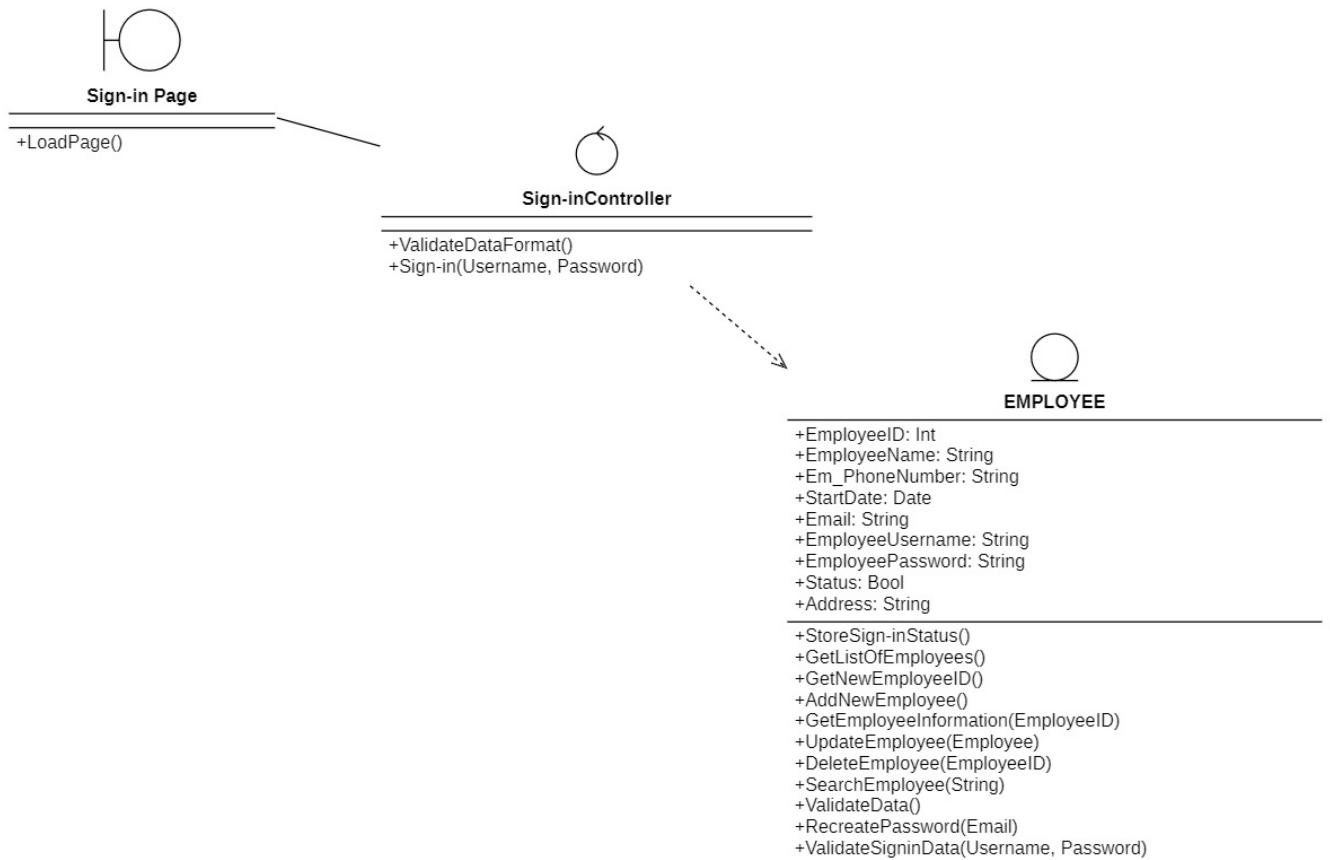
3.3.2 Sequence diagram and class diagram of Employee

3.3.2.1 Sign in

3.3.2.1a Sequence diagram

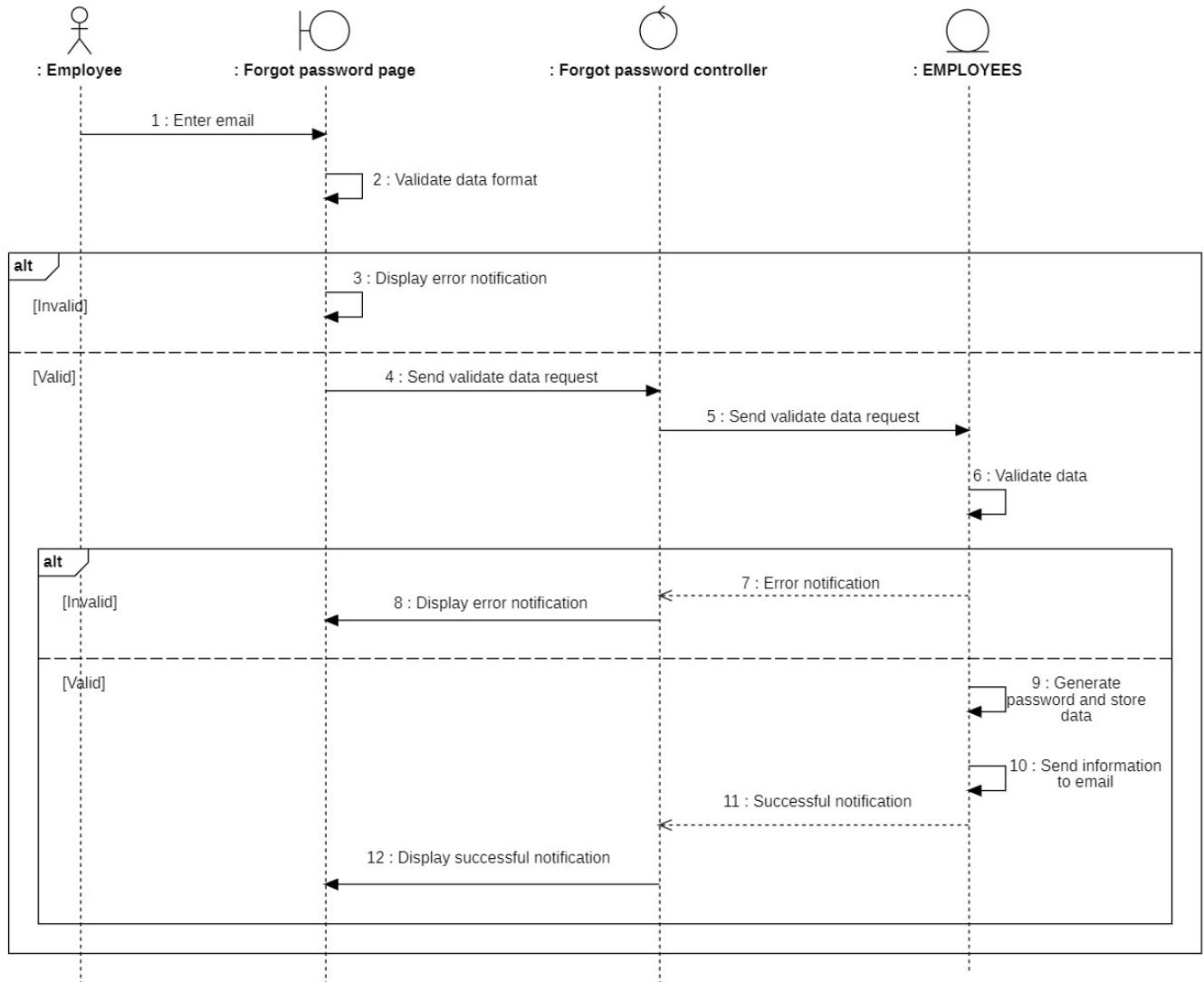


3.3.2.1b Class diagram

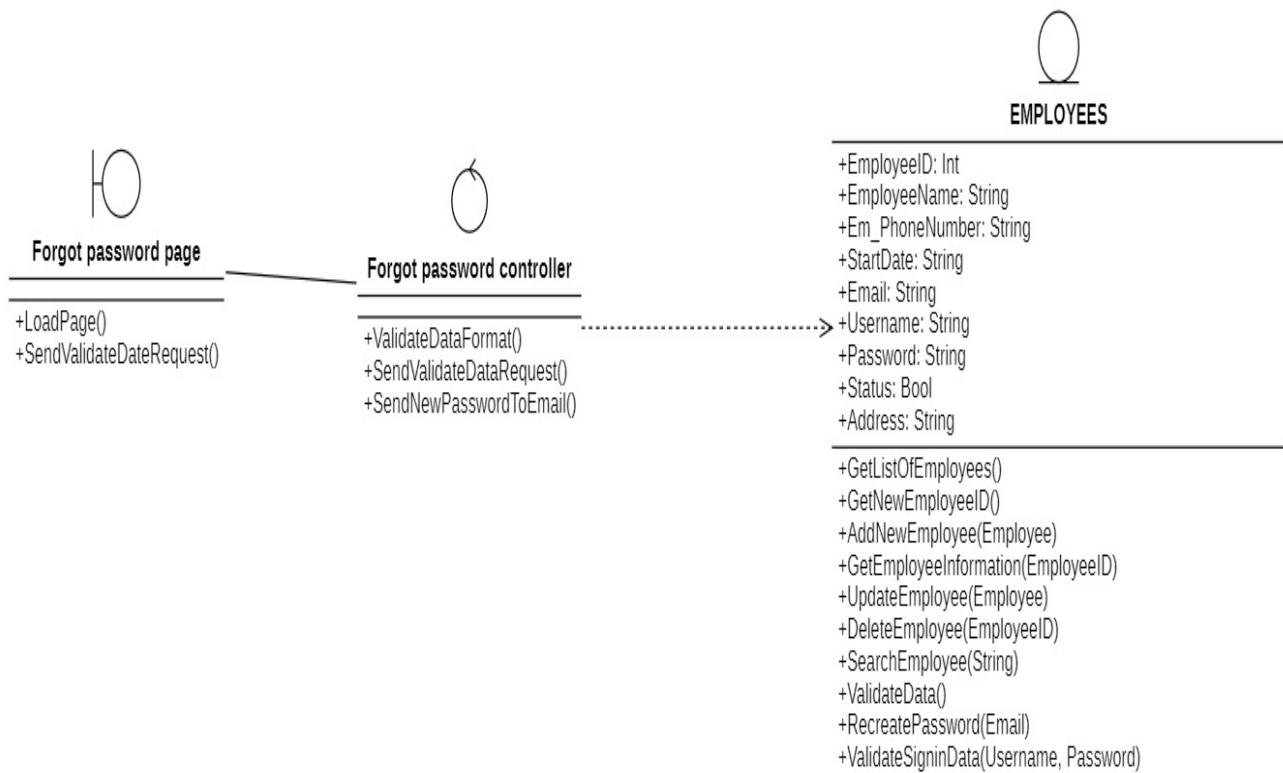


3.3.2.2 Re-create password

3.3.2.2a Sequence diagram

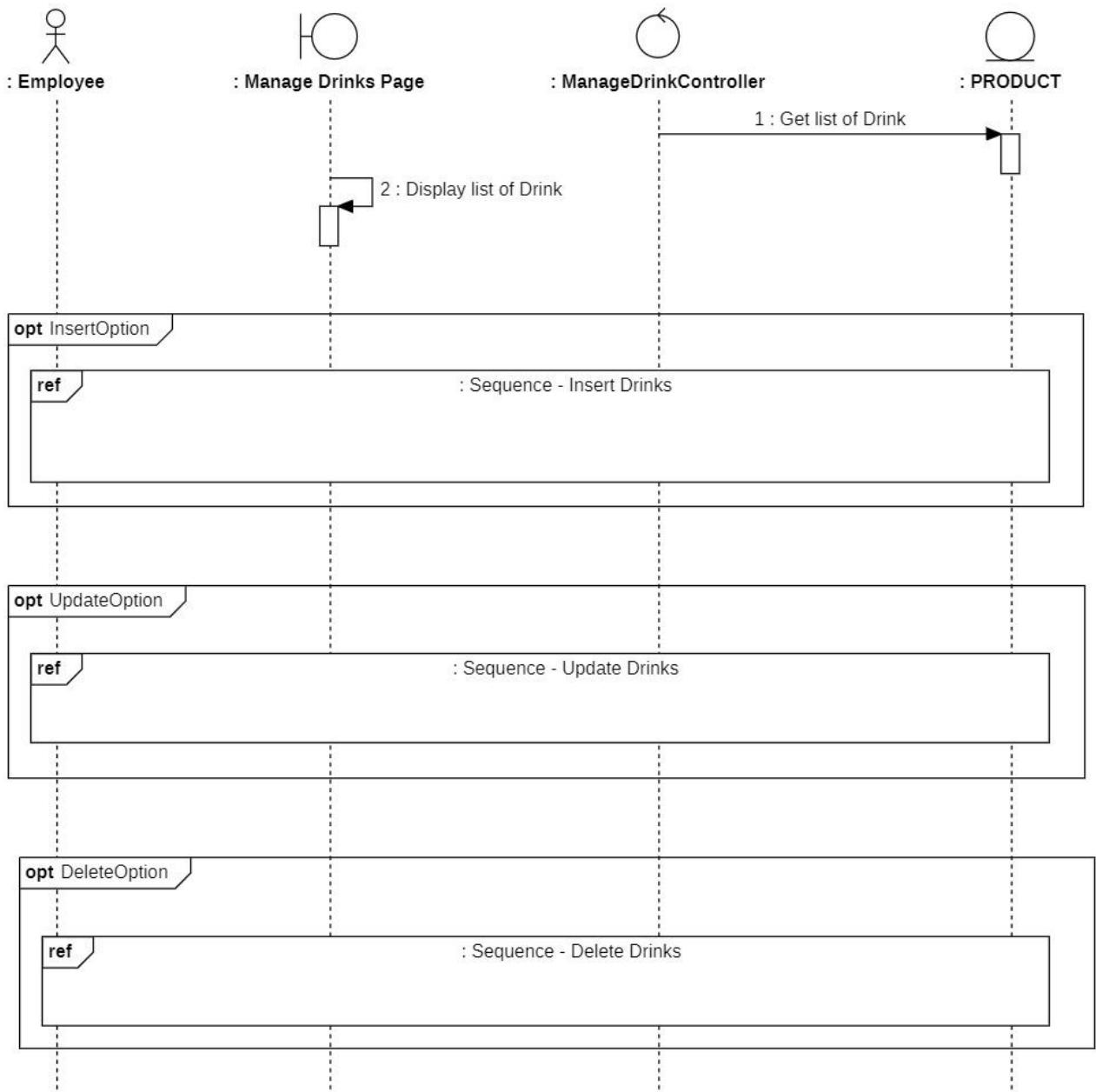


3.3.2.2b Class diagram

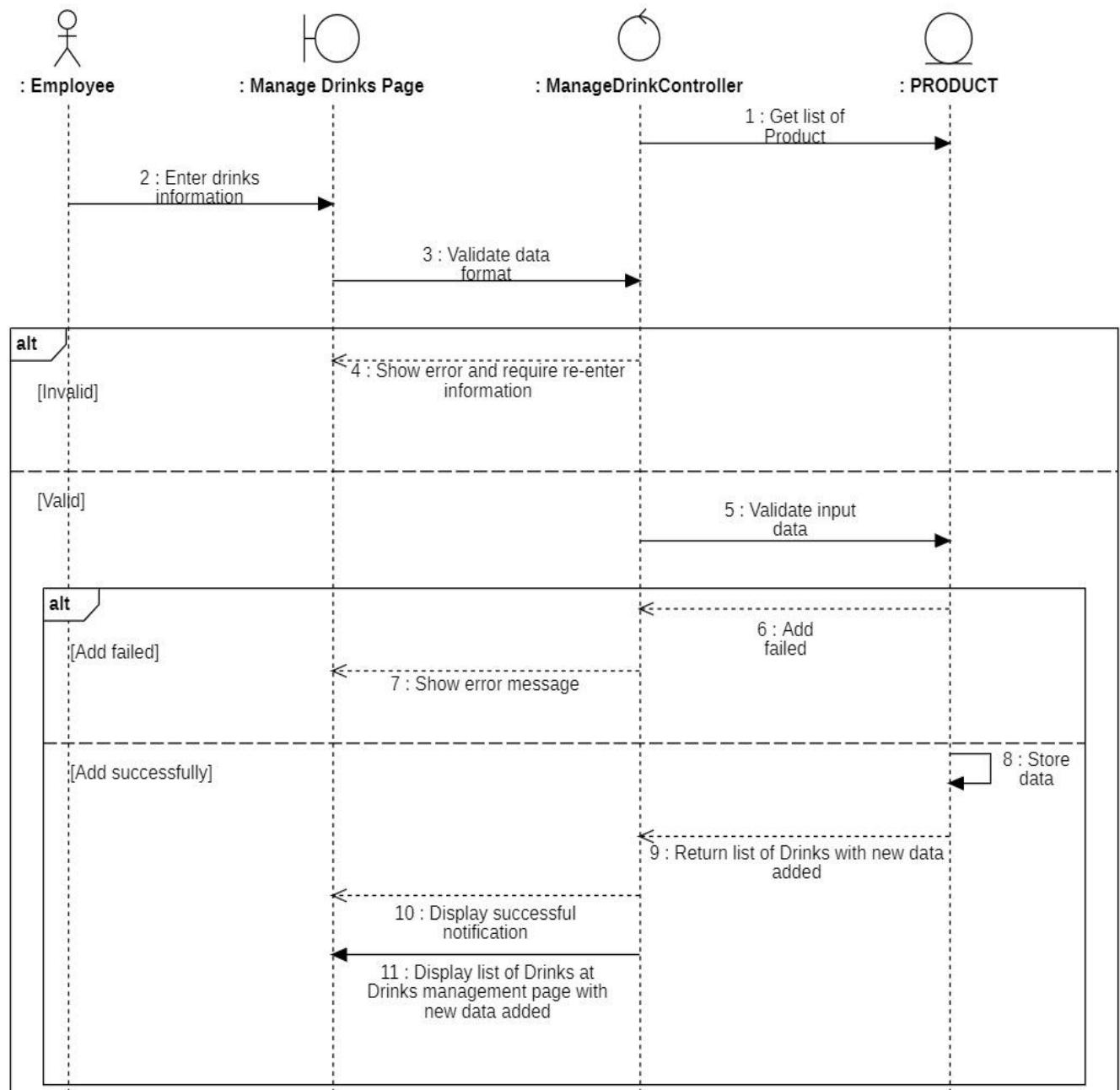


3.3.2.3 Manage drinks

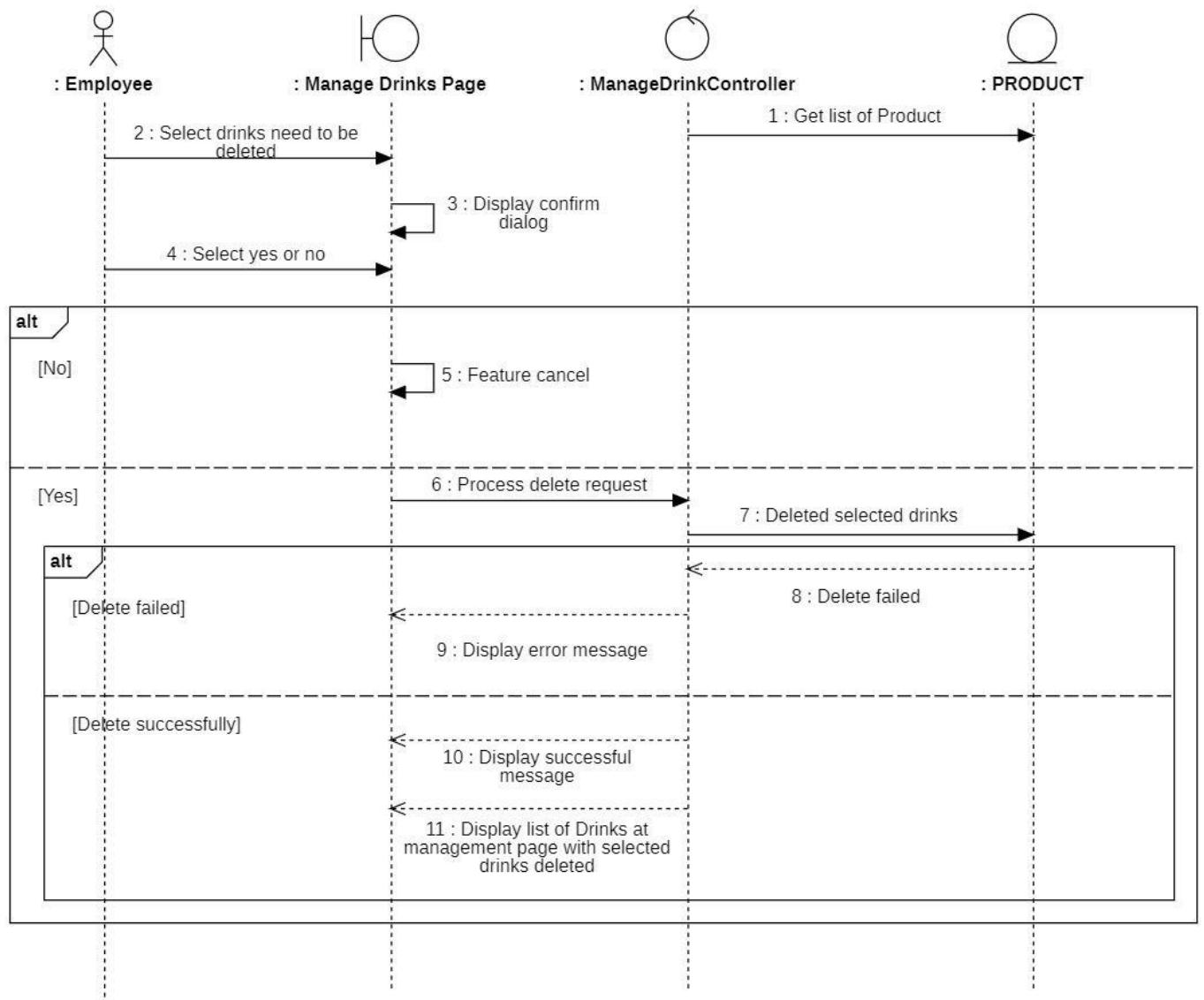
3.3.2.3a Sequence diagram



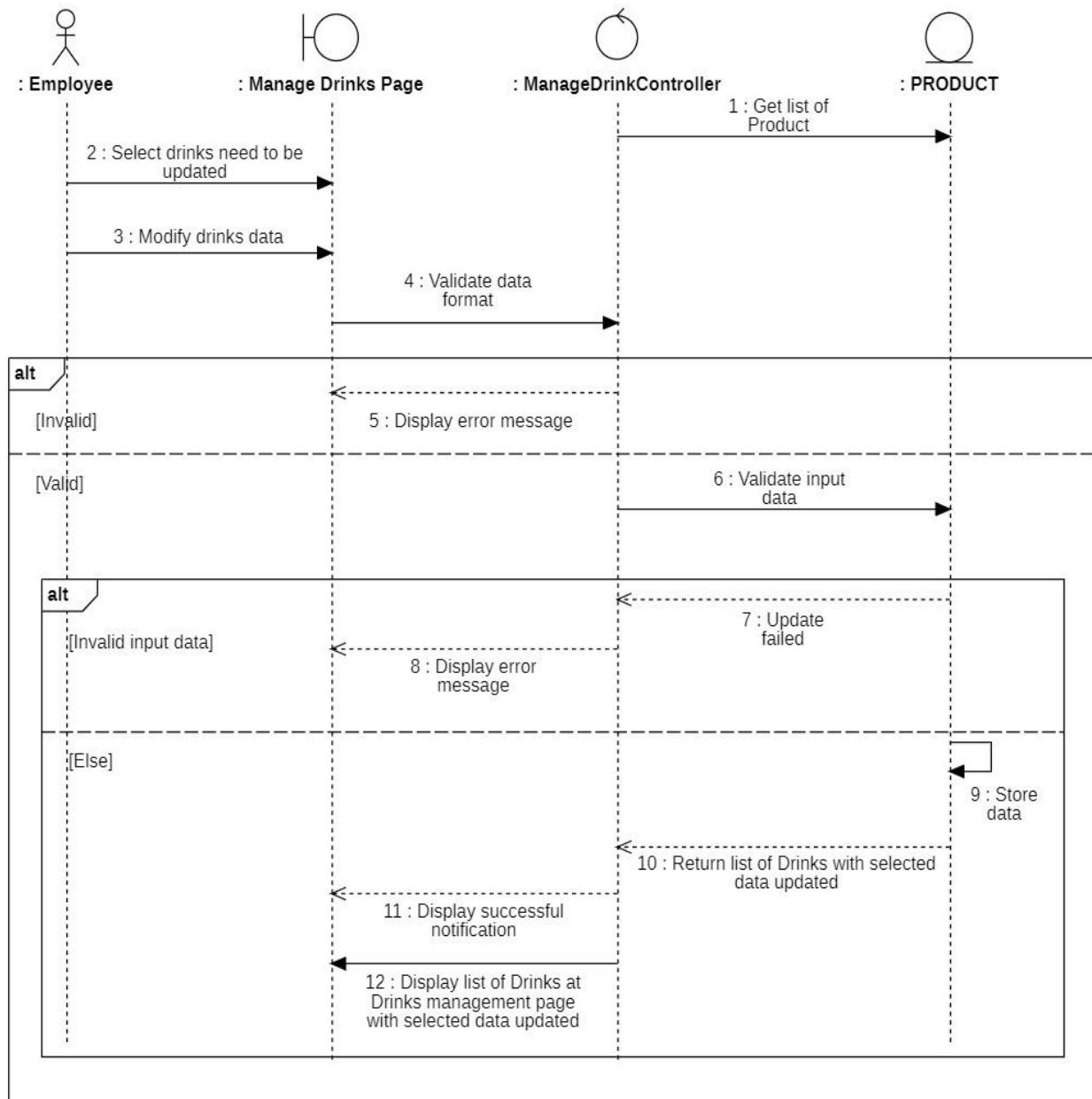
1. Add new drinks



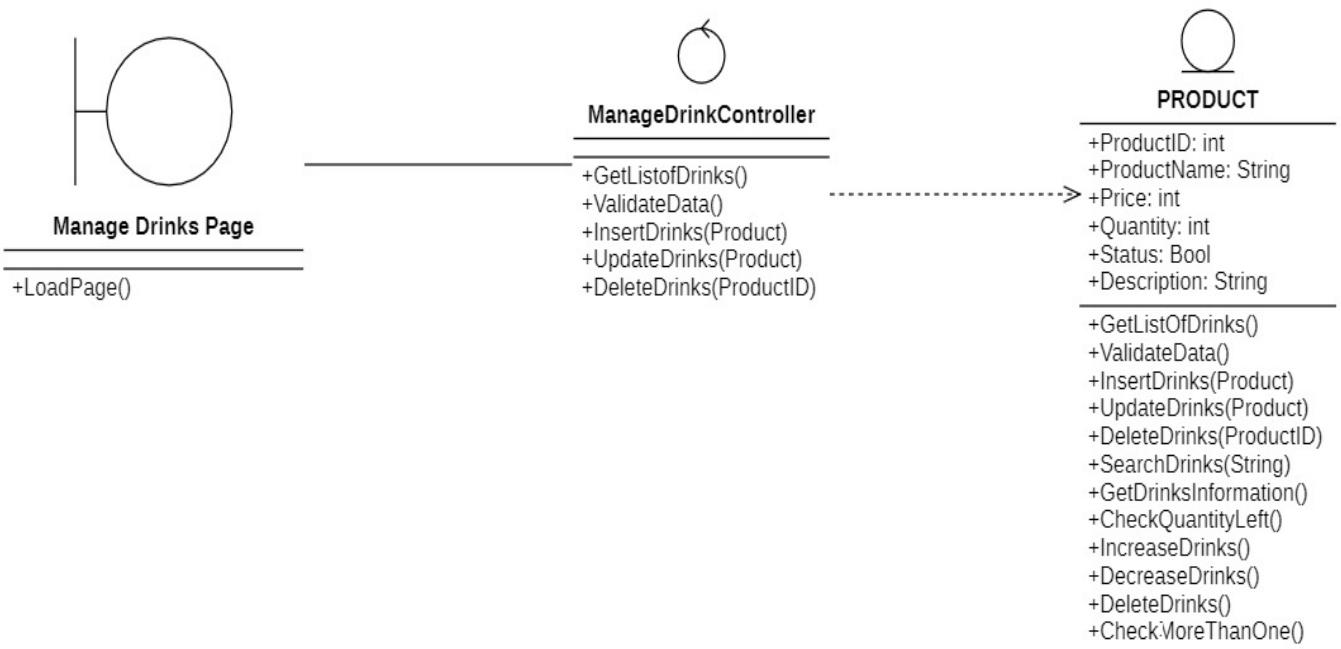
2. Delete drinks



3. Update drinks

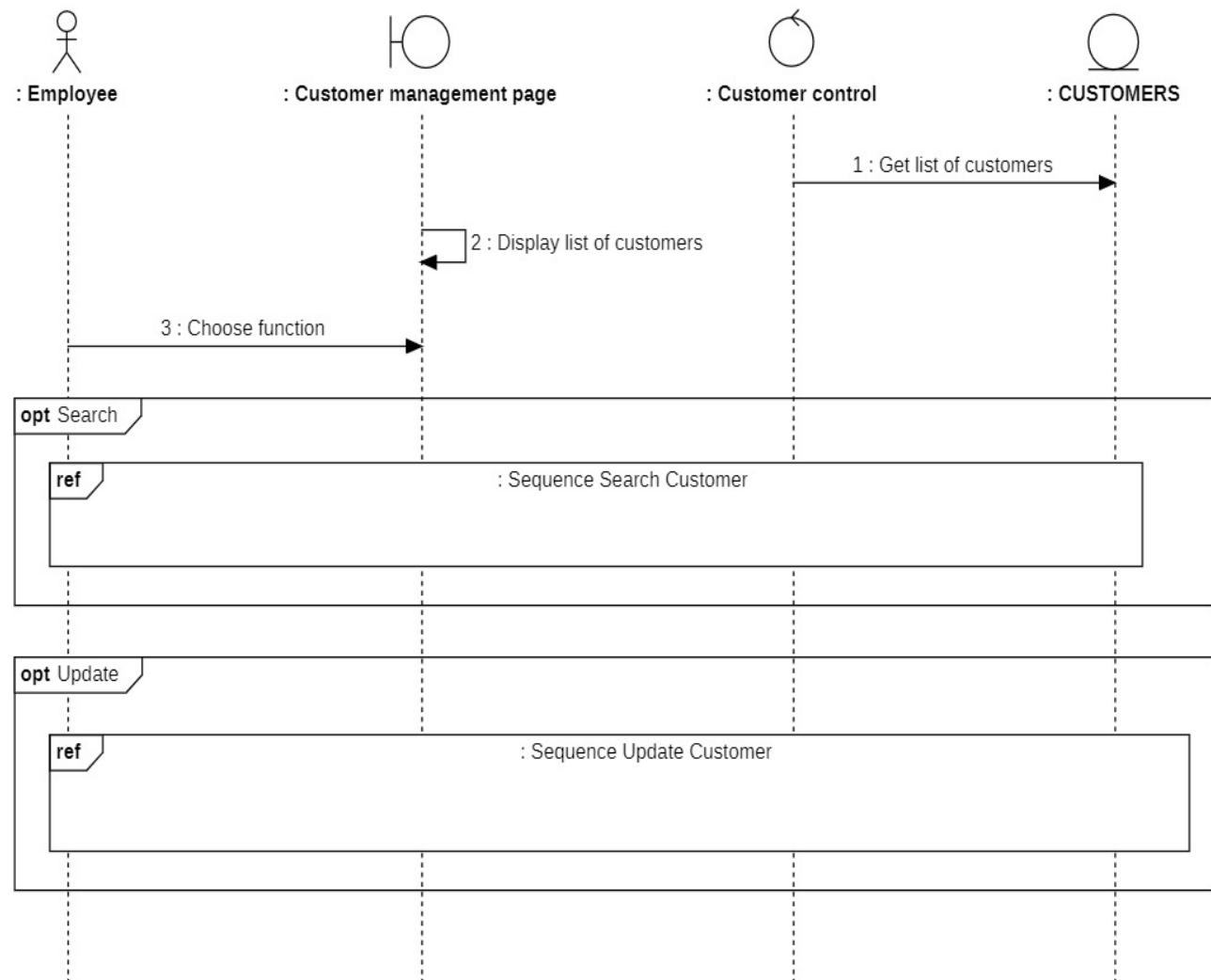


3.3.2.3b Class diagram

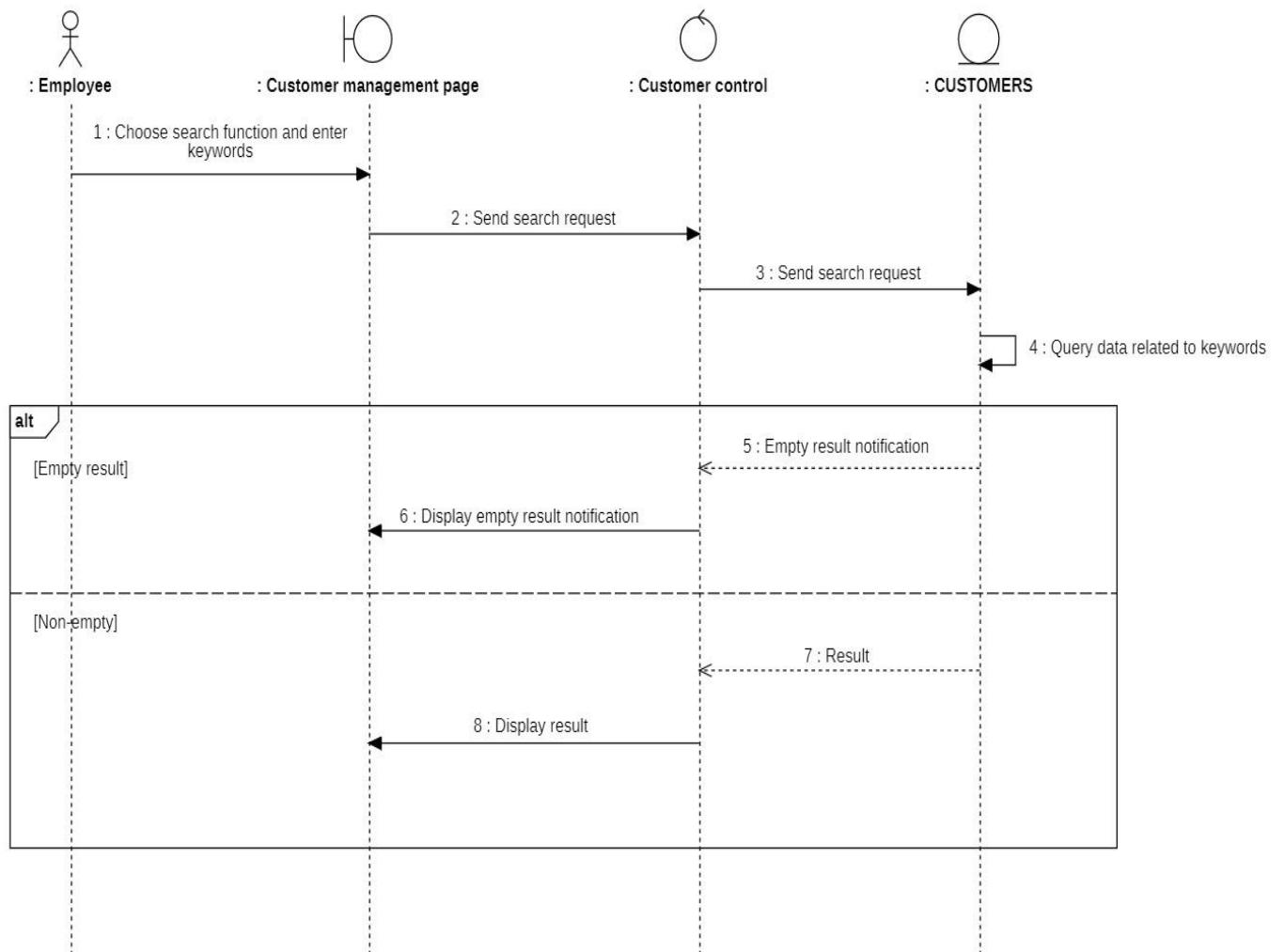


3.3.2.4 Manage customer

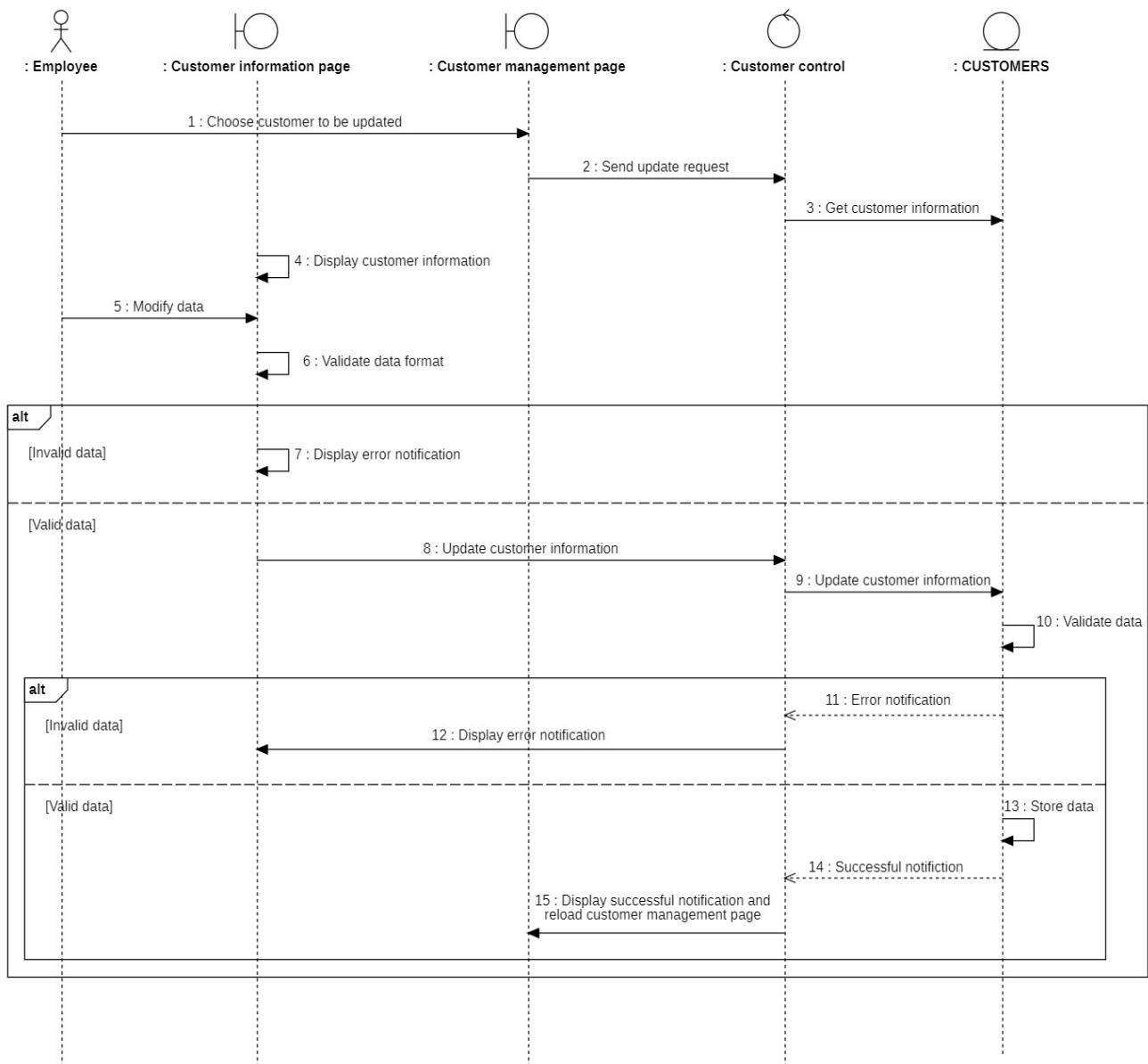
3.3.2.4a Sequence diagram



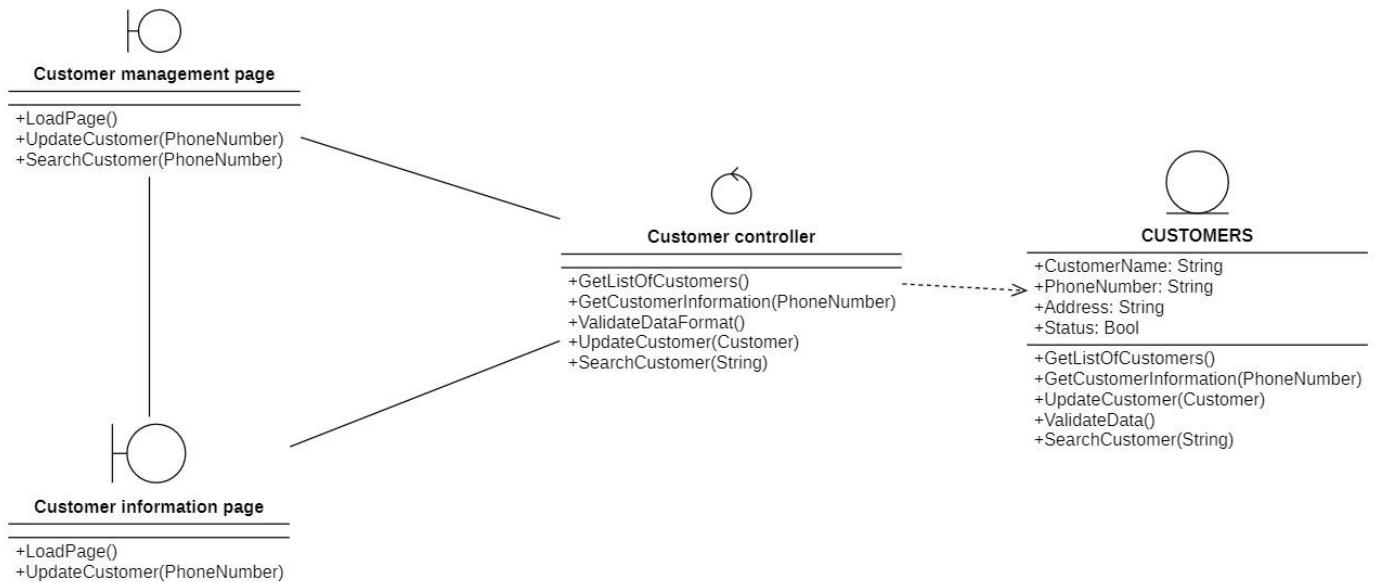
1. Search



2. Update

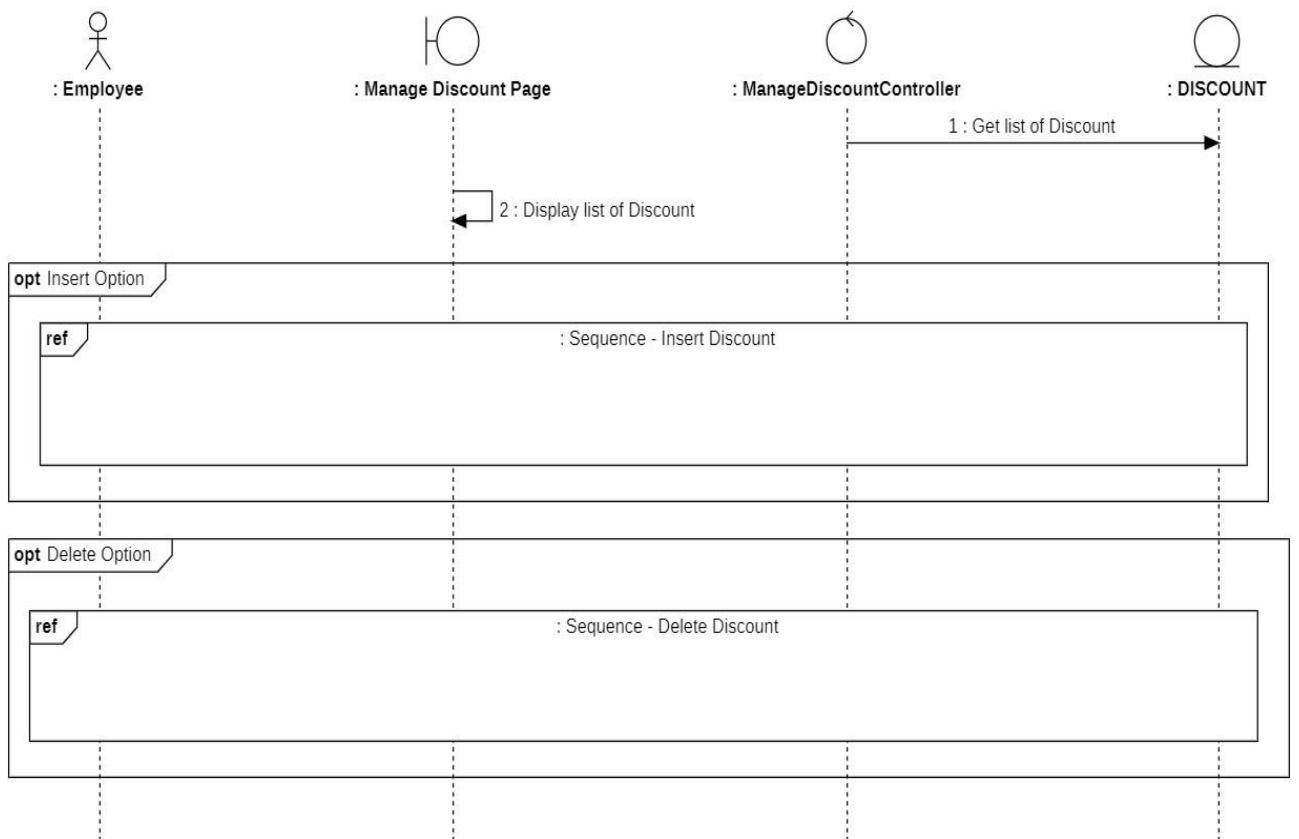


3.3.2.4b Class diagram

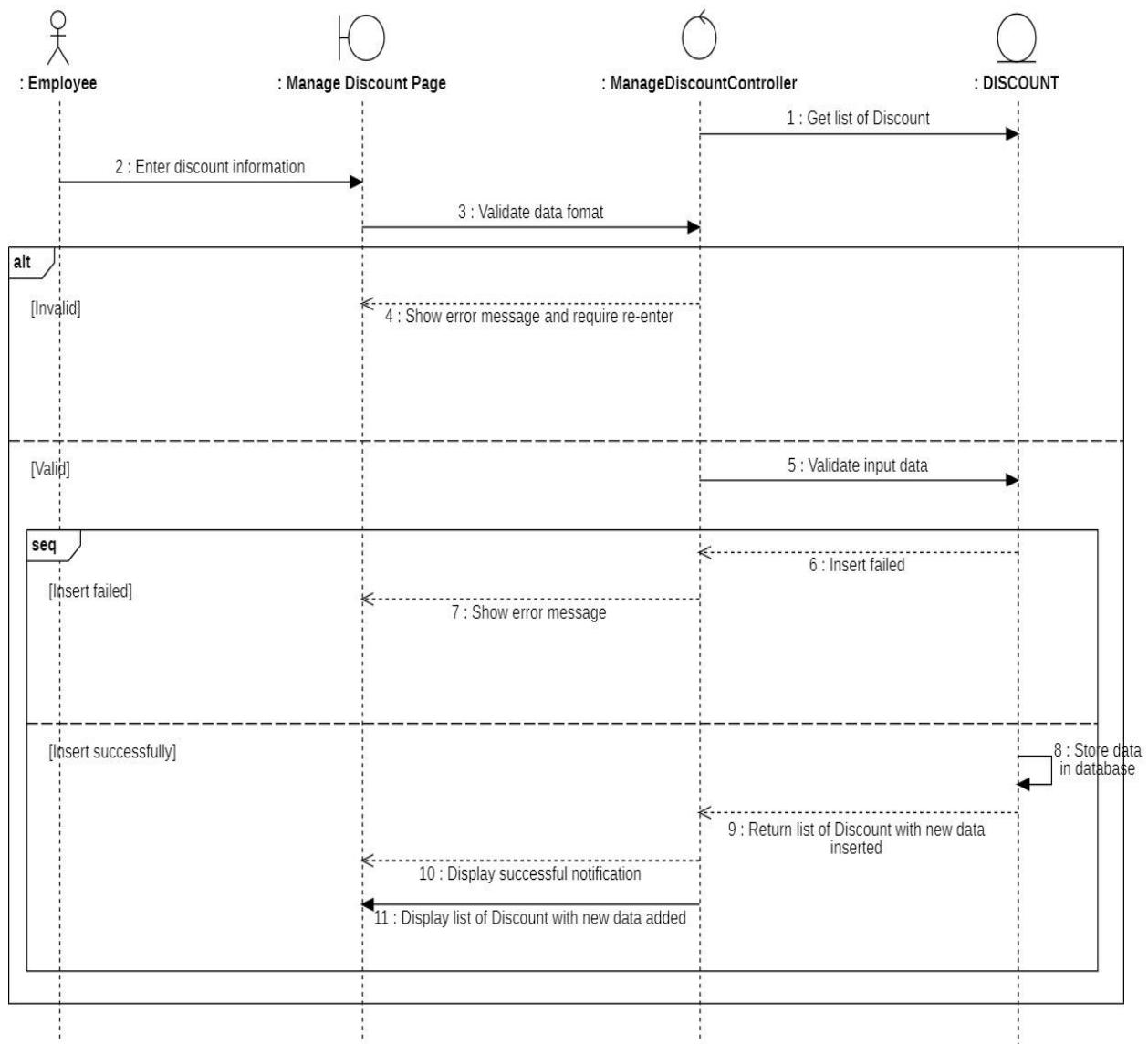


3.3.2.5 Manage discount

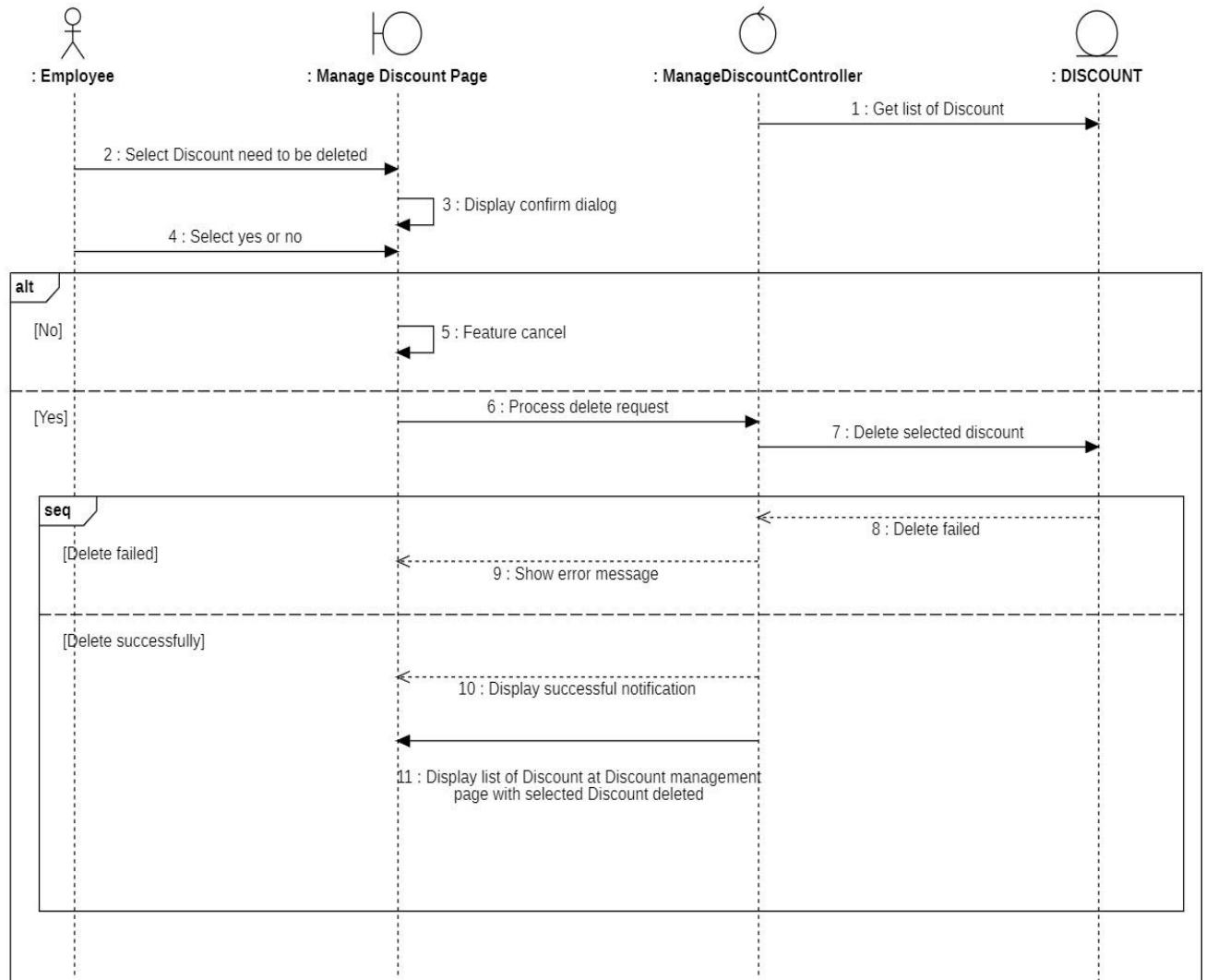
3.3.2.5a Sequence diagram



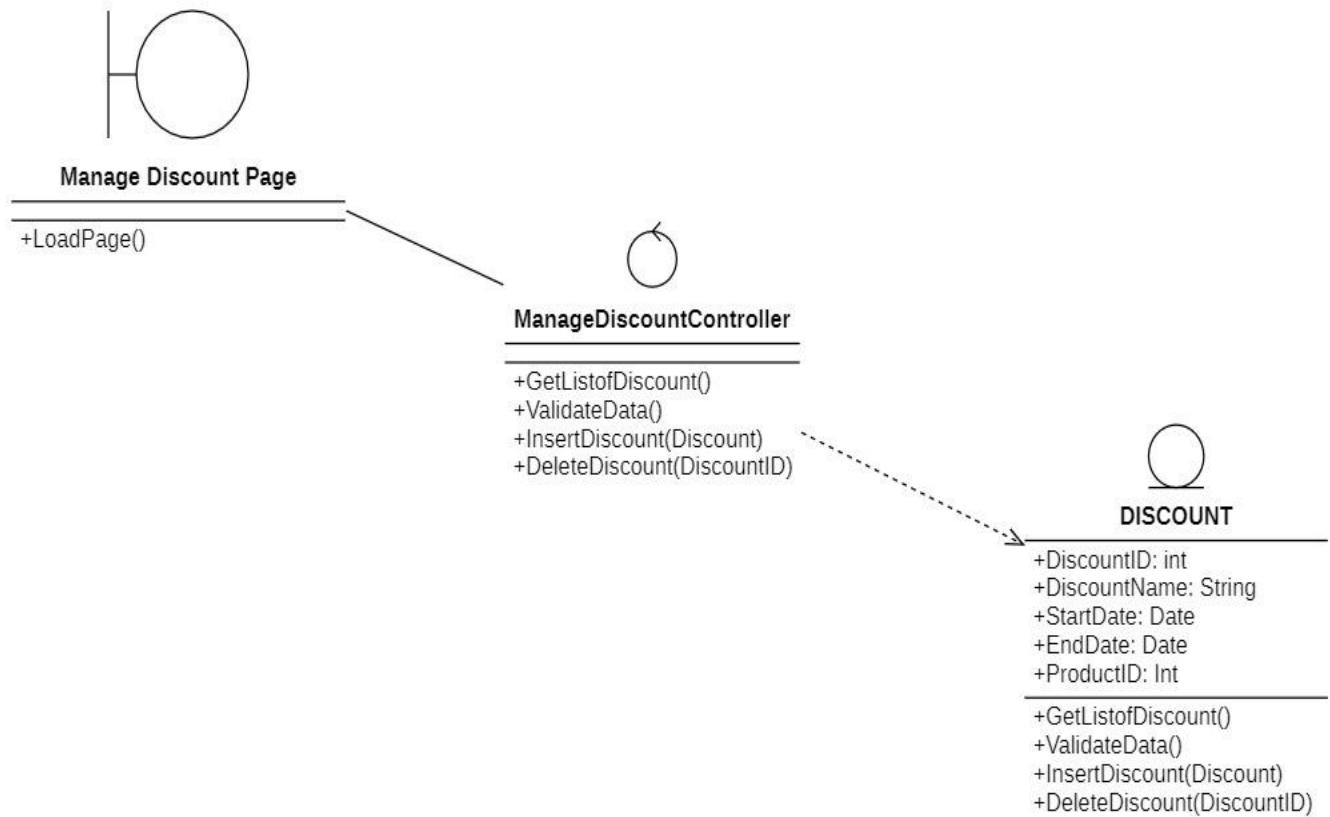
1. Insert discount



2. Delete discount



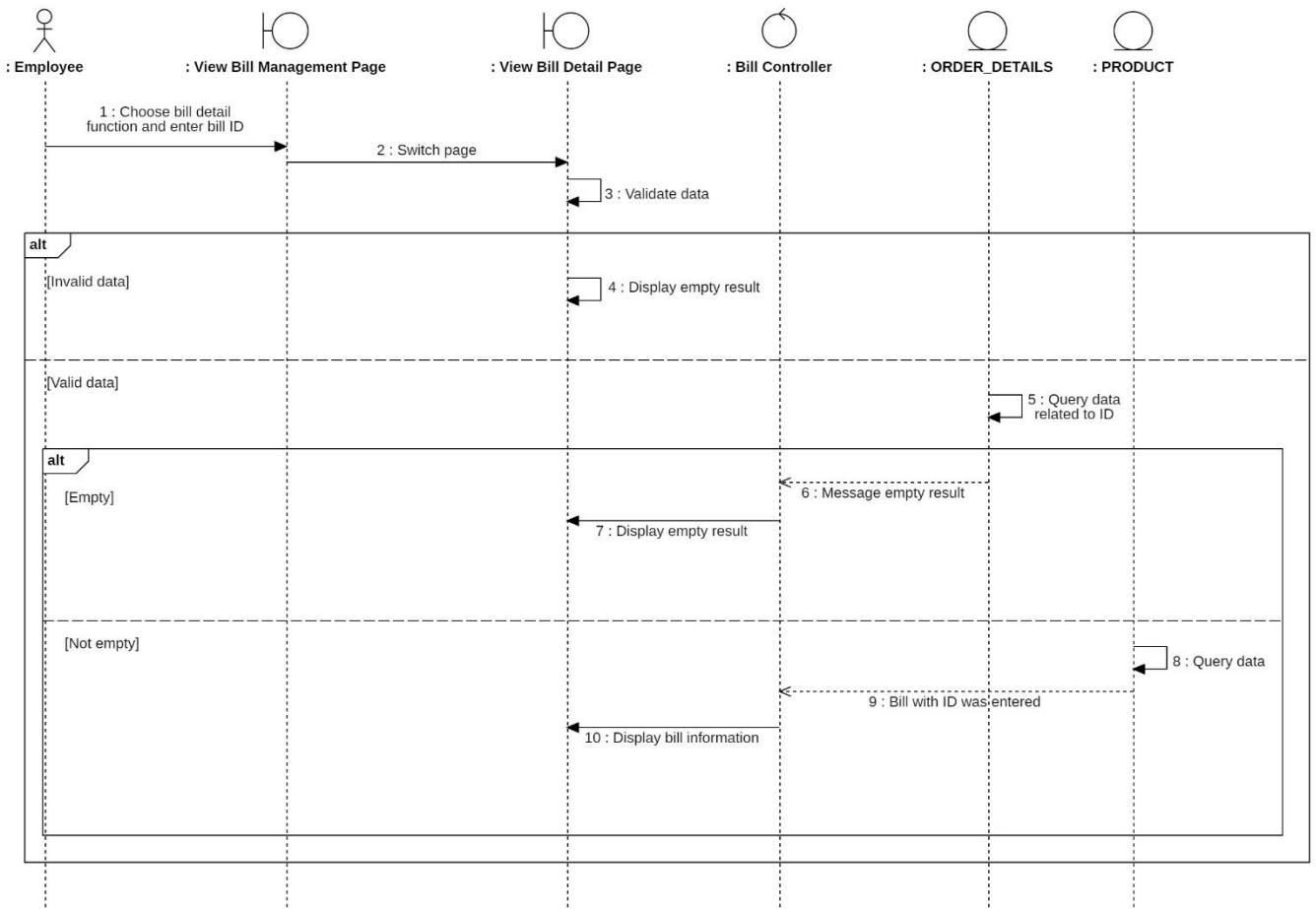
3.3.2.5b Class diagram



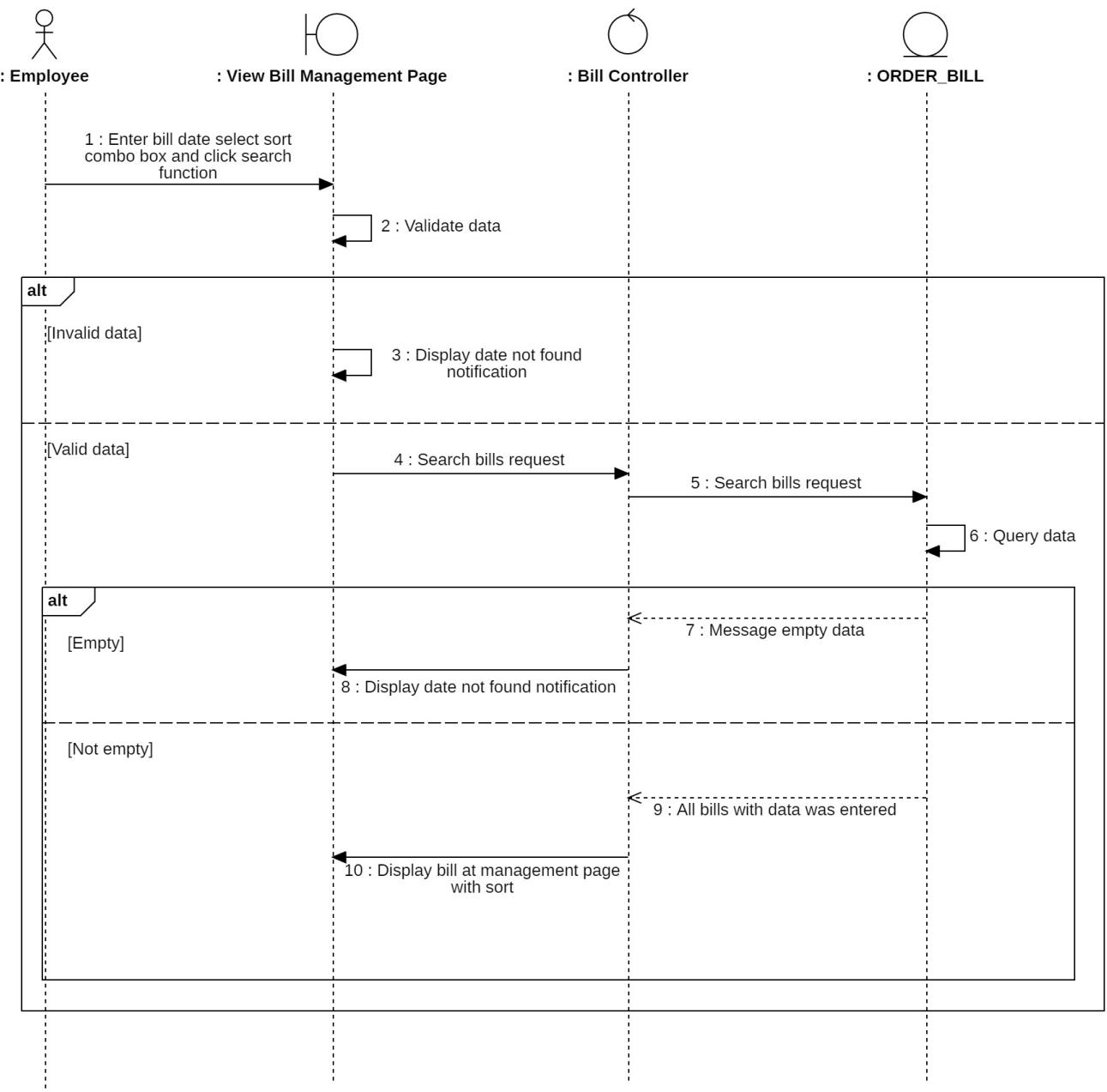
3.3.2.6 View bill

3.3.2.6a Sequence diagram

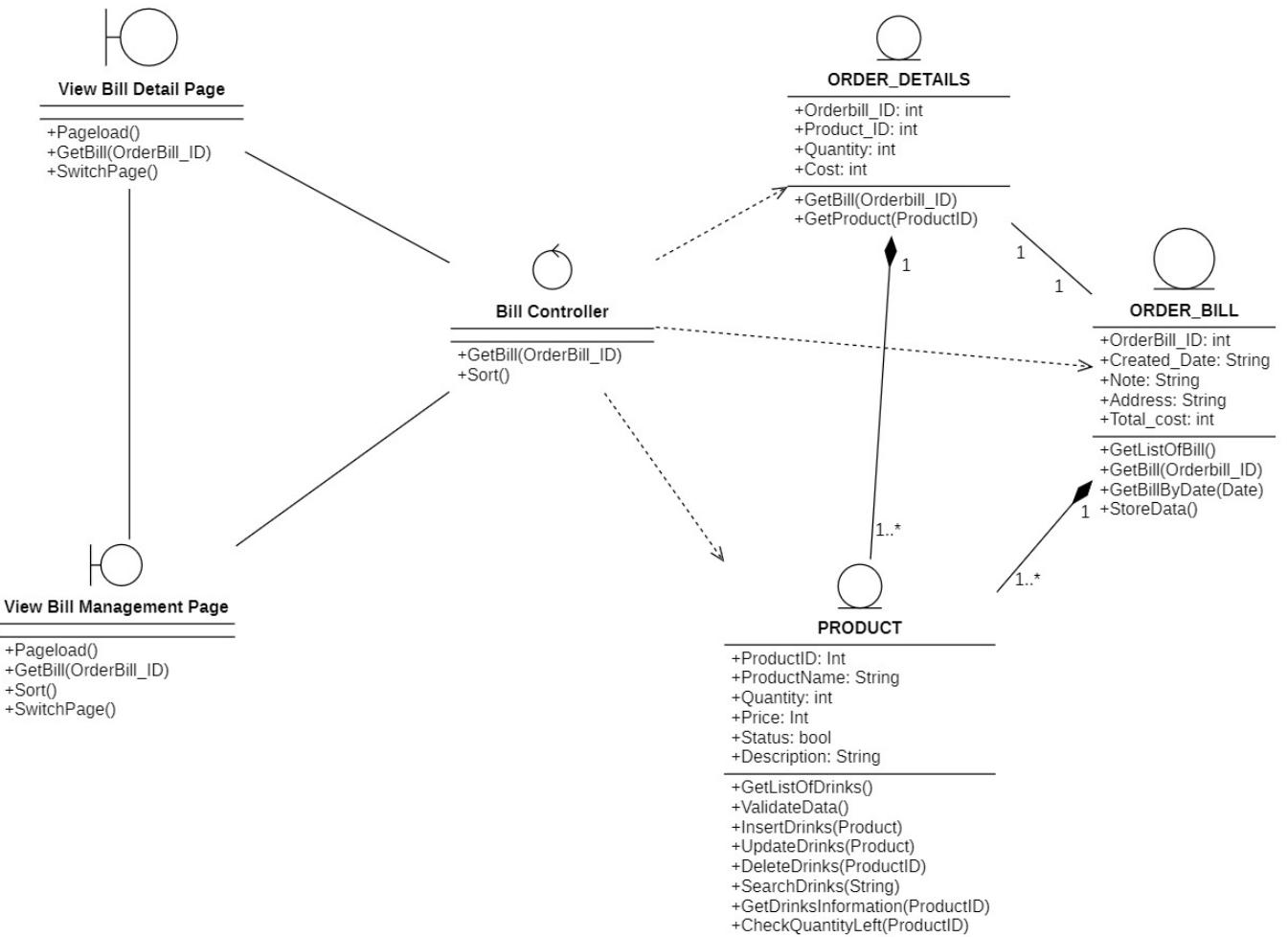
1. Bill detail



2. Search bill



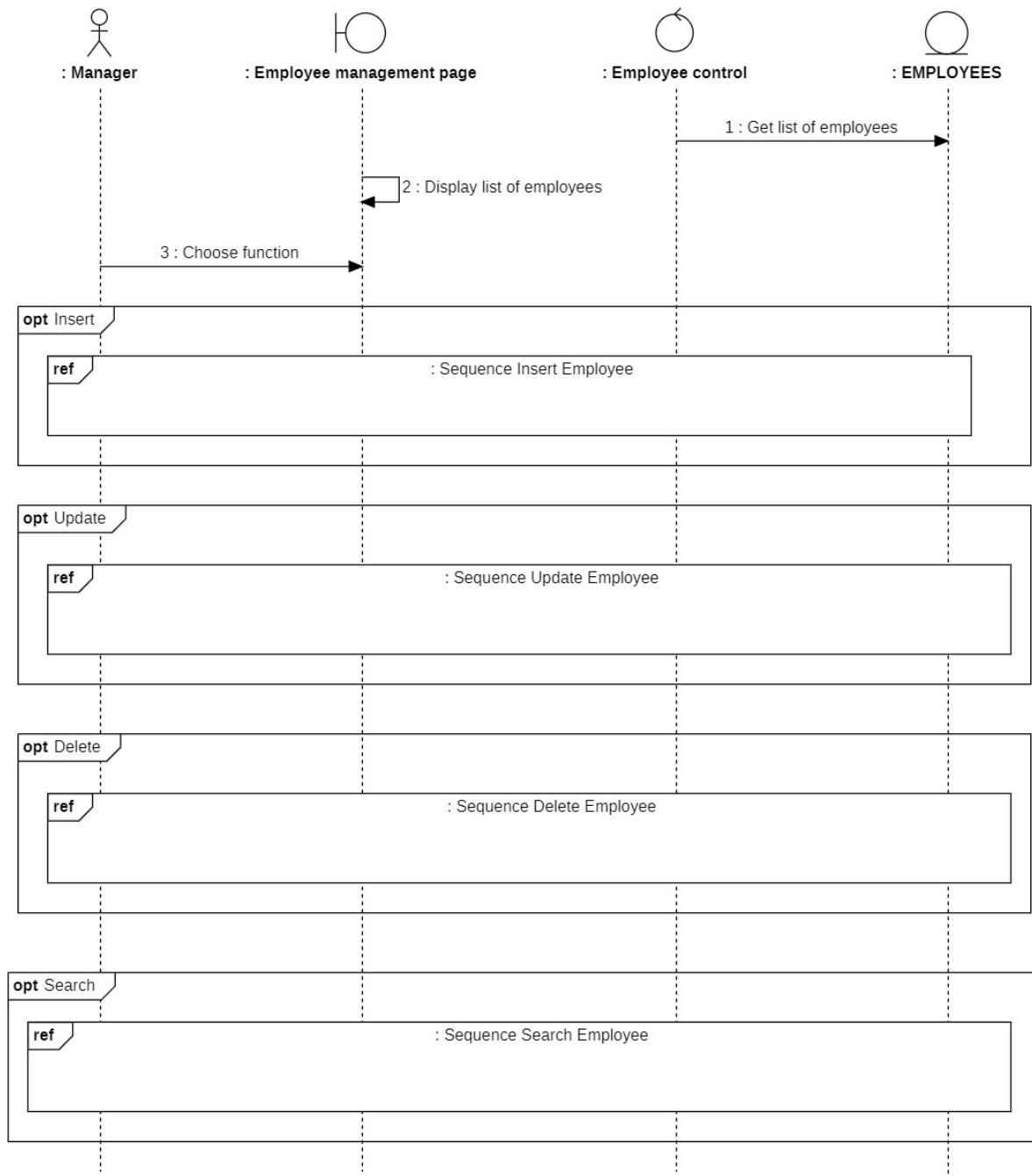
3.3.2.6b Class diagram



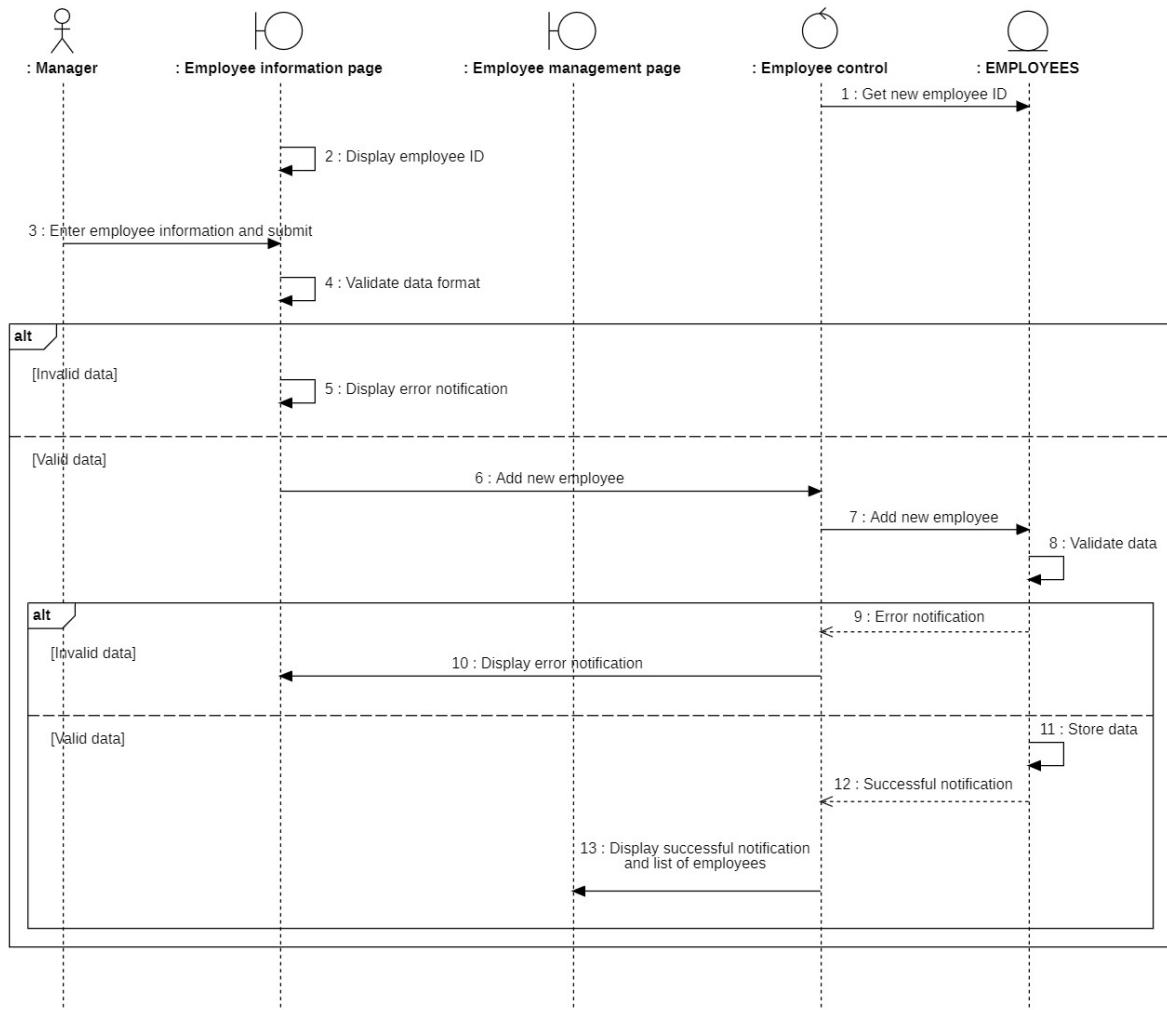
3.3.3 Sequence diagram and class diagram of Manager

3.3.3.1 Manage employee

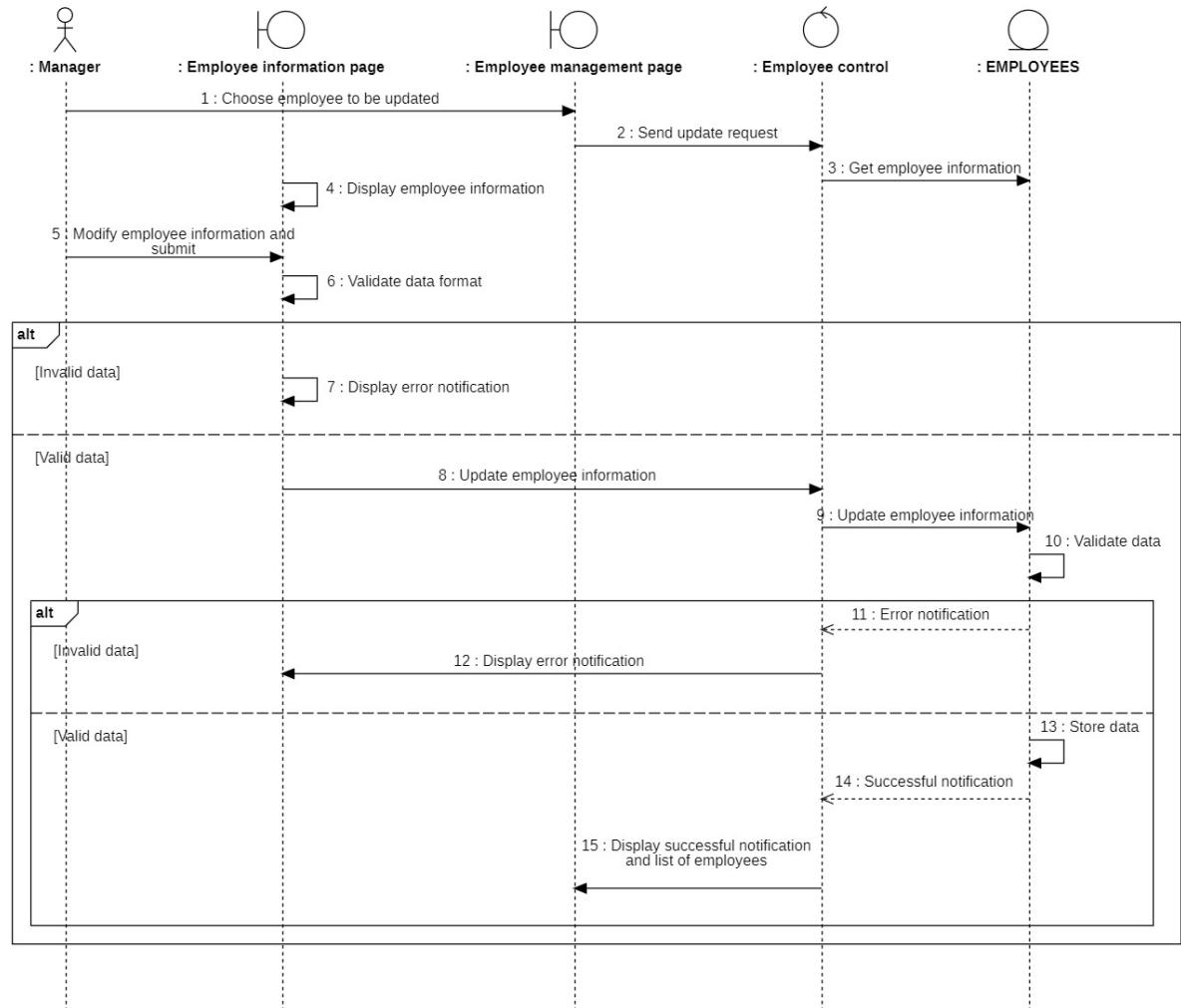
3.3.3.1a Sequence diagram



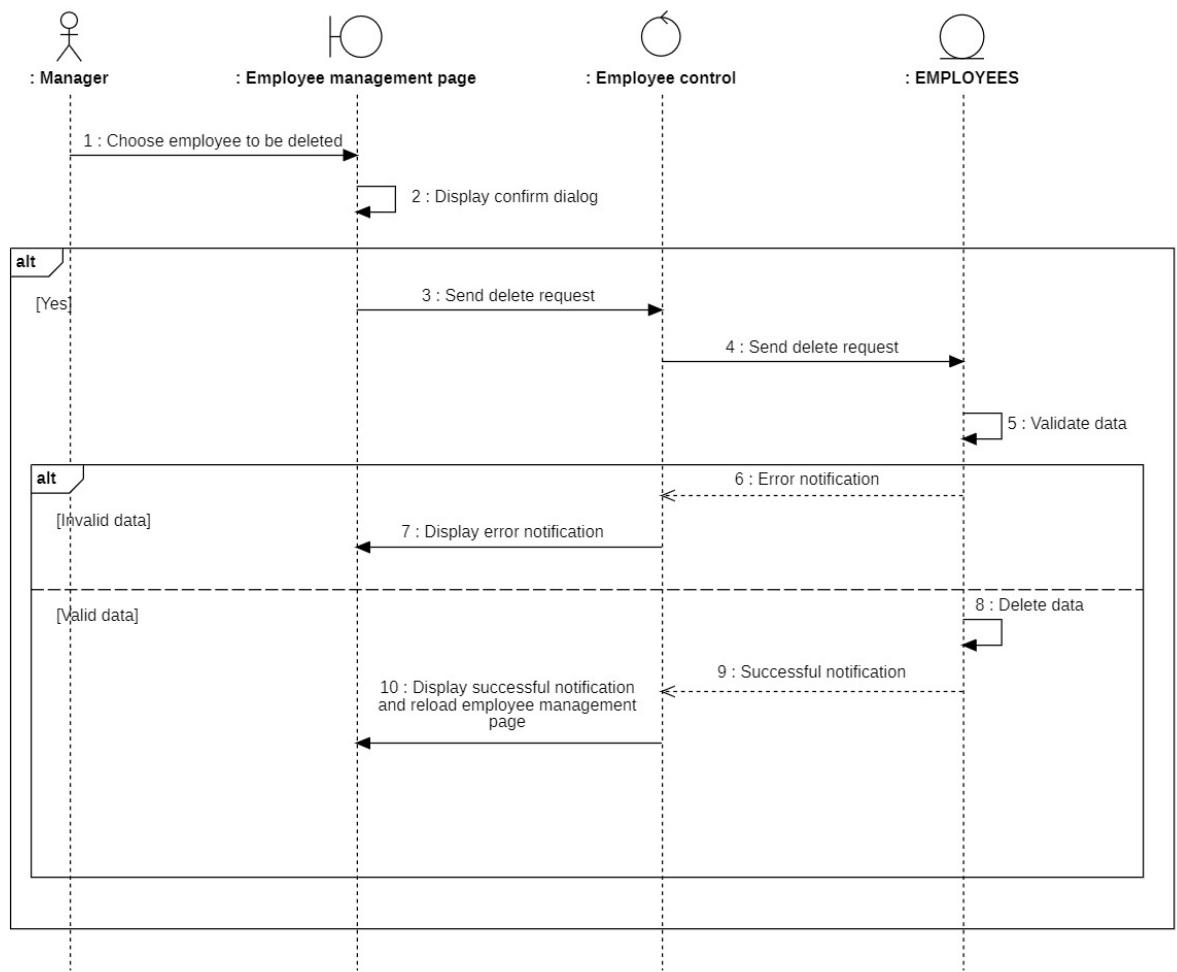
1. Insert



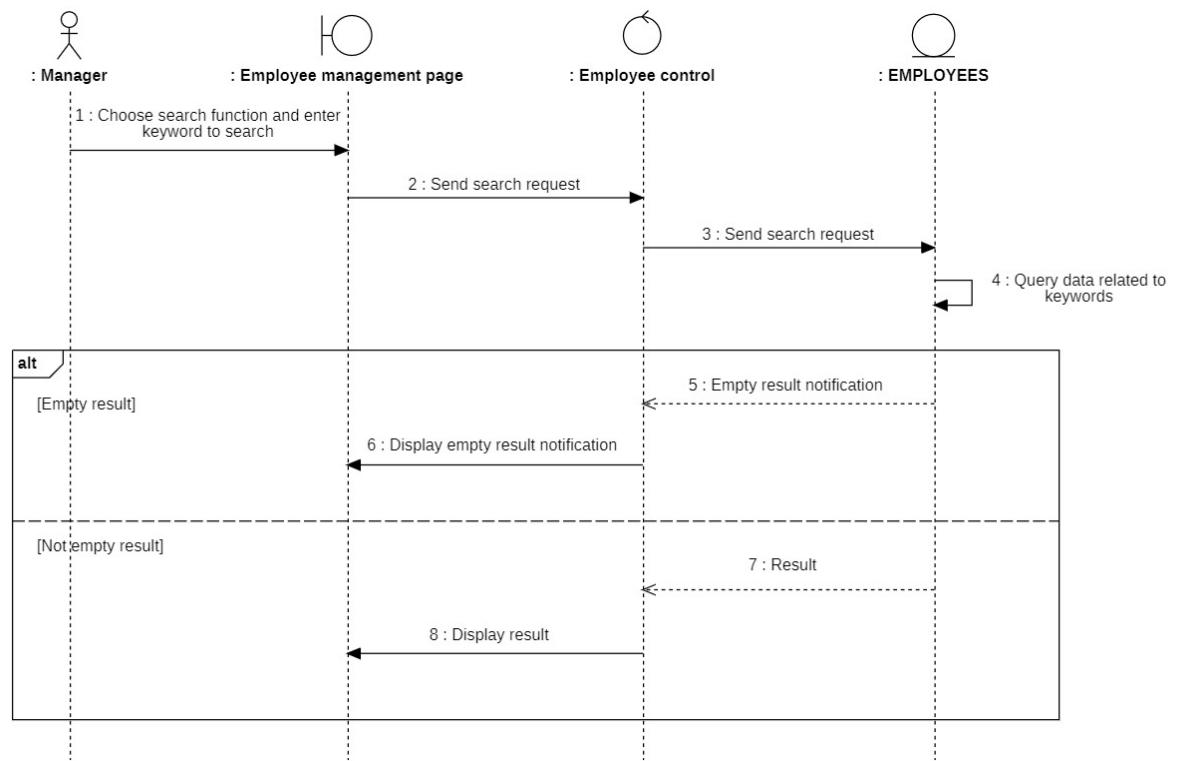
2. Update



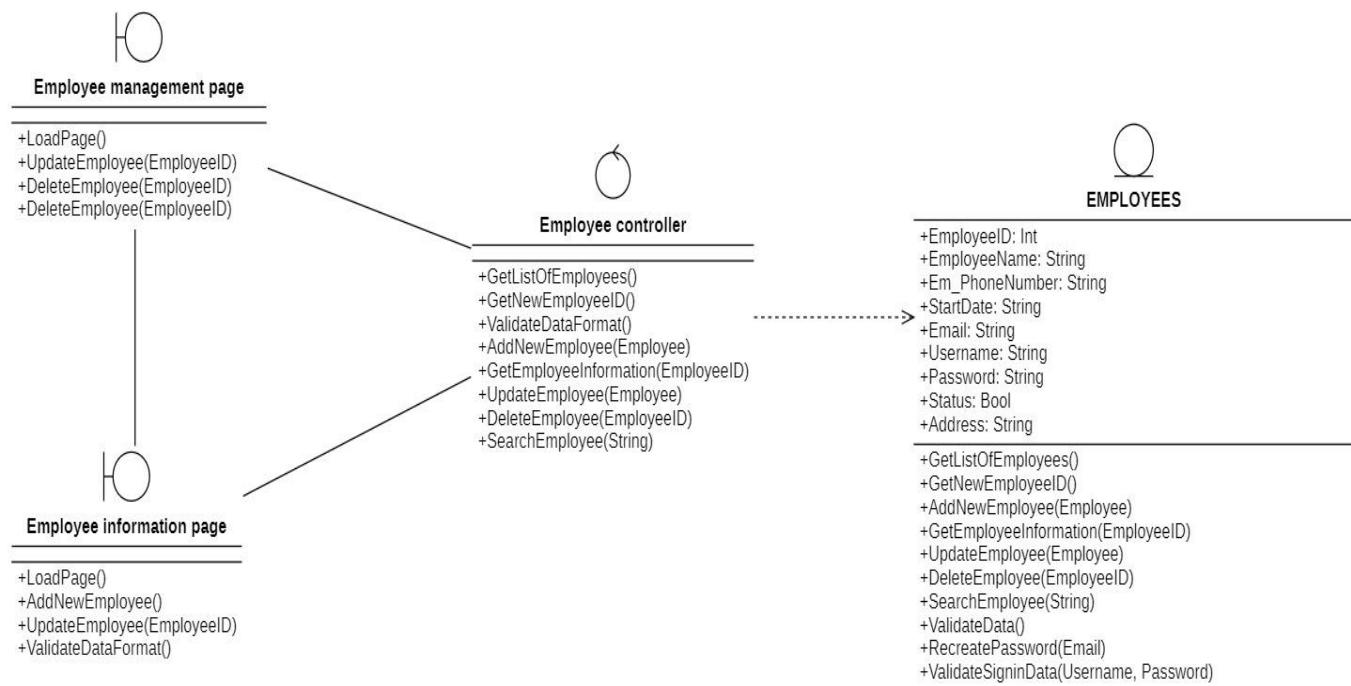
3. Delete



4. Search

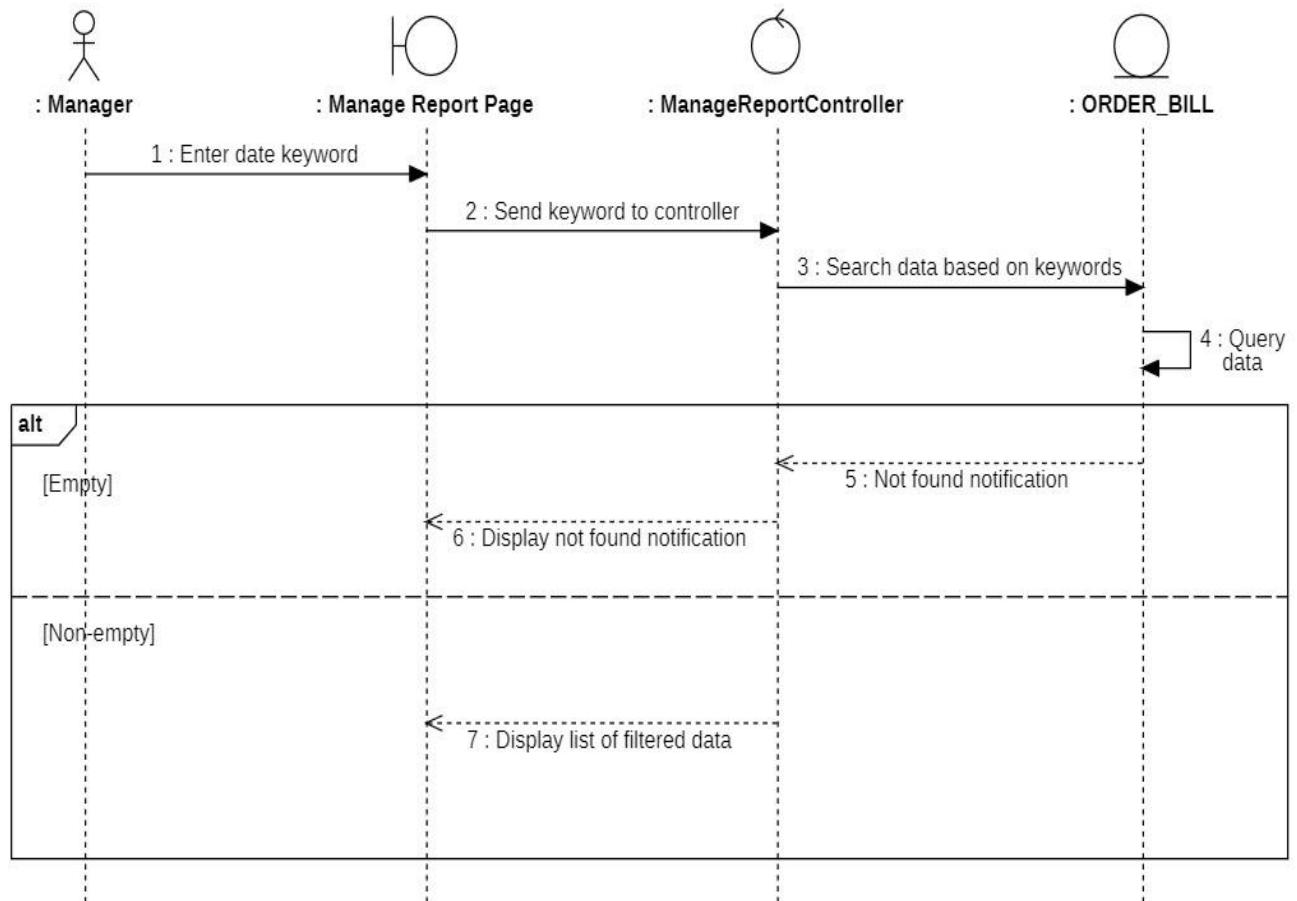


3.3.3.1b Class diagram

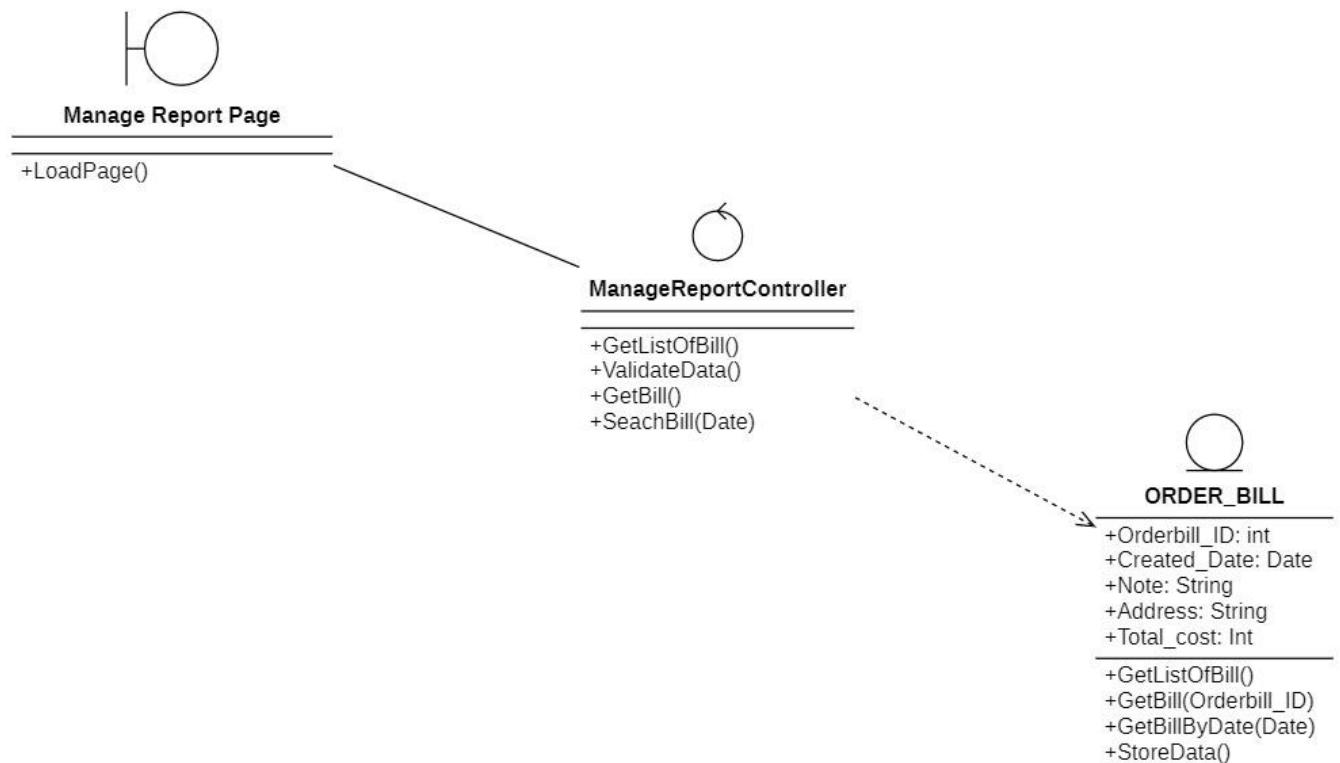


3.3.3.2 Report

3.3.3.2a Sequence diagram

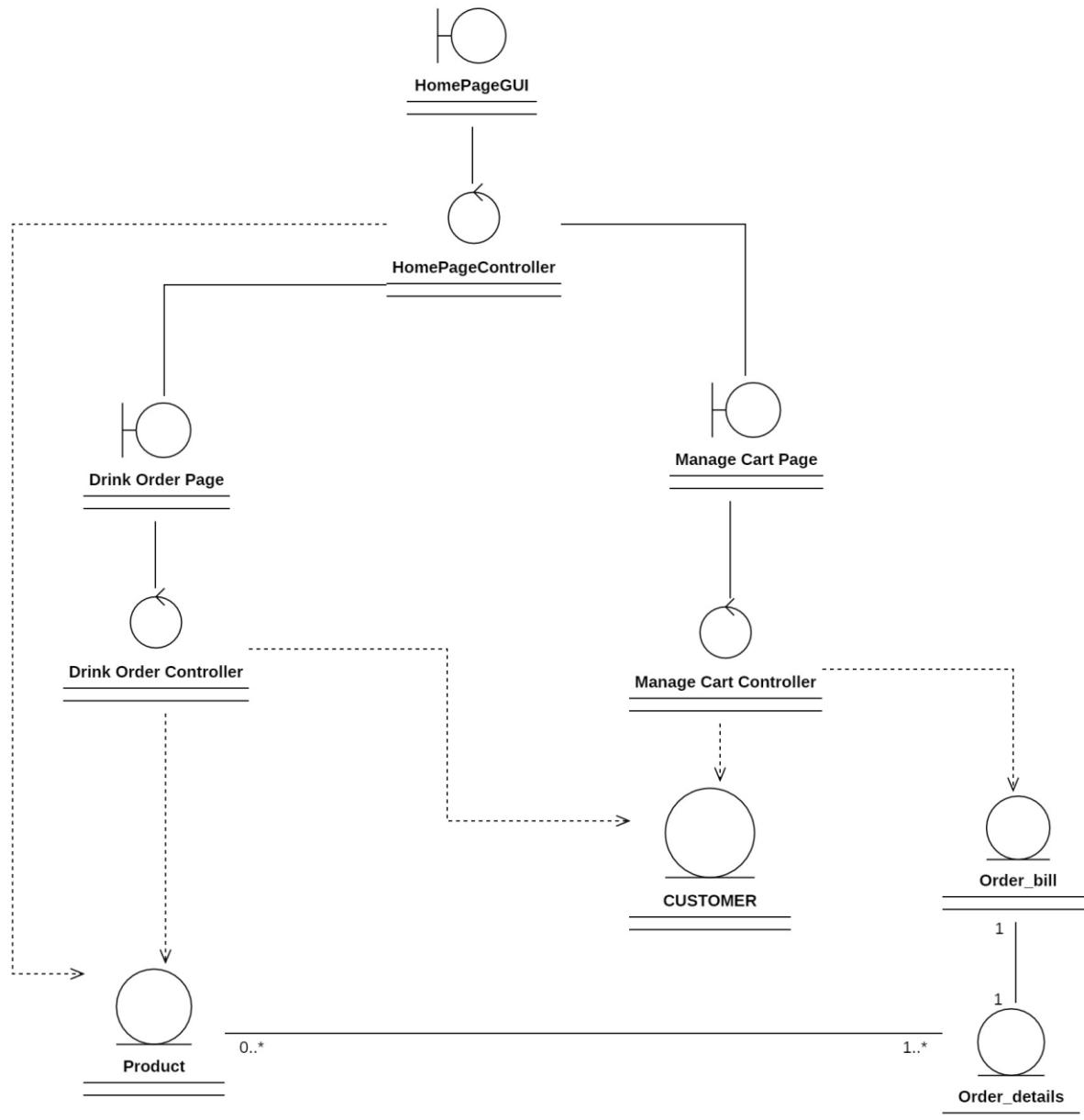


3.3.3.2b Class diagram

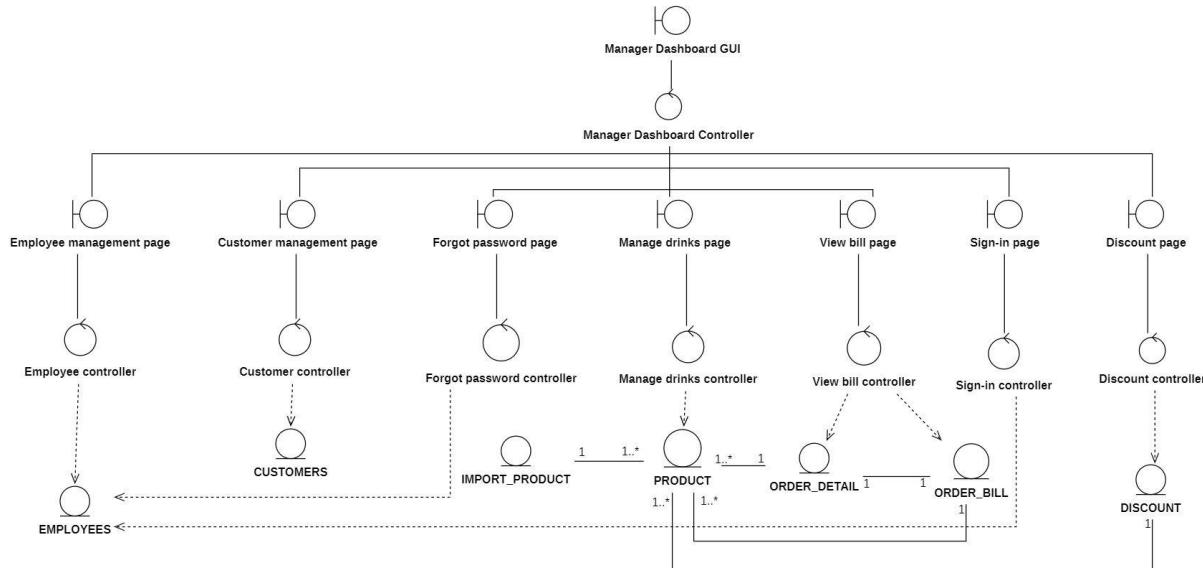


3.4 Class diagram and entity class diagram

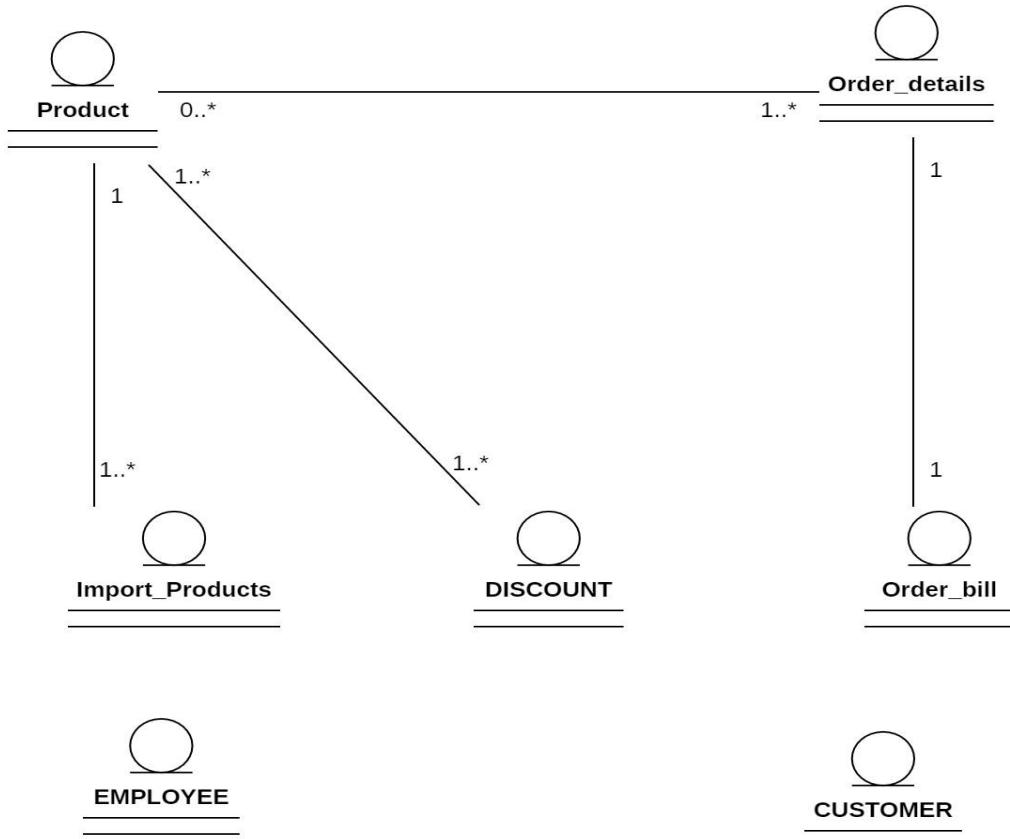
3.4.1 Customer overview class diagram



3.4.2 Manager overview class diagram



3.4.3 Entity class diagram



3.4.4 Class diagram specification

3.4.4.1 Customer

Customer		
Attribute	Variable	Meaning
	CustomerName: String	Full name of customer
	PhoneNumber: String	Phone number of customer
	Address: String	Address of customer
	Status: Bool	Status of customer (1: Active, 0: Inactive)
Operation	Function	Meaning
	GetListOfCustomer(): Array	Get list of customers to perform on the customer management page
	GetCustomerInformation(PhoneNumber):Array	Return a customer information matched with the PhoneNumber
	UpdateCustomer(Customer): Bool	Update information on an existing customer and return to a boolean variable
	ValidateData(): Bool	Validate if the input data is duplicated to the existing data
	SearchCustomer(String): Array	Search customers related to keyword for displaying on the

		customer management page

3.4.4.2 Employee

Employee		
Attribute	Variable	Meaning
	EmployeeID: Int	Employee ID generated by the system automatically
	EmployeeName: String	Full name of employee
	Em_PhoneNumber: String	Phone number of employee
	Email: String	Email of employee
	Username: String	Username of employee
	Password: String	Password of employee
	Address: String	Address of employee
	Status: Bool	Status of employee (1: Active, 0: Inactive)
Operation	Function	Meaning
	StoreSign-inStatus(): Bool	Change the sign-in status attribute
	GetListOfEmployees(): Array	Get list of employees to perform on the employee management page
	GetNewEmployeeID(): Int	Get new employee ID
	AddNewEmployee(Employee): void	Add new employee information in the database
	GetEmployeeInformation(EmployeeID): Employee	Return a employee information matched

		with the Employee_ID.
	UpdateEmployee(Employee): Bool	Update information on an existing employee and return to a boolean variable
	DeleteEmployee(EmployeeID): Bool	Delete a employee existed in database and return to a boolean variable
	SearchEmployee(String): Array	Search employees related to keyword for displaying on the employee management page
	ValidateData(): Bool	Validate if the input data is duplicate to the existed data
	RecreatePassword>Email): void	Generate new password and send it to employee email
	ValidateSigninDataUsername, Password): Bool	Validate if the sign in data is correct with database.

3.4.4.3 Product

Product		
Attribute	Variable	Meaning
	ProductID: Int	Product ID generated by system automatically
	ProductName: String	Name of product
	Quantity: Int	Quantity left of product
	Price: Int	Price of 1 product
	Status: Bool	Status of product (1: Active, 0: Inactive)
	Description: String	Description of product
Operation	Function	Meaning
	GetListOfDrinks(): Array	Get list of drinks to perform on the order place page and manage drinks page
	ValidateData(): Bool	Validate if the input data is duplicate to the existed data
	InsertDrinks(Product): Bool	Create a new drink in the database and return to a boolean variable
	UpdateDrinks(Product): Bool	Update information on an existing product and return to a boolean variable
	DeleteDrinks(ProductID): Bool	Delete information on an existing product and return

		to a boolean variable
	SearchDrinks(String): Array	Search drinks related to keyword for displaying on the order page
	GetDrinksInformation(ProductID): Product	Return a drinks information matched with the Product_ID.
	CheckQuantityLeft(ProductID): Bool	Check the number of product left in the database and return to boolean variable.
	IncreaseDrinks(): Bool	Increase the number of product in the cart and return to boolean variable
	DecreseDrinks(): Bool	Decrease the number of product in the cart and return to boolean variable
	DeleteDrinks(): Bool	Delete product in the cart and return to boolean variable.
	CheckMoreThanOne(): Bool	Validate if the quantity of product more than 1

3.4.4.4 Order bill

Order_Bill		
Attribute	Variable	Meaning
	Orderbill_ID: Int	Order bill id generated by system automatically
	Create_Date: Date	The date that bill was create
	Note: String	Note of bill
	Address: String	Address of customer
	Total_cost: int	Total cost of bill
Operation	Fuction	Meaning
	GetListOfBill(): Array	Get list of bill to perform on the view bill page
	GetBill(Orderbill_ID): Bill	Return a bill information matched with the Orderbill_ID.
	GetBillByDate(Date): Array	Get list of bill has Create_Date = Date
	StoreData(): Bool	Store data in database

3.4.4.5 Discount

Discount		
Attribute	Variable	Meaning
	DiscountID: Int	Discount id generated by system automatically
	DiscountName: String	Name of discount
	StartDate: Date	Discount start date
	EndDate: Date	Discount end date
	ProductID: Int	Product id generated by system automatically
Operation	Fuction	Meaning
	GetListOfDiscount(): Array	Get list of discount to perform on the manage discount page

	ValidateData(): Bool	Validate if the input data is duplicate to the existed data
	InsertDiscount(Discount): Bool	Create a new discount in the database and return to a boolean variable
	DeleteDiscount(DiscountID): Bool	Delete information on an existing discount and return to a boolean variable

3.4.4.6 Order detail

Order detail		
Attribue	Variable	Meaning
	Orderbill_ID: Int	Order bill id generated by system automatically
	Product_ID: Int	Product id generated by system automatically
	Quantity: Int	The quantity of product
	Cost: Int	The Cost of product
Operation	Function	Meaning
	GetBill(Orderbill_ID): Bill	Return a bill information matched with the Orderbill_ID.
	GetProduct(ProductID): Product	Return a product information matched with the Product_ID.

3.4.4.7 Sign-in controller

Sign-in controller		
Operation	Fuction	Meaning
	ValidateDateFormat(): Bool	Validate if the input information format is correct
	Sign-in.Username, Password): Bool	Request to sign-in to the database

3.4.4.8 Forgot password controller

Forgot password controller		
Operation	Function	Meaning
	ValidateDataFormat(): Bool	Validate if the input information format is correct
	SendValidateDataRequest(): void	Send validate data request to database
	SendNewPasswordToEmail(): void	Send new password to employee's email.

3.4.4.9 Customer controller

Customer controller		
Operation	Function	Meaning
	GetListOfCustomer(): Array	Request database to get the list of customer information to display on the GUI
	GetCustomerInformation(PhoneNumber): Customer	Request database to get the customer information selected by employee/manger
	UpdateCustomer(Customer): Bool	Request database to make a change to an existed customer
	ValidateDataFormat(): Bool	Validate if the input information format is correct
	SearchCustomer(String): Array	Request database to get a list of customer related to the string parameter

3.4.4.10 Manage drinks controller

Manage drinks controller		
Operation	Function	Meaning
	GetListOfDrinks(): Array	Request database to get the list of drinks information to display on the GUI
	ValidateData(): Bool	Validate if the input information format is correct
	InsertDrinks(Product): Bool	Request database to create a new drinks
	UpdateDrinks(Product): Bool	Request database to make a change to an existed drinks
	DeleteDrinks(ProductID): Bool	Request database to delete the selected drinks
	SearchDrinks(String): Array	Request database to get a list of drinks related to the string parameter
	CheckQuantityLeft(ProductID): Bool	Request database Check the number of product left.
	IncreaseDrinks(): Bool	Request increase the number of product in the cart
	DecreaseDrinks(): Bool	Request decrease the number of product in the cart
	DeleteDrinks(): Bool	Request delete product in the cart

3.4.4.11 Manage discount controller

Manage discount controller		
Operation	Function	Meaning
	GetListOfDiscount(): Array	Request database to get the list of discount information to display on the GUI
	ValidateData(): Bool	Validate if the input information format is correct
	InsertDiscount(Discount): Bool	Request database to create a new discount
	DeleteDiscount(DiscountID): Bool	Request database to delete the selected discount

3.4.4.12 View bill controller

View bill controller		
Operation	Function	Meaning
	GetListOfBill(): Array	Request database to get the list of bill information to display on the GUI
	ValidateData(): Bool	Validate if the input information format is correct
	GetBill(Orderbill_ID): Bill	Request database to get the bill information selected by employee/manger
	SearchBill(Date)	Request database to get a list of bill which has Create_Date = Date
	Sort(): void	Sort the order of bill display

3.4.4.13 Employee controller

Employee controller		
Operation	Function	Meaning
	GetListOfEmployees(): Array	Request database to get the list of bill information to display on the GUI
	GetNewEmployeeID(): Int	Request database to get the new employee id
	ValidateDataFormat(): Bool	Validate if the input information format is correct
	AddNewEmployee(Employee): Bool	Request database create a new employee
	GetEmployeeInformation(EmployeeID): Employee	Request database to get information of employee selected by manager
	UpdateEmployee(Employee): Bool	Request database to make a change on a existed employee
	DeleteEmployee(EmployeeID): Bool	Request database to delete the selected employee
	SearchEmployee(String): Array	Request database to get a list of employee related to the string parameter

3.4.4.14 Manage report controller

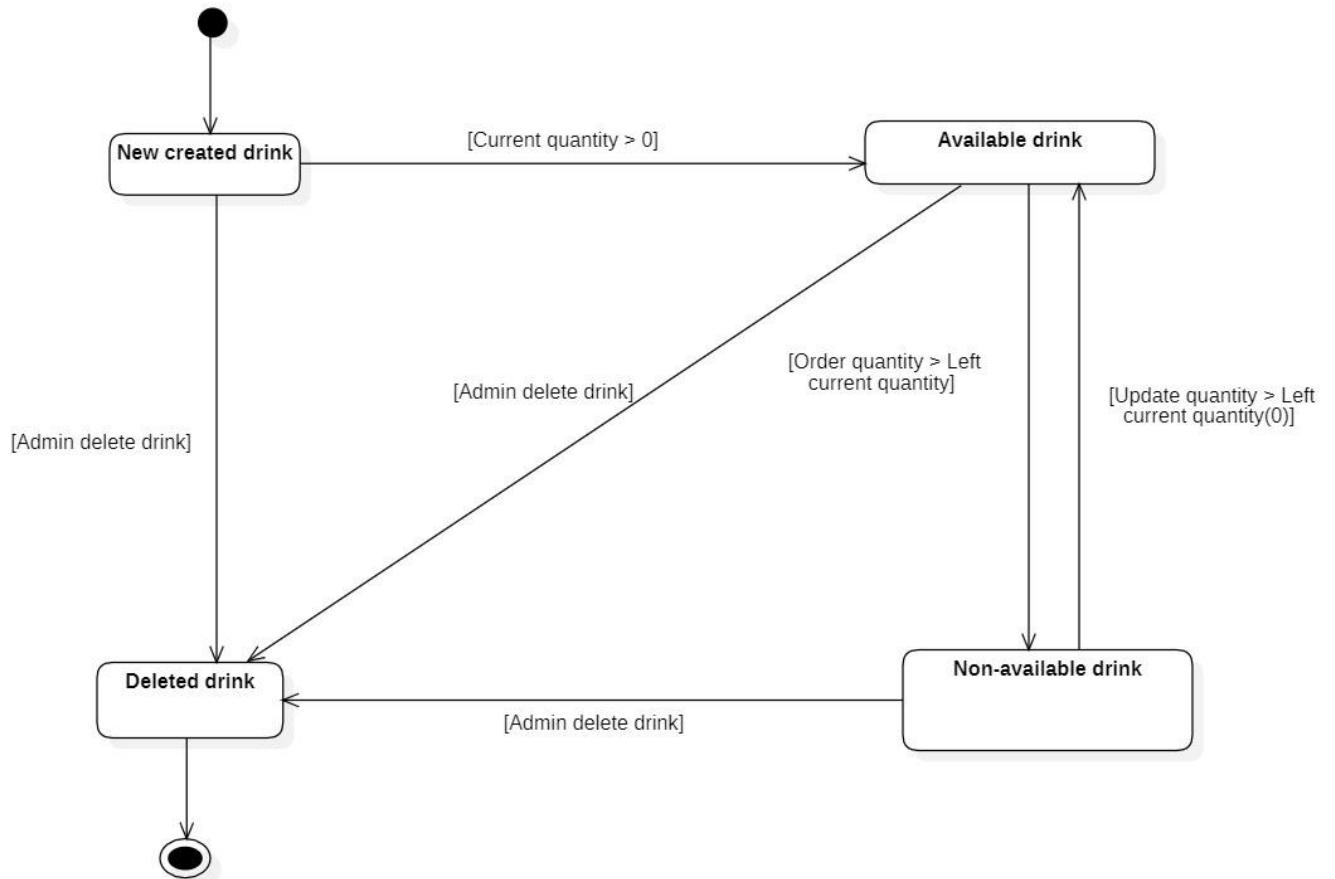
Manage report controller		
Operation	Function	Meaning
	GetListOfBill(): Array	Request database to get the list of bill information to display on the GUI
	ValidateData(): Bool	Validate if the input information format is correct
	GetBill(Orderbill_ID): Bill	Request database to get the bill information selected by manager
	SearchBill(Date): Array	Request database to get a list of bill which has Create_Date = Date

3.4.4.15 Order place controller

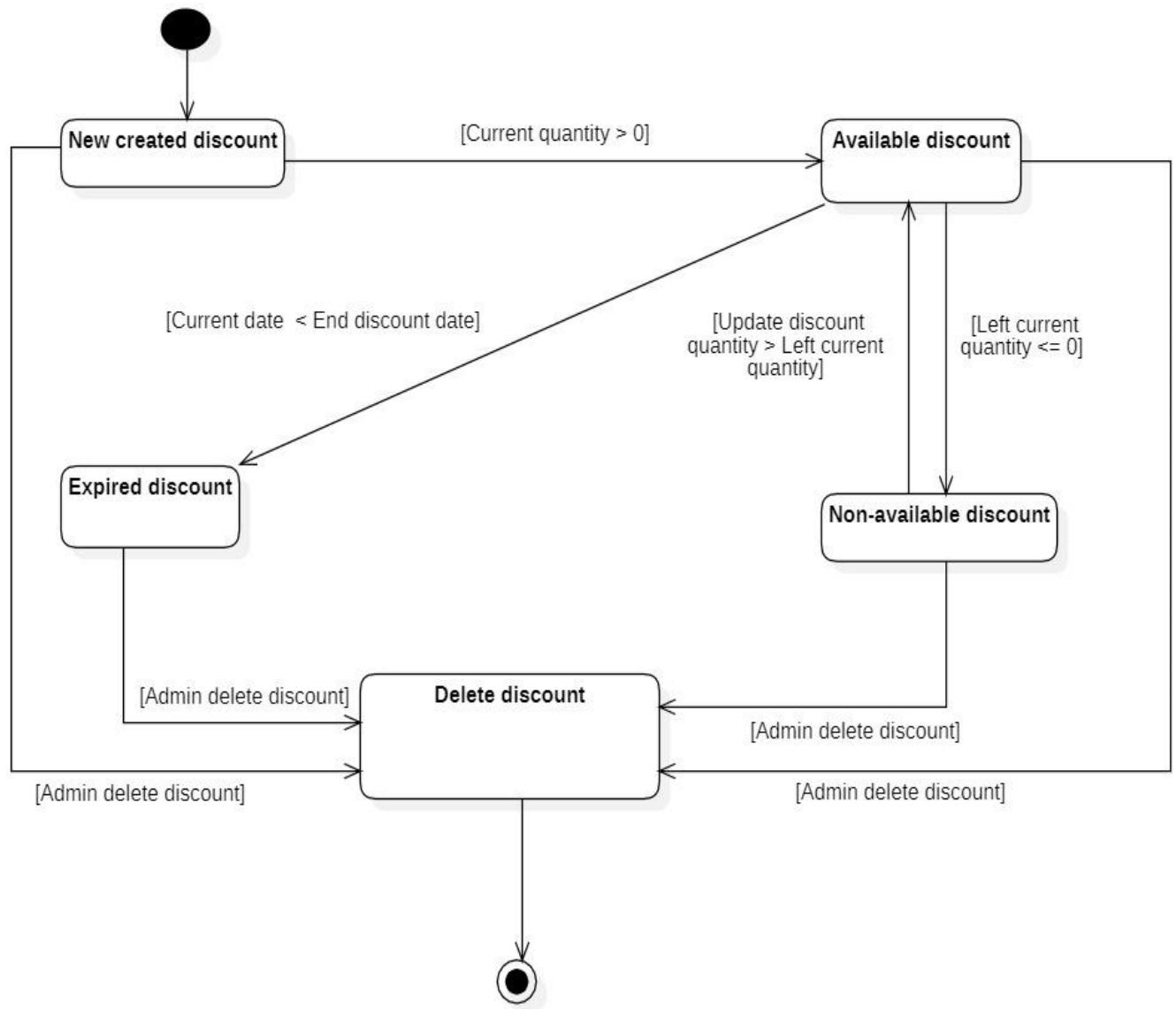
Order place controller		
Operation	Function	Meaning
	GetListOfDrinks(): Array	Request database to get the list of drinks information to display on the GUI
	CheckQuantityLeft(ProductID): Bool	Request database check quantity left of the drinks selected
	GetDrinksInformation(ProductID): Product	Request database to get the drinks information selected by customer
	SearchDrinks(String): Array	Request database to get a list of drinks related to the string parameter

3.5 State diagram

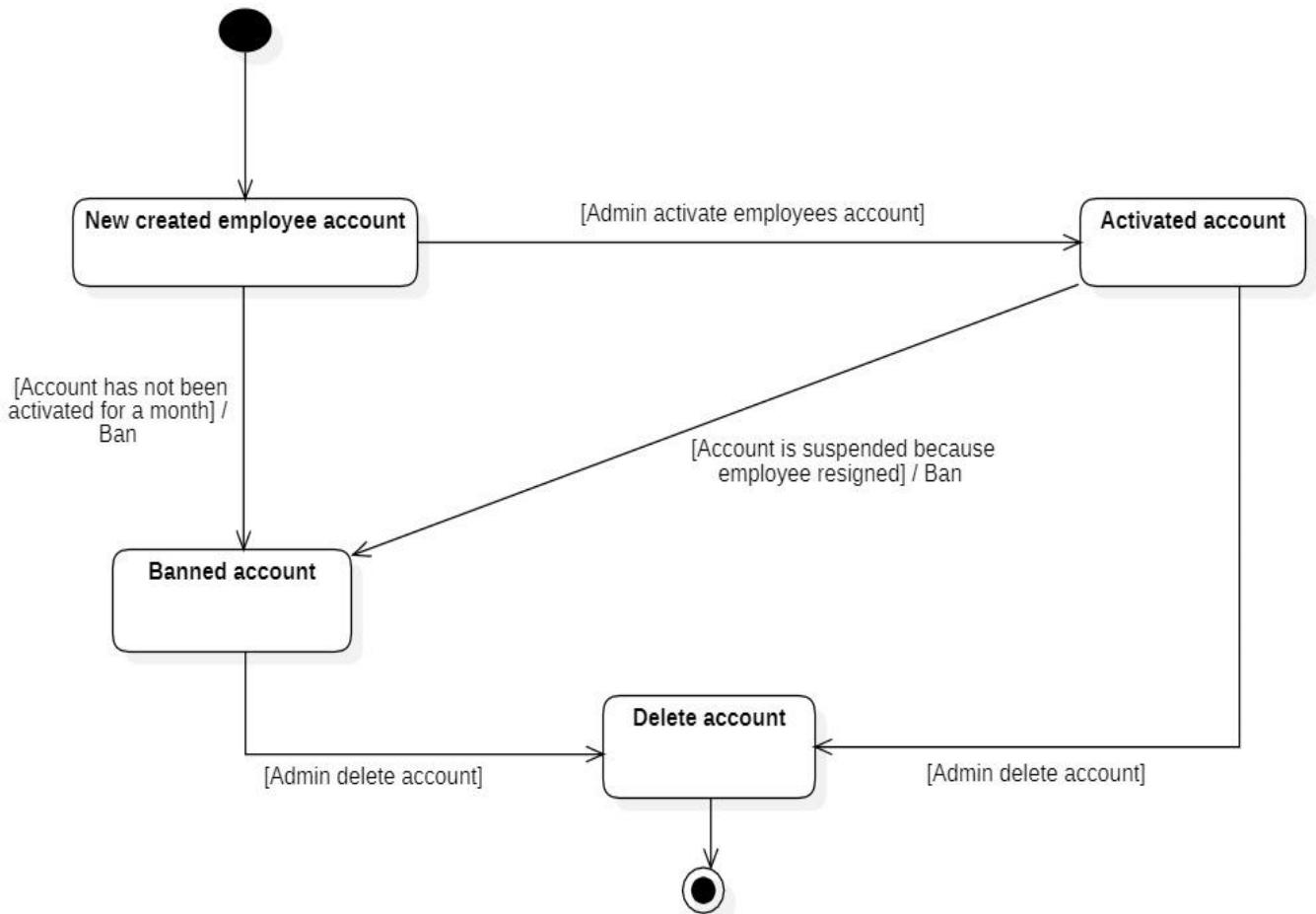
3.5.1 Drinks



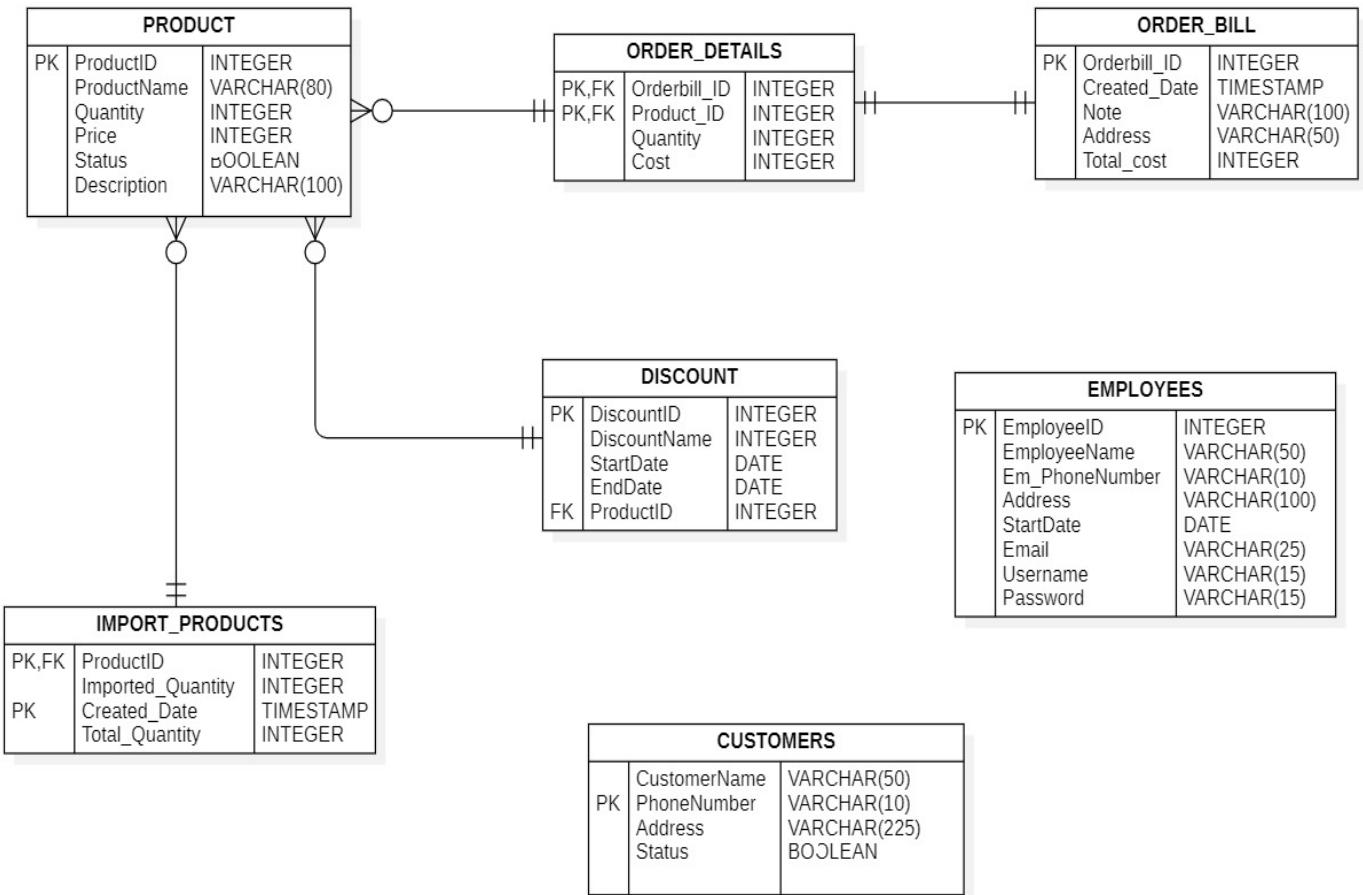
3.5.2 Discount



3.5.3 Employee



3.6 Mapping to Relational Model from Entity class diagram



CHAPTER 4:

IMPLEMENTATION AND TESTING

4.1 Introduction to the implementation environment

The system includes a desktop application installed for managers and employees. A website for customers to perform operations such as ordering drinks, viewing product list, ...

4.1.1 Development environment

- Operating system: Window 10
- Database: MySQL
- Text Editor:
 - Visual studio code (website)
 - Netbeans (desktop app)
- Web server: XAMPP
- Web development language: PHP, HTML, CSS, JAVASCRIPT
 - Framework: CodeIgniter
 - HTML, CSS, and JavaScript used for designing the user interface.
- Programming language for app development: Java Swing

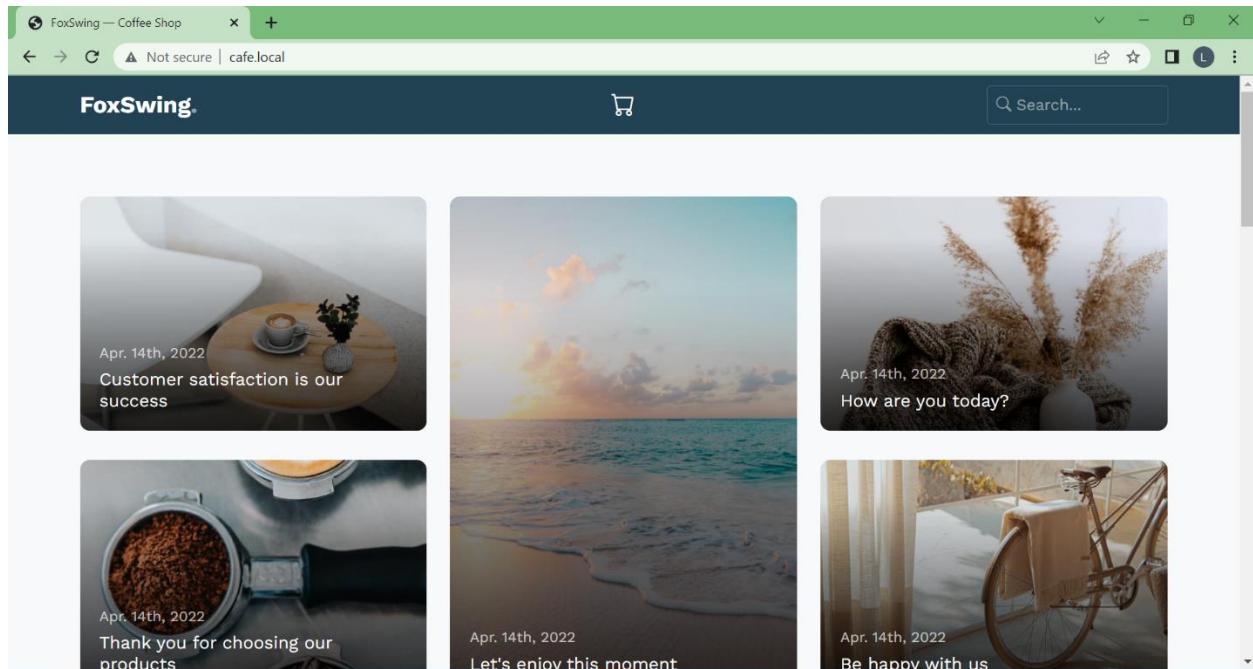
4.1.2 Deployment environment

- Software version must be compatible with the operating system, such as:
XAMPP Control Panel, MySQL và CodeIgniter, NetBeans IDE support for Windows 7, Windows 8/8.1 và Windows 10, MacOS, Linux.
- RAM memory:

Minimum 2GB RAM, but recommend at least 4GB RAM for better experience.

4.2 User Interface

4.2.1 Customer interface



Picture 1. Customer's homepage

SẢN PHẨM

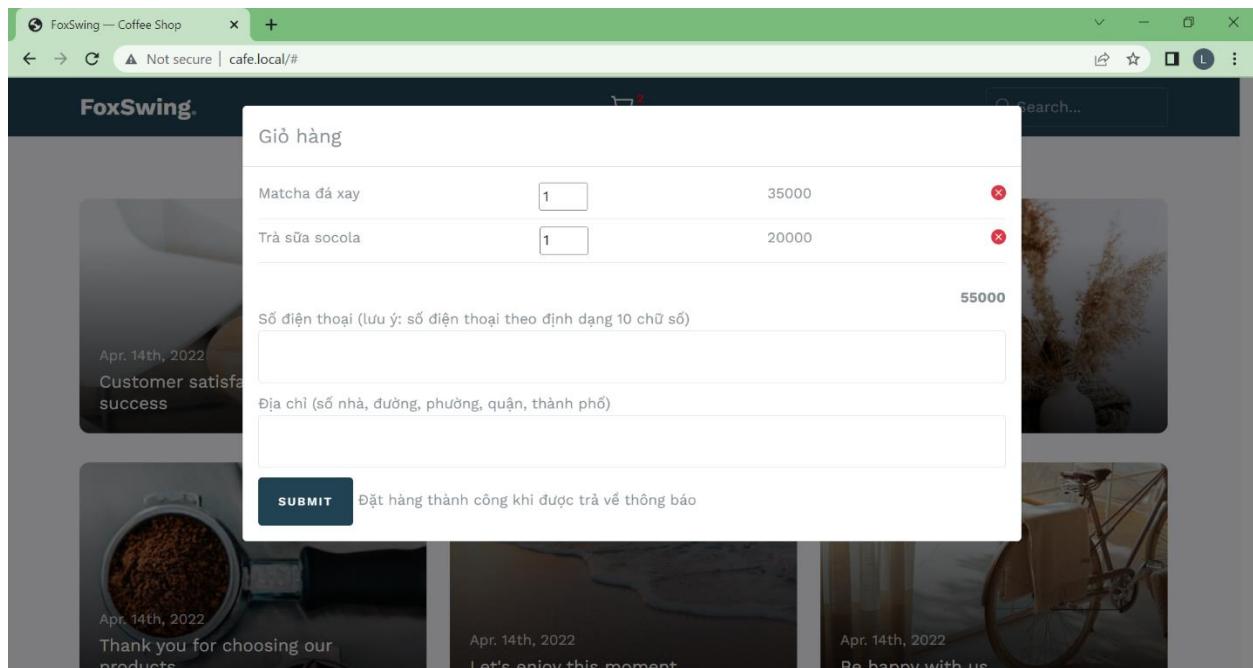
Tên Sản Phẩm	Giá	Mô Tả	Đutton
Cà phê phin sữa đá	12000đ	Sự kết hợp hoàn hảo giữa hạt cà phê Robusta & Arabica thượng hạng	Add to cart
Cà phê phin sữa đá	20000đ	Sự kết hợp hoàn hảo giữa hạt cà phê Robusta & Arabica thượng hạng	Add to cart
Matcha đá xay	35000đ	một thức uống chưa bao giờ hết hot	Add to cart

Picture2. List of product

Tên Sản Phẩm	Giá	Mô Tả	Đutton
Trà đào cam sả	25000đ	sự thanh mát của quả đào, vị ngọt của cam và hương thơm từ sả	Add to cart
Trà sữa socola	20000đ	thơm ngon đến từng giọt cuối cùng	Add to cart

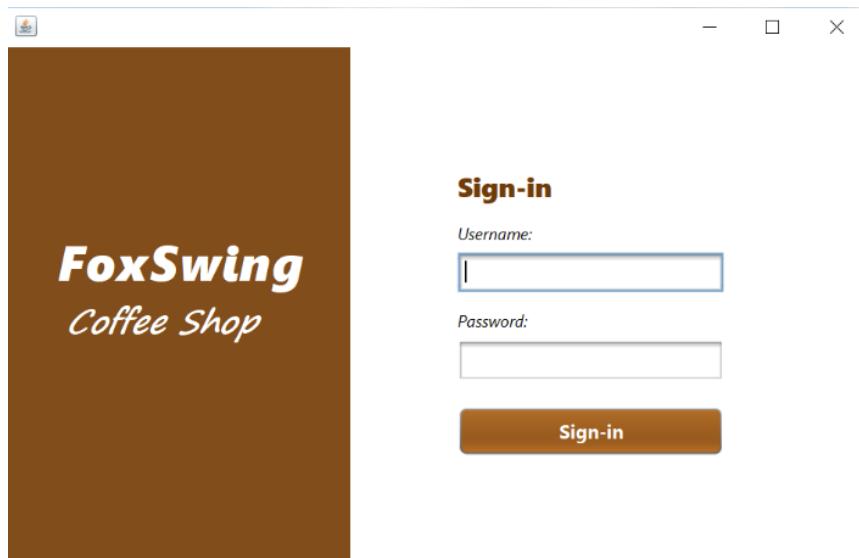
[cafe.local/single.html](#)

Picture3. List of product

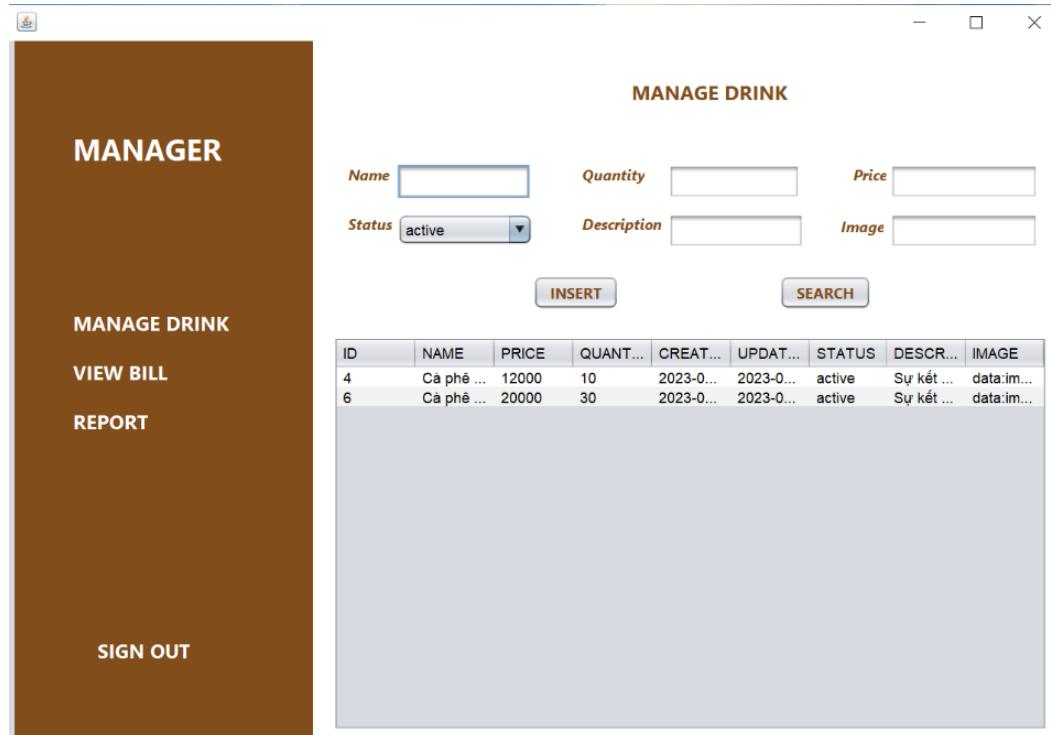


Picture4.Cart

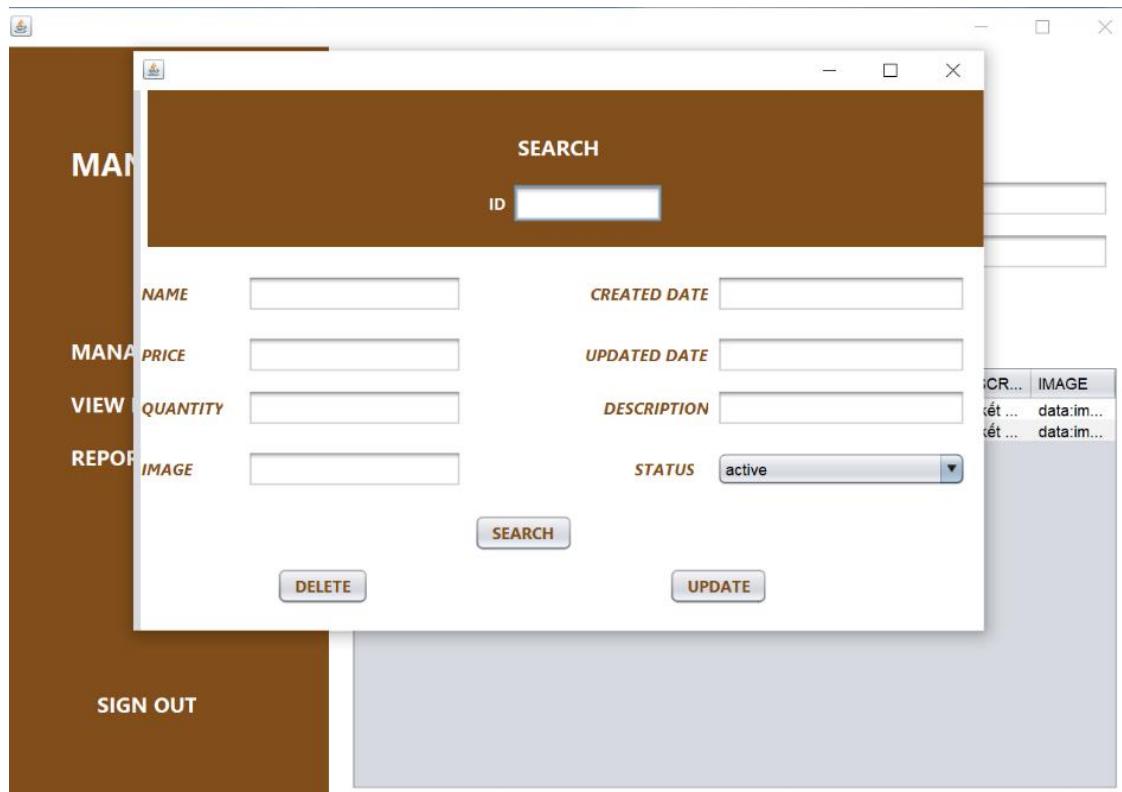
4.2.1 Manager interface



Picture1. Sign in



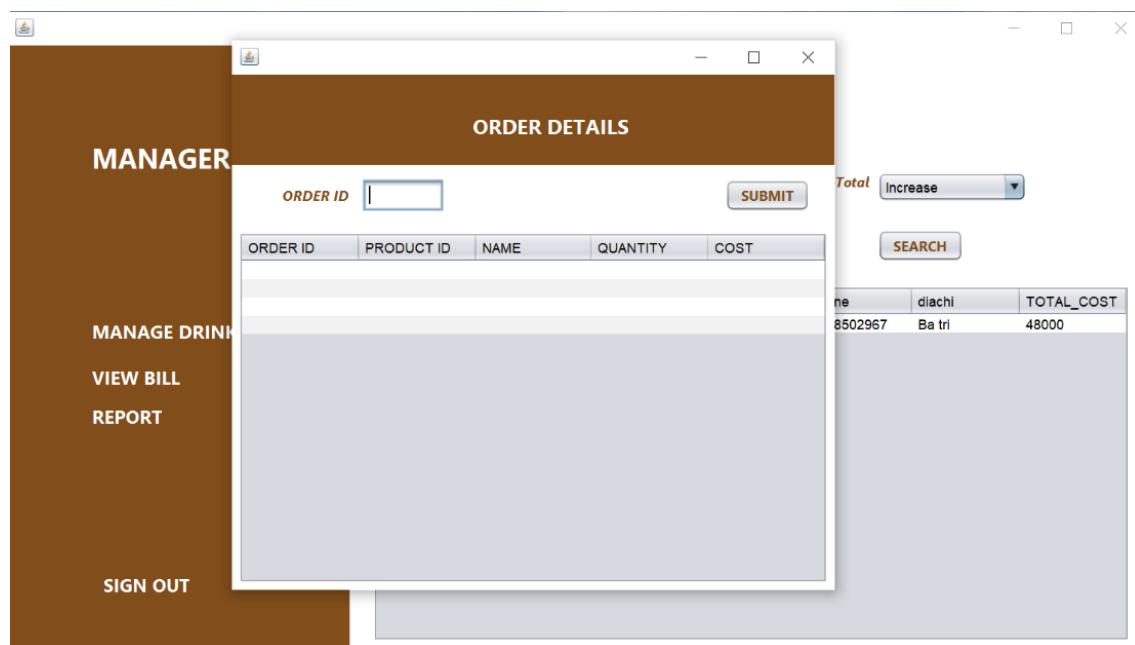
Picture2. Manage Drink



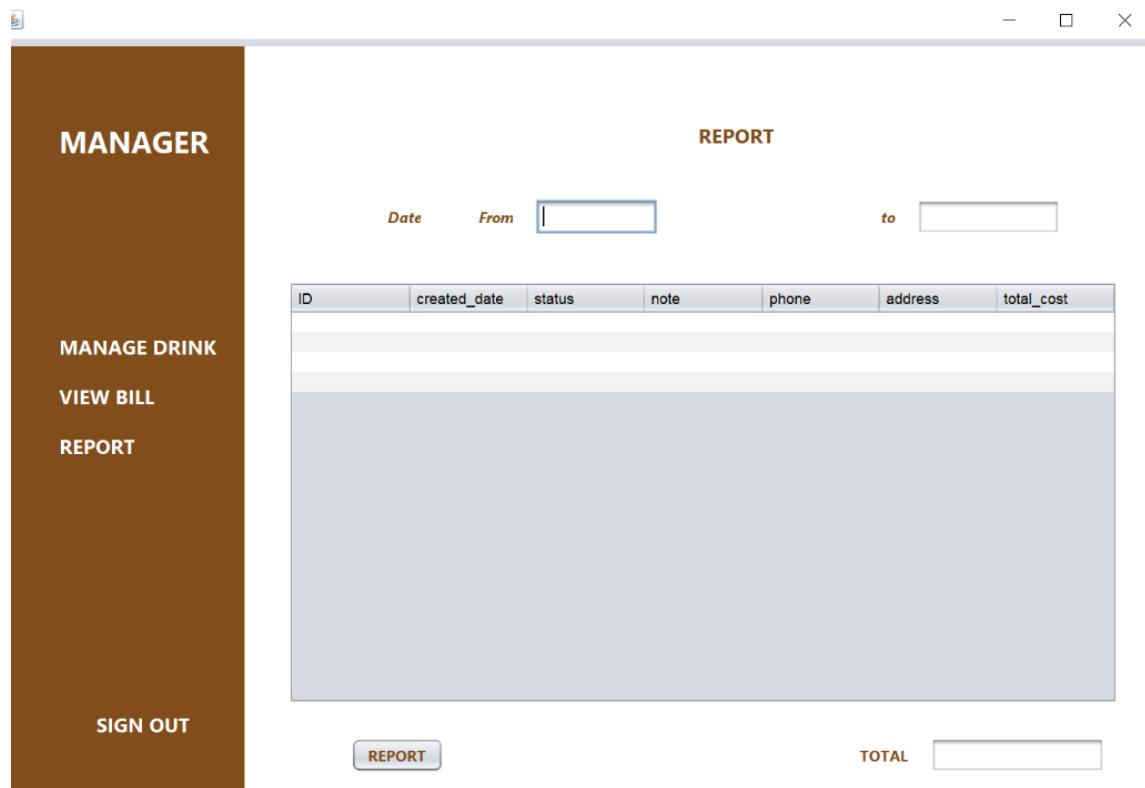
Picture3.Search Drink



Picture4. View bill



Picture5. Order details



Picture6. Report(order_bill)

CHAPTER 5:

CONCLUSION

5.1 Project achievement

Ordering coffee online through coffee shop websites is no longer uncommon. We usually select our favorite coffee shop and start ordering drinks on their website. However, with the coffee ordering application project, our team wants to integrate everything into one platform, giving users more choices thanks to the concentration of menus from many different coffee shops. We have built two sites, one for customers who want to order drinks and the other for administrators who control and operate the system. These tasks took us a lot of time, but it is worth it. Below are the primary functions of our system:

- Manage drinks cart
- Sign in for Administrators
- Manage drinks
- Manage employee
- Manage customers
- Manage Order bill
- Manage users
- Manage discount
- Create a loyalty program for frequent customers.

We used ReactJs, NodeJs and MySQL technologies to build the systems. Thanks to the MVC design pattern, our project is separated into many modules easily, which helps us solve a lot of problems about project management.

5.2 Limitations

Due to the limited time of the study, the coffee shop website project has the following limitations:

-
- Cannot process payments directly on the website.
 - No integration with third-party delivery services.
 - The website is not optimized for search engine rankings.
 - Limited customization options for drinks (e.g. specific milk options, extra shots).
 - No social media sharing options for orders.

5.3 Development direction

In the future, the team will develop and complete the coffee shop website project with the following goals:

- Enable direct payment processing on the website.
- Integrate with third-party delivery services for home delivery.
- Optimize the website for search engine rankings to increase traffic and visibility.
- Offer more customization options for drinks to better meet customer preferences.
- Add social media sharing options for orders to increase brand awareness and customer engagement.