

VIETNAM NATIONAL UNIVERSITY - HO CHI MINH CITY
UNIVERSITY OF INFORMATION TECHNOLOGY
FACULTY OF INFORMATION SYSTEMS



CAPSTONE PROJECT FINAL REPORT
INFORMATION SYSTEM ANALYSIS AND DESIGN
BUFFET RESTAURANT MANAGEMENT SYSTEM

Instructors: PhD. Cao Thị Nhạn, Ms. Dương Phi Long

Nguyễn Trương Đình Giang-21520215
Đỗ Hiền Thảo-21520460
Nguyễn Dương Chí Tâm-21520439
Huỳnh Mạnh Huy-21520259

THANK YOU

In the journey of life, perhaps everyone has experienced different emotions, there are successes and failures, but that is also the achievement of individuals as well as groups. And behind that is the help from everyone around. Today, to be able to complete this course project, we are very grateful to the teachers who have supported us wholeheartedly, provided us with knowledge as well as profound advice.

First of all, we would like to sincerely thank all the lecturers of the University of Information Technology - National University of Ho Chi Minh City as well as the school who have shared knowledge and enthusiasm with us during the time we carried out the course project.

Especially, we would like to express our sincere thanks to Ms. Cao Thi Nhan - Theoretical lecturer of System Analysis and Design course and Mr. Duong Phi Long - Practical lecturer of System Analysis and Design course who directly taught us, conveyed knowledge, experience, and guided us in a specific way to help us complete our course project well. We wish you good health and always passionate about your profession so that you can continue to teach and guide future generations of students.

In addition, we would like to thank the entire class IS201.N23.HTCL in general as well as the members in our group in particular who have studied together, researched together, learned from each other to carry out the course project in the best way.

Finally, we have completed a project called "Buffet restaurant management system". Although we have maximized what we have learned but still difficult to avoid mistakes, so we hope to receive comments from teachers to continue improving our project in the best way. Through that, draw lessons and accumulate more experience as a luggage for the future.

We would like to end here and sincerely thank you teachers and classmates!

Ho Chi Minh City, April 13, 2023

Contents

Chapter 1: INTRODUCTION	8
1.1. Introduction	8
1.2. Problems of Project	8
1.3. Objectives of Project	8
1.4. Scope of project	9
1.5. Developer tools & technology	9
Chapter 2: REQUIREMENT SPECIFICATION	9
2.1. Situation survey/ Survey of the current situation	9
2.1.1. Interview	9
2.1.2. Survey of the organization/ Current organizational situation	13
2.1.3. Survey of business processes, major activities	14
2.1.3.2 Service Department	15
2.1.3.3 Other Department	16
2.2. Software requirements	16
2.2.1. Functional requirements	16
2.2.1.1. Storage requirements	16
2.2.1.2. Feature requirements	16
2.2.2. Non-functional requirements	16
Chapter 3: ANALYSIS AND DESIGN	17
3.1. Use-case diagram	17
3.1.1. List of actor	18
3.1.2. List of use-case	19
3.2. Use-case specification and activity diagram	20
3.2.1 Sign up	20
3.2.2 Sign in(for customers)	21
3.2.3 Manage profile	24
3.2.4 Book tickets	25
3.2.5 Search	27
3.2.6 Review transaction history	28
3.2.7 Sign in (for administrator)	29

3.2.8. Manage customers	31
3.2.8.1 Add customers.....	31
3.2.8.2 Delete customers.....	32
3.2.8.3 Search customers	33
3.2.8.4 Update customers	34
3.2.12 Manage Employees	36
3.2.12.1 Add employee	36
3.2.12.2 Update employee.....	38
3.2.12.3 Search employee	40
3.2.12.4 Delete employee	42
3.2.10 Manage tickets	43
3.2.10.1 Add tickets	43
3.2.10.2 Update tickets.....	45
3.2.10.3 Search tickets	47
3.2.10.4 Delete tickets	48
3.2.11 Manage ingredients	50
3.2.11.1 Add ingredients.....	50
3.2.11.2 Update ingredients	52
3.2.11.3 Search ingredients.....	54
3.2.11.4 Delete ingredients.....	55
3.2.12 Manage suppliers	57
3.2.12.1 Add suppliers	57
3.2.12.2 Delete suppliers	59
3.2.12.3 Search suppliers	60
3.2.12.4 Update suppliers	61
3.2.13 Manage Promotional ads	63
3.2.13.1 Add promotional ads.....	63
3.2.13.1 Update promotional ads	65
3.2.13.3 Search promotional ads	66
3.2.13.4 Delete promotional ads.....	68
3.2.14 Revenue analysis	69

3.3 Sequence diagram and class diagram	71
3.3.1 Sign up	71
3.3.1.1 Sequence diagram.....	71
3.3.1.2 Class diagram.....	71
3.3.2 Sign in (for customers).....	72
3.3.2.1 Sequence diagram.....	72
3.3.2.2 Class diagram.....	73
3.3.3 Manage profile.....	74
3.3.3.1 Sequence diagram.....	74
3.3.3.2 Class diagram.....	75
3.3.4 Book tickets	76
3.3.4.1 Sequence diagram.....	76
3.3.4.2 Class diagram.....	77
3.3.5 Search	78
3.3.5.1 Sequence diagram.....	78
3.3.5.2 Class diagram.....	79
3.3.6 Review transaction history.....	80
3.3.6.1 Sequence diagram.....	80
3.3.6.2 Class diagram.....	80
3.3.7 Sign in (for administrator)	81
3.3.7.1 Sequence diagram.....	81
3.3.7.2 Class diagram.....	82
3.3.8. Manage customers.....	83
3.3.8.1 Sequence diagram.....	83
3.3.8.2 Class diagram.....	87
3.3.9 Manage employees	88
3.3.9.1 Sequence diagram.....	88
3.3.9.2 Class diagram.....	92
3.3.10 Manage tickets.....	93
3.3.10.1 Sequence diagram	93
3.3.10.2 Class diagram.....	97

3.3.11 Manage ingredients.....	98
3.3.11.1 Sequence diagram	98
3.3.11.2 Class diagram.....	102
3.3.12 Manage suppliers	102
3.3.12.1 Sequence diagram	103
3.3.12.2 Class diagram.....	107
3.3.13 Manage promotional ads	108
3.3.13.1 Sequence diagram	108
3.3.13.2 Class diagram.....	112
3.3.14 Analysis Revenue	113
3.3.14.1 Sequence diagram	113
3.3.14.2 Class diagram.....	113
3.4 Class diagram and entity class diagram	114
3.4.1 Customer overview class diagram.....	114
3.4.2 Administrator overview class diagram.....	114
3.4.3 Entity class diagram	115
3.4.4 Class diagram specification.....	115
3.4.4.1 Customer	116
3.4.4.2 Invoice.....	117
3.4.4.3 Administrator	118
3.4.4.4 Table	119
3.4.4.5 Meal.....	119
3.4.4.6. Ticket.....	120
3.4.4.7. Employee	121
3.4.4.8 Ingredient.....	122
3.4.4.9 Supplier.....	123
3.4.4.10 Customer sign in control.....	124
3.4.4.11 Customer sign up control	125
3.4.4.12 Profile control	125
3.4.4.13 Booking control.....	126
3.4.4.14 Search control.....	127

3.4.4.15 transaction history control	127
3.4.4.16 Admin sign in control	128
3.4.4.17 Manage customer control.....	128
3.4.4.18 Manage employee control.....	128
3.4.4.19 Manage ticket control.....	129
3.4.4.20 Manage ingredient control	130
3.4.4.21 Manage supplier control	131
3.4.4.22 Manage promotional ads control	132
3.4.4.23 Revenue analysis control.....	133
3.5 State diagram	133
3.5.1 Ticket	133
3.5.2 ingredient.....	134
3.5.3 customer account.....	135
3.6 Mapping to Relational Model diagram from Entity class diagram	136
4. IMPLEMENTATION AND TESTING	137
4.1 Introduction to the implementation environment.....	137
4.2 User Interface	137
4.2.1 Customer interface.....	137
4.2.2 Admin interface	141
5. CONCLUSION	145
5.1 Project achievement	146
5.2 Limitations.....	146
5.3 Development direction	146

Chapter 1: INTRODUCTION

1.1. Introduction

With the ongoing development of society in today's era, material life as well as the requirements of each individual are increasingly focused on and strengthened. "Industrial revolution 4.0" was born with the modern development of technological and electronic devices. The competition is tough; also, client demand for use and purchasing is expanding. As a result, manual store management will encounter numerous challenges such as not being able to manage customers, manufacturer information, product prices, product management, document storage... The loss will generate problems, affecting the operations and income of the business. During the research, our team discovered that some retailers still did not comprehend the process or apply technology to management.

Our team expects that this topic will help organizations reduce challenges, and improve management, resulting in optimum work efficiency. Furthermore, when implemented, this project will save the most money, optimize human resource management, and reduce risk. In addition to constructing and developing software, the management method is especially important.

1.2. Problems of Project

To solve store management issues, the team proposed developing a web application that would allow employees to easily manage customers, income, and so on. At the same time, customers may get menu information, table reservation information, and meal combo prices. Moreover, the online application also makes management and client service more convenient and sensible.

1.3. Objectives of Project

Build a complete barbecue buffet management software system to make the management of staff, booking customers' tables convenient. Provide detailed information about servings, prices of each meal, offers in the store department. Customers can know information: Name of each dish, table number, ticket price ... In addition, the application also helps customers conveniently book tables online as well as change tables, cancel tables at the store. Help store system management staff easily statistics revenue, add promotions, as well as take care of customers.

- Know how to analyze information system construction: Structurally and object-oriented.
- Completion of the subject, achievement of results and outcome standards of the subject.

- Practice teamwork ability, analyze and design information systems according to the requirements of the organization, synthesize and write reports.
- Successfully build grill buffet store management software.

1.4. Scope of project

Due to the limited time to implement the project, our team will concentrate on some main features in building the application of buffet store management system such as:

- Customer management.
- Employee management.
- Supplier management.
- Warehouse management.
- Booking.
- Retrieval information.

1.5. Developer tools & technology

- Oracle as Model Component in MVC.
- Microsoft Studio as text editor.
- ... as View Component in MVC.

Chapter 2: REQUIREMENT SPECIFICATION

2.1. Situation survey/ Survey of the current situation

2.1.1. Interview

Table 1. Survey Strategy

General survey				
System: OMG buffet				
Creator: Nguyễn Dương Chí Tâm			Created Date: 20/04/2023	
Ord	Topic	Requirement	Start	End
1	Tickets management	To get detailed information about ticket management, ticket sales	20/04/2023	20/04/2023

2	Customers management	To get detailed information about the management of customer information, membership card	20/04/2023	20/04/2023
3	Analysis and reports on sales	To get detailed information about revenue management and financial reporting	20/04/2023	20/04/2023
4	Manage new and promotions	to get information about the promotion, how to apply the promotion.	20/04/2023	20/04/2023

Table 2. Survey form

Survey Forms	
System: OMG buffet	
Interviewee: Đỗ Hiền Thảo	Interviewee: Nguyễn Dương Chí Tâm
Place: OMG buffet Target: Collect and understand working process of specific buffet restaurant.	Start: 7:00 am 20/04/2023 End: 11:30 am 20/04/2023
Details of the interview: • Introduction • Overview about current system • Survey plan: • Topic 1: Tickets management • Topic 2: Customers management • Topic 3: Analysis and reports on sales	Estimated Time: • 5 mins • 15 mins • 40 mins

• Topic 4: Manage new and promotions	
General observe	
Unexpected occurrence	

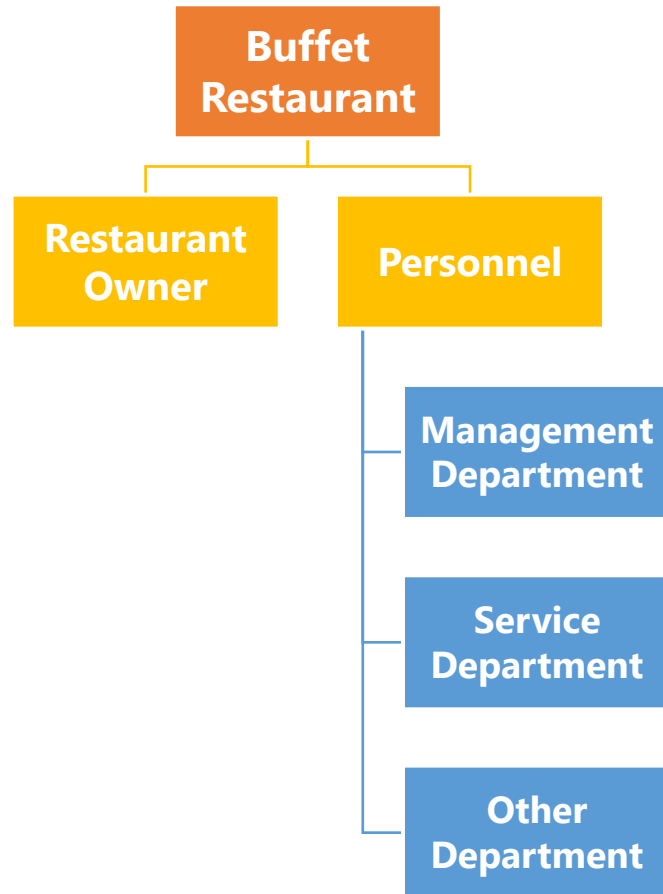
Table 3. Question & records

Interviewee: Nguyễn Trương Đình Giang		
Questions		Records
Topic 1	How do customers who have booked a table but arrive later than scheduled?	In case if the customer has booked a table but arrives late compared to the schedule, the restaurant will consider the status of customers at that time more or less. If it is crowded, it will keep the booked table for 30 minutes, if the person who booked it is more than 30 minutes late, it will give up that table to another guest. On the contrary, it will keep the table set until the customer arrives
	How are children's tickets handled?	For children, everyone shorter than 120cm will be considered a child ticket, with no age regulation, only height. And child tickets are priced at 30% of the new adult ticket
	Will ticket prices on public holidays change and how will they change?	Ticket prices on public holidays will increase by 10% compared to the recommended price and promotions are not applicable
Topic 2	Can customers save their information	Customer information will be saved on the invoice but as a normal customer. And not a member, customers

	without a membership card?	cannot book tickets online, customers can only Book a table at the cashier
	Do customer levels have different priorities?	<p>Different degrees and priorities will be assigned to member customers:</p> <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 and Free one-time water call if the customer's score is between 1500 and 2999. • Over 3000: VIP members receive a 10% discount on their bill and free freshwater.
	So what can I do to get a membership card?	To be able to make a membership card: The customer's total bill must be over one million Vietnam Dong and accompanied by self-identification documents
Topic 3	How long does it take to find out how many tickets have been sold?	The restaurant's statistics and sales revenue will be collected each day, and each week will compile those metrics if the manager requests data reporting
	What stats does the restaurant need?	<p>The system includes the following statistics:</p> <ul style="list-style-type: none"> • Statistics on the number of tickets sold for each meal • Statistics of the restaurant's revenue for a year for the specified period • Statistics on the quantity of ingredients that have been supplied and imported into the warehouse of the restaurant

Topic 4	In addition to promotions, are there any additional bonuses?	In addition to the introduced promotions, the restaurant will give souvenirs to member customers on the restaurant's anniversaries
	How will news and promotions be announced and updated?	News and promotions will be announced and updated as soon as possible so that customers can stay informed the best
	Do promotions hierarchize according to customer level?	For promotions, there will be no hierarchy according to the level of customers. The promotions apply to all customers, but each promotion will vary by day of the week

2.1.2. Survey of the organization/ Current organizational situation



2.1.3. Survey of business processes, major activities

Management Department consists of 4 employees:

- 1 Manager
- 1 Deputy Manager
- 1 Accountant
- 1 IT Staff

The Management Department is responsible for managing the import of materials, information about materials, information about employees in the store, and coordinating all activities of the store. The management team must grasp the situation of the store to report it to the store owner. Reporting is done monthly, quarterly, and on-demand.

Import Business: a new product that will be handled by an IT staff member who will store data in a management system. The information includes: barcode, purchase price, selling price, manufacturing date, expiration date, amount imported, unit price and notes.

- *Situation:* The delivery supplier or the store needs to import goods by another method.

- *Implementation:* Checking goods based on import invoices and updating data on goods entered into the system.

Human resource management Business: A manager will oversee all of the store's employees' actions. Including: recruiting employees, entering employee information, updating employee information, managing shifts and employee work history, periodically reporting employee activities, etc.

Analysis Business: All of the store's information, including personnel information, material management, revenue management, and so on, will be managed by a manager and a deputy manager.

- *Situation:* Make reports to store owners periodically or when required.
- *Implementation:* Use the statistical function supported by the support system. The system will automatically summarize and provide the findings based on your criteria and the time it takes to create data.

Marketing Business: The Management Department section develops customer-attracting techniques and promotions. The accountant will manage and make statistics of the costs of all the store's activities.

2.1.3.2 Service Department

Service Department consists of 10 employees:

- 3 Kitchen Staffs
- 5 Staffs
- 2 Cashiers

Staff working in shifts. Each employee works at designated locations. Customers choose the combo of buffet, and the staff notifies the kitchen crew. The cashier is responsible for paying the customer after the customer orders, as well as records each bill for each desk and reports to management after each shift.

Sales Business: Serving food to customers, maintaining the operation of the store. Including: ordering process and payment process.

- *Situation:* Customers after being guided and ordered by the service staff.
- *Implementation:* Enter the invoice, record the meal package, and number of servings ordered. Invoice customers, print invoices, and keep the system up to date.

Customer Services Business: Customer service that is dedicated and sensitive. Establish a close relationship between the store and customers; inform and help customers update

on the newest information about the store, especially promotional events. Customers who have the store's membership card will also have more preferential treatment.

2.1.3.3 Other Department

Security Department: Consists of 2 employees.

Responsible for securing the store, keeping the car, and guiding customers into the store. Ensure store security and order.

Quality Department: Consists of 2 employees.

Check and grasp the status of raw materials to ensure that the ingredients sent to customers are always fresh, delicious, and of the highest quality. When detecting a problem, it must promptly report it to the management department for appropriate resolution and coordination measures.

2.2. Software requirements

2.2.1. Functional requirements

2.2.1.1. Storage requirements

Must have customer data, staff, meals, number of people, number of tables, incentives (if any) customer reservation data.

2.2.1.2. Feature requirements

- + For customers: View information about meals, choose tables, number of tables booked by the store, view their reservation history.
- + For customer management staff, staff: Add staff, remove staff, edit employee information, look up customer information, fix customer information, search for customers. Statistics on the number of tables that staff have received guests.
- + For reservation managers: Add table information, edit table information.
- + For technical staff: Management of machinery systems.
- + For promotion managers: Add, remove, edit promotions.
- + For meal management staff: Add meal information, edit meal information, delete portions, list revenue of all types of meals by month and year.

2.2.2. Non-functional requirements

interface requirements: Multi-language support, easy to use, customer friendly, highly interactive

Quality requirements: The application works effectively, without causing discomfort to the user

High security: password protection, user accounts.

Availability requirements: The application can work continuously 24/7

Chapter 3: ANALYSIS AND DESIGN

3.1. Use-case diagram





3.1.1. List of actor

Index	Actors	Meaning
1	Administrator	The administrator is the person who manages the system
2	Customer	Customers are users of the service system

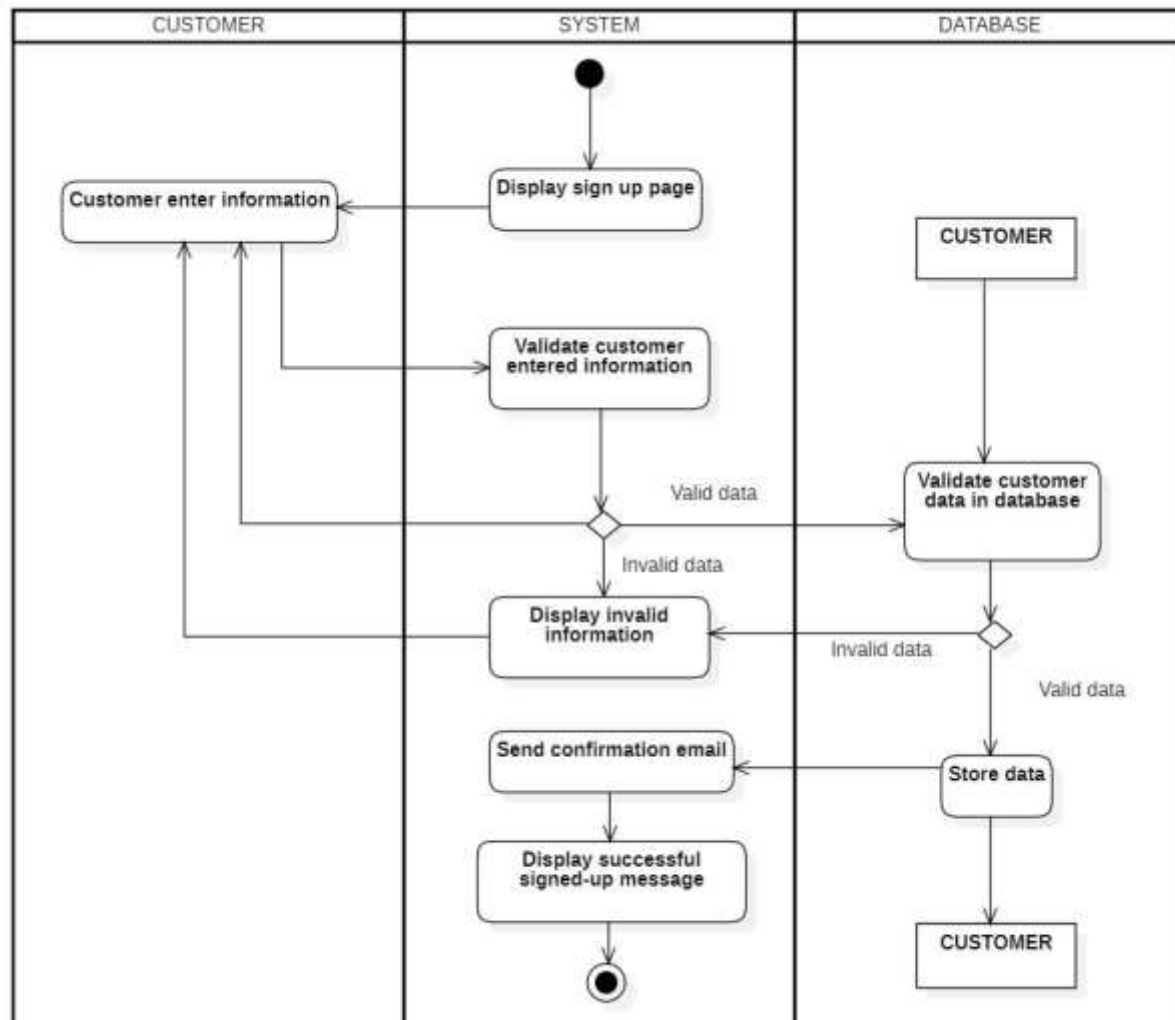
3.1.2. List of use-case

Role	Use case	Meaning
Customer's role	Sign up	Sign up for membership
	Sign in	Sign in to use the service
	Manage profile	Customers can manage their own personal information
	Book tickets	Book ticket and purchase
	Search	Customers can find information such as number of empty tables, information about dishes, services, ...
	Review transaction history	Help customers view transaction history details
Admin's role	Sign in	Help the administrator to login to the management page
	Manage customers	Add/Delete/Search/Update customers
	Manage employees	Add/Delete/Search/Update employees
	Manage tickets	Add/Delete/Search/Update ticket
	Manage ingredients	Add/Delete/Search/Update ingredients
	Manage suppliers	Add/Delete/Search/Update suppliers
	Manage promotional ads	Add/Delete/Search/Update Promotional ads
	Revenue analysis	Make it easier for managers to monitor revenue

3.2. Use-case specification and activity diagram

3.2.1 Sign up

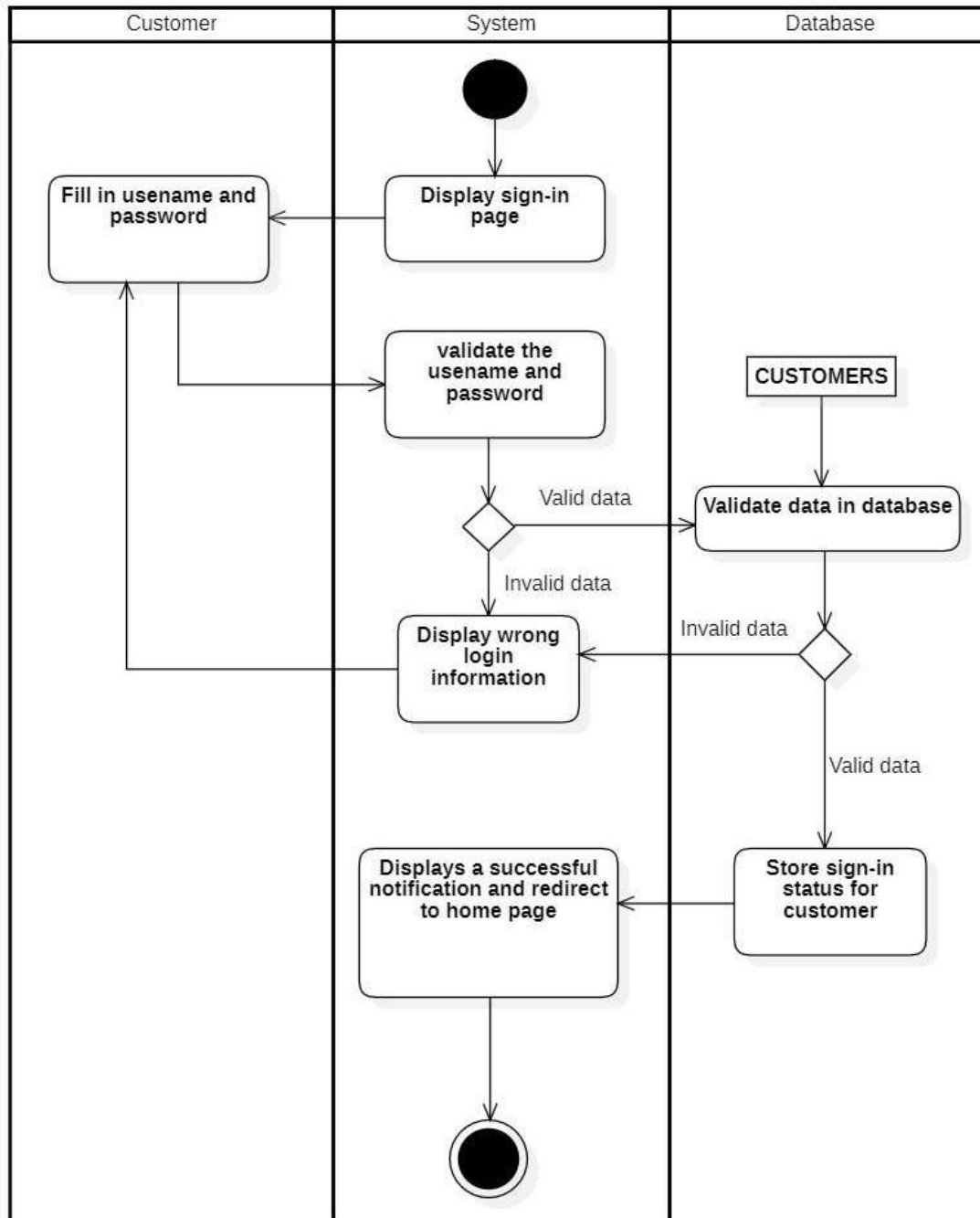
Use-case	Sign up
Description	Customers create an account for all system services.
Trigger	Customers click to button "Sign Up" or "Create an account".
Pre-condition	Customer's devices must connect to Internet.
Post-condition	Customer gets a notification about sign up successful.
Basic flow	<ul style="list-style-type: none">0. Customers clicks on sign in /create an account button1. System display sign up page.2. Customer enter information such as email, password, name, ... and click confirm3. System validate customer entered information in correct format.4. System validates customer information unique in database.5. System store customer information in the database.6. System send verification code via email.7. System notify customer that account created, customer can sign-in by using this account.
Alternative flow	<ul style="list-style-type: none">3a. System display invalid information due to wrong format. Use-case continue at step 2.4a. System display invalid information due to maybe have an existed account with customer entered information. Use-case continue at step 2.
Exception flow	



3.2.2 Sign in(for customers)

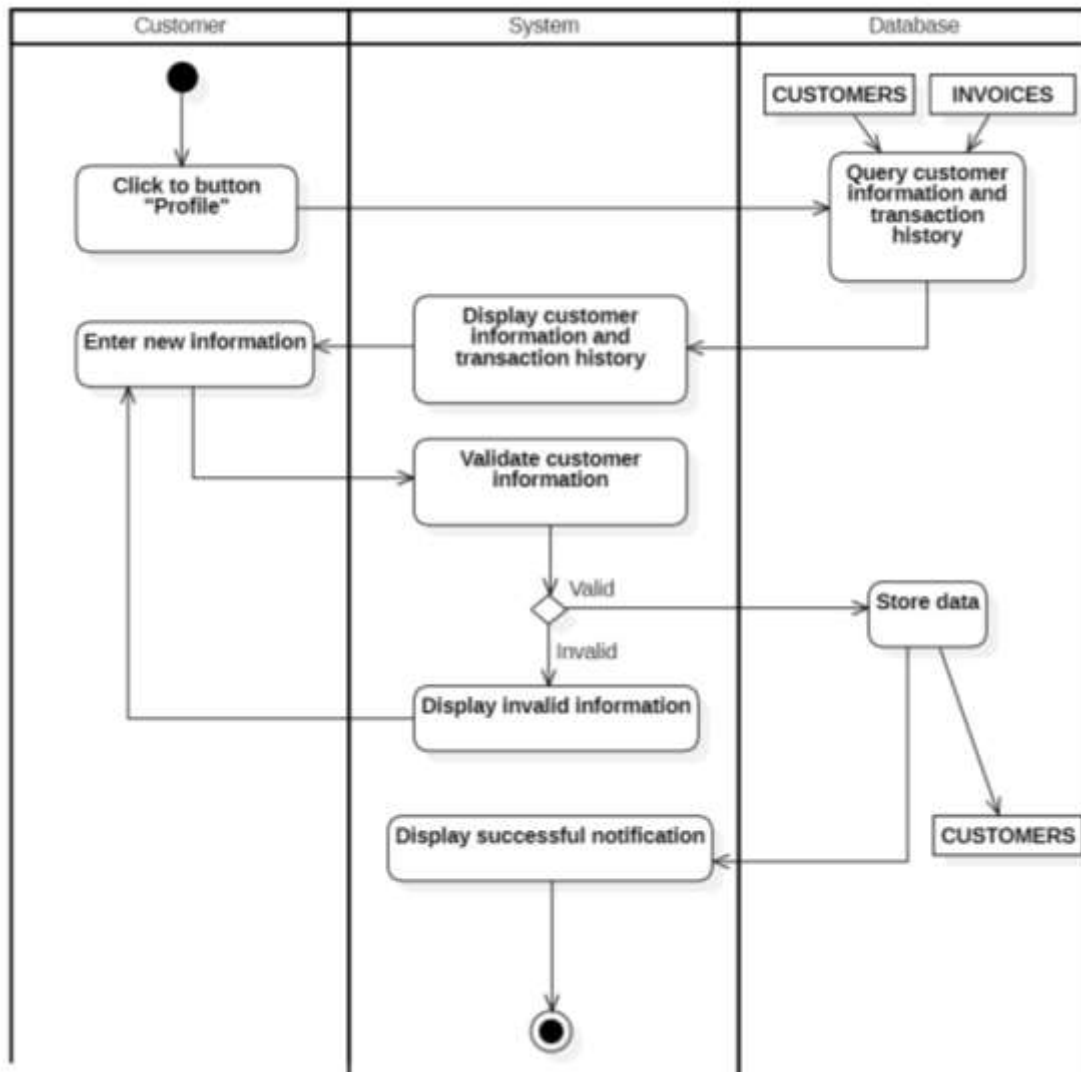
Use-case name	Sign in for customer
Description	Customers uses an existing account to login to system for all services.
Trigger	Customers clicks to button "Sign in" .
Pre-condition	Customer's device must connect to Internet Customer's account is existed.
Post-condition	Customers gets a notification about sign in successful. Customers is redirected to home page and customers can use services
Basic flow	1. Customers clicks to button "Sign in". 2. System displays sign-in page.

	<p>3. Customers enters username and password.</p> <p>4. System validates Customers account in the correct format.</p> <p>5. System validates Customer's data with existed customers account in database.</p> <p>6. System stores signed-in status of customers using sessions.</p> <p>7. System displays successful signed-in notification and redirect to home page.</p>
Alternative flow	<p>4a. System display invalid information due to wrong format. Use-case continues at step 3.</p> <p>5a. System display invalid information due to customer's account does not match with existing account information in database. Use-case continues at step 3.</p>
Exception flow	



3.2.3 Manage profile

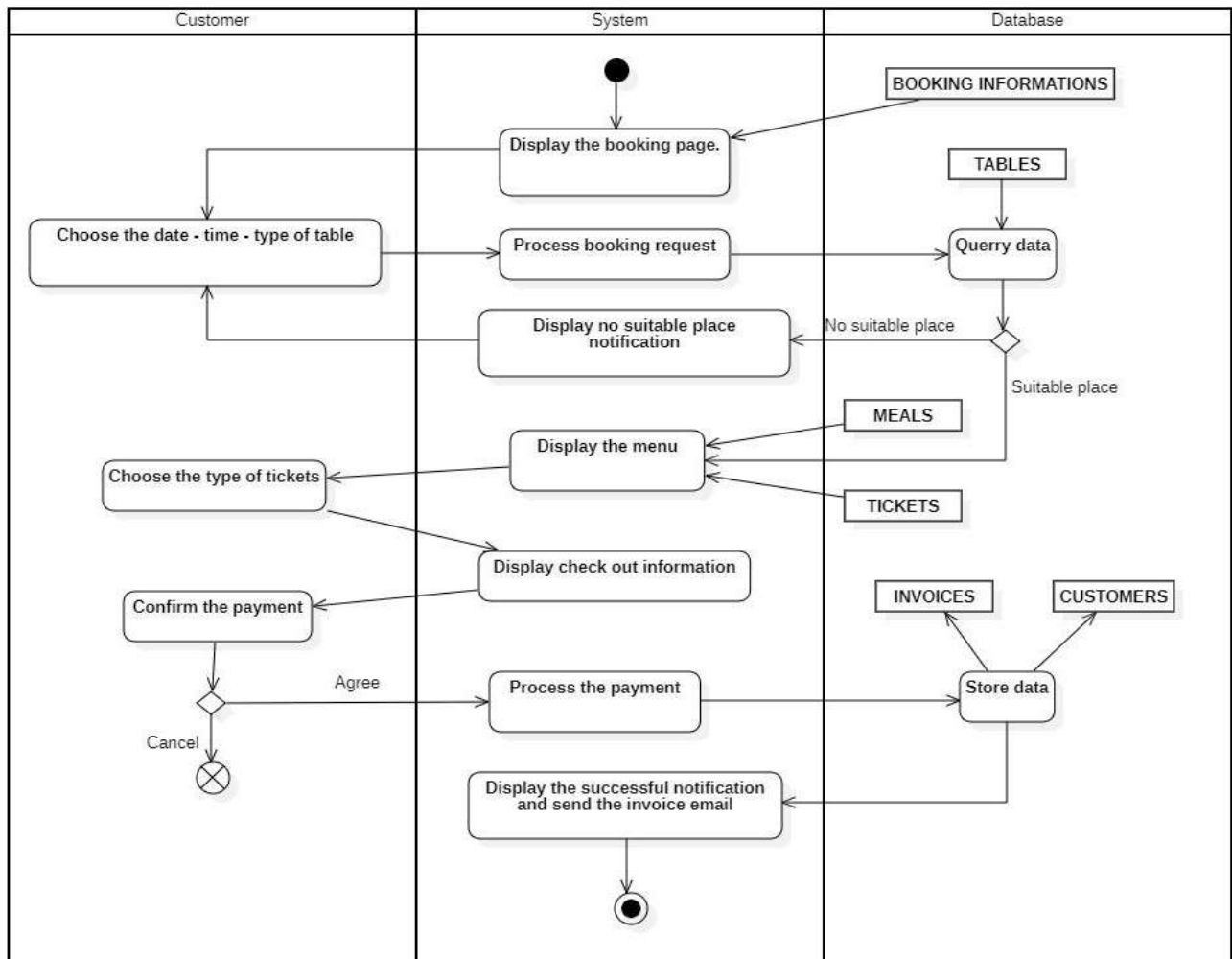
Use-case name	Manage profile
Description	Customers change private information.
Trigger	Customers click to the button "Profile".
Pre-condition	Customer's device must connect to Internet Customer's account is signed in
Post-condition	Customers get transaction history with detailed information.
Basic flow	<ol style="list-style-type: none">1. Customers click to button "Profile".2. System queries customer information and their transaction history based on signed-in status.3. System displays transaction history.4. Customers enter new information such as name, gender, date of birth, phone number, etc.5. System validates customer information in the correct format.6. System stores data in the database.7. System displays a successful notification.
Alternative flow	5a. System display invalid information due to the wrong format. Use-case continues at step 4.
Exception flow	



3.2.4 Book tickets

Use-case name	Book ticket
Description	Customers book ticket for a meal
Trigger	Customers click to button "Booking"
Pre-condition	Customer's devices must connect to the Internet. Customer's accounts existed. Customer's status is signed in.
Post-condition	Customers book tickets successfully. Customer's payment information is stored into the database.
Basic flow	1. The system displays the booking page. 2. The customer chooses the date, time, type of table.

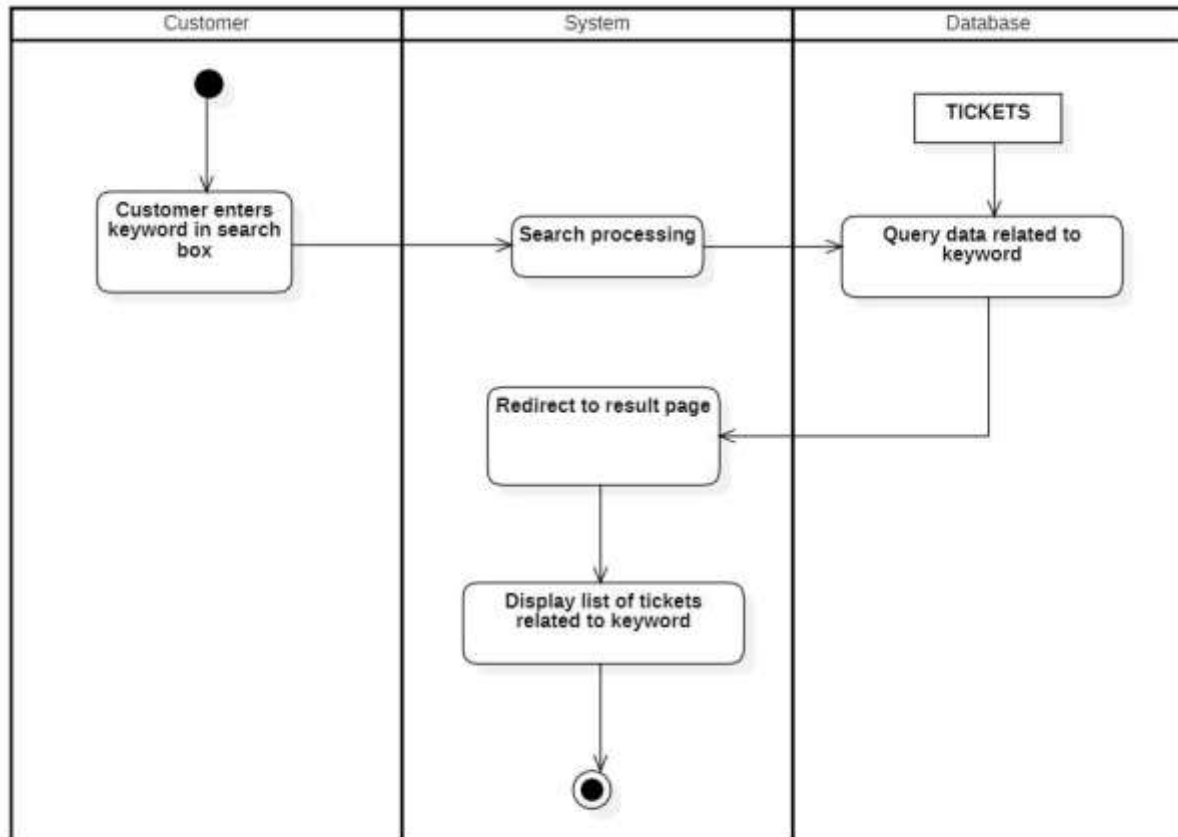
	<ol style="list-style-type: none"> 3. The system processes booking requests. 4. The system query for the suitable place. 5. The system displays the menu including the type of ticket and meal information. 6. The customer chooses the type of tickets. 7. The system displays check out information. 8. The customer confirms the payment. 9. The system processes the payment and stores it in the database. 10. The system displays the successful notification and send the invoice email.
Alternative flow	3a. If there are no suitable places the system displays no suitable place notification. The use case continues at step 2.
Exception flow	



3.2.5 Search

Use-case name	Search ticket
Description	Customers search a ticket including some words
Trigger	Customers click to button "Search" button
Pre-condition	Customer's devices must connect to Internet
Post-condition	Customer gets a list of tickets with detail information
Basic flow	<ol style="list-style-type: none"> 1. Customers enter keywords on text field then click "Search" button (with magnifying glass icon). 2. System processing request. 3. System queries list of tickets related to keywords. 4. System redirect customers to result page. 5. System display result with hyperlinks to tickets.

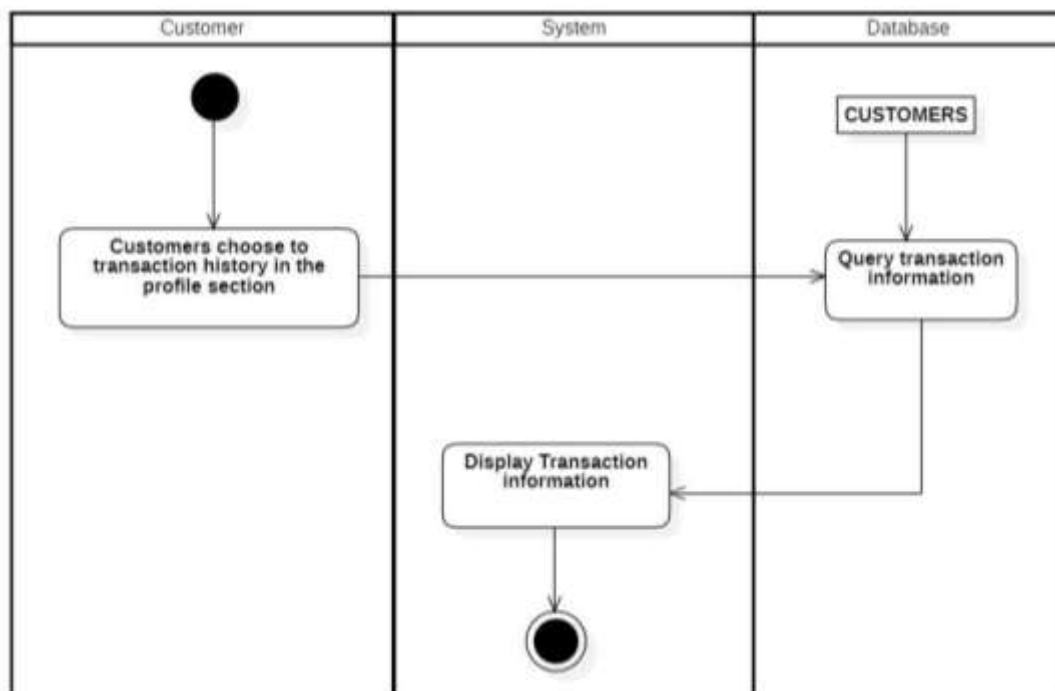
Alternative flow	
Exception flow	



3.2.6 Review transaction history

Use-case name	Review transaction history
Description	Customers can view transaction history information.
Trigger	Customers choose to transaction history in the profile section.
Pre-condition	Customer's device must connect to Internet Customer's account is signed in Customer's account existed.
Post-condition	Customers get transaction history with detailed information.
Basic flow	1. Customers choose to transaction history in the profile section.

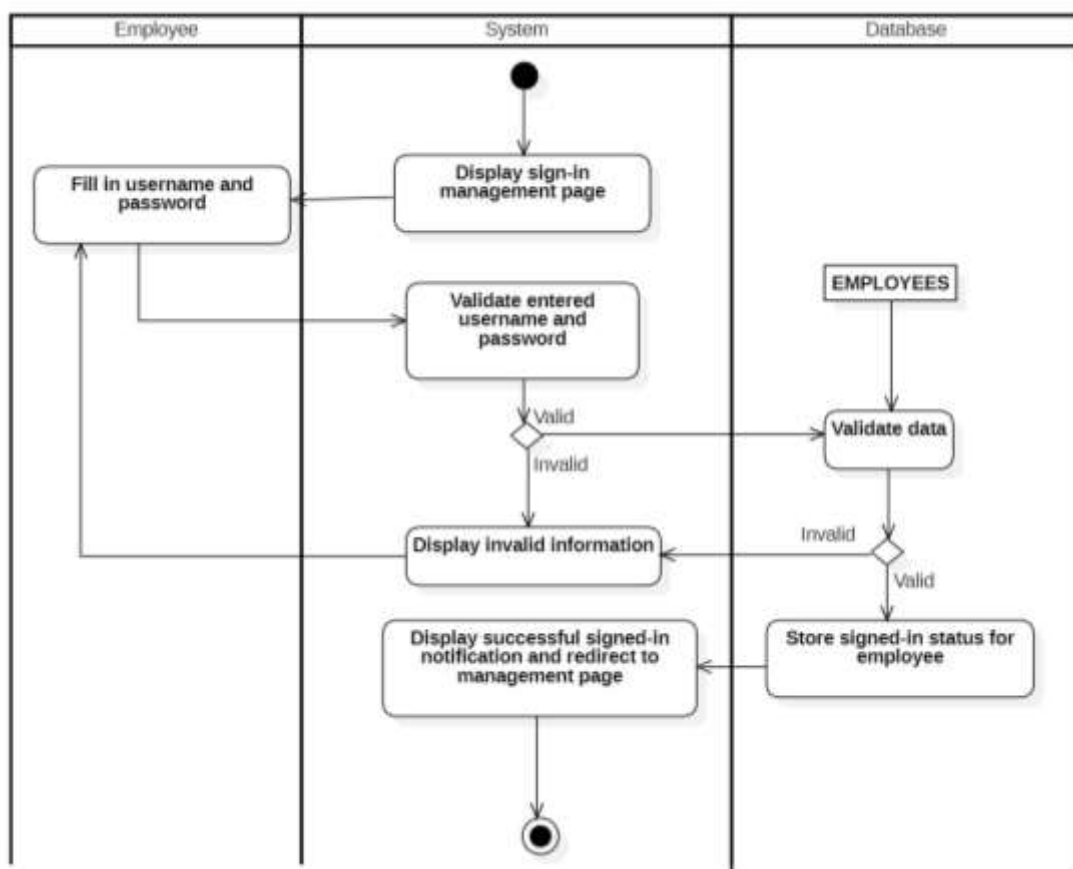
	<ol style="list-style-type: none"> System queries customer information and their transaction history based on signed-in status. System displays transaction history on the page with detailed information.
Alternative flow	
Exception flow	



3.2.7 Sign in (for administrator)

Use-case name	Sign in for administrator
Description	Administrator uses an existing management account to log in to system for all monitoring.
Trigger	Administrator clicks to button "Sign in" at management page.
Pre-condition	Administrator's device must connect to Internet Administrator's account is existed.
Post-condition	Administrator gets a notification about signed-in successful. Administrator is redirected to management page.
Basic flow	<ol style="list-style-type: none"> Administrator clicks to button "Sign in".

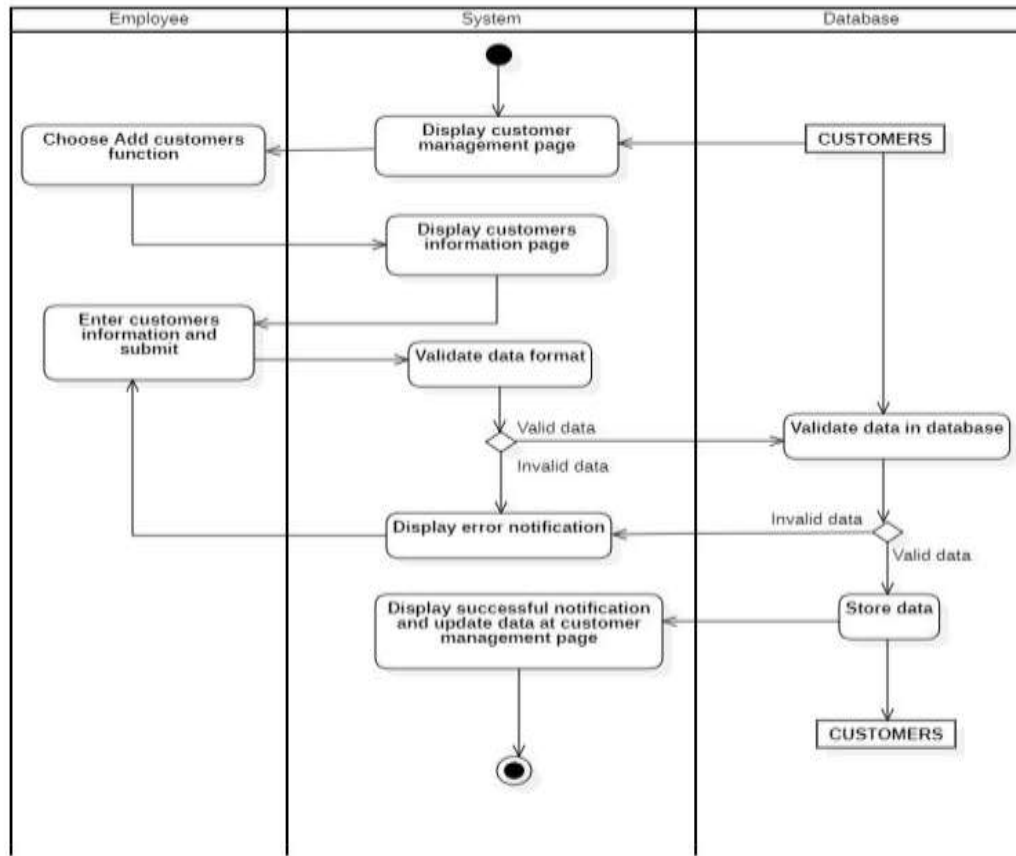
	<ol style="list-style-type: none"> 2. System displays management sign-in page. 3. Administrator enters username and password. 4. System validates administrator account in the correct format. 5. System validates administrator' data with existed administrator account in database. 6. System stores signed-in status of administrators using sessions. 7. System displays successful signed-in notification and redirect to management page.
Alternative flow	<p>4a. System display invalid information due to wrong format. Use-case continues at step 3.</p> <p>5a. System display invalid information due to administrator' account does not match with existed account information in database. Use-case continues at step 3.</p>
Exception flow	



3.2.8. Manage customers

3.2.8.1 Add customers

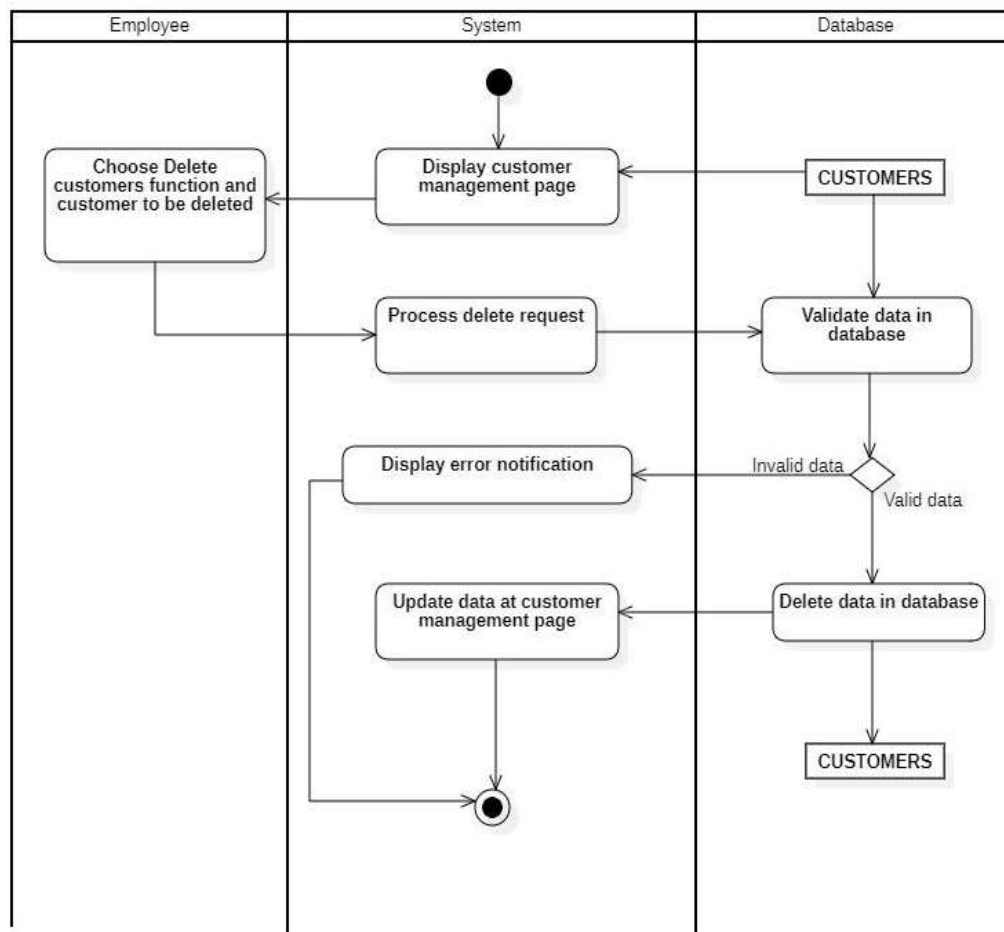
Use-case name	Add customers (for administrator)
Description	Administrators can add new customers to the database.
Trigger	Administrator clicks to button "Add customers" at customer management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed in.
Post-condition	Customer information is added to the database. Data at customer management page is updated with new customers.
Basic flow	<ol style="list-style-type: none">1. System display customer management page.2. The administrator chooses to add the customer's function.3. System displays customers information page.4. Administrators enter customers' information and submit.5. System validates customers' information in correct format.6. System validates customers' information with existing data in database.7. The system stores customers information in a database.8. System displays successful notification and updates data at management page.
Alternative flow	<p>5a. System displays invalid information due to wrong format. Use-case continues at step 4.</p> <p>6a. The system displays invalid information because data had been existed in the database. Use-case continues at step 4.</p>
Exception flow	



3.2.8.2 Delete customers

Use-case name	Delete customers (for administrator)
Description	Administrators can delete customers' information from the database.
Trigger	Administrator clicks to button "Delete customers" at customer management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed in. Customers' information must be existed in database. No external data involved.
Post-condition	Data at customer management page is updated.
Basic flow	<ol style="list-style-type: none"> 1. System display customer management page. 2. Administrator chooses to delete customer's function and customer to be deleted. 3. System processes data deleted request. 4. System validates customer information that needs to be deleted.

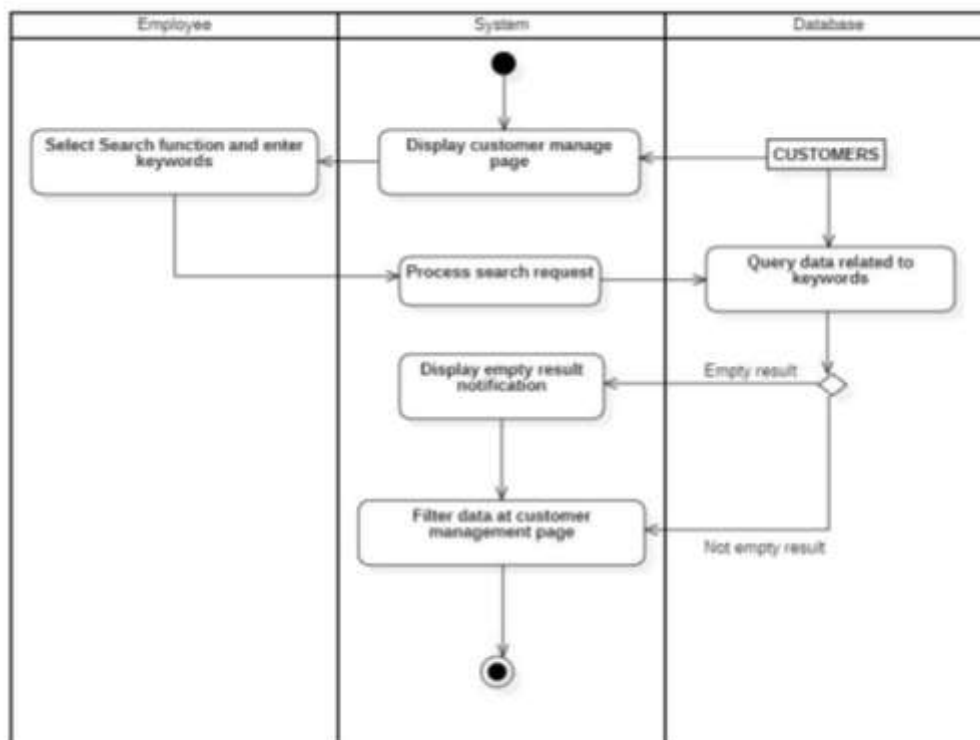
	<p>5. System deletes customer information in the database.</p> <p>6. System updates data at customer management page.</p>
Alternative flow	5a. System display error notification because no data existed, or external data had required that customer information. Use-case stops.
Exception flow	



3.2.8.3 Search customers

Use-case name	Search customers (for administrator)
Description	Administrator can search customers in the database
Trigger	Administrators clicks to button "Search" at customers management page
Pre-condition	Administrators' devices must connect to the Internet.

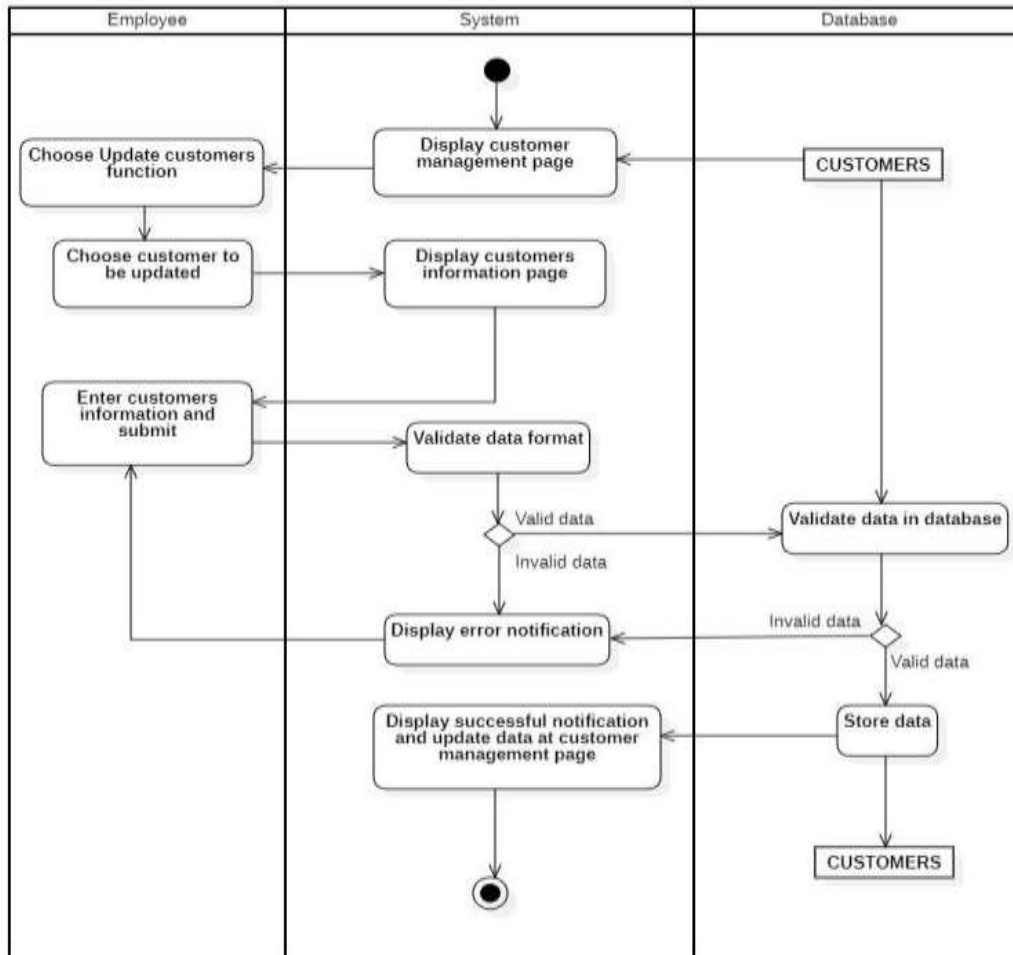
	Administrators' account existed. Administrator's status is signed in.
Post-condition	Data at management page is filtered by keywords
Basic flow	<ol style="list-style-type: none"> 1. System display customer management page with data loaded from database (existed customers) 2. Administrators select search function and enter keywords. 3. System processes searching data. 4. System queries data based on keywords. 5. System updates data at customer management page (data related to keyword is retained)
Alternative flow	4a. The system displays empty result notification. Use-case continue at step 5 with no data is retained
Exception flow	



3.2.8.4 Update customers

Use-case name	Update customers (for administrator)
---------------	--------------------------------------

Description	Administrators can update to the database.
Trigger	Administrator clicks to button "Update customers" at customer management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed in. Customers information must be existed in database before updating.
Post-condition	Customer information updated to the database.
Basic flow	<ol style="list-style-type: none"> 1. System display customer management page. 2. Administrator chooses to update customer's function. 3. Administrator chooses customers to be updated. 4. System displays customers information page. 5. Administrators enter customers' information and submit. 6. System validates customers' information in correct format. 7. System validates customers' information with existing data in database. 8. The system stores customers' information in a database. 9. System displays successful notification and updates data at customer management page.
Alternative flow	<p>6a. System displays invalid information due to wrong format. Use-case continues at step 5.</p> <p>7a. The system displays invalid information because data had existed in the database. Use-case continues at step 5.</p>
Exception flow	

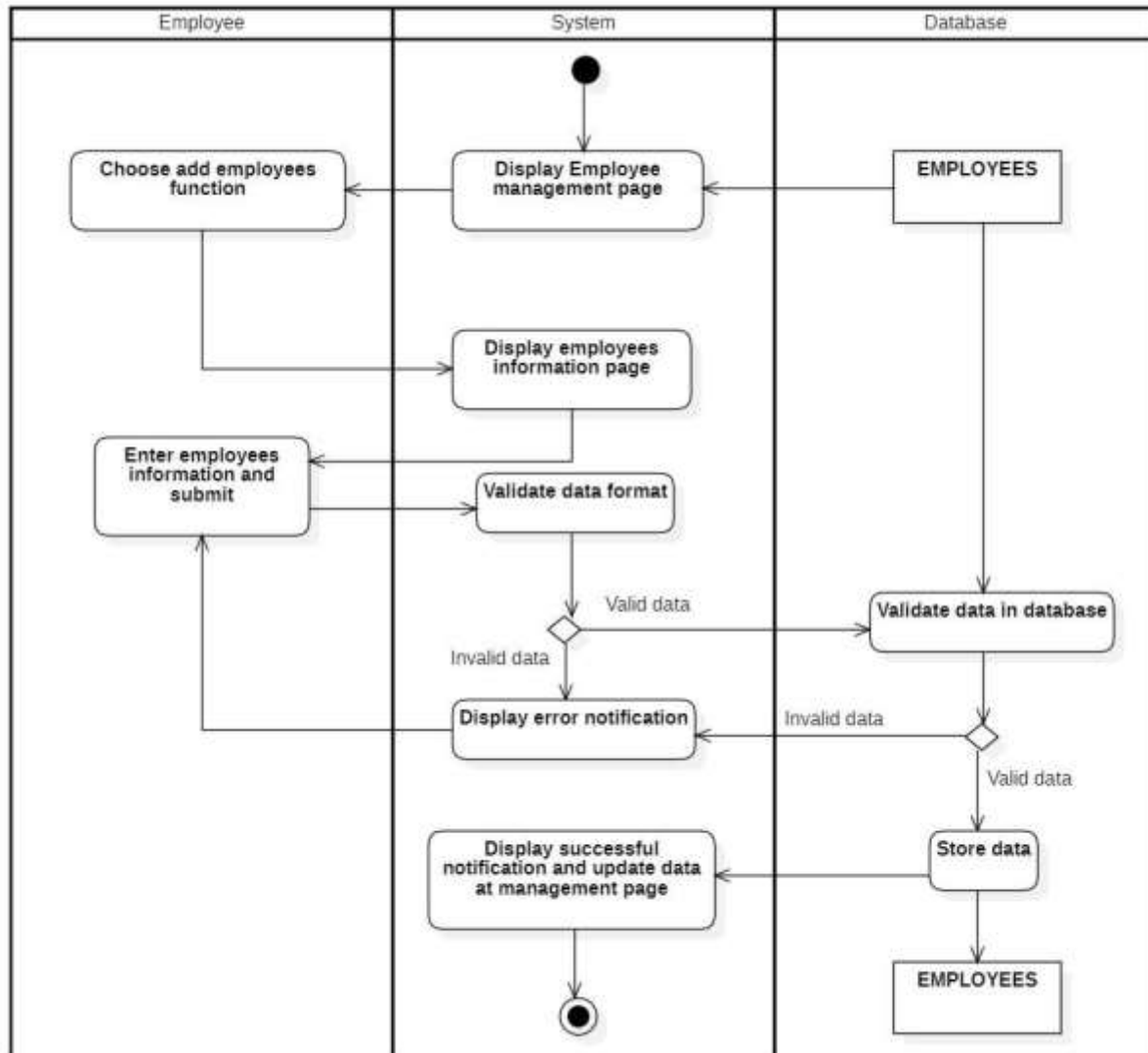


3.2.12 Manage Employees

3.2.12.1 Add employee

Use-case name	Add employee (for administrator)
Description	Administrator can add new employee to the database.
Trigger	Administrator clicks to button "Add employee" at employee management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in.
Post-condition	Employee information successful add to the database. Data at employee management page is updated (with new employee).
Basic flow	<ol style="list-style-type: none"> 1. System display employee management page. 2. Administrator chooses add employee function. 3. System display employee information page. 4. Administrator enter employee information and submit. 5. System validate employee information in correct format.

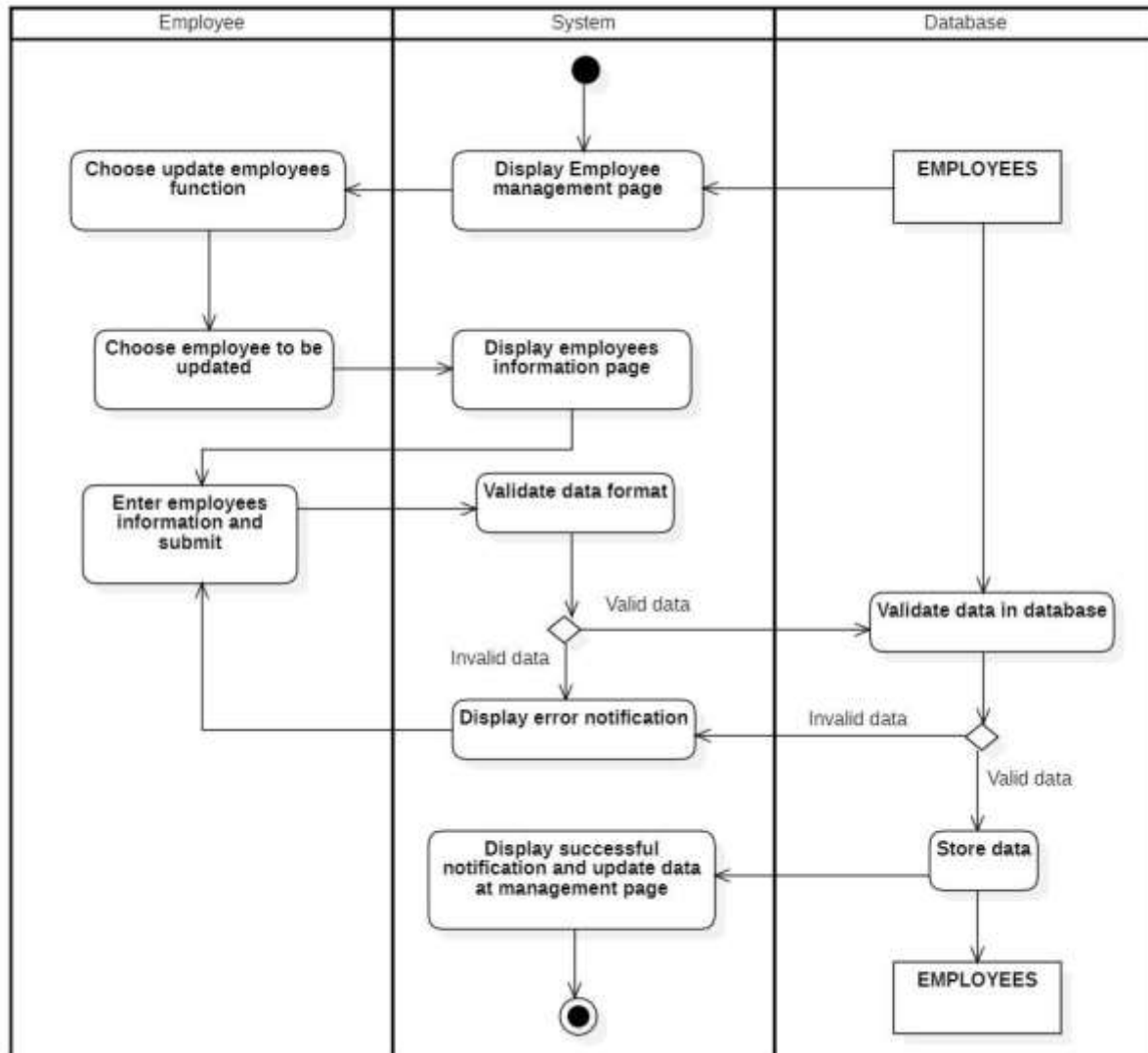
	6. System validate employee information with existing data in database. 7. System stores employee information in database. 8. System displays successful notification and updates data at management page (new data).
Alternative flow	5a. System display invalid information due to wrong format. Use-case continues at step 4. 6a. System display invalid information because data had been existed in the database. Use-case continues at step 4.
Exception flow	



3.2.12.2 Update employee

Use-case name	Update employee (for administrator)
Description	Administrator can update to the database.
Trigger	Administrator clicks to button "Update employee" at ingredient management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in. Employee information must be existed in database before updating.
Post-condition	Employee information updated to the database.

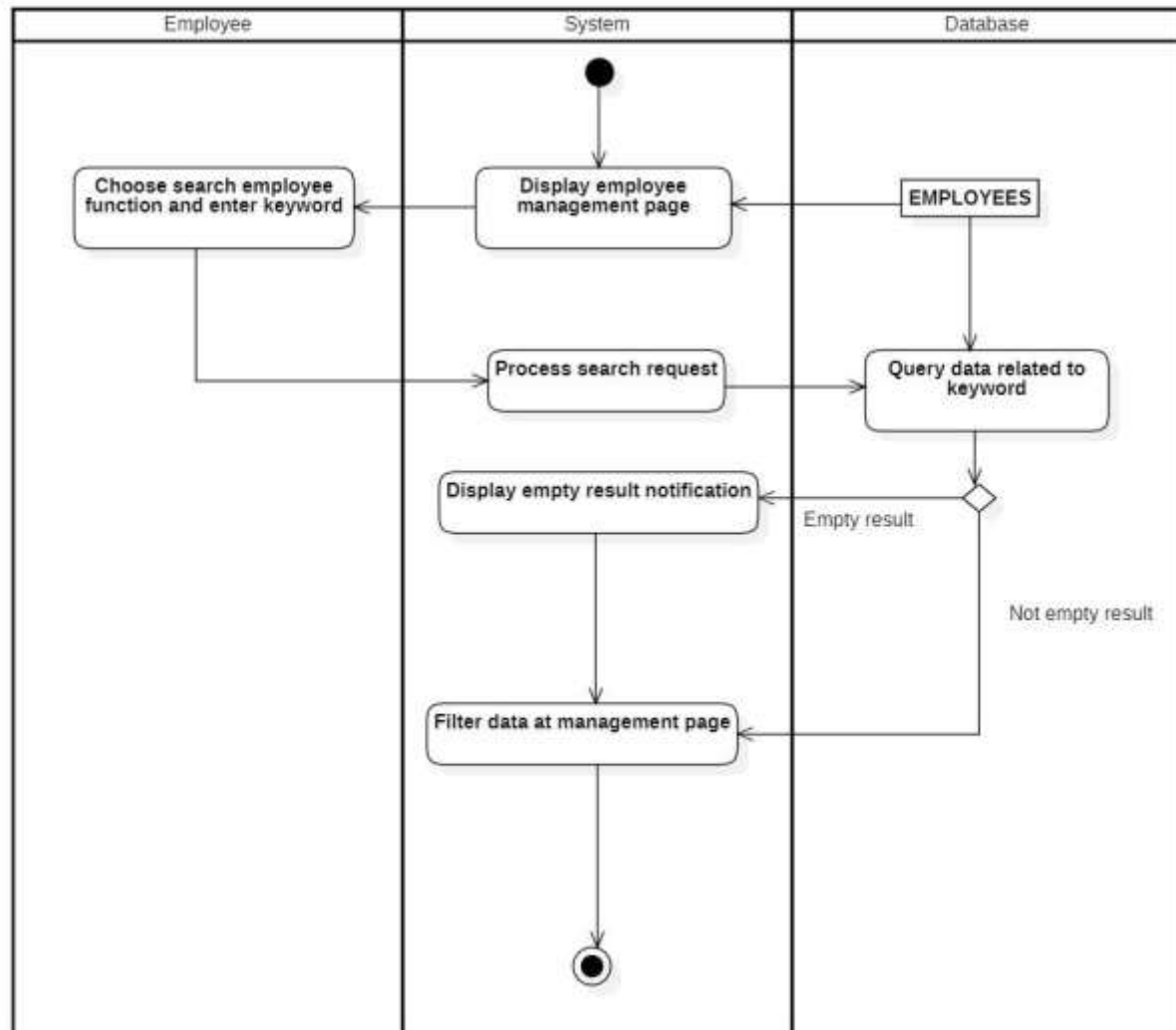
Basic flow	<ol style="list-style-type: none"> 1. System display employee management page. 2. Administrator chooses update employee function. 3. Administrator chooses employee to be updated. 4. System display employee information page. 5. Administrator enter employee information and submit. 6. System validate employee information in correct format. 7. System validate employee information with existing data in database. 8. System stores employee information in database. 9. System displays successful notification and updates data at management page.
Alternative flow	<p>6a. System display invalid information due to wrong format. Use-case continues at step 5.</p> <p>7a. System display invalid information because data had been existed in the database. Use-case continues at step 5.</p>
Exception flow	



3.2.12.3 Search employee

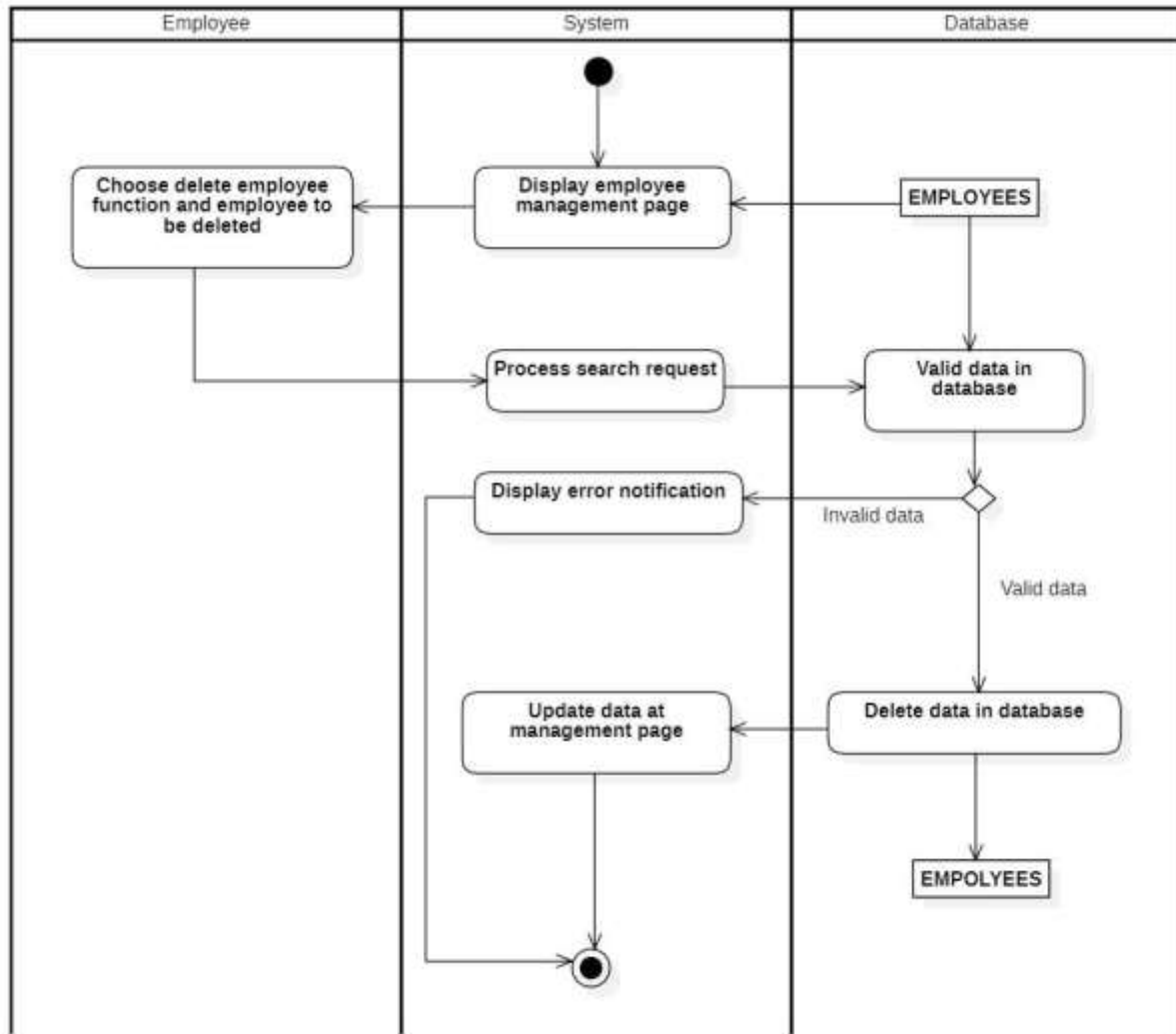
Use-case name	Search employee (for administrator)
Description	Administrator can search employee from the database.
Trigger	Administrator clicks to button "Search employee" at restaurant management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in. Employee information must be existed in database.
Post-condition	Data at employee management page is filtered by keywords.
Basic flow	1. System display employee management page.

	2. Administrator chooses search employee function and enters keywords. 3. System processes data search request. 4. System queries data related to keywords. 1. System updates data at management page.
Alternative flow	4a. System display empty result notification. Use-case continues at step 5.
Exception flow	



3.2.12.4 Delete employee

Use-case name	Delete employee (for administrator)
Description	Administrator can delete employee information in the database.
Trigger	Administrator clicks to button "Delete employee" at employee management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in. Employee information must be existed in database. No external data involved.
Post-condition	Data at employee management page is updated.
Basic flow	<ol style="list-style-type: none">1. System display employee management page.2. Administrator chooses delete employee function and employee to be deleted.3. System processes data delete request.4. System validates employee information that need to be deleted.5. System delete ingredient information in the database.6. System updates data at management page.
Alternative flow	5a. System display error notification because no data existed or external data had required that employee information. Use-case stops.
Exception flow	

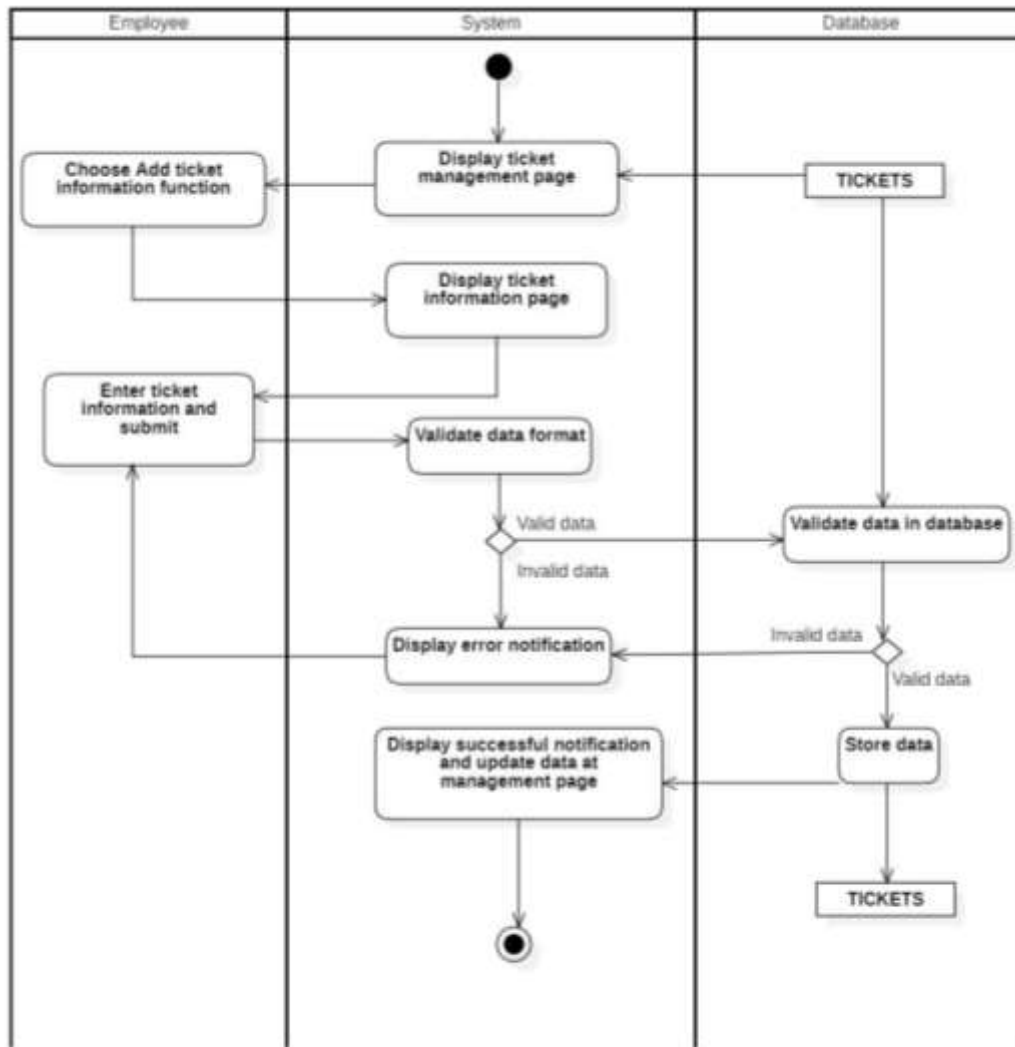


3.2.10 Manage tickets

3.2.10.1 Add tickets

Use-case name	Add tickets (for administrator)
Description	Administrator can add new tickets to the database.
Trigger	Administrator clicks to button "Add tickets" at tickets management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in.
Post-condition	Tickets information add to the database. Data at tickets management page is updated with new ingredients.
Basic flow	1. System display tickets management page. 2. Administrator chooses add tickets function.

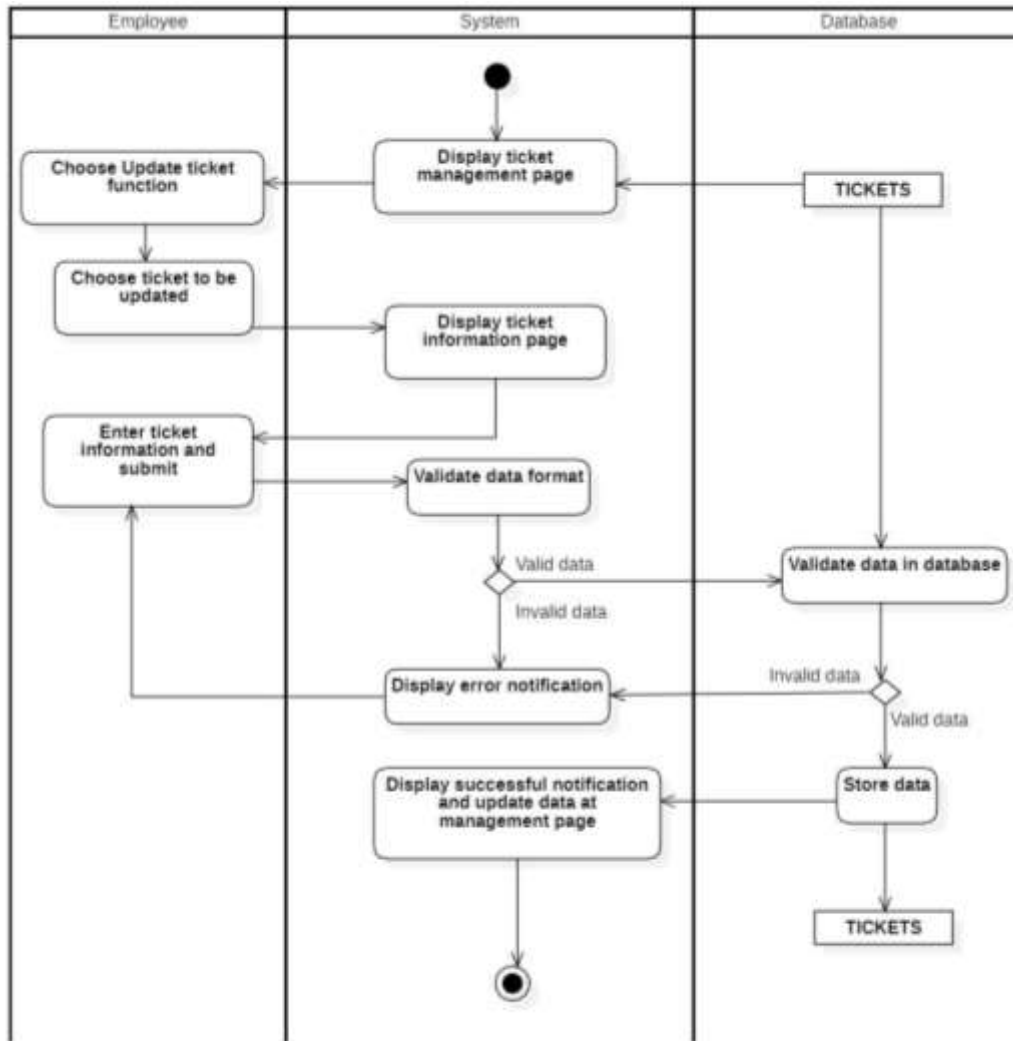
	<ul style="list-style-type: none"> 3. System display tickets information page. 4. Administrator enter tickets information and submit. 5. System validate tickets information in correct format. 6. System validate tickets information with existing data in database. 7. System stores tickets information in database. 8. System displays successful notification and updates data at management page.
Alternative flow	<p>5a. System display invalid information due to wrong format. Use-case continues at step 4.</p> <p>6a. System display invalid information because data had been existed in the database. Use-case continues at step 4.</p>
Exception flow	



3.2.10.2 Update tickets

Use-case name	Update tickets (for administrator)
Description	Administrator can update to the database.
Trigger	Administrator clicks to button "Update tickets" at tickets management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in. Tickets information must be existed in database before updating.
Post-condition	Tickets information updated to the database.
Basic flow	1. System display tickets management page.

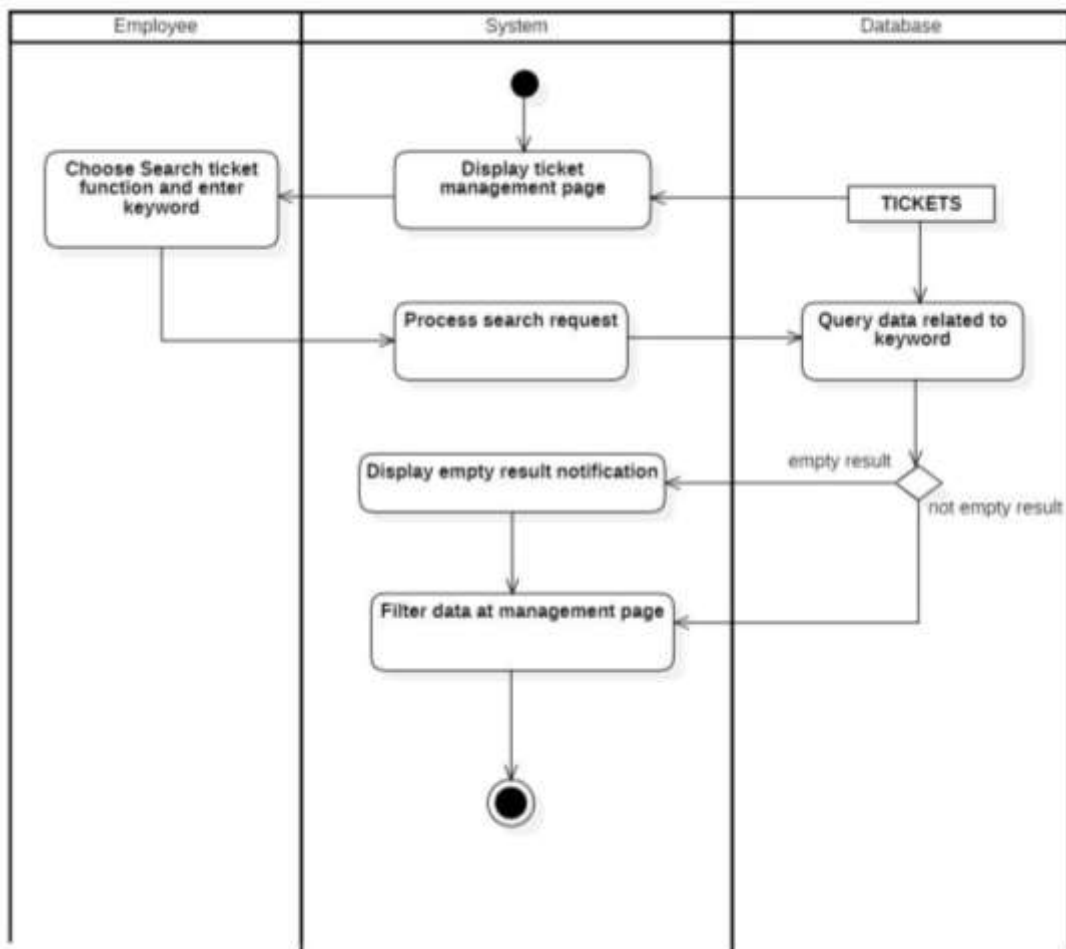
	<ol style="list-style-type: none"> 2. Administrator chooses update tickets function. 3. Administrator chooses tickets to be updated. 4. System display tickets information page. 5. Administrator enter tickets information and submit. 6. System validate tickets information in correct format. 7. System validate tickets information with existing data in database. 8. System stores tickets information in database. 9. System displays successful notification and updates data at management page.
Alternative flow	<p>6a. System display invalid information due to wrong format. Use-case continues at step 5.</p> <p>7a. System display invalid information because data had been existed in the database. Use-case continues at step 5.</p>
Exception flow	



3.2.10.3 Search tickets

Use-case name	Search tickets (for administrator)
Description	Administrator can search tickets from the database.
Trigger	Administrator clicks to button "Search tickets" at restaurant management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in. Ingredients information must be existed in database.
Post-condition	Data at tickets management page is filtered by keywords.
Basic flow	<ol style="list-style-type: none"> 1. System display tickets management page. 2. Administrator chooses search tickets function and enters keywords.

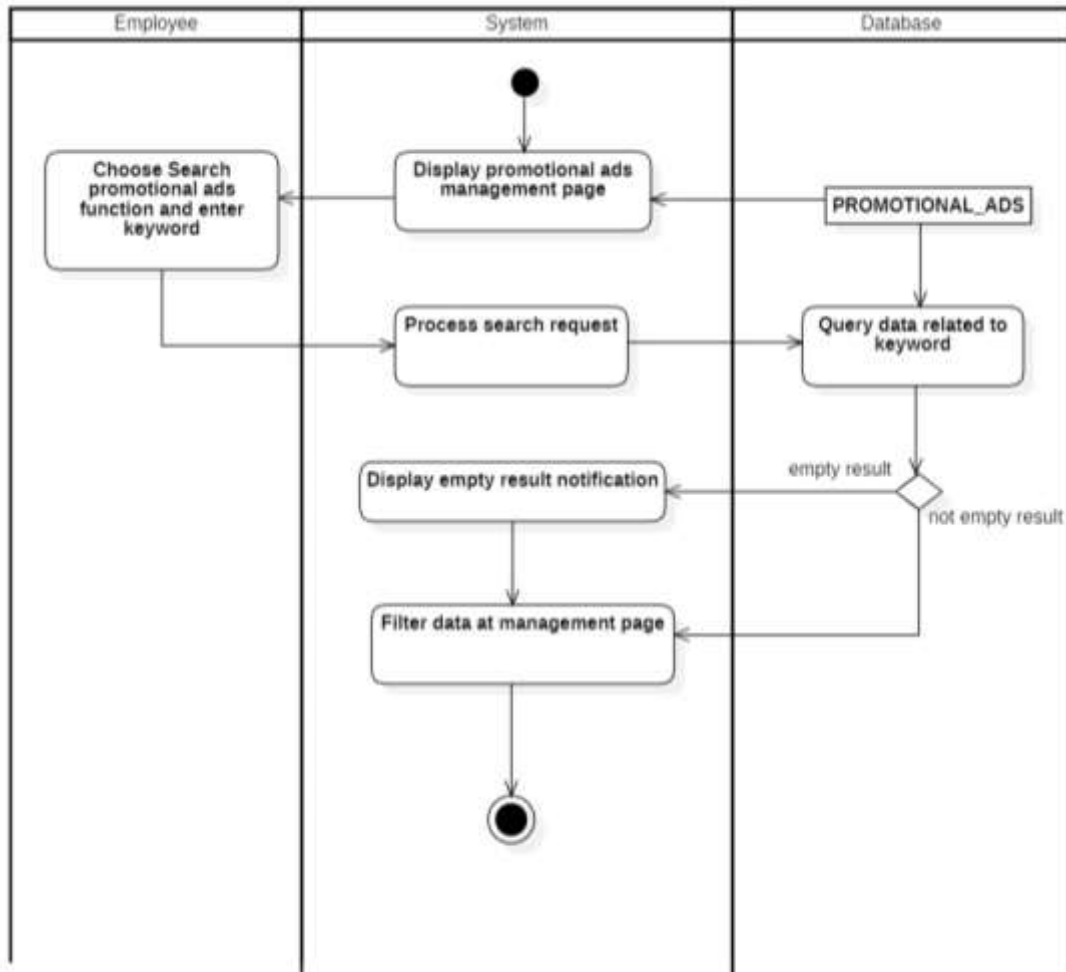
	3. System processes data search request. 4. System queries data related to keywords. 5. System updates data at management page.
Alternative flow	4a. System display empty result notification. Use-case continues at step 5.
Exception flow	



3.2.10.4 Delete tickets

Use-case name	Delete tickets (for administrator)
Description	Administrator can delete tickets information in the database.

Trigger	Administrator clicks to button "tickets ingredients" at tickets management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in. Ingredients information must be existed in database. No external data involved.
Post-condition	Data at tickets management page is updated.
Basic flow	<ol style="list-style-type: none"> 1. System display tickets management page. 2. Administrator chooses delete tickets function and tickets to be deleted. 3. System processes data delete request. 4. System validates tickets information that need to be deleted. 5. System delete tickets information in the database. 6. System updates data at management page.
Alternative flow	5a. System display error notification because no data existed or external data had required that ingredient information. Use-case stops.
Exception flow	

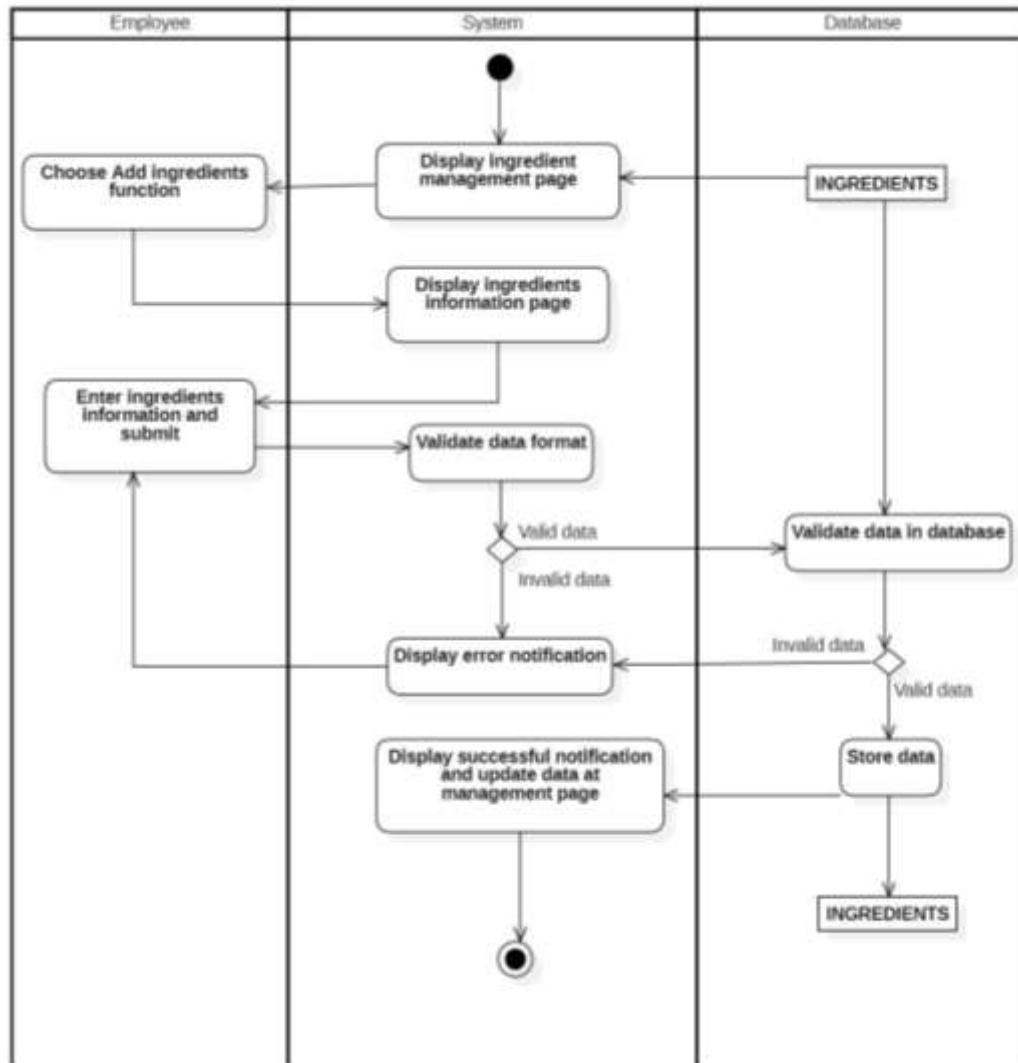


3.2.11 Manage ingredients

3.2.11.1 Add ingredients

Use-case name	Add ingredients (for administrator)
Description	Administrator can add new ingredients to the database.
Trigger	Administrator clicks to button "Add ingredients" at ingredient management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in.
Post-condition	Ingredient information add to the database. Data at ingredient management page is updated with new ingredients.
Basic flow	<ol style="list-style-type: none"> 1. System display ingredient management page. 2. Administrator chooses add ingredients function. 3. System display ingredients information page.

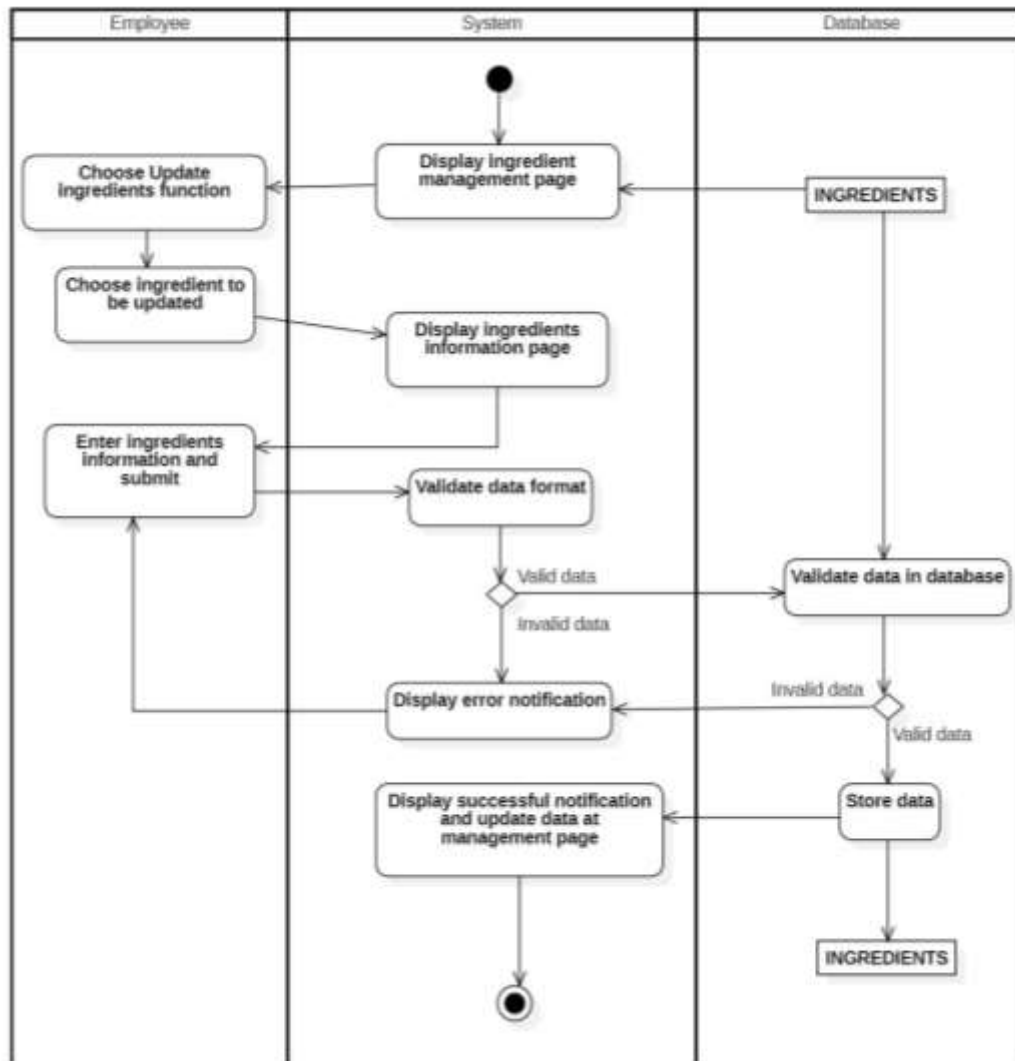
	<ol style="list-style-type: none"> 4. Administrator enter ingredients information and submit. 5. System validate ingredients information in correct format. 6. System validate ingredients information with existing data in database. 7. System stores ingredients information in database. 8. System displays successful notification and updates data at management page.
Alternative flow	<p>5a. System display invalid information due to wrong format. Use-case continues at step 4.</p> <p>6a. System display invalid information because data had been existed in the database. Use-case continues at step 4.</p>
Exception flow	



3.2.11.2 Update ingredients

Use-case name	Update ingredients (for administrator)
Description	Administrator can update to the database.
Trigger	Administrator clicks to button "Update ingredients" at ingredient management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in. Ingredients information must be existed in database before updating.
Post-condition	Ingredient information updated to the database.
Basic flow	1. System display ingredient management page.

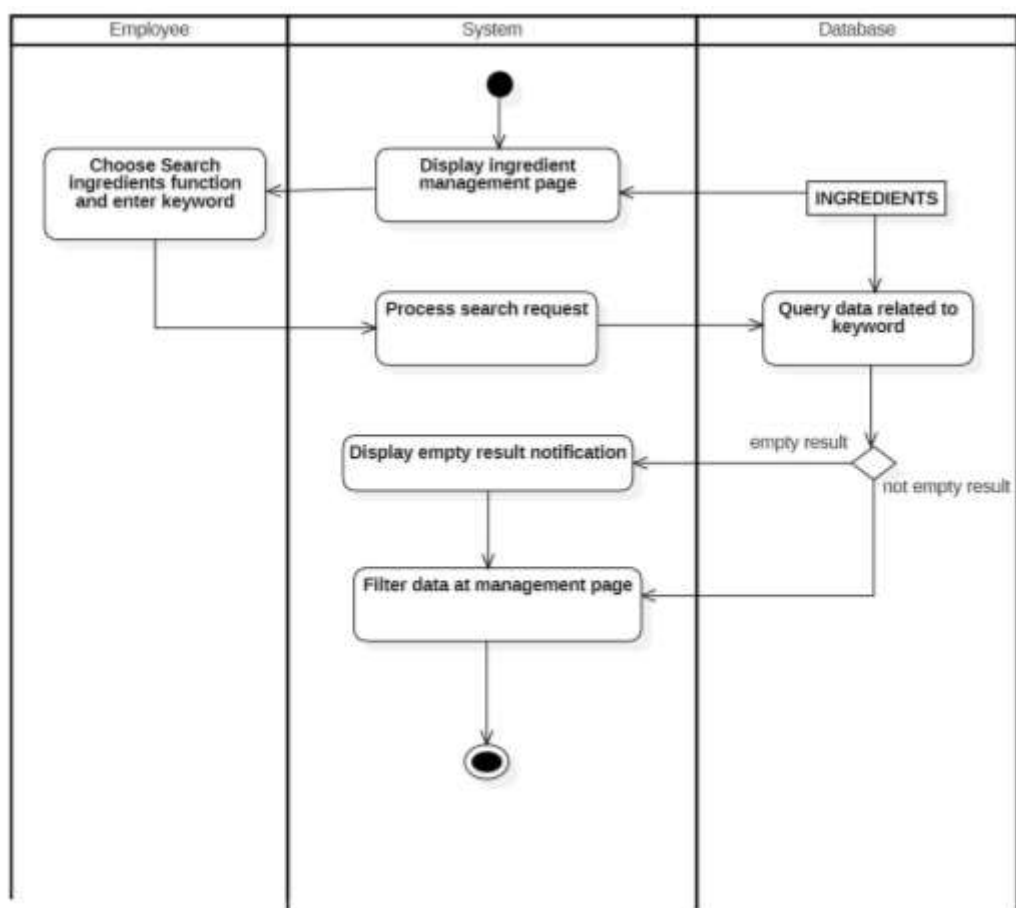
	<ol style="list-style-type: none"> 2. Administrator chooses update ingredients function. 3. Administrator chooses ingredients to be updated. 4. System display ingredients information page. 5. Administrator enter ingredients information and submit. 6. System validate ingredients information in correct format. 7. System validate ingredients information with existing data in database. 8. System stores ingredients information in database. 9. System displays successful notification and updates data at management page.
Alternative flow	<p>6a. System display invalid information due to wrong format. Use-case continues at step 5.</p> <p>7a. System display invalid information because data had been existed in the database. Use-case continues at step 5.</p>
Exception flow	



3.2.11.3 Search ingredients

Use-case name	Search ingredients (for administrator)
Description	Administrator can search ingredients from the database.
Trigger	Administrator clicks to button "Search ingredients" at restaurant management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in. Ingredients information must be existed in database.
Post-condition	Data at ingredient management page is filtered by keywords.
Basic flow	1. System display ingredient management page.

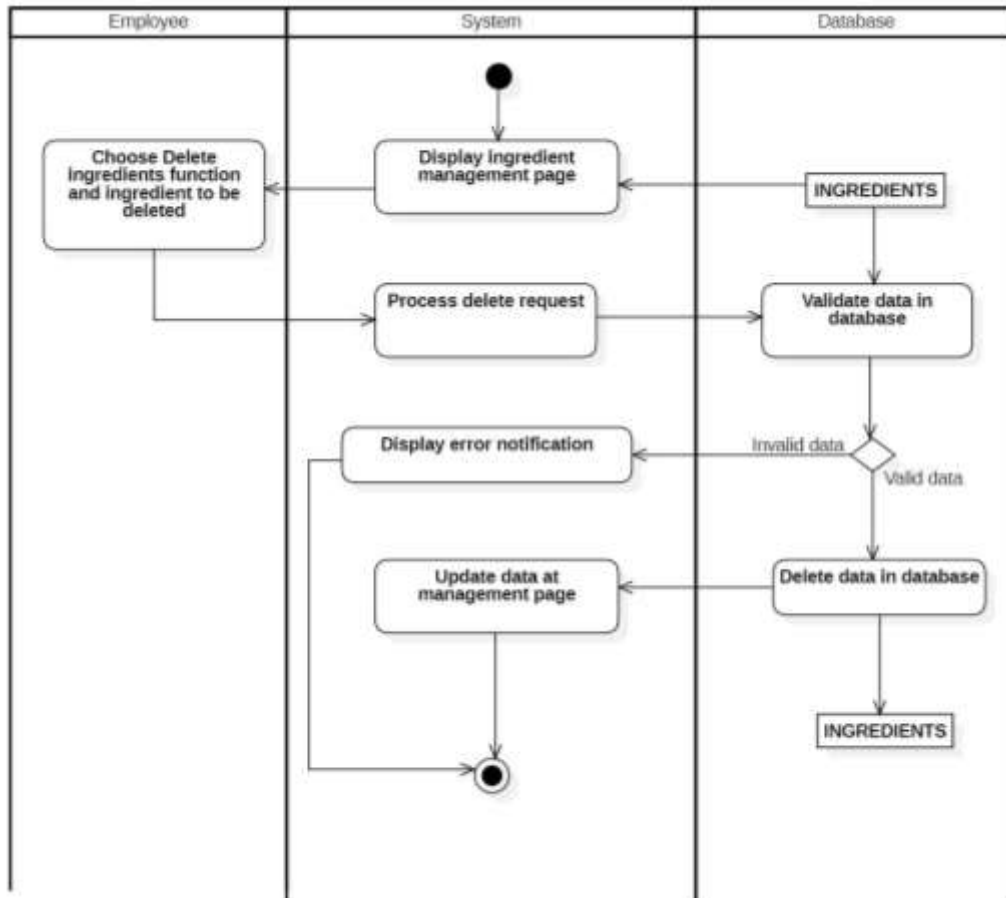
	<ol style="list-style-type: none"> 2. Administrator chooses search ingredients function and enters keywords. 3. System processes data search request. 4. System queries data related to keywords. 5. System updates data at management page.
Alternative flow	4a. System display empty result notification. Use-case continues at step 5.
Exception flow	



3.2.11.4 Delete ingredients

Use-case name	Delete ingredients (for administrator)
---------------	--

Description	Administrator can delete ingredients information in the database.
Trigger	Administrator clicks to button "Delete ingredients" at ingredient management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in. Ingredients information must be existed in database. No external data involved.
Post-condition	Data at ingredient management page is updated.
Basic flow	<ol style="list-style-type: none"> 1. System display ingredient management page. 2. Administrator chooses delete ingredients function and ingredient to be deleted. 3. System processes data delete request. 4. System validates ingredient information that need to be deleted. 5. System delete ingredient information in the database. 6. System updates data at management page.
Alternative flow	5a. System display error notification because no data existed or external data had required that ingredient information. Use-case stops.
Exception flow	

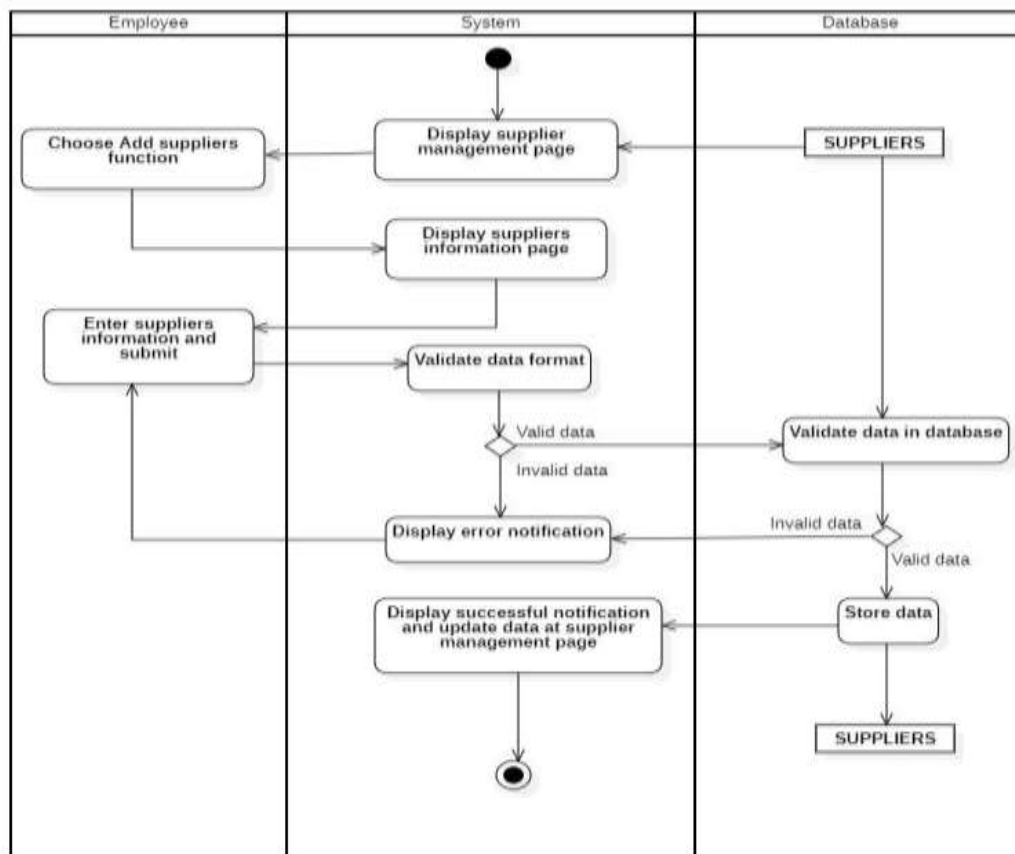


3.2.12 Manage suppliers

3.2.12.1 Add suppliers

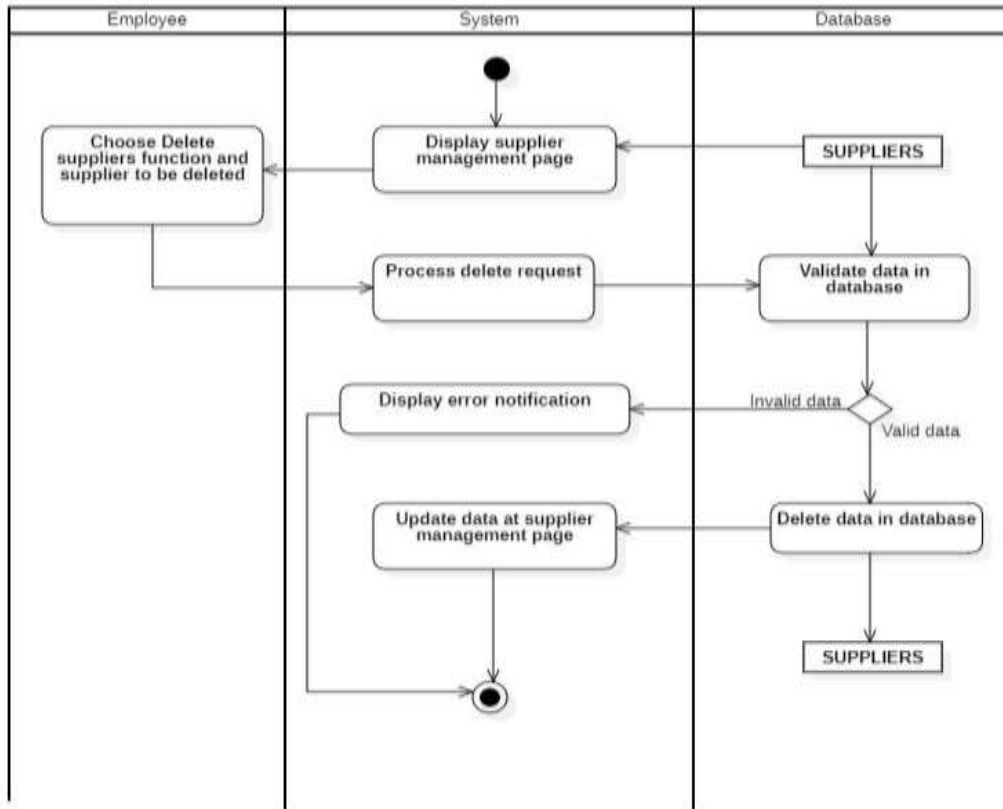
Use-case name	Add suppliers (for administrator)
Description	Administrators can add new suppliers to the database.
Trigger	Administrator clicks to button "Add suppliers" at supplier management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed in.
Post-condition	Supplier information is added to the database. Data at supplier management page is updated with new suppliers.
Basic flow	<ol style="list-style-type: none"> 1. System display supplier management page. 2. The administrator chooses to add the supplier's function. 3. System displays suppliers information page. 4. Administrators enter suppliers' information and submit.

	<p>5. System validates suppliers' information in correct format.</p> <p>6. System validates suppliers' information with existing data in database.</p> <p>7. The system stores suppliers information in a database.</p> <p>8. System displays successful notification and updates data at management page.</p>
Alternative flow	<p>5a. System displays invalid information due to wrong format. Use-case continues at step 4.</p> <p>6a. The system displays invalid information because data had been existed in the database. Use-case continues at step 4.</p>
Exception flow	



3.2.12.2 Delete suppliers

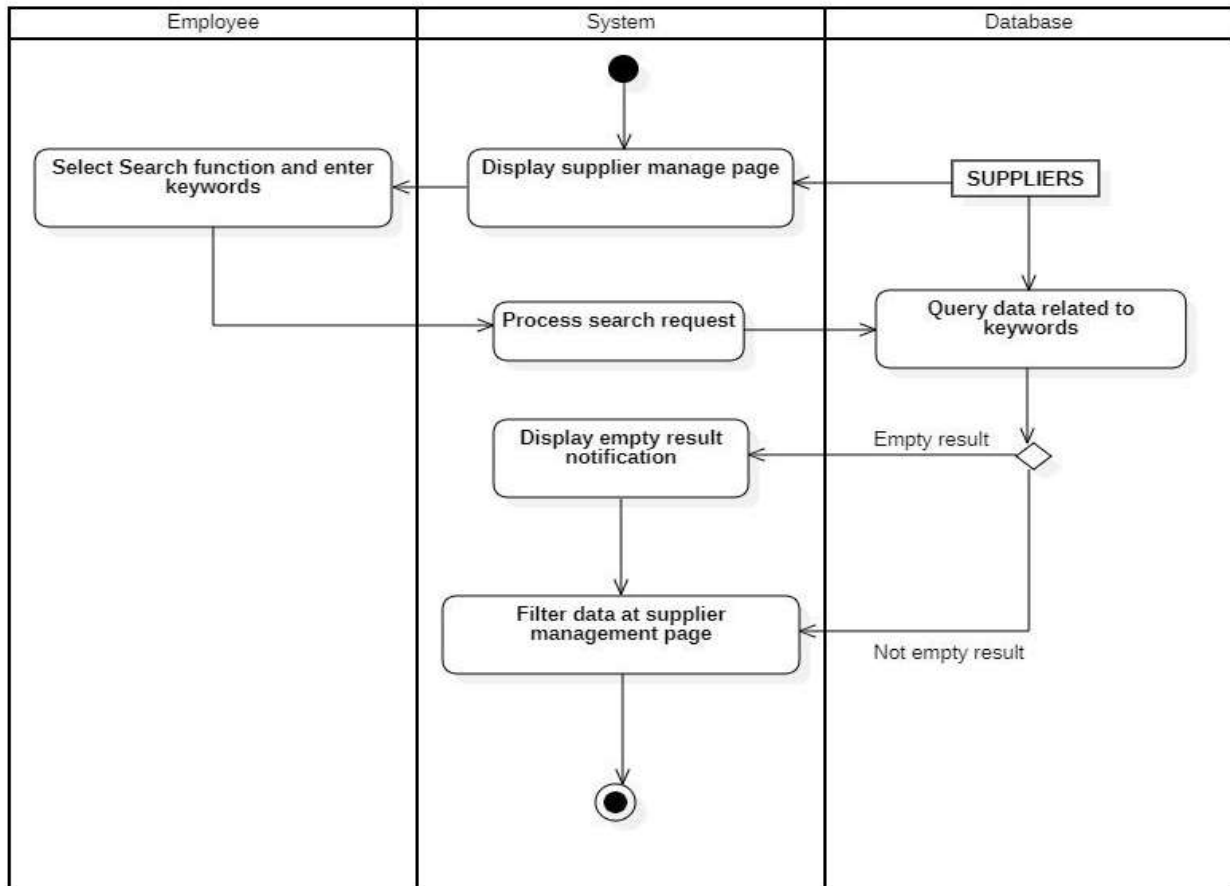
Use-case name	Delete suppliers (for administrator)
Description	Administrators can delete suppliers' information from the database.
Trigger	Administrator clicks to button "Delete suppliers" at supplier management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed in. Suppliers' information must be existed in database. No external data involved.
Post-condition	Data at supplier management page is updated.
Basic flow	<ol style="list-style-type: none">1. System display supplier management page.2. Administrator chooses to delete supplier's function and supplier to be deleted.3. System processes data deleted request.4. System validates supplier information that needs to be deleted.5. System deletes supplier information in the database.6. System updates data at supplier management page.
Alternative flow	5a. System display error notification because no data existed, or external data had required that supplier information. Use-case stops.
Exception flow	



3.2.12.3 Search suppliers

Use-case name	Search suppliers (for administrator)
Description	Administrator can search suppliers in the database
Trigger	Administrators clicks to button "Search" at suppliers management page
Pre-condition	Administrators' devices must connect to the Internet. Administrators' account existed. Administrator'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 supplier management page with data loaded from database (existed suppliers) 2. Administrators select search function and enter keywords. 3. System processes searching data. 4. System queries data based on keywords. 5. System updates data at supplier management page (data related to keyword is retained)

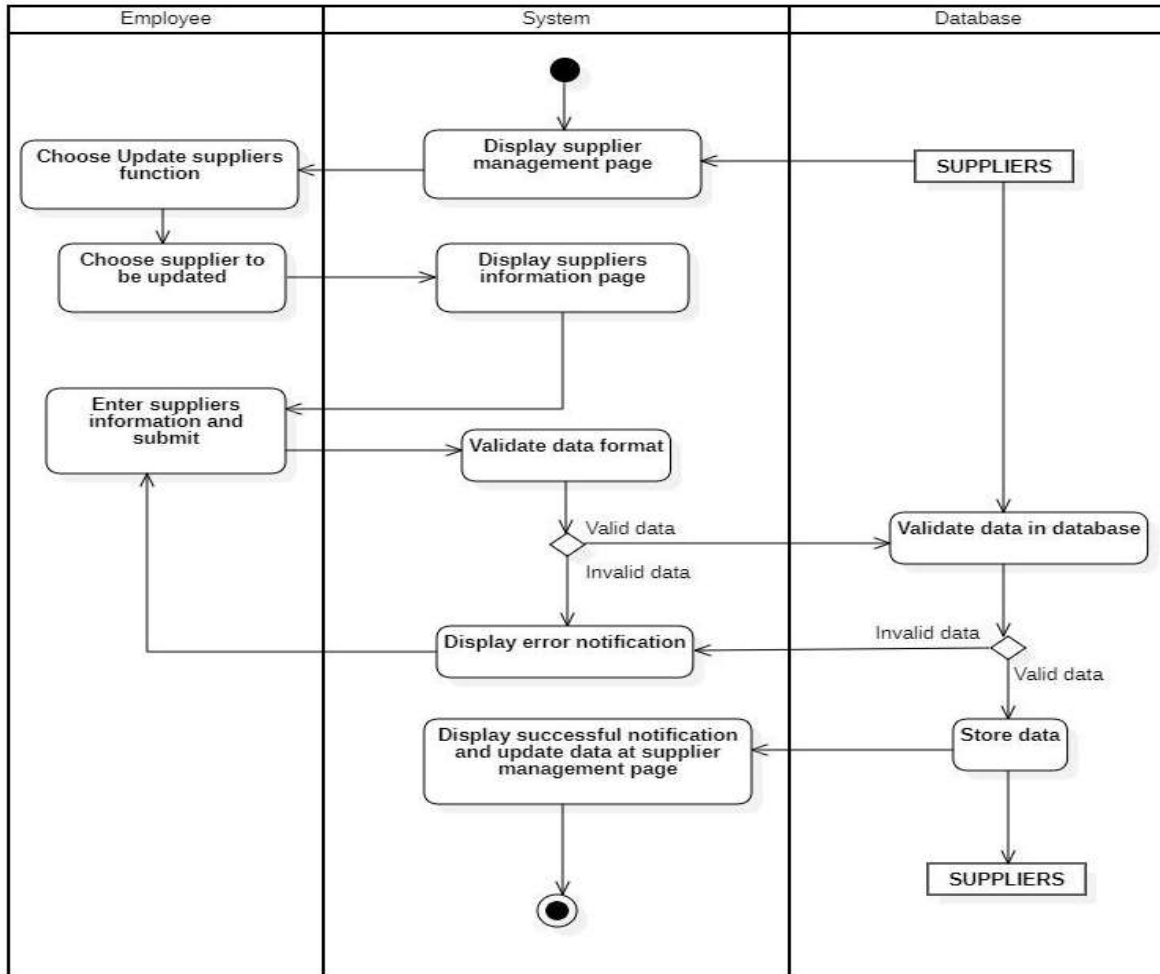
Alternative flow	4a. The system displays empty result notification. Use-case continue at step 5 with no data is retained
Exception flow	



3.2.12.4 Update suppliers

Use-case name	Update suppliers (for administrator)
Description	Administrators can update to the database.
Trigger	Administrator clicks to button "Update suppliers" at supplier management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed in. Suppliers information must be existed in database before updating.

Post-condition	Supplier information updated to the database.
Basic flow	<ol style="list-style-type: none"> 1. System display supplier management page. 2. Administrator chooses to update supplier's function. 3. Administrator chooses suppliers to be updated. 4. System displays suppliers information page. 5. Administrators enter suppliers' information and submit. 6. System validates suppliers' information in correct format. 7. System validates suppliers' information with existing data in database. 8. The system stores suppliers' information in a database. 9. System displays successful notification and updates data at supplier management page.
Alternative flow	<p>6a. System displays invalid information due to wrong format. Use-case continues at step 5.</p> <p>7a. The system displays invalid information because data had existed in the database. Use-case continues at step 5.</p>
Exception flow	

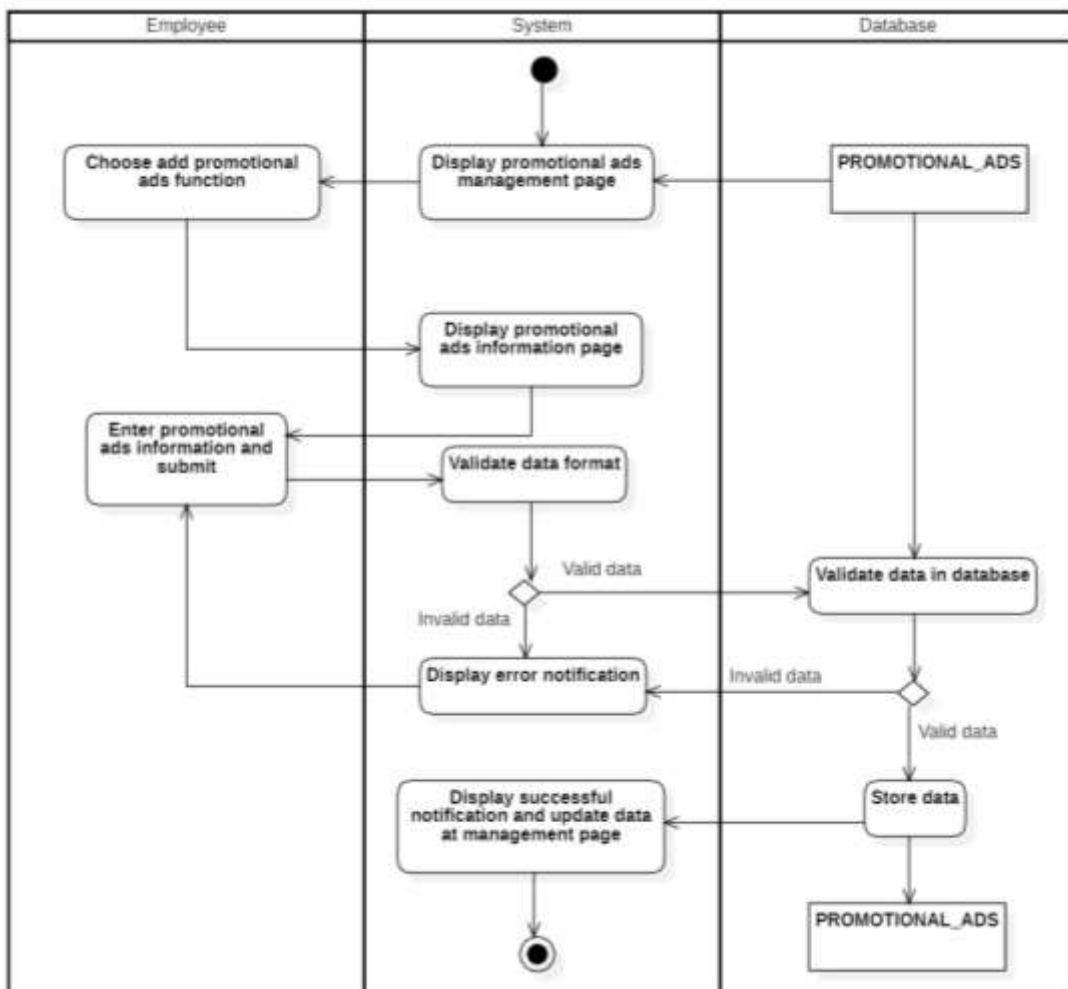


3.2.13 Manage Promotional ads

3.2.13.1 Add promotional ads

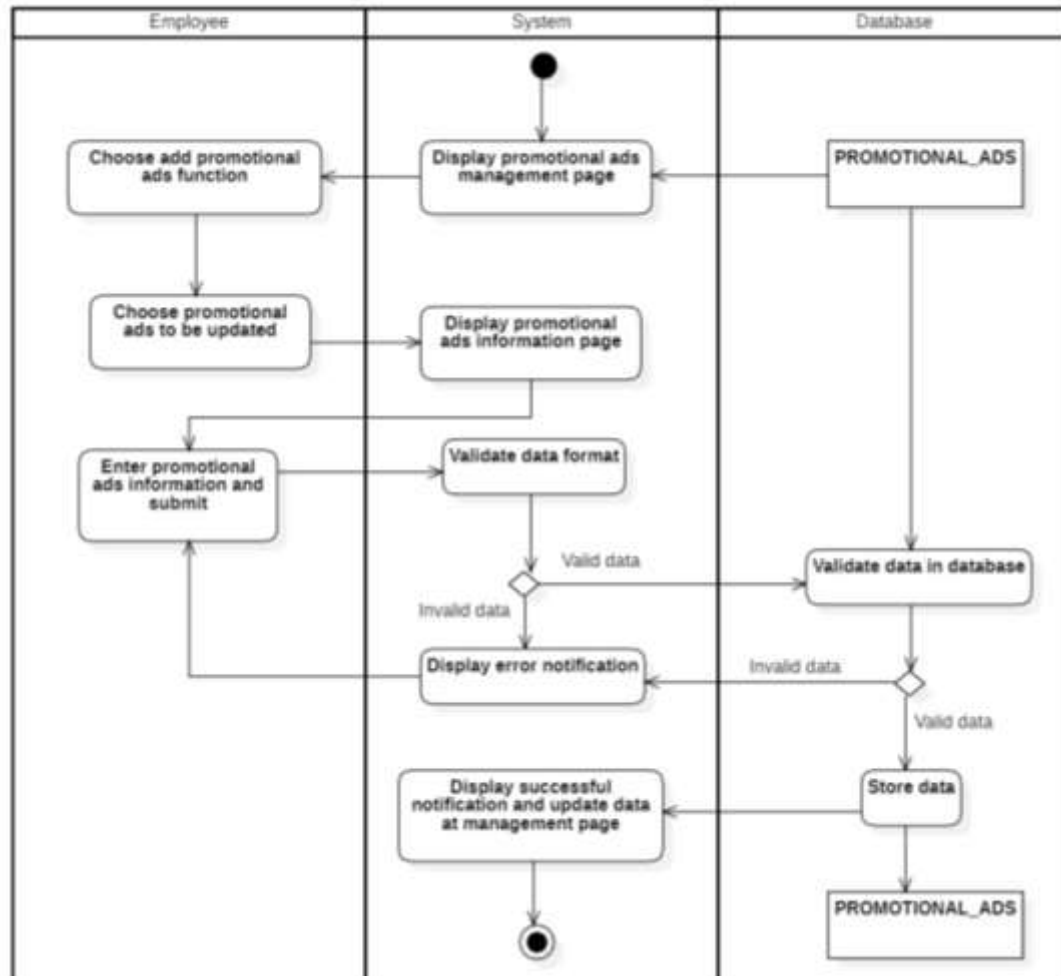
Use-case name	Add promotional ads (for administrator)
Description	Administrator can add new promotional ads to the database.
Trigger	Administrator clicks to button "Add promotional ads" at ingredient management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in.
Post-condition	Promotional ads information add to the database. Data at promotional ads management page is updated with new promotional.
Basic flow	<ol style="list-style-type: none"> 1. System display promotional ads management page. 2. Administrator chooses add promotional ads function. 3. System display promotional ads information page.

	<ol style="list-style-type: none"> 4. Administrator enter promotional ads information and submit. 5. System validate promotional ads information in correct format. 6. System validate promotional ads information with existing data in database. 7. System stores promotional ads information in database. 8. System displays successful notification and updates data at management page.
Alternative flow	<p>5a. System display invalid information due to wrong format. Use-case continues at step 4.</p> <p>6a. System display invalid information because data had been existed in the database. Use-case continues at step 4.</p>
Exception flow	



3.2.13.1 Update promotional ads

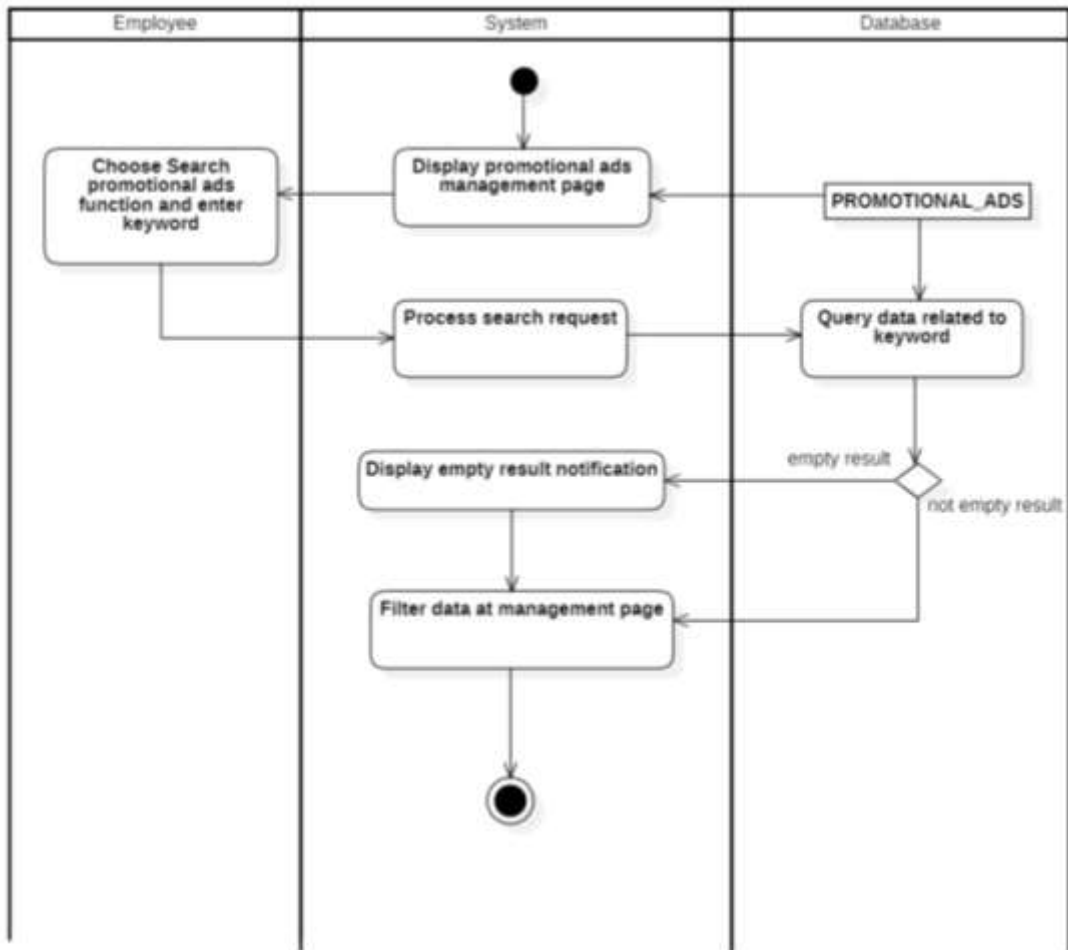
Use-case name	Update promotional ads(for administrator)
Description	Administrator can update to the database.
Trigger	Administrator clicks to button "Update promotional ads" at promotional ads management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in. Promotional ads information must be existed in database before updating.
Post-condition	Promotional ads information updated to the database.
Basic flow	<ol style="list-style-type: none">1. System display promotional ads management page.2. Administrator chooses update promotional ads function.3. Administrator chooses promotional ads to be updated.4. System display promotional ads information page.5. Administrator enter promotional ads information and submit.6. System validate promotional ads information in correct format.7. System validate promotional ads information with existing data in database.8. System stores promotional ads information in database.9. System displays successful notification and updates data at management page.
Alternative flow	<p>6a. System display invalid information due to wrong format. Use-case continues at step 5.</p> <p>7a. System display invalid information because data had been existed in the database. Use-case continues at step 5.</p>
Exception flow	



3.2.13.3 Search promotional ads

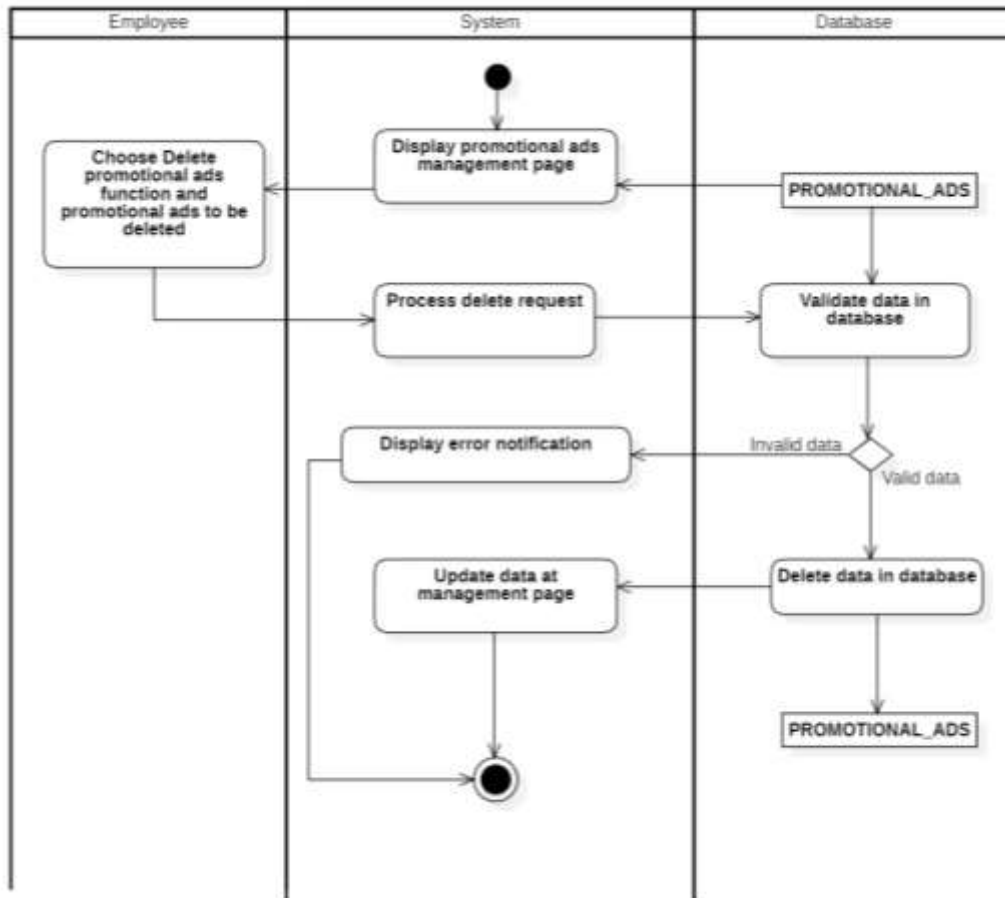
Use-case name	Search promotional ads(for administrator)
Description	Administrator can search promotional ads from the database.
Trigger	Administrator clicks to button "Search promotional ads" at restaurant management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in. Promotional ads information must be existed in database.
Post-condition	Data at promotional ads management page is filtered by keywords.
Basic flow	<ol style="list-style-type: none"> 1. System display promotional ads management page. 2. Administrator chooses search promotional ads function and enters keywords. 3. System processes data search request. 4. System queries data related to keywords.

	5. System updates data at management page.
Alternative flow	4a. System display empty result notification. Use-case continues at step 5.
Exception flow	



3.2.13.4 Delete promotional ads

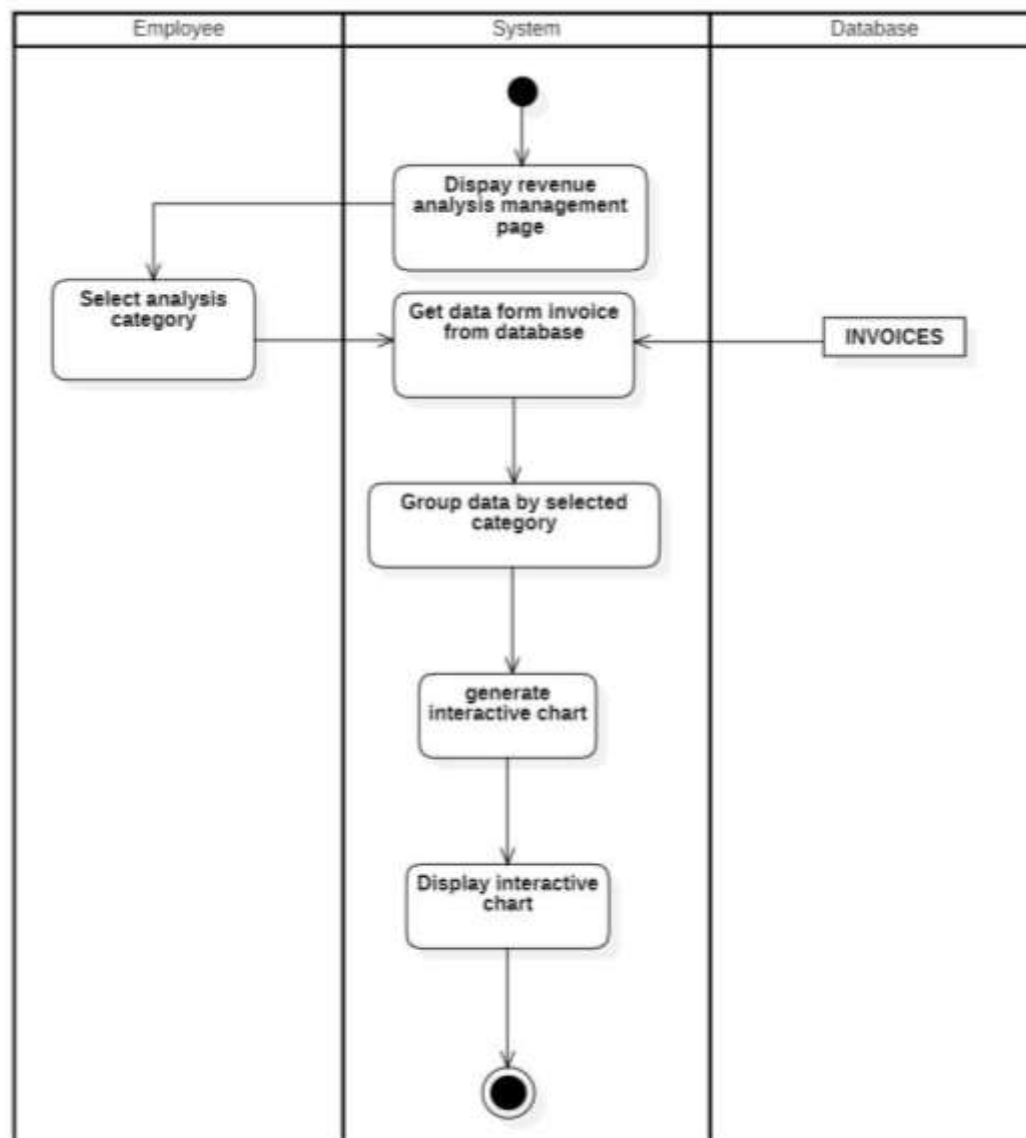
Use-case name	Delete promotional ads (for administrator)
Description	Administrator can delete promotional ads information in the database.
Trigger	Administrator clicks to button "Delete promotional ads" at promotional management page.
Pre-condition	Administrator's device must connect to Internet. Administrator's account is signed-in. Promotional ads information must be existed in database. No external data involved.
Post-condition	Data at promotional ads management page is updated.
Basic flow	<ol style="list-style-type: none">1. System display promotional ads management page.2. Administrator chooses delete promotional ads function and promotional ads to be deleted.3. System processes data delete request.4. System validates promotional ads information that need to be deleted.5. System delete promotional ads information in the database.6. System updates data at management page.
Alternative flow	5a. System display error notification because no data existed or external data had required that promotional ads information. Use-case stops.
Exception flow	



3.2.14 Revenue analysis

Use-case name	Revenue analysis (for administrator)
Description	Administrator can revenue analysis by various categories
Trigger	Administrators clicks to button "revenue analysis" at management page
Pre-condition	Administrators' device must connect to Internet. Administrator's status is signed-in.
Post-condition	
Basic flow	<ol style="list-style-type: none"> 1. Administrator click to button "revenue analysis". 2. System display revenue analysis management page. 3. Administrator select analysis category (restaurant system, ticket, month) 4. System get list of invoices from database. 5. Group data by selected category. 6. generate interactive chart.

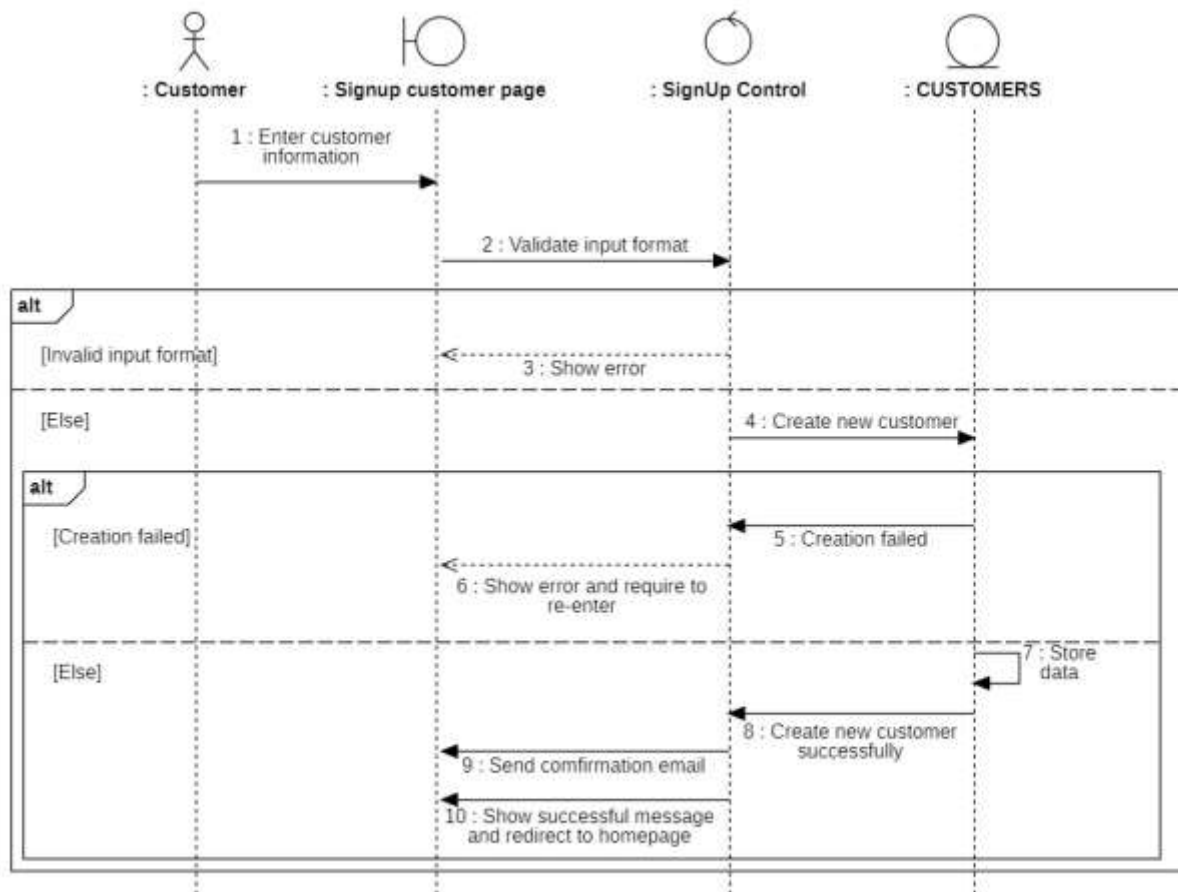
	7. Display interactive charts.
Alternative flow	
Exception flow	



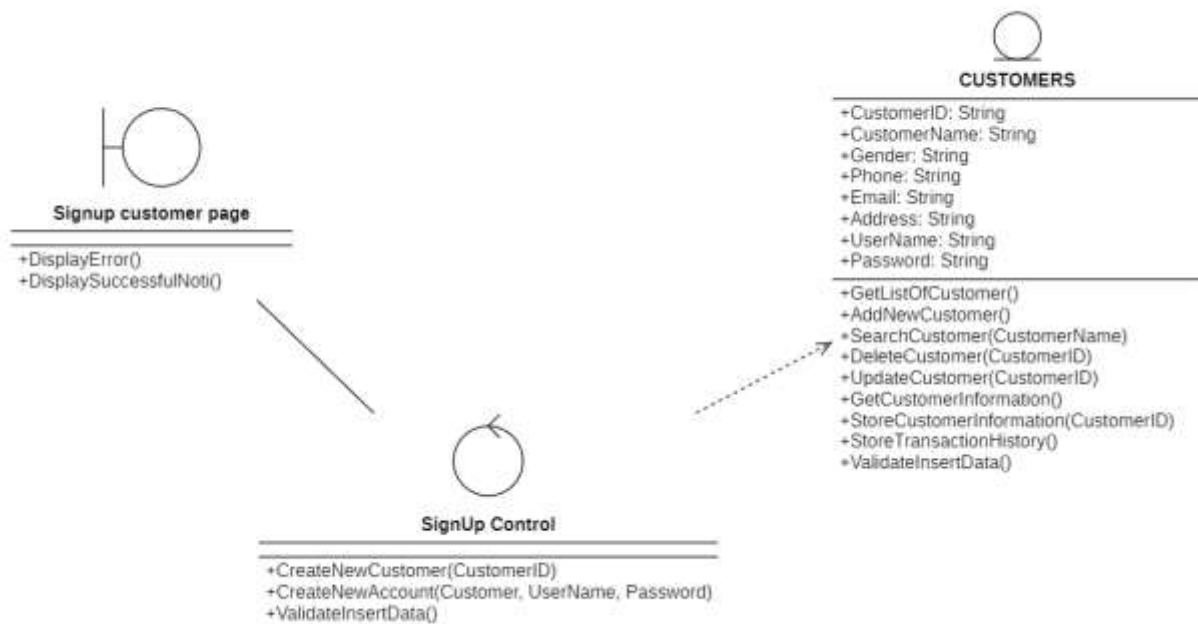
3.3 Sequence diagram and class diagram

3.3.1 Sign up

3.3.1.1 Sequence diagram

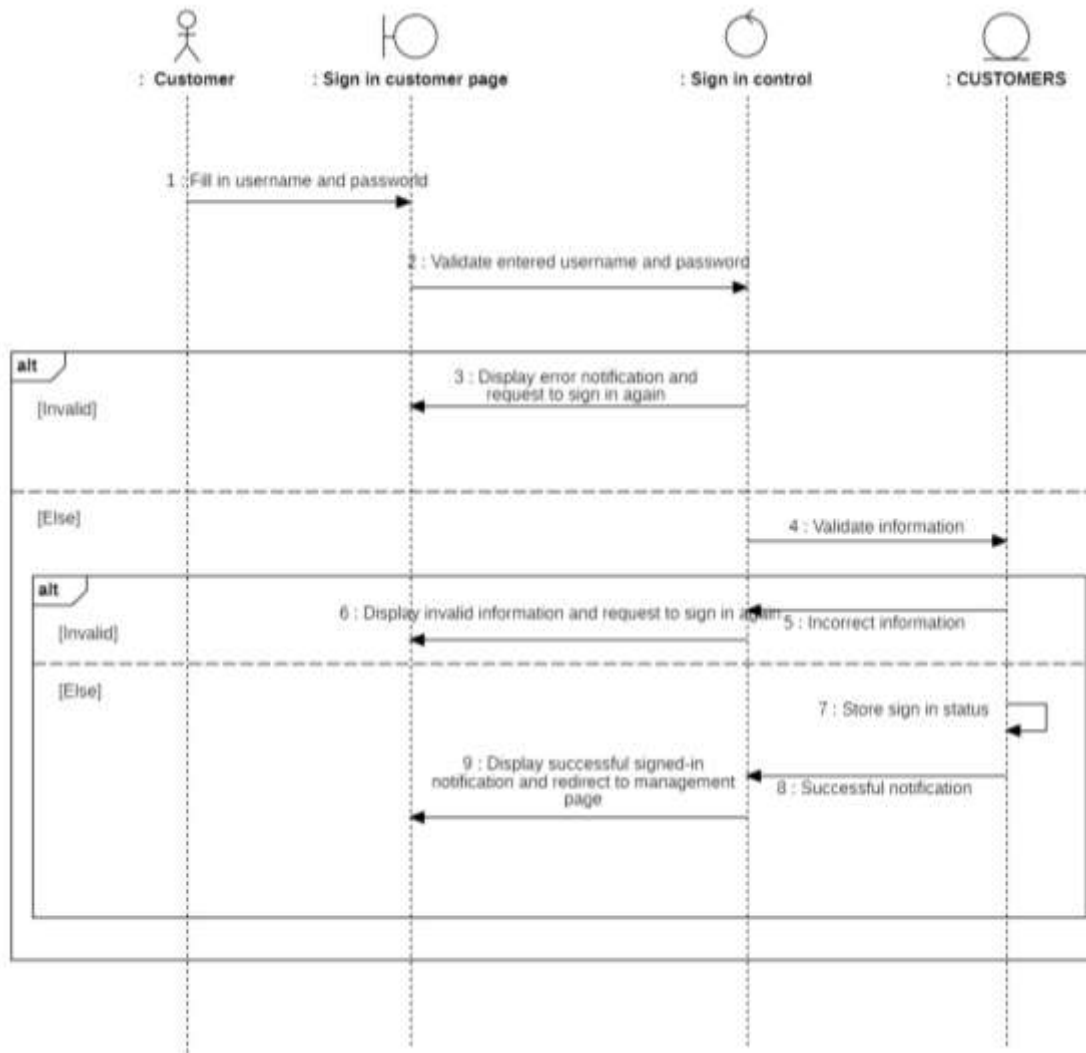


3.3.1.2 Class diagram

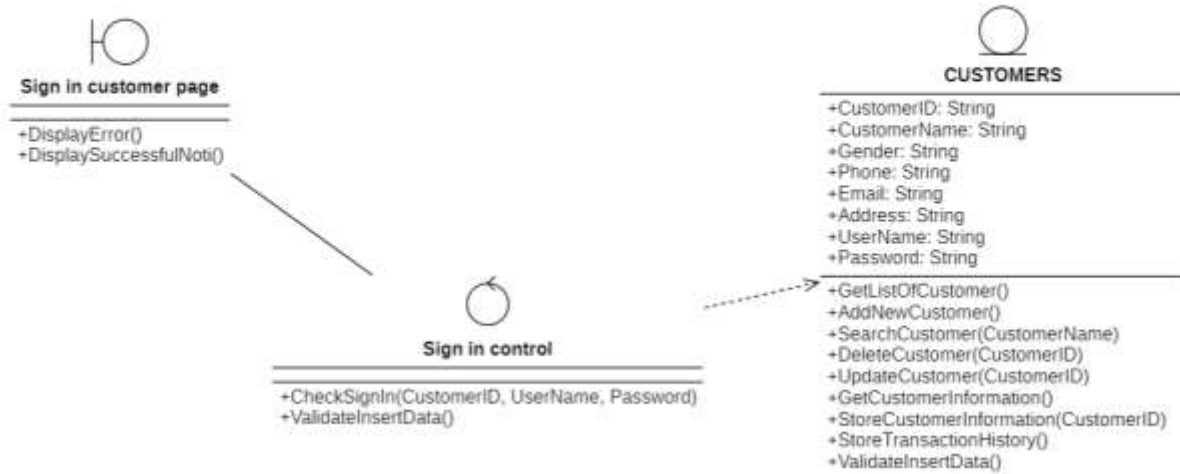


3.3.2 Sign in (for customers)

3.3.2.1 Sequence diagram

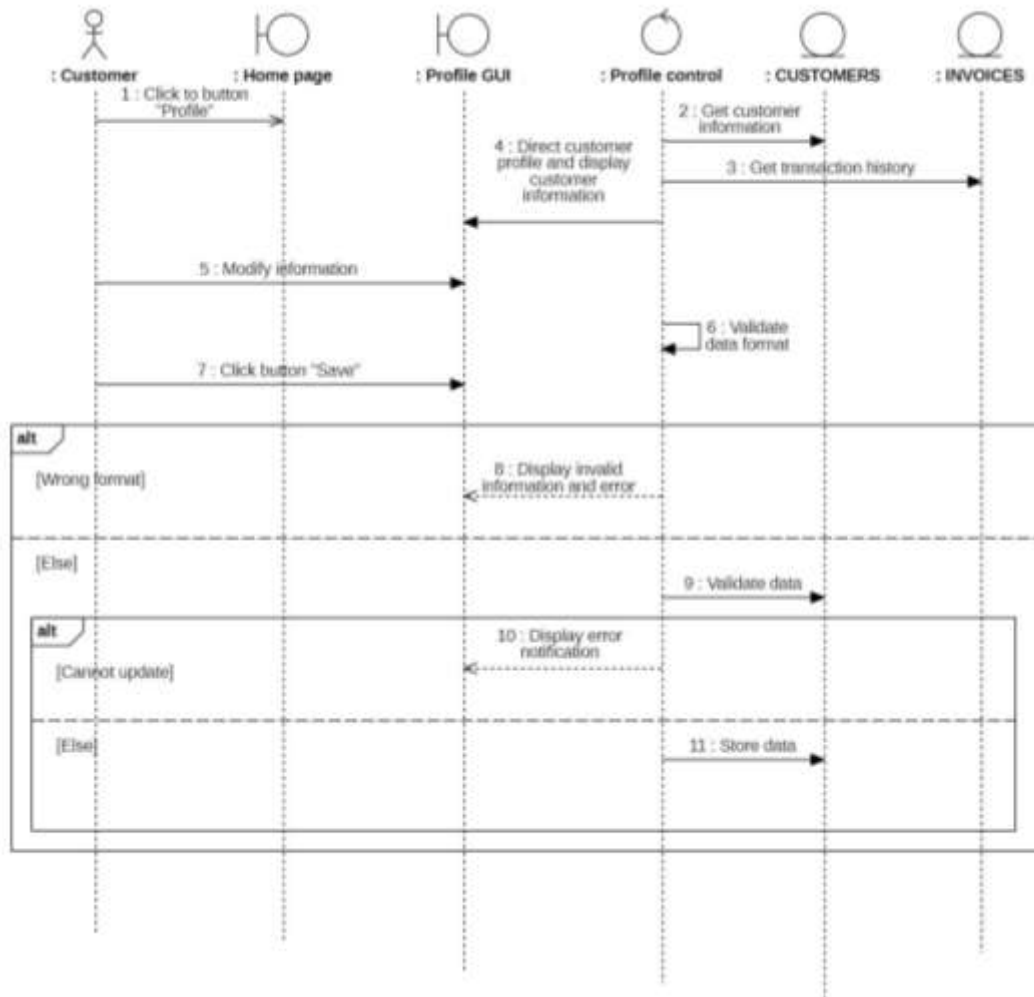


3.3.2.2 Class diagram

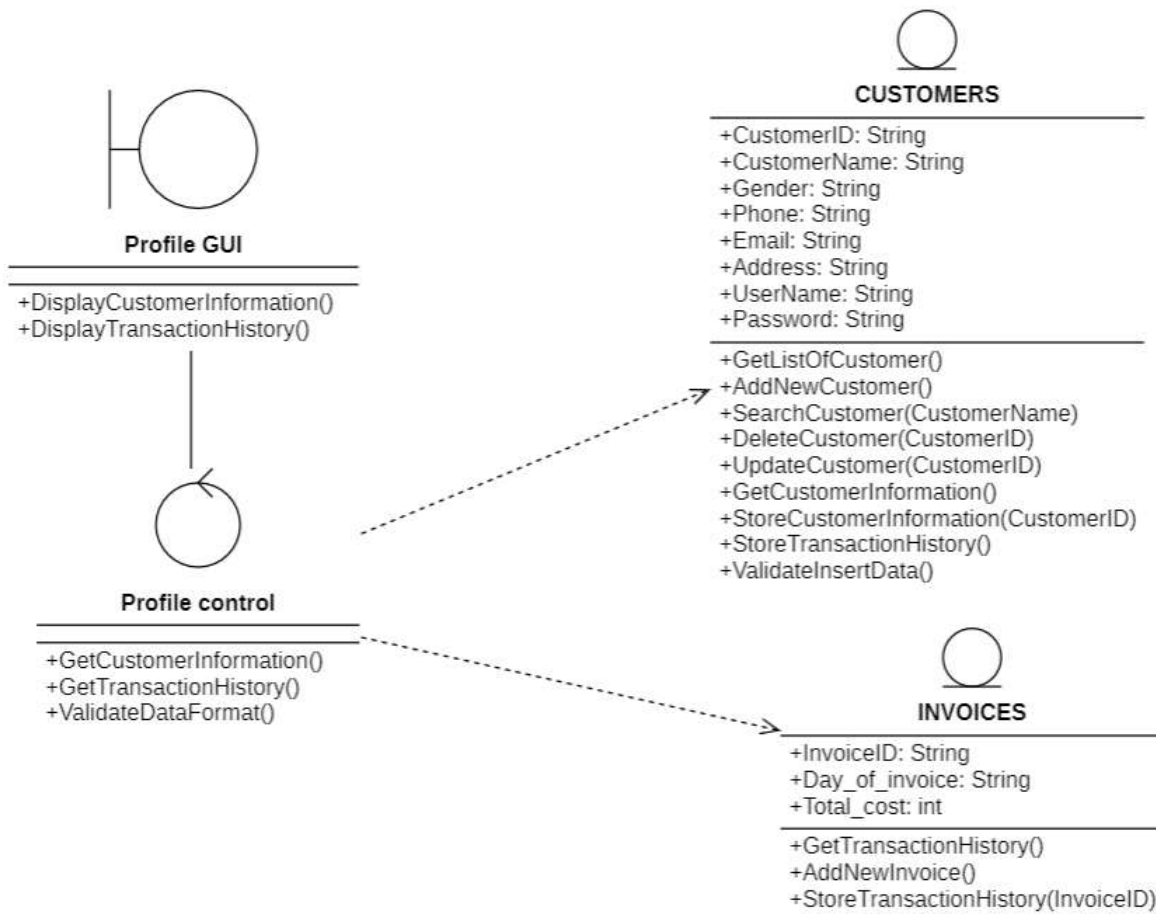


3.3.3 Manage profile

3.3.3.1 Sequence diagram

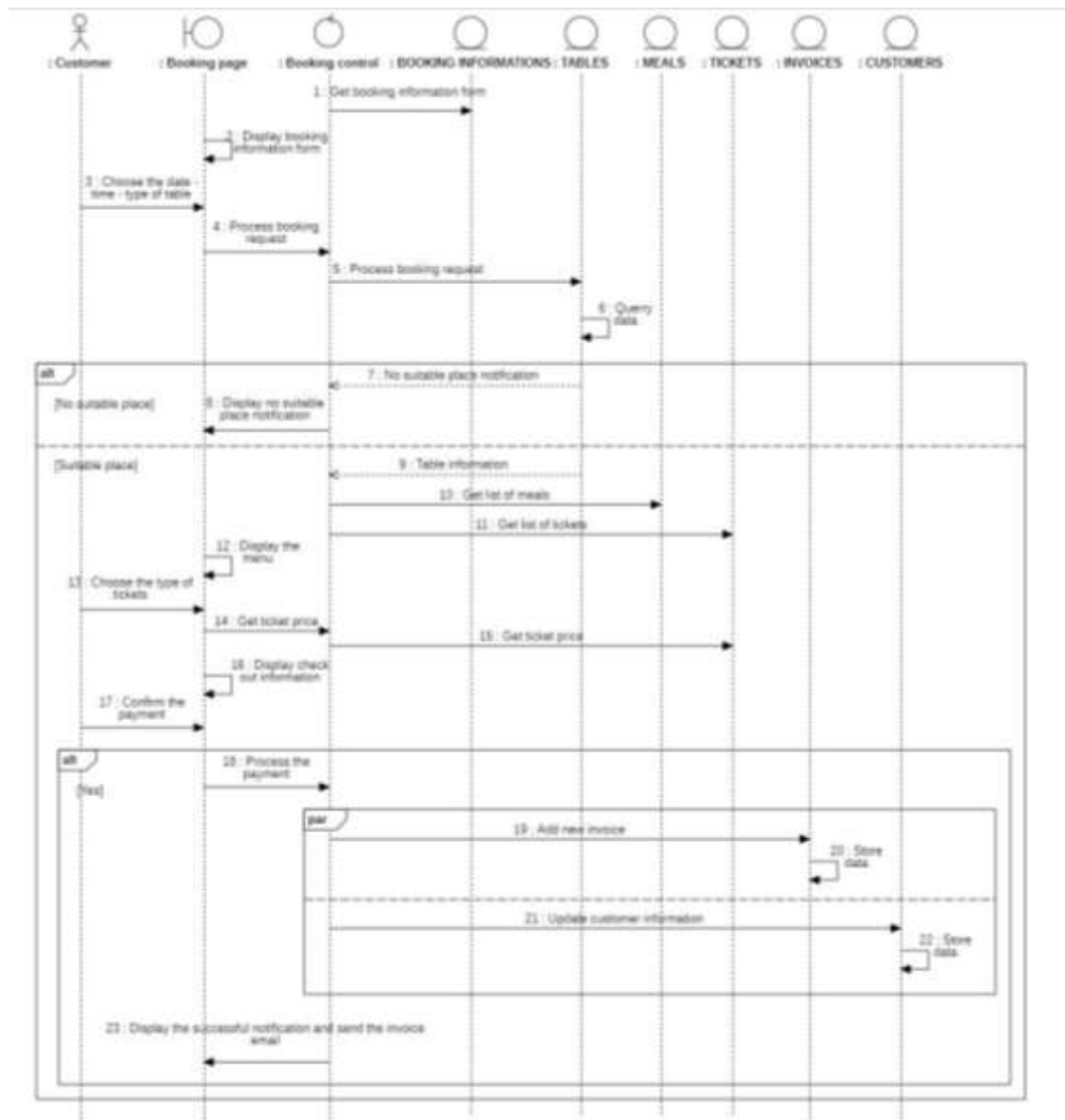


3.3.3.2 Class diagram

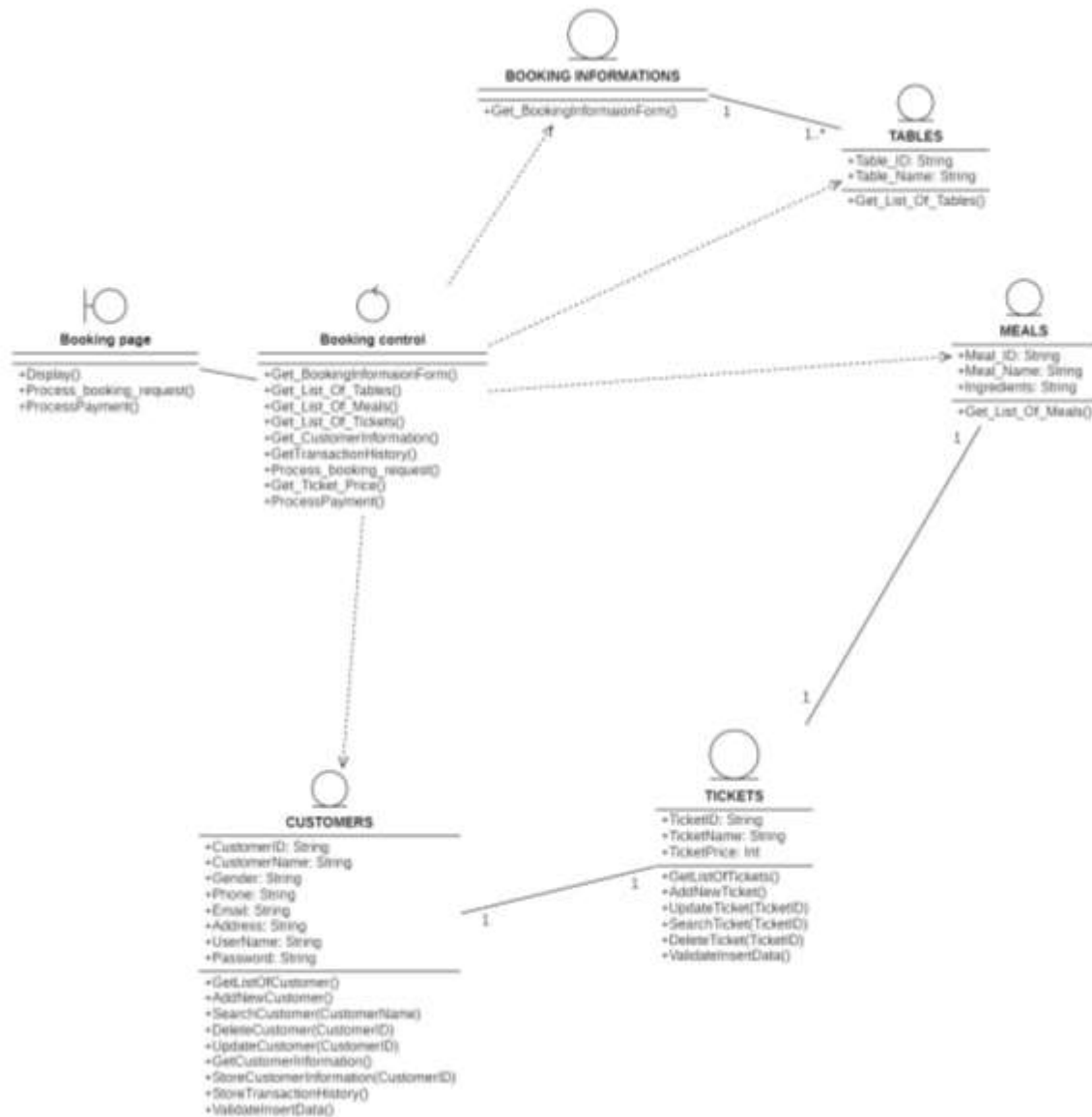


3.3.4 Book tickets

3.3.4.1 Sequence diagram

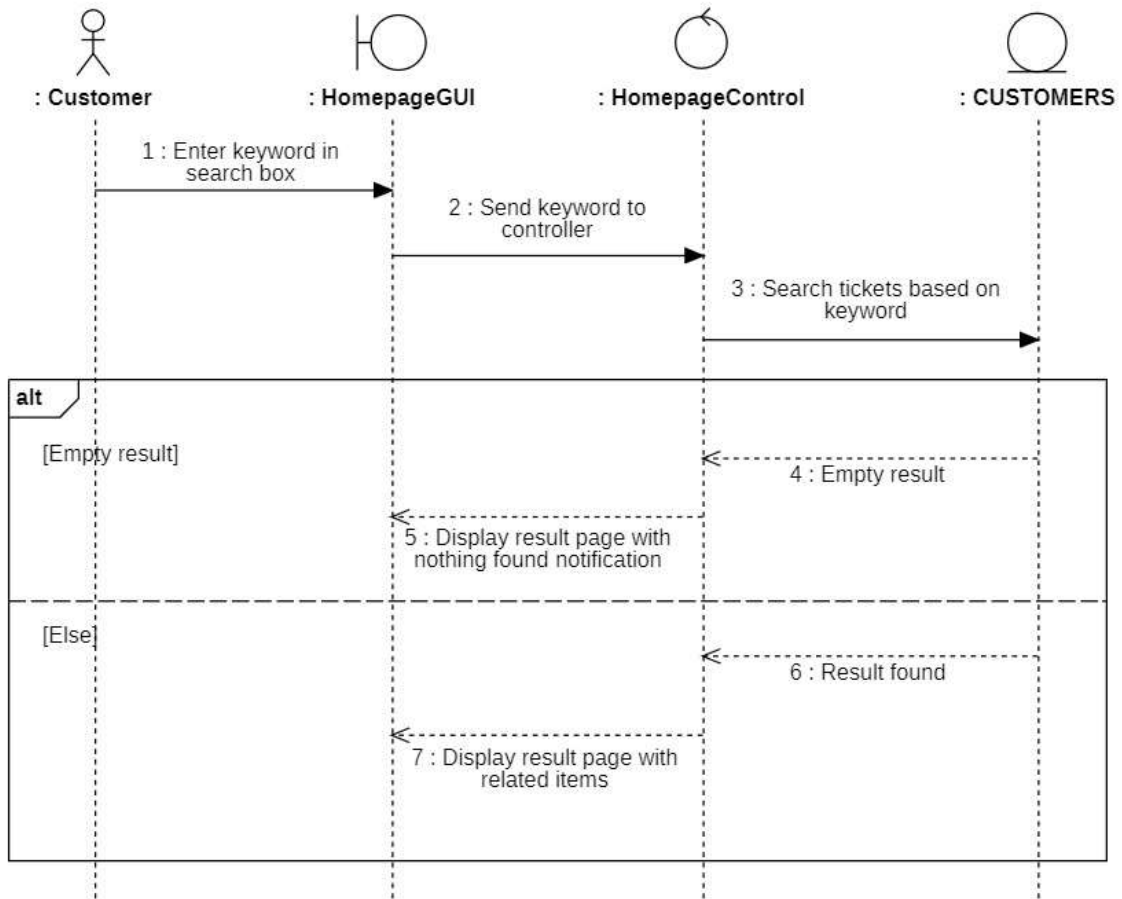


3.3.4.2 Class diagram

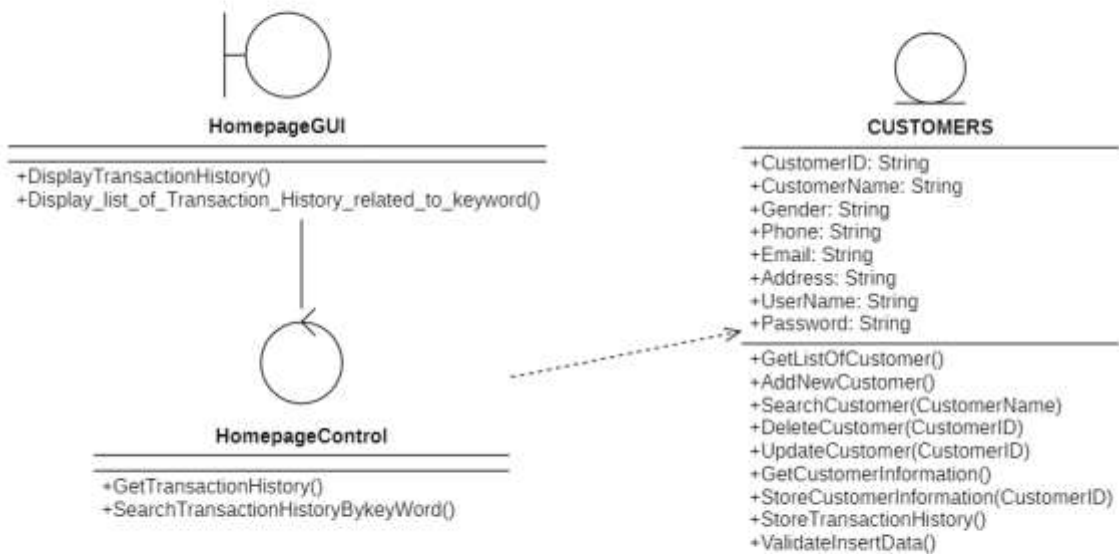


3.3.5 Search

3.3.5.1 Sequence diagram

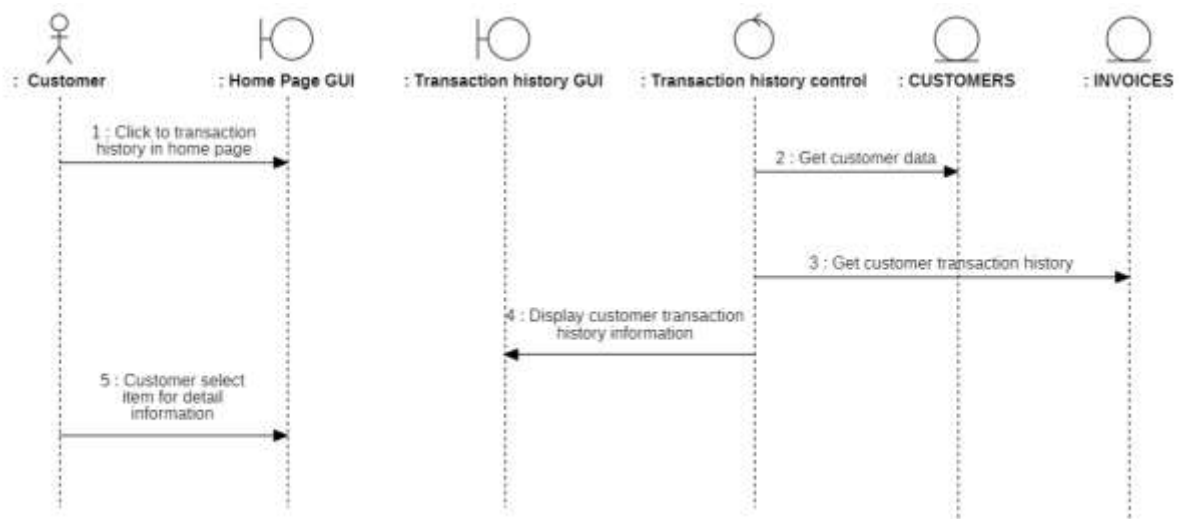


3.3.5.2 Class diagram

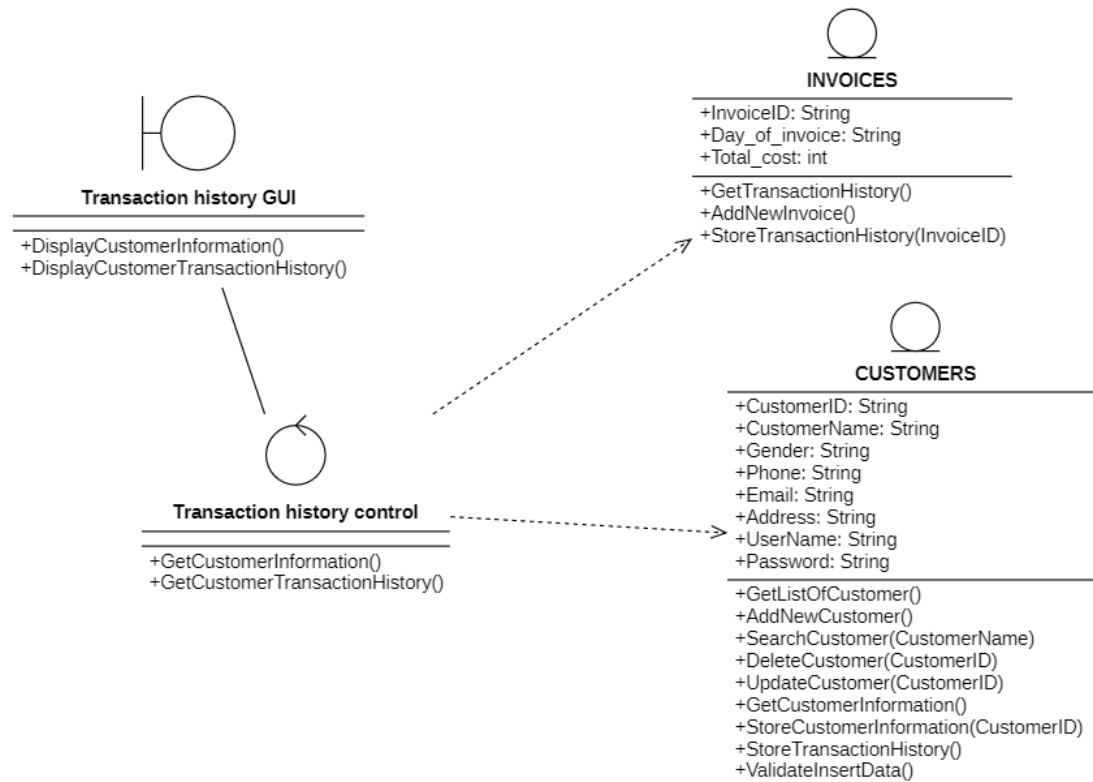


3.3.6 Review transaction history

3.3.6.1 Sequence diagram

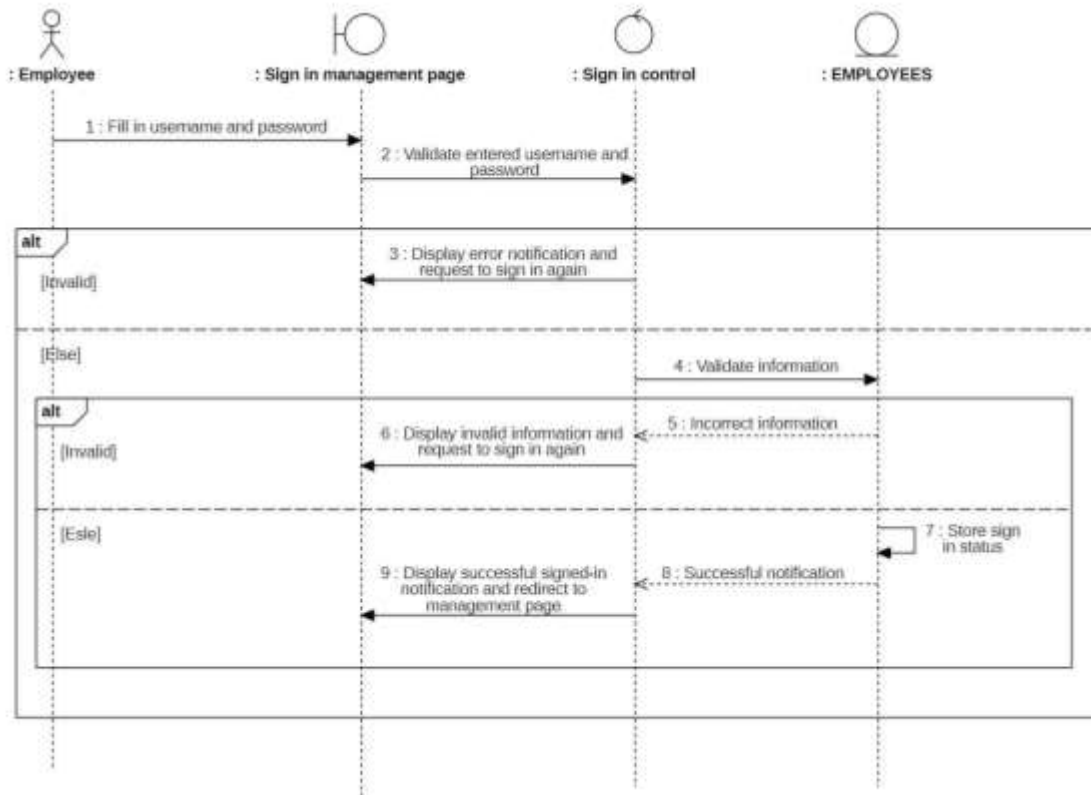


3.3.6.2 Class diagram

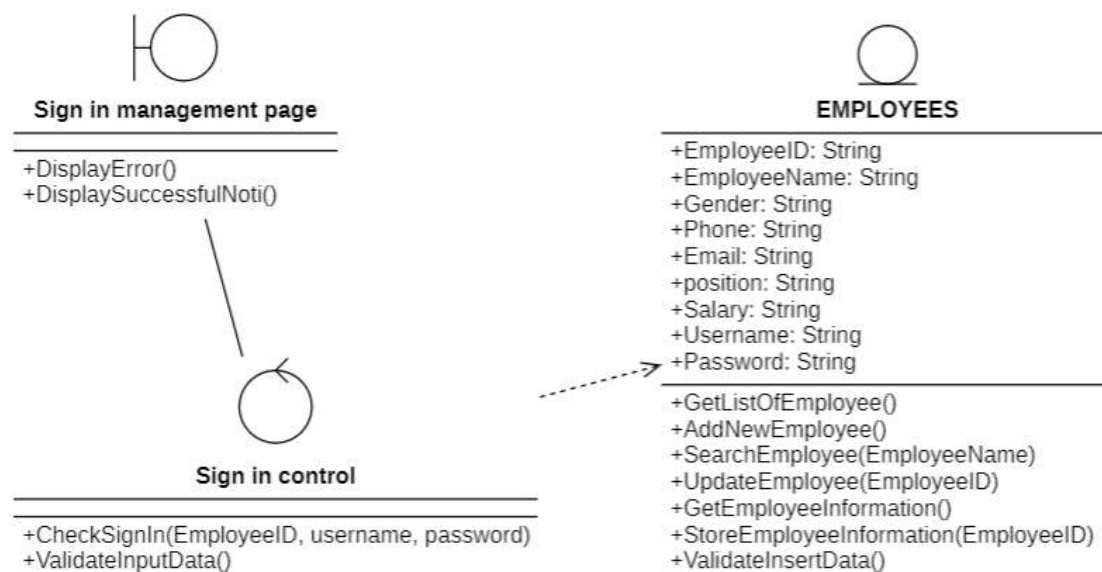


3.3.7 Sign in (for administrator)

3.3.7.1 Sequence diagram

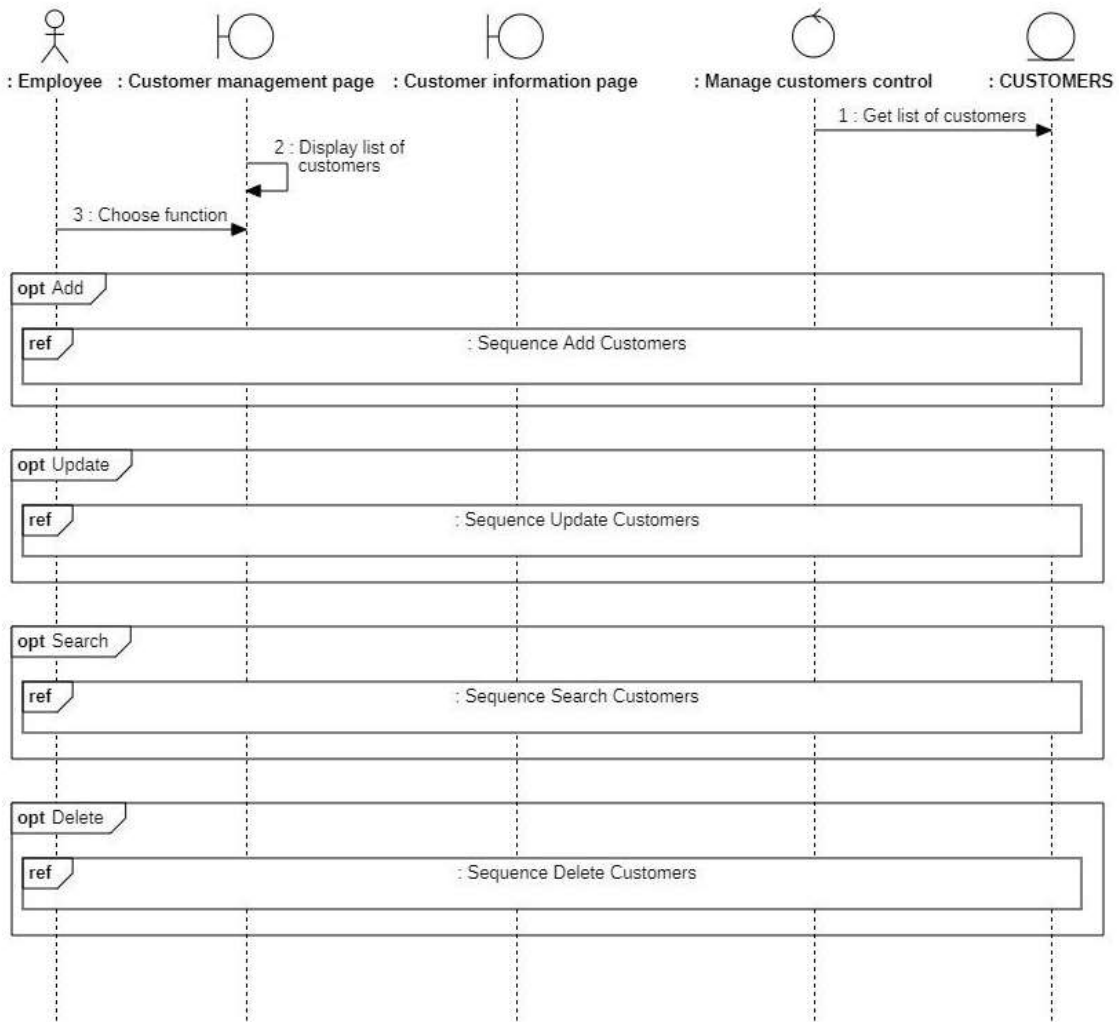


3.3.7.2 Class diagram

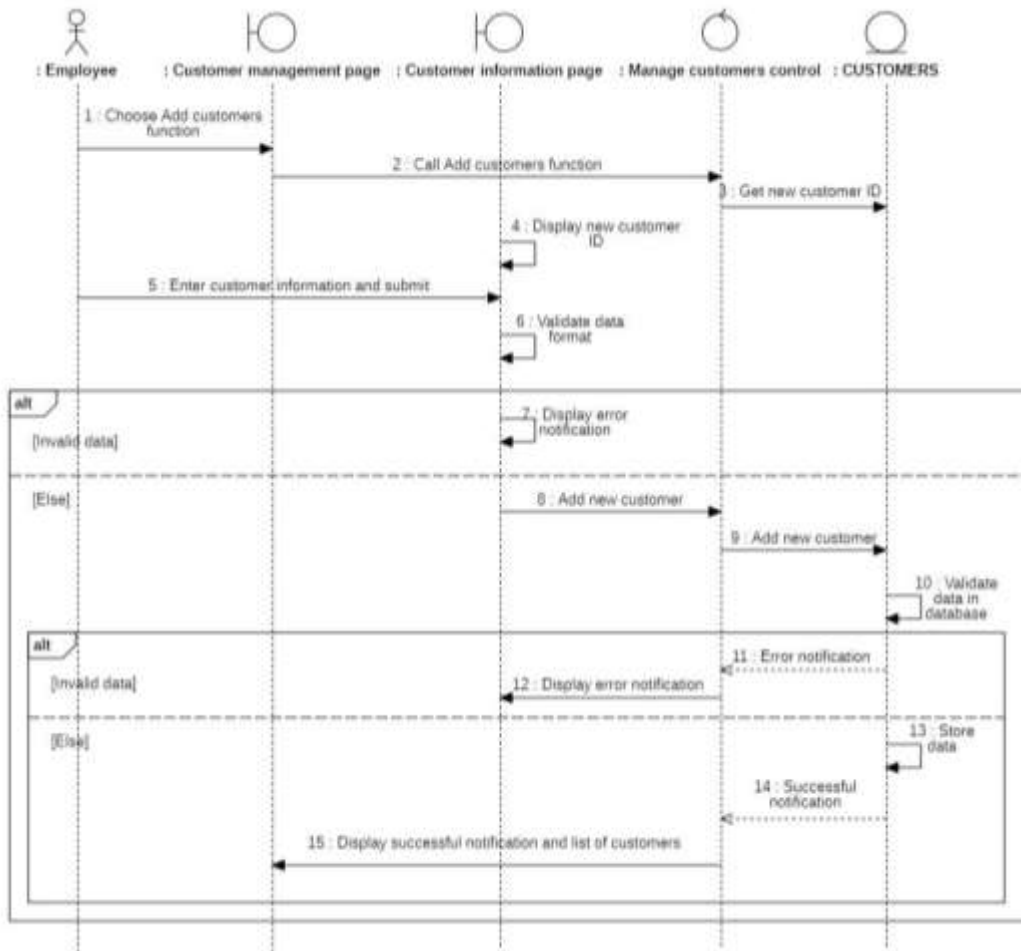


3.3.8. Manage customers

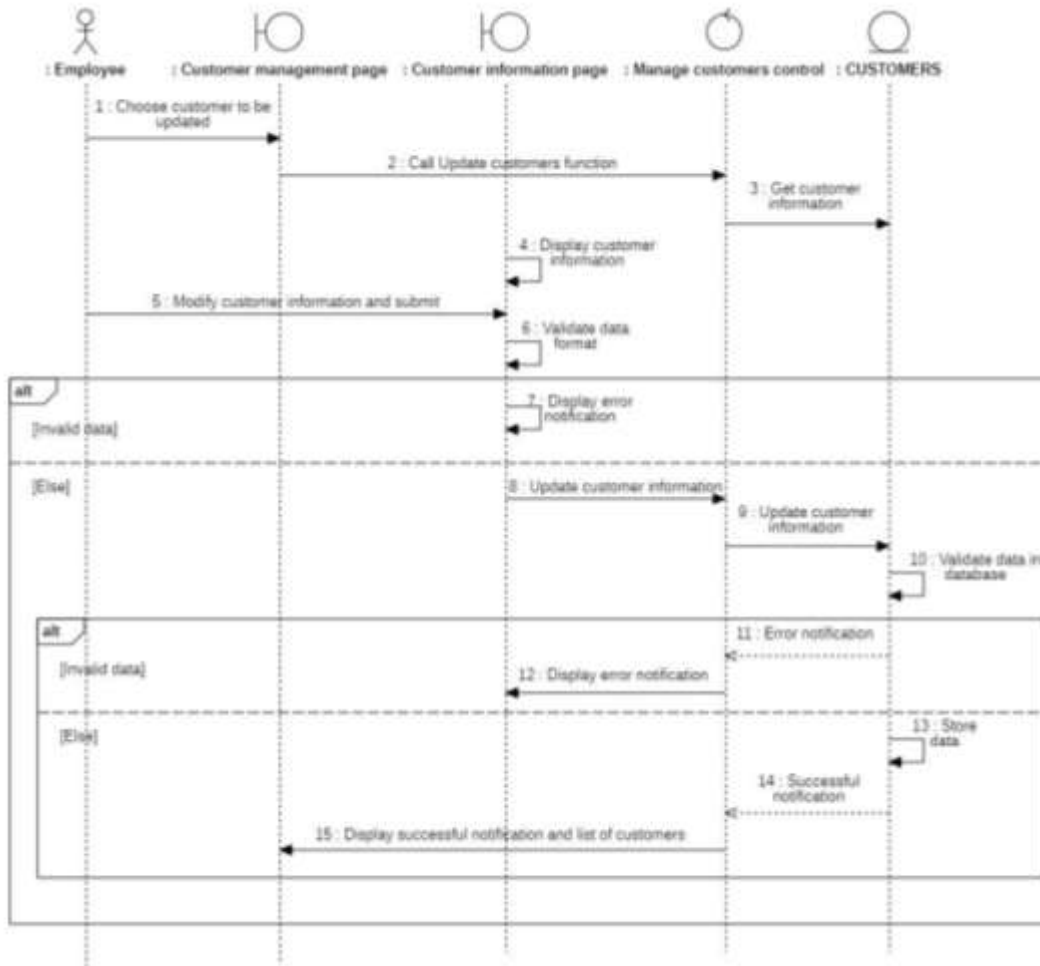
3.3.8.1 Sequence diagram



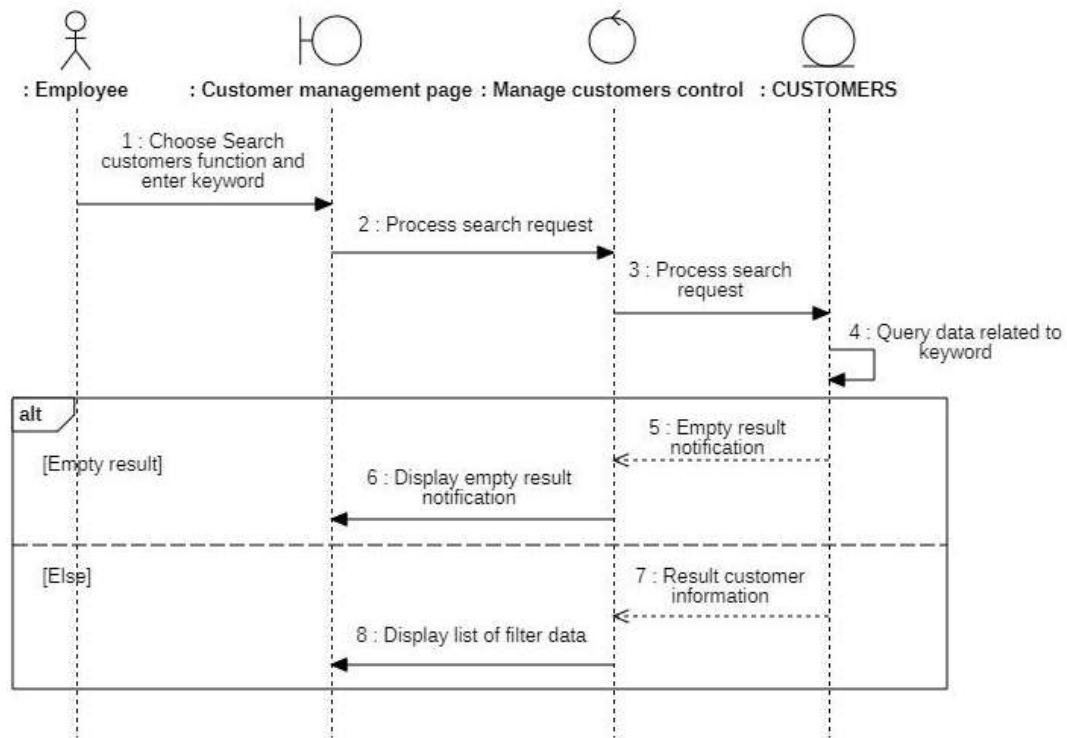
-Add customer



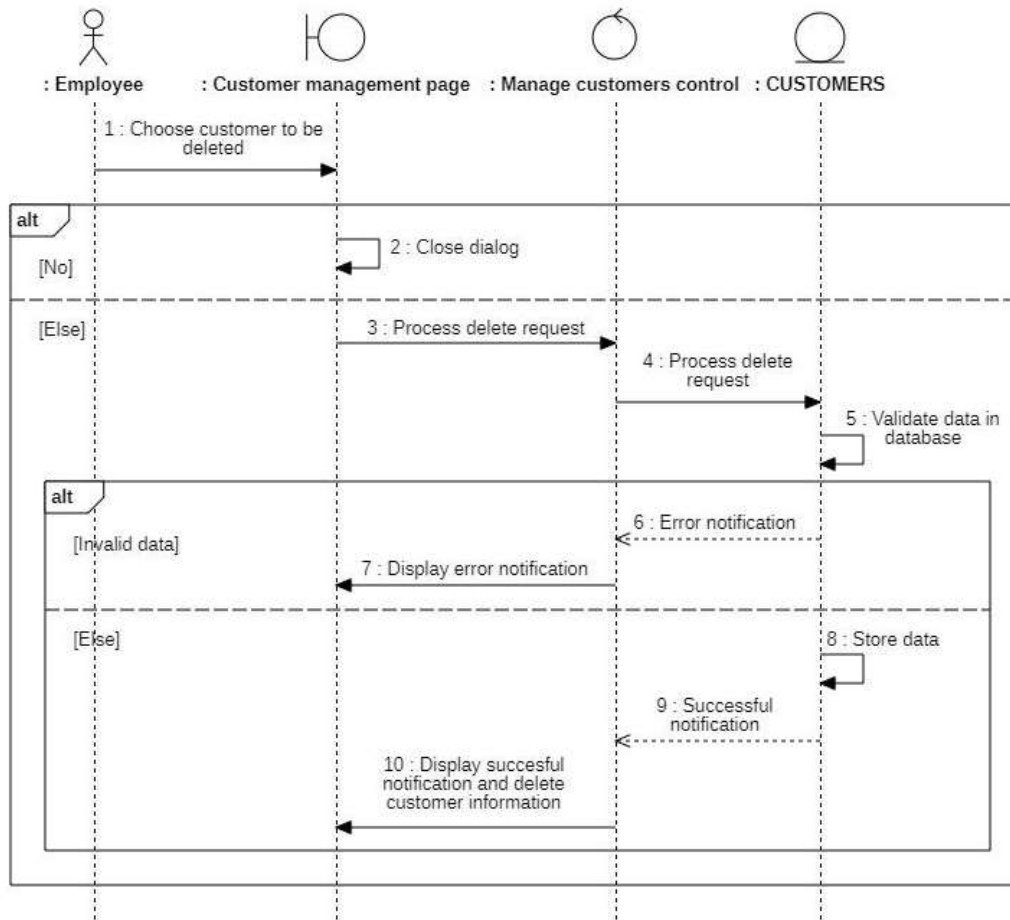
- Update customer



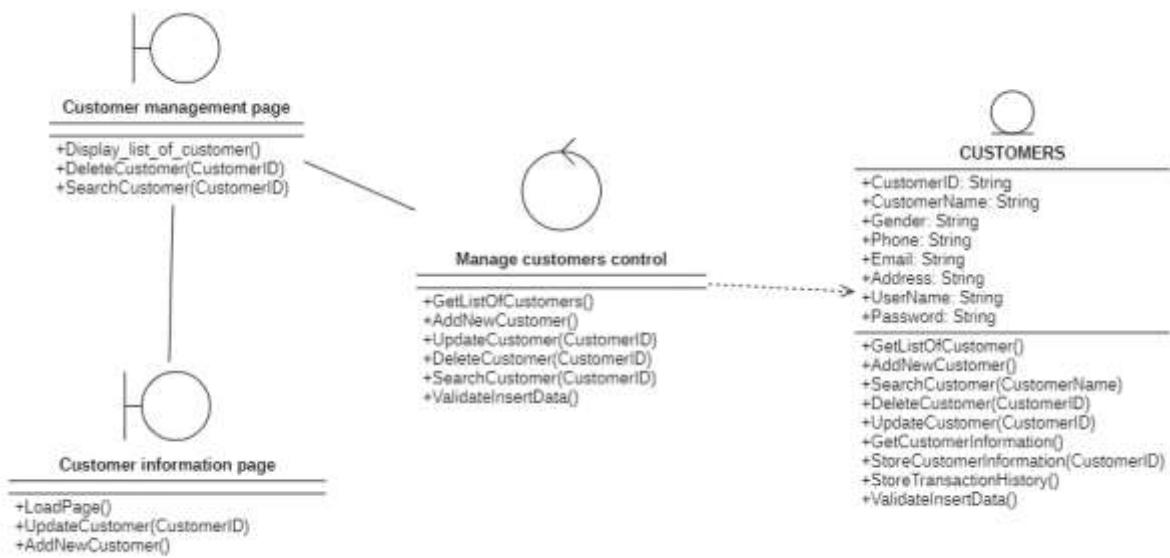
- Search customer



- Delete customer

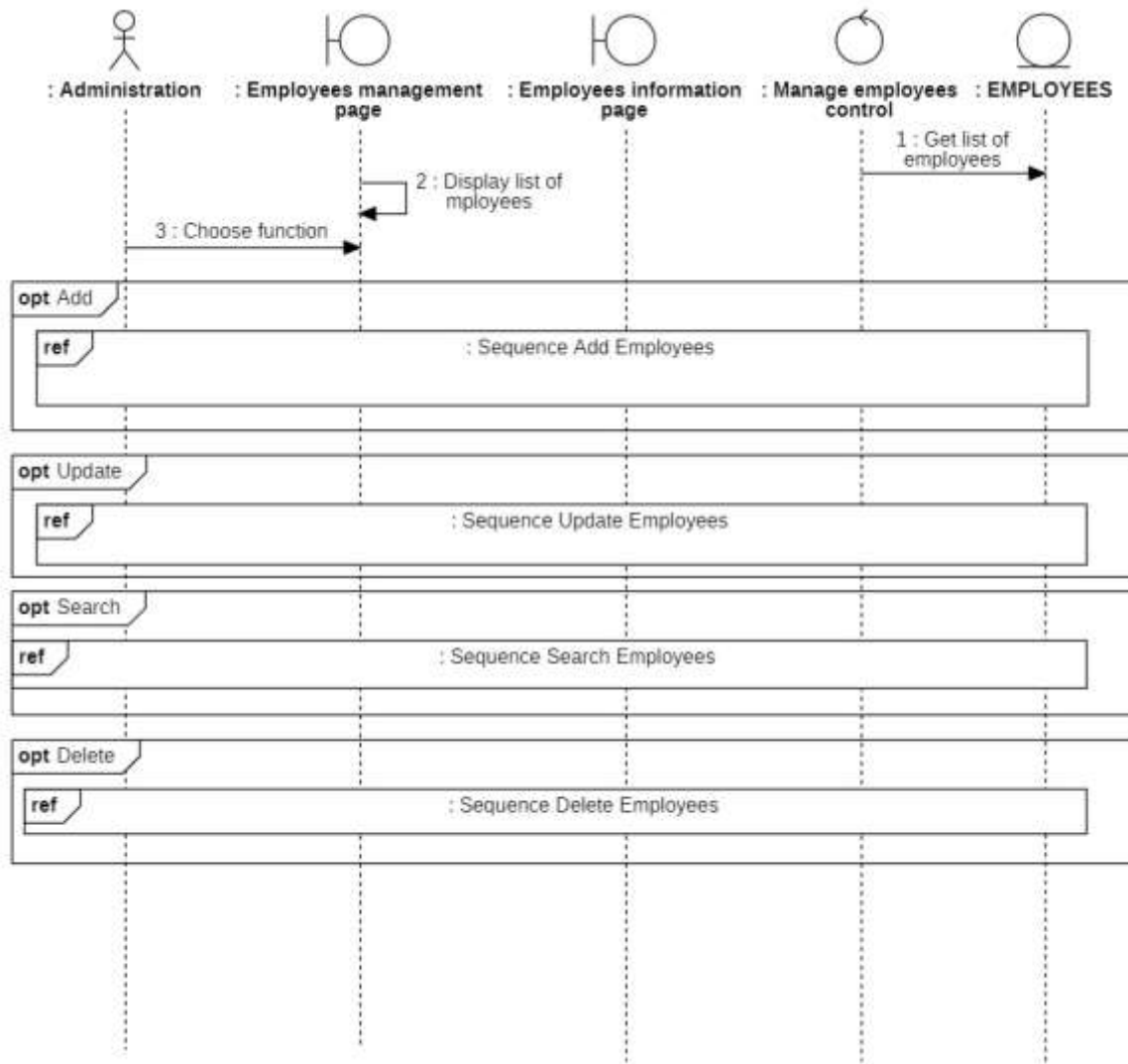


3.3.8.2 Class diagram

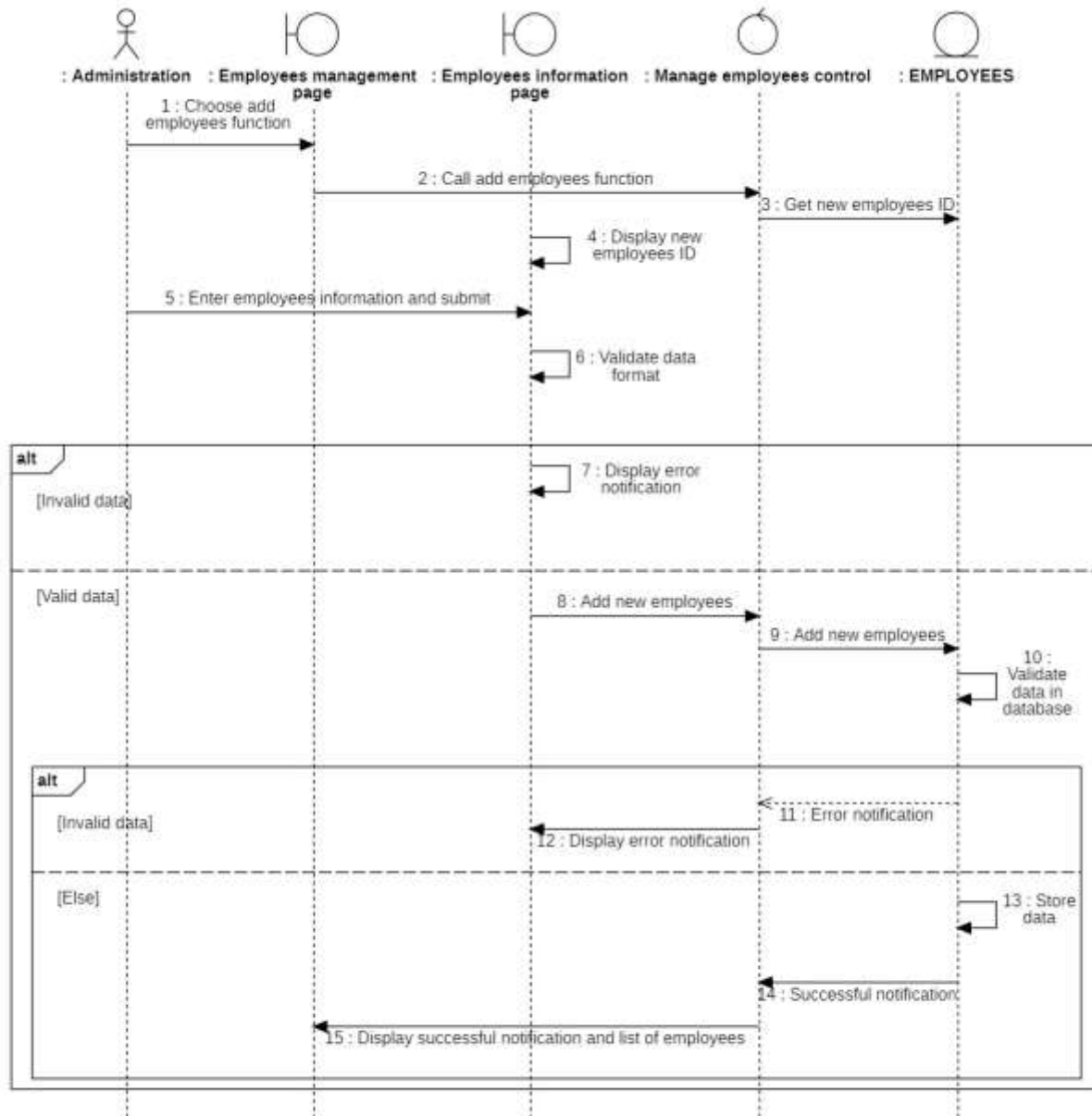


3.3.9 Manage employees

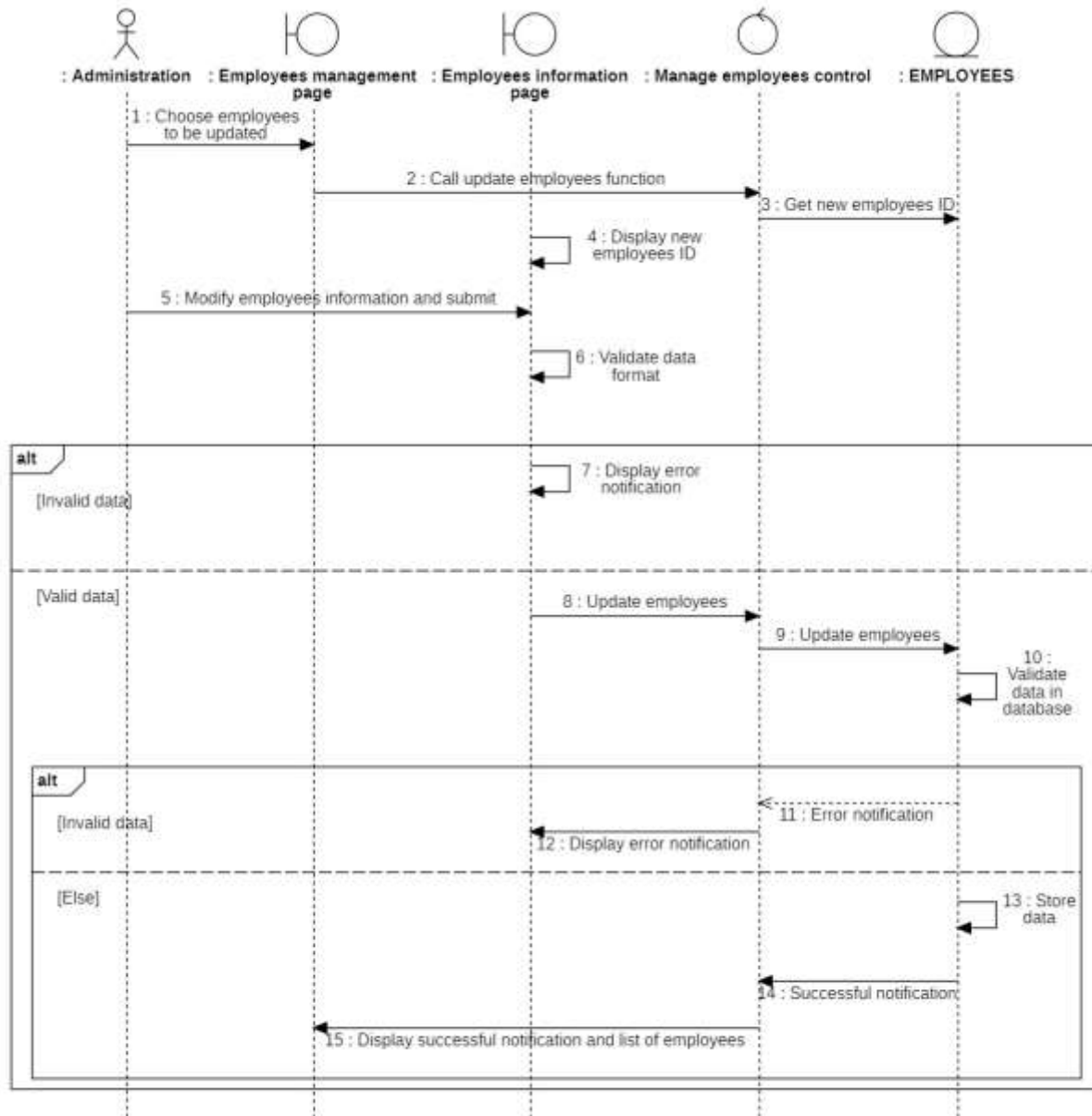
3.3.9.1 Sequence diagram



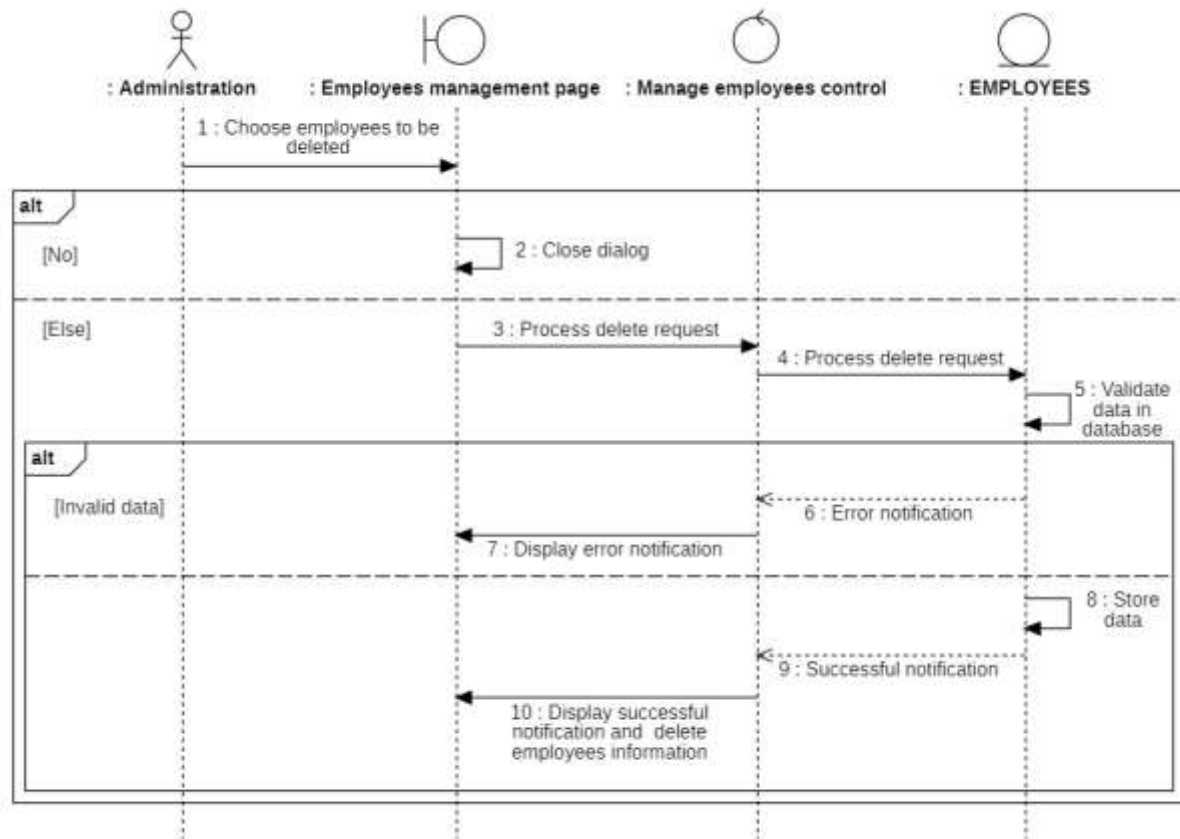
-Add employee



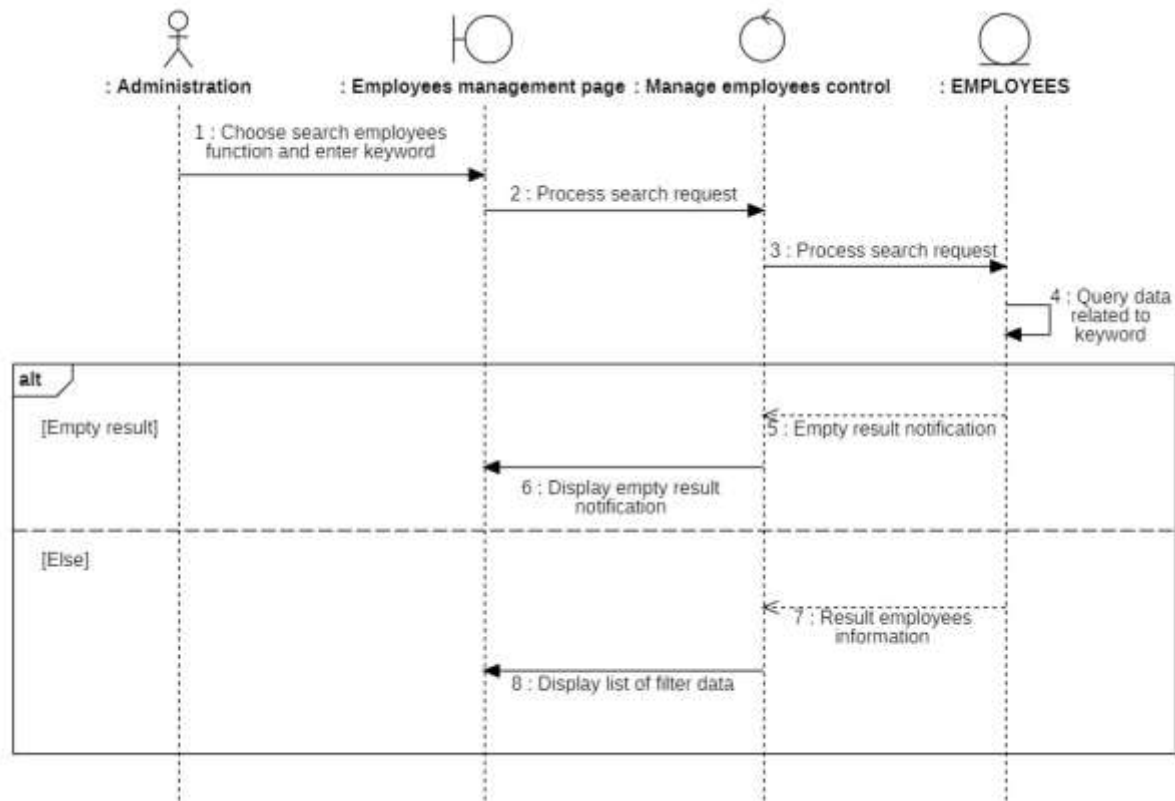
-Update employee



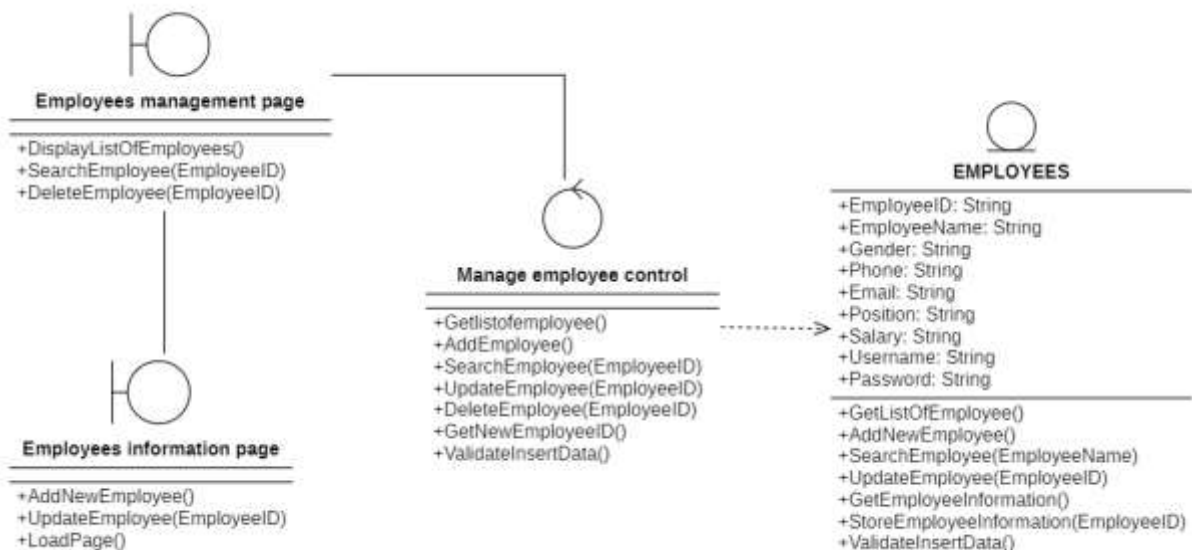
-Search employee



-Delete employee

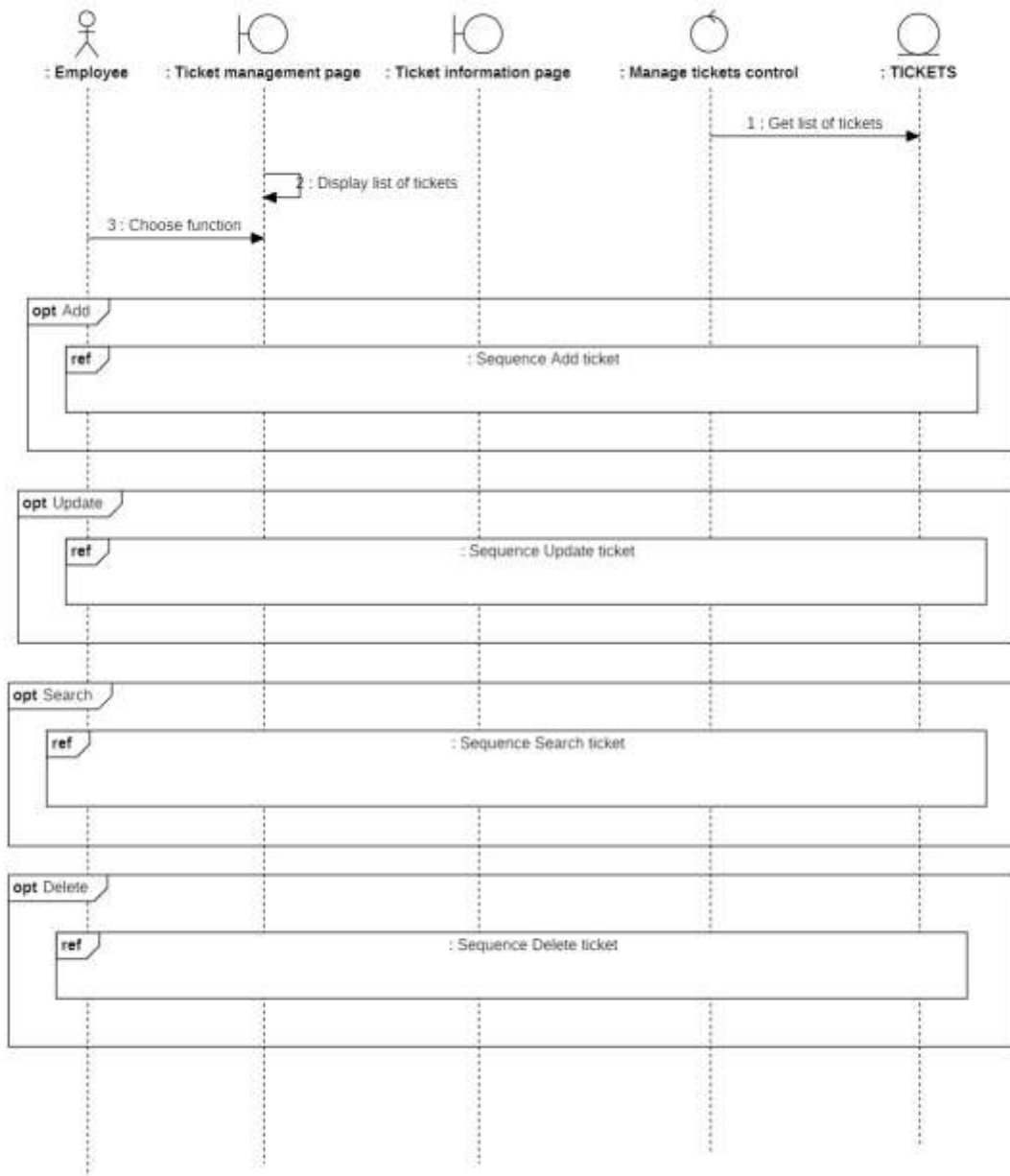


3.3.9.2 Class diagram

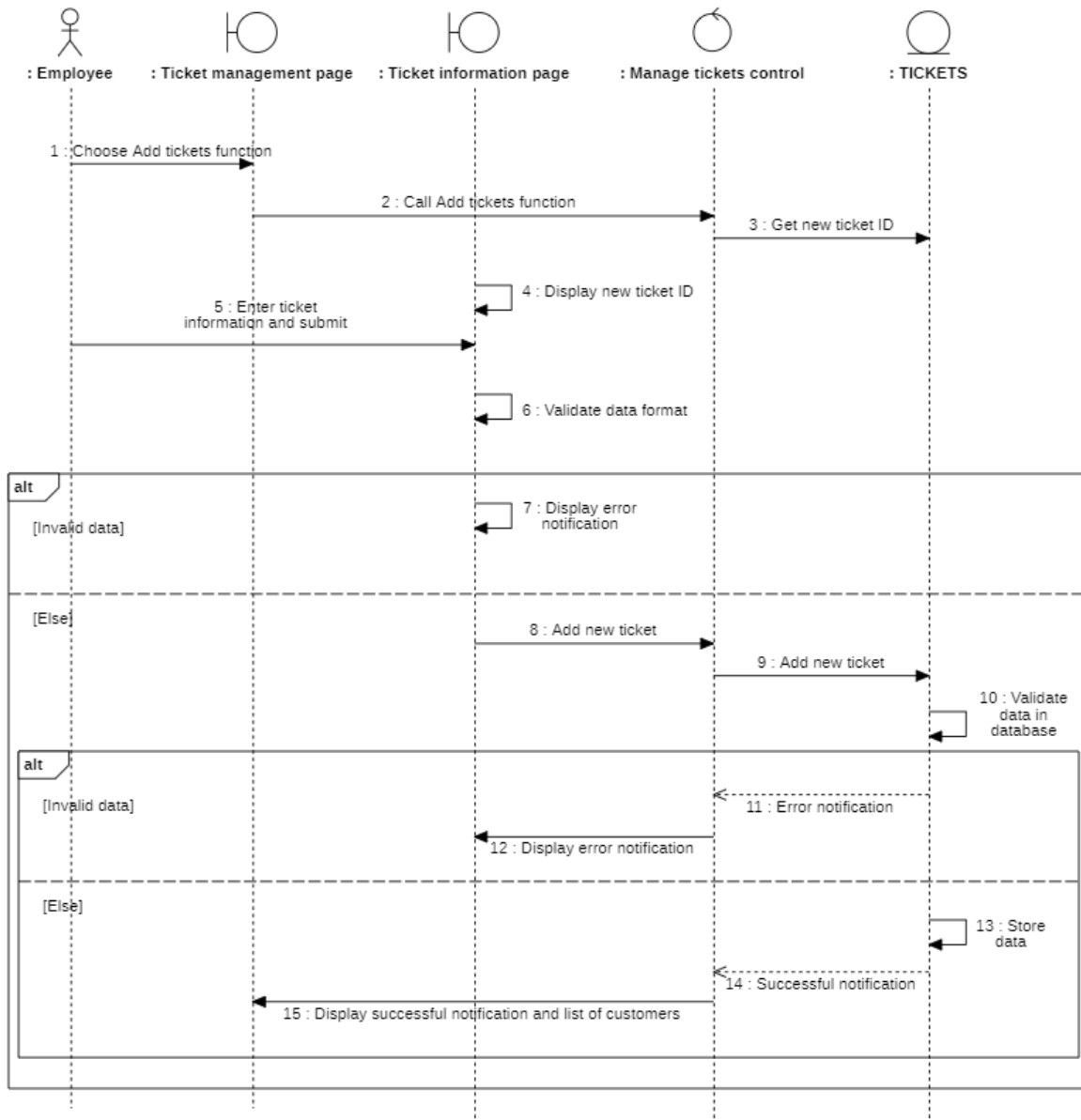


3.3.10 Manage tickets

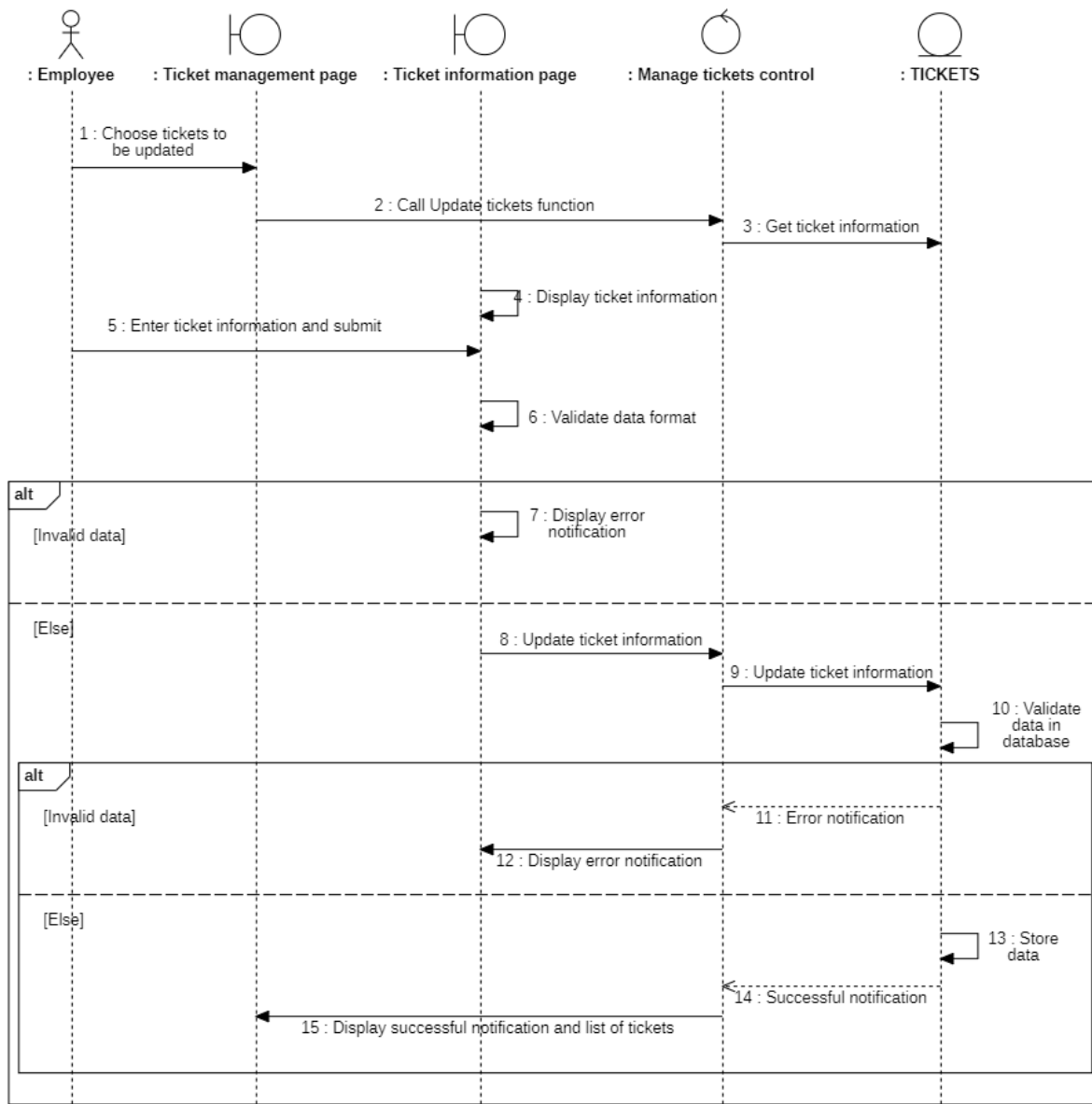
3.3.10.1 Sequence diagram



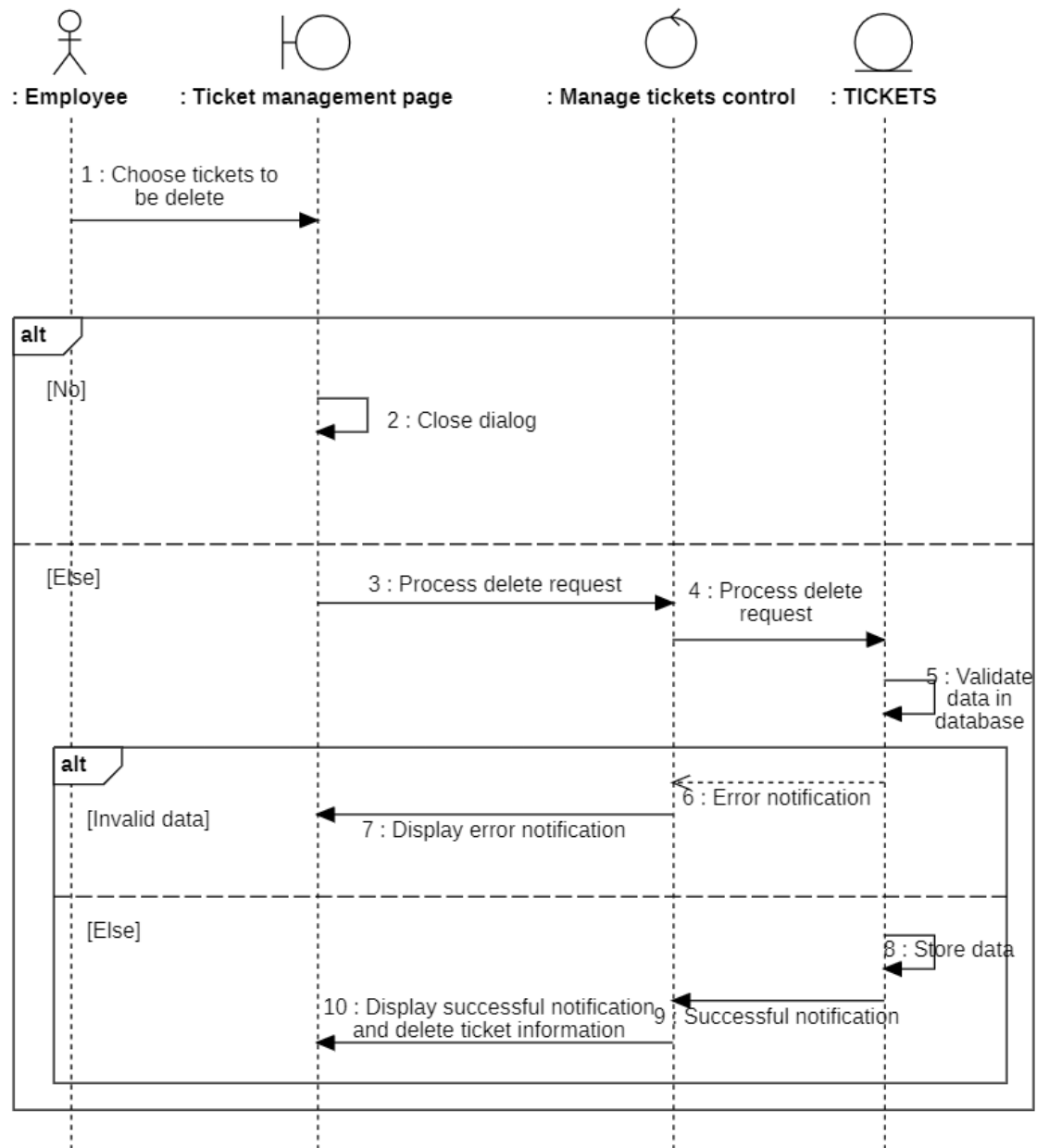
-Add ticket



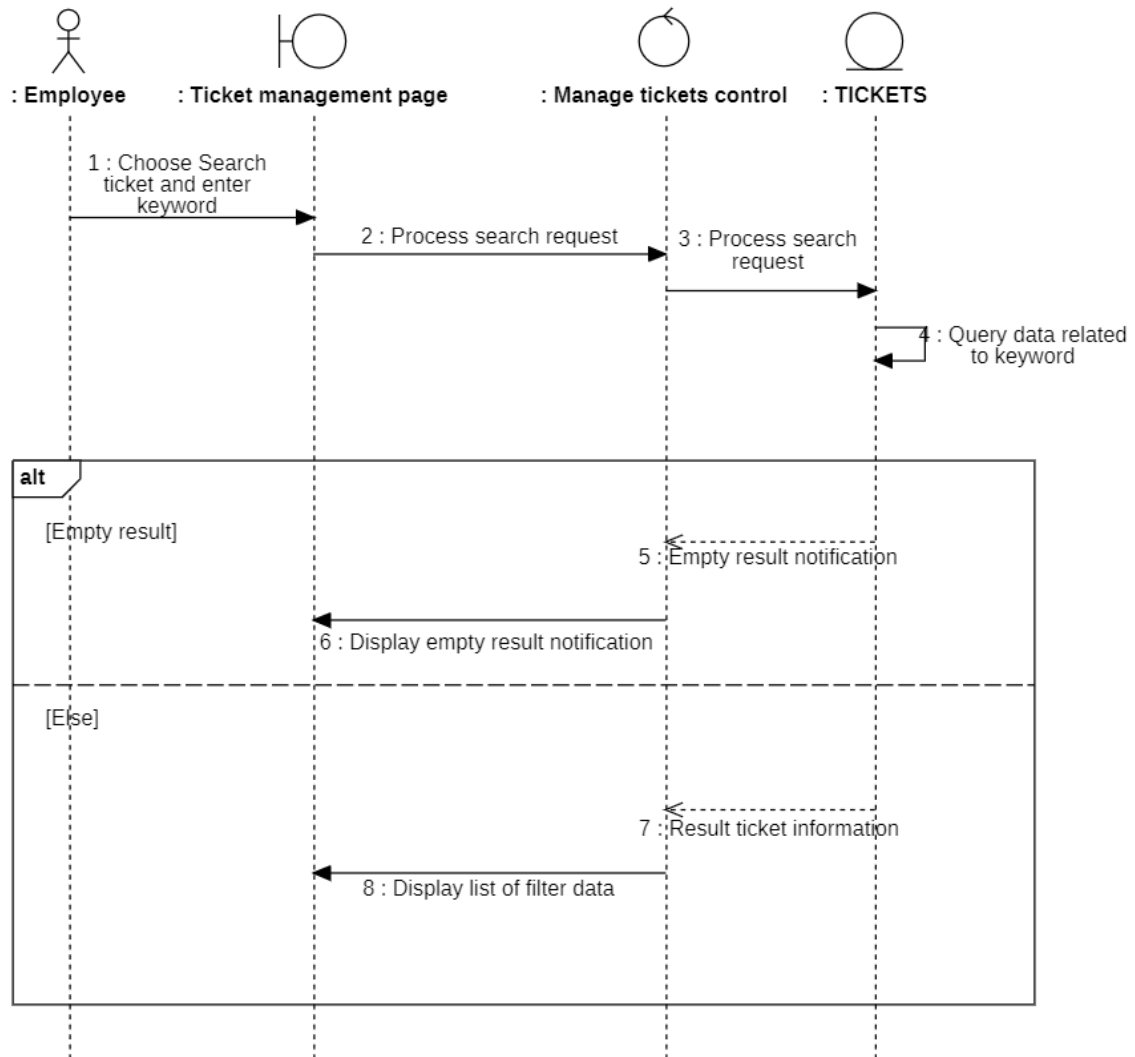
-Update ticket



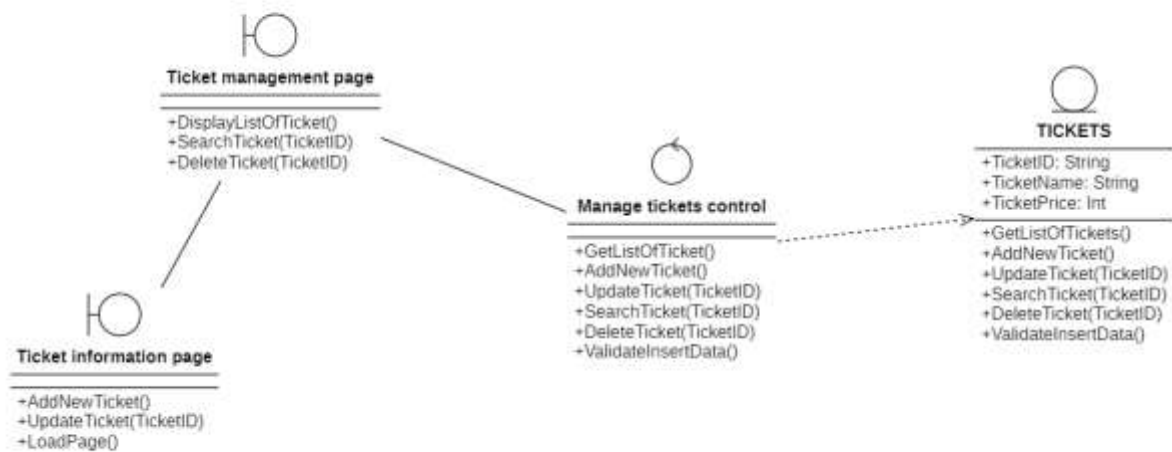
-Delete ticket



-Search ticket

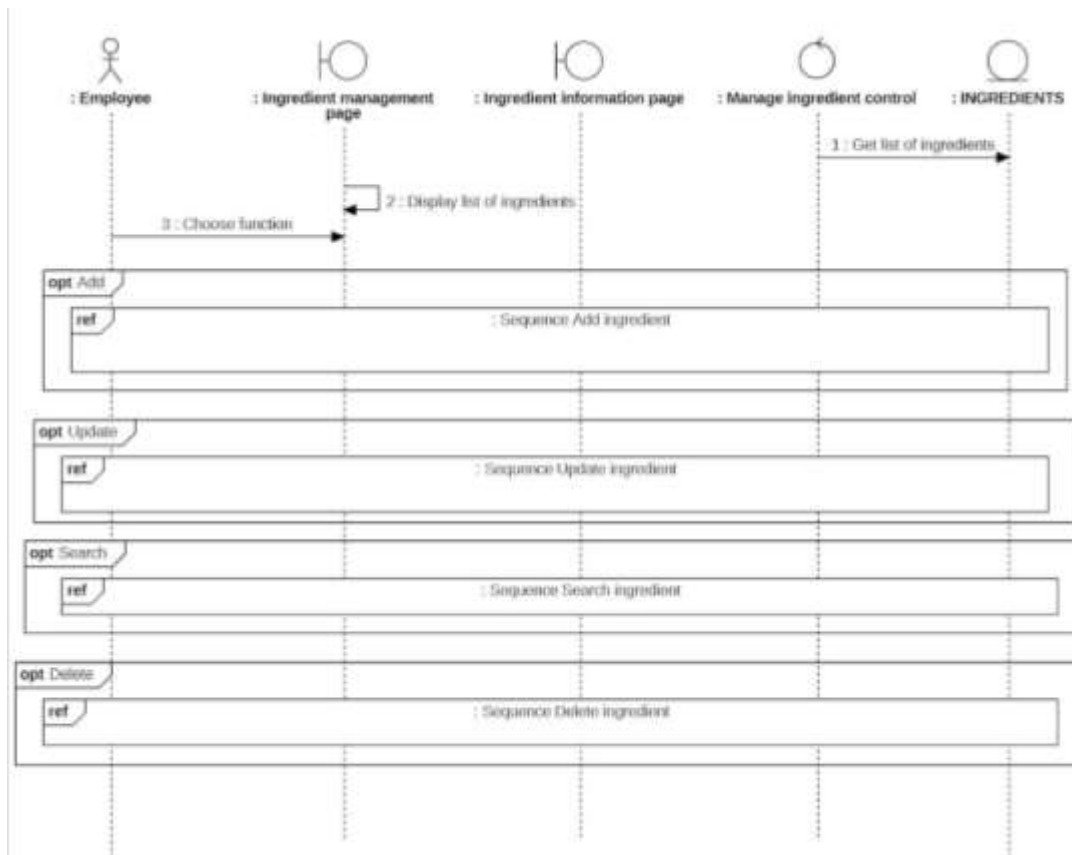


3.3.10.2 Class diagram

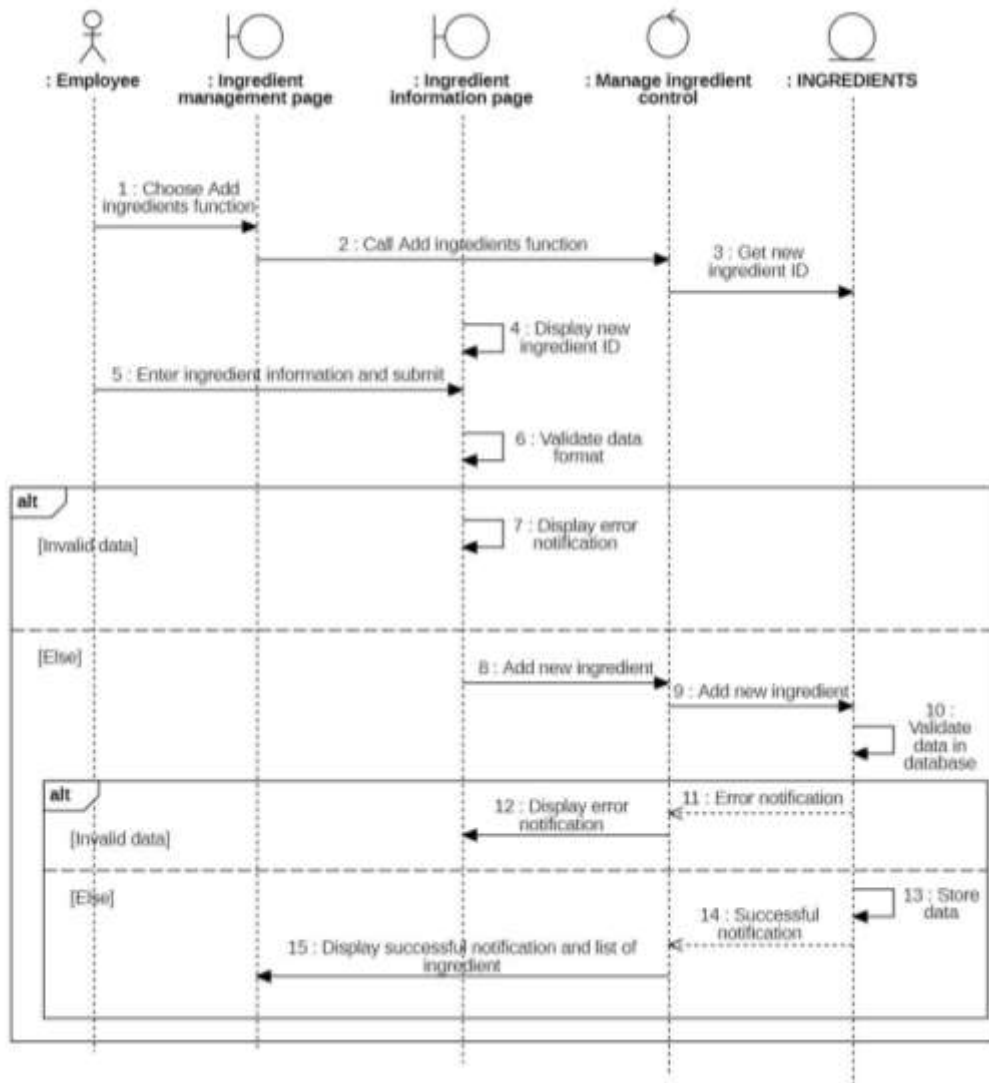


3.3.11 Manage ingredients

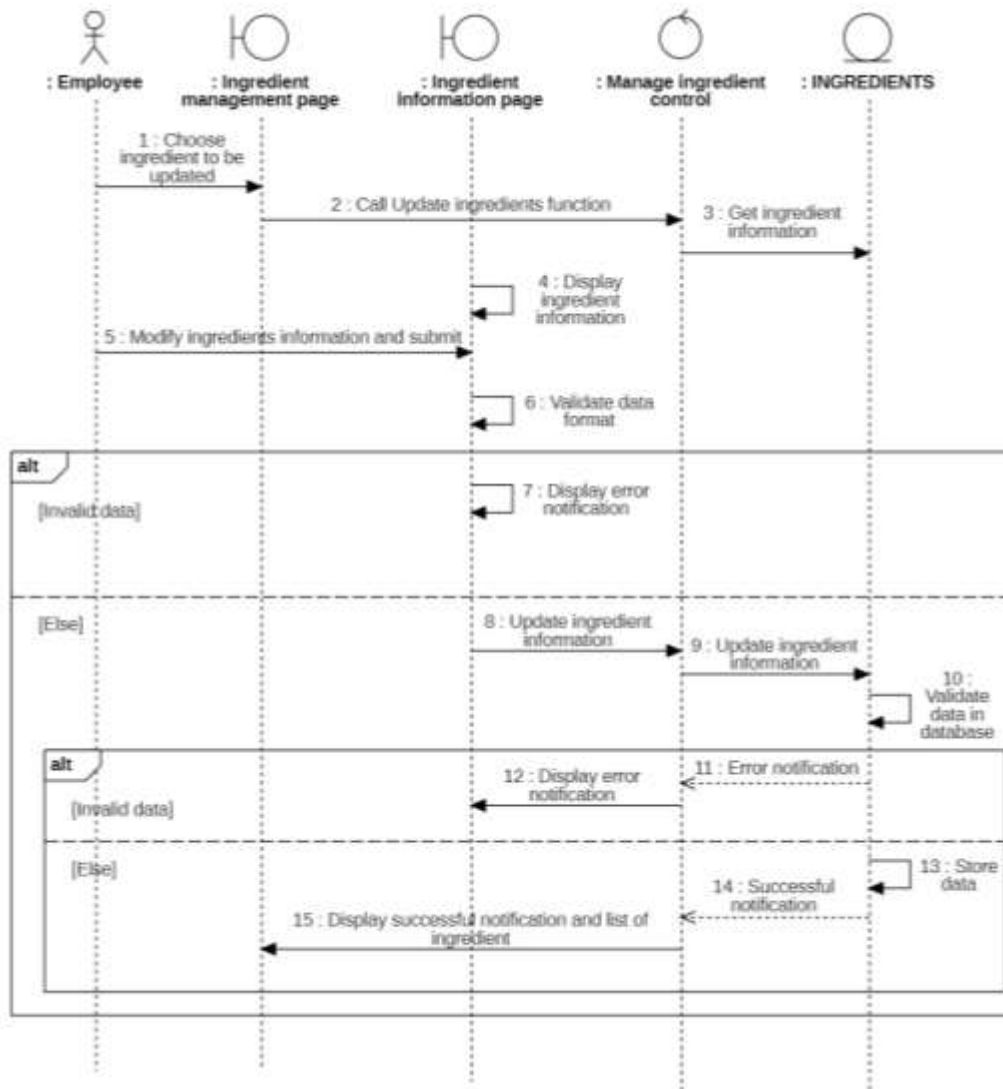
3.3.11.1 Sequence diagram



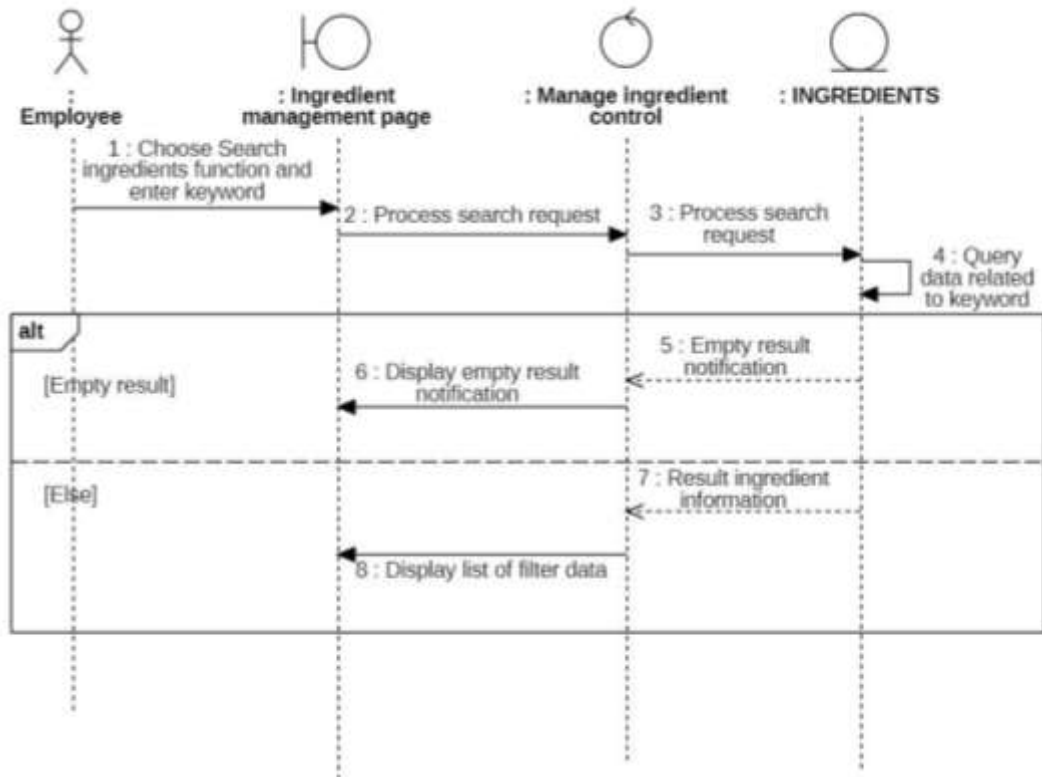
- Add ingredient



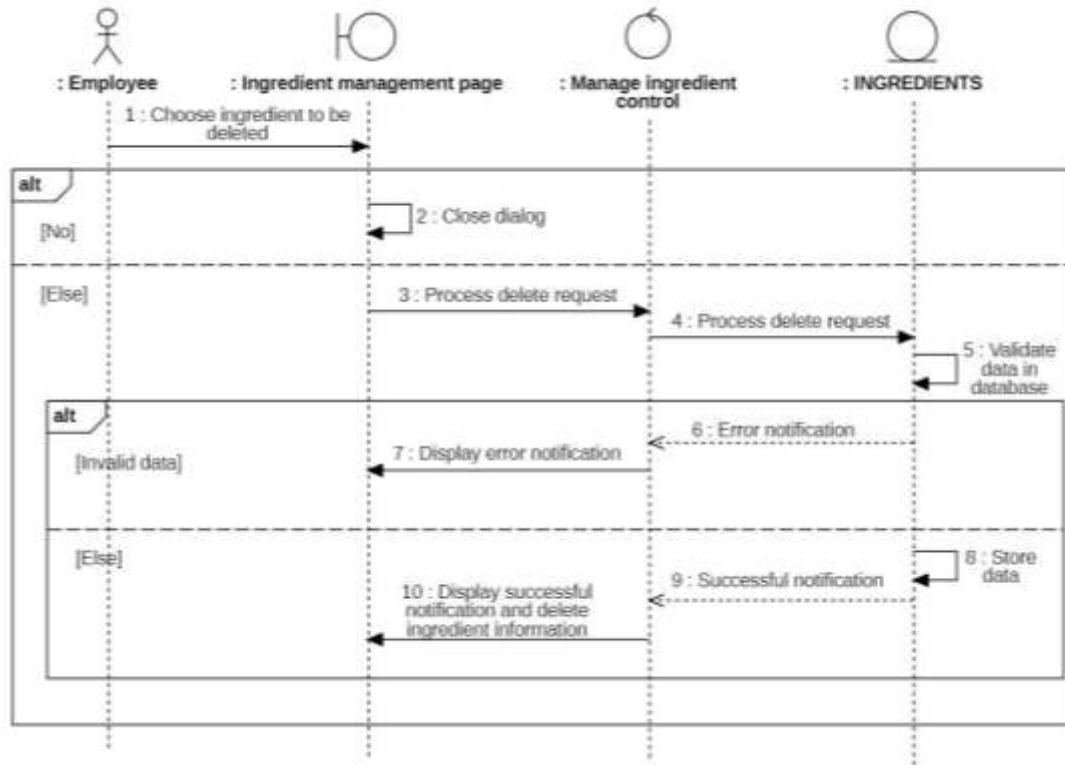
- Update ingredient



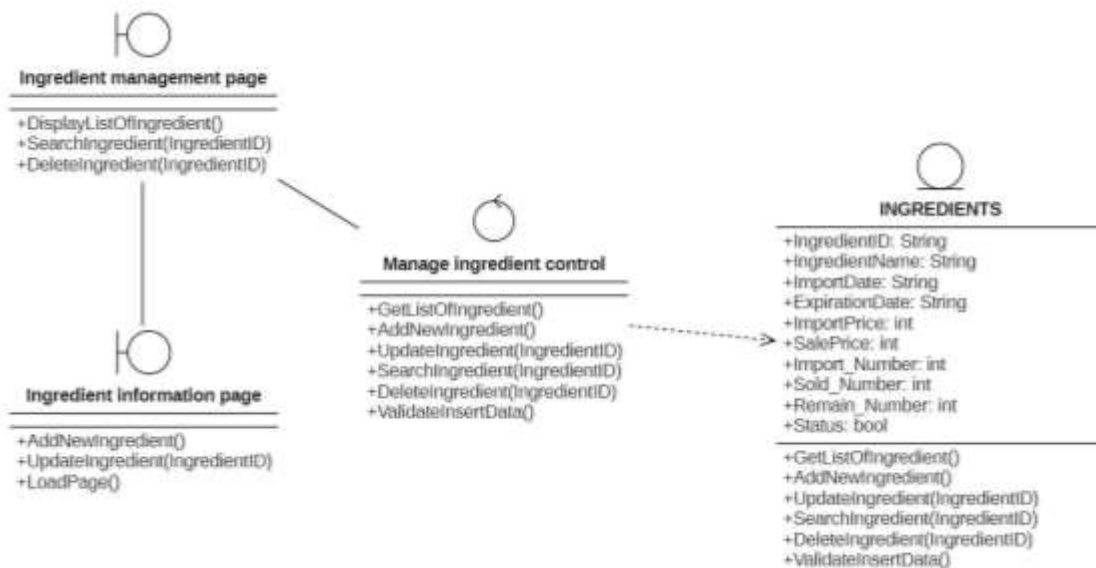
-Search ingredient



- Delete ingredient

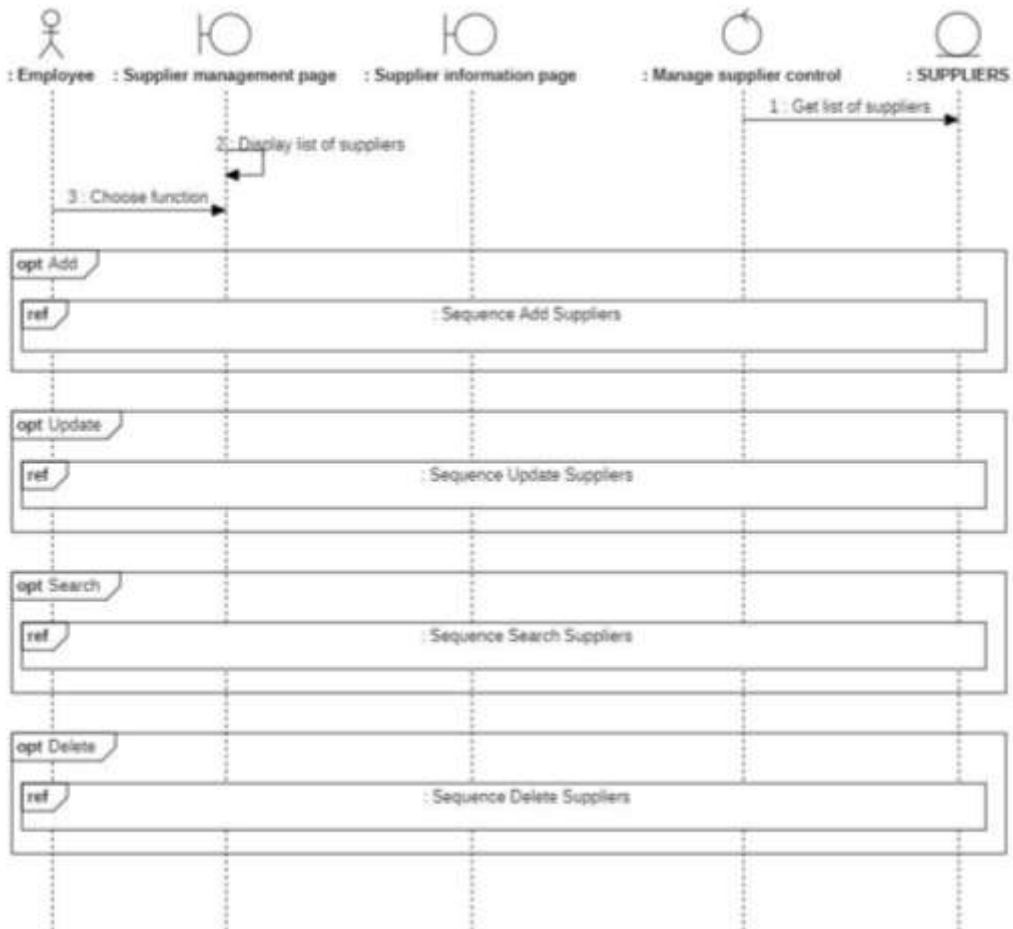


3.3.11.2 Class diagram

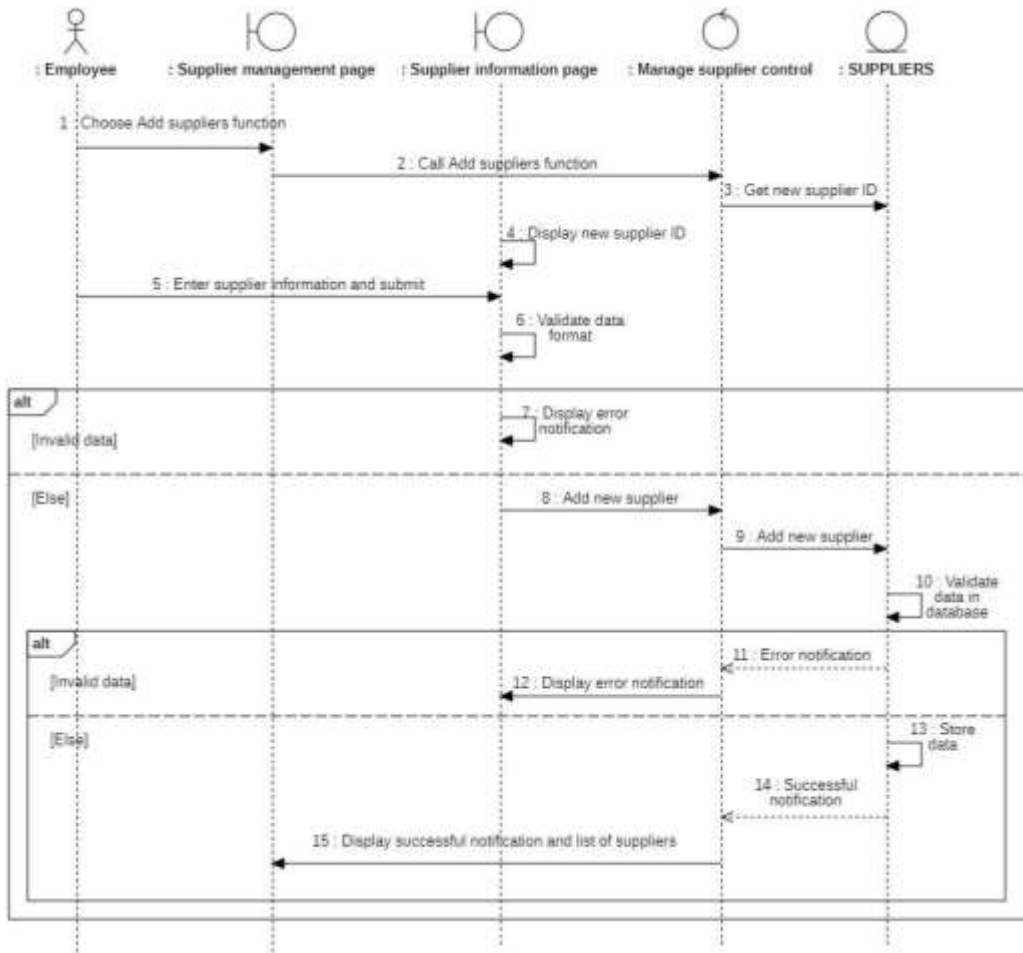


3.3.12 Manage suppliers

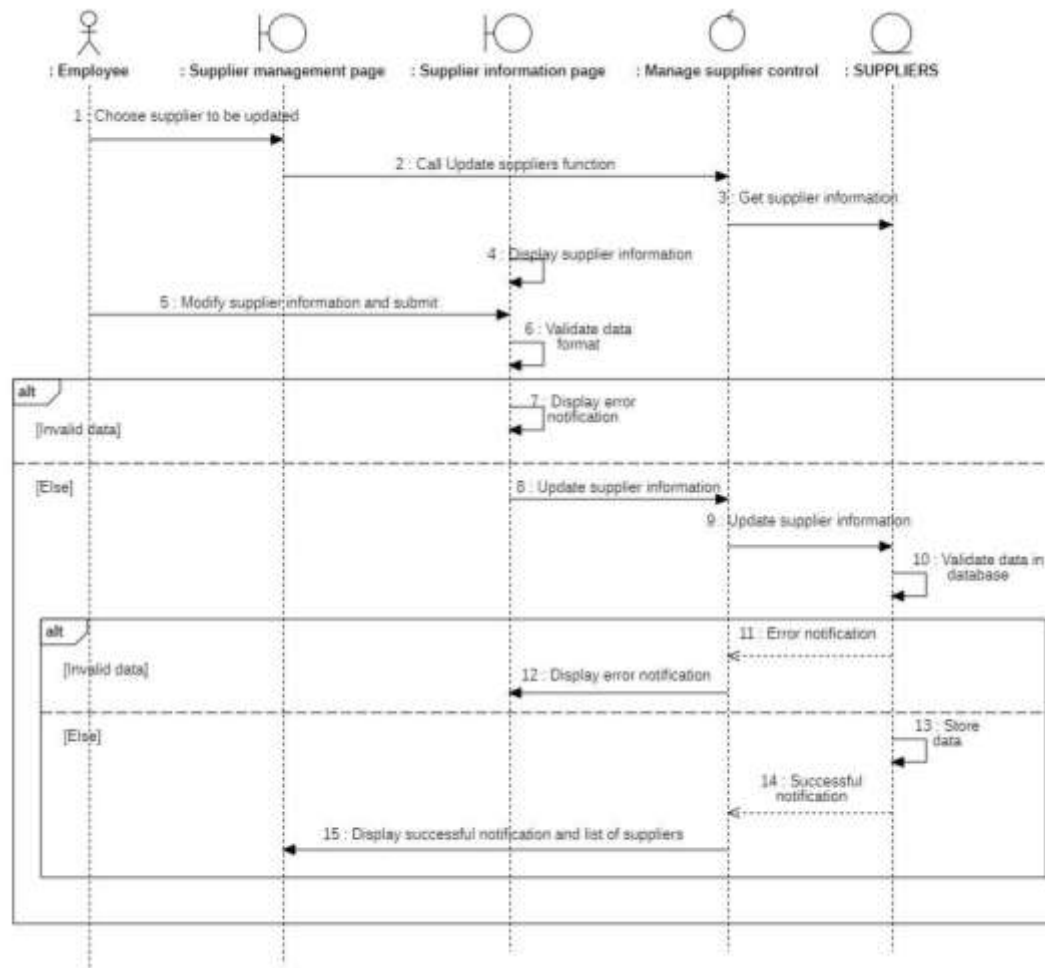
3.3.12.1 Sequence diagram



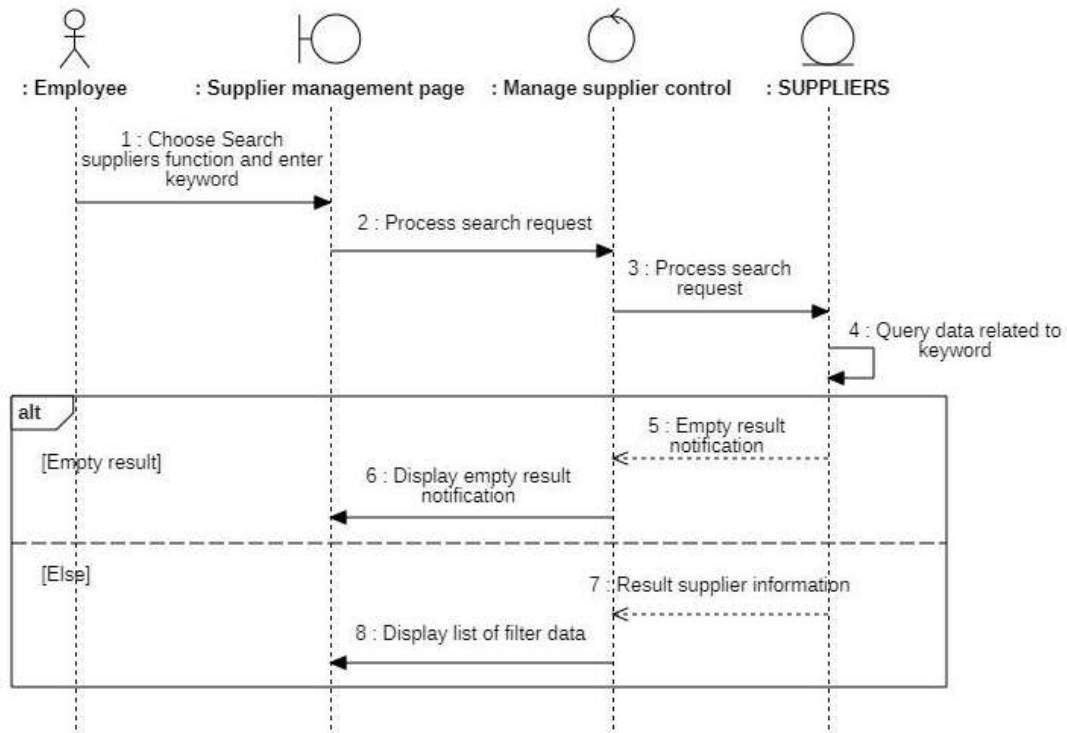
- Add supplier



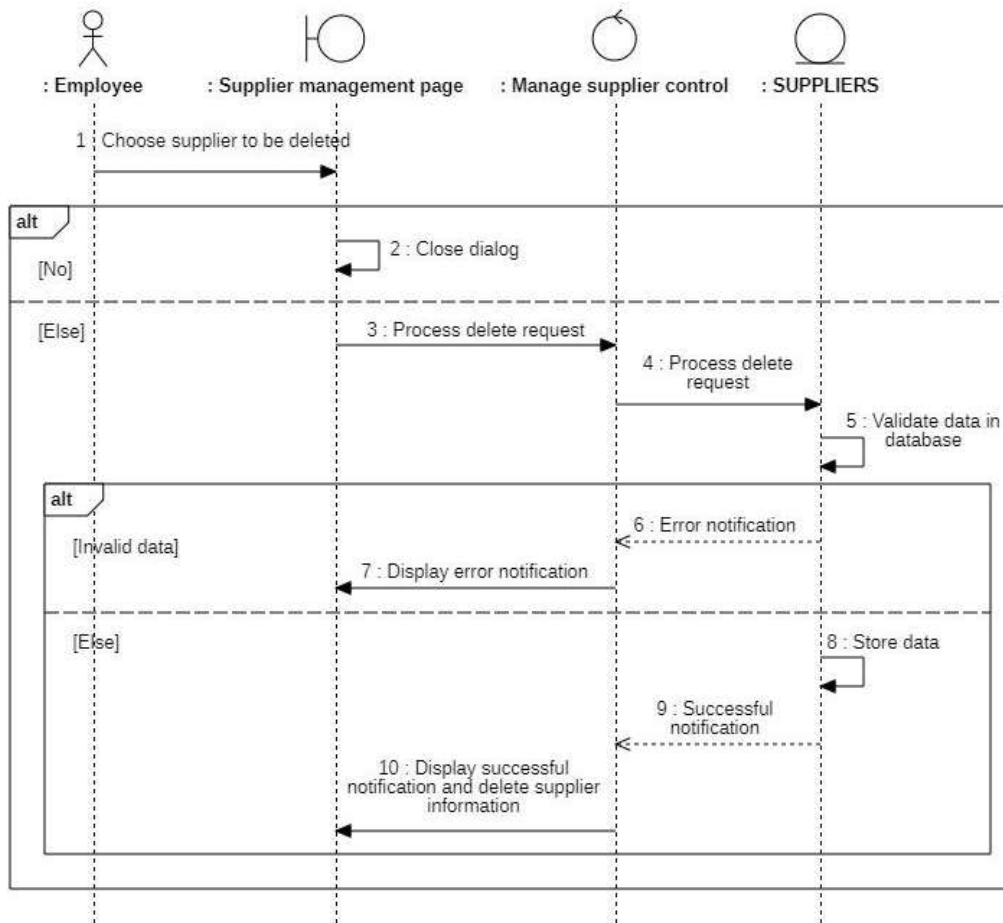
- Update supplier



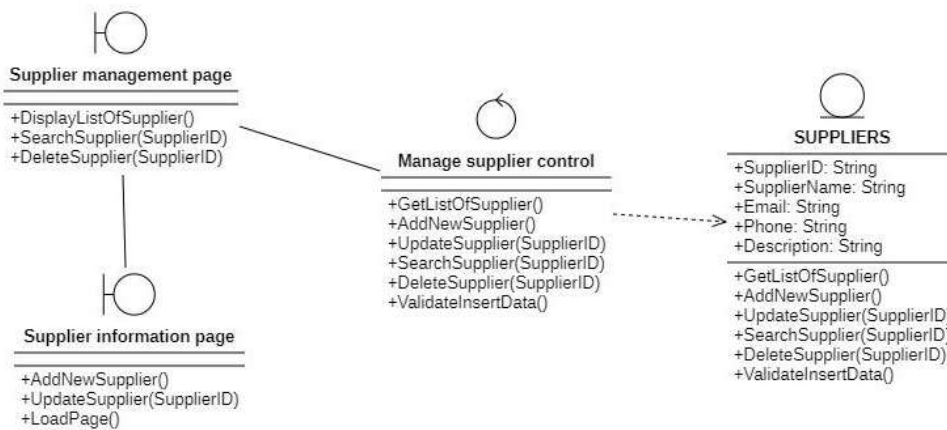
- Search supplier



- Delete supplier

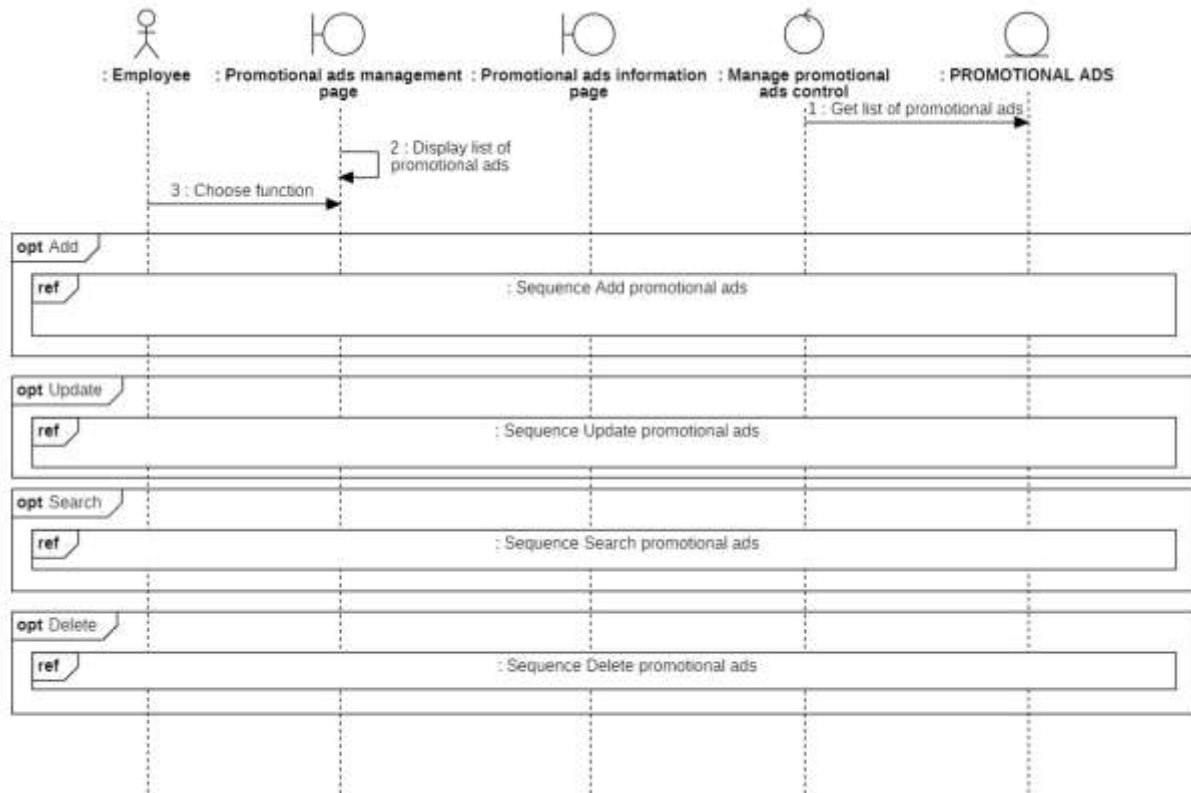


3.3.12.2 Class diagram

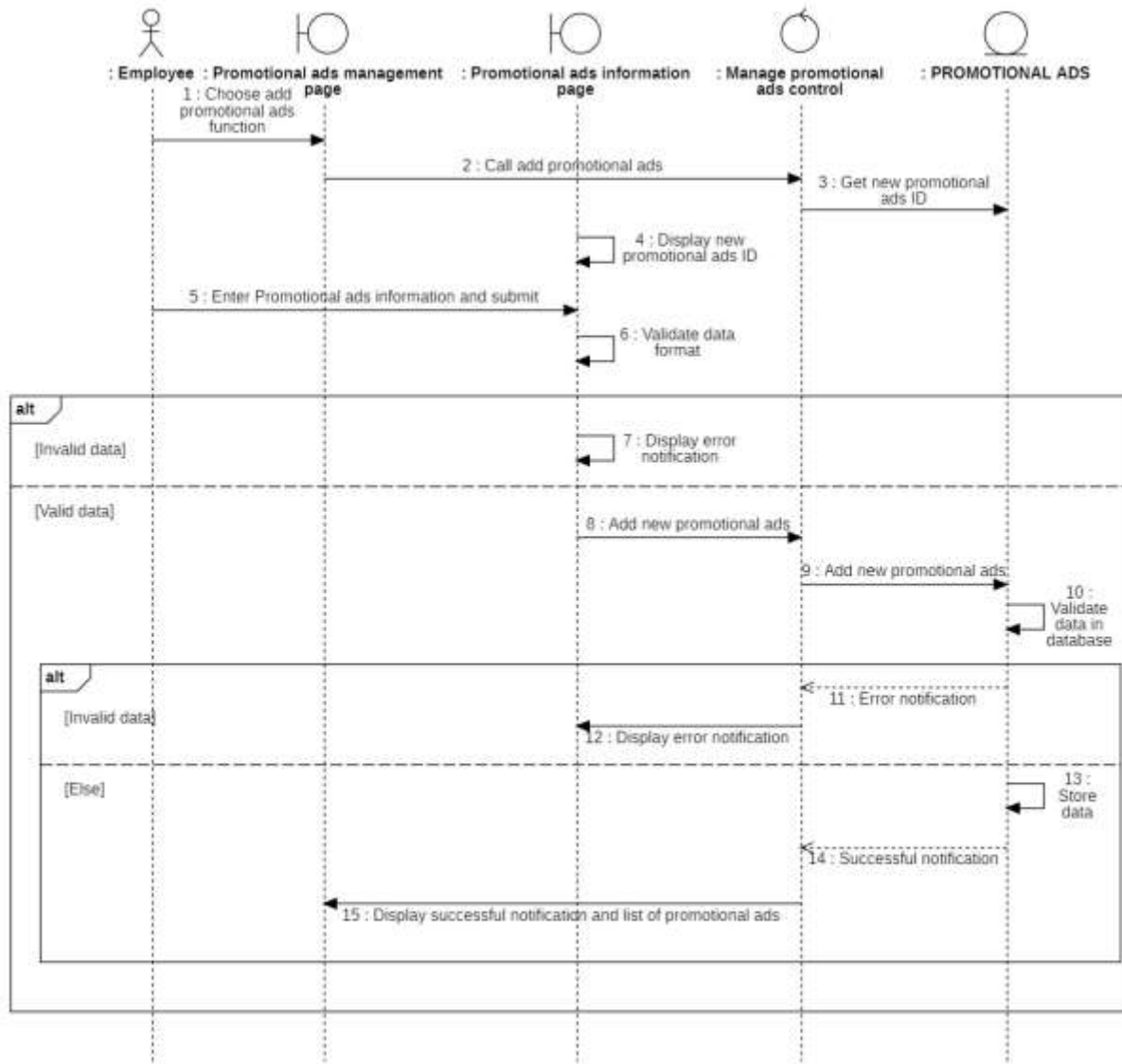


3.3.13 Manage promotional ads

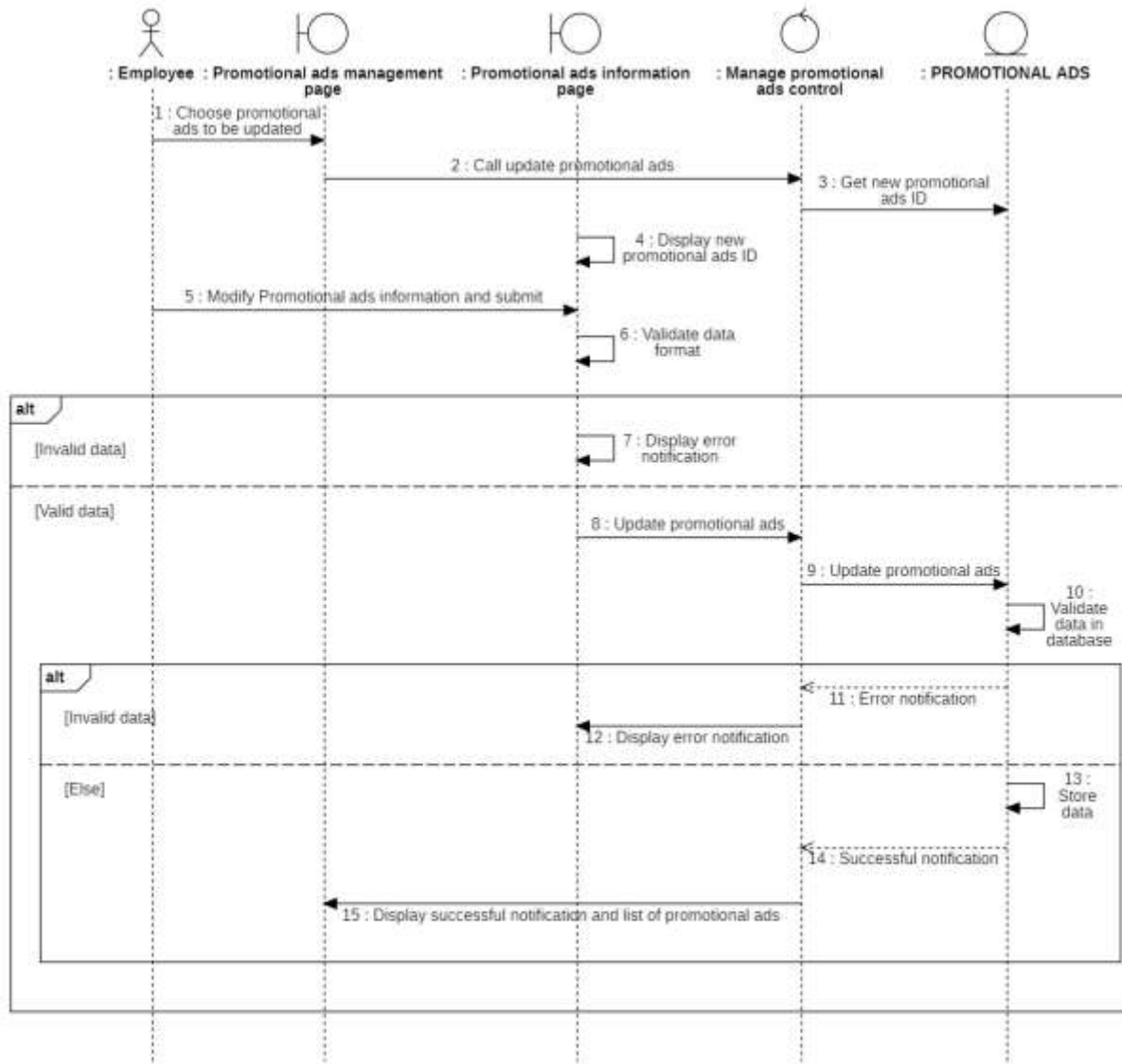
3.3.13.1 Sequence diagram



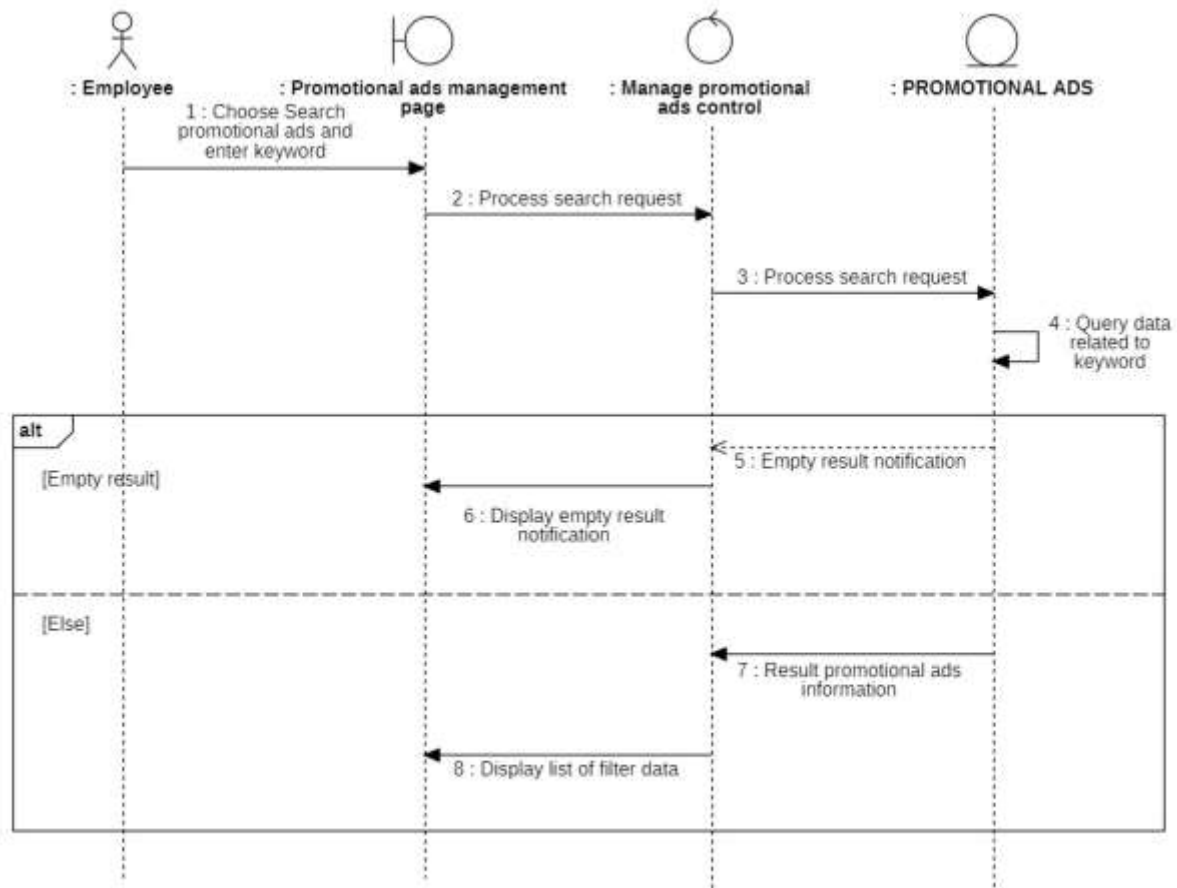
-Add promotional ads



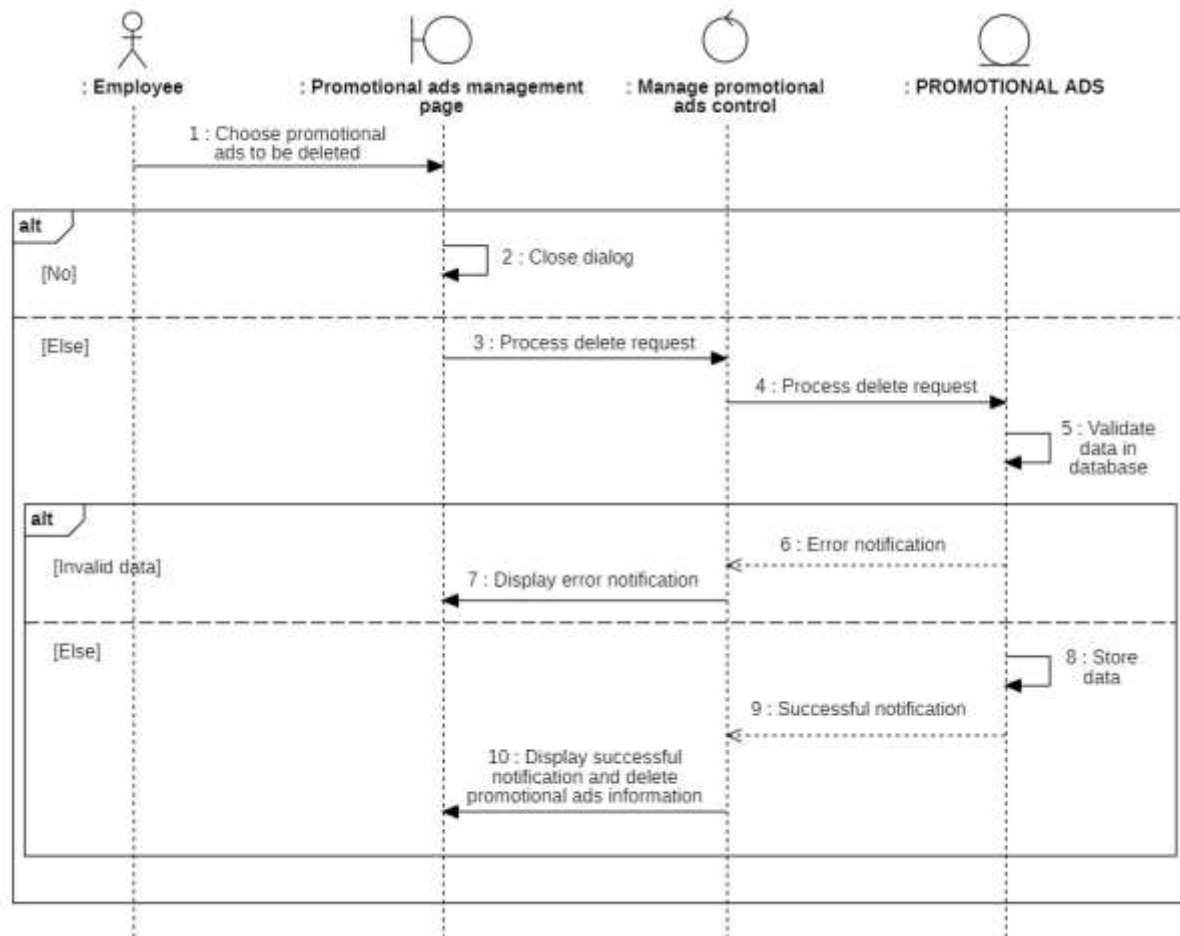
-Update promotional ads



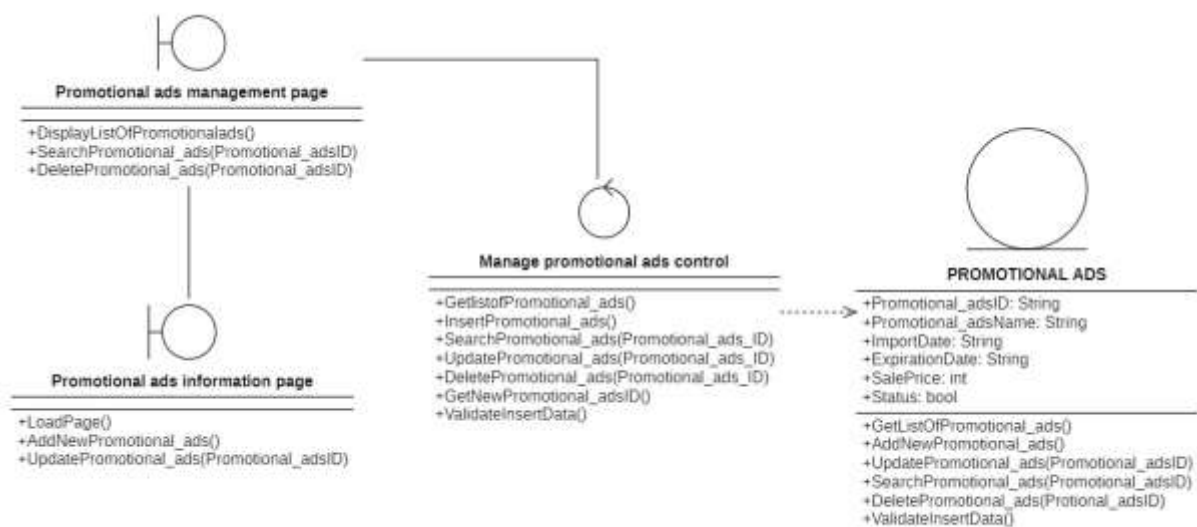
-Search promotional ads



-Delete promotional ads

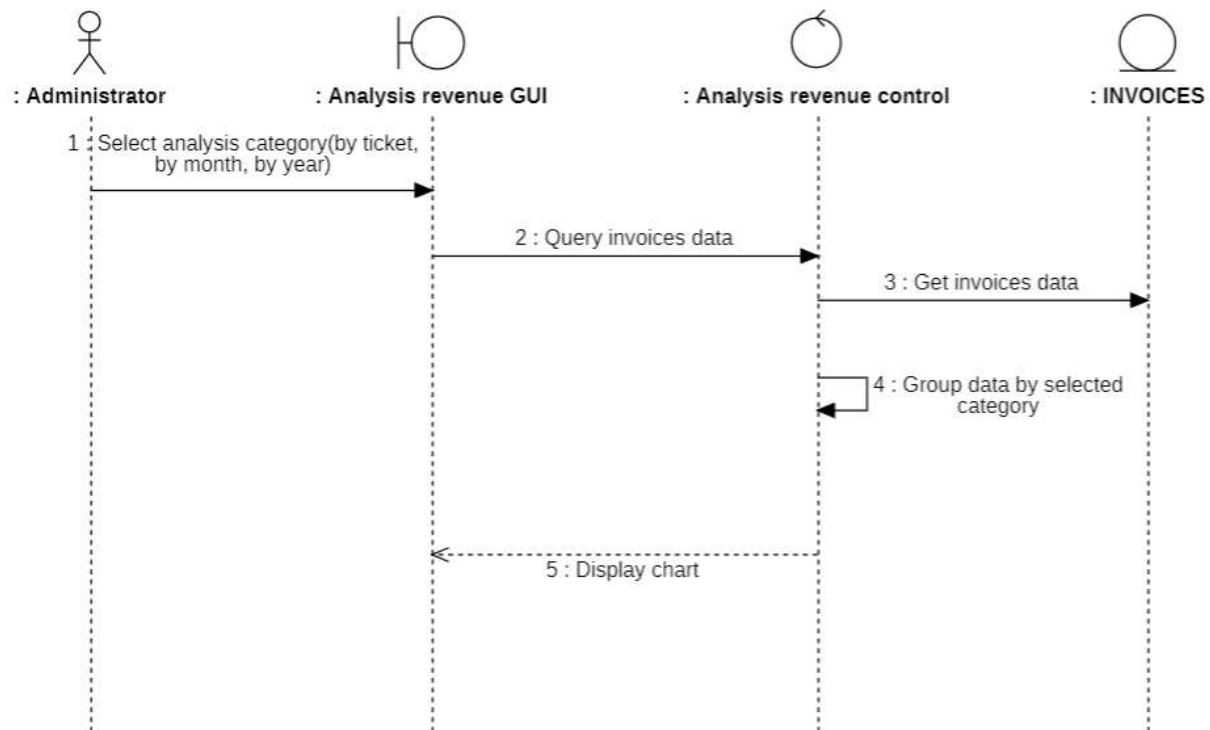


3.3.13.2 Class diagram

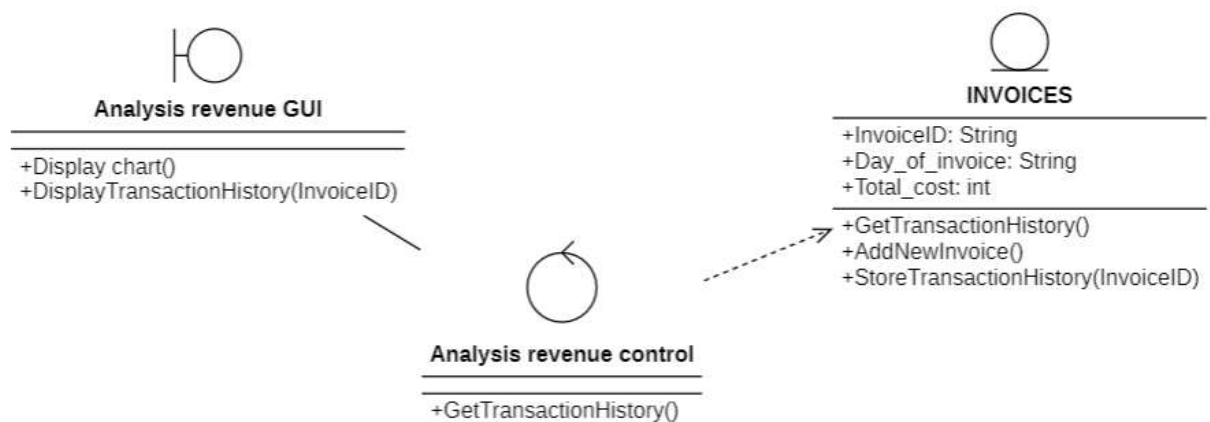


3.3.14 Analysis Revenue

3.3.14.1 Sequence diagram

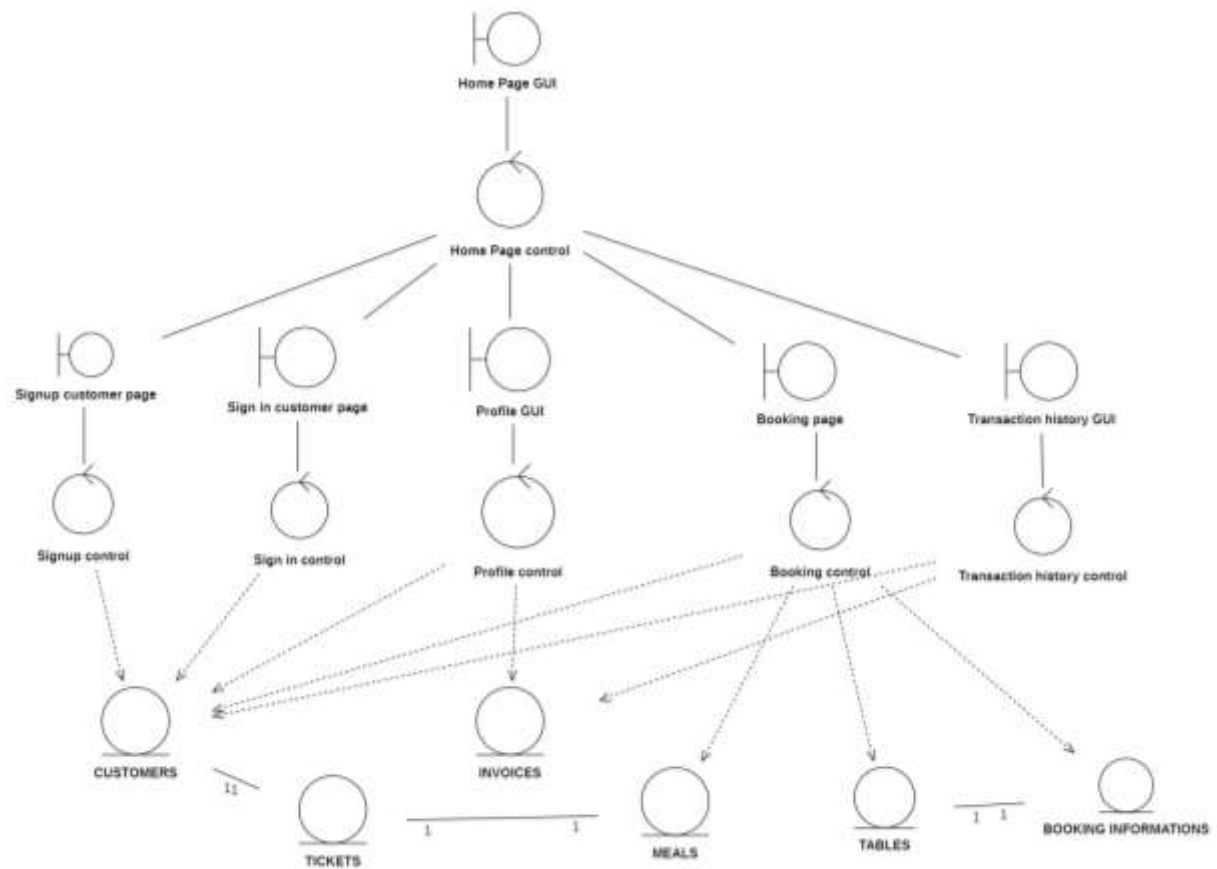


3.3.14.2 Class diagram



3.4 Class diagram and entity class diagram

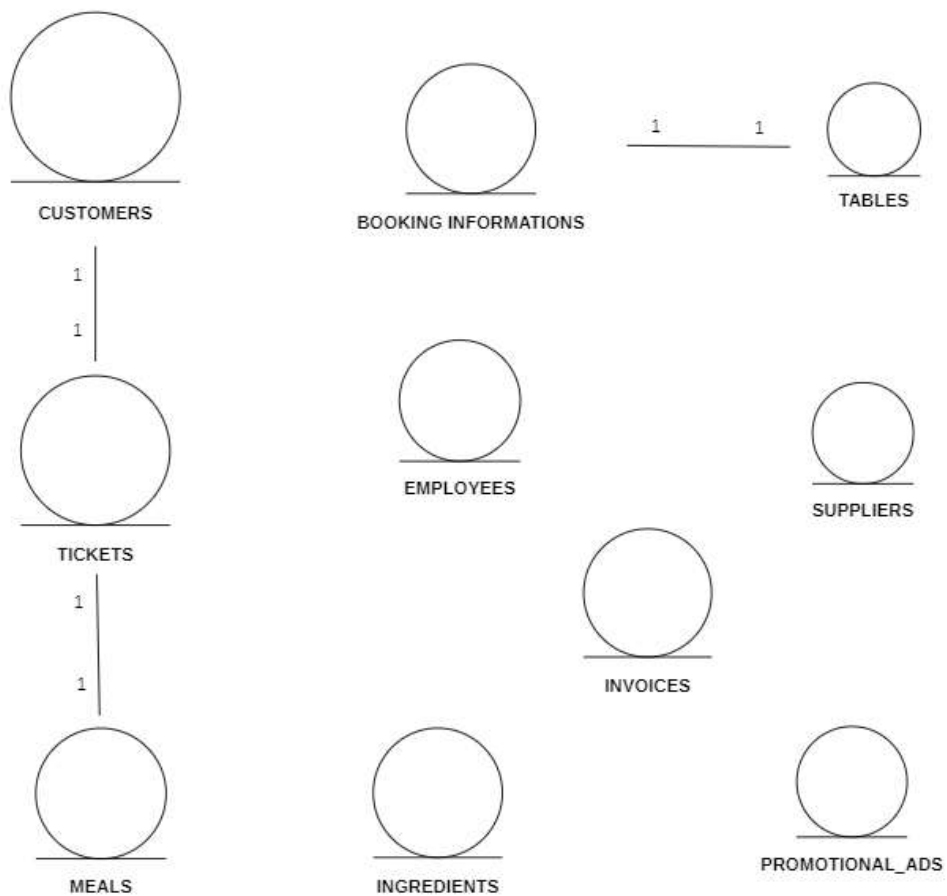
3.4.1 Customer overview class diagram



3.4.2 Administrator overview class diagram



3.4.3 Entity class diagram



3.4.4 Class diagram specification

3.4.4.1 Customer

Customer		
Attribute	Variable	Meaning
	Customer_ID: String	Customer ID generated by system
	Customer_Name: String	Full name of customer
	Gender: String	Gender of customer
	Phone: String	Phone number of customer
	Email: String	Email of customer
	UserName: String	User name of customer
	Password: String	Password of customer
Operation	Function	Meaning
	Get_list_of_customer(): Array	Get list of customers to perform on the customer management page of administrator
	AddNewCustomer(): Bool	Create a new customer on the database. Validation function is prerequisite for this function
	UpdateCustomer(Customer_ID): Bool	Update information of an existing customer. Validation function is prerequisite for this function
	DeleteCustomer(CustomerID): Bool	Delete a customer existed in the database and return to a boolean variable

	SearchCustomer(CustomerID): Array	Search customers related to keyword for displaying on the customer management page of administrator
	ValidateInsertData(): Bool	Validate if the inserting data is duplicate to the existed data
	GetCustomerInformation(): Customer	Get customer information
	StoreCustomerInformation(Customer_ID): Bool	Store Customer information after validation
	StoreTransactionHistory(): Bool	Store the transaction history after making purchase

3.4.4.2 Invoice

Invoice		
Attribute	Variable	Meaning
	InvoiceID: String	Invoice ID generated by system
	Day_of_invoice: String	The create day of invoice
	Total_cost: Int	Total cost of the invoice
Operation	Function	Meaning

	GetTransactionHistory(): String	Request database to get the transaction history of customer
	AddNewInvoice(): Bool	Create a new invoice on the database. Validation function is prerequisite for this function
	StoreTransactionHistory(InvoiceID): Bool	Create new invoice for customer after making a purchase and store it to database

3.4.4.3 Administrator

Administrator		
Attribute	Variable	Meaning
	AdminID: String	Admin ID generated by system
	AdminName: String	Full name of administrator
	Username: String	Username of administrator
	Password: String	Password of administrator
Operation	Function	Meaning
	GetlistofAdministrator(): Array	Get list of administrator

	StoreEmployeeInformation(EmployeeID): Bool	Store administrator information after validation
--	--	--

3.4.4.4 Table

Table		
Attribute	Variable	Meaning
	TableID: String	Table ID generated by system
	TableName: String	Name of table
	SeatAmount: Int	Seats amount of table
	TableSeats: Seatsmap	Seats map of the restaurant. Seatsmap is self-defining data type.
Operation	Function	Meaning
	GetListOfTable(): Array	Get list of tables
	ValidateInsertData(Table): Bool	Validate if the inserting data is duplicate to the existed data
	InsertTable(Table): Bool	Create a new table in the database. Validation function is prerequisite for this function
	GetTable(ID): Table	Return a table information matched with the ID. This function is used to get a specific data to pour in the information form
	UpdateTable(Table): Bool	Update information of an existed table. Validation function is prerequisite for this function
	DeleteTable(ID): Bool	Delete a table existed in the database and return to a boolean variable
	SearchTableByKeyword(String): Array	Return a list of tables related to the keyword.
	GetTableSeats(): SeatsMap	Get the seats map related to the table.

3.4.4.5 Meal

Meal		
Attribute	Variable	Meaning
	Meal_ID: String	Meals ID generated by system automatically
	Meal_Name: String	Name of the Meal
	Ingredients: String	the detailed ingredients of the meal
Operation	Function	Meaning
	Get_list_of_meals(): Array	Get list of all existed meal to perform on GUI

3.4.4.6. Ticket

Ticket		
Attribute	Variable	Meaning
	TicketID: String	Ticket ID generated by system
	TicketName: String	Name of ticket
	TicketPrice: Int	Price of ticket
Operation	Function	Meaning
	GetListOfTickets(): Array	Request database to get list of tickets
	AddNewTicket(): Bool	Create a new ticket on the

		database. Validation function is prerequisite for this function
	UpdateTicket(TicketID): Bool	Update information of an existing ticket. Validation function is prerequisite for this function
	SearchTicket(TicketID): Array	Return a list of ticket related to the keyword.
	DeleteTicket(TicketID): Bool	Delete a ticket existed in the database and return to a boolean variable
	ValidateInsertData(): Bool	Validate if the inserting data is duplicate to the existed data

3.4.4.7. Employee

Employee		
Attribute	Variable	Meaning
	EmployeeID: String	Employee ID generated by system
	EmployeeName: String	Full name of employee
	Gender: String	Gender of employee
	Phone: String	Phone number of employee
	Email: String	Email of employee
	Address: String	Address of employee

	Role: String	Role of employee
	Salary: int	Salary of employee
	Username:String	Username of employee
Operation	Function	Meaning
	Getlistofemployee(): Array	Get list of employees
	AddEmployee (): Bool	Create a new employee in the database. Validation function is prerequisite for this function
	SearchEmployee(EmployeeName): Array	Return a list of employees related to keyword
	UpdateEmployee(EmployeeID): Bool	Update information of an employee existed in the database. Validation function is prerequisite for this function
	DeleteEmployee(EmployeeID): Bool	Delete a employee existed in the database and return to the boolean variable
	ValidateInsertData(): Bool	Validate if the inserting data is duplicate to the existed data
	GetEmployeeInformation(): Employee	Get employee information
	StoreEmployeeInformation(EmployeeID): Bool	Store employee information after validation

3.4.4.8 Ingredient

Ingredient		
Attribute	Variable	Meaning
	IngredientID: String	Ingredient ID generated by system

	IngredientName: String	Name of ingredient
	ImportDate: String	Date ingredients import to storage
	ExpirationDate: String	Date ingredients expire
	SalePrice: Int	Price of ingredient
	Import_Number: Int	The number of ingredients imported
	Sold_Number: Int	The number of ingredients is sold
	Remain_Number: Int	The number of ingredients remains
	Status: Boolean	The status of ingredients
Operation	Function	Meaning
	GetListOfIngredient(): Array	Get list of ingredients
	ValidateInsertData(Ingredient): Bool	Validate if the inserting data is duplicate to the existed data
	AddNewIngredient(): Boolean	Insert a new ingredient in the database. Validation function is prerequisite for this function
	UpdateIngredient(): Boolean	Update information of an existed ingredient. Validation function is prerequisite for this function
	DeleteIngredient(): Boolean	Delete an ingredient existed in the database and return to a boolean variable
	SearchIngredientByKeyword(String): Array	Return a list of ingredients related to the keyword.

3.4.4.9 Supplier

Supplier		
	Variable	Meaning
	SupplierID: String	Supplier ID generated by system

Attribute	SupplierName: String	Full name of supplier
	Phone: String	Phone number of supplier
	Email: String	Email of supplier
	Description: String	the people who provide the necessary ingredients for the restaurant
Operation	Function	Meaning
	GetListOfSupplier(): Array	Get list of all existed Suppliers to perform on GUI
	AddNewSupplier(): Bool	Create a new supplier on the database. Validation function is prerequisite for this function
	UpdateSupplier(SupplierID): Bool	Update information of an existing supplier. Validation function is prerequisite for this function
	SearchSupplier(SupplierID): Array	Return a list of suppliers related to the keyword.
	DeleteSupplier(SupplierID): Bool	Delete a supplier existed in the database and return to a boolean variable
	ValidateInsertData(): Bool	Validate if the inserting data is duplicate to the existed data

3.4.4.10 Customer sign in control

Customer sign in control		
Operation	Function	Meaning

	CheckSignIn(CustomerID, UserName, PassWord): Bool	Request database to check information of customer. Validation function is prerequisite for this function
	ValidateInsertData(): Bool	Validate if the insert information format is correct

3.4.4.11 Customer sign up control

Customer sign up control		
Opretation	Function	Meaning
	CreateNewCustomer(CustomerID): Bool	Request the database to create a newncustomer on it. Validation function is prerequisite for this function
	CreateNewAccount(CustomerID, UseName, Password): Bool	Request the database to create a new account for customer. Validation function is prerequisite for this function
	ValidateInsertData() : Bool	Validate if the insert information format is correct

3.4.4.12 Profile control

Profile control		
Operation	Function	Meaning

	GetCustomerInformation(): Customer	Request database to get customer profile information to display on Profile GUI
	GetTransactionHistory()	Request database to get all the transaction history related to this customer
	ValidateDataFormat(): Bool	Validate data is in the correct format or not

3.4.4.13 Booking control

Booking control		
Operation	Function	Meaning
	DisplayBookingInformationFrom(): Void	Display booking information form for updating or inserting
	GetListOfTables(): Array	Request database to get list of tables
	GetListOfMeals(): Array	Request database to get list of meals
	GetListOfTickets(): Array	Request database to get list of tickets
	GetCustomerInformation(ID): String	Request database to get customer information based on customerID
	GetTransactionHistory(ID): String	Request database to get the transaction history of customer based on customerID
	ProcessBookingRequest(): Boolean	Store the request to database
	GetTicketPrice(ID): Int	Request database to get price of ticket based on ticketID
	ProcessPayment(): Boolean	Store the payment to database after user make a purchase successfully

3.4.4.14 Search control

Search control		
Operation	Function	Meaning
	GetTransactionHistory(): String	Request database to get information of a transaction history selected by user
	SearchTransactionHistoryByKeyword(): Array	Request database to get a list of transaction history system related to the string parameter

3.4.4.15 transaction history control

Transaction history control		
Operation	Function	Meaning
	GetCustomerInformation(): String	Request database to get information of customer selected by user
	GetCustomerTransactionHistory(CustomerID): String	Request database to get customer transaction history system related to the string parameter

3.4.4.16 Admin sign in control

Admin sign in control		
Operation	Function	Meaning
	CheckSignIn(EmployeeID, username, password): Bool	Check information of administrator to login to the database
	ValidateInputData(): Bool	Validate if the input information format is correct

3.4.4.17 Manage customer control

Manage customer control		
Operation	Function	Meaning
	GetListOfCustomer(): Array	Request database to get list of all existed customers
	ValidateInsertData(): Boolean	Validate if the inserting data is duplicate to the existed data
	AddNewCustomer(): Boolean	Request database to create a new customer. Validation function is prerequisite for this function
	UpdateCustomer(CustomerID): Boolean	Request database to make a change on an existing customer. Validation function is prerequisite for this function
	DeleteCustomer(CustomerID): Boolean	Request database to delete a customer selected by the user
	SearchCustomer(String): Array	Request database to get a list of customers related to the string parameter

3.4.4.18 Manage employee control

Manage employee control		
Operation	Function	Meaning
	GetListOfEmployee(): Array	Request database to get list of Employee
	AddEmployee(): Bool	Request database to create a new employee. Validation function is prerequisite for this function
	UpdateEmployee(Employee_ID): Bool	Request database to make a change in an existing employee. Validation function is prerequisite for this function
	SearchEmployee(Employee_ID): Array	Request database to get a list of employee related to the string parameter
	DeleteEmployee(Employee_ID): Bool	Request database to delete a employee selected by the user
	GetNewEmployeeID(): EmployeeID	Request database to get new of a employeeid selected by the user
	ValidateInsertData(): Bool	Validate if the input information format is correct

3.4.4.19 Manage ticket control

Manage ticket control		
Operation	Function	Meaning
	GetListOfTicket(): Array	Request database to get list of ticket

	AddNewTicket() :Bool	Request database to create a new ticket. Validation function is prerequisite for this function
	UpdateTicket(TicketID) : Bool	Request database to make a change in an existing ticket. Validation function is prerequisite for this function
	SearchTicket(TicketID) :Array	Request database to get a list of ticket related to the string parameter
	DeleteTicket(TicketID) : Bool	Request database to delete a ticket selected by the user
	ValidateInsertData(): Bool	Validate if the input information format is correct.

3.4.4.20 Manage ingredient control

Manage ingredient control		
Operation	Function	Meaning
	GetListOfIngredient(): Array	Request database to get list of ingredient

	AddNewIngredient(): Bool	Request database to create a new ingredient. Validation function is prerequisite for this function
	UpdateIngredient(IngredientID): Bool	Request database to make a change in an existing ingredient. Validation function is prerequisite for this function
	SearchIngredient(IngredientID): Array	Request database to get a list of ingredient
	DeleteIngredient(IngredientID): Bool	Request database to delete a ingredient selected by the user
	ValidateInsertData(): Bool	Validate if the input information format is correct

3.4.4.21 Manage supplier control

Supplier control		
Operation	Function	Meaning
	GetListOfSupplier(): Array	Request database to get list of all existed suppliers
	ValidateInsertData(): Boolean	Validate if the inserting data is duplicate to the existed data
	AddNewSupplier(): Boolean	Request database to create a new supplier. Validation function is prerequisite for this function
	UpdateSupplier(SupplierID): Boolean	Request database to make a change on an existing supplier. Validation function is prerequisite for this function
	DeleteSupplier(SupplierID): Boolean	Request database to delete a supplier selected by the user
	SearchSupplier(String): Array	Request database to get a list of suppliers related to the string parameter

3.4.4.22 Manage promotional ads control

Promotional ads		
Attribute	Variable	Meaning
	PromotionalAdsID: String	Promotional ads ID generated by system
	PromotionalAdsName:String	Name of promotional ads
	PromotionDate: String	Promotion start date
	ExpirationDate: String	Promotion end date
	SalePrice: Int	Each price has a different promotion
	Status: Bool	This status is used when Promotional ads signs in. True is has-sign-in status. False is the reverse
Operation	Function	Meaning
	GetListOfPromotionalAds(): Array	Get list of all existed promotional ads to perform on GUI
	AddNewPromotionalAds(): Bool	Create new promotional ads in the database. Validation function is prerequisite for this function
	UpdatePromotionalAds(PromotionalAds_ID): Bool	Update information of existing promotional ads. Validation function is prerequisite for this function

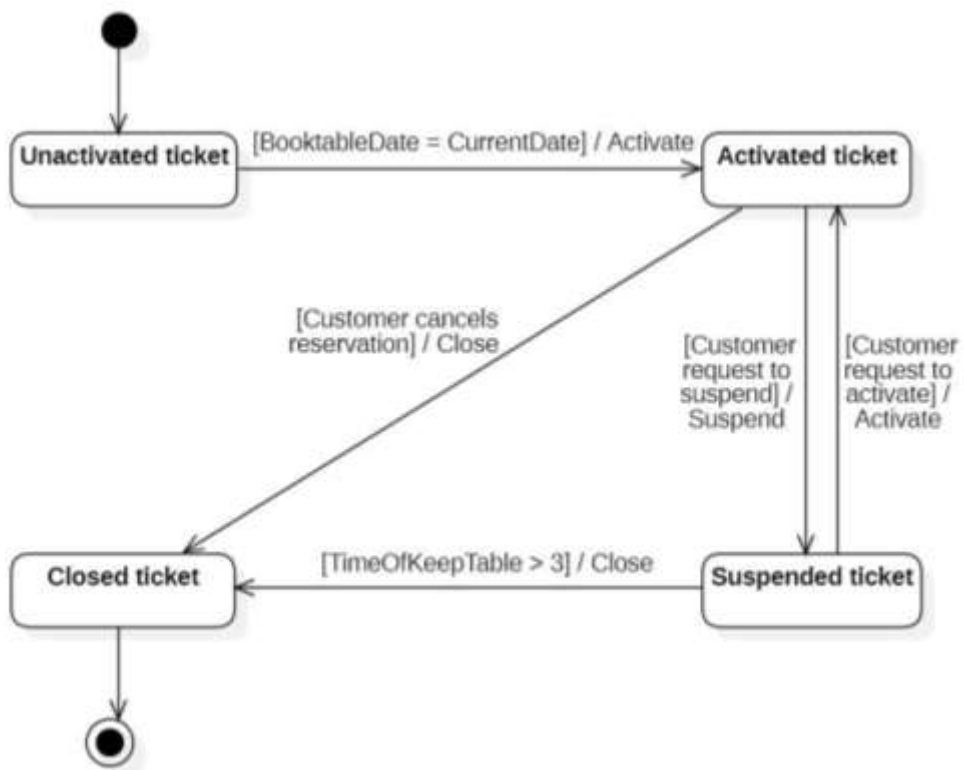
	SearchPromotionalAds(PromotionalAds_ID): Array	Return a list of promotional ads related to the keyword.
	DeletePromotionalAds(PromotionalAds_ID): Bool	Delete a promotional ads existed in the database and return to a boolean variable
	ValidateInsertData(): Bool	Validate if the inserting data is duplicate to the existed data

3.4.4.23 Revenue analysis control

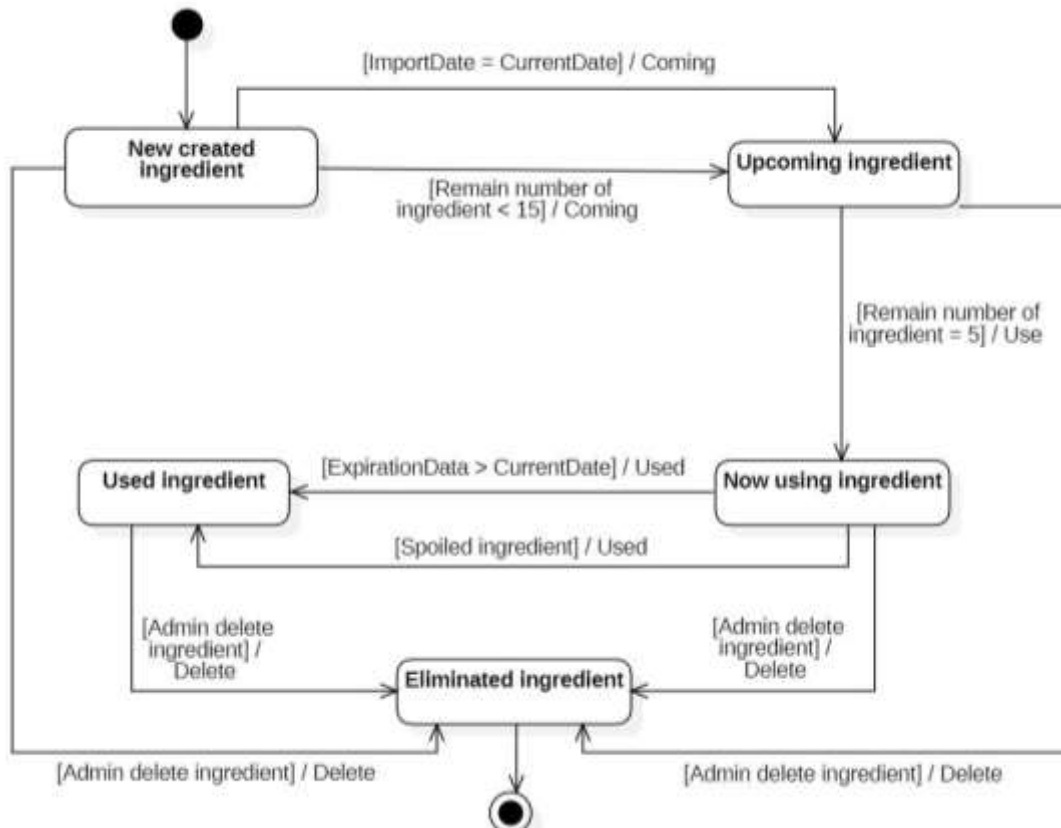
Revenue analysis control		
Operation	Function	Meaning
	GetTransactionHistory():Array	Request database to get information of a transaction history selected by user

3.5 State diagram

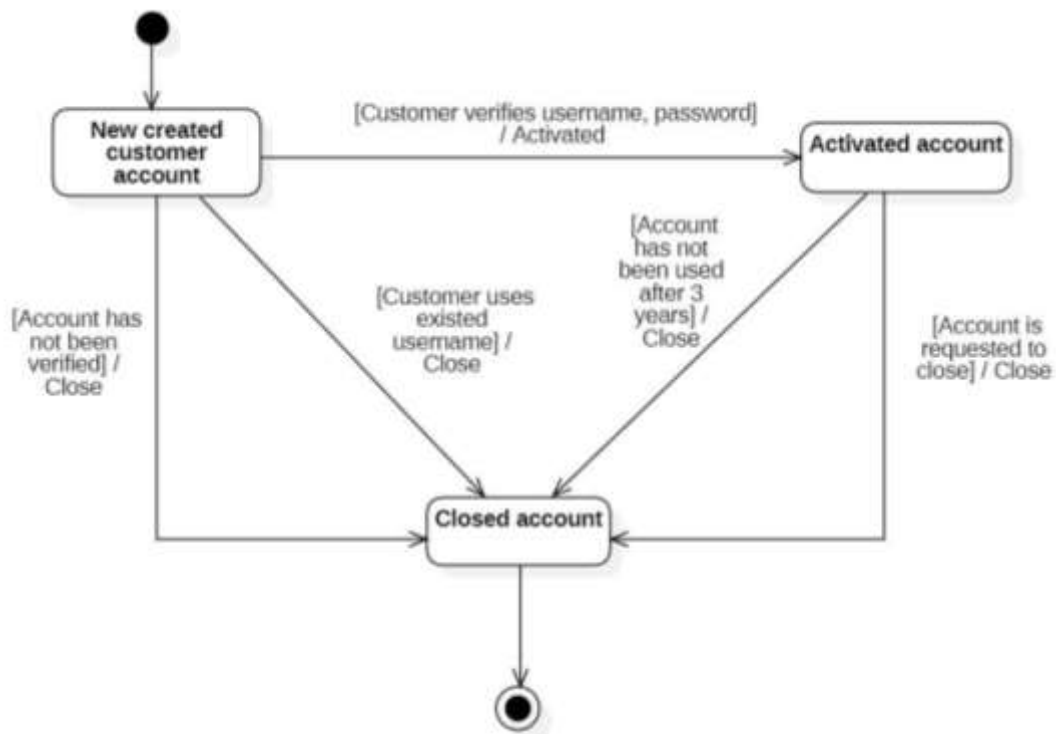
3.5.1 Ticket



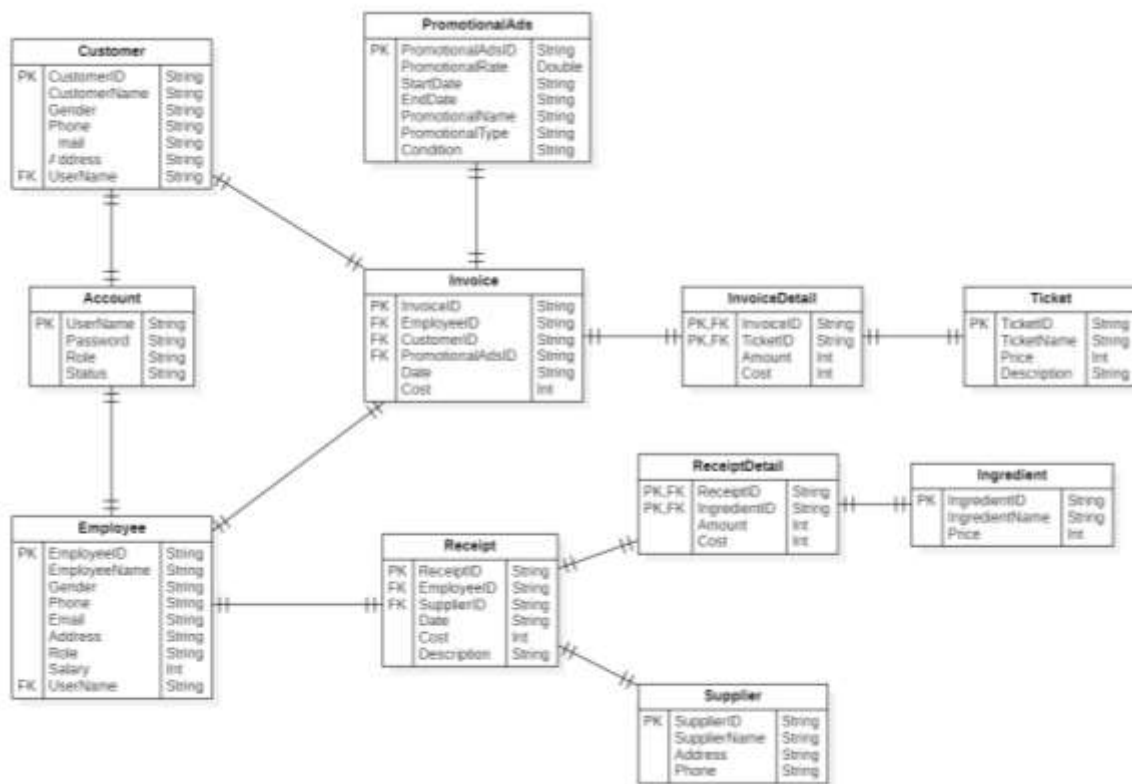
3.5.2 ingredient



3.5.3 customer account



3.6 Mapping to Relational Model diagram from Entity class diagram



4. IMPLEMENTATION AND TESTING


4.1 Introduction to the implementation environment

Operating system: Window 11

- Database: Oracle database
- Text Editor: NetBeans IDE 17

4.2 User Interface

4.2.1 Customer interface



Đăng nhập

Tên đăng nhập

Mật khẩu


☐ *Hiện mật khẩu*

Đăng nhập

Chưa có tài khoản ? **Đăng ký**

Thoát

Picture 1.Sign in page



Đăng ký

Tên đăng nhập

Mật khẩu

Nhập lại mật khẩu

☐ Hiện mật khẩu

Đăng ký

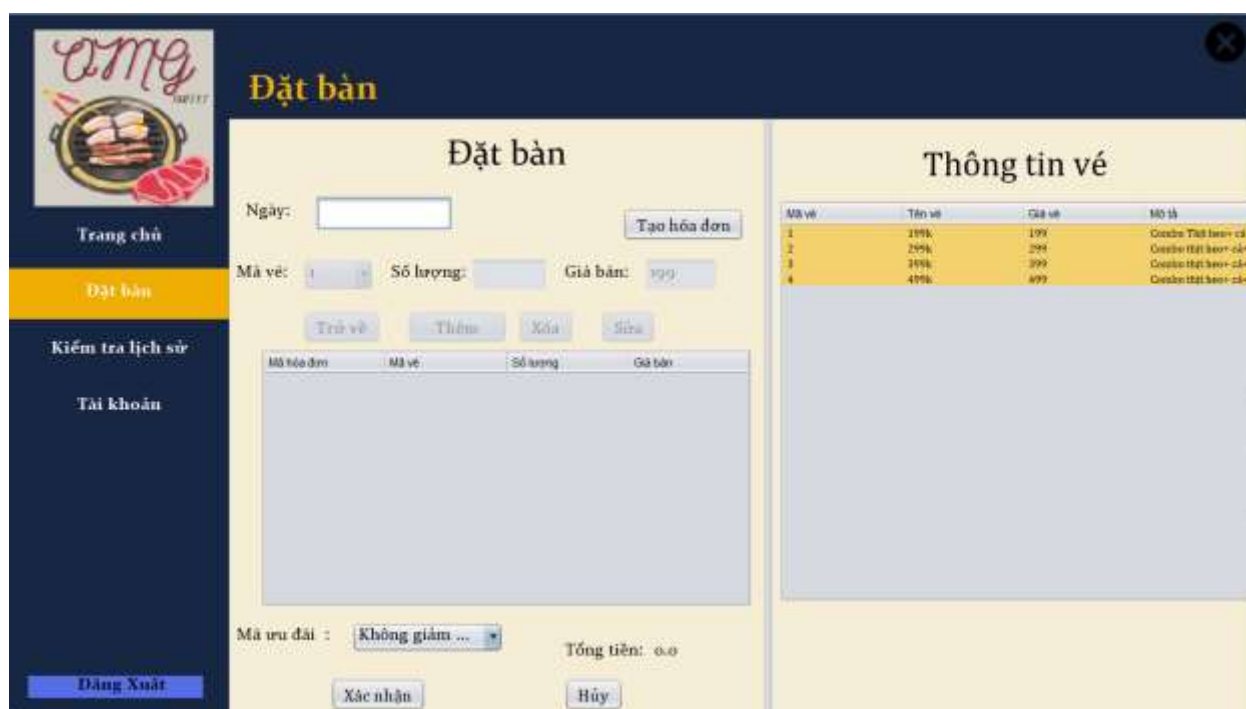
[Đã có tài khoản ?](#) [Đăng nhập](#)

[Thoát](#) [Trợ giúp ?](#)

Picture 2. Sign up page



Picture 3.Home page



Picture 4.Booking page



Picture 5.Transaction history page



Picture 6.Profile page

4.2.2 Admin interface



Picture 7. Admin home page



Picture 8. Customer management page



Quản lý nhân viên

Quản lý khách hàng

Quản lý nhân viên

Quản lý đặt bàn

Quản lý nhà cung cấp

Quản lý phiếu nhập

Quản lý ưu đãi

Thống kê doanh thu

Đăng Xuất

Mã nhân viên	Họ tên	Giới tính	Số điện thoại	Địa chỉ	Email	Chức vụ	Lương	Tên DN
0	admin	Nam	082321	HCM	phu@gmail.com	Điều bếp	0	admin
1	Phúc	Nam	0213212	HCM	ste@gmail.com	Quản lý	7000	admin
2	Đạt	Nam	0827424	HCM	Chung@gmail.c...	Thu ngân	4000	admin
3	Chung	Nữ	0284274	HCM	Vinh@gmail.com	Bồi bàn	4000	admin
4	Vinh	Nam	083217	HCM	long@gmail.com	Bồi bàn	4000	admin
5	Long	Nam						

Mã nhân viên
 Họ Tên
 Giới Tính
 Chức vụ

Số điện thoại
 Địa chỉ
 Email
 Lương

Picture 9. Employee management page



Quản lý đặt bàn

Quản lý khách hàng

Quản lý nhân viên

Quản lý đặt bàn

Quản lý nhà cung cấp

Quản lý phiếu nhập

Quản lý ưu đãi

Thống kê doanh thu

Đăng Xuất

Hóa đơn

Mã hóa đơn
 Mã nhân viên

Mã khách hàng

Mã ưu đãi
 Tỷ lệ ưu đãi: Không giảm giá

Ngày bán: 19/6/2023 14:11:3

Tổng tiền: 0

Mã HD	Mã NV	Mã KH	Mã UD	Ngày bán	Tổng tiền
1	0	0	0	30/06/2	1394
2	2	1	0	19/06/2	2890
3	0	4	1	19/06/2	647.5
4	4	5	0	19/06/2	18342
5	3	4	0	19/06/2	118441
6	0	8	0	20/06/2	199
7	0	8	1	21/06/2	299

Chi tiết hóa đơn

Mã hóa đơn

Mã vé

Số lượng:

Giá bán: 199

Mã hóa đơn	Mã vé	Số lượng	Giá bán
1	1	4	199
1	2	2	299
2	1	4	199
2	2	3	299
2	3	3	399
3	1	2	199
3	2	3	299
4	1	24	199
4	3	34	399
5	1	123	199
5	2	2	299
5	3	234	399
6	1	1	199
7	2	2	299

Vé

Mã vé

Tên vé

Giá vé

Mô tả

Mã vé	Tên vé	Giá vé	Mô tả
1	199k	199	Combo 1th
2	299k	299	Combo 2th
3	399k	399	Combo 3th
4	499k	499	Combo 4th

Picture 10. Ticket and invoice management page



Quản lý khách hàng

Quản lý nhân viên

Quản lý đặt bàn

Quản lý nhà cung cấp

Quản lý phiếu nhập

Quản lý ưu đãi

Thống kê doanh thu

Dăng Xuất

Quản lý nhà cung cấp

Mã nhà cung cấp	Tên nhà cung cấp	Địa chỉ	Số điện thoại
1	OPP	HCM	03934
2	Anh	HCM	0482134
3	Mega	HCM	0942134
4	Hoàng Đông	HCM	0394214
5	FamFood	HCM	027731

Tim kiếm

Mã nhà cung cấp

Tên nhà cung cấp

Số điện thoại

Địa chỉ

Trở về

Thêm

Xóa

Sửa

Picture 11. Supplier management page



Quản lý khách hàng

Quản lý nhân viên

Quản lý đặt bàn

Quản lý nhà cung cấp

Quản lý phiếu nhập

Quản lý ưu đãi

Thống kê doanh thu

Dăng Xuất

Quản lý phiếu nhập

Phiếu nhập

Mã phiếu nhập:

Mã nhân viên:

Mã nhà cung cấp:

Ngày nhập: 19/6/2022 14:13:58

Địa chỉ:

Tổng tiền: 0

Trở về

Thêm

Xóa

Sửa

Chi tiết phiếu nhập

Mã phiếu nhập:

Mã nguyên liệu:

Số lượng:

Giá nhập: 300

Trở về

Thêm

Xóa

Sửa

Nguyên liệu

Mã nguyên liệu:

Tên nguyên liệu:

Giá nguyên liệu:

Trở về

Thêm

Xóa

Sửa

Mã phiếu	Mã nhân	Mã nhà	Ngày nh.	Địa chỉ	Tổng tiền
1	1	1	19/06/20		2748
2	2	3	19/06/20		1847
3	3	5	19/06/20		8200
4	0	4	19/06/20		1829

Mã phiếu nhập	Mã nguyên liệu	Số lượng	Giá nhập
1	2	3	300
1	3	1	149
1	7	4	500
2	1	2	300
2	3	3	149
3	8	23	400
4	2	2	300
4	3	4	149
4	4	3	5
4	9	3	4

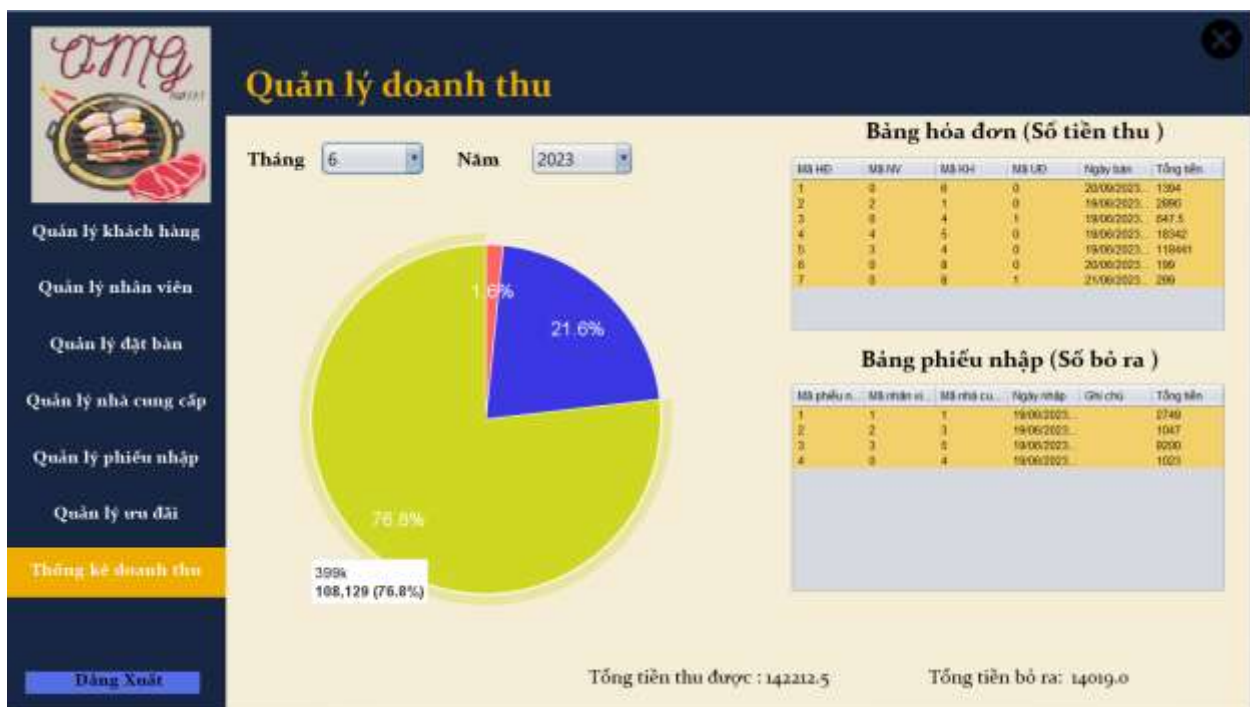
Mã nguyên liệu	Tên nguyên liệu	Giá nguyên liệu
1	Thịt bò	300
2	Thịt heo	200
3	Cà phê	149
4	Chè	5
5	Sữa	5
8	Tôm	400
7	Mực	500
8	Tôm lươn chạch	13
9	Rau muống	4

Picture 12. Receipt detail and Ingredient management page

Page 144 | 147



Picture 13. Promotional ads management page



Picture 14. Analysis revenue page

5. CONCLUSION

5.1 Project achievement

-During the implementation of the project, the team has accumulated a lot of knowledge related to the analysis and design of information systems in the direction of objects using UML. From that design, it can be implemented into a management software equivalent to design analysis.

-The team's final app also accomplished some of the basic functions originally outlined:

- Booking tickets
- Sign in/ Sign up for customer
- Sign in for administrator
- Manage profile
- Search
- Review transaction history
- Manage customers
- Manage employees
- Manage tickets
- Manage ingredients
- Manage suppliers
- Manage promotional ads
- Revenue analysis

5.2 Limitations

-Due to limited time and manpower, the group has the following limitations:

- Software not yet compatible across a wide range of devices and operating systems.
- Failure to identify and nurture potential customers.
- The software does not yet have multilingual interface support.
- There is no data recovery and backup function yet.

5.3 Development direction

- The completed application meets the functional requirements of the system, but due to the limited time to complete the project, the software also suffers from certain limitations. In the future, the team offers some development directions for the current software as follows:

- Manage customer support/repair requests.
- Compatible across a wide range of devices and operating systems.
- Identify and nurture the list of potential customers.

- Multilingual interface support.
- Build recovery and backup function.