**ASSIGNMENT**

Topic : Restaurant Management System

Team 2:

Bùi Việt Thắng

Tống Thị Ngọc Anh

Phạm Duy Khánh

Hà Khánh Duy

# Introduction

Restaurant business is a very growing service business in Vietnam and around the world. Nowadays, the economic life is developing day by day, people tend to higher both material and spiritual values. It is not simply a relationship between buyers - sellers, customers looking for food and drink services, but also for enjoyment and enjoyment. So how to make customers come to you in the most convenient way, easily leave the best impression, the core issue that is indispensable for all types of businesses or organizations is to give a rational and unified management model. The parts that work smoothly and properly are the premise for an efficient and professional operation. This restaurant management system we build is suitable for medium and medium sized restaurants with the most complete and simple functions.

\* \* \*

# Part I: System analysis

1. **Describe the problem**

A restaurant serving food, business form served at the table. Reception Department will receive CUSTOMER information, check whether new or old customers and serve the Menu, customers choose dishes with unlimited number of times and quantities. Staff will make an Order (request form) based on the request of the customer. Customer's order will be transferred to Cashier department.

The kitchen depends on the customer's Order (received from the cashier) to quantify the necessary food processing for customers.

After the customer ends up using the service, the Invoice will be recorded at the Cashier's department, all of the customer's order will be added to the bill. Order slip is considered as detailed invoice of payment. A customer can have multiple invoices, but one invoice is in the name of only one customer.

The restaurant will have sections to manage: Kitchen, accountant, cashier, receptionist. Each department will manage its staff. After each week, every month and after a year, the accounting department will synthesize and report the customer situation, import food, inventory,… to the board of directors.

# Part II: System requirements

## General requirements

* The external actors in the relationship with the system are customers and management. The customer requests the service to the restaurant for the menu, response and payment.
* The restaurant needs to build up information about customers, issue invoices, set up orders to serve dishes and print payment invoices when customers run out of service requirements.
* Prepare periodic reports as required by management.

**Business system**

**Customer**

**Leadership**

Information Customer

Request a service

Payment request

Menu

Order, Bill

Information request

Command

Report

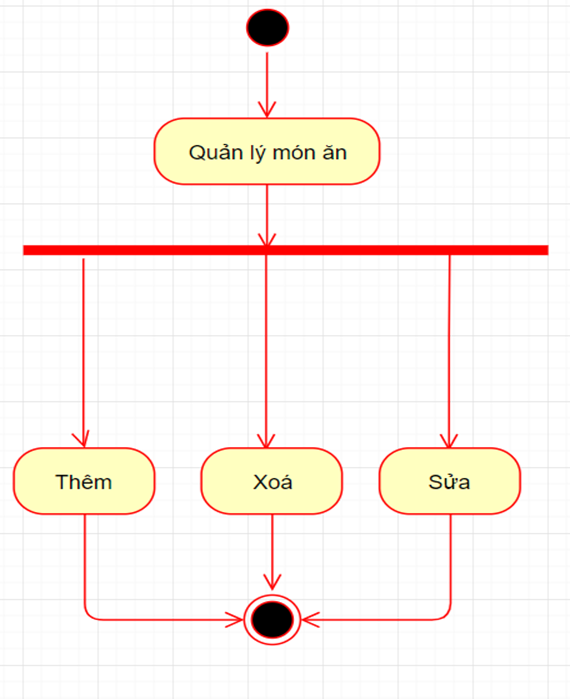
## 2.2 Business Rules

|  |  |
| --- | --- |
| ID | Description |
| BR01 | Account's email address must be valid. |
| BR02 | Account's password must be from 8-16 characters in length and must contain at least 1 uppercase letter, lower case letter and digit. |
| BR03 | When registering, or changing password, user must enter the new password  twice. |
| BR04 | A guest must provide their email address and password when registering an  account. |
| BR05 | A guest cannot register with an email that has already been registered. |
| BR06 | User must provide their account's email address and password when logging into the website. |
| BR07 | An account must belong to one of the three roles: User, Admin, Staff. |
| BR08 | Only Admin or Staff accounts can login to the CMS system. |
| BR09 | Only Administrator accounts can manage User, Bill, Function in the system. |
| BR10 | Food or drink type must be chosen. |
| BR11 | Amount of Food or Drink must be chosen. |
| BR12 | Price must be a positive number. |
| BR13 | User must be received confirmation to go to the payment page. |
| BR14 | User only can be set the menu when completed payment. |
| BR15 | The search section will be on top of the screen as part of the one-page layout of the web version. |
| BR16 | Each Facebook account can only be used to register to the service once. |
| BR17 | Purchased money cannot be refunded. |
| BR18 | The searching process will only happen after the user hits Enter. |
| BR19 | There cannot have two Food or Drink with the same name. |
| BR20 | When login with Facebook, this account is auto set in User role. |
| BR21 | Each email address can be used for only one account |
| BR22 | There cannot have two visitor types with the same CMND-CCCD. |
| BR23 | Must have at least one Food or Drink in the system. |
| BR24 | No need to fill in all the information fields when searching |
| BR25 | There cannot have two categories with the same name. |
| BR26 | All information of category must be entered. |
| BR27 | Invoicing when guests start using the service. |
| BR28 | The dishes you have selected are recorded on the order form to request the kitchen to order. |
| BR29 | Detailed reports with specific queries and print for leaders. |
| BR30 | Customers pay according to the valid amount in the invoice. |

**USE CASE**

**Food Management**

|  |  |  |
| --- | --- | --- |
| **ID** | **Actor** | **Name** |
| FM-1 | Admin | Create new food in menu |
| FM-2 | Admin, Staff | Edit food in menu |
| FM-3 | Admin, Staff | Remove food in menu |
| FM-4 | Admin, Staff | Edit the title and describe |
| FM-45 | Admin | Edit the price of food |
| FM-5 | Admin, Staff, User | View all food in menu |
| FM-6 | Admin, Staff, User | Search food in menu |
| FM-7 | Admin, Staff, User | View customer’s order in cart |
| FM-8 | User | Edit customer’s order in cart |

****

• Summary: Use case allows the manager to add, delete, and modify dishes.

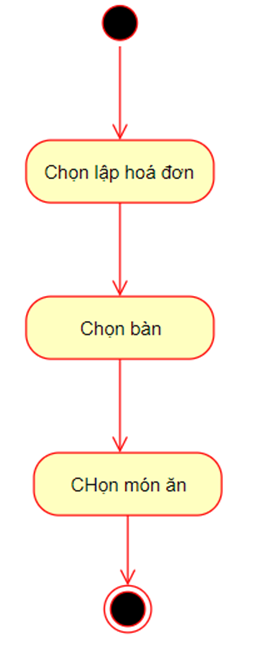
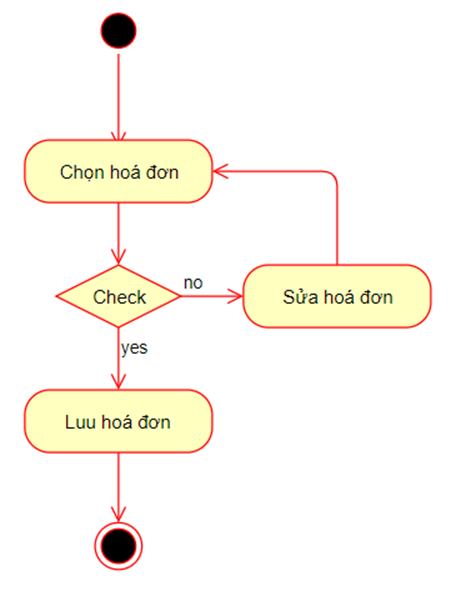
• Event stream: Manager logs into the system with Management rights and proceeds:

* Add dishes: Manage enter full information and click add dishes.
* Delete food: Manage to select the item to delete and click delete.
* Edit: Manage select and edit dishes, modify dish information according to needs and press update.

• Special requirements: No.

**Bill Management**

|  |  |  |
| --- | --- | --- |
| **ID** | **Actor** | **Name** |
| FM-1 | Admin, Staff | Create bill for customer |
| FM-2 | Admin, Staff | Edit bill |
| FM-3 | Admin, Staff | Remove bill if information in bill error |
| FM-4 | Admin, Staff | Edit the title and describe |
| FM-6 | Customer | Check information in bill |
| FM-7 | Admin, Staff | Put the completed bill in the statistics table |
| FM-8 | Admin, Staff, User | Search bill information |
| FM-9 | Admin, Staff | Edit the statistics table of bill |
| FM-10 | Admin | Check all of bills in months |

** **

Use case for invoicing:

• Summary:

* Use-case description of Cashier Staff (NVTN) invoicing, recording the dishes customers order with the waitress (NVPV)

• Event flow:

* Use case starts when NVPV brings customer order slip to the cashier counter. Staff enter the dishes that customers order into the payment slip on app\_website.

Use case to save invoices:

• Summary:

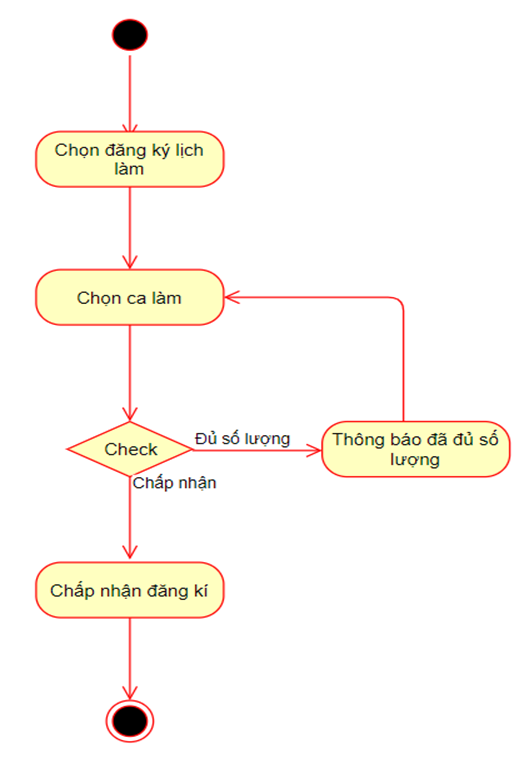
* The use case describes how NVTN prints payment invoices and stores them in the database of the restaurant.

• Event stream:

* Main event line: when the customer requests payment, the sales agent will print the existing invoice and bring the NVPV to the sitting table. After payment is completed, NVTN records the invoice in the database.
* Secondary event stream: if the customer receives the invoice, there is an error check, NVPV will check again. If wrong, then NVTN will correct the invoice and send the customer to pay.

**Schedule Management**

|  |  |  |
| --- | --- | --- |
| **ID** | **Actor** | **Name** |
| FM-1 | Admin, Staff | Create schedule for staff |
| FM-2 | Admin | Edit schedule of staff |
| FM-3 | Admin, Staff | Check schedule |
| FM-4 | Admin | Edit the title and describe |
| FM-5 | Customer | Check information in scheduler |
| FM-6 | Staff | apply for leave of absence from work |
| FM-7 | Admin | Statistics work schedule after 1 month |

****

Summary: usecase describes how employees go to the web to register work

+ Main event:

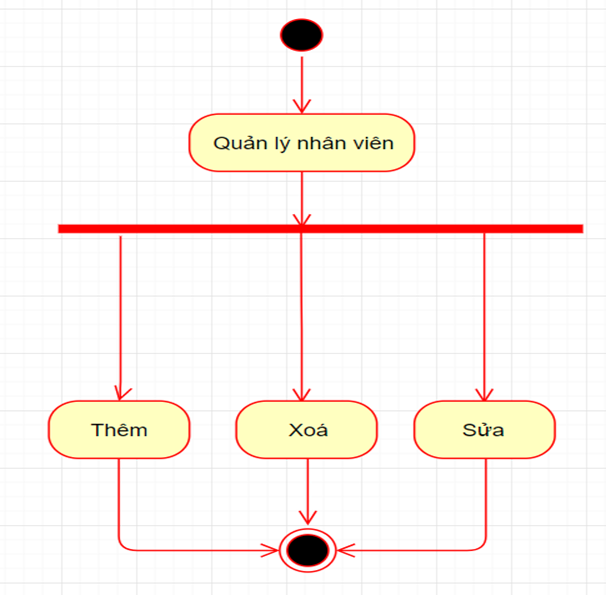
* This usecase starts when the employee chooses to register for the schedule.
* The system offers options including:
* Position.
* The work shifts for staff to choose.
* The system checks selected information and saves it in the database
* Manager login to the system with Management rights. Consider day-to-day shifts. If there is a shift without staff, the manager will contact other staff to supplement the minimum number of staff needed to operate normally.

+ Other events:

If in the main event, the employee chooses the same schedule, the system will refuse to receive that schedule and ask the employee to choose again. Show each shift, each employee on duty that shift (kitchen / service).

**Staff Management**

|  |  |  |
| --- | --- | --- |
| **ID** | **Actor** | **Name** |
| FM-1 | Admin | Add staff |
| FM-2 | Admin | Remove staff |
| FM-3 | Admin, Staff | Add and edit information of staff |
| FM-4 | Admin | Edit the title and describe |
| FM-5 | Customer, staff, admin | Check information |
| FM-6 | Admin | Change salary of staff |
| FM-7 | Admin | division of work for staff |

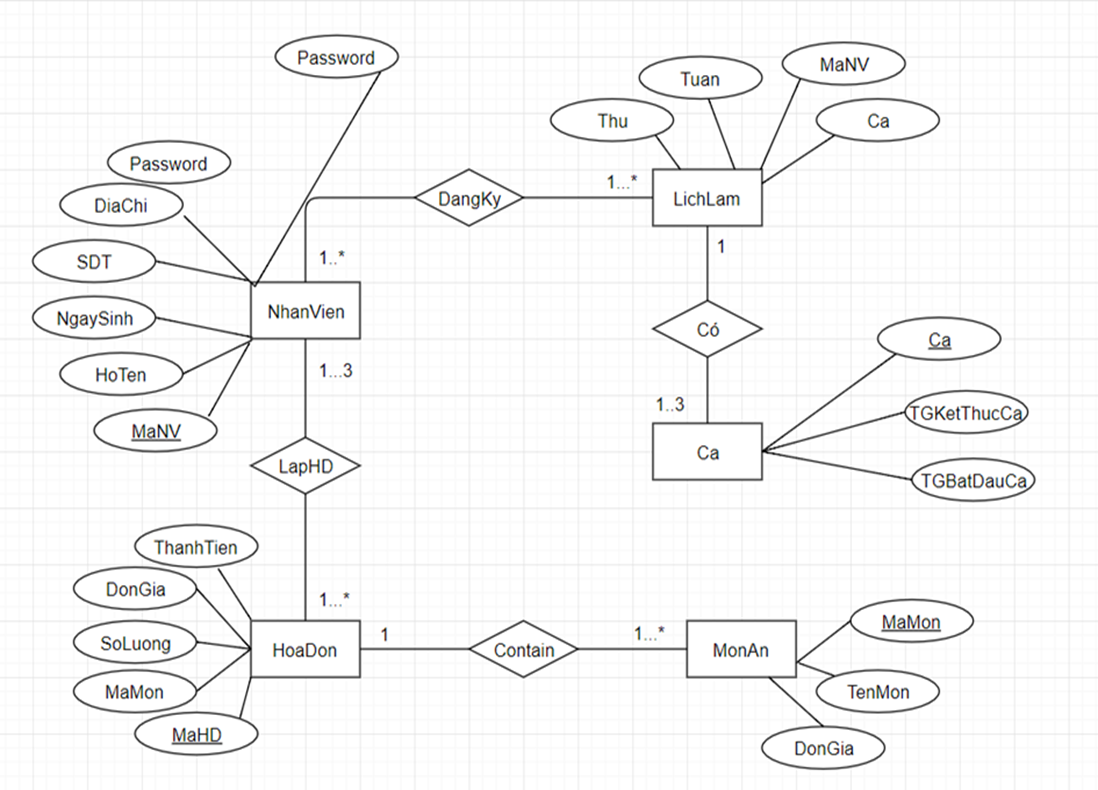
****

• Summary: Use case allows managers to add, delete, and edit employees.

• Event stream: Manager logs into the system with Management rights and proceeds:

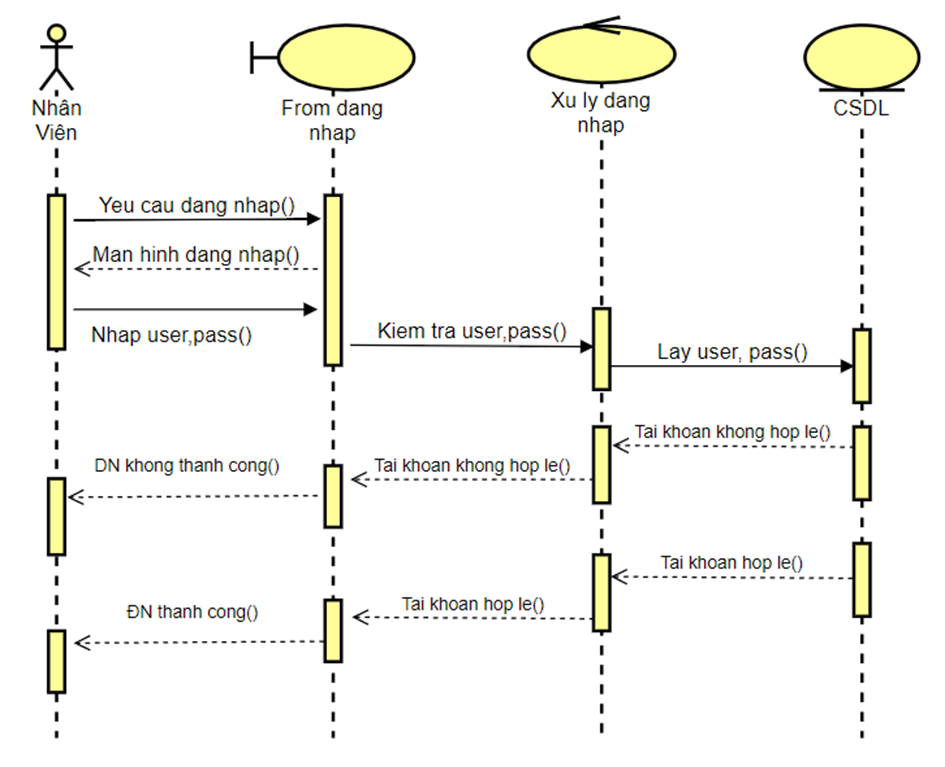
* Add employee: Manage enter full employee information and click add employee.
* Delete employee: The manager selects the employee to delete and press delete.
* Edit: Manage, choose to edit staff, modify employee information according to needs and press update.

• Special requirements: No.

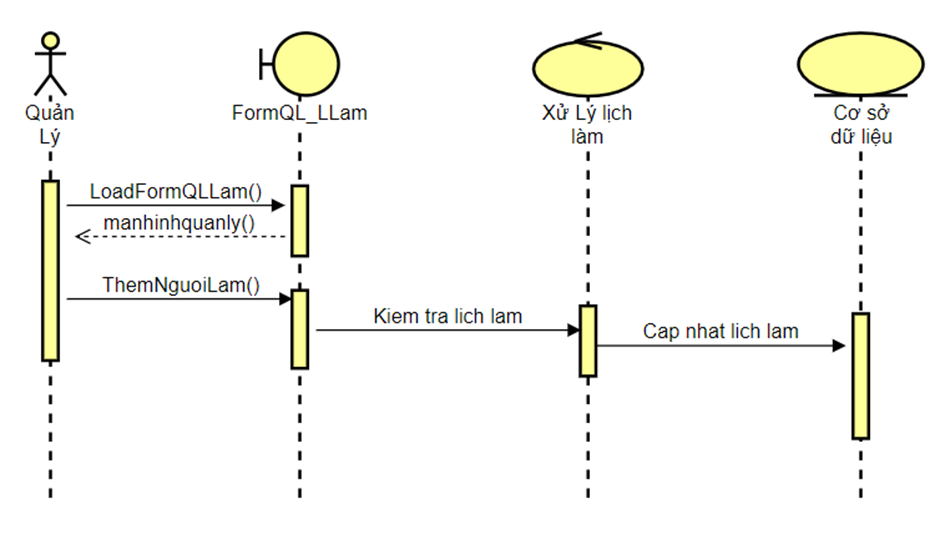


**Class Sequences**

1. Login

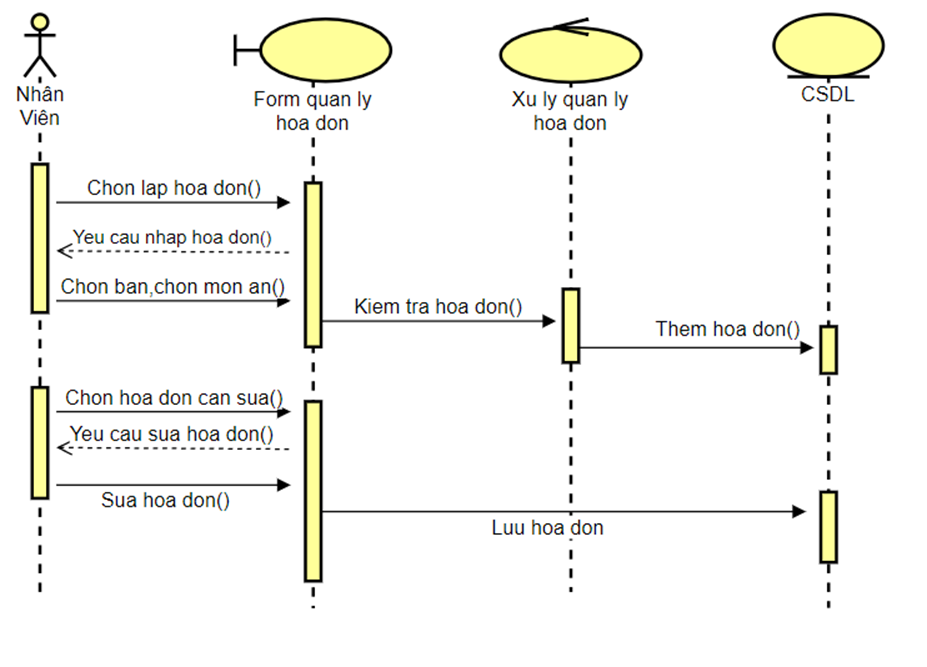


1. Schedule Management



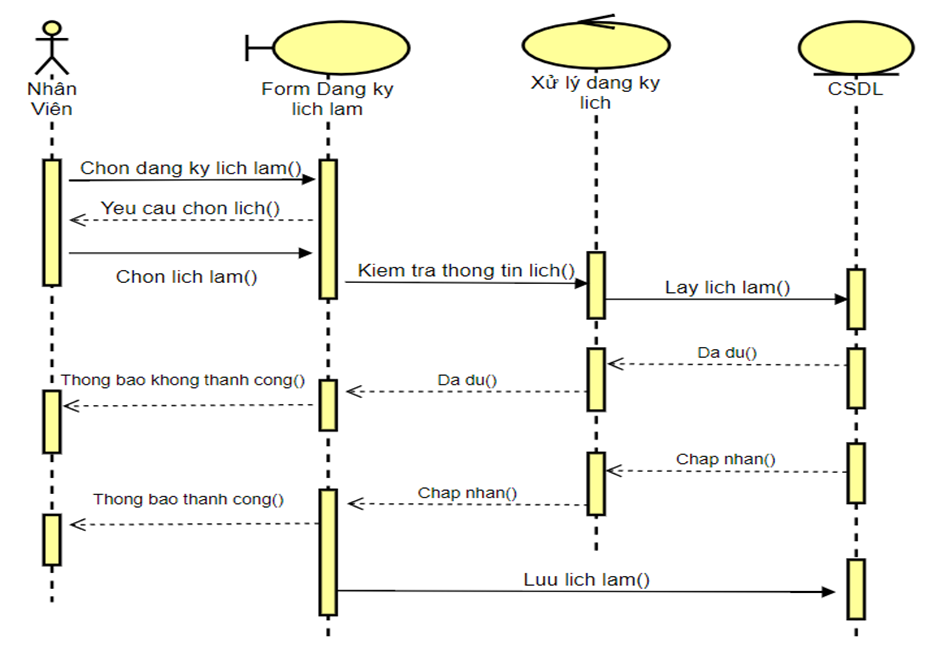
Hiển thị lại lên màn hình quản lý

1. Bill Management



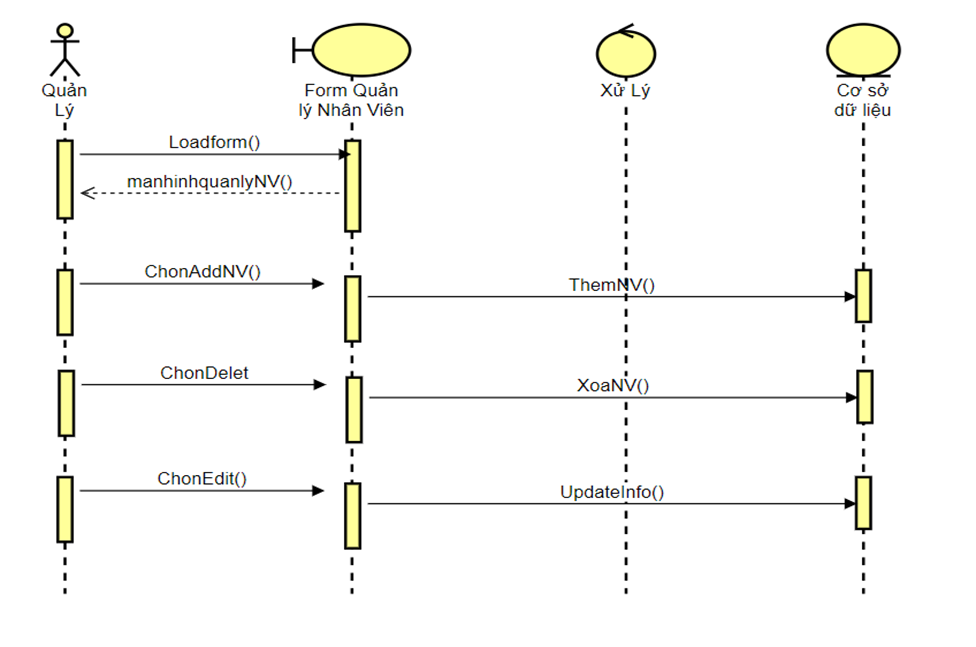
Hiển thị lại lên màn hình quản lý

1. Registration Schedule



Cập nhật lại lên màn hình quản lý

1. Staff Management

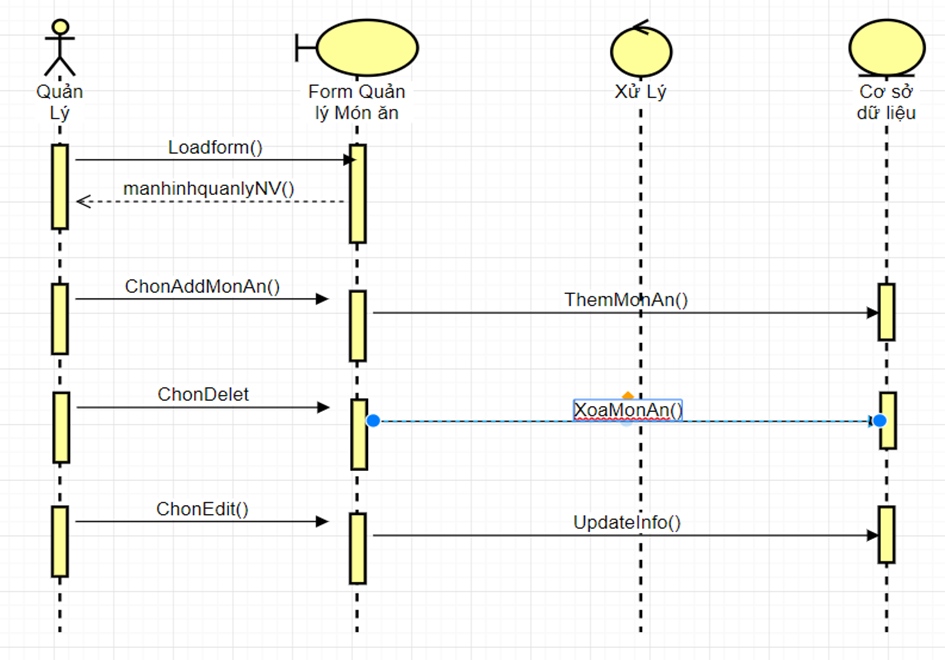


Hiển thị lại lên màn hình quản lý

Hiển thị lại lên màn hình quản lý

Hiển thị lại lên màn hình quản lý

1. Food Management

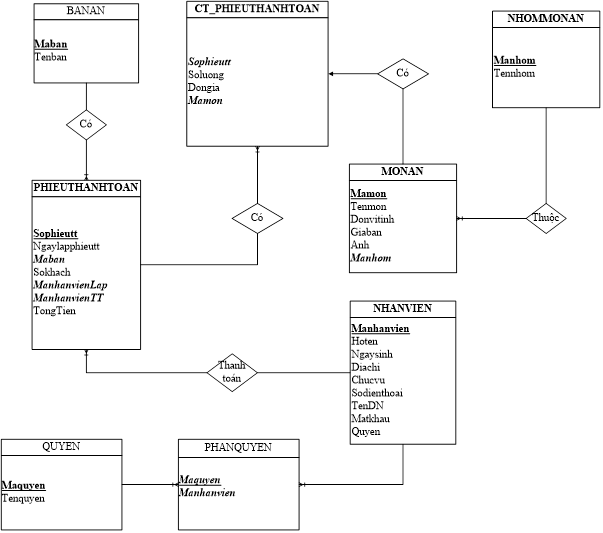


Hiển thị lại lên màn hình quản lý

Hiển thị lại lên màn hình quản lý

Hiển thị lại lên màn hình quản lý

**Class Diagram**



\* \* \*

**--FINISHED--**