# 



**<JAVA – Canteen MS>**

**Documentation**

|  |  |
| --- | --- |
| **Authors** | **C2108L-Team 3** |
| **Ho Gia Huy**  **Duong Van Hoang**  **Pham Hai Hung** |
| **Date** | **14-9-2022** |
| **Instructor** | **Tran Van Diep** |

Hanoi, 27 – 5 – 2022**Index**

[1.](#_heading=h.30j0zll) Problem Definition 3

[1.1.](#_heading=h.1fob9te) Problem Abstraction 3

[1.2.](#_heading=h.3znysh7) The Current System 3

[1.3. The Proposed System 4](#_heading=h.2et92p0)

[1.3.1. Boundaries of the System 5](#_heading=h.tyjcwt)

[1.3.2.](#_heading=h.3dy6vkm) Hardware and Software Requirements 5

[2.](#_heading=h.1t3h5sf) Customer Requirements Specification 6

[2.1.](#_heading=h.4d34og8) Users of the System 6

[2.2.](#_heading=h.2s8eyo1) System functions 7

[3.](#_heading=h.17dp8vu) System Designs 8

[3.1.](#_heading=h.3rdcrjn) Entity Relationship Diagram 8

[3.2.](#_heading=h.26in1rg) Database Design 9

[3.3.](#_heading=h.lnxbz9) Applicatiom diagram 12

[3.3.1.](#_heading=h.35nkun2) Main page 12

[3.3.2.](#_heading=h.1ksv4uv) The report page 14

[3.3.3.](#_heading=h.44sinio) The information about us page 16

[3.3.4.](#_heading=h.2jxsxqh) The register and login page 17

[3.3.5.](#_heading=h.z337ya) Admin page 19

[3.4.](#_heading=h.3j2qqm3) System fucntions design 21

[3.4.1.](#_heading=h.1y810tw) Register function 21

[3.4.2.](#_heading=h.4i7ojhp) Login function 22

[3.4.3.](#_heading=h.2xcytpi) Sending report function 23

[3.4.4.](#_heading=h.1ci93xb) Show table function 24

[3.4.5.](#_heading=h.3whwml4) Delete database function 25

[4.](#_heading=h.2bn6wsx) Task sheet 26

[5.](#_heading=h.qsh70q) Validation Checklists 28

# Problem Definition

## Problem Abstraction

People in ancient times managed everything by copying and recording it in books. Although copying, recording and preserving books so that they do not get lost or torn is a problem that in ancient times was difficult to deal with.

We here review an application that will provide brief details about the products and the categories of each product. The app is supposed to provide a user-friendly and navigational environment. Menus with each item are clear and color combinations are expected.

## The Current System

Management 4.0 allows users to manage work methodically, modernly, intuitively, easily control and update information quickly.

Easily track work progress remotely, feedback and record through the management software. Data and documents are updated directly to the common system, decentralized and shared for different individuals. Statistics on the work and results of each member of the company easily, helping to reward the right people for the right job. Record all information data, easy access, avoid getting lost and difficult searching.

## The Proposed System

This application has a number of features such as registration, login, view detailed information of each product, order management along with employee management to help canteen managers easily manage several databases. data such as adding, editing, deleting employees; product and sales statistics. The app has a clear category menu so users can easily find what they want.

## Boundaries of the System

The goal in this app is to help this canteen activity. The application serves 3 types of objects: Managers, Service staff with activities such as:

+ Manage products sold in the canteen

+ Create and manage sales invoices

+ Employee management

+ Statistics on products and sales.

## Hardware and Software Requirements

In this application we use Java, JavaFX, CSS to create the user interface. Java has made the application stable and user-friendly.

It can run with minimum configuration so that you don’t need a strong computer to access it.

# Customer Requirements Specification

## Users of the System

Users can only use functions such as registering, logging in, viewing customer, buying and selling product, and show customer and their income, order, revenue to the system.

1. Employers.

* Log in and log out.
* Manage employee.
* Manage customer.
* Choose customer to sell product.
* Choose product and quantity to sell.
* Show cart and ask customer to confirm.
* Confirm order.
* Show order in the past.
* Show revenue of all orders and calculate total revenue.

1. Employees.

* Log in and log out.
* Manage customer.
* Choose customer to sell product.
* Choose product and quantity to sell.
* Show cart and ask customer to confirm
* Confirm order.

1. Customers

* Buy product from customer.
* See product at menu and ask employees to choose product and quantity.
* Confirm order and buy product.

## System functions

The first function we want to mention is the subscription function. Regarding the registration function, users will need to fill in some information to have an account. Information about this user will be sent to the database and admin can easily manage this user.

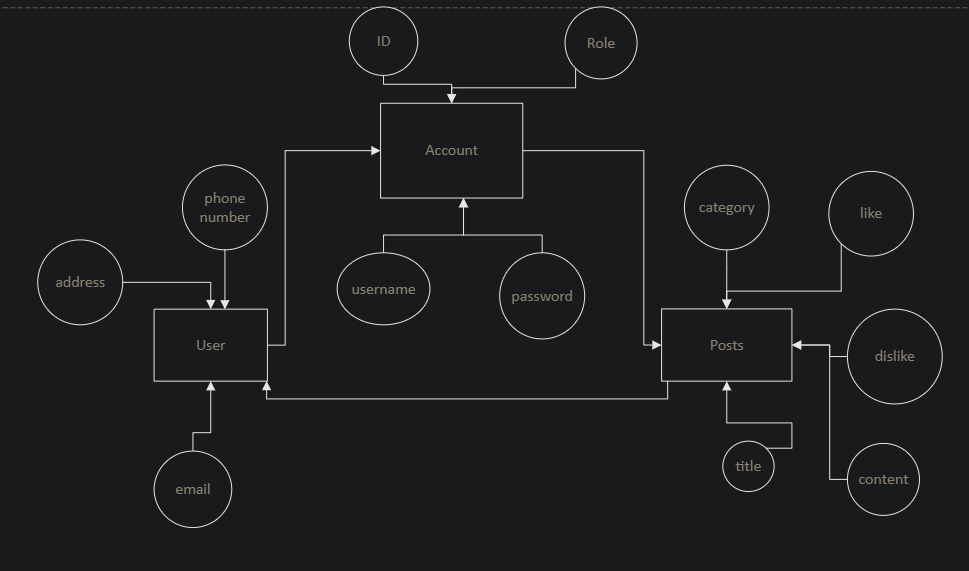
The next function that we want to mention is the login function. When the user fills in the email or phone number and password, if it matches and the account is in the database, the user will login successfully.

The third function is to display employees, customers who want to order and products sold in the canteen, allowing users to view employee information as well as salary and working time about each person of the canteen manager and the canteen's products being sold. Store staff can view their salary, order products when customers order, see what products need to be done to serve customers, show all order in the past and revenue of order to calculate total revenue.

And the last function is to add, edit, delete employees, customers, products, and edit income. When the manager presses the add, edit or delete button to add, edit or remove the employee, customer, product, and only edit income. The application sends information to the database and this information in the database is deleted. And then we can't see this information even in the database.

# System Designs

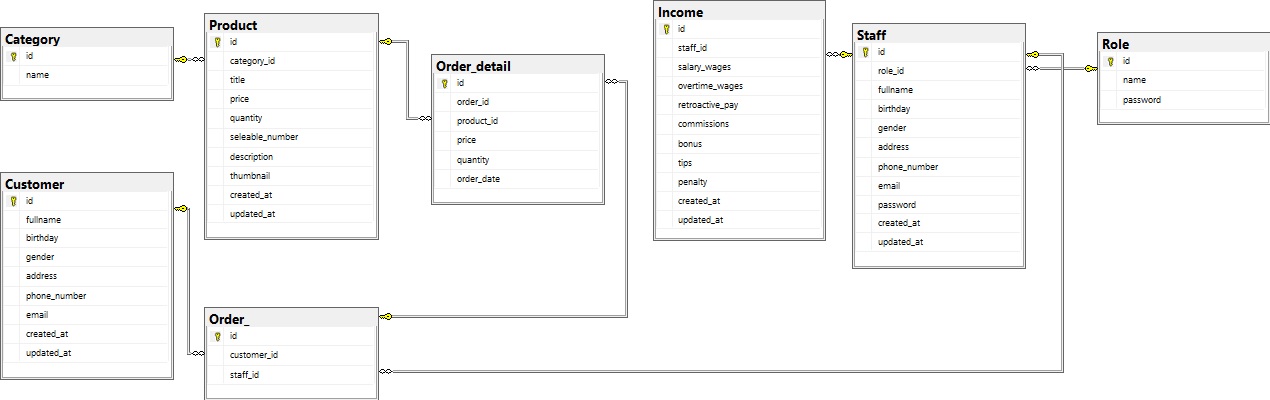
## Entity Relationship Diagram



Picture 3.1: Entity Relationship diagram

User affect by some element like address, phone number, email address. And user will affect to account. Account is affected by id, role, username, password. And account can affort to a post. A post has some element like category, number of like, number of dislike, title and content. And many posts can affort to users.

## Database Design



Picture 3.2: The database design

In this project we have eight tables of database:

* The first table in here is table Staff. In this table we will have information about the staff. In here we have id of staff, id of role, username, staff’s email address, staff’s password, staff’s phone number, and we can know when this account created with staff’s created time and when this account updated with staff’s updated time. The primary key in this table is the Id (of staff). The foreign key is role\_id. In this table we used type of data as table 3.1:

Table 3.1: Type of data in table Staff

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Datatype | Constraint | Description |
| Id | INT |  | Auto\_increment |
| Role\_id | INT |  |  |
| Fullname | Varchar(50) |  |  |
| Birthday | Date |  |  |
| Gender | varchar(16) |  |  |
| Address | Varchar(50) |  |  |
| Phone\_number | INT | UNIQUE, NOT NULL |  |
| Email | INT | UNIQUE, NOT NULL |  |
| Password | Varchar(100) |  |  |
| Created\_at | Datetime |  |  |
| Updated\_at | Datetime |  |  |

* The second table in here is table role. In that table we will have only three information. We have role’s id, role name and role password. The role table will lead us permission as a user or an admin. The primary key in this table is role’s id. In this table we used type of data as table 3.2:

Table 3.2: Type of data in table Role

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Datatype | Constraint | Description |
| Id | INT |  | Auto\_increment |
| Name | Varchar(30) |  |  |
| Password | Varchar(100) |  |  |

* The third table is income table. The income table will receive information user send and show it to the admin. In the report table we have 6 information. We have income’s id, staff’id , salary of staff, overtime wages of staff, retroactive pay of staff, commissions wages of staff, bonus and tips of staff, and penalty of staff, and we can know when this income is created with income’s created time and when this income is updated with income updated time. The primary key is Id(of income). The foreign key is Id(of staff). In this table we used type of data as table 3.3:

Table 3.3: Type of data in table Income

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Datatype | Constraint | Description |
| Id | INT |  | Auto\_increment |
| Staff\_id | INT |  |  |
| Salary\_wages | FLOAT |  |  |
| Overtime\_wages | FLOAT |  |  |
| Retroactive\_pay | FLOAT |  |  |
| Commissions | FLOAT |  |  |
| Bonus | FLOAT |  |  |
| Tips | FLOAT |  |  |
| Penalty | FLOAT |  |  |
| Created\_at | Datetime |  |  |
| Updated\_at | Datetime |  |  |

* The fourth table is category table. The category table has category’s id, category’s name, and we can know when this category created with category’s created time and when this category updated with category’s updated time. The primary key is category’s id. In this table we used type of data as table 3.4:

Table 3.4: Type of data in table Category

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Datatype | Constraint | Description |
| Id | INT |  | Auto\_increment |
| Name | Varchar(15) |  |  |
| Created\_at | Datetime |  |  |
| Updated\_at | Datetime |  |  |

* The fifth table is customer table. The customer table has customer’s id, customer’s fullname, customer’s birthday, customer’s gender, customer’s address, customer’s phone number, customer’s email. The primary key is customer’s id and we can know when this customer created with customer’s created time and when this customer updated with customer’s updated time.In this table we used type of data as table 3.5:

Table 3.5: Type of data in table Customer

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Datatype | Constraint | Description |
| Id | INT |  | Auto\_increment |
| Fullname | Varchar(15) |  |  |
| Birthday | Date |  |  |
| Gender | Varchar(16) |  |  |
| Address | Varchar(50) |  |  |
| Phone\_number | INT |  |  |
| Email | Varchar(30) |  |  |
| Created\_at | Datetime |  |  |
| Updated\_at | Datetime |  |  |

* The next table is product table. The product table has category of product, product’s title, quantity of product, number of product sold, description of product, thumbnail of product, and we can know when this product created with product created time and when this product updated with product’s updated time. The primary key is Id (of Product). The foreign key is category\_id. In this table we used type of data as table 3.6:

Table 3.6: Type of data in table Product

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Datatype | Constraint | Description |
| Id | INT |  | Auto\_increment |
| Category\_id | INT |  |  |
| Title | FLOAT |  |  |
| Quantity | INT |  |  |
| Seleable\_number | INT |  |  |
| Description | Varchar(100) |  |  |
| Thumbnail | Varchar(500) |  |  |
| Created\_at | Datetime |  |  |
| Updated\_at | Datetime |  |  |

* The next table is sale\_off table. The sale\_off table has product\_id, quantity of product, price of product, date\_start and date\_end. The foreign key is product\_id. In this table we used type of data as table 3.7:

Table 3.7: Type of data in table Sale\_off

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Datatype | Constraint | Description |
| Product\_id | INT |  |  |
| Quantity | INT |  |  |
| Price | FLOAT |  |  |
| Date\_start | Datetime |  |  |
| Date\_end | Datetime |  |  |

* The next table is comment table. The comment table has customer\_id, product\_id, rade, evaluate. The primary key is comment’id and we can know when this comment created with comment’s created time and when this comment updated with comment’s updated time. In this table we used type of data as table 3.8:

Table 3.8: Type of data in table Comment

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Datatype | Constraint | Description |
| Id | INT |  | Auto\_increment |
| Customer\_id | INT |  |  |
| Product\_id | INT |  |  |
| Rate | INT |  |  |
| Evaluate | Varchar(100) |  |  |
| Created\_at | Datetime |  |  |
| Updated\_at | Datetime |  |  |

* The next table is order table. The order table has customer\_id, staff\_id. The primary key is order’id and we can know when this order created with order’s created time and when this order updated with order’s updated time. In this table we used type of data as table 3.9:

Table 3.9: Type of data in table Order\_

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Datatype | Constraint | Description |
| Id | INT |  | Auto\_increment |
| Customer\_id | INT |  |  |
| Staff\_id | INT |  |  |
| Created\_at | Datetime |  |  |
| Updated\_at | Datetime |  |  |

* The next table is order\_detail table. The product table has order\_id, product\_id, quantity of product, price of produc, order\_date. The primary key is order\_detail’id and we can know when this order\_detail created with order\_detail’s created time and when this order\_detail updated with order\_detail’s updated time. In this table we used type of data as table 3.10:

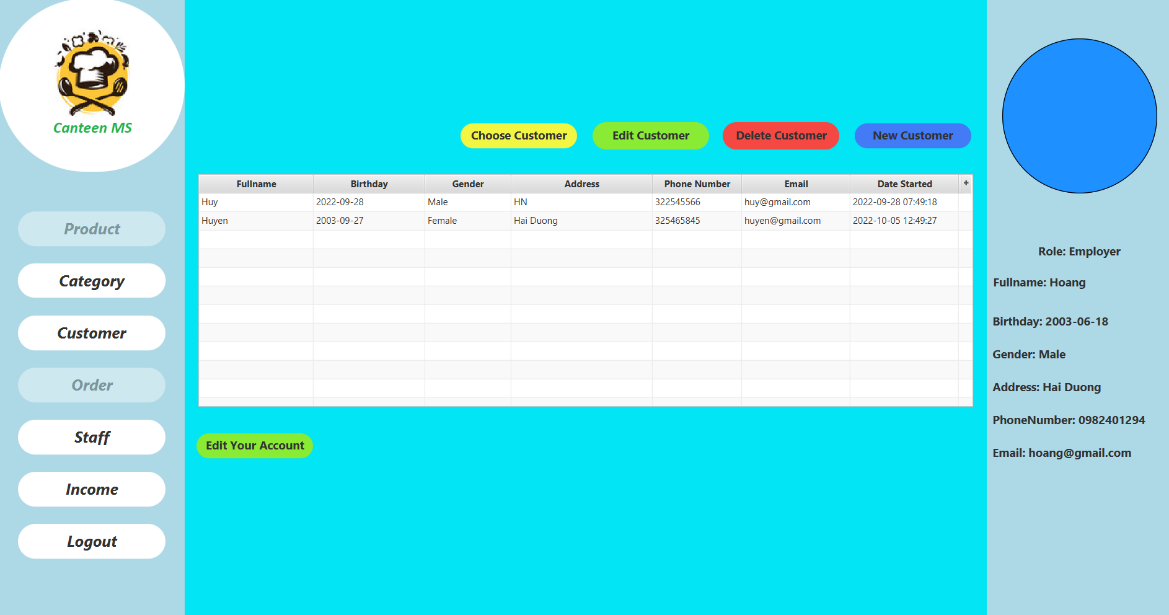
Table 3.10: Type of data in table Order\_detail

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Datatype | Constraint | Description |
| Id | INT |  | Auto\_increment |
| Order\_id | INT |  |  |
| Product\_id | INT |  |  |
| Quantity | INT |  |  |
| Price | FLOAT |  |  |
| Order\_date | Datetime |  |  |
| Created\_at | Datetime |  |  |
| Updated\_at | Datetime |  |  |

## Application diagram

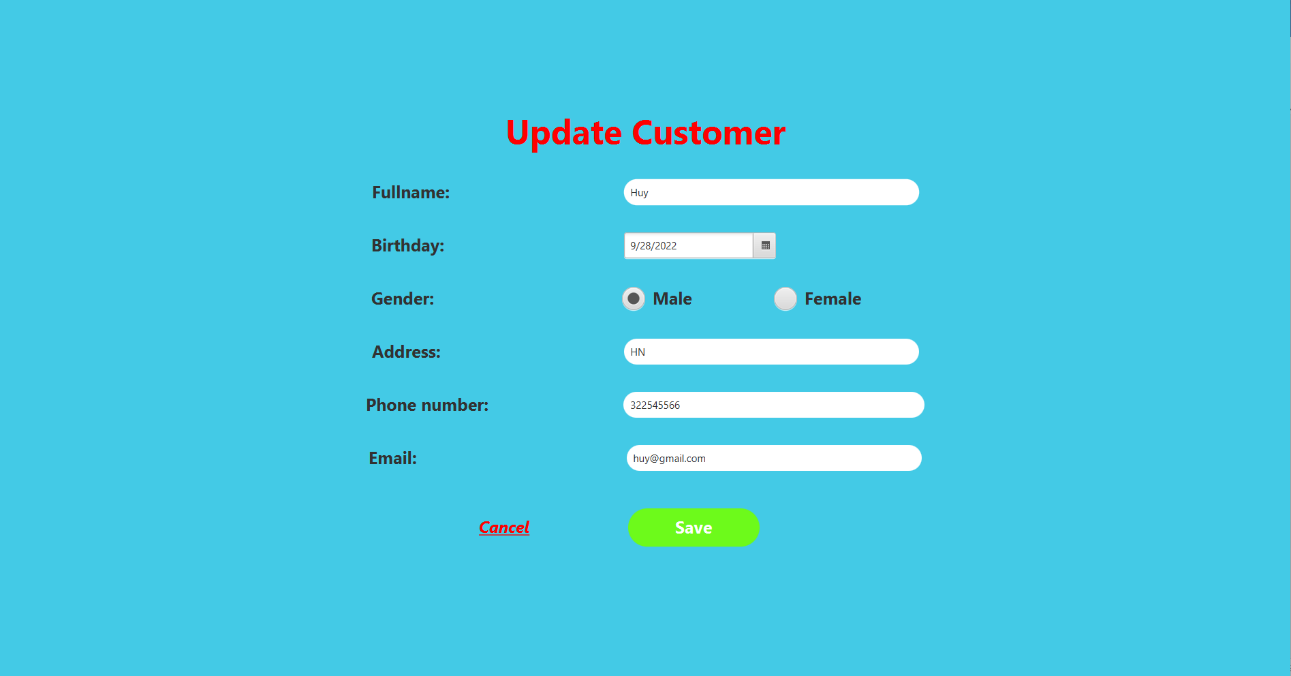
## Managecustomer page

After logging in your account information, you will go to the managecustomer page. Here we leave two interfaces between the employer and the employee. The first is the interface when logging in with an employer account:

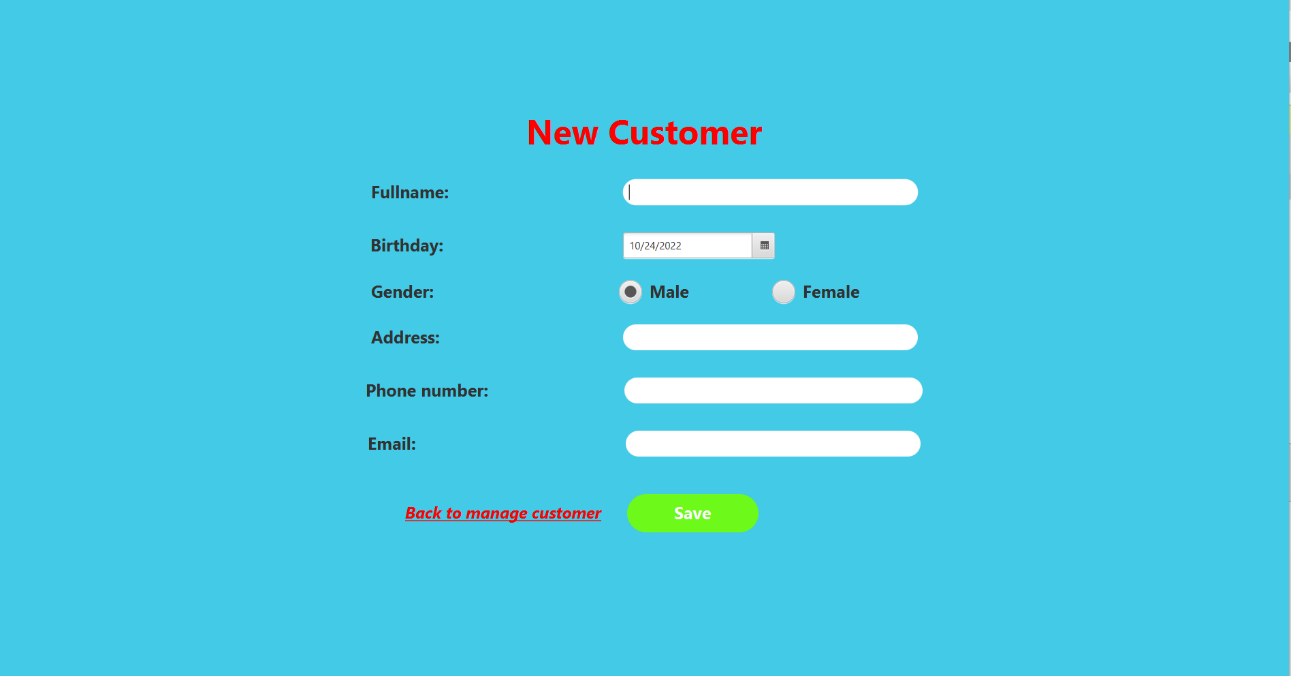


Picture 3.3: Managecustomer with employer account

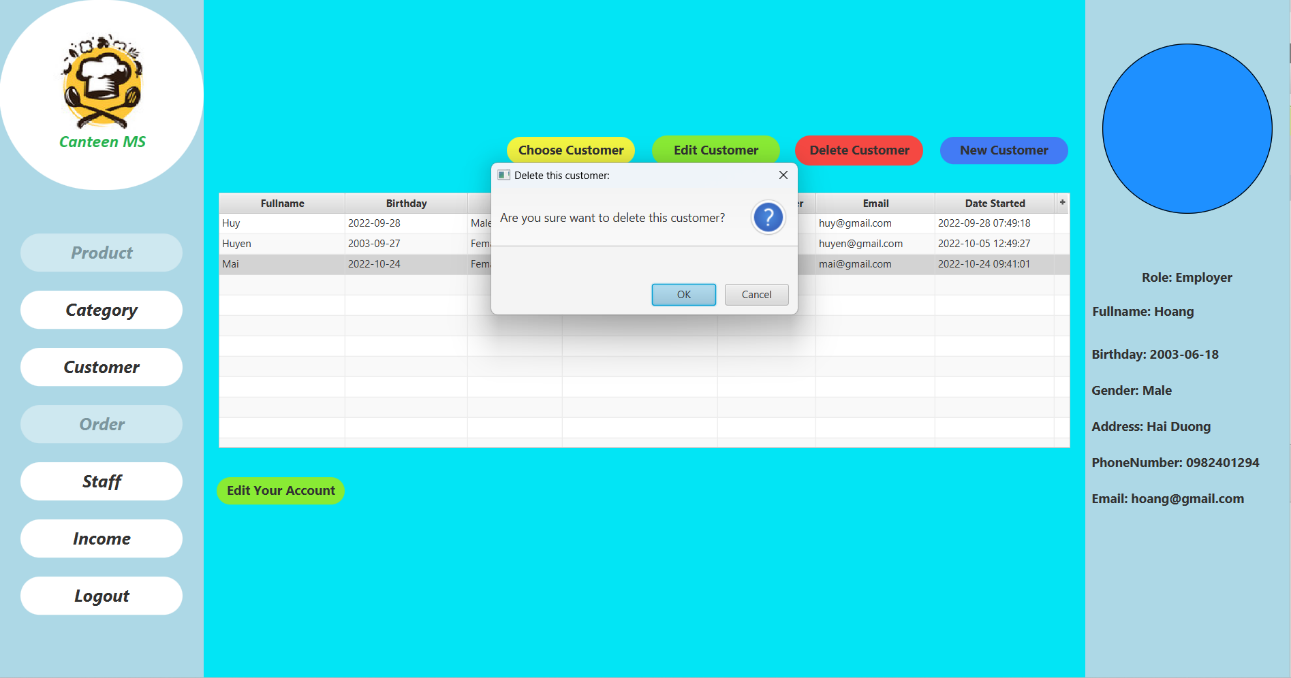
* With this interface, there will be 2 buttons for products and orders that are hidden when customers have not been selected. The reason we do this is so that the staff can control who orders and what orders.
* Next, there are buttons to choose customers, add, edit, delete customers and edit your own account.



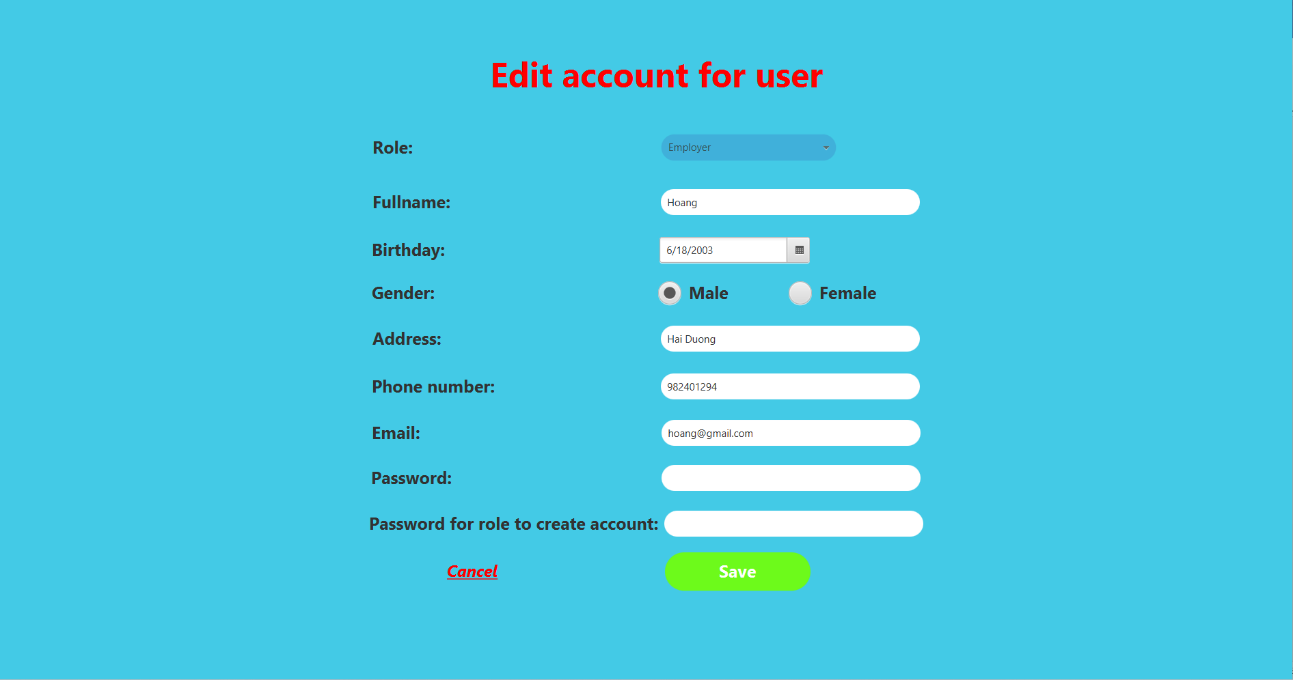
Picture 3.4: When you click edit customer



Picture 3.5: When you click new customer



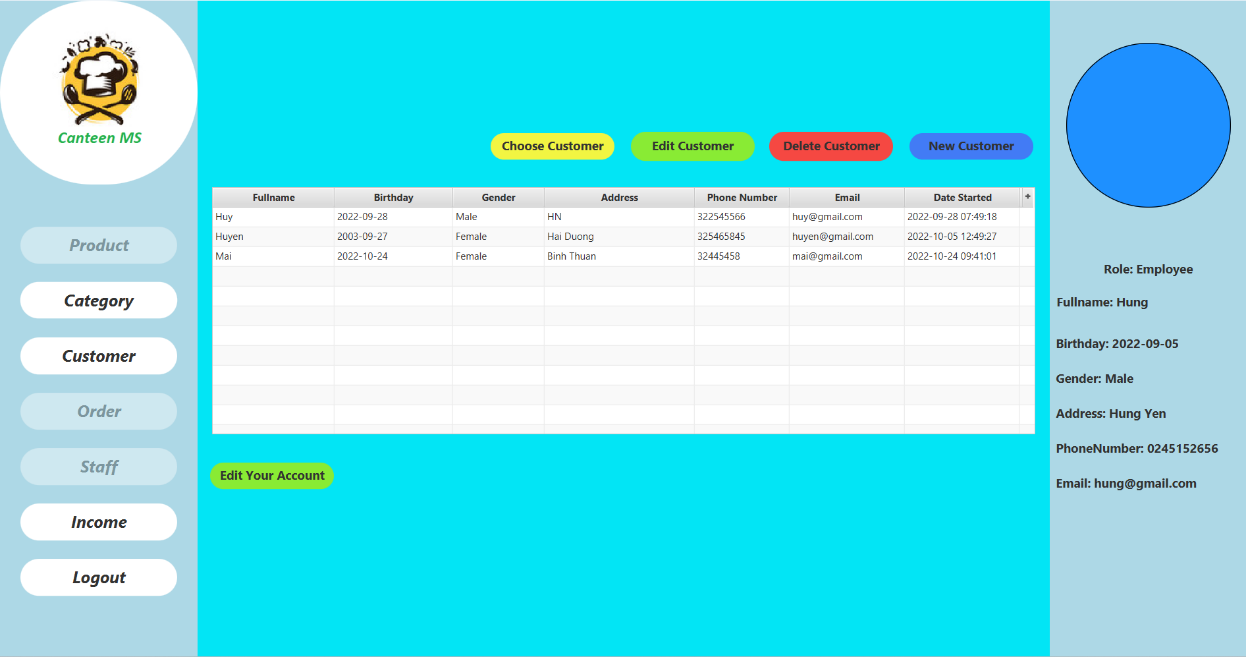
Picture 3.6: When you click delete customer after choose customer



Picture 3.7: When you click edit account

* Besides, the menu on the left side of the interface, there are buttons to view category, staff, income and log out. The right side displays the account's information including role, fullname, birthday, gender, address, email and phonenumber so that users can see their information.

The second is the interface when logging in with an employee account:



Picture 3.8: Managecustomer with employee account

The interface when logging in with an employee account has a similar function to the employer account, but the difference is that the staff button will be hidden.

## The Product and Order page

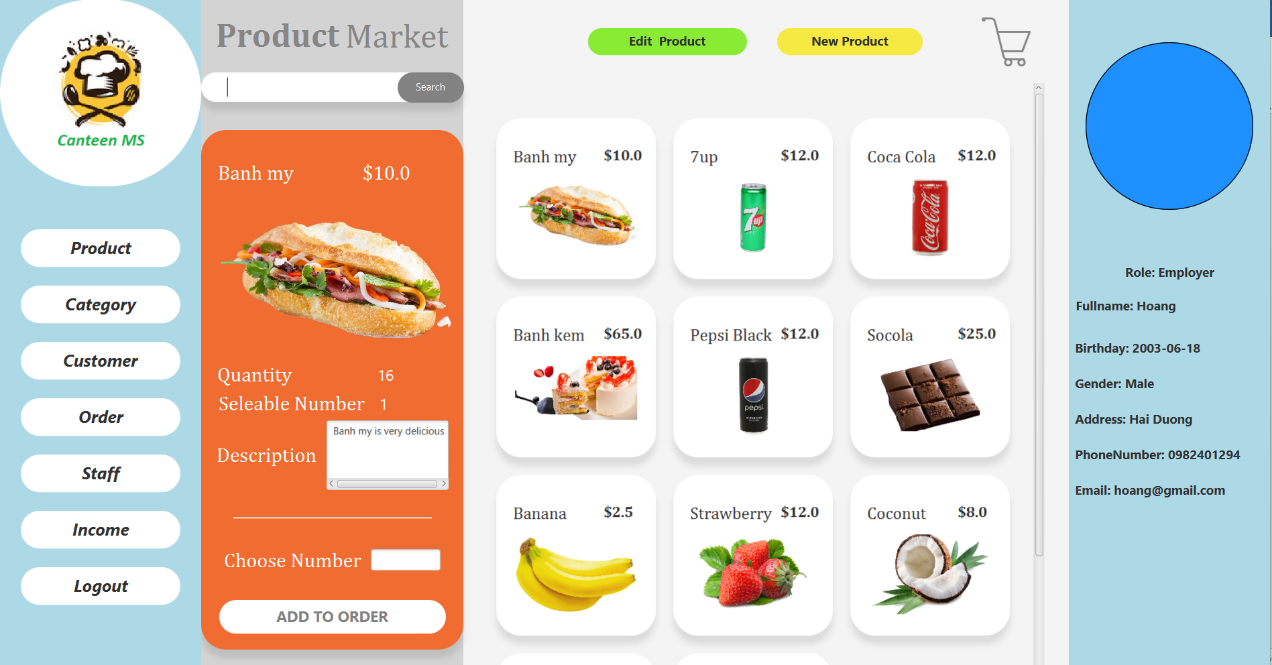


Picture 3.9: How to get to the Product page

To get to the Product page we need to select the customer in the list and press the choose customer button if the customer is not in the list, we need to create a new one and then follow the steps above.

After successfully selecting customers, we have the product page interface as follows:

Just like with the managecustomer page, the product page also has 2 interfaces with two login accounts between employees and customers.

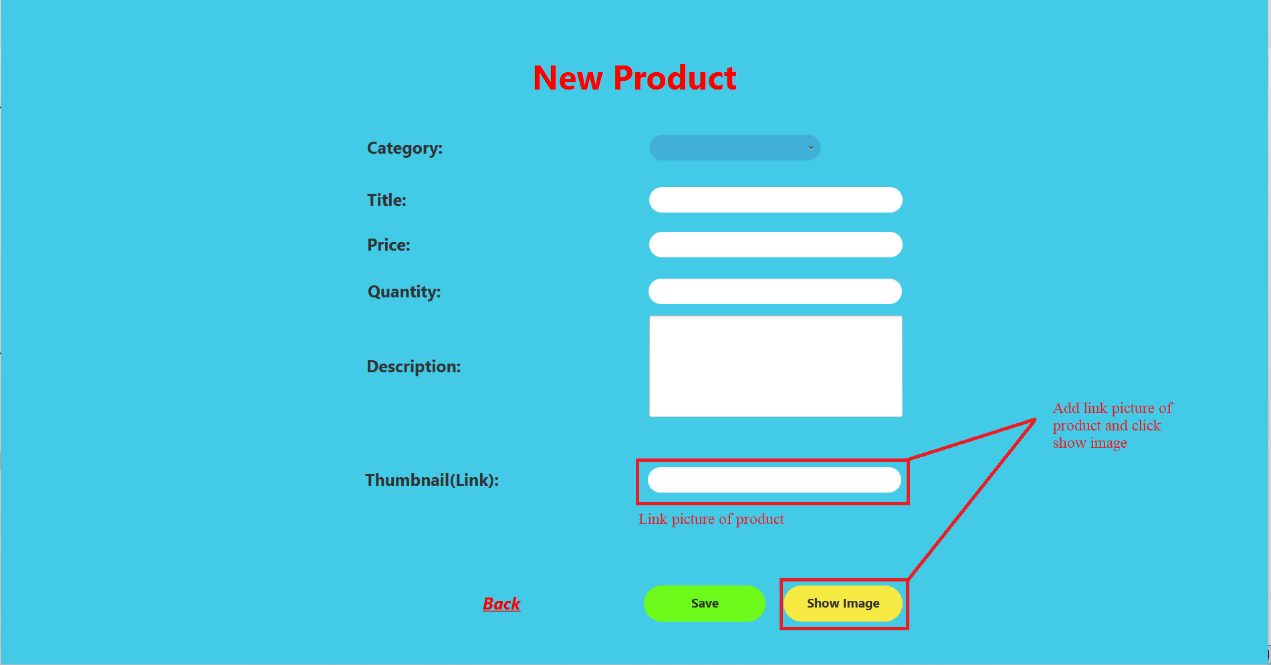


Picture 3.10: Product with employer account

In this interface, when the customer has been selected, the menu on the left has displayed two buttons Product and Order.

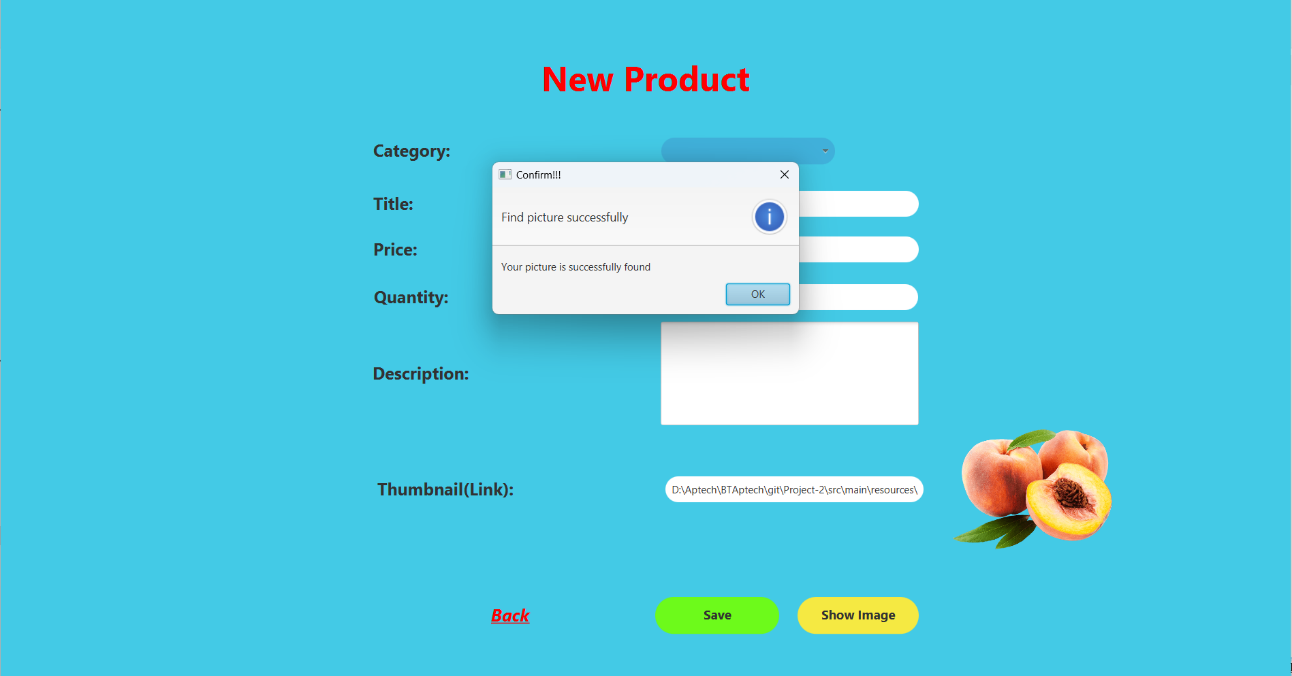
In the interface, there is a detailed product information section about title, price, quantity, seleable number, thumbnail is picture, description and choose number. The rest is a product umbrella list with more minimalist information but designed to be eye-catching and accessible. Every time you select the product box, you will be shown full product information in the details section.

There is also a button to add and edit products. When you click on the two buttons above, the interface will appear as follows:

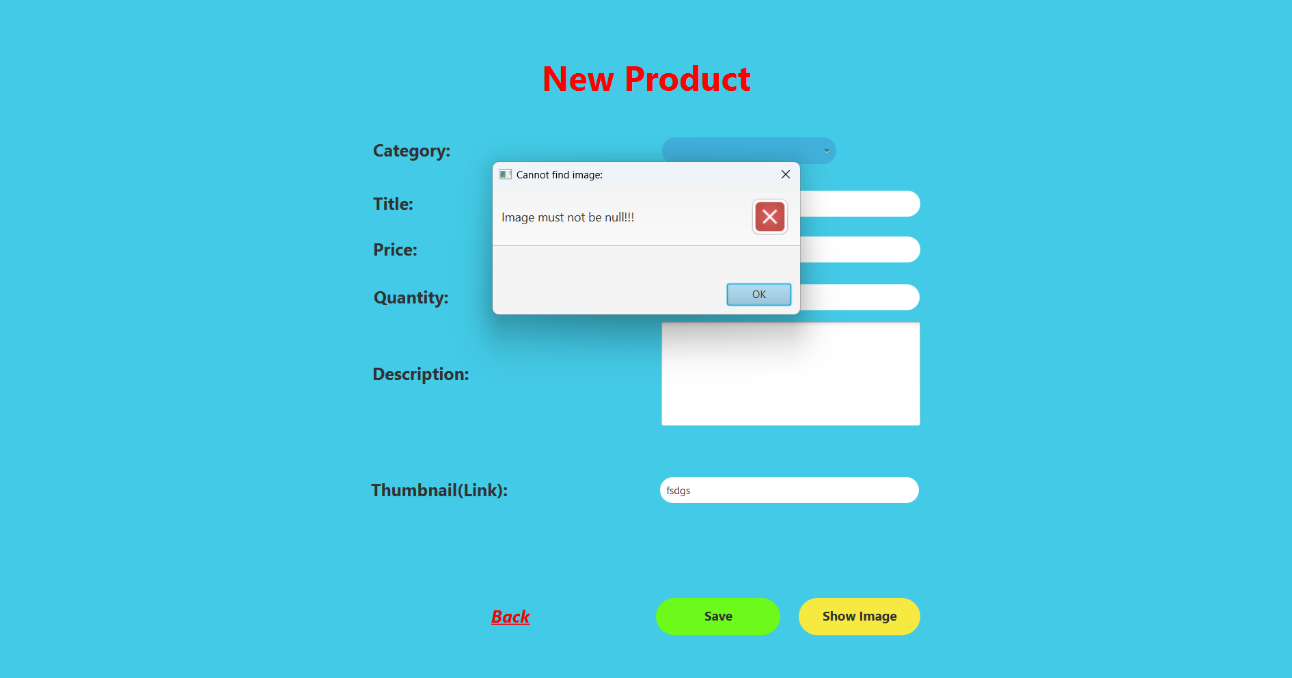


Picture 3.11: When you click new product

When you click the show image button, if the image link exists, an image will appear next to the link; if it doesn't exist, a message will show you the error path.

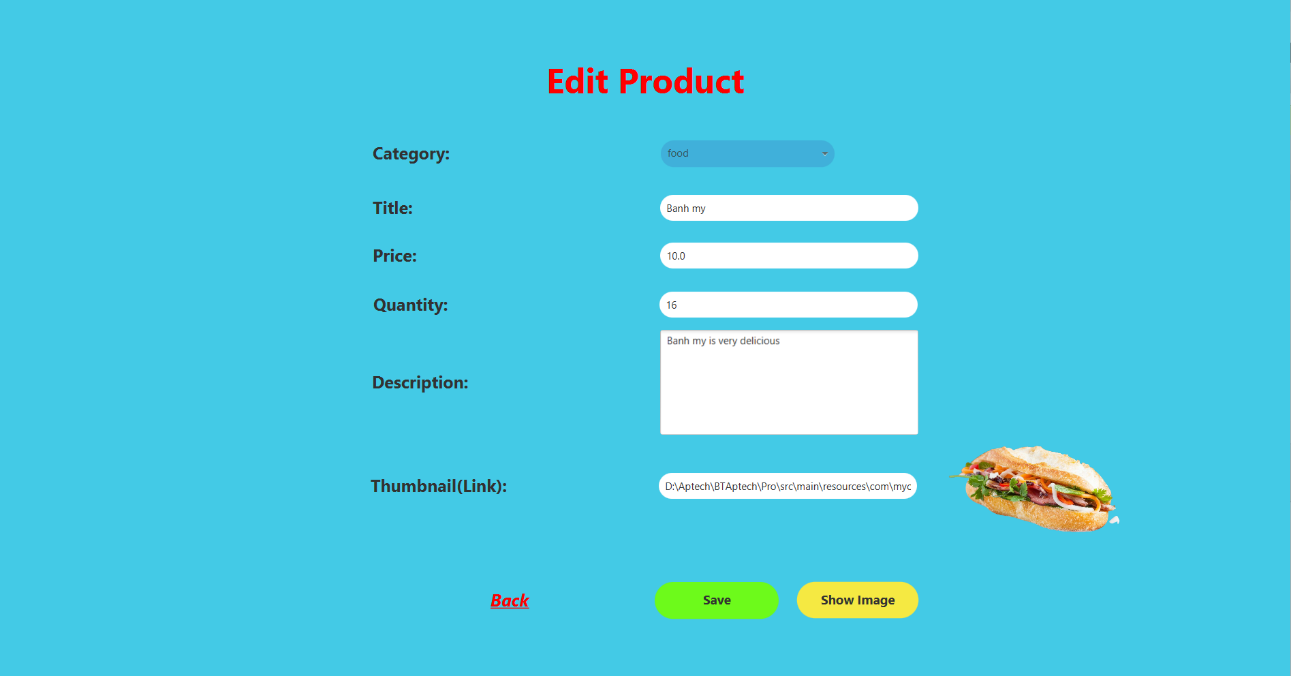


Picture 3.12: When you add link successfully



Picture 3.13: When you add link fail

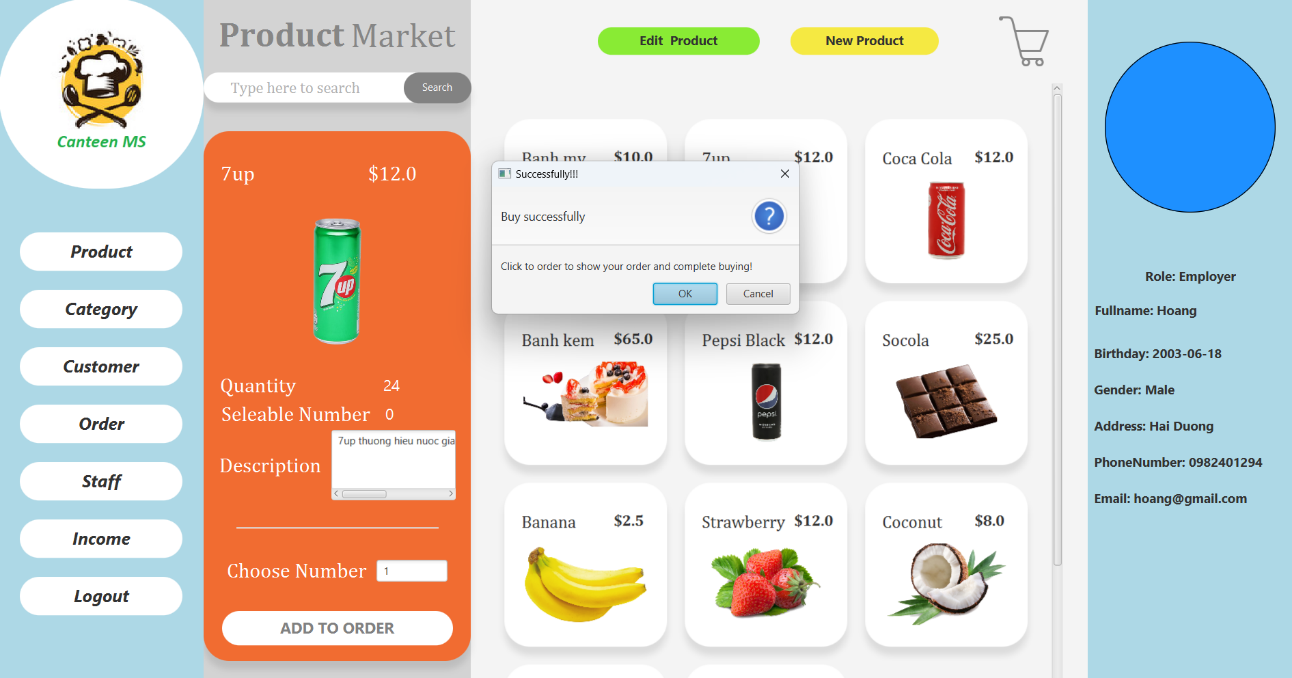
Next when pressing the edit button:



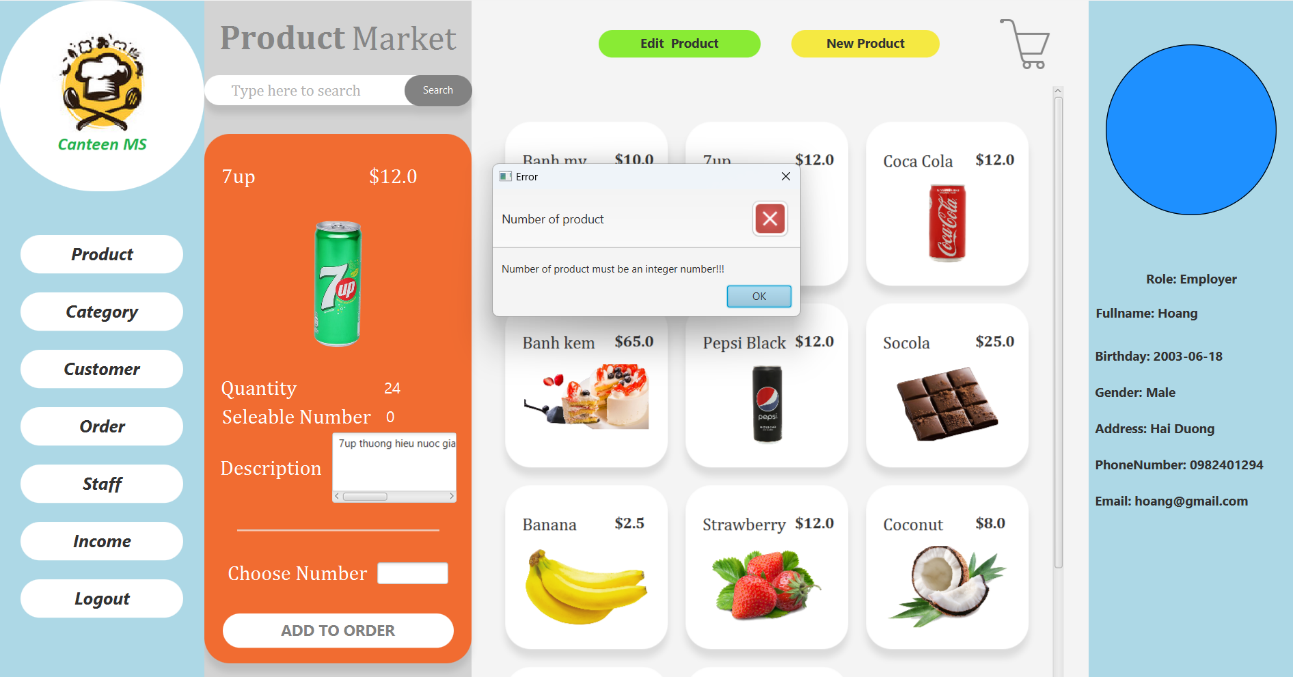
Picture 3.14: When you click edit product

To edit the product we need to select the product box in the list and click the edit product button the edit page will appear as above and the thumbnail we do the same as when adding the product.

When you order a product, you need to enter the quantity you want to buy of that product, if the quantity is correct in the difference between quantity and seleable number, then when you press the add to order button, it will be successful, otherwise there will be an error message and you need to re-enter.

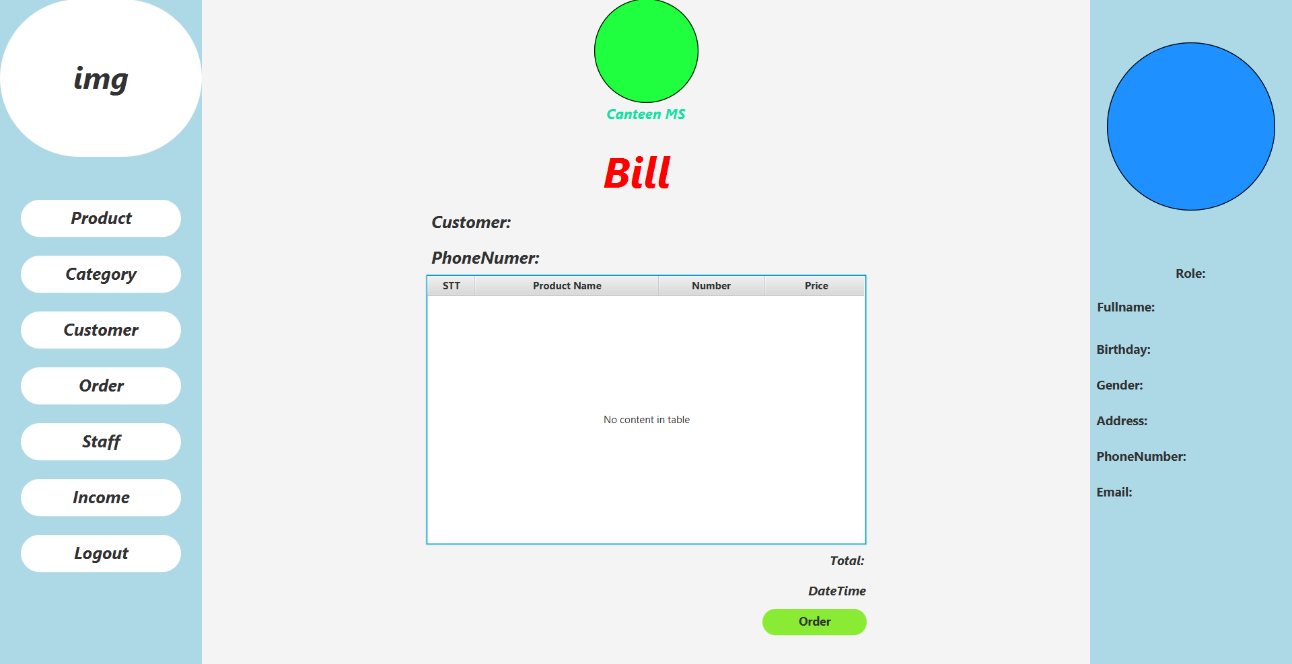


Picture 3.15: When you add to order successfully



Picture 3.16: When you add to order fail

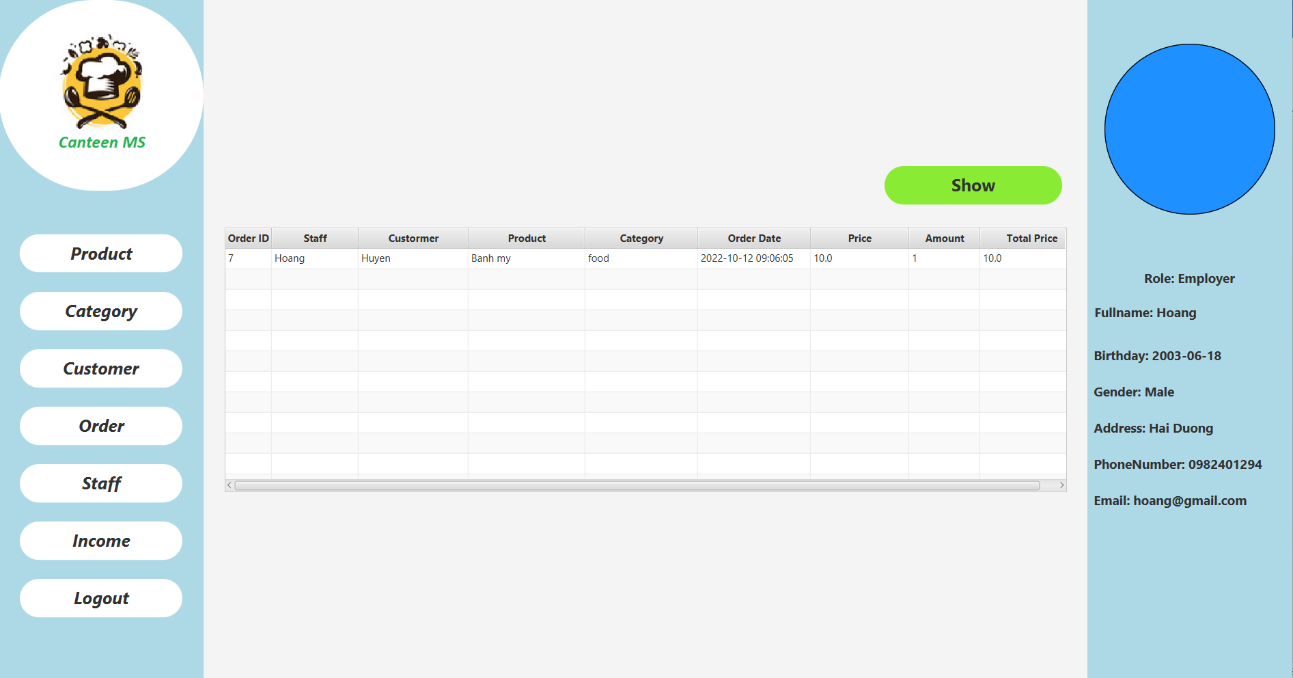
To view the bill you ordered, click on the cart.



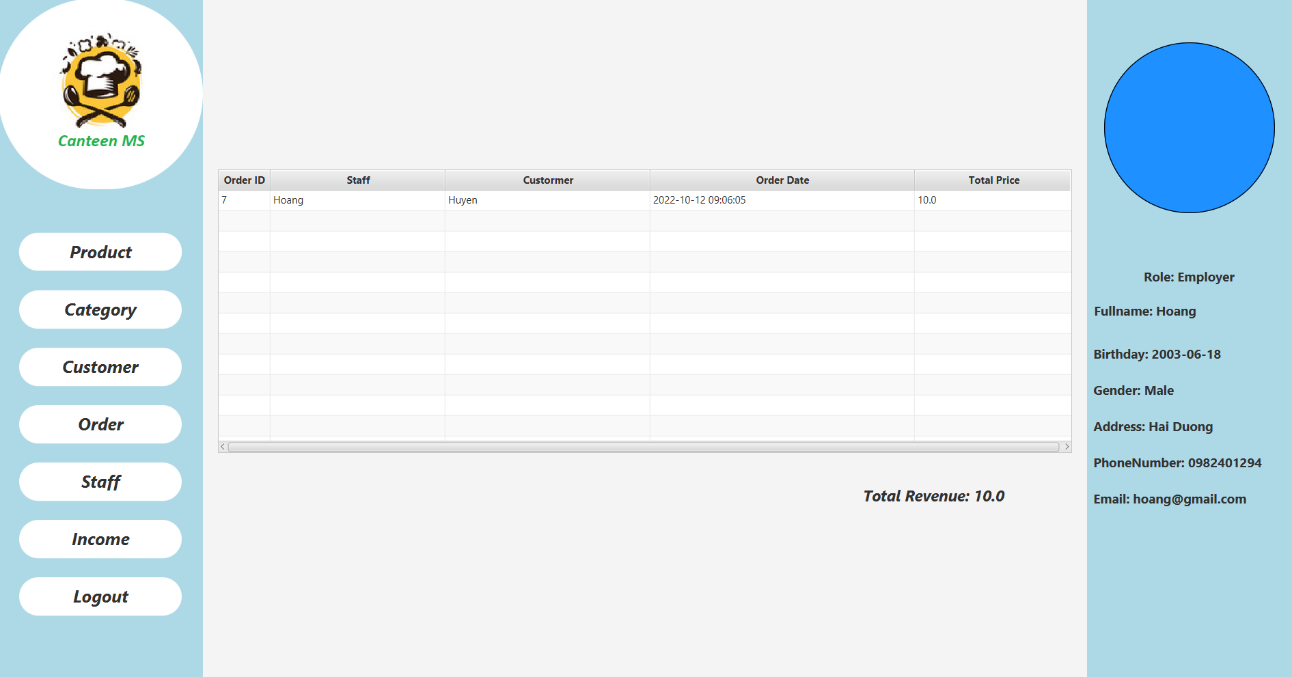
Picture 3.17: Bill page

When you press the order button, the product will be pushed to the system.

To see what products to do to serve customers, we select the order button in the menu on the left.



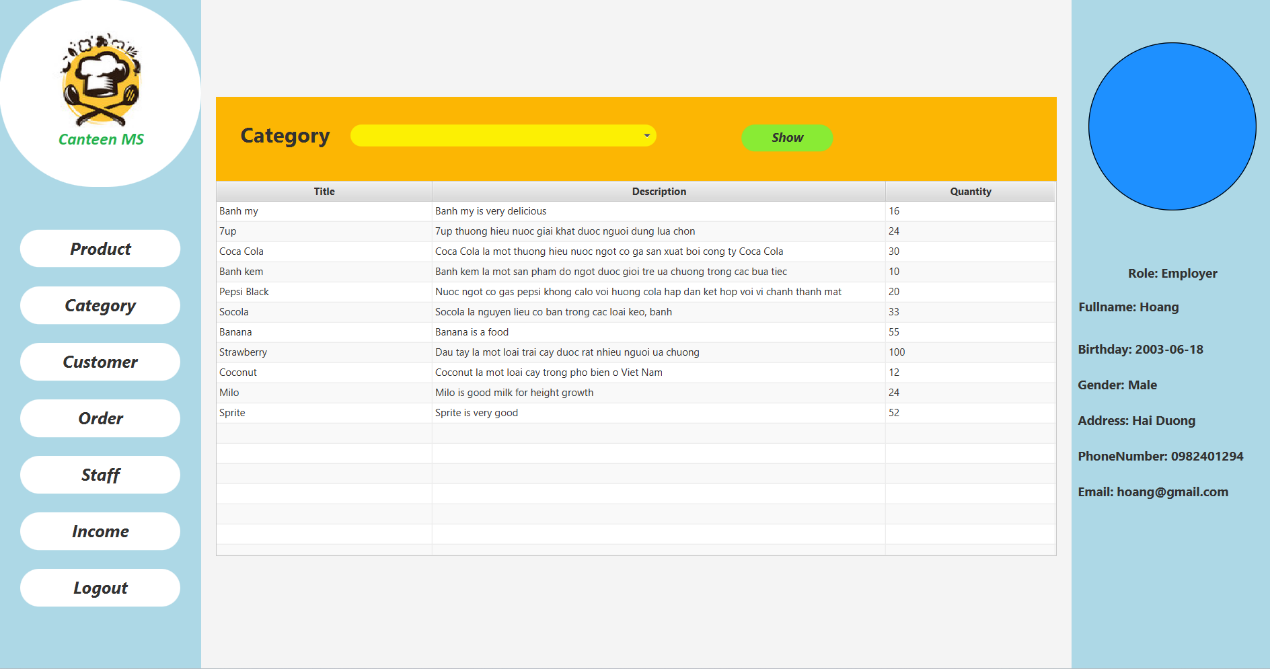
Picture 3.18: Order detail page



Picture 3.19: When you click show

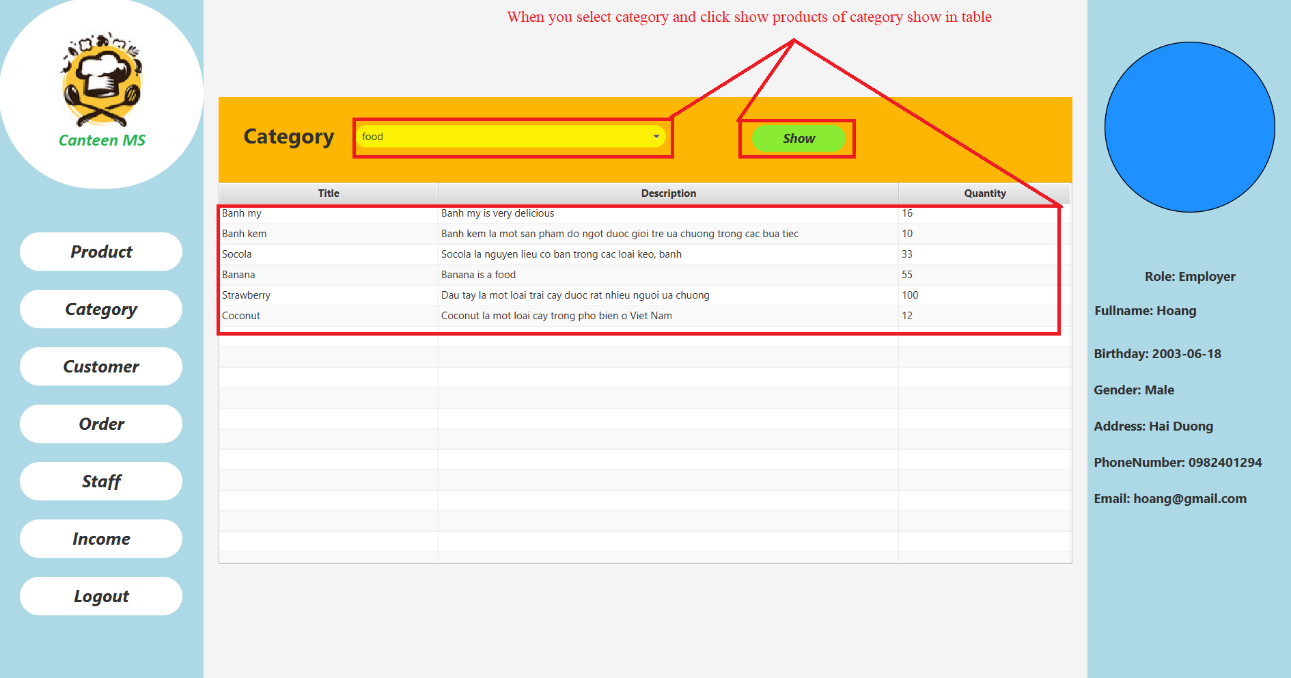
## The Category page

The category page allows users to see all the products that the canteen is selling.



Picture 3.20: Category page

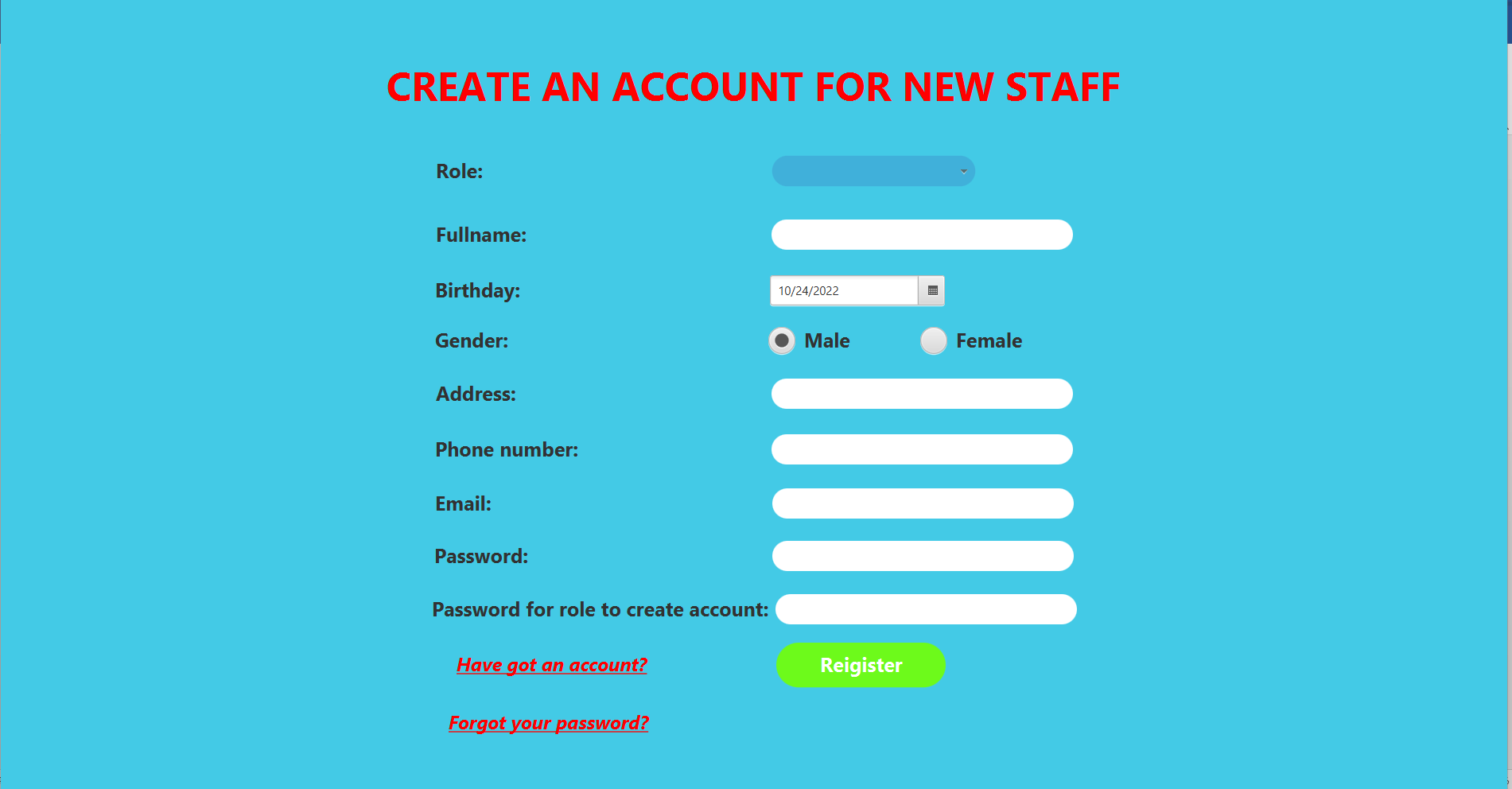
To view products by category, we select the category to view and press the show button, the product will be displayed according to the selected category in the table.



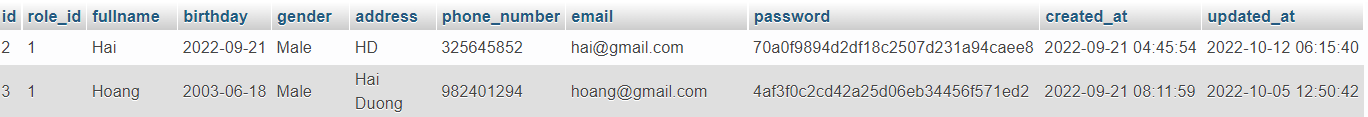
Picture 3.21: Product of category

## The Register and Login page

About register page you need to fill in some information to register an account. You need to fill in role, fullname, birthday, gender, address, phone number, email, password and finally is password for role. If your email or phone number has been used, we will alert you and you need to use another email or phone number to register. Make sure that you remember your password because now we haven’t developed forgot password feature. And if your register successfully the information will be sent to the database, and you will link to login page. And if you had an account you could click to “*Have got an account?*” and it will let you to login page.

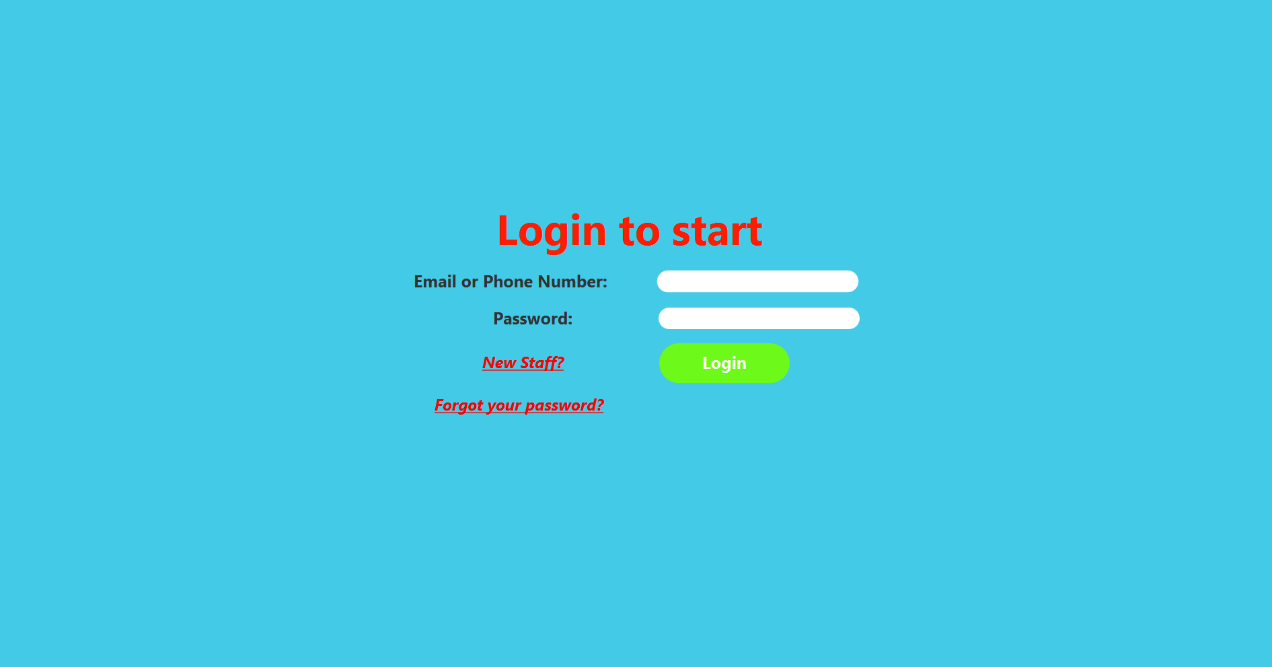


Picture 3.22: Register page

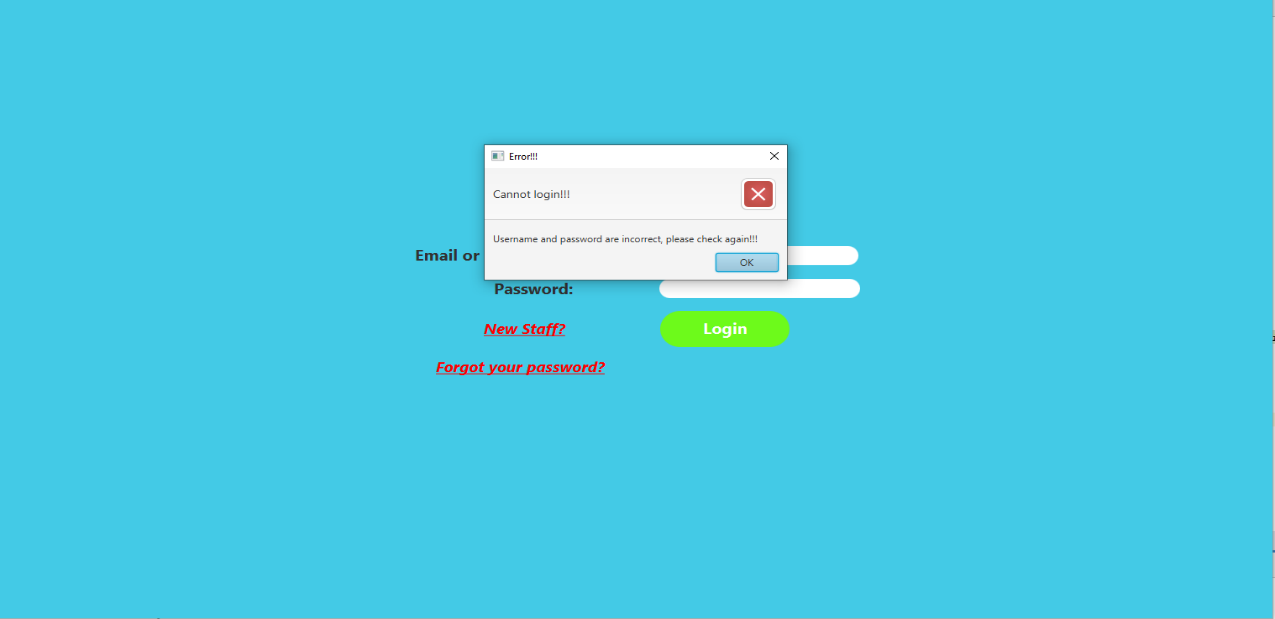


Picture 3.14: The database received your register successfully

In the login page you only need to fill in username and password. If it is correct, you will be sent home page. If your username and password don’t match the database, we will alert you so you can check your account. If you don’t have an account, you can click to “*New Staff?*” and it will lead you to register page.



Picture 3.23: Login page



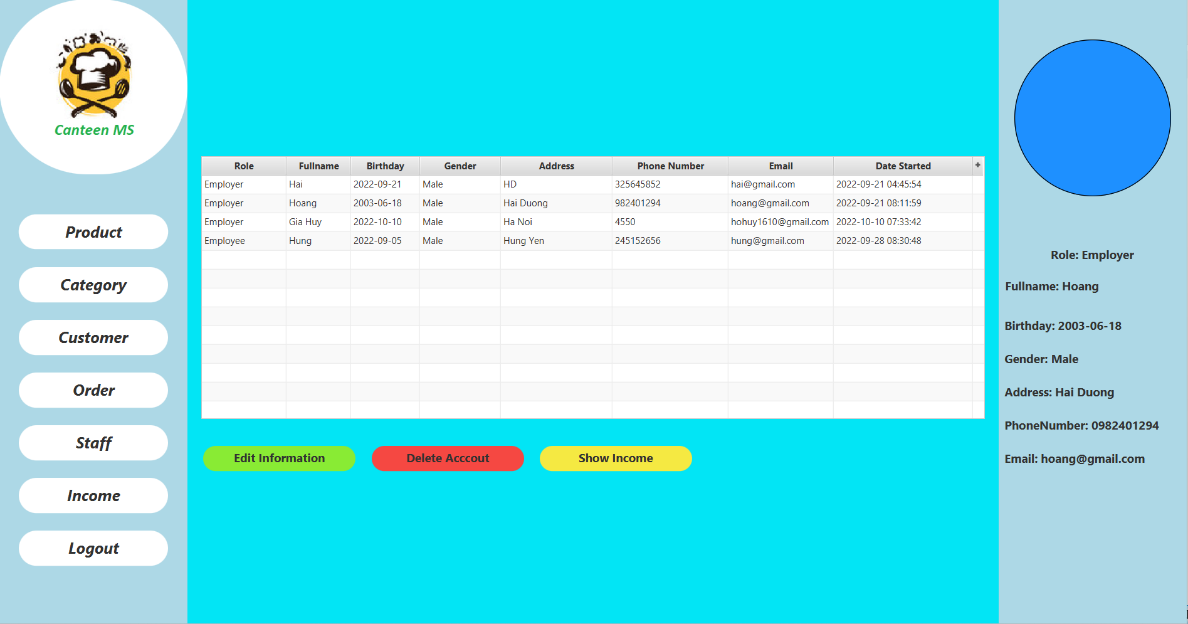
Picture 3.24: When you fill in wrong username or password

## Staff and Income page

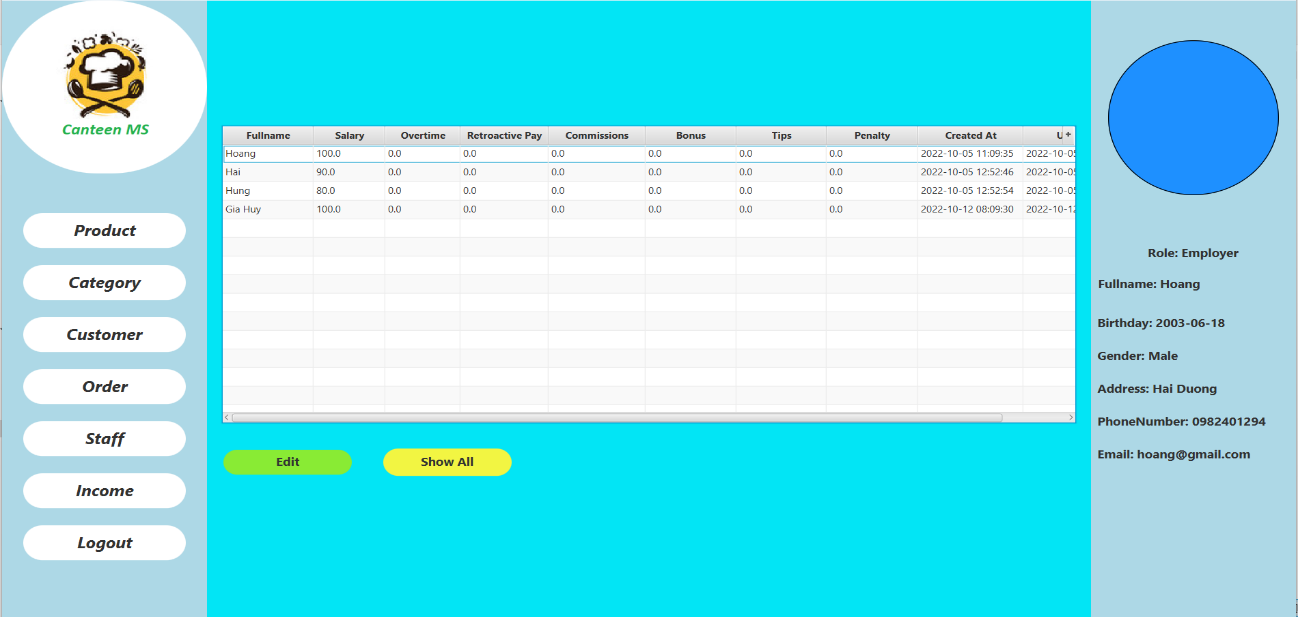
On the staff page, you can view employee information such as role, fullname, birthday, gender, address, phone number, email, date started.

Besides, there are buttons to edit and delete accounts, view income.

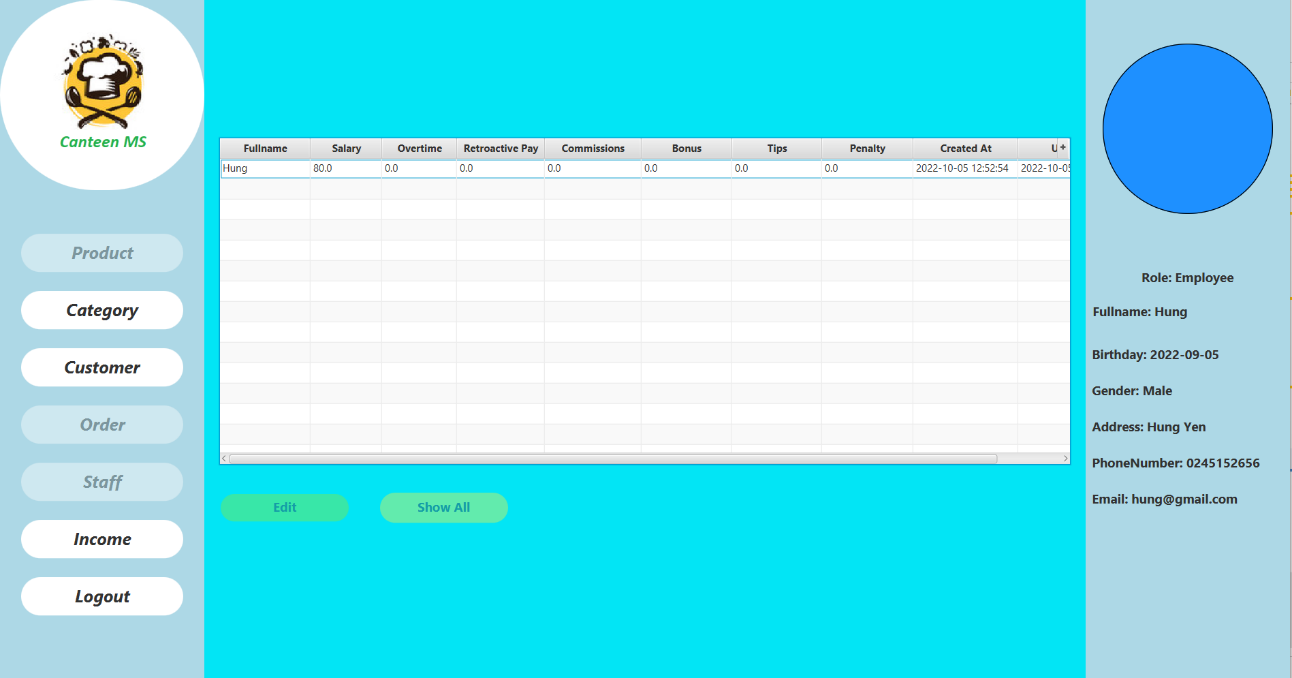
On the icome page, we will see the details of the salary, the employer will see all of everyone's salary and edit the salary accordingly, and the employee can only see his/her own salary.



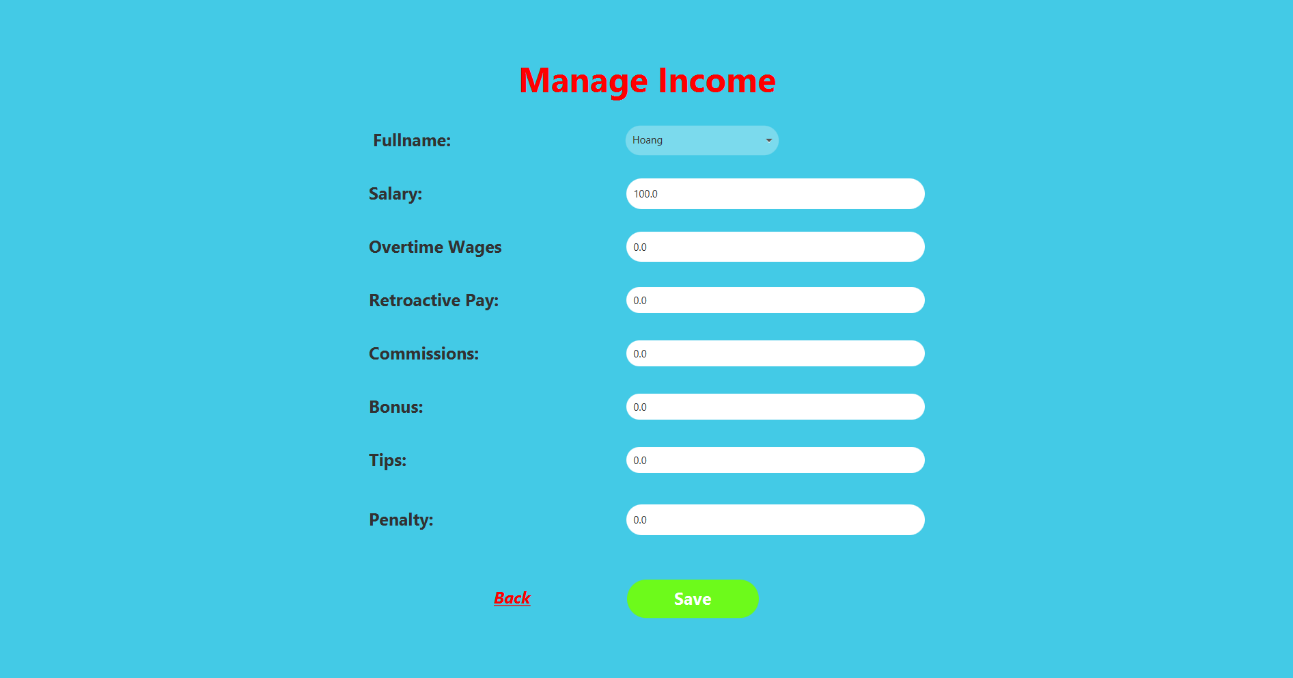
Picture 3.25: Staff page



Picture 3.26: Income with employer account



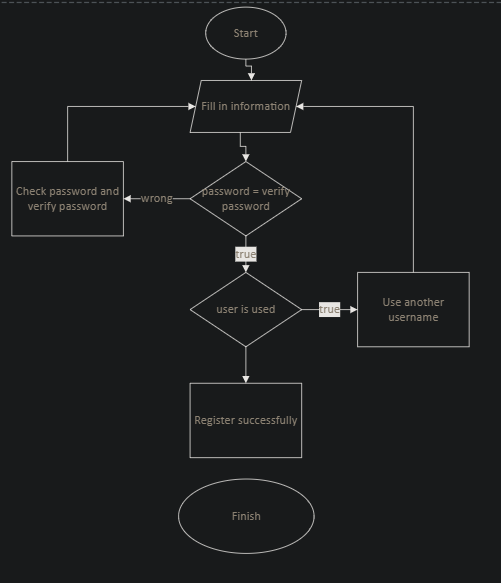
Picture 3.27: Income with employee account



Picture 3.28: Edit income

## System fucntions design

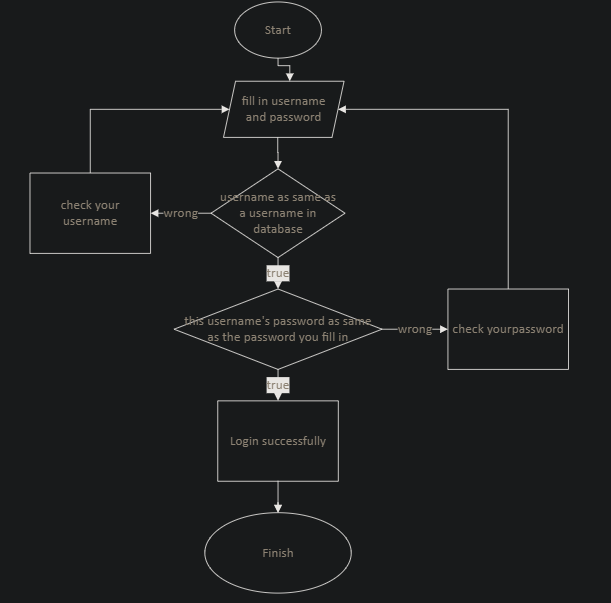
## Register function



Picture 3.19: Function registration page

When you fill in information and send a register account form if password is not as same as verify password we must check password and verifypassword. If the password is as same as verify password the database will check that had username been used. If the username was used we must find another username. If no one used this username we can use this username.

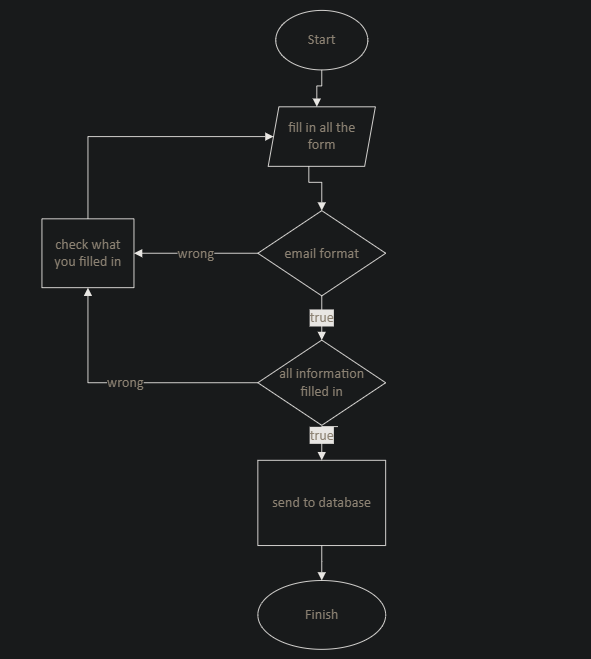
## Login function



Picture 3.20: Function of login

After you fill in your username and password, we will select username and password from the database which has username is as same as username you filled. If there is no username in database is as same as username you fill you will need to check your username. If there is a username as same as your username, it will check if the password of its as same as your password. If it correctly your login complete but if it wrong you will need to check your password again.

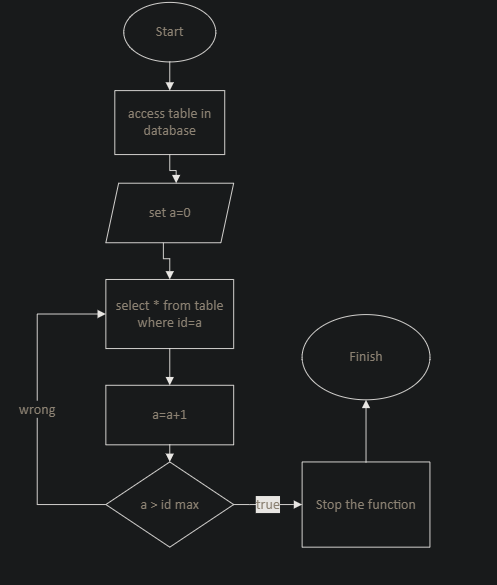
## Sending report function



Picture 3.21: Function of sending a report

When you start function, you must fill in all the form in this page. First of all the system will check your email format. If it is true you will go to check all information. And it all information is checked that you have fill in all information, it will sent information to the database successfully. If there is any wrong in checking email format step or in checking all information you fill in you need to check what you filled in and after that you must fix what you fill in to and check it again.

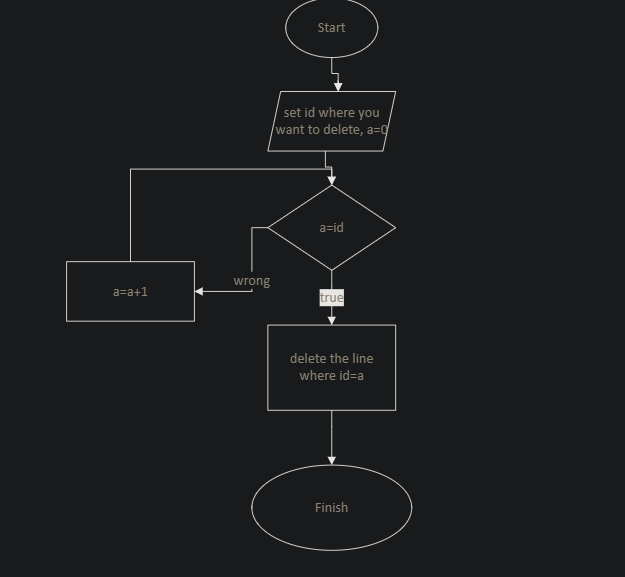
## Show table function



Picture 3.22: Function show table of database

When you start the function, you set a variable a = 0, after that you select all information from that table you want where id=a. After that you set a=a+1. If a isn’t greater than id max you turn back to step select all information from that table you want where id=a. If a greater than id max you stop the function and finish it

## Delete database function



Picture 3.22: Function delete a database

When you start the function, you set what id you want to delete and set a variable a=0. If a=id you want to delete you will delete this line and finish function. But if a<id you will let a=a+1 and repeat check a with id.

# Task sheet

Table 4.1: Task sheet

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task** | **Start time** | **Finish time** | **Implemented by** | **Evaluation**  **(%)** |
| **Feasibility study phase** |  |  |  |  |
| Write document | 04/04/2022 | 04/04/2022 | Duong Van Hoang | 20 |
| Contribute ideas | 04/04/2022 | 04/04/2022 | Nguyen Huu Hieu | 30 |
| Contribute ideas | 04/04/2022 | 04/04/2022 | Pham Dang Duc | 10 |
| Contribute ideas | 04/04/2022 | 04/04/2022 | Nguyen Trung Duc | 10 |
| Approve and update | 04/04/2022 | 05/04/2022 | Duong Van Hoang  Nguyen Huu Hieu | 30 |
| **Requirement Analysis phase** |  |  |  |  |
| Analysis Database | 04/04/2022 | 07/04/2022 | Nguyen Huu Hieu | 30 |
| Analysis Functions | 04/04/2022 | 07/04/2022 | Duong Van Hoang | 30 |
| Contribute ideas | 07/04/2022 | 07/04/2022 | Pham Dang Duc | 10 |
| Approve and update | 08/04/2022 | 08/04/2022 | Nguyen Huu Hieu | 30 |
|  |  |  |  |  |
|  |  |  |  |  |
| **Design phase** |  |  |  |  |
| Buid Database | 08/04/2022 | 09/04/2022 | Nguyen Huu Hieu | 15 |
| Contribute ideas | 09/04/2022 | 09/04/2022 | Duong Van Hoang | 5 |
| Approve and update | 09/04/2022 | 09/04/2022 | Nguyen Huu Hieu | 10 |
| Deign pages | 18/04/2022 | 18/04/2022 | Duong Van Hoang | 15 |
| Contribute ideas | 18/04/2022 | 18/04/2022 | Nguyen Huu Hieu | 5 |
| Contribute ideas | 18/04/2022 | 18/04/2022 | Nguyen Trung Duc | 5 |
| Contribute ideas | 18/04/2022 | 18/04/2022 | Pham Dang Duc | 5 |
| Approve and update | 18/04/2022 | 18/04/2022 | Duong Van Hoang | 10 |
| Insert data | 29/04/2022 | 29/04/2022 | Nguyen Huu Hieu | 10 |
| Contribute ideas | 02/05/2022 | 02/05/2022 | Nguyen Trung Duc | 5 |
| Approve and update | 04/05/2022 | 04/05/2022 | Nguyen Huu Hieu | 10 |
|  |  |  |  |  |
|  |  |  |  |  |
| **Development phase** |  |  |  |  |
| Buid pages | 18/04/2022 | 18/04/2022 | Duong Van Hoang | 10 |
| Contribute ideas | 18/04/2022 | 23/04/2022 | Pham Dang Duc | 5 |
| Contribute ideas | 18/04/2022 | 20/04/2022 | Nguyen Trung Duc | 5 |
| Approve and update | 20/04/2022 | 04/05/2022 | Duong Van Hoang | 10 |
| Fill data from database | 29/04/2022 | 29/04/2022 | Nguyen Huu Hieu | 10 |
| Contribute ideas | 02/05/2022 | 02/05/2022 | Nguyen Trung Duc | 5 |
| Approve and update | 04/05/2022 | 04/05/2022 | Nguyen Huu Hieu | 5 |
| Admin pages | 04/05/2022 | 04/05/2022 | Duong Van Hoang | 10 |
| Contribute ideas | 05/05/2022 | 05/05/2022 | Nguyen Huu Hieu | 5 |
| Approve and update | 05/05/2022 | 07/05/2022 | Duong Van Hoang | 10 |
| Responsive pages | 25/04/2022 | 25/04/2022 | Duong Van Hoang | 10 |
| Contribute ideas | 26/04/2022 | 27/04/2022 | Nguyen Huu Hieu | 5 |
| Approve and update | 27/04/2022 | 29/04/2022 | Duong Van Hoang | 10 |
|  |  |  |  |  |
|  |  |  |  |  |
| **Testing phase** |  |  |  |  |
| Testing project | 24/05/2022 | 26/05/2022 | Nguyen Huu Hieu | 25 |
| Testing project | 24/05/2022 | 26/05/2022 | Pham Dang Duc | 25 |
| Testing project | 24/05/2022 | 26/05/2022 | Nguyen Trung Duc | 25 |
| Testing project | 24/05/2022 | 26/05/2022 | Duong Van Hoang | 25 |
| **Deployment phase** |  |  |  |  |
| Record demo video | 27/05/2022 | 27/05/2022 | Nguyen Huu Hieu | 25 |
| Slide introduce project | 27/05/2022 | 27/05/2022 | Duong Van Hoang | 25 |
| Approve | 27/05/2022 | 27/05/2022 | Duong Van Hoang | 25 |
| Deploy files | 27/05/2022 | 27/05/2022 | Duong Van Hoang  Nguyen Huu Hieu | 25 |

# Validation Checklists

Table 5.1: Validation checklists

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Items | Sub-item | Step to Excute | Expected output | Microsoft Edge | Chrome | Cốc Cốc |
| Front end | |  |  |  |  |  |  |
| 1 | Home page | Display | 1. Open home page | Every layout as same as designed | pass | pass | pass |
| 2 | Home page | Link to other page | 1.Open home page. 2. Click any link to go to other page | Every layout as same as designed, all the link work good | pass | pass | pass |
| 3 | Home page | Scroll-up | 1. Open home page 2. Scroll down the page 3.Click the scroll up button | Every layout still good, Scroll up button not hidden. This button can help scroll up to the head of the page easier | pass | pass | pass |
| 4 | Admin page | All database can show to admin see | 1. Open admin page 2. Check the admin table | Every information can show in the table in the admin page, there is no error about display | pass | pass | pass |
| 5 | Admin page | Delete an account or a report | 1. Open admin page 2. Check the admin table 3. Click delete to delete a line | Ony this line was deleted, no other lines were deleted | pass | pass | pass |
| 5 | Register page | Create an account | 1. Open register page 2. Fill in information | Everyform can fill in normal, if you fill wrong format you cannot register | pass | pass | pass |
| 6 | Login page | Login an account | 1. Open login page 2. Fill in information | Everyform can fill in normal | pass | pass | pass |
| 7 | Report page | Send a report | 1. Open report page 2. Fill in information | Everyform can fill in normal, if you fill wrong format or not fill all information you cannot register | pass | pass | pass |
| Back end | | | | | | | | |
| 1 | Register page | Create an account | 1. Open register page 2. Fill in information | Database can receive all this information | pass | pass | pass |
| 2 | Login page | Login an account | 1. Open login page 2. Fill in information | Login page can access database. If you fill correct username and password you can login | pass | pass | pass |
| 3 | Report page | Send a report | 1. Open report page 2. Fill in information | You can send all the report to the database | pass | pass | pass |
| 4 | Admin page | Delete an account or a report | 1. Open admin page 2. Check the admin table 3. Click delete to delete a line | Ony this line in database was deleted, no other lines were deleted | pass | pass | pass |
| 5 | Admin page | All database can show to admin see | 1. Open admin page 2. Check the admin table | Every information in admin page is as same as database. When database change it will change too | pass | pass | pass |