

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC - KỸ THUẬT MÁY TÍNH



HỆ CƠ SỞ DỮ LIỆU (CO2013)

BÀI TẬP LỚN 1

**Đăng ký khách hàng thành viên tại chuỗi
cửa hàng café**

Giáo viên hướng dẫn: Đỗ Thanh Thái

Nhóm: 01

Sinh viên: Nguyễn Huy Hoàng - 1913438
Nguyễn Đức Triết - 2012537
Hoàng Nhật Linh Kiều - 2120034

TP. HỒ CHÍ MINH, THÁNG 12/2022

Mục lục

1	Giới thiệu	3
1.1	Giới thiệu đề tài	3
1.2	Nội dung báo cáo	3
2	Phân công công việc toàn bài tập	4
3	Giai đoạn 1	5
3.1	Tìm hiểu các yêu cầu dữ liệu, yêu cầu chức năng, và yêu cầu phi chức năng của mỗi hệ thống	5
3.1.1	Yêu cầu dữ liệu	5
3.1.2	Yêu cầu chức năng	7
3.1.3	Yêu cầu phi chức năng	8
3.2	Thiết kế cơ sở dữ liệu ý niệm	9
3.3	Thiết kế cơ sở dữ liệu luận lý	9
3.3.1	Lược đồ cơ sở dữ liệu quan hệ	10
3.3.2	Định nghĩa lược đồ cơ sở dữ liệu quan hệ và các nhóm người dùng trên DBMS.	10
3.3.3	Công nghệ phát triển ứng dụng	12
3.3.4	Hệ quản trị cơ sở dữ liệu	12
3.4	Tổng kết	14
3.4.1	Kết quả đạt được	14
3.4.2	Hướng phát triển	14
4	Giai đoạn 2 (Phát triển ứng dụng)	15
4.1	Viết các Functions/Procedures/Assertions/Triggers.	15
4.1.1	Các CREATE Procrdures	15
4.1.2	Các CREATE TRIGGER	18
4.1.3	Các CREATE FUNCTION	21
4.2	Định nghĩa các nhóm người dùng và phân quyền ở mức ứng dụng	22
4.3	Kiến trúc hệ thống	22
4.4	Thiết kế chi tiết cho ứng dụng.	25
4.5	Chuẩn hóa về dạng chuẩn BCNF cho lược đồ cơ sở dữ liệu.	28
4.6	Chỉ mục	32
4.7	Triển khai thiết kế và hiện thực phần mềm	34

4.8	Đánh giá hệ thống	41
TÀI LIỆU THAM KHẢO		41

1 Giới thiệu

1.1 Giới thiệu đề tài

Việc lưu trữ dữ liệu khách hàng đăng ký làm thành viên của một chuỗi cafe bằng phương pháp cũ thường gây ra rất nhiều bất tiện cho nhân viên và cả quản lý. Những khó khăn có thể kể đến như: tìm kiếm thông tin khách hàng; thêm, xóa, chỉnh sửa thông tin khách hàng thành viên, dễ dàng thất lạc khi lưu trữ dạng giấy,...

Xuất phát từ thực tiễn trên, nhóm đề xuất thiết lập Hệ thống đăng ký khách hàng thành viên cho chuỗi cafe. Với hệ thống này, người dùng với tư cách là quản lý và nhân viên có thể dễ dàng quản lý Hệ cơ sở dữ liệu của bộ phận đăng ký khách hàng; người dùng với tư cách là khách hàng sẽ dễ dàng đăng ký thành viên hơn và được nhận thông báo các khuyến mãi. Ngoài ra, Hệ thống cũng giúp tiết kiệm nhiều thời gian và tài nguyên.

Hệ thống sẽ giúp người dùng hạn chế việc trùng lặp trong việc lưu trữ thông tin, đảm bảo nhất quán, toàn vẹn dữ liệu; cho phép nhân viên truy xuất dữ liệu theo nhiều cách khác nhau. Hệ thống cũng có thể giúp giảm nhẹ khối lượng công việc của nhân viên và quản lý.

1.2 Nội dung báo cáo

Nội dung phần báo cáo của nhóm bao gồm 3 phần chính:

- **Phần 1:** Bao gồm các yêu cầu dữ liệu, các yêu cầu chức năng và các yêu cầu phi chức năng.
- **Phần 2:** Là cơ sở dữ liệu ý niệm(EER).
- **Phần 3:** Bao gồm việc lựa chọn hệ quản trị cơ sở dữ liệu, công nghệ phát triển ứng dụng và lược đồ cơ sở dữ liệu quan hệ(Mapping).

2 Phân công công việc toàn bài tập

Phương diện	Huy Hoàng	Đức Triết	Linh Kiều
Mức độ tham gia (Tổng 100%)	33.3%	33.3%	33.3%
Đánh giá độ hoàn thiện công việc được giao	100%	100%	100%

3 Giai đoạn 1

3.1 Tìm hiểu các yêu cầu dữ liệu, yêu cầu chức năng, và yêu cầu phi chức năng của mỗi hệ thống

Các yêu cầu có độ ưu tiên từ trên xuống dưới.

3.1.1 Yêu cầu dữ liệu

1. **Khách hàng** được đăng kí bởi một **Nhân viên thu ngân** và Nhân viên thu ngân có thể đăng kí cho nhiều khách hàng.
2. Để đăng ký thành viên thì **Khách hàng** phải có số điện thoại, thôn tin tên, năm sinh.
3. Mỗi **Khách hàng** chỉ được đăng ký thành viên cho chính bản thân mình.
4. **Quán cafe** phải có ít nhất 4 **Nhân viên**: một **Nhân viên pha chế**, một **Nhân viên phục vụ**, một **Nhân viên thu ngân** và một **Người quản lý**. tuy nhiên nhân viên phục vụ không có thuộc tính riêng biệt nên không sinh ra lớp con.
5. **Nhân viên** của chuỗi siêu thị phải trên 18 tuổi, có CMND/CCCD, có các thông tin tên, năm sinh, địa chỉ, số điện thoại, tài khoản ngân hàng.
6. **Nhân viên** có nhân viên toàn thời gian và bán thời gian. nhân viên toàn thời gian được trả lương theo tháng, bán thời gian được trả lương theo giờ.
7. **Nhân viên** chỉ làm việc tại một **quán cafe** và một quán cafe có thể có nhiều Nhân viên đang làm việc.
8. **Người quản lý** chỉ có thể quản lý một **quán cafe**.
9. **Đơn hàng** chỉ thuộc một **Khách hàng** duy nhất và tất cả Khách hàng đều phải có ít nhất một đơn hàng.
10. **Đơn hàng** được xử lý bởi một **Thu ngân** và một Thu ngân có thể xử lý nhiều đơn hàng.
11. **Khách hàng** có thể có nhiều **Khuyến mãi** và một Khuyến mãi có thể được nhận bởi nhiều Khách hàng.
12. **Đơn hàng** chỉ được áp dụng một **Mã khuyến mãi** và một Mã khuyến mãi chỉ được áp dụng cho một Đơn hàng.

Trong đó:

Các thực thể mạnh là:

- Khách hàng: Mã khách hàng(thuộc tính khóa), Họ và Tên(phức)(Họ và tên lót, Tên), SĐT, Giới tính, Năm sinh, Điểm tích lũy.
- Nhân viên: Mã nhân viên(thuộc tính khóa), Họ và Tên(phức)(Họ và tên lót, Tên), SĐT(đa trị), Giới tính, Năm sinh, Tuổi(dẫn xuất), Địa chỉ, Số CMND/CCCD(thuộc tính khóa), Thời gian làm việc, mức lương, tài khoản ngân hàng.
- Ngân viên thu ngân: kế thừa thuộc tính của nhân viên, KPI.
- Người quản lý: kế thừa thuộc tính của nhân viên, chứng chỉ.
- Quán cà phê: Mã chi nhánh (thuộc tính khóa), Địa chỉ.
- Đơn hàng: Mã đơn hàng (thuộc tính khóa), Tổng số tiền (dẫn xuất), Ngày thanh toán, Điểm thưởng, ghi chú.
- Thực phẩm: Tên, Mã hàng hóa(thuộc tính khóa), mô tả, giá niêm yết, ngày sản xuất, hạn sử dụng.
- Đồ uống: kế thừa từ Thực phẩm, dung tích.
- Thức ăn kế thừa từ Thực phẩm, khối lượng.

Các thực thể yếu là: Khuyến mãi: Mã khuyến mãi (thuộc tính khóa riêng phần), Thời hạn sử dụng, Nội dung, Giảm giá.

Thuộc tính đa trị là: Số điện thoại, Chứng chỉ, Mô tả.

Thuộc tính dẫn xuất là: Tổng số tiền, Tuổi.

Thuộc tính phức là:

- Họ và Tên: Họ và tên lót, Tên.
- KPI: Số giao dịch, Quầy, Số tiền giao dịch.

Các ràng buộc:

Các ràng buộc ngữ nghĩa:

- Mỗi Khách hàng chỉ được đăng ký thành viên cho bản thân mình.
- Mỗi **Khách hàng** chỉ được đăng ký thành viên cho chính bản thân mình.
- **Quán cafe** ít nhất phải có 4 nhân viên: 1 Nhân viên phục vụ, 1 pha chế., 1 Nhân viên thu ngân, 1 Người quản lý.

Các ràng buộc miền trị:

- Để đăng ký thành viên thì Khách hàng phải trên 16 tuổi, phải có CMND/CCCD.
- **Nhân viên** của chuỗi siêu thị phải trên 18 tuổi, có CMND/CCCD, phải làm ít nhất 4 tiếng 1 ngày.
- **Người quản lý** phải có ít nhất 3 năm kinh nghiệm.

3.1.2 Yêu cầu chức năng

- Đối với khách hàng
 - Khi đăng ký thành viên, khách hàng có thể xem thông tin hỗ trợ cho thành viên của chuỗi cửa hàng. Với yêu cầu này, luồng dữ liệu đơn giản sẽ đi từ hệ thống đến khách hàng với đầu vào là thông tin hỗ trợ cho khách hàng thành viên và đầu ra là nội dung thông tin hỗ trợ.
 - Hệ thống đăng kí thành viên xác nhận khách hàng là thành viên chính thức và gửi thông báo cho khách hàng khi tài khoản được tạo thành công.
 - Hệ thống cho phép khách hàng xem thông tin của mình.
 - Khách hàng có thể chỉnh sửa thông tin của mình. Trên hệ quy chiếu dữ liệu, dữ liệu cập nhật từ khách hàng sẽ được gửi đến hệ thống với đầu vào là thông tin khách hàng sau khi chỉnh sửa và đầu ra là nội dung thông tin khách hàng đã chỉnh sửa.
 - Khách hàng có thể xem lịch sử giao dịch của mình.
 - Khách hàng có thể yêu cầu hệ thống gửi nội dung thống kê giao dịch hàng tháng cho mình.
 - Khách hàng có thể gửi câu hỏi về chương trình đăng kí thành viên lên hệ thống để thu ngân trả lời.
 - Khách hàng thành viên được tự động tích điểm sau mỗi lần mua hàng.
- Đối với quản lý
 - Xem thống kê số lượng khách hàng đăng ký trong một khoảng thời gian (tuần, tháng,...).
 - Xem thống kê số lượng đơn hàng và doanh thu trong một khoảng thời gian (tuần, tháng,...).
 - Xem và chỉnh sửa thông tin về các loại thực phẩm.

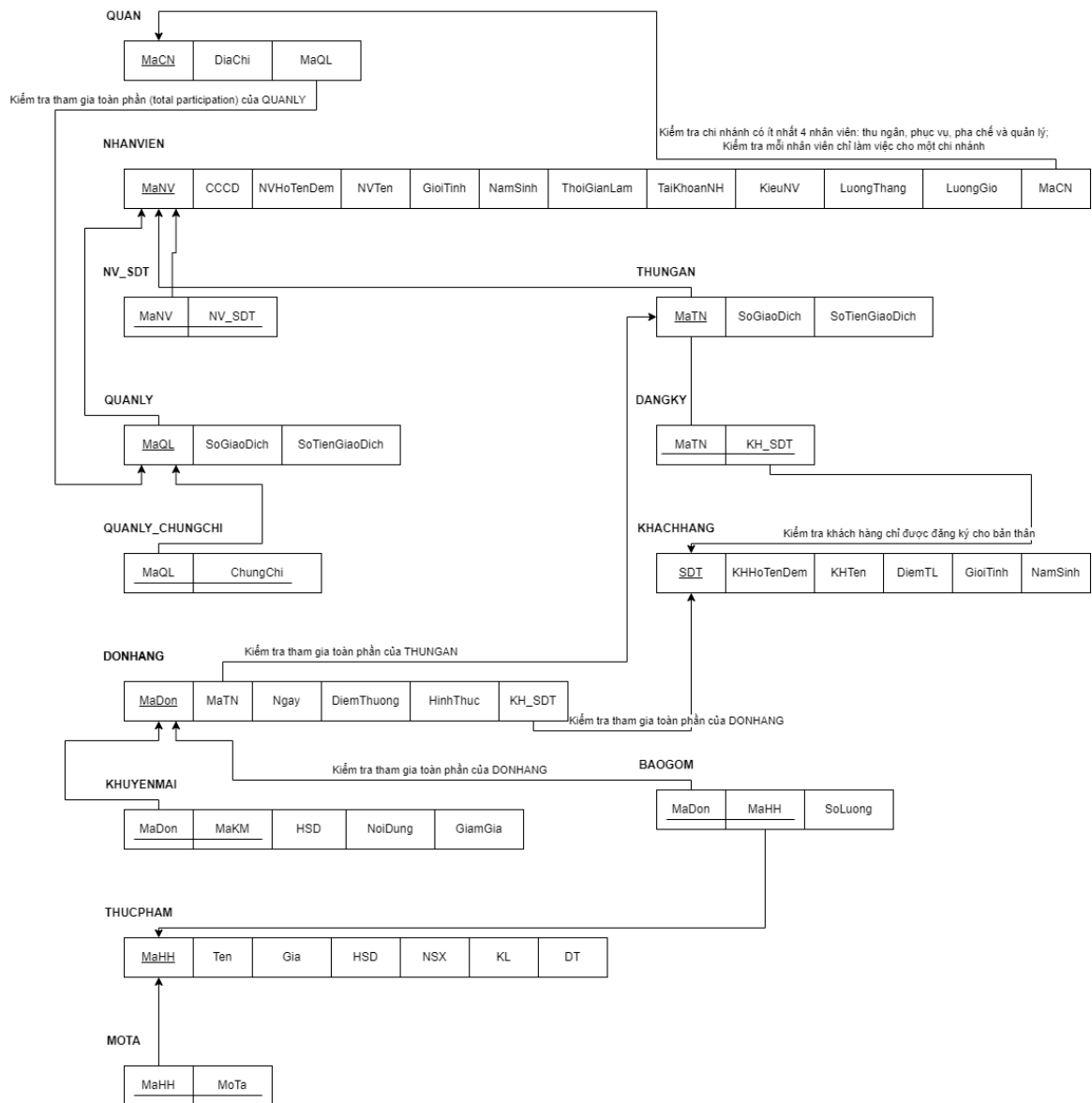
- Quản lý có thể xem thông tin khách hàng và chỉnh sửa trong trường hợp cần thiết.
- Quản lý có thể xem thông tin về lý lịch, ca làm, kiểu nhân viên của toàn bộ nhân viên thuộc chi nhánh mà mình quản lý.
- Đối với nhân viên
 - Nhân viên được thay đổi nội dung về thông tin liên lạc của bản thân.
 - Nhân viên bán thời gian được chọn ca làm trong tuần.
 - Nhân viên thu ngân có thể xem thống kê số lượng đơn hàng và doanh thu trong một khoảng thời gian (tuần, tháng,...).
 - Nhân viên phục vụ và pha chế được xem thông tin về các loại thực phẩm.

3.1.3 Yêu cầu phi chức năng

1. Giao diện đăng ký thành viên thân thiện, dễ sử dụng, khách hàng, nhân viên và quản lý có thể sử dụng thành thạo sau khi được hướng dẫn trong 30 phút.
2. Bảo mật cơ sở dữ liệu của khách hàng, nhân viên phải đáp ứng các yêu cầu theo quy định; đảm bảo tuyệt đối không rò rỉ thông tin của khách hàng và nhân viên. Yêu cầu đến từ những nhóm người dùng nhân viên và khách hàng.
3. Thông báo cho khách hàng với độ trễ không quá 6 tiếng.
4. Thời gian mỗi lần bảo trì hệ thống ít hơn 1 giờ.
5. Hệ thống sẽ tải dưới 5 giây khi số lượng người dùng đồng thời lớn hơn 200.
6. Hệ thống có thể xử lý đồng thời 1000 người mà không bị quá tải.
7. Hệ thống có thể lưu trữ tối đa 500.000 thành viên.

3.3.1 Lược đồ cơ sở dữ liệu quan hệ

NHÓM 01: LƯỢC ĐỒ CƠ SỞ DỮ LIỆU - QUAN HỆ



Hình 3.2: Lược đồ cơ sở dữ liệu quan hệ

[Link tới hình ảnh](#)

3.3.2 Định nghĩa lược đồ cơ sở dữ liệu quan hệ và các nhóm người dùng trên DBMS.

- Định nghĩa lược đồ cơ sở dữ liệu quan hệ trên DBMS đã được chọn và cài đặt:

sách khách hàng được họ đăng ký thành viên và đơn hàng do họ ghi.

- Khách hàng: có quyền truy vấn dữ liệu về thực phẩm, thông tin cá nhân và thông tin về các chương trình khuyến mãi

3.3.3 Công nghệ phát triển ứng dụng

Công nghệ phát triển ứng dụng: Web.

Front-end: ReactJS, HTML, CSS.

Back-end: NodeJS, Express.

Framework: Bootstrap.

Lý do sử dụng:

Với mô hình đăng ký chuỗi cafe thì xây dựng một trang web để quản lý đăng ký khách hàng là một yêu cầu cần thiết. Nhóm em xây dựng web với những công nghệ như: ReactJS, NodeJS là những công nghệ phổ biến ngày nay, dễ sử dụng và bảo trì hệ thống.

- ReactJS: giúp việc viết các đoạn code JS dễ dàng hơn, có nhiều công cụ phát triển, dễ để viết các test case giao diện,
- NodeJS: có tốc độ xử lý nhanh, dễ dàng mở rộng khi có nhu cầu, có khả năng xử lý nhiều Request/s trong thời gian ngắn nhất,

3.3.4 Hệ quản trị cơ sở dữ liệu

Nhóm chọn hệ quản trị cơ sở dữ liệu MySQL:



Hình 3.4: Hệ quản trị cơ sở dữ liệu MySQL

MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (Relational Database Management System, viết tắt là RDBMS) hoạt động theo mô hình client-server.

Lý do sử dụng:

- Linh hoạt và dễ dùng: quá trình cài đặt tương đối đơn giản, có thể dễ dàng chỉnh sửa source code.
- Hiệu năng cao: dữ liệu được đáp ứng với tốc độ cao, mượt mà.
- Tiêu chuẩn trong ngành: người Ngành công nghệ và dữ liệu đã sử dụng MySQL nhiều năm, vì vậy nó là một kỹ năng căn bản một chuyên gia lập trình.
- An toàn: Vấn đề an toàn luôn là vấn đề cực kì quan trọng trong ngành dữ liệu và MySQL đảm bảo được tiêu chuẩn bảo mật rất cao.

3.4 Tổng kết

3.4.1 Kết quả đạt được

Thông qua đề tài này, chúng em đã được thực hành kĩ các kiến thức đã được học, từ yêu cầu dữ liệu, chức năng, phi chức năng, đến cách thiết kế cơ sở dữ liệu ý niệm và luận lý.

Thêm vào đó, nhóm tự tin khẳng định rằng các yêu cầu đề tài mà nhóm đã thực hiện đã đáp ứng tốt các yêu cầu cần thiết để phát triển hệ thống đăng kí khách hàng thành viên của chuỗi cafe.

Trong quá trình triển khai, nhóm em đã rút ra được những điều làm được và chưa làm được, cụ thể:

Làm được:

- Các yêu cầu đề tài đã đáp ứng cơ bản yêu cầu của hệ thống.
- Tổ chức và làm việc nhóm thường xuyên thông qua Google Meet(1-2 lần/tuần).

Chưa làm được:

- Các yêu cầu đề tài còn có một số sai sót nhất định.
- Chỉ tổ chức và làm việc nhóm thông qua Google Meet gây ra một số khó khăn và sai sót trong đề tài.

3.4.2 Hướng phát triển

Sau khi hoàn thành các yêu cầu đề tài, tiếp theo nhóm chúng em sẽ hiện thực hệ thống thực đúng theo sát các yêu cầu đã trình bày. Trong quá trình hiện thực hệ thống chúng em sẽ tiếp tục chỉnh sửa, hoàn thiện lại các yêu cầu đề tài đã làm.

4 Giai đoạn 2 (Phát triển ứng dụng)

4.1 Viết các Functions/Procedures/Assertions/Triggers.

Trong bài nộp đính kèm gồm có 3 file sql:

1. Procedure.sql
2. Trigger.sql
3. Function.sql

Để tạo cơ sở dữ liệu hoàn chỉnh, ta tiến hành nạp code ở các file cùng với file main.sql và init.sql đầu vào.

4.1.1 Các CREATE Procrdures

1. Khách hàng có thể xem thông tin hỗ trợ cho thành viên của chuỗi siêu thị.

```
DROP PROCEDURE IF EXISTS get_NontificationSuper;  
DELIMITER $$  
CREATE PROCEDURE get_NontificationSuper()  
BEGIN  
    SELECT Ngay_thong_bao, Noi_dung  
    FROM thong_bao_thong_tin  
    WHERE Nguoi_thong_bao = 'Employee';  
END;
```

2. Hệ thống đăng kí thành viên xác nhận khách hàng là thành viên chính thức và gửi thông báo cho khách hàng khi tài khoản được tạo thành công.

```
DROP PROCEDURE IF EXISTS insert_Customer;  
DELIMITER $$  
CREATE PROCEDURE insert_Customer(  
    KHHoTenDem varchar(50),  
    KHTen varchar(20),  
    DiemTL int,  
    GioiTinh varchar(1),  
    NamSinh smallint,  
    SDT varchar(10),
```


MaTN varchar(10))

BEGIN

INSERT INTO KHACHHANG

VALUES (KHHoTenDem, KHTen, DiemTL, GioiTinh, NamSinh, SDT);

INSERT INTO DANGKY

VALUES (MaTN, SDT);

SELECT Noi_dung

FROM thong_bao_thong_tin

WHERE Ma_thong_bao = '001';

END;

3. **Hệ thống cho phép khách hàng xem thông tin của mình.**

DROP PROCEDURE IF EXISTS get_InfoCustomer;

DELIMITER \$\$

CREATE PROCEDURE get_InfoCustomer(

SDT varchar(10)

)

BEGIN

SELECT *

FROM KHACHHANG

WHERE KHACHHANG.SDT = SDT;

END;

4. **Khách hàng có thể chỉnh sửa thông tin của mình.**

DROP PROCEDURE IF EXISTS tt_khachhang;

DELIMITER \$\$

CREATE PROCEDURE tt_khachhang(

KHHoTenDem varchar(50),

KHTen varchar(20),

GioiTinh varchar(1),

NamSinh smallint,

SDT varchar(10))

BEGIN

UPDATE KHACHHANG

SET KHHoTenDem = KHHoTenDem, KHTen = KHTen, GioiTinh = GioiTinh, NamSinh = NamSinh

WHERE KHACHHANG.SDT= SDT;

END;

5. Khách hàng có thể xem lịch sử giao dịch của mình.

DROP PROCEDURE IF EXISTS get_TransactionHistory;

DELIMITER \$\$

CREATE PROCEDURE get_TransactionHistory(
 SDT varchar(10)

)

BEGIN

SELECT *

FROM DONHANG

WHERE KH_SDT = SDT;

END;

6. Hệ thống gửi nội dung thống kê giao dịch hàng tháng cho khách hàng.

DROP PROCEDURE IF EXISTS get_TransactionStatisticsMonth;

DELIMITER \$\$

CREATE PROCEDURE get_TransactionStatisticsMonth(
 SDT varchar(10),

 Required_Month int,

 Required_Year int

)

BEGIN

SELECT *

FROM DONHANG

WHERE KH_SDT = SDT **AND** MONTH(Ngay) = Required_Month **AND**

 YEAR(Ngay) = Required_Year;

END ;

7. Khách hàng có thể gửi câu hỏi về chương trình đăng kí thành viên cho hệ thống.

DROP PROCEDURE IF EXISTS cauhoi;

DELIMITER \$\$

CREATE PROCEDURE cauhoi(
 Ma_thong_bao varchar(30),

 Ngay_thong_bao date,

 Noi_dung varchar(255)

```
)  
BEGIN  
    INSERT INTO thông_bao_thông_tin  
    VALUES (Ma_thông_bao, Ngay_thông_bao, Noi_dung, 'Customer');  
END;
```

4.1.2 Các CREATE TRIGGER

1. Trigger - Kiểm tra khách hàng trên 16 tuổi khi thêm mới

```
DROP TRIGGER IF EXISTS check_AgeOfCustomer;  
DELIMITER $$  
CREATE TRIGGER check_AgeOfCustomer  
    BEFORE INSERT  
    ON KHACHHANG FOR EACH ROW  
BEGIN  
    DECLARE year_now int;  
    SELECT YEAR(NOW()) INTO year_now;  
    IF(year_now - NEW.Namsinh < 16)  
        THEN SIGNAL SQLSTATE '45000'  
        SET MESSAGE_TEXT = 'Please enter age of customer is greater  
than 16';  
    END IF;  
END;
```

2. Trigger - Kiểm tra khách hàng trên 16 tuổi khi chỉnh sửa

```
DROP TRIGGER IF EXISTS check_Update_AgeOfCustomer;  
DELIMITER $$  
CREATE TRIGGER check_Update_AgeOfCustomer  
    BEFORE UPDATE  
    ON KHACHHANG FOR EACH ROW  
BEGIN  
    DECLARE year_now int;  
    SELECT YEAR(NOW()) INTO year_now;  
    IF(year_now - NEW.NamSinh < 16)  
        THEN SIGNAL SQLSTATE '45000'  
        SET MESSAGE_TEXT = 'Please enter age of customer is greater
```

than 16';

END IF;

END;

3. Trigger - Kiểm tra nhân viên trên 18 tuổi khi thêm mới

DROP TRIGGER IF EXISTS check_AgeOfEmployee;

DELIMITER \$\$

CREATE TRIGGER check_AgeOfEmployee

BEFORE INSERT

ON NHANVIEN FOR EACH ROW

BEGIN

DECLARE year_now int;

SELECT YEAR(NOW()) INTO year_now;

IF(year_now - NEW.NamSinh < 18)

THEN SIGNAL SQLSTATE '45000'

SET MESSAGE_TEXT = 'Please enter age of employee is greater

than 18';

END IF;

END;

4. Trigger - Kiểm tra nhân viên trên 18 tuổi khi chỉnh sửa

DROP TRIGGER IF EXISTS check_Update_AgeOfEmployee;

DELIMITER \$\$

CREATE TRIGGER check_Update_AgeOfEmployee

BEFORE UPDATE

ON NHANVIEN FOR EACH ROW

BEGIN

DECLARE year_now int;

SELECT YEAR(NOW()) INTO year_now;

IF(year_now - NEW.NamSinh < 18)

THEN SIGNAL SQLSTATE '45000'

SET MESSAGE_TEXT = 'Please enter age of employee is greater

than 18';

END IF;

END;

5. Trigger - Kiểm tra nhân viên làm việc ít nhất 4 giờ khi thêm mới

```
DROP TRIGGER IF EXISTS check_HourOfEmployee;  
DELIMITER $$  
CREATE TRIGGER check_HourOfEmployee  
    BEFORE INSERT  
    ON NHANVIEN FOR EACH ROW  
BEGIN  
    IF(NEW.ThoiGianLam < 4)  
        THEN SIGNAL SQLSTATE '45000'  
        SET MESSAGE_TEXT = 'Please enter the working time of employee  
is greater than 4';  
    END IF;  
END;
```

6. Trigger - Kiểm tra nhân viên làm việc ít nhất 4 giờ khi chỉnh sửa

```
DROP TRIGGER IF EXISTS check_Update_HourOfEmployee;  
DELIMITER $$  
CREATE TRIGGER check_Update_HourOfEmployee  
    BEFORE UPDATE  
    ON NHANVIEN FOR EACH ROW  
BEGIN  
    IF(NEW.ThoiGianLam < 4)  
        THEN SIGNAL SQLSTATE '45000'  
        SET MESSAGE_TEXT = 'Please enter the working time of employee  
is greater than 4';  
    END IF;  
END;
```

7. Trigger - Cập nhật điểm thưởng

```
DROP TRIGGER IF EXISTS update_CusPoint;  
DELIMITER $$  
CREATE TRIGGER update_CusPoint  
    AFTER INSERT  
    ON DONHANG FOR EACH ROW  
BEGIN  
    DECLARE DiemThuong int;  
    DECLARE SDT varchar(10);  
    DECLARE oldPoint int;
```

```
SELECT DiemThuong INTO DiemThuong FROM INSERTED;  
SELECT KH_SDT INTO SDT FROM INSERTED;  
SELECT DiemTL INTO oldPoint FROM KHACHHANG  
WHERE KHACHHANG.SDT = SDT;  
UPDATE KHACHHANG  
SET DiemTL = oldPoint + DiemThuong  
WHERE KHACHHANG.SDT = SDT;  
END;
```

4.1.3 Các CREATE FUNCTION

1. Function - Kiểm tra quán cà phê có ít nhất 4 nhân viên

```
DROP FUNCTION IF EXISTS Check_NoEmployee;  
DELIMITER $$  
CREATE FUNCTION Check_NoEmployee(  
    MaCN  varchar(10)  
)  
RETURNS BOOLEAN  
DETERMINISTIC  
BEGIN  
    DECLARE no_employee int;  
    SELECT count(*) INTO no_employee FROM NHANVIEN  
    WHERE NHANVIEN.MaCN = MaCN;  
    IF(no_employee >= 4) THEN  
        RETURN TRUE;  
    ELSE RETURN FALSE;  
    END IF;  
END;
```

2. Function - Kiểm tra khách hàng có ít nhất 1 giao dịch trong vòng 1 năm

```
DROP FUNCTION IF EXISTS Check_TradeOfCustomer;  
DELIMITER $$  
CREATE FUNCTION Check_TradeOfCustomer(  
    SDT  varchar(10)  
)  
RETURNS BOOLEAN  
DETERMINISTIC
```

BEGIN

```
    DECLARE LastYearDate date;
    DECLARE CusLastTrade date;
    SELECT CURRENT_DATE() - 365 INTO LastYearDate;
    SELECT max(Ngay) INTO CusLastTrade FROM DONHANG
    WHERE KH_SDT = SDT;
    IF (CusLastDate > LastYearDate) THEN
        RETURN TRUE;
    ELSE RETURN FALSE;
    END IF;
END;
```

4.2 Định nghĩa các nhóm người dùng và phân quyền ở mức ứng dụng

Ứng dụng được thiết kế với 3 nhóm người dùng:

- khách hàng: khi đăng nhập vào ứng dụng với tư cách là khách hàng, khách hàng có thể xem thông tin cá nhân, xem đơn hàng, gửi các câu hỏi thắc mắc để nhân viên có thể trả lời.
- Nhân viên: khi đăng nhập với tư cách là nhân viên, nhân viên có thể đăng ký tài khoản thành viên cho khách hàng mới, ngoài ra nhân viên có thể xem và trả lời các câu hỏi mà khách hàng đã gửi.
- Quản lý: Quản lý có thể xem tổng số nhân viên, Chỉnh sửa thông tin khách hàng.

4.3 Kiến trúc hệ thống

Client

- Một SPA (Single-page application) đơn giản được làm bằng React với giao diện trực quan, dễ sử dụng và quản lý đối với mọi nhóm đối tượng sử dụng.
- Phía **Client** xem và cập nhật thông tin bằng cách gửi request thông qua các API được cung cấp bởi **Server** (Cả **Server** và **Client** đều được host trên ở cùng một nơi).
- Với mỗi request đính kèm một Authorization header dùng để phân biệt **Client** và bảo mật API khỏi các request lạ, **Client** tạo header này bằng cách gửi

request kèm thông tin đăng nhập vào **route** `/login`, sau khi **Server** đúng thông tin sẽ trả về một **access_token** tồn tại trong khoảng 30 phút, và **Client** sẽ đính kèm **access_token** này vào phần header để không phải đăng nhập lần nữa.

- **Client** gửi request dưới dạng **Ajax** request thông qua package **Axios**

Server

- Một hệ thống đơn giản gồm các API trả về kết quả dưới dạng JSON được xây dựng bằng **NodeJS**
- **Server** tương tác với **MySQL database** thông qua package **mysql** và sử dụng những **procedure** đã được định nghĩa sẵn để cho đơn giản hóa việc thực hiện và giúp code thân thiện hơn.
- **Server** sẽ chỉ chấp nhận những request có Authorization header đã được quy định, đối với những request khác **Server** '401: Unauthorized' cho những request khác
- Tùy vào mỗi request **Server** sẽ tương tác với bên thứ ba hoặc với **Database** để trả về data thích hợp.

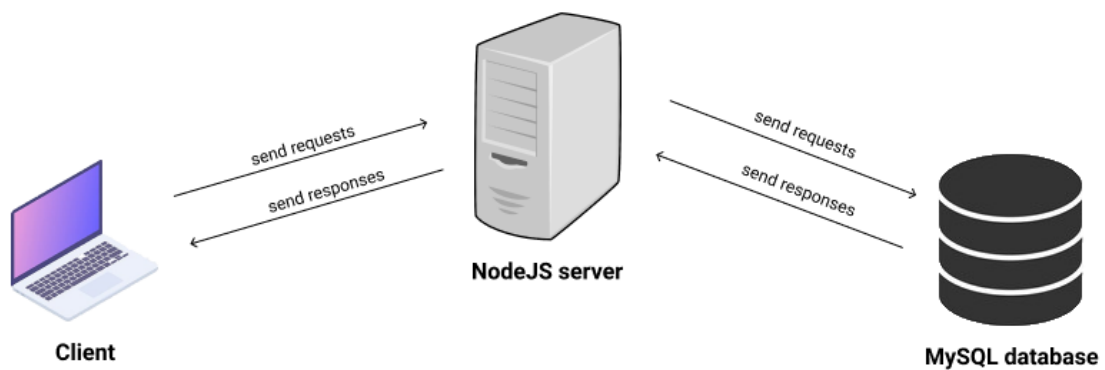
Database

- Sử dụng **MySQL database** với việc tạo các **table** và gửi các **query** được hiện thực bằng package **mysql** ở phía backend.
- Mỗi **route** của **Server** sẽ tương ứng với một **query** được gửi tới **MySQL** và tùy vào kiểu của HTTP request (GET, POST, ...) sẽ tương ứng với các **query** SELECT, INSERT, UPDATE ...
- Các **procedure** tùy theo chức năng sẽ có các prefix như **get_** hoặc **insert_** để dễ phân biệt.

Ghi chú:

API: viết tắt bởi Application Programming Interface , tức giao diện lập trình ứng dụng. Đây là phương tiện cho hai hoặc nhiều ứng dụng trao đổi, tương tác với nhau, tạo ra tương tác giữa người dùng với ứng dụng hiệu quả và tiện lợi hơn.

Authorization: API sử dụng Authorization để đảm bảo rằng người dùng truy cập dữ liệu một cách an toàn. Với Authorization header, thông tin để xác thực người dùng được gửi kèm trong header.



Hình 4.1: Mô hình Client - Server

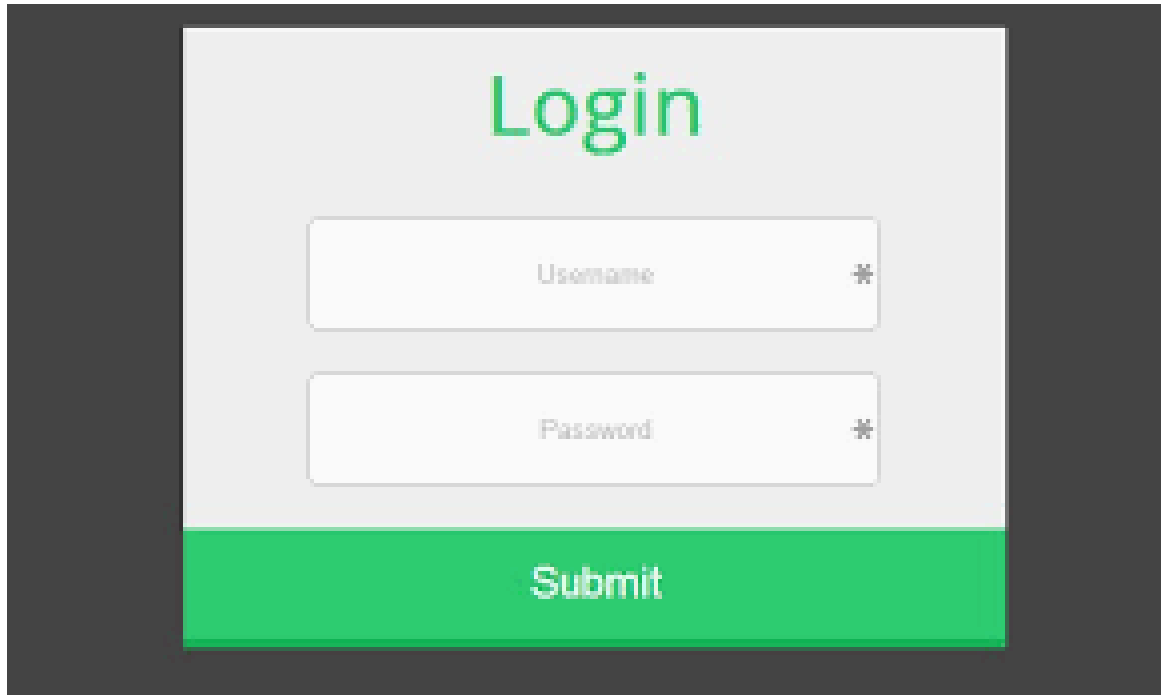
401 Unauthorized: là một mã trạng thái http (http code 401). Mã trạng thái http là các số nguyên có ba chữ số. Số đầu tiên được dùng để xác định mã nằm trong 5 danh mục là “thông tin”, “thành công”, “chuyển hướng”, “lỗi từ client chỉ ra rằng không thể hoàn thành hoặc sai cú pháp”, “lỗi từ server”.

Query language (Ngôn ngữ truy vấn): được sử dụng để tạo query trong cơ sở dữ liệu và Query language của Microsoft (SQL) là tiêu chuẩn.

4.4 Thiết kế chi tiết cho ứng dụng.

Thiết kế 3 trang đại diện cho 3 chức năng chính của ứng dụng: đăng nhập, đăng ký thành viên, thay đổi thông tin khách hàng.

Trang login

The image shows a login form with a light gray background. At the top, the word "Login" is written in a large, green, sans-serif font. Below it, there are two white input fields with rounded corners. The first field is labeled "Username" in a small, gray font, and the second field is labeled "Password" in a small, gray font. Both fields have a small asterisk icon on the right side. Below the input fields, there is a large, green rectangular button with the word "Submit" written in a white, sans-serif font. The entire form is centered on a dark gray background.

Hình 4.2: Trang login

Trang đăng ký thành viên

Đăng ký khách hàng

Username	
<input type="text" value="nhanvien"/>	
Password	
<input type="password" value="....."/>	
Họ và tên lót	Tên
<input type="text" value="Nguyễn Văn"/>	<input type="text" value="A"/>
Năm sinh	Giới tính
<input type="text"/>	<input type="text" value="Nam"/> ▼
Địa chỉ	
<input type="text"/>	

Hình 4.3: Trang đăng ký thành viên

Trang chỉnh sửa thông tin

CHỈNH SỬA THÔNG TIN KHÁCH HÀNG

Chọn khách hàng

Vo Thanh Hieu



Họ và tên lót

Tên

Năm sinh

Giới tính

Nam



Địa chỉ

Số CMND/ CCCD

Cập nhật

Hình 4.4: Trang chỉnh sửa thông tin

4.5 Chuẩn hóa về dạng chuẩn BCNF cho lược đồ cơ sở dữ liệu.

- **KHACHHANG** (SDT, KHHoTenDem, KHTen, DiemTL, GioiTinh, NamSinh)

Khóa chính: {SDT}

FD: $SDT \rightarrow \{KHHoTenDem, KHTen, DiemTL, GioiTinh, NamSinh\}$

Lược đồ quan hệ này thỏa **1NF** vì chỉ có một giá trị tại mỗi giao điểm giữa hàng và cột, không có thuộc tính đa trị, thuộc tính phức và quan hệ lồng nhau;

Lược đồ quan hệ này thỏa **2NF** vì tất cả các thuộc tính đều phụ thuộc toàn phần vào khóa chính;

Lược đồ quan hệ này thỏa **3NF** vì nó thỏa **2NF** và mỗi thuộc tính đều phụ thuộc trực tiếp vào khóa chính;

Lược đồ quan hệ này thỏa **BCNF** vì mỗi FD $X \rightarrow A$ thì X là siêu khóa của lược đồ.

- **QUAN** (MaCN, DiaChi)

Khóa chính: {MaCN}

FD: $MaCN \rightarrow \{DiaChi\}$

Lược đồ quan hệ này thỏa **1NF** vì chỉ có một giá trị tại mỗi giao điểm giữa hàng và cột, không có thuộc tính đa trị, thuộc tính phức và quan hệ lồng nhau;

Lược đồ quan hệ này thỏa **2NF** vì tất cả các thuộc tính đều phụ thuộc toàn phần vào khóa chính;

Lược đồ quan hệ này thỏa **3NF** vì nó thỏa **2NF** và mỗi thuộc tính đều phụ thuộc trực tiếp vào khóa chính;

Lược đồ quan hệ này thỏa **BCNF** vì mỗi FD $X \rightarrow A$ thì X là siêu khóa của lược đồ.

- **DONHANG** (MaDon, DiemThuong, Ngay, MaTN, HinhThuc, KH_SDT)

Khóa chính: {MaDon}

FD: $MaDon \rightarrow \{DiemThuong, Ngay, MaTN, HinhThuc, KH_SDT\}$

Lược đồ quan hệ này thỏa **1NF** vì chỉ có một giá trị tại mỗi giao điểm giữa hàng và cột, không có thuộc tính đa trị, thuộc tính phức và quan hệ lồng nhau;

Lược đồ quan hệ này thỏa **2NF** vì tất cả các thuộc tính đều phụ thuộc toàn phần vào khóa chính;

Lược đồ quan hệ này thỏa **3NF** vì nó thỏa **2NF** và mỗi thuộc tính đều phụ thuộc trực tiếp vào khóa chính;

Lược đồ quan hệ này thỏa **BCNF** vì mỗi FD $X \rightarrow A$ thì X là siêu khóa của lược đồ.

- **THUCPHAM** (MaHH, Ten, Gia, HSD, NSX, KL, DT)

Khóa chính: {MaHH}

FD: $\text{MaHH} \rightarrow \{\text{Ten, Gia, HSD, NSX, KL, DT}\}$

Lược đồ quan hệ này thỏa **1NF** vì chỉ có một giá trị tại mỗi giao điểm giữa hàng và cột, không có thuộc tính đa trị, thuộc tính phức và quan hệ lồng nhau;

Lược đồ quan hệ này thỏa **2NF** vì tất cả các thuộc tính đều phụ thuộc toàn phần vào khóa chính;

Lược đồ quan hệ này thỏa **3NF** vì nó thỏa **2NF** và mỗi thuộc tính đều phụ thuộc trực tiếp vào khóa chính;

Lược đồ quan hệ này thỏa **BCNF** vì mỗi FD $X \rightarrow A$ thì X là siêu khóa của lược đồ.

- **NHANVIEN** (MaNV, GioiTinh, NVHoTenDem, NVTen, ThoiGianLam, NamSinh, TaiKhoanNH, KieuNV, LuongThang, LuongGio, MaCN, MaNV, CCCD)

Khóa chính: {MaNV}

FD: $\{\text{MaNV}\} \rightarrow \{\text{GioiTinh, NVHoTenDem, NVTen, ThoiGianLam, NamSinh, TaiKhoanNH, KieuNV, LuongThang, LuongGio, MaCN, MaNV, CCCD}\}$

Lược đồ quan hệ này thỏa **1NF** vì chỉ có một giá trị tại mỗi giao điểm giữa hàng và cột, không có thuộc tính đa trị, thuộc tính phức và quan hệ lồng nhau;

Lược đồ quan hệ này thỏa **2NF** vì tất cả các thuộc tính đều phụ thuộc toàn phần vào khóa chính;

Lược đồ quan hệ này thỏa **3NF** vì nó thỏa **2NF** và mỗi thuộc tính đều phụ thuộc trực tiếp vào khóa chính;

Lược đồ quan hệ này thỏa **BCNF** vì mỗi FD $X \rightarrow A$ thì X là siêu khóa của lược đồ.

- **NV_SDT** (MaNV, NV_SDT)

Khóa chính: {MaNV, NV_SDT}

FD: Không có

Lược đồ quan hệ này thỏa **BCNF** vì chỉ có khóa và không có FD.

- **KHUYENMAI** (MaDon, MaKM, HSD, NoiDung, GiamGia)

Khóa chính: {MaDon, MaKM}

FD: {MaDon} \rightarrow {HSD, NoiDung, GiamGia}

Lược đồ quan hệ này thỏa **1NF** vì chỉ có một giá trị tại mỗi giao điểm giữa hàng và cột, không có thuộc tính đa trị, thuộc tính phức và quan hệ lồng nhau;

Lược đồ quan hệ này thỏa **2NF** vì tất cả các thuộc tính đều phụ thuộc toàn phần vào khóa chính;

Lược đồ quan hệ này thỏa **3NF** vì nó thỏa **2NF** và mỗi thuộc tính đều phụ thuộc trực tiếp vào khóa chính;

Lược đồ quan hệ này thỏa **BCNF** vì mỗi FD $X \rightarrow A$ thì X là siêu khóa của lược đồ.

- **THUNGAN** (MaTN, SoGiaoDich, SoTienGiaodich)

Khóa chính: {MaTN}

FD: {MaTN} \rightarrow {SoGiaoDich, SoTienGiaodich}

Lược đồ quan hệ này thỏa **1NF** vì chỉ có một giá trị tại mỗi giao điểm giữa hàng và cột, không có thuộc tính đa trị, thuộc tính phức và quan hệ lồng nhau;

Lược đồ quan hệ này thỏa **2NF** vì tất cả các thuộc tính đều phụ thuộc toàn phần vào khóa chính;

Lược đồ quan hệ này thỏa **3NF** vì nó thỏa **2NF** và mỗi thuộc tính đều phụ thuộc trực tiếp vào khóa chính;

Lược đồ quan hệ này thỏa **BCNF** vì mỗi FD $X \rightarrow A$ thì X là siêu khóa của lược đồ.

- **QUANLY** (MaQL)

Khóa chính: {MaQL}

FD: Không có Lược đồ quan hệ này thỏa **BCNF** vì chỉ có khóa và không có FD.

- **QUANLY_CHUNGCHI** (MaQL, ChungChi)

Khóa chính: {MaQL, ChungChi}

FD: Không có

Lược đồ quan hệ này thỏa **BCNF** vì chỉ có khóa và không có FD.

- **DANGKY** (MaTN, KH_SDT)

Khóa chính: {MaTN, KH_SDT}

FD: Không có Lược đồ quan hệ này thỏa **BCNF** vì chỉ có khóa và không có FD.

- **TP_MOTA** (MaHH, MoTa)

Khóa chính: {MaHH, MoTa}

FD: Không có

Lược đồ quan hệ này thỏa **BCNF** vì chỉ có khóa và không có FD.

- **BAOGOM** (MaDon, MaHH, SoLuong)

Khóa chính: {MaDon, MaHH}

FD: {MaDon, MaHH} → {SoLuong}

Lược đồ quan hệ này thỏa **1NF** vì chỉ có một giá trị tại mỗi giao điểm giữa hàng và cột, không có thuộc tính đa trị, thuộc tính phức và quan hệ lồng nhau;

Lược đồ quan hệ này thỏa **2NF** vì tất cả các thuộc tính đều phụ thuộc toàn phần vào khóa chính;

Lược đồ quan hệ này thỏa **3NF** vì nó thỏa **2NF** và mỗi thuộc tính đều phụ thuộc trực tiếp vào khóa chính;

Lược đồ quan hệ này thỏa **BCNF** vì mỗi FD $X \rightarrow A$ thì X là siêu khóa của lược đồ.

- **USER** (id, username, password, role, associated_id)

Khóa chính: {id}

FD: {id} → {username, password, role, associated_id}

Lược đồ quan hệ này thỏa **1NF** vì chỉ có một giá trị tại mỗi giao điểm giữa hàng và cột, không có thuộc tính đa trị, thuộc tính phức và quan hệ lồng nhau;

Lược đồ quan hệ này thỏa **2NF** vì tất cả các thuộc tính đều phụ thuộc toàn phần vào khóa chính;

Lược đồ quan hệ này thỏa **3NF** vì nó thỏa **2NF** và mỗi thuộc tính đều phụ thuộc trực tiếp vào khóa chính;

Lược đồ quan hệ này thỏa **BCNF** vì mỗi FD $X \rightarrow A$ thì X là siêu khóa của lược đồ.

- **thong_bao_thong_tin** (Ma_thong_bao, Ngay_thong_bao, Noi_dung, Nguoi_thong_bao)

Khóa chính: {Ma_thong_bao}

FD: {Ma_thong_bao} \rightarrow {Ngay_thong_bao, Noi_dung, Nguoi_thong_bao}

Lược đồ quan hệ này thỏa **1NF** vì chỉ có một giá trị tại mỗi giao điểm giữa hàng và cột, không có thuộc tính đa trị, thuộc tính phức và quan hệ lồng nhau;

Lược đồ quan hệ này thỏa **2NF** vì tất cả các thuộc tính đều phụ thuộc toàn phần vào khóa chính;

Lược đồ quan hệ này thỏa **3NF** vì nó thỏa **2NF** và mỗi thuộc tính đều phụ thuộc trực tiếp vào khóa chính;

Lược đồ quan hệ này thỏa **BCNF** vì mỗi FD $X \rightarrow A$ thì X là siêu khóa của lược đồ.

Kết luận: Lược đồ CSDL trên thỏa **BCNF** vì tất cả lược đồ quan hệ trong lược đồ CSDL đều thỏa **BCNF**.

4.6 Chỉ mục

- Cho table user column 'id':

create index user_id on user(id);

- Cho table nhân viên column 'ma_nhan_vien':

create index employee_id on nhan_vien_sieu_thi(Ma_nhan_vien);

- Cho table `khach_hang` column 'Ma_khach_hang':
`create index customer_id on khach_hang(Ma_khach_hang);`
- Cho table `hang_hoa` column 'Ma_hang_hoa':
`create index item_id on hang_hoa(Ma_hang_hoa);`

Bảo mật:

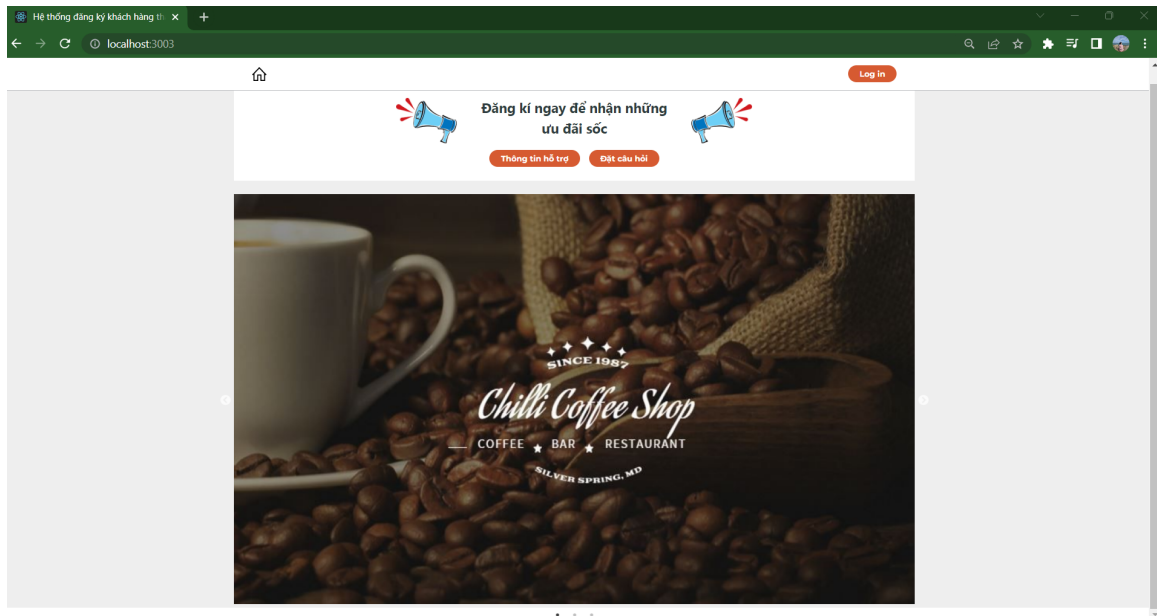
Mỗi user sẽ có một account bao gồm username và password, password sẽ được encrypt bởi bcrypt. Server sẽ kiểm tra password thông qua function `hash()` của bcrypt.

API của hệ thống:

- Về phía USER:
 - **POST** `/api/users/login`: Để login vào hệ thống
 - **GET** `/api/users/role`: Để lấy role của người dùng
 - **GET** `/api/users/info`:
 - **GET** `/api/users`
 - **GET** `/api/users/:id`
 - **POST** `/api/users/`
 - **PUT** `/api/users/:id`
- Đối với 'câu hỏi':
 - **POST** `/api/questions/`
 - **GET** `/api/questions/`
 - **PUT** `/api/questions/:question_id/answer`
- Đối với 'đơn hàng':
 - **GET** `/api/orders`
 - **GET** `/api/orders/:id`

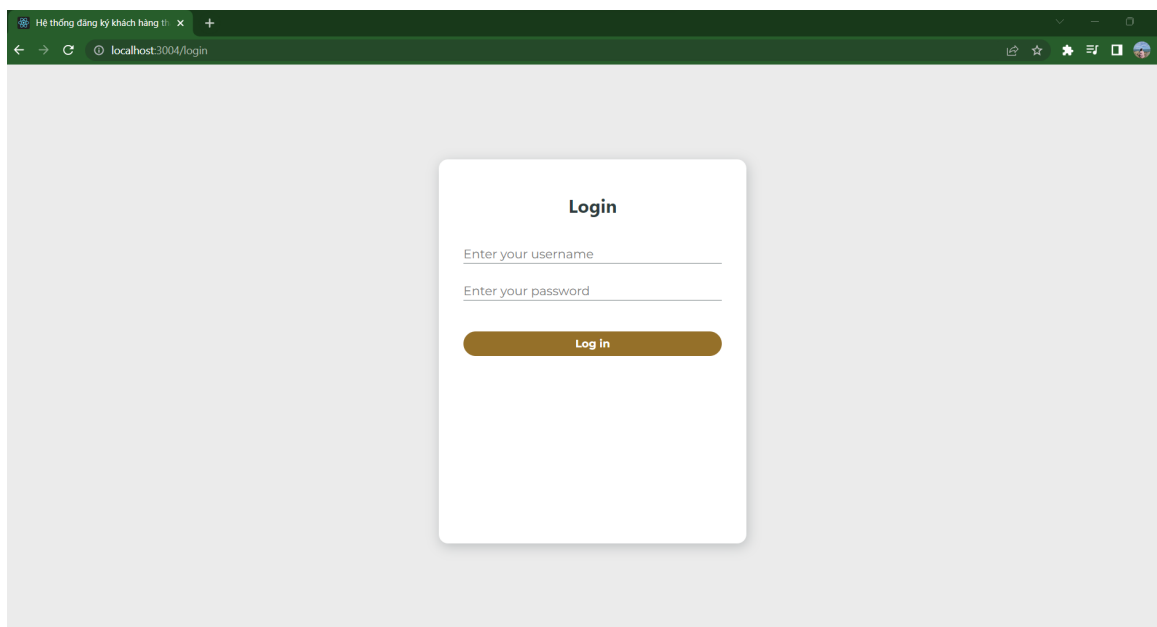
4.7 Triển khai thiết kế và hiện thực phần mềm

Trang chủ của trang Web sau khi truy cập



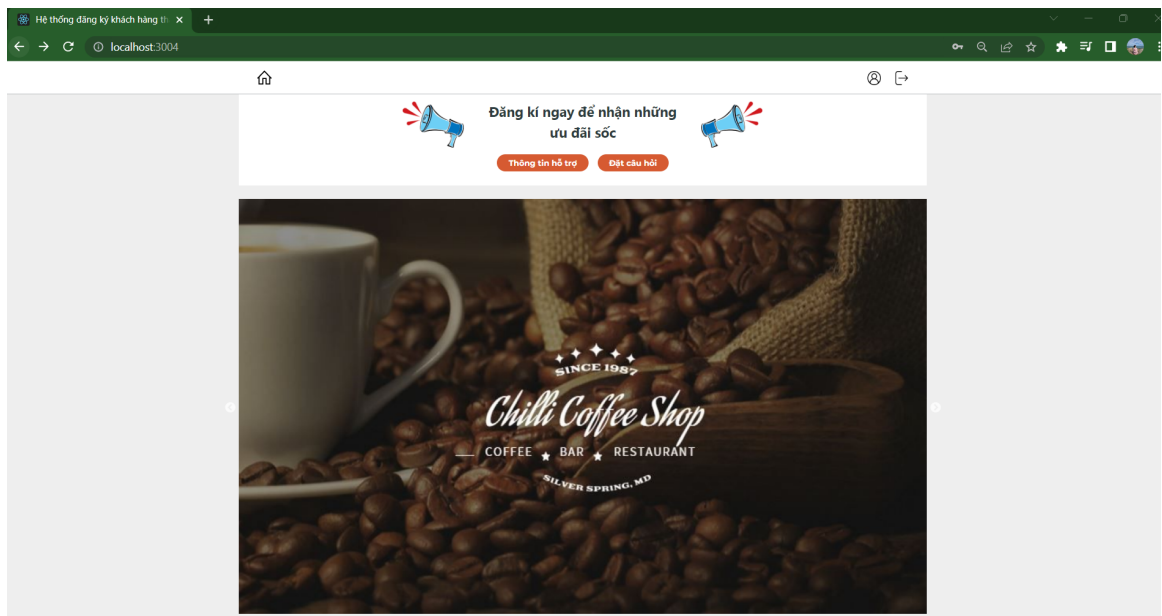
Hình 4.5: Trang chủ

Trang đăng nhập của Web



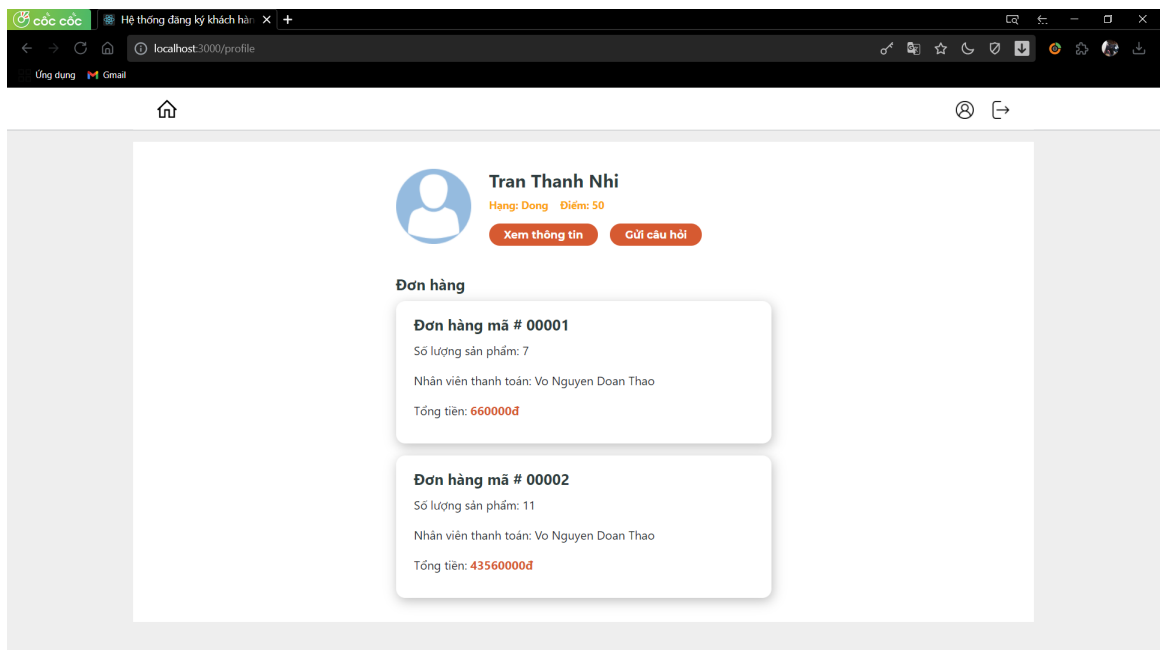
Hình 4.6: Trang đăng nhập

Trang hiển thị sau khi khách hàng đăng nhập



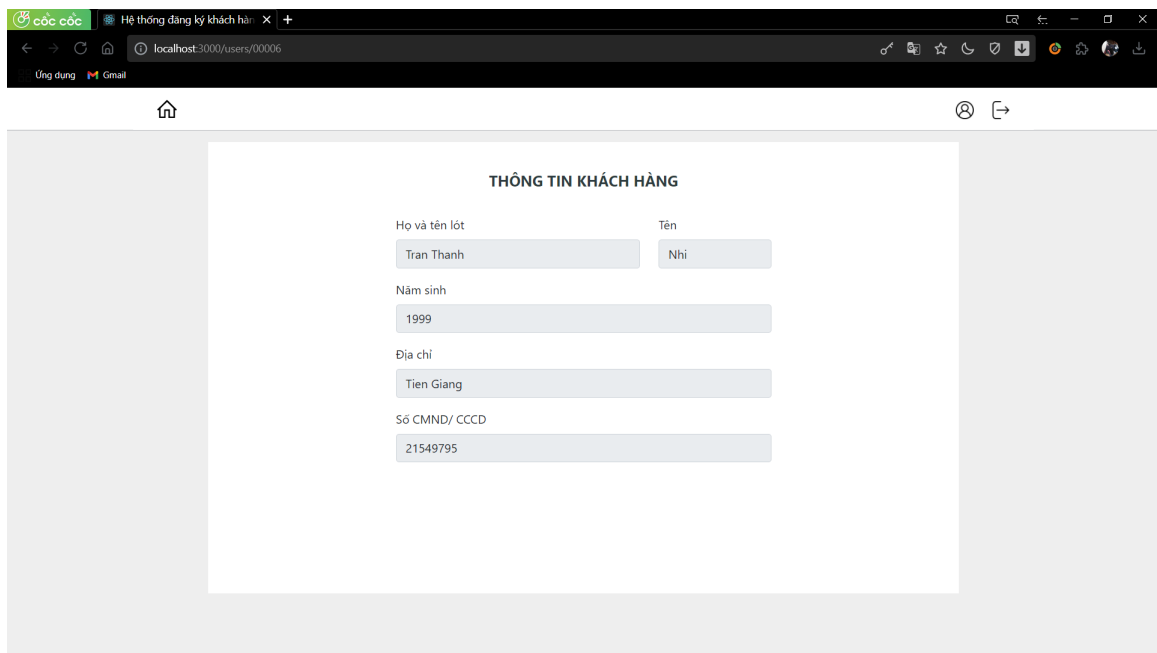
Hình 4.7: Trang chủ của khách hàng

Trang cá nhân của khách hàng



Hình 4.8: Trang cá nhân khách hàng

Trang thông tin cá nhân khách hàng



THÔNG TIN KHÁCH HÀNG

Họ và tên lót: Tran Thanh

Tên: Nhi

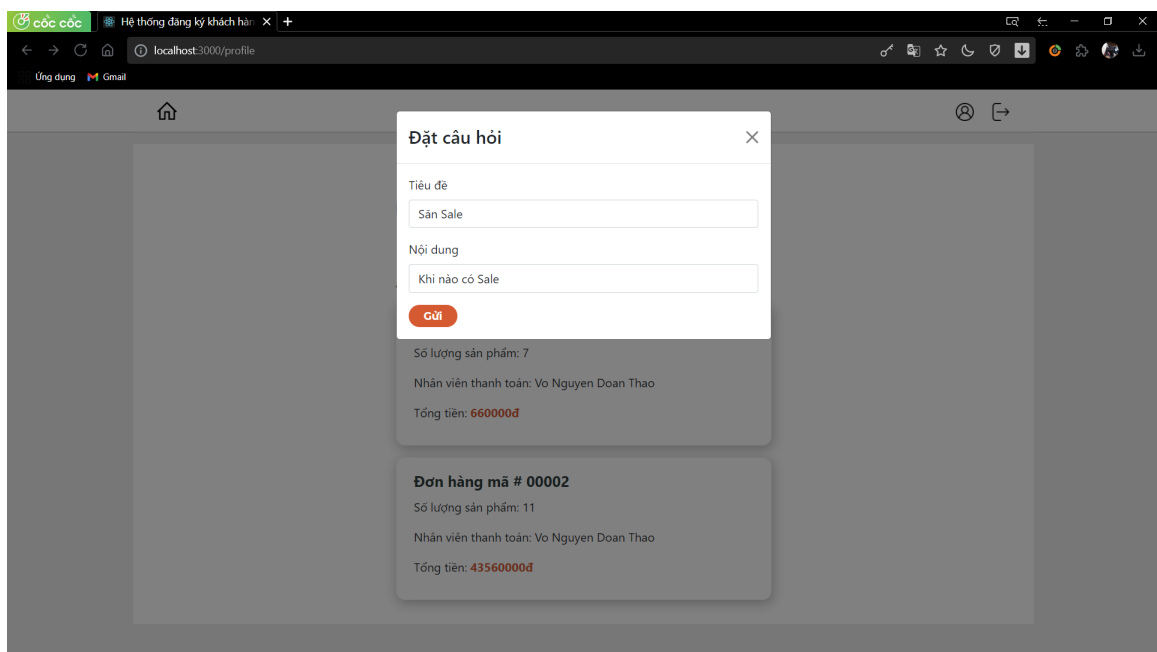
Năm sinh: 1999

Địa chỉ: Tien Giang

Số CMND/ CCCD: 21549795

Hình 4.9: Trang thông tin cá nhân khách hàng

Trang đặt câu hỏi của khách hàng



Đặt câu hỏi

Tiêu đề: Săn Sale

Nội dung: Khi nào có Sale

Gửi

Số lượng sản phẩm: 7

Nhân viên thanh toán: Vo Nguyen Doan Thao

Tổng tiền: 660000đ

Đơn hàng mã # 00002

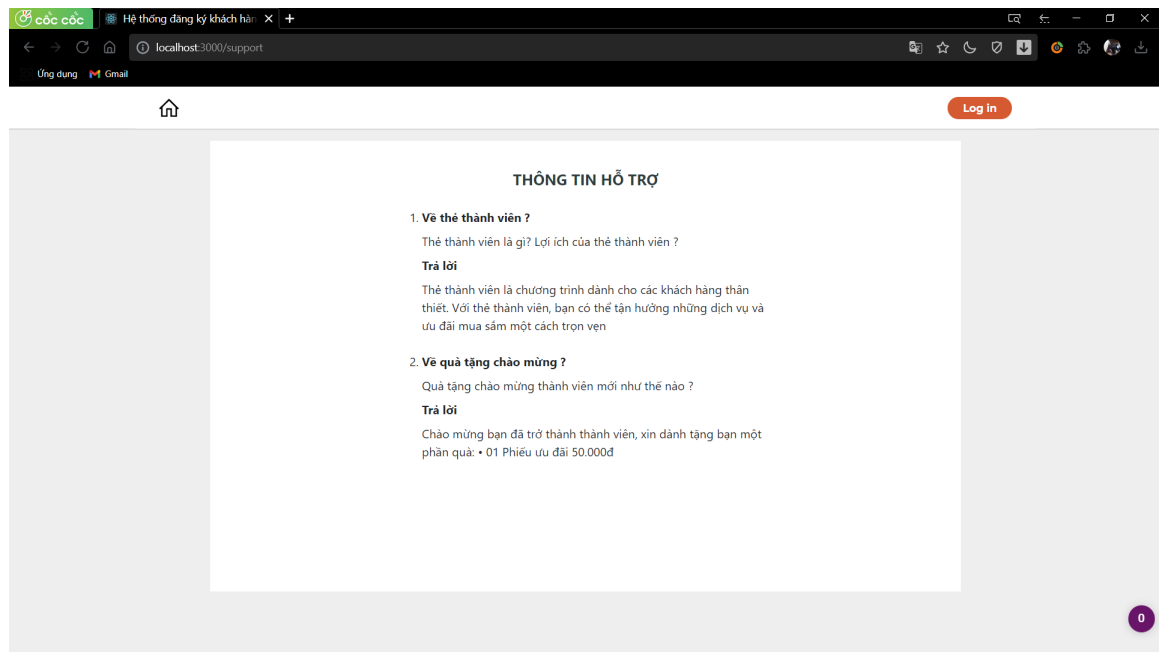
Số lượng sản phẩm: 11

Nhân viên thanh toán: Vo Nguyen Doan Thao

Tổng tiền: 43560000đ

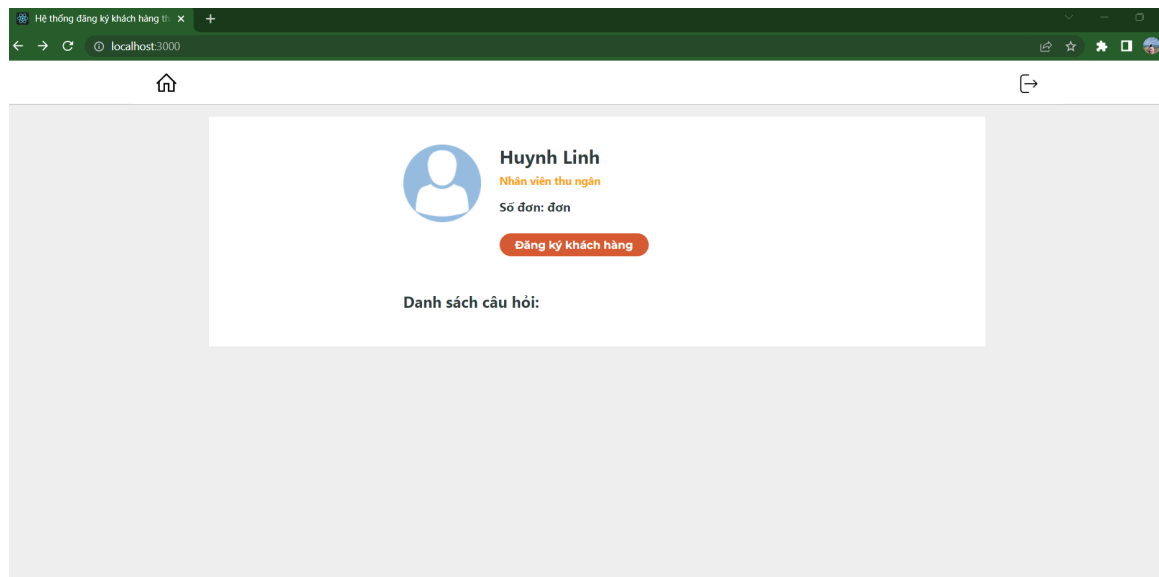
Hình 4.10: Trang đặt câu hỏi

Trang thông tin hỗ trợ khách hàng



Hình 4.11: Trang thông tin hỗ trợ

Trang cá nhân sau khi nhân viên đăng nhập



Hình 4.12: Trang cá nhân của nhân viên

Trang trả lời câu hỏi của nhân viên

Trả lời câu hỏi

Câu trả lời

Mua hàng

Trả lời

Làm sao để tôi đăng nhập vào tài khoản ?

Trả lời

Cách để lên nhóm Vàng ?

Ấn danh

Cho tôi hỏi về các yêu cầu để lên nhóm Vàng ?

Trả lời

Săn Sale

Tran Thanh Nhi

Săn Sale

Trả lời

Hình 4.13: Trang trả lời câu hỏi

Trang đăng kí thành viên

Đăng ký khách hàng

Username

Password

Họ và tên lót

Tên

Năm sinh

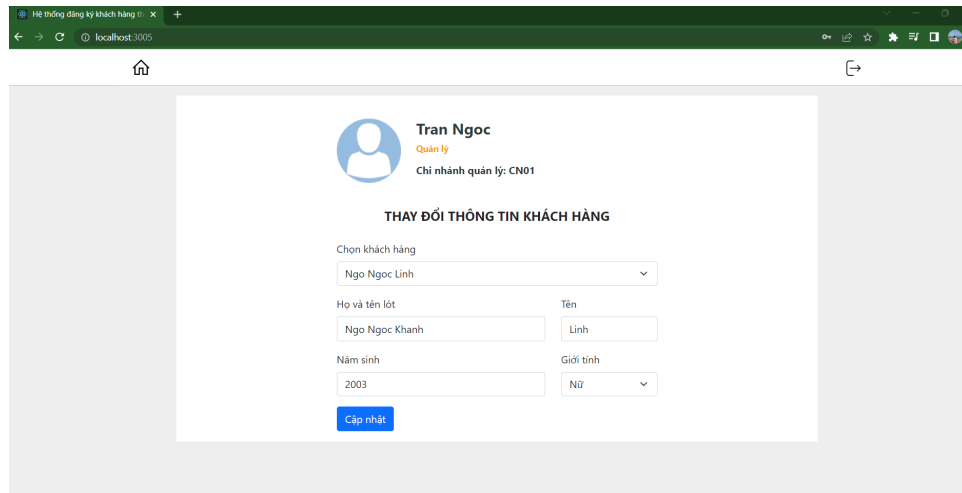
Giới tính

Số điện thoại

Đăng ký

Hình 4.14: Trang đăng kí thành viên

Trang cá nhân của quản lý: quản lý có thể chỉnh sửa thông tin khách hàng



Hệ thống đăng ký khách hàng

Tran Ngoc
Quản lý
Chi nhánh quản lý: CN01

THAY ĐỔI THÔNG TIN KHÁCH HÀNG

Chọn khách hàng
Ngo Ngọc Linh

Họ và tên lót: Ngo Ngọc Khanh
Tên: Linh
Năm sinh: 2003
Giới tính: Nữ

Cập nhật

Hình 4.15: Trang cá nhân của quản lý

Các chức năng chính:

Đăng ký khách hàng: Chức năng này được thực hiện bởi các user có role là "Employee" (nhân viên đăng ký), nhân viên đăng ký sẽ điền thông tin của khách hàng vào form sau đó sẽ gửi tới API POST /api/users/

Gửi câu hỏi: Ở đây cả khách hàng lẫn người dùng chưa đăng nhập đều có thể gửi câu hỏi lên hệ thống. Các câu hỏi từ người dùng chưa đăng nhập đã được trả lời sẽ gộp chung vào phần 'thông tin hỗ trợ' để những người dùng khác tham khảo.

Trả lời câu hỏi: Chức năng này được sử dụng bởi nhân viên đăng ký khách hàng, câu hỏi sau khi được trả lời sẽ được cập nhật trên 'thông tin hỗ trợ'.

Chỉnh sửa thông tin khách hàng: Chức năng này được thực hiện bởi quản lý siêu thị, quản lý gửi thông tin tới hệ thống qua API PUT /api/users/:user_id/

4.8 Đánh giá hệ thống

Sau khoảng thời gian thiết kế và phát triển hệ thống, nhóm em xin tự đánh giá về hệ thống của nhóm:

- Hệ thống của nhóm đã được hoàn thiện và đáp ứng gần đủ các nhu cầu cần thiết của một hệ thống đăng kí khách hàng thành viên của chuỗi cửa hàng cà phê.
- Hệ thống đã thêm được một số tính năng mở rộng như: Đặt và trả lời câu hỏi, gửi thông báo cho khách hàng, ...
- Hệ thống đơn giản về thao tác cho người dùng mới.
- Hệ thống có bảo mật cho người dùng thông qua tài khoản người dùng.
- Có phân vùng riêng cho nhân viên, quản lý và khách hàng.

Trong tương lai, nhóm muốn cải tiến hệ thống hơn so với hiện tại:

- Hoàn thiện các Yêu cầu chức năng còn sót.
- Hiện thực việc đăng nhập thông qua Google.
- Có thể gửi thông báo qua Email và SMS.
- Hệ thống hóa lại code của dự án để dễ dàng mở rộng và liên kết với các hệ thống khác trong chuỗi cửa hàng cà phê trong tương lai.
- Nâng cao hiệu năng của dự án

[Link Source Code của hệ thống](#)

Tài liệu

- [1] Functional vs Non-Functional Requirements: The Definitive Guide. Truy cập tại https://qracorp.com/functional-vs-non-functional-requirements/?fbclid=IwAR0x4GY5vzOGMERg8FtW9MDhQo5aSf_IQyYMk7k1tMYzrpE95jj1h8oZoFM
- [2] Why is the difference between functional and Non-functional requirements important? Truy cập tại https://reqtest.com/requirements-blog/functional-vs-non-functional-requirements/?fbclid=IwAR0knn_I8sH3cpt9GKwXXxZc9JSX-JQjJVm5DVzkDz0a--3B1nrwTgWZxG4
- [3] Slide môn học Hệ cơ sở dữ liệu (CO2013).
- [4] Mô hình erd là gì? Cách chuyển mô hình erd sang mô hình quan hệ. Truy cập tại <https://dinhnghia.vn/mo-hinh-erd-la-gi.html>
- [5] Sự khác biệt giữa Sơ đồ ER và EER. Truy cập tại <https://vi.strephonsays.com/er-and-vs-eer-diagram-10963>
- [6] MySQL là gì? Ưu nhược điểm và tầm quan trọng của MySQL. Truy cập tại <https://www.thegioididong.com/game-app/mysql-la-gi-uu-nhuoc-diem-va-tam-quan-trong-cua-mysql-1352651>
- [7] Phân biệt Stored Procedure, Function, Trigger. Truy cập tại https://www.c-sharpcorner.com/blogs/about-store-proc-function-trigger-in-brif?fbclid=IwAR2W8UbDP0qq3SCJRpeCeCEr0GCoVuBcRzhEpi_xcnVfQ6nFhC9zuInkf3Q
- [8] Phân biệt Stored Procedure và Function. Truy cập tại https://www.dotnettricks.com/learn/sqlserver/difference-between-stored-procedure-and-function-in-sql-server?fbclid=IwAR0SkqeLQChchVIz1Bz_iYwCsxm7mI_a2E3Z3aqB7JYEi10Y7e2wWbl6Jic
- [9] React - Một thư viện Javascript xây dựng giao diện người dùng Truy cập tại <https://vi.reactjs.org/>