

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN
VÀ TRUYỀN THÔNG



**BÁO CÁO BÀI TẬP
LỚN**

ĐỀ TÀI:

TRANG WEB DỊCH VỤ DỌN DẸP NHÀ CỬA

GVHD: T.s Nguyễn Bình Minh

Nhóm 26:

1. Nguyễn Ngọc Thọ	CNTT 10 - K62	20173387
2. Trần Trung Nghĩa	KTMT 08 - K62	20173281
3. Đinh Lê Nhất Thống	CNTT 11 – K62	20173391
4. Vũ Ngọc Trường	KTMT 06 – K62	20173426

Hà Nội, 12/2020

Nội dung

1. Giới thiệu về hệ thống

1.1 Giới thiệu tổng quan

1.2 Công nghệ sử dụng

1.2.1 Front-end

1.2.2 Back-end

1.2.3 Thiết kế cơ sở dữ liệu

2. Phân tích nghiệp vụ

2.1 Biểu đồ use case

2.2 Biểu đồ hoạt động

2.3 Thiết kế Database

3. Đặc tả ca sử dụng

4. Phân công nhiệm vụ

1. Giới thiệu về hệ thống

1.1 Giới thiệu về sản phẩm

Việc vệ sinh nhà cửa chưa bao giờ là một điều dễ dàng. Thời gian dành cho việc dọn dẹp là rất lớn cũng như công sức bỏ ra để có được một tổ ấm sạch sẽ là rất đáng kể. Hơn nữa, cùng với sự phát triển liên tục của xã hội, quỹ thời gian rảnh rỗi để chăm sóc cho bản thân và gia đình của nhiều người bị thu hẹp lại. Nhận thấy điều đó, dịch vụ dọn dẹp vệ sinh nhà cửa được sinh ra nhằm giúp người sử dụng tiết kiệm được khoản thời gian quý giá để có thể chăm sóc tốt hơn cho bản thân và gia đình.

Từ đó, để đáp ứng nhu cầu về dịch vụ, áp dụng công nghệ thông tin và truyền thông để mang đến cho khách hàng trải nghiệm tốt nhất, chúng tôi mang đến sản phẩm ***“Trang web dịch vụ vệ sinh nhà cửa”***

1.2 Công nghệ sử dụng

1.2.1. Front-end



React là một thư viện JavaScript nhằm đơn giản hóa việc phát triển giao diện người dùng. Được phát triển bởi Facebook và bản release đầu tiên trên thế giới vào năm 2013 và được sử dụng để viết ra Facebook, instagram và được sự ủng hộ sử dụng của rất nhiều công ty khác trên thế giới. Mục tiêu của react chính là đơn giản để phát triển. Tất cả trạng thái đều được tập trung tại một thời điểm, bằng cách chia giao diện người dùng thành tập hợp các thành phần (components)

Với những ưu điểm của mình, React được Nhóm 26 lựa chọn để sử dụng xây dựng phần front-end của dự án ***“Trang web dịch vụ vệ sinh nhà cửa”***.

1.2.2. Back-end



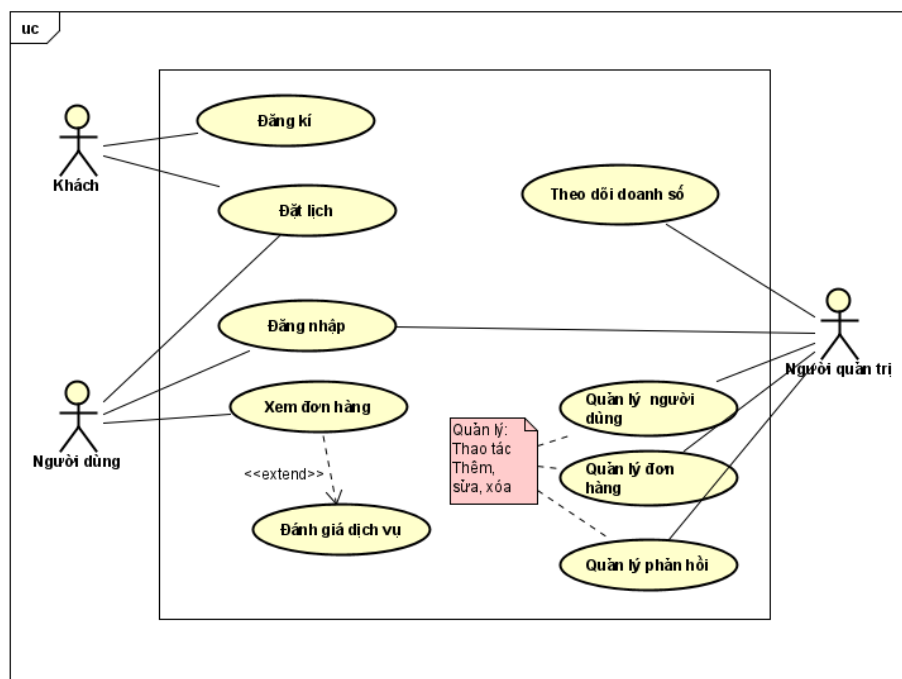
Laravel là một framework mã nguồn mở miễn phí dựa trên ngôn ngữ PHP, được xây dựng nhằm hỗ trợ phát triển các phần mềm ứng dụng.

Ưu điểm của Laravel là rất đáng kể như: nguồn tài nguyên lớn và sẵn có, thuận tiện để tham khảo dành cho các lập trình viên, tốc độ xử lý nhanh, tính bảo mật cao,.. Ngoài ra Laravel hỗ trợ các tính năng như Restful API, Artisan,... giúp Laravel trở thành công cụ tốt để xây dựng các API cần thiết cho dự án **“Trang web dịch vụ vệ sinh nhà cửa”**

2. Phân tích nghiệp vụ

2.1 Biểu đồ use case

- Biểu đồ use case tổng quan:

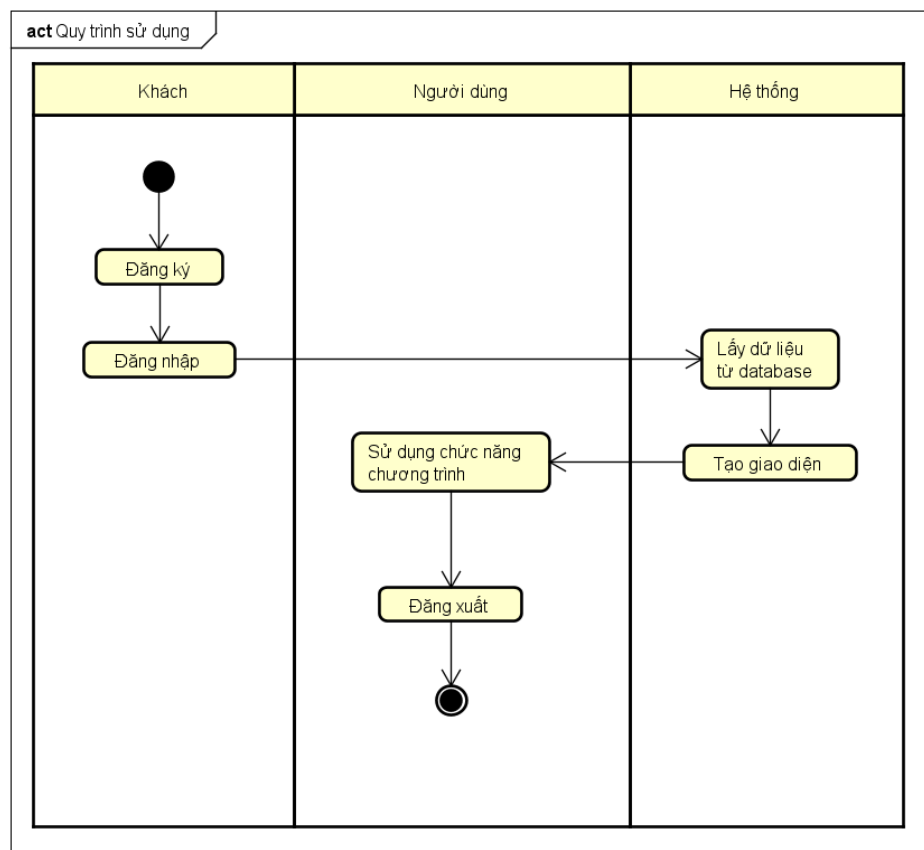


Các tác nhân trong hệ thống bao gồm:

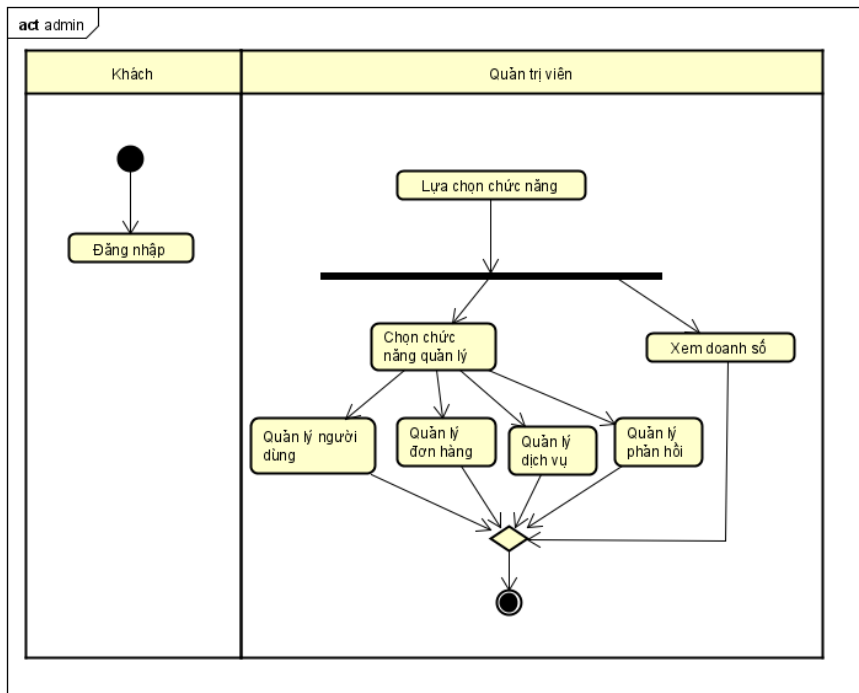
- + Khách: Người ghé thăm website và chưa có tài khoản đăng nhập
- + Người dùng: Người đã có tài khoản trong hệ thống
- + Người quản trị: Người được cấp tài khoản quản trị

2.2 Biểu đồ hoạt động

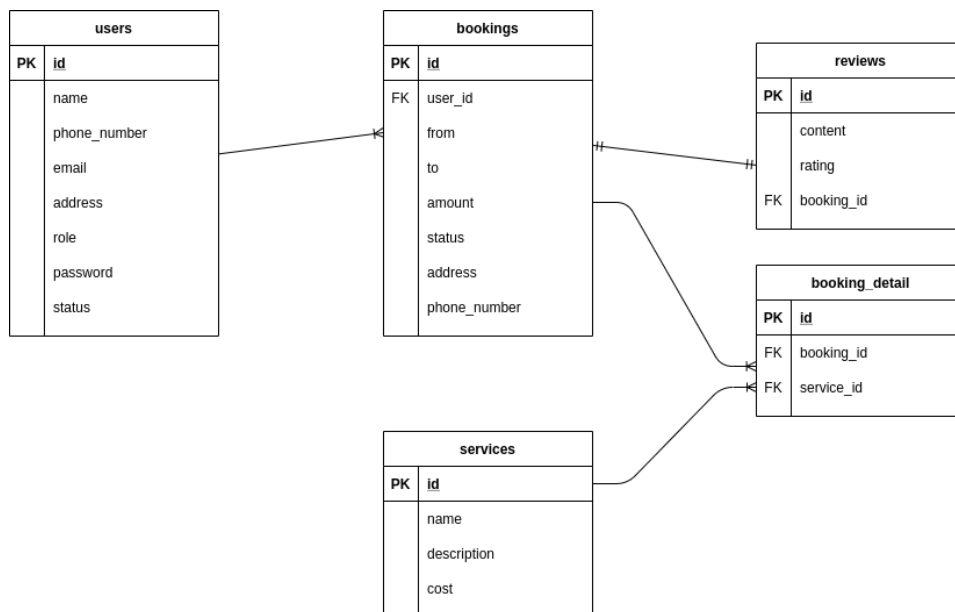
- Quy trình sử dụng hệ thống:



- Quy trình hoạt động của người quản trị:



2.3 Thiết kế Database



Cơ sở dữ liệu của dự án **“Trang web dịch vụ vệ sinh nhà cửa”** gồm các bảng thành phần bao gồm:

- Users: Lưu trữ thông tin về người dùng sau khi đăng kí tài khoản
- Bookings: Lưu trữ thông tin về các đơn hàng
- Reviews: Lưu trữ thông tin về các bài đánh giá
- Services: Thông tin về các gói dịch vụ

- Booking_details: Thể hiện liên kết giữa đơn hàng và dịch vụ (xác định đơn hàng thuộc loại dịch vụ nào)

3. Đặc tả ca sử dụng

- Use case Đăng kí:

Use case ID (Mã use case)	UC01
Use case name (Tên use case)	Đăng kí
Description (Mô tả)	Người dùng đăng kí tài khoản để sử dụng các chức năng của website
Actor (Tác nhân)	Khách
Trigger (Điều kiện kích hoạt)	Người dùng muốn đăng kí tài khoản
Pre-condition(s) (Tiền điều kiện)	+ Người dùng chưa có tài khoản hoặc muốn tạo mới tài khoản khác + Thiết bị của người dùng đã được kết nối internet khi thực hiện đăng kí
Post-condition(s) (Hậu điều kiện)	+ Người dùng đăng kí tài khoản thành công + Hệ thống ghi nhận tài khoản đã được đăng kí
Basic flow (Luồng chính)	1. Người dùng mở website 2. Người dùng chọn đăng kí tài khoản 3. Hệ thống hiển thị giao diện đăng kí tài khoản 4. Người dùng nhập thông tin tài khoản, mật khẩu một số thông tin cần thiết khác 5. Hệ thống xác thực thông tin đăng kí thành công và cho phép người dùng đăng nhập ứng dụng với tài khoản đã đăng kí 6. Hệ thống ghi nhận tài khoản đăng kí thành công
Alternative flow (Luồng thay thế)	Không có
Exception flow (Luồng ngoại lệ)	5a. Hệ thống xác thực thông tin đăng kí không thành công và hiển thị thông báo lỗi tương ứng (username bị trùng, mật khẩu không đúng yêu cầu/ không đủ an toàn, không nhập đủ các thông tin bắt buộc,...)

	5a1. Người dùng chọn hủy đăng kí <i>Use case dừng lại</i>
--	--

- Dữ liệu đầu vào yêu cầu:
 - Họ và tên
 - Số điện thoại: Không chứa chữ và kí tự đặc biệt, tối thiểu 3 chữ số.
 - Địa chỉ E-mail: Theo đúng định dạng
 - VD: xxx@yyy.zzz
 - Địa chỉ thực tế
 - Mật khẩu: Tối thiểu chứa 6 kí tự
- Use case Đăng nhập:

Use case ID (Mã use case)	UC02
Use case name (Tên use case)	Đăng nhập
Description (Mô tả)	Người dùng đăng nhập vào ứng dụng để sử dụng các chức năng của website
Actor (Tác nhân)	Khách
Trigger (Điều kiện kích hoạt)	Người dùng muốn đăng nhập vào website để mua hàng
Pre-condition(s) (Tiền điều kiện)	+ Tài khoản người dùng được đăng kí, tạo sẵn + Thiết bị của người dùng đã được kết nối internet khi thực hiện đăng nhập
Post-condition(s) (Hậu điều kiện)	+ Người dùng đăng nhập website thành công + Hệ thống ghi nhận hoạt động đăng nhập thành công
Basic flow (Luồng chính)	1. Người dùng mở website 2. Người dùng chọn đăng nhập 3. Hệ thống hiển thị giao diện đăng nhập 4. Người dùng nhập tài khoản, mật khẩu và chọn đăng nhập 5. Hệ thống xác thực thông tin đăng nhập thành công và cho phép người dùng truy cập website 6. Hệ thống ghi nhận hoạt động đăng nhập thành công
Alternative flow (Luồng thay thế)	Không có

Exception flow (Luồng ngoại lệ)	<p>5c. Hệ thống xác thực thông tin đăng nhập không thành công và hiển thị thông báo lỗi tương ứng (sai username/ password, nhập thiếu username/ password,...)</p> <p>5c1. Người dùng chọn hủy đăng nhập</p> <p><i>Use case dừng lại</i></p>
------------------------------------	---

- Use case Đặt lịch dịch vụ

Use case ID (Mã use case)	UC03
Use case name (Tên use case)	Đặt lịch
Description (Mô tả)	Người sử dụng đặt lịch dịch vụ
Actor (Tác nhân)	Khách/ Người dùng
Trigger (Điều kiện kích hoạt)	Người dùng muốn đặt lịch dịch vụ của website
Pre-condition(s) (Tiền điều kiện)	+ Thiết bị của người dùng đã được kết nối internet khi thực hiện đăng nhập
Post-condition(s) (Hậu điều kiện)	+ Website hiển thị thông báo đặt lịch thành công + Hệ thống ghi nhận đặt lịch thành công
Basic flow (Luồng chính)	<p>1. Người dùng mở website</p> <p>2. Người dùng chọn đặt lịch</p> <p>3. Hệ thống hiển thị giao diện đặt lịch</p> <p>4. Người dùng nhập biểu mẫu cho sẵn</p> <p>5. Hệ thống xác thực thông tin hợp lệ và báo thành công</p> <p>6. Hệ thống ghi nhận đơn hàng vào cơ sở dữ liệu</p>
Alternative flow (Luồng thay thế)	Không có
Exception flow (Luồng ngoại lệ)	<p>5c. Hệ thống kiểm tra thấy thiếu thông tin</p> <p>5c1. Người dùng chọn hủy</p> <p><i>Use case dừng lại</i></p>

- Dữ liệu đầu vào yêu cầu:

- Tên người đặt:
- Địa chỉ: Địa điểm đặt lịch dọn dẹp
- Số điện thoại: Không chứa chữ và kí tự đặc biệt, tối thiểu 3 chữ số

- Ngày đặt lịch: Ngày đặt phải sau thời điểm hiện tại
- Giờ đặt lịch cụ thể: Theo định dạng 24 giờ.

4. Phân công công việc

Phân công của từng thành viên trong nhóm:

- + Nguyễn Ngọc Thọ: API Back-end, thiết kế phần mềm
- + Trần Trung Nghĩa: Front-end
- + Đinh Lê Nhất Thống: Front-end
- + Vũ Ngọc Trường: Báo cáo, trình chiếu