



# WEBSITE ĐỀ XUẤT DỊCH VỤ

Version 1.11a

Tác giả: Nhóm 3  
Thành viên : Cao Chánh Dương 1610571  
Hồ Văn Long 1611861  
Lương Tuấn Kiệt 1611695  
Thân Đức Tài 1613027  
Lê Đức Mạnh 1611985

HCMUT, May 2019

### Lịch sử chỉnh sửa

Ngày	Phiên bản	Thay đổi	Người thay đổi
09/03/2019	1.0	Giới thiệu Mô tả tổng quan mô hình hệ thống	Nhóm 3
16/03/2019	1.1a	Thêm mô tả chi tiết nhánh chức năng 3.1	Mạnh
16/03/2019	1.1b	Thêm mô tả chi tiết nhánh chức năng 3.2	Tài
16/03/2019	1.1c	Thêm mô tả chi tiết nhánh chức năng 3.3	Dương
16/03/2019	1.1d	Thêm mô tả chi tiết nhánh chức năng 3.4	Kiệt
16/03/2019	1.1e	Thêm mô tả chi tiết nhánh chức năng 3.5	Long
23/03/2019	1.2a	Sửa đổi, bổ sung phần chi tiết mô tả 3.1	Mạnh
23/03/2019	1.2b	Sửa đổi, bổ sung phần chi tiết mô tả 3.2	Tài
23/03/2019	1.2c	Sửa đổi, bổ sung phần chi tiết mô tả 3.3	Dương
23/03/2019	1.2d	Sửa đổi, bổ sung phần chi tiết mô tả 3.4	Kiệt
23/03/2019	1.2e	Sửa đổi, bổ sung phần chi tiết mô tả 3.5	Long
30/03/2019	1.3a	Thêm user story và lưu đồ 3.1	Mạnh
30/03/2019	1.3b	Thêm user story và lưu đồ 3.3	Tài
30/03/2019	1.3c	Thêm user story và lưu đồ 3.3	Dương
30/03/2019	1.3d	Thêm user story và lưu đồ 3.4	Kiệt
30/03/2019	1.3e	Thêm user story và lưu đồ 3.5	Long
06/04/2019	1.4a	Thêm UI phần 4.1	Dương
06/04/2019	1.4b	Thêm UI phần 4.2	Kiệt
06/04/2019	1.4c	Thêm UI phần 4.3	Tài

06/04/2019	1.5d	Thêm UI phần 4.4	Long
06/04/2019	1.5e	Thêm UI phần 4.5	Mạnh
13/04/2019	1.5a	Thêm mô tả UI phần 4.1	Dương
13/04/2019	1.5b	Thêm mô tả UI phần 4.2	Kiệt
13/04/2019	1.5c	Thêm mô tả UI phần 4.3	Tài
13/04/2019	1.5d	Thêm mô tả UI phần 4.4	Long
13/04/2019	1.5e	Thêm mô tả UI phần 4.5	Mạnh
20/04/2019	1.6a	Thêm mô tả Kiến trúc hệ thống phần 5 và class Diagram phần 6.1	Dương
20/04/2019	1.6b	Thêm mô tả các method cho hệ thống phần 6.2.1	Long
20/04/2019	1.6c	Thêm mô tả các method cho hệ thống phần 6.2.2	Kiệt
20/04/2019	1.6d	Thêm mô tả các method cho hệ thống phần 6.2.3	Tài
20/04/2019	1.6e	Thêm mô tả các method cho hệ thống phần 6.2.4	Mạnh
27/04/2019	1.7a	Sửa UI cho hệ thống phần 4.1	Dương
27/04/2019	1.7b	Sửa UI cho hệ thống phần 4.2	Kiệt
27/04/2019	1.7c	Sửa UI cho hệ thống phần 4.3	Long
27/04/2019	1.7d	Sửa UI cho hệ thống phần 4.4	Mạnh
27/04/2019	1.7e	Sửa UI cho hệ thống phần 4.5	Tài
27/04/2019	1.7f	Thêm link git repo cho hệ thống phần 7	Nhóm 3



04/05/2019	1.8a	Sửa Class Diagram và Method Specification phần 6	Nhóm 3
11/05/2019	1.9a	Sửa Class Diagram và Method Specification phần 6	Nhóm 3
18/05/2019	1.10a	Sửa Class Diagram và Method Specification phần 6	Nhóm 3
25/05/2019	1.11a	Sửa Class Diagram và Method Specification phần 6	Nhóm 3



## Mục lục

1. Giới thiệu	8
2. Tổng quan hệ thống	8
3. Mô tả chi tiết	10
3.1 Nhánh chức năng: hiển thị quảng cáo, đăng ký quảng cáo và theo dõi quảng cáo	10
3.1.1 Chức năng hiển thị quảng cáo	10
a) User Story	10
b) Use-case Scenario Tabular	11
c) Sequence Diagram	12
3.1.2 Chức năng: Đăng ký quảng cáo	12
a) User Story:	12
b) Use-case Scenario Tabular	12
c) Sequence Diagram	14
3.1.3 Chức năng: Theo dõi quảng cáo	14
a) User Story	14
b) Use-case Scenario Tabular	14
c) Sequence Diagram	15
3.2 Nhánh chức năng: thêm mới loại dịch vụ, dịch vụ.	16
3.2.1 Thêm mới một loại dịch vụ	16
a) User Story:	16
b) Use-case Scenario Tabular	17
c) Sequence Diagram	18
3.2.2 Nhà cung cấp dịch vụ thêm mới dịch vụ	19
a) User Story	19
b) Use-case Scenario Tabular	19
c) Sequence Diagram	21
3.3 Nhánh chức năng: tìm kiếm, hiển thị và đề xuất	21
3.3.1 Tìm kiếm và xem chi tiết dịch vụ	21
a) User Story	21
b) Use-case Scenario Tabular	21
c) Sequence Diagram	23
3.3.2 Đề xuất dịch vụ tốt nhất trên hệ thống	23



a) User Story	23
b) Use-case Scenario Tabular:	23
c) Sequence Diagram	24
3.4 Nhánh chức năng: Bình luận, trả lời bình luận, chia sẻ	25
3.4.1 Trả lời bình luận về chất lượng dịch vụ	25
a) User Story	25
b) Use-case Scenario	25
c) Sequence Diagram	26
3.4.2 Bình luận về chất lượng dịch vụ	26
a) User Story	26
b) Use-case Scenario Tabular	27
c) Sequence Diagram	28
3.4.3 Chia sẻ dịch vụ	28
a) User Story	28
b) Use-case Scenario Tabular	29
3.5 Nhánh chức năng: Đăng ký tìm hiểu dịch vụ, đánh giá chất lượng dịch vụ	30
3.5.1 Đăng ký tìm hiểu dịch vụ	30
a) User Story	30
b) Use-case Scenario Tabular	30
c) Sequence Diagram	31
3.5.2 Đánh giá chất lượng dịch vụ	31
a) User Story	31
b) Use-case Scenario Tabular	32
c) Sequence Diagram	33
4. User Interface	34
4.1 Nhánh chức năng: tìm kiếm, hiển thị và đề xuất	34
4.2 Nhánh chức năng: Bình luận, trả lời bình luận, chia sẻ	38
4.3 Nhánh chức năng: Thêm mới loại dịch vụ, dịch vụ	41
4.4 Nhánh chức năng: Đăng ký tìm hiểu dịch vụ, đánh giá chất lượng dịch vụ	45
4.5 Nhánh chức năng: hiển thị quảng cáo, đăng ký quảng cáo và theo dõi quảng cáo	47
5. Kiến trúc hệ thống	50
5.1 Thiết kế kiến trúc	50
5.2 Mô tả kiến trúc	51



5.2.1 Người dùng bình thường (User)	51
5.2.2 Admin	52
5.2.3 Provider	53
5.2.4 Server	54
5.2.5 Database	56
6. Module Design	56
6.1 Class Diagram	56
6.2 Method Specification	57
6.2.1 BaseUser Class	57
6.2.2 User Class	59
6.2.3 Admin Class	62
6.2.4 DAO Class	65
6.2.5 Provider Class	78
7. Github Repository	81



## 1. Giới thiệu

Tài liệu này mô tả mô hình hệ thống và cách thức hoạt động của Website đề xuất dịch vụ, được phát triển bởi Nhóm 3, trường Đại học Bách Khoa TP Hồ Chí Minh.

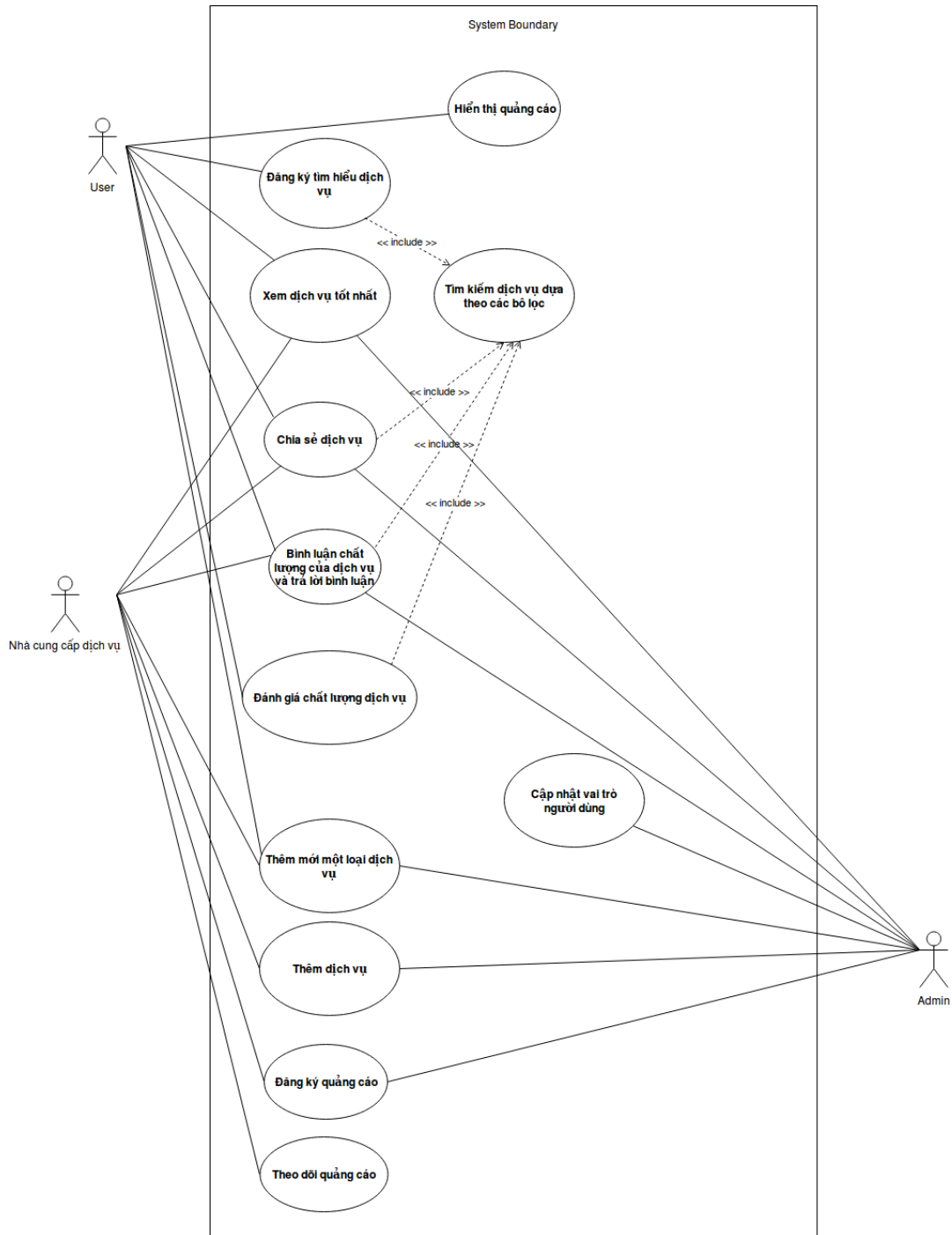
Website hướng đến việc hỗ trợ đề xuất các dịch vụ cho người dùng, các dịch vụ có thể bao gồm y tế, giáo dục,...

Khi đã được hoàn thành, trang web có khả năng giúp người dùng tra cứu dịch vụ, tìm kiếm dịch vụ tốt nhất, giúp các nhà cung cấp dịch vụ có thể tiếp cận người dùng dễ dàng hơn, Website sẽ là cầu nối đưa người dùng và nhà cung cấp dịch vụ đến gần hơn với nhau.

## 2. Tổng quan hệ thống

Mô hình hệ thống được đặc tả use-case diagram sau:





Hình 1 – Usecase Diagram Hệ thống đề xuất dịch vụ

Như có thể thấy từ hình 1, hệ thống bao gồm 5 modules chính sau:

- Module 1 (Dương): Cho phép User/Nhà cung cấp dịch vụ/Admin có thể tra cứu, tìm kiếm dịch vụ theo các bộ lọc, xem chi tiết dịch vụ và đồng thời có thể xem các dịch vụ tốt nhất hiện có trên hệ thống trên trang chủ được xếp theo thứ tự đánh giá từ cao xuống thấp.

- Module 2 (Long): Cho phép User có thể đăng ký sử dụng dịch vụ để được nhà cung cấp liên hệ thông qua thông tin mà User đã đăng ký với hệ thống và có thể đánh giá chất lượng dịch vụ bằng cách rating bằng số sao khi nhấn vào xem một dịch vụ cụ thể.
- Module 3 (Mạnh): Các doanh nghiệp (nhà cung cấp) tham gia vào hệ thống có thể đăng ký quảng cáo của mình lên hệ thống. Ngoài ra, hệ thống còn cung cấp chức năng giúp doanh nghiệp theo dõi quảng cáo của mình đã được duyệt về số lượt xem quảng cáo dưới dạng trực quan dạng biểu đồ, User sẽ xem được các quảng cáo trên hệ thống.
- Module 4 (Tài): Cho phép User/nhà cung cấp thêm mới đề xuất một loại dịch vụ mới cho hệ thống. Ngoài ra, với mỗi một loại dịch vụ được thêm vào ở danh mục trên, nhà cung cấp dịch vụ có thể đăng ký một dịch vụ cụ thể mới để thêm vào hệ thống.
- Module 5 (Kiệt): Cho phép User có thể thể hiện ý kiến của mình về dịch vụ bằng cách bình luận/trả lời bình luận trên trang hiện chi tiết về dịch vụ, ngoài ra User còn có thể chia sẻ dịch vụ tới bạn bè của mình thông qua các trang mạng xã hội, ngoài ra admin và nhà cung cấp dịch vụ cũng có thể bình luận và trả lời bình luận.

### **3. Mô tả chi tiết**

#### **3.1 Nhánh chức năng: hiển thị quảng cáo, đăng ký quảng cáo và theo dõi quảng cáo**

##### **3.1.1 Chức năng hiển thị quảng cáo**

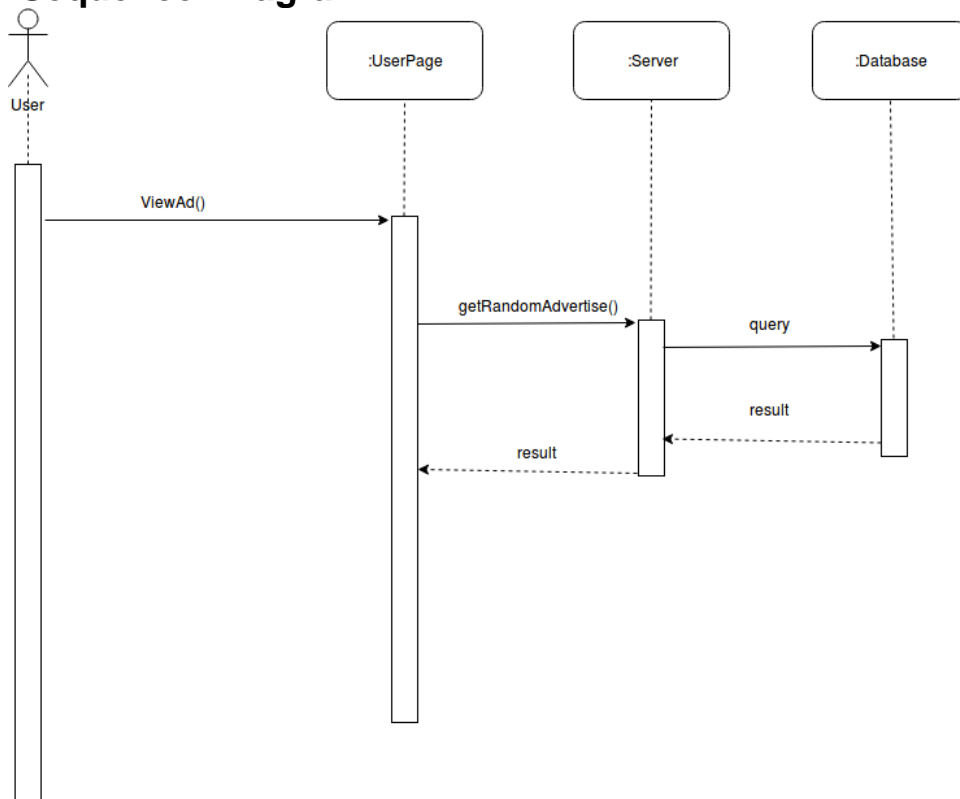
###### **a) User Story**

1. Các doanh nghiệp cung cấp dịch vụ có thể đăng kí quảng cáo dịch vụ trên hệ thống.
2. Quảng cáo giúp dịch vụ của nhà cung cấp tiếp cận được nhiều người hơn, thúc đẩy doanh số cho nhà cung cấp.
3. Quảng cáo hiển thị 3 quảng cáo bất kì trong các quảng cáo hiện có ở các banner quảng cáo trên hệ thống ở bất kì trang nào của hệ thống.

## b) Use-case Scenario Tabular

<b>Use-case ID:</b>	1001
<b>Use-case name:</b>	Hiển thị quảng cáo
<b>Created By:</b> Lê Đức Mạnh	<b>Last Updated By:</b> Lê Đức Mạnh
<b>Date Created:</b> 30/3/2019	<b>Date Last Updated:</b> 30/3/2019
<b>Actors:</b>	User
<b>Description:</b>	Hiển thị quảng cáo tới người dùng
<b>Trigger:</b>	Người dùng vào bất kì trang nào của hệ thống.
<b>Preconditions:</b>	Không.
<b>Postconditions:</b>	Không.
<b>Normal flow:</b>	<ol style="list-style-type: none"><li>1. Người dùng truy cập vào bất kì trang nào của hệ thống.</li><li>2. Banner quảng cáo trên trang web hiển thị ảnh của các quảng cáo được sắp ngẫu nhiên.</li><li>3. Người dùng click vào ảnh của quảng cáo để xem quảng cáo.</li></ol>
<b>Alternative flows:</b>	Alternative flow 1: Tại bước 3 3a. Người dùng có thể không click vào quảng cáo mà bỏ qua quảng cáo.
<b>Exceptions:</b>	Không.
<b>Notes and Issues:</b>	Không.

### c) Sequence Diagram



Hình 2 – Sequence Diagram Hiển thị quảng cáo

### 3.1.2 Chức năng: Đăng ký quảng cáo

#### a) User Story:

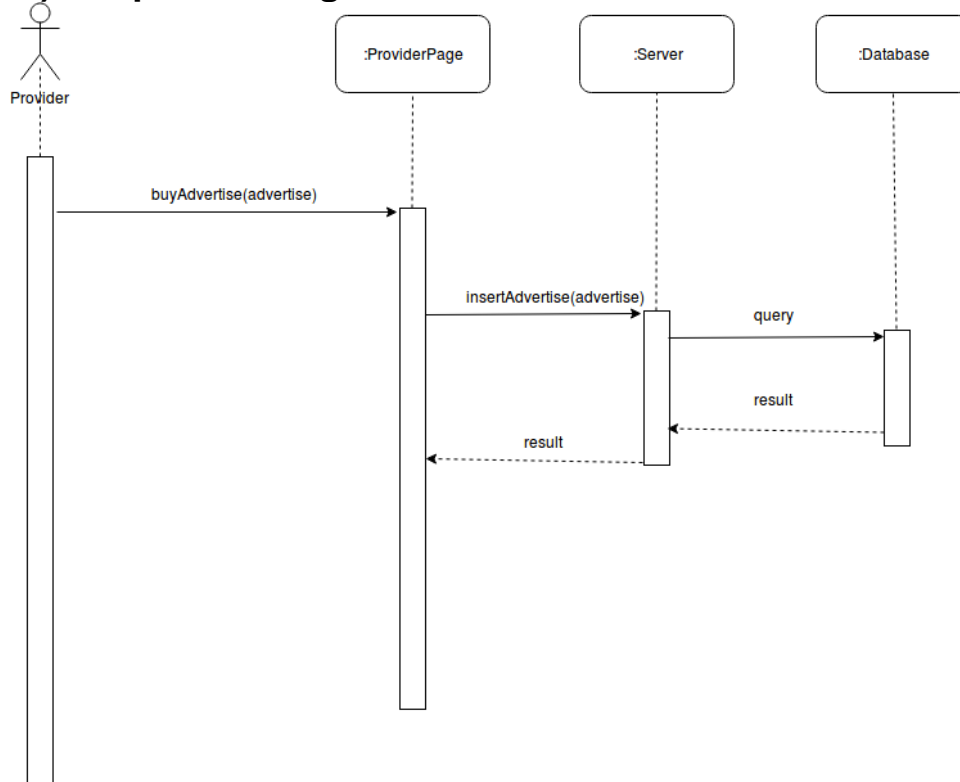
1. Nhà cung cấp dịch vụ sử dụng dashboard để yêu cầu thêm mới một quảng cáo của mình.
2. Thông tin bao gồm Tên, đường dẫn, avatar và loại quảng cáo (Normal/Enterprise), hai loại quảng cáo này khác nhau về số lượng lượt xem tối đa.
3. Nhà cung cấp chờ admin duyệt đăng ký của mình để quảng cáo của mình có thể được hiển thị trên trang của User.

#### b) Use-case Scenario Tabular

<b>Use-case ID:</b>	1002
<b>Use-case name:</b>	Đăng ký quảng cáo

<b>Created By:</b> Lê Đức Mạnh	<b>Last Updated By:</b> Lê Đức Mạnh
<b>Date Created:</b> 30/3/2019	<b>Date Last Updated:</b> 30/3/2019
<b>Actors:</b>	Nhà cung cấp dịch vụ
<b>Description:</b>	Nhà cung cấp dịch vụ theo dõi chiến dịch quảng cáo đã và đang chạy.
<b>Trigger:</b>	Nhà cung cấp dịch vụ nhấn vào tab "Buy Advertisements" trên Trang của nhà cung cấp.
<b>Preconditions:</b>	Nhà cung cấp đã đăng nhập tài khoản của mình trên trang của nhà cung cấp dịch vụ.
<b>Postconditions:</b>	Không.
<b>Normal flow:</b>	<ol style="list-style-type: none"> <li>1. Nội dung trong tab "Buy Advertisements" hiện ra, trong đó có mục "Add Advertisements" hiển thị một khung để nhà cung cấp điền thông tin về quảng cáo để đăng ký.</li> <li>2. Nhà cung cấp điền các thông tin về quảng cáo vào khung.</li> <li>3. Nhà cung cấp dịch vụ nhấn nút "REGISTER" để yêu cầu thêm quảng cáo được gửi đến admin để chờ duyệt.</li> </ol>
<b>Alternative flows:</b>	Không.
<b>Exceptions:</b>	Không.
<b>Notes and Issues:</b>	Không.

### c) Sequence Diagram



Hình 3 – Sequence Diagram Đăng ký quảng cáo

### 3.1.3 Chức năng: Theo dõi quảng cáo

#### a) User Story

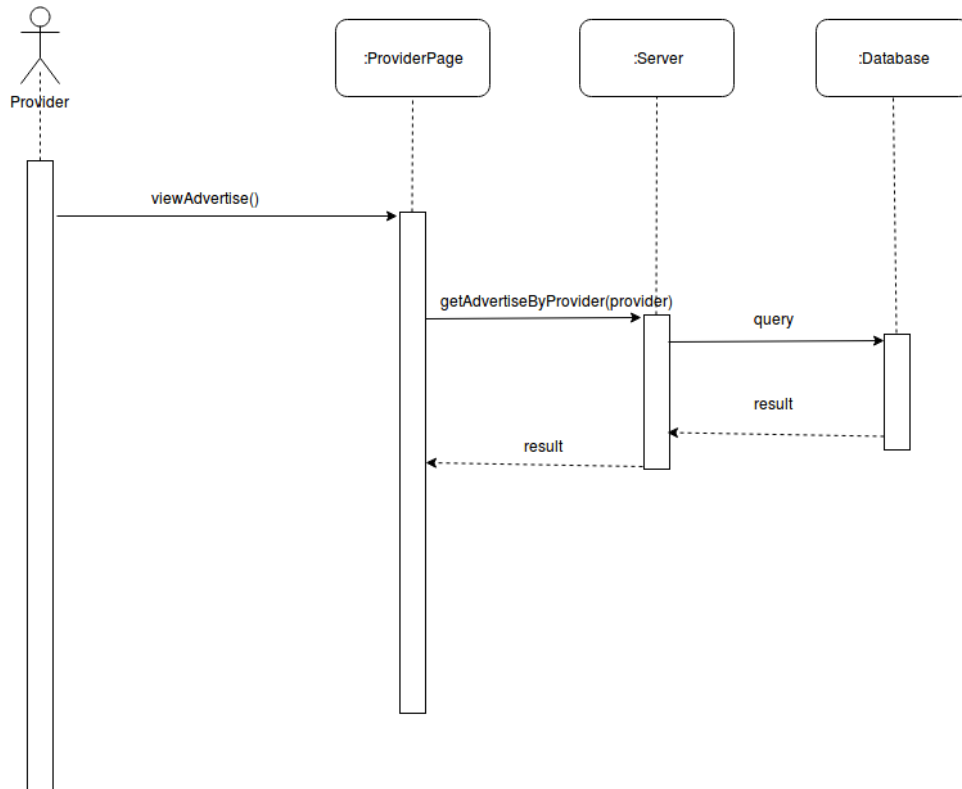
1. Nhà cung cấp dịch vụ sử dụng dashboard quảng cáo để theo dõi các thông tin về quảng cáo đã chạy.
2. Thông tin bao gồm số lượt views, trạng thái của quảng cáo, loại của quảng cáo, vv
3. Thông tin được hiển thị dưới dạng biểu đồ trực quan và theo dạng bảng.

#### b) Use-case Scenario Tabular

<b>Use-case ID:</b>	1003
<b>Use-case name:</b>	Theo dõi quảng cáo
<b>Created By:</b> Lê Đức Mạnh	<b>Last Updated By:</b> Lê Đức Mạnh
<b>Date Created:</b> 30/3/2019	<b>Date Last Updated:</b> 30/3/2019
<b>Actors:</b>	Nhà cung cấp dịch vụ

<b>Description:</b>	Nhà cung cấp dịch vụ theo dõi chiến dịch quảng cáo đã và đang chạy.
<b>Trigger:</b>	Nhà cung cấp dịch vụ nhấn vào tab "Statistics" hoặc "Buy Advertisements".
<b>Preconditions:</b>	Nhà cung cấp đã đăng nhập tài khoản của mình trên trang của nhà cung cấp.
<b>Postconditions:</b>	Không.
<b>Normal flow:</b>	1. Nội dung của tab "Statistics" hiện ra là biểu đồ về các quảng cáo đang chạy của nhà cung cấp hoặc nội dung của tab "Buy Advertisements" hiện ra với bảng số liệu về các quảng cáo của nhà cung cấp.
<b>Alternative flows:</b>	Không.
<b>Exceptions:</b>	Exception 1: Tại bước 1 1a. Nếu nhà cung cấp chưa có quảng cáo nào đã mua thì không hiện thông tin gì và chỉ hiện khung để nhà cung cấp đăng ký quảng cáo (nếu là tab "Buy Advertisements") hoặc không hiển thị biểu đồ nào (nếu là tab "Statistics").
<b>Notes and Issues:</b>	Không.

### c) Sequence Diagram



Hình 4– Sequence Diagram Theo dõi quảng cáo

## 3.2 Nhánh chức năng: thêm mới loại dịch vụ, dịch vụ.

### 3.2.1 Thêm mới một loại dịch vụ

#### a) User Story:

1. Người dùng nhấn vào nút "Suggest New Service Type".
2. Nút " Suggest New Service Type" bắt sự kiện click chuột của người dùng.
3. Một component hiện ra để thêm mới loại dịch vụ.
4. Người dùng nhập tên của loại dịch vụ mới mà mình muốn yêu cầu thêm vào.
5. Người dùng click nút "OK".
6. Trang web nhận được phản hồi và hiện thông báo lên màn hình rằng yêu cầu đã được gửi đi và đang chờ xử lý hoặc thông báo nếu như loại dịch vụ đã tồn tại hoặc đang chờ duyệt.
7. Khi có nhiều yêu cầu thêm mới của một loại dịch vụ được gửi về cho server, admin sẽ xem xét mở thêm loại dịch vụ đó trên trang web.

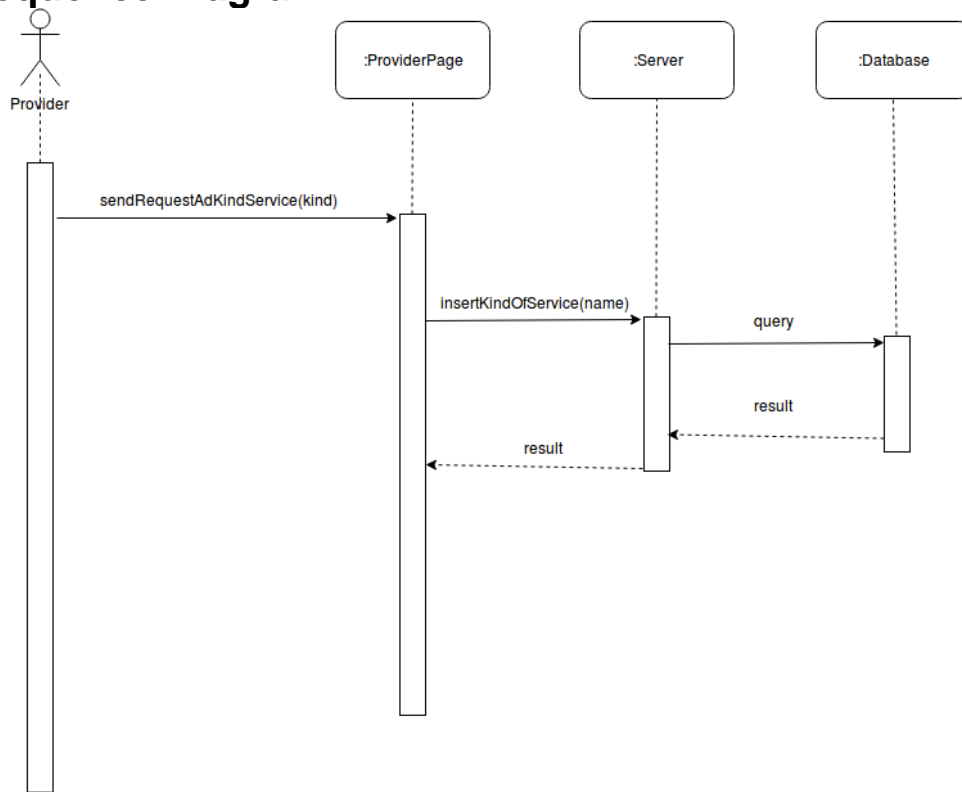


8. Ngoài ra nhà cung cấp dịch vụ cũng có thể yêu cầu thêm mới dịch vụ bằng cách vào mục "Add Service Type", nhập tên loại dịch vụ vào và nhấn nút "ADD".

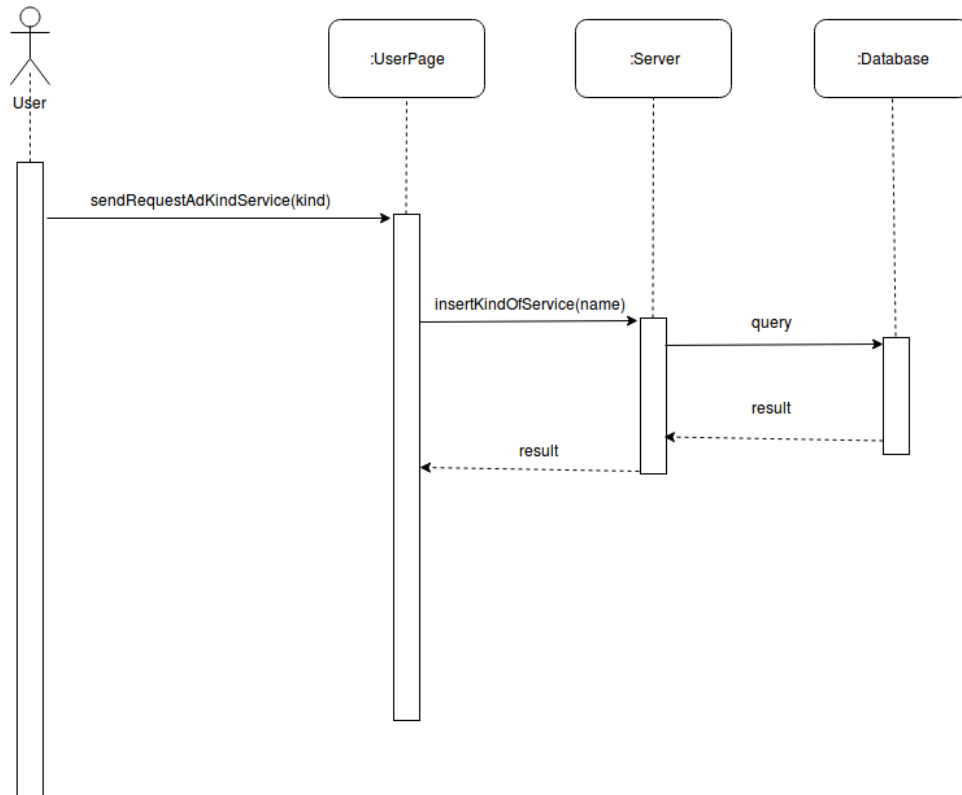
#### b) Use-case Scenario Tabular

<b>Use-case ID:</b>	2001
<b>Use-case name:</b>	Thêm mới loại dịch vụ
<b>Created By:</b> Thân Đức Tài	<b>Last Updated By:</b> Thân Đức Tài
<b>Date Created:</b> 30/3/2019	<b>Date Last Updated:</b> 30/3/2019
<b>Actors:</b>	User/Nhà cung cấp dịch vụ
<b>Description:</b>	Actor yêu cầu thêm mới một loại dịch vụ.
<b>Trigger:</b>	Nút "Suggest New Service Type" được click hoặc tab "Buy Advertisements" được nhấn.
<b>Preconditions:</b>	Actor đã đăng nhập vào hệ thống.
<b>Postconditions:</b>	Yêu cầu thêm mới loại dịch vụ được gửi đến hệ thống và chờ Admin duyệt.
<b>Normal flow:</b>	<ol style="list-style-type: none"> <li>1. Trang điền thông tin loại dịch vụ hiện ra.</li> <li>2. Actor nhập thông tin loại dịch vụ và bấm "ADD" hoặc "OK".</li> <li>3. Trang web hiển thị thông báo rằng yêu cầu đã được gửi đi.</li> </ol>
<b>Alternative flows:</b>	Không có.
<b>Exceptions:</b>	Exception 1: Tại bước 4 3a. Nếu đã tồn tại loại dịch vụ trên hệ thống hoặc đã có loại dịch vụ với tên tương tự đang chờ duyệt thì thông báo ra cho người dùng.
<b>Notes and Issues:</b>	Không có

### c) Sequence Diagram



Hình 5 – Sequence Diagram Thêm mới loại dịch vụ (Provider)



Hình 6 – Sequence Diagram Thêm mới loại dịch vụ (User)

### 3.2.2 Nhà cung cấp dịch vụ thêm mới dịch vụ

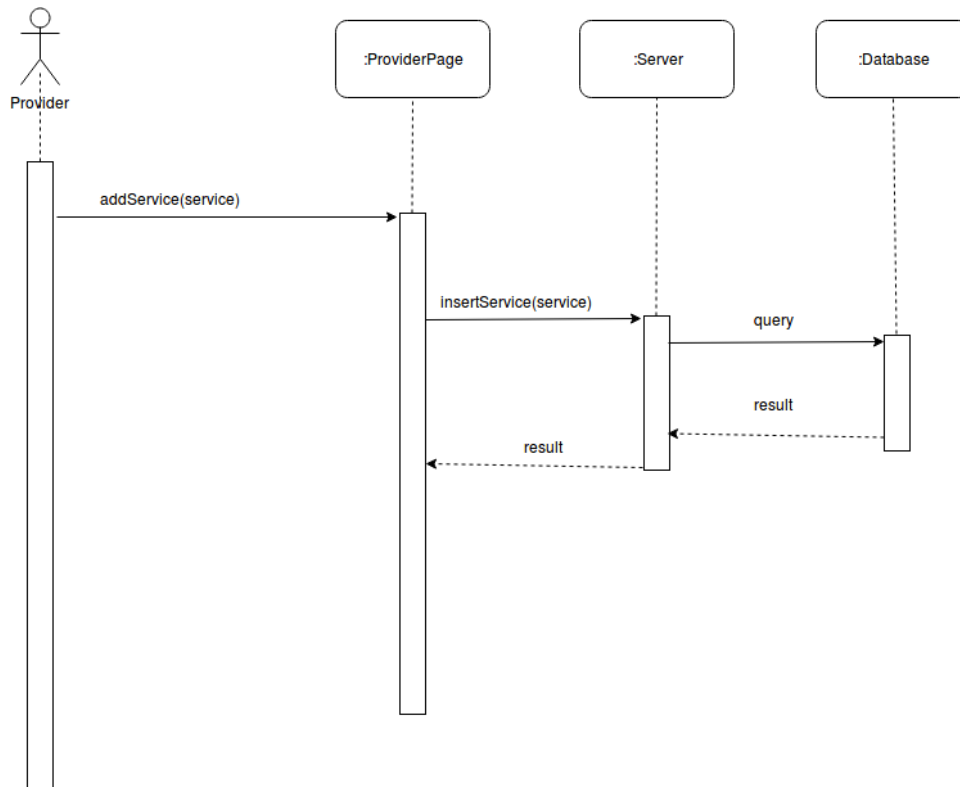
#### a) User Story

1. Nhà cung cấp dịch vụ nhấn vào nút "+" trong tab "Services".
2. Nút "+" bắt sự kiện click chuột của người dùng.
3. Một pop-up hiện ra để nhà cung cấp dịch vụ điền thông tin.
4. Nhà sở hữu dịch vụ nhập thông tin mô tả dịch vụ mới mà mình sẽ cung cấp.
5. Nhà cung cấp dịch vụ click nút "ADD SERVICE".
6. Trang web gửi API request cho server.
7. Trang web nhận được phản hồi và hiện thông báo lên màn hình rằng yêu cầu đã được gửi đi và đang chờ xử lý.
8. Khi yêu cầu đăng ký dịch vụ đã được admin xác minh, dịch vụ sẽ được thêm mới vào trang web và nhà sở hữu dịch vụ sẽ nhận được thông báo.

#### b) Use-case Scenario Tabular

<b>Use-case ID:</b>	2002
<b>Use-case name:</b>	Thêm mới dịch vụ
<b>Created By:</b> Thân Đức Tài	<b>Last Updated By:</b> Thân Đức Tài
<b>Date Created:</b> 30/3/2019	<b>Date Last Updated:</b> 30/3/2019
<b>Actors:</b>	Nhà cung cấp dịch vụ
<b>Description:</b>	Actor yêu cầu thêm mới một dịch vụ do mình sở hữu vào hệ thống.
<b>Trigger:</b>	Nút “ADD SERVICE” được click.
<b>Preconditions:</b>	Actor đã đăng nhập vào hệ thống.
<b>Postconditions:</b>	Yêu cầu thêm mới được gửi đến hệ thống.
<b>Normal flow:</b>	<ol style="list-style-type: none"> <li>1. Một pop-up để điền thông tin dịch vụ hiện ra.</li> <li>2. Actor nhập thông tin dịch vụ và bấm “ADD SERVICE”.</li> <li>3. Trang web hiển thị thông báo rằng yêu cầu đã được gửi đi.</li> </ol>
<b>Alternative flows:</b>	Không có.
<b>Exceptions:</b>	<p>Exception 1: Tại bước 3</p> <p>3a. Nếu gửi đến server gặp lỗi, thông báo lỗi sẽ hiện ra thay vì thông báo gửi thành công.</p>
<b>Notes and Issues:</b>	Không có

### c) Sequence Diagram



Hình 7 – Sequence Diagram Thêm mới dịch vụ

## 3.3 Nhánh chức năng: tìm kiếm, hiển thị và đề xuất

### 3.3.1 Tìm kiếm và xem chi tiết dịch vụ

#### a) User Story

1. Người dùng nói chung (User/Admin/Nhà cung cấp) nhấn vào nút "SERVICY" (tên của hệ thống).
2. Ở đây có ô tìm kiếm, người dùng nhập dịch vụ cần tìm kiếm, nhập các bộ lọc và ấn nút "SEARCH".
3. Trang chủ hiển thị kết quả tìm kiếm ra.
4. Người dùng chọn loại dịch vụ mà mình quan tâm bấm vào đó để xem nội dung chi tiết.
5. Nội dung chi tiết về dịch vụ sẽ hiển thị ra.

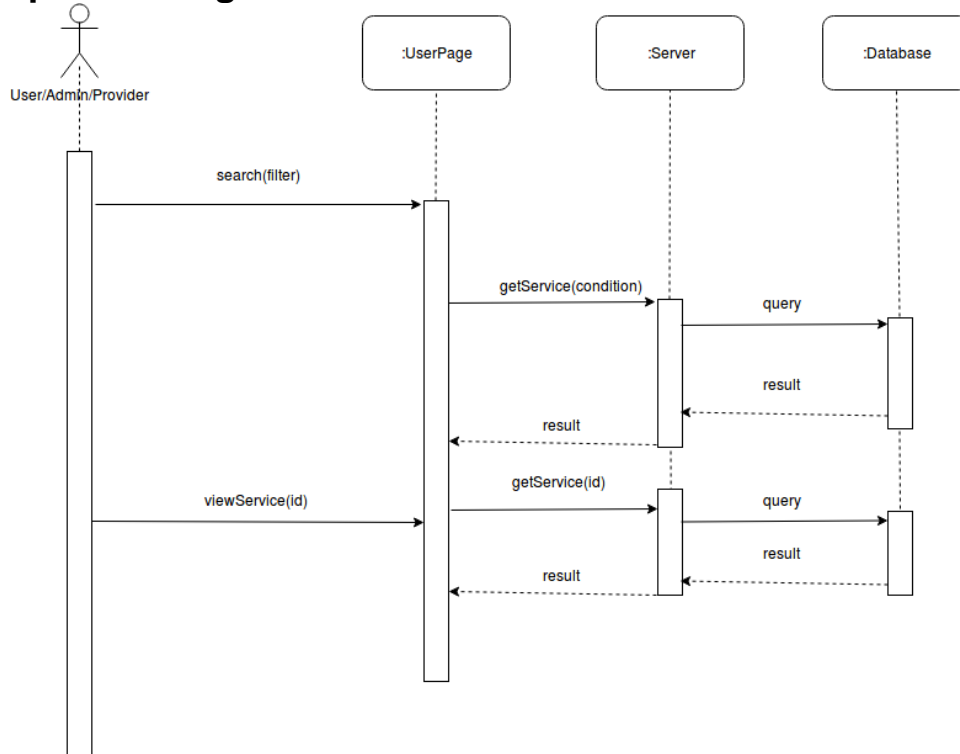
#### b) Use-case Scenario Tabular

Use-case ID:	3001
--------------	------

<b>Use-case name:</b>	Tìm kiếm và xem chi tiết dịch vụ.
<b>Created By:</b> Cao Chánh Dương	<b>Last Updated By:</b> Cao Chánh Dương
<b>Date Created:</b> 30/3/2019	<b>Date Last Updated:</b> 30/3/2019
<b>Actors:</b>	User/Admin/Nhà cung cấp dịch vụ
<b>Description:</b>	Người dùng tìm kiếm dịch vụ dựa theo các bộ lọc có sẵn và thanh tìm kiếm, có thể nhấn vào dịch vụ cụ thể để xem chi tiết.
<b>Trigger:</b>	Người dùng vào hệ thống.
<b>Preconditions:</b>	Người dùng có thể đã/không đăng nhập vào hệ thống.
<b>Postconditions:</b>	Người dùng tìm kiếm được dịch vụ thỏa yêu cầu tìm kiếm nhập vào, xem được chi tiết một dịch vụ.
<b>Normal flow:</b>	<ol style="list-style-type: none"> <li>1. Người dùng nhập vào thanh tìm kiếm và lựa chọn các bộ lọc.</li> <li>2. Người dùng nhấn vào nút "SEARCH".</li> <li>3. Website hiển thị kết quả tìm kiếm theo độ ưu tiên từ cao xuống thấp.</li> <li>4. Người dùng nhấn vào 1 kết quả tìm kiếm để xem chi tiết dịch vụ.</li> </ol>
<b>Alternative flows:</b>	<p>Alternative flow 1: Tại bước 1</p> <p>1a. Người dùng có thể chỉ nhập vào thanh tìm kiếm mà không chọn các bộ lọc hoặc chỉ chọn các bộ lọc mà không nhập vào thanh tìm kiếm.</p> <p>Alternative flow 2: Tại bước 4</p> <p>4a. Người dùng có thể không nhấn vào xem chi tiết dịch vụ.</p>
<b>Exceptions:</b>	<p>Exception 1: Tại bước 3</p> <p>3a. Nếu hệ thống không tìm thấy được loại dịch vụ nào thỏa điều kiện tìm kiếm trong database thì sẽ không hiển thị gì.</p>

<b>Notes and Issues:</b>	Không có

### c) Sequence Diagram



Hình 8 – Sequence Diagram Tìm kiếm và xem chi tiết dịch vụ

### 3.3.2 Đề xuất dịch vụ tốt nhất trên hệ thống

#### a) User Story

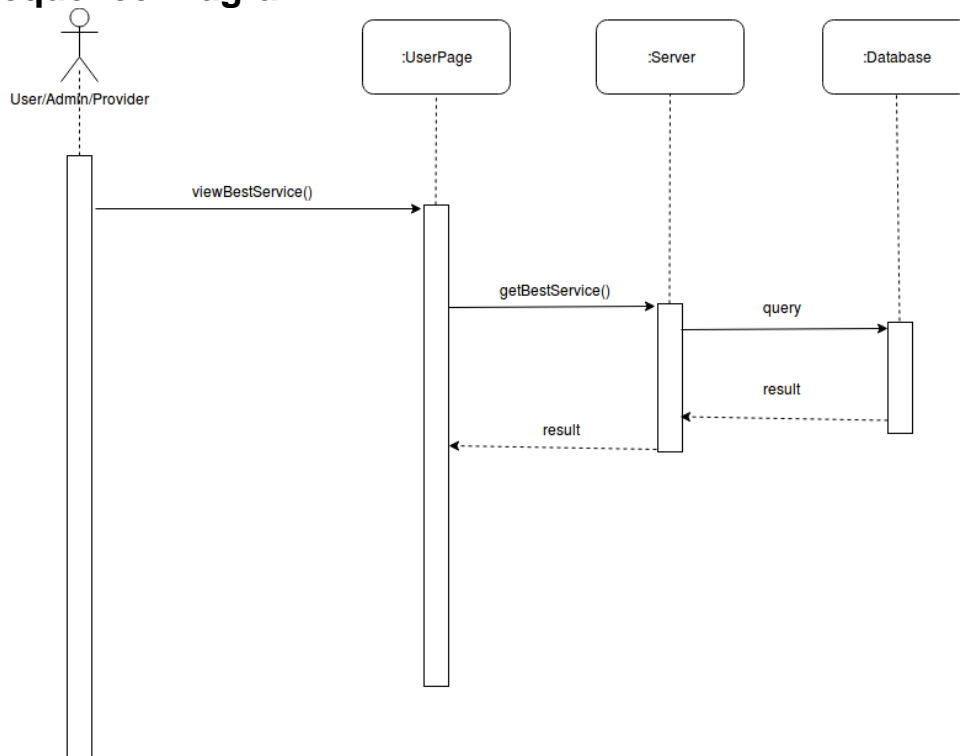
1. Người dùng nhấn vào nút "SERVICY" để quay về trang chủ.
2. Trang chủ hiển thị các dịch vụ tốt nhất ra trên trang chủ trong mục Best Service.

#### b) Use-case Scenario Tabular:

<b>Use-case ID:</b>	3002
<b>Use-case name:</b>	Xem dịch vụ tốt nhất được đề xuất
<b>Created By:</b> Cao Chánh Dương	<b>Last Updated By:</b> Cao Chánh Dương
<b>Date Created:</b> 30/3/2019	<b>Date Last Updated:</b> 30/3/2019

<b>Actors:</b>	User/admin/nhà cung cấp dịch vụ
<b>Description:</b>	Người dùng xem dịch vụ tốt nhất hiện có trên hệ thống.
<b>Trigger:</b>	Người dùng vào Trang chủ.
<b>Preconditions:</b>	Không có
<b>Postconditions:</b>	Người dùng xem được các dịch vụ tốt nhất.
<b>Normal flow:</b>	1. Người chọn vào nút Trang chủ . 2. Màn hình hiển thị ra các dịch vụ tốt nhất hiện có trên trang chủ.
<b>Alternative flows:</b>	Không có
<b>Exceptions:</b>	Exception 1: Tại bước 2 2a. Nếu trong Database không có dịch vụ nào thì không hiển thị ra dịch vụ tốt nhất.
<b>Notes and Issues:</b>	Không có

### C) Sequence Diagram



Hình 9 – Sequence Diagram Xem dịch vụ tốt nhất được đề xuất



### 3.4 Nhánh chức năng: Bình luận, trả lời bình luận, chia sẻ

#### 3.4.1 Trả lời bình luận về chất lượng dịch vụ

##### a) User Story

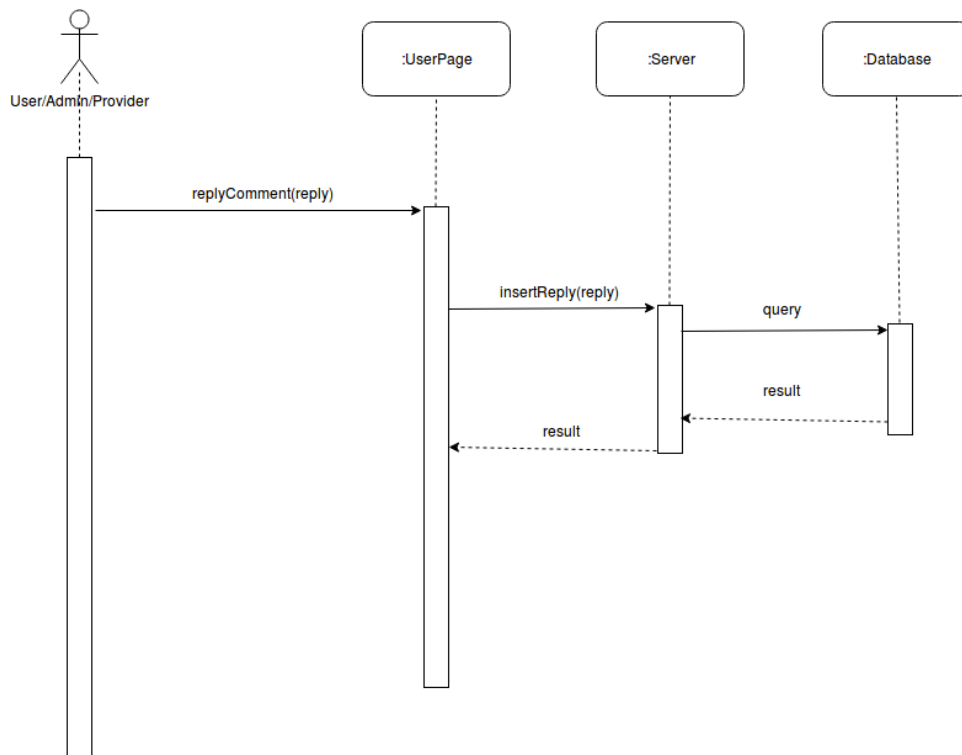
1. Người dùng nhấn vào nút "SERVICY".
2. Ở đây có ô tìm kiếm, người dùng nhập dịch vụ cần tìm kiếm và ấn nút "SEARCH".
3. Người dùng chọn dịch vụ mà mình quan tâm bấm vào đó để xem nội dung chi tiết.
4. Ở mục nội dung chi tiết của dịch vụ, người dùng nhấn vào nút "NHẬN XÉT" để xem các bình luận về sản phẩm.
5. Ở mỗi bình luận về dịch vụ, có mục "Reply" phía bên dưới. Người dùng nhấn vào nút "Reply" để trả lời những bình luận mà mình quan tâm về dịch vụ.
6. Nhập nội dung trả lời vào khung trả lời và ấn nút "Add Reply" để gửi nội dung trả lời của mình.
7. Nội dung trả lời được cập nhật.

##### b) Use-case Scenario

<b>Use-case ID:</b>	4001
<b>Use-case name:</b>	Trả lời bình luận về chất lượng dịch vụ
<b>Created By:</b> Lương Tuấn Kiệt	<b>Last Updated By:</b> Lương Tuấn Kiệt
<b>Date Created:</b> 30/3/2019	<b>Date Last Updated:</b> 30/3/2019
<b>Actors:</b>	User/Admin/Nhà cung cấp dịch vụ
<b>Description:</b>	Người dùng trả lời bình luận chất lượng dịch vụ ở một bình luận cụ thể.
<b>Trigger:</b>	Người dùng nhấn vào nút "Reply" vào bình luận mà mình muốn trả lời.
<b>Preconditions:</b>	Người dùng cần đăng nhập vào hệ thống để được trả lời bình luận.

<b>Postconditions:</b>	Nội dung trả lời bình luận được hiển thị.
<b>Normal flow:</b>	<ol style="list-style-type: none"> <li>1. Người dùng nhập nội dung trả lời của mình vào ô text dưới bình luận tương ứng.</li> <li>2. Người dùng nhấn "Add Reply".</li> <li>3. Nội dung trả lời được cập nhật.</li> </ol>
<b>Alternative flows:</b>	Không có
<b>Exceptions:</b>	Không có
<b>Notes and Issues:</b>	Không có

### c) Sequence Diagram



Hình 10 – Sequence Diagram Trả lời bình luận về chất lượng dịch vụ

## 3.4.2 Bình luận về chất lượng dịch vụ

### a) User Story

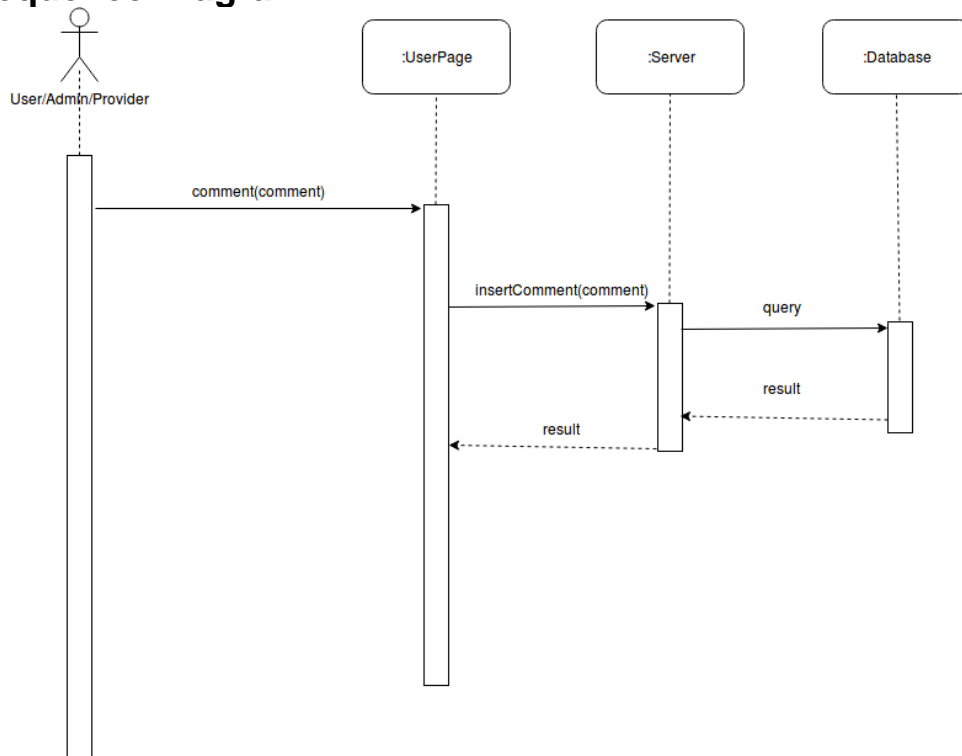
1. Người dùng nhấn vào nút "SERVICY".
2. Ở đây có ô tìm kiếm, người dùng nhập dịch vụ cần tìm kiếm và ấn nút "SEARCH".
3. Người dùng chọn loại dịch vụ mà mình quan tâm bấm vào đó để xem nội dung chi tiết.
4. Ở mục nội dung chi tiết của dịch vụ, người dùng nhấn vào nút "NHẬN XÉT" để xem các bình luận về sản phẩm.
5. Có một khung bình luận sẽ xuất hiện phía dưới cho phép người dùng bình luận sản phẩm.
6. Nhập nội dung bình luận vào khung bình luận và ấn nút "Add Comment" để gửi nội dung bình luận của mình.
7. Nội dung bình luận được cập nhật.

#### b) Use-case Scenario Tabular

<b>Use-case ID:</b>	4002
<b>Use-case name:</b>	Bình luận về chất lượng dịch vụ
<b>Created By:</b> Lương Tuấn Kiệt	<b>Last Updated By:</b> Lương Tuấn Kiệt
<b>Date Created:</b> 30/3/2019	<b>Date Last Updated:</b> 30/3/2019
<b>Actors:</b>	User/Admin/Nhà cung cấp dịch vụ
<b>Description:</b>	Người dùng để lại bình luận vào dịch vụ mà họ quan tâm.
<b>Trigger:</b>	Người dùng vào mục "NHẬN XÉT" trong trang hiện chi tiết dịch vụ.
<b>Preconditions:</b>	Người dùng cần đăng nhập vào hệ thống để bình luận.
<b>Postconditions:</b>	Nội dung bình luận được hiển thị.
<b>Normal flow:</b>	<ol style="list-style-type: none"><li>1. Người dùng nhập nội dung bình luận của mình vào ô text dưới mục bình luận.</li><li>2. Người dùng nhấn "Add Comment".</li><li>3. Nội dung bình luận được cập nhật.</li></ol>

<b>Alternative flows:</b>	Không có
<b>Exceptions:</b>	Không có
<b>Notes and Issues:</b>	Không có

### c) Sequence Diagram



Hình 11 – Sequence Diagram Bình luận về chất lượng dịch vụ

### 3.4.3 Chia sẻ dịch vụ

#### a) User Story

1. Người dùng nhấn vào nút "SERVICY".
2. Ở đây có ô tìm kiếm, người dùng nhập dịch vụ cần tìm kiếm và ấn nút "SEARCH".
3. Người dùng chọn dịch vụ mà mình quan tâm bấm vào đó để xem nội dung chi tiết.
4. Trong mục thông tin chi tiết của mỗi dịch vụ, sẽ có các nút để người dùng chia sẻ dịch vụ yêu thích của mình đến bạn bè qua mạng xã hội,
5. Người dùng chọn một trong các mạng xã hội để chia sẻ.

6. Người dùng đăng nhập vào tài khoản mạng xã hội và accept nội dung chia sẻ.

### b) Use-case Scenario Tabular

<b>Use-case ID:</b>	4003
<b>Use-case name:</b>	Chia sẻ dịch vụ
<b>Created By:</b> Lương Tuấn Kiệt	<b>Last Updated By:</b> Lương Tuấn Kiệt
<b>Date Created:</b> 30/3/2019	<b>Date Last Updated:</b> 30/3/2019
<b>Actors:</b>	User/Admin/Nhà cung cấp dịch vụ
<b>Description:</b>	Người dùng chia sẻ dịch vụ yêu thích của họ lên mạng xã hội.
<b>Trigger:</b>	Người dùng nhấn trang xem chi tiết dịch vụ.
<b>Preconditions:</b>	Người dùng có thể đăng nhập hoặc không đăng nhập
<b>Postconditions:</b>	Người dùng chia sẻ được dịch vụ tới bạn bè.
<b>Normal flow:</b>	<ol style="list-style-type: none"> <li>1. Người dùng chọn một trong số các nút tương ứng với mạng xã hội tương ứng để chia sẻ.</li> <li>2. Pop-up hiện lên yêu cầu người dùng đăng nhập vào mạng xã hội.</li> <li>3. Người dùng nhập tài khoản đăng nhập vào mạng xã hội.</li> <li>4. Người dùng nhấn accept để chia sẻ dịch vụ qua mạng xã hội.</li> </ol>
<b>Alternative flows:</b>	<p>Alternative 1: Tại bước 2</p> <p>2a. Nếu người dùng đã đăng nhập vào mạng xã hội rồi thì tới bước 4.</p>

<b>Exceptions:</b>	Không có
<b>Notes and Issues:</b>	Không có

### 3.5 Nhánh chức năng: Đăng ký tìm hiểu dịch vụ, đánh giá chất lượng dịch vụ

#### 3.5.1 Đăng ký tìm hiểu dịch vụ

##### a) User Story

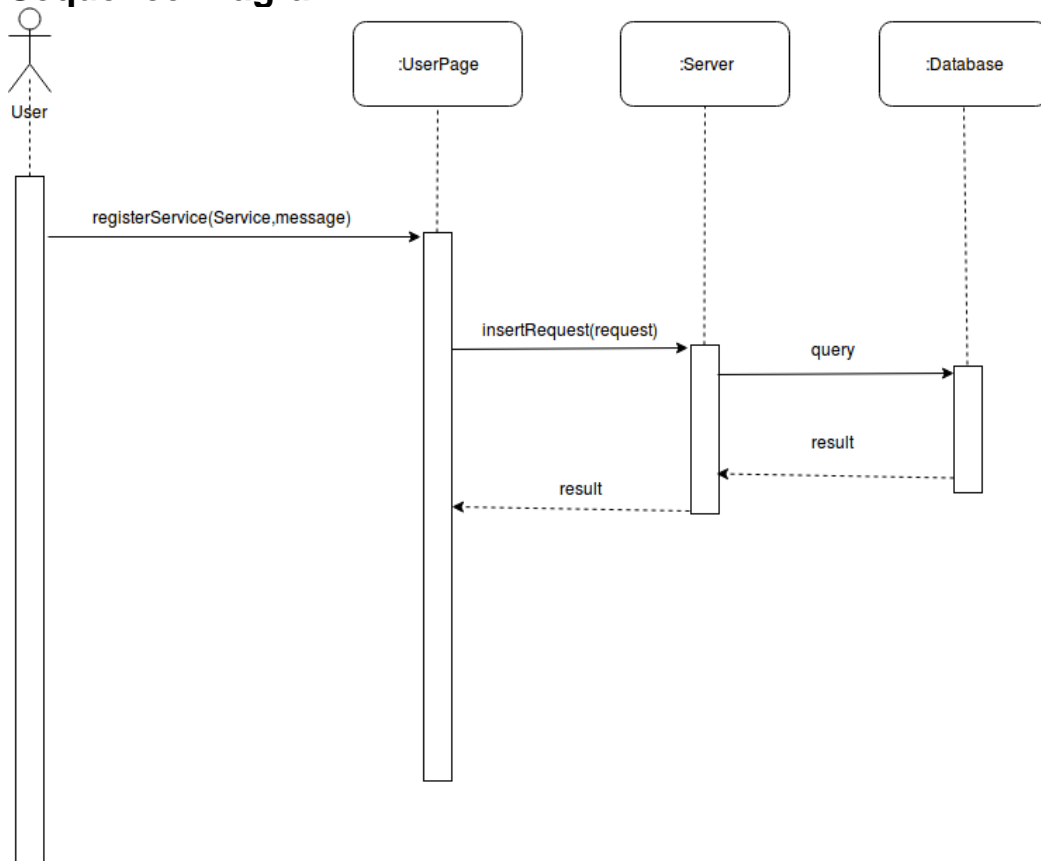
1. Khi User muốn đăng kí tìm hiểu tìm hiểu dịch vụ nào đó thì click nút “đăng kí dịch vụ” ở bản tin dịch vụ.
2. Một pop-up "Subscribe" hiển thị ra để người dùng nhập "message" gửi đến nhà cung cấp dịch vụ.
3. User nhấn nút "subscribe" để gửi yêu cầu đăng ký tìm hiểu đến nhà cung cấp dịch vụ.
4. Nhà cung cấp dịch vụ nhận được lời nhắn của User và liên hệ với User thông qua thông tin cá nhân của họ.

##### b) Use-case Scenario Tabular

<b>Use-case ID:</b>	5001
<b>Use-case name:</b>	Đăng ký đăng ký tìm hiểu dịch vụ
<b>Created By:</b> Hồ Văn Long	<b>Last Updated By:</b> Hồ Văn Long
<b>Date Created:</b> 30/3/2019	<b>Date Last Updated:</b> 30/3/2019
<b>Actors:</b>	User
<b>Description:</b>	User đăng kí tìm hiểu dịch vụ
<b>Trigger:</b>	User vào dịch vụ cần đăng kí.
<b>Preconditions:</b>	User cần đăng nhập vào hệ thống để đăng kí tìm hiểu dịch vụ.
<b>Postconditions:</b>	Thông báo kết quả đăng kí thành công.
<b>Normal flow:</b>	<ol style="list-style-type: none"> <li>1. User nhấn vào nút "đăng ký dịch vụ" trong bản tin chi tiết dịch vụ.</li> <li>2. Pop-up hiện ra yêu cầu User nhập Message.</li> <li>3. User nhập Message.</li> </ol>

	<p>4. User nhấn "subscribe" để gửi yêu cầu đi.</p> <p>5. Hệ thống thông báo yêu cầu gửi thành công.</p>
<b>Alternative flows:</b>	Không có
<b>Exceptions:</b>	<p>Exception 1: Tại bước 5</p> <p>5a. Nếu có lỗi xảy ra hệ thống sẽ báo lỗi cho người dùng biết.</p>
<b>Notes and Issues:</b>	Không có

### c) Sequence Diagram



Hình 12 – Sequence Diagram Đăng ký tìm hiểu dịch vụ

## 3.5.2 Đánh giá chất lượng dịch vụ

### a) User Story

1. User nhấn vào nút "SERVICY".
2. Ở đây có ô tìm kiếm, User nhập dịch vụ cần tìm kiếm và ấn nút

“SEARCH”.

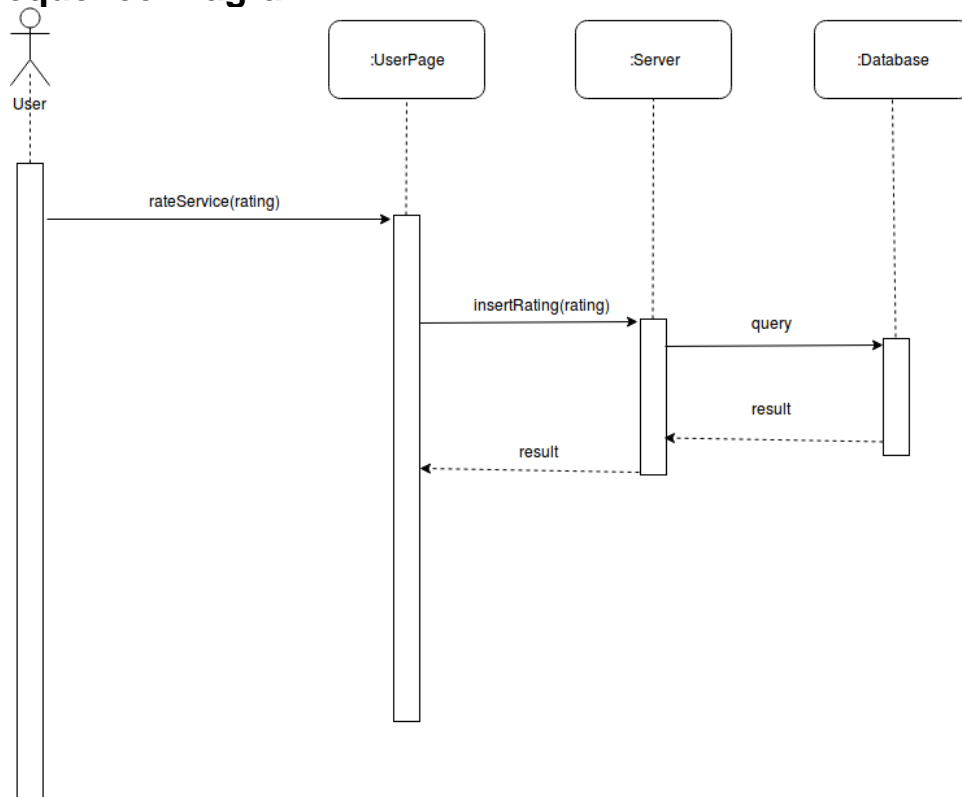
3. User chọn loại dịch vụ mà mình quan tâm bấm vào đó để xem nội dung chi tiết.
4. Ở mục nội dung chi tiết dịch vụ, một component đánh giá dạng 5 sao sẽ hiện ra, User nhấp vào để chọn số sao đánh giá và gửi yêu cầu đi
5. Đánh giá của User được cập nhật.

#### b) Use-case Scenario Tabular

<b>Use-case ID:</b>	5002
<b>Use-case name:</b>	Đánh giá chất lượng dịch vụ
<b>Created By:</b> Hồ Văn Long	<b>Last Updated By:</b> Hồ Văn Long
<b>Date Created:</b> 30/3/2019	<b>Date Last Updated:</b> 30/3/2019
<b>Actors:</b>	User
<b>Description:</b>	User đánh giá chất lượng dịch vụ mà mình sử dụng theo thang điểm 5 sao
<b>Trigger:</b>	User vào trang chi tiết dịch vụ.
<b>Preconditions:</b>	User cần đăng nhập vào hệ thống để được đánh giá
<b>Postconditions:</b>	Điểm mà User đánh giá được hiển thị.
<b>Normal flow:</b>	<ol style="list-style-type: none"><li>1. User nhấn vào component rating để đánh giá chất lượng dịch vụ theo số sao.</li><li>2. Rating của người dùng được cập nhật lên dịch vụ.</li></ol>
<b>Alternative flows:</b>	Không có
<b>Exceptions:</b>	Không có
<b>Notes and Issues:</b>	Không có



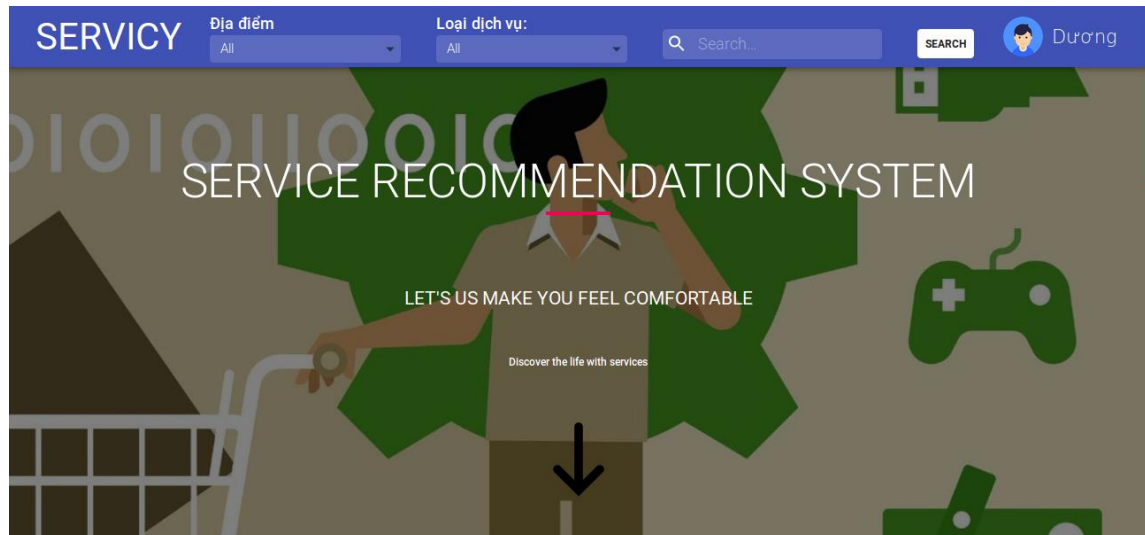
### c) Sequence Diagram



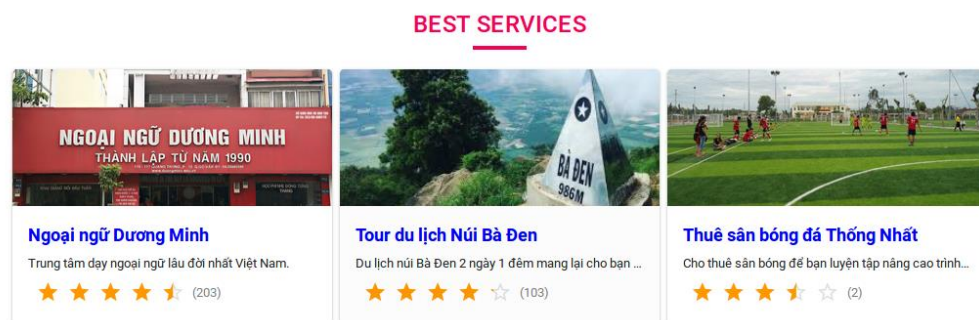
Hình 13 – Sequence Diagram Đánh giá chất lượng dịch vụ

## 4. User Interface

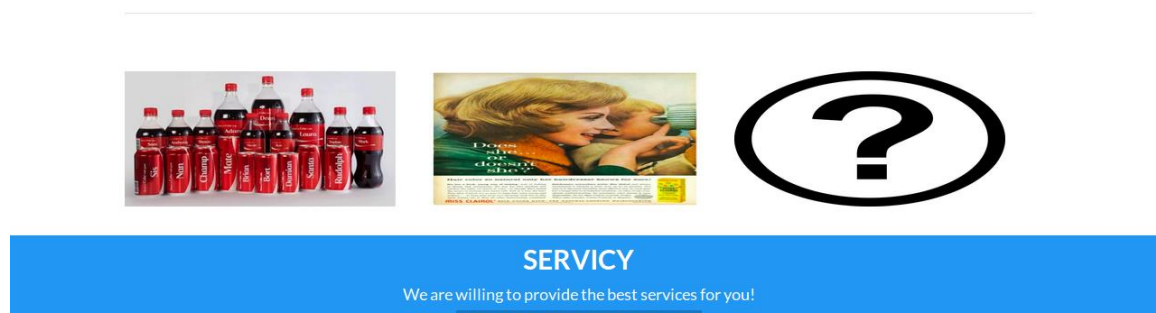
### 4.1 Nhánh chức năng: tìm kiếm, hiển thị và đề xuất



Hình 14. Trang chủ (1)



Hình 15. Trang chủ (2)



Hình 16. Trang chủ (3)

Đây là màn hình hiện ra lúc người dùng mới vào hệ thống, nó sẽ hiển thị các quảng cáo và các dịch vụ tốt nhất hiện có trên hệ thống (dựa trên các tiêu chí đánh giá) và thanh search cũng như các filter để tìm kiếm, đồng thời cũng có nút để người dùng nhấn vào để đăng nhập, đăng ký và đăng xuất khỏi hệ thống.

## Business Rules

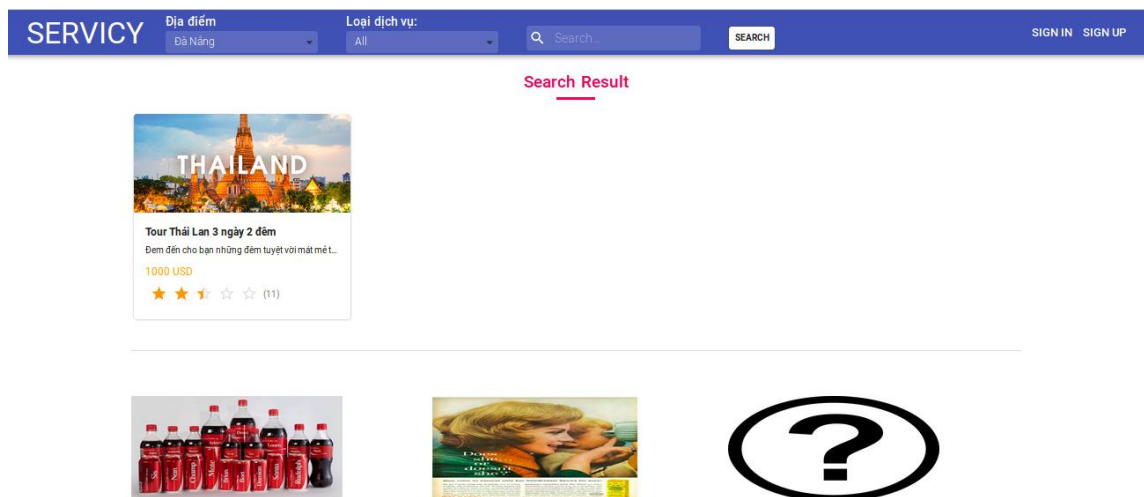
Không có.

## Acceptance Criteria

1. Người dùng có thể nhấn vào nút "SIGN IN" để đăng nhập vào hệ thống.
2. Người dùng có thể nhấn vào nút "SIGN UP" để ký tài khoản sử dụng hệ thống.
3. Người dùng có thể nhập vào ô text tìm kiếm để nhập nội dung tìm kiếm.
4. Người dùng có thể lựa chọn các lựa chọn trong các Dropdown để filter dịch vụ phục vụ cho việc tìm kiếm.
5. Người dùng có thể nhấp vào nút "SEARCH" để bắt đầu tìm kiếm dựa trên các thông tin.
6. Người dùng có thể nhấn vào các phần chứa quảng cáo để hiển thị ra chi tiết của quảng cáo.
7. Người dùng có thể nhấn vào 1 trong các dịch vụ tốt nhất để xem thông tin chi tiết của dịch vụ đó.

Item	Action	Description
Nút "SIGN IN"	Click	Một trang sẽ hiện ra yêu cầu người dùng nhập tài khoản và mật khẩu để đăng nhập.
Nút "SIGN UP"	Click	Một trang sẽ hiện ra yêu cầu người dùng nhập các thông tin cần thiết để đăng ký sử dụng hệ thống.
Text Input "Tìm kiếm"	Type	Người dùng nhập thông tin tìm kiếm vào đây.
Dropdown "Địa điểm"	Select	Người dùng chọn địa điểm để tìm kiếm dịch vụ

Dropdown "Loại dịch vụ"	Select	Người dùng chọn loại dịch vụ để tìm kiếm dịch vụ
Nút "SEARCH"	Click	Người dùng nhấn vào để bắt đầu tìm kiếm.
Ô quảng cáo	Click	Chi tiết về quảng cáo hiển thị.
Ô dịch vụ tốt nhất	Click	Chi tiết về dịch vụ tốt nhất được chọn sẽ hiển thị ra.



Hình 17. Trang Kết quả tìm kiếm

Đây là màn hình hiện ra lúc người dùng nhấn nút "SEARCH", các dịch vụ kết quả tương ứng với nội dung tìm kiếm sẽ được hiển thị ra, còn lại các phần như các ô quảng cáo, công cụ tìm kiếm, nút "SIGN IN" hay "SIGN UP" tương tự như Trang chủ.

## Business Rules

Không có.

## Acceptance Criteria

1. Người dùng có thể nhấn vào nút "SIGN IN" để đăng nhập vào hệ thống.

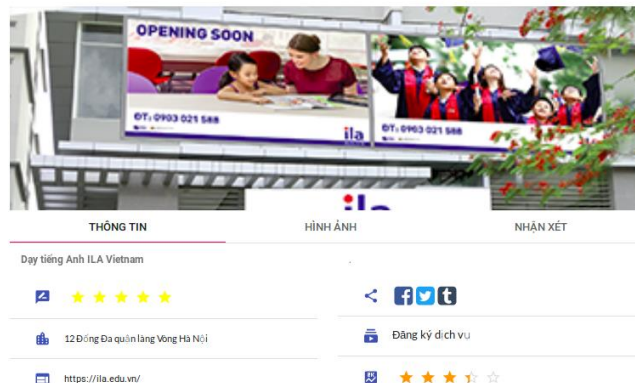
2. Người dùng có thể nhấn vào nút "SIGN UP" để ký tài khoản sử dụng hệ thống.
3. Người dùng có thể nhập vào ô text tìm kiếm để nhập nội dung tìm kiếm.
4. Người dùng có thể lựa chọn các lựa chọn trong các Dropdown để filter dịch vụ phục vụ cho việc tìm kiếm.
5. Người dùng có thể nhấp vào nút "SEARCH" để bắt đầu tìm kiếm dựa trên các thông tin.
6. Người dùng có thể nhấn vào các phần chứa quảng cáo để hiển thị ra chi tiết của quảng cáo.
7. Người dùng có thể nhấn vào 1 trong các kết quả tìm kiếm dịch vụ để xem thông tin chi tiết của dịch vụ đó.

Item	Action	Description
Nút "SIGN IN"	Click	Một popup sẽ hiện ra yêu cầu người dùng nhập tài khoản và mật khẩu để đăng nhập.
Nút "SIGN UP"	Click	Một popup sẽ hiện ra yêu cầu người dùng nhập các thông tin cần thiết để đăng ký sử dụng hệ thống.
Text Input "Tìm kiếm"	Type	Người dùng nhập thông tin tìm kiếm vào đây.
Dropdown "Địa điểm"	Select	Người dùng chọn địa điểm để tìm kiếm dịch vụ
Dropdown "Loại dịch vụ"	Select	Người dùng chọn loại dịch vụ để tìm kiếm dịch vụ
Nút "SEARCH"	Click	Người dùng nhấn vào để bắt đầu tìm kiếm.
Ô quảng cáo	Click	Chi tiết về quảng cáo hiển thị.

Ô dịch vụ kết quả	Click	Chi tiết về dịch vụ kết quả được chọn sẽ hiển thị ra.
-------------------	-------	---

## 4.2 Nhánh chức năng: Bình luận, trả lời bình luận, chia sẻ

### SERVICY



### SERVICY

We are willing to provide the best services for you!

© 2019 Copyright: servicyharokapp.com

Hình 18. Trang Hiển thị thông tin chi tiết của dịch vụ

Đây là màn hình lúc người dùng click vào một trong những dịch vụ hiện có trên hệ thống. Màn hình sẽ hiển thị chi tiết thông tin về dịch vụ mà người dùng quan tâm như: Tên dịch vụ, nhà cung cấp, số lượt rating, bình luận, đánh giá về dịch vụ,...

### Business Rules

Không có.

### Acceptance Criteria

1. Người dùng có thể nhấn vào nút "Chia sẻ" để chia sẻ dịch vụ tới bạn bè trên mạng xã hội.
2. Người dùng có thể nhấn vào nút "Đăng ký dịch vụ" để đăng ký tìm hiểu dịch vụ từ nhà cung cấp.
3. Người dùng có thể nhấn vào tab "Thông tin" để xem thêm thông tin chi tiết về dịch vụ, nhà cung cấp, các chi nhánh.
4. Người dùng có thể nhấn vào tab "Hình ảnh" để xem các hình ảnh về dịch vụ.

5. Người dùng có thể nhấn vào tab "Nhận xét" để xem các nhận xét, bình luận của người dùng về dịch vụ.

Item	Action	Description
Nút "Chia sẻ"	Click	Một pop-up sẽ hiện ra yêu cầu người dùng đăng nhập mạng xã hội để chia sẻ.
Nút "Đăng ký dịch vụ"	Click	Một pop-up hiện ra yêu cầu người dùng nhập message muốn gửi đến nhà cung cấp dịch vụ.
Tab "Thông tin"	Click	Nội dung bên dưới thanh tab bar sẽ thị nội dung chi tiết về thông tin dịch vụ như địa chỉ dịch vụ, website của dịch vụ,vv...
Tab "Hình ảnh"	Click	Nội dung bên dưới thanh tab bar sẽ hiển thị các hình ảnh liên quan tới dịch vụ dưới dạng slide.
Tab "Nhận xét"	Click	Nội dung bên dưới thanh tab bar sẽ hiển thị các nội dung bình luận về dịch vụ, cũng như các câu trả lời cho các bình luận đó.


## Comments


 **Dương** 25/05/2019 19:32:13  
alo  
[Reply](#)


 **Dương**  
helu

 **Dương** 25/05/2019 19:51:55  
alo alo  
[Reply](#)

Add comments here...

 **Add Reply**

 **Zalo Group** 26/05/2019 21:53:15  
Hay ghé tôi đang muốn đi Thái  
[Reply](#)

 **Add Comment**

Hình 19. Bình luận/Trả lời bình luận

Đây là nội dung bên dưới thanh tab bar lúc người dùng click vào tab "Nhận xét". Màn hình này sẽ cho phép người dùng xem bình luận, bình luận và trả lời nội dung bình luận về dịch vụ.

## Business Rules

Không có.

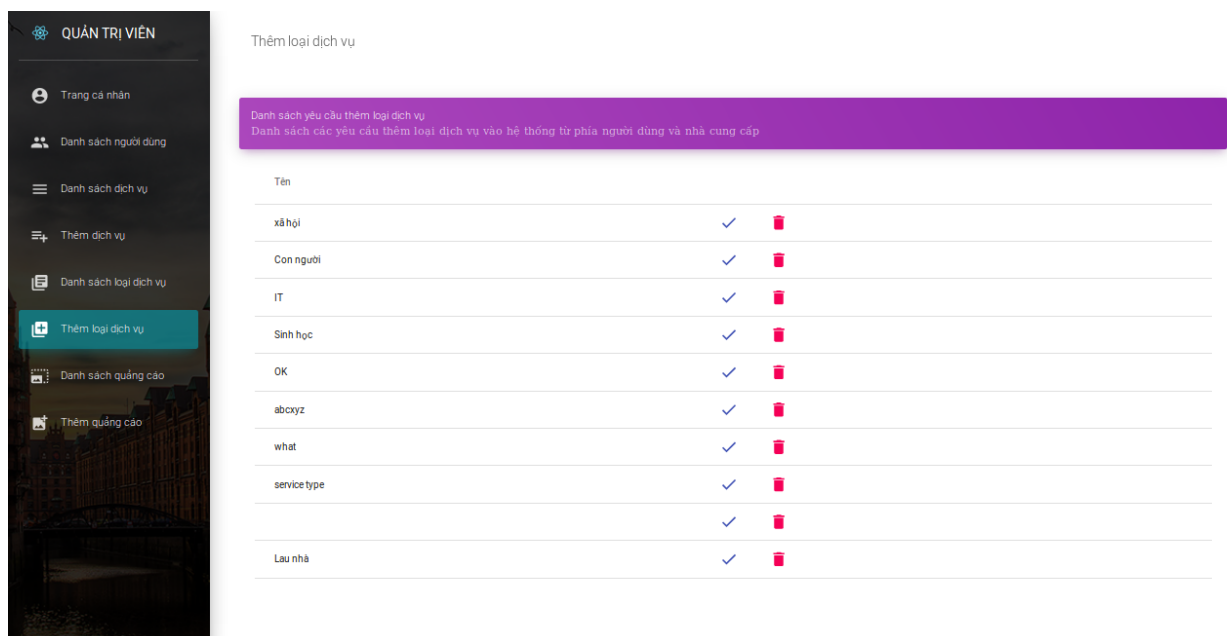
## Acceptance Criteria

1. Người dùng có thể gõ nội dung bình luận vào ô trống hiện ra và ấn "Add Comment" để bình luận.
2. Người dùng có thể nhấn nút "Reply" để trả lời cho một bình luận nào đó trước đó.
3. Người dùng có thể nhấn nút "Add Reply" để gửi nội dung trả lời bình luận.



Item	Action	Description
Nút "Add Comment"	Click	Nội dung bình luận được gửi đi.
InputText "Bình luận"	Type	Người dùng nhập nội dung muốn bình luận.
Nút "Add Reply"	Click	Nội dung trả lời bình luận được gửi đi.
InputText "Reply"	Click	Người dùng nhập nội dung trả lời bình luận vào.
Nút "Reply"	Click	Một ô trống bên dưới bình luận sẽ được hiện ra để người dùng nhập nội dung trả lời bình luận.

### 4.3 Nhánh chức năng: Thêm mới loại dịch vụ, dịch vụ





Hình 20. Các yêu cầu thêm mới loại dịch vụ



Là một quản trị viên (admin) của trang web, tôi muốn xét duyệt các yêu cầu thêm mới loại dịch vụ gửi về từ phía User và nhà cung cấp dịch vụ.

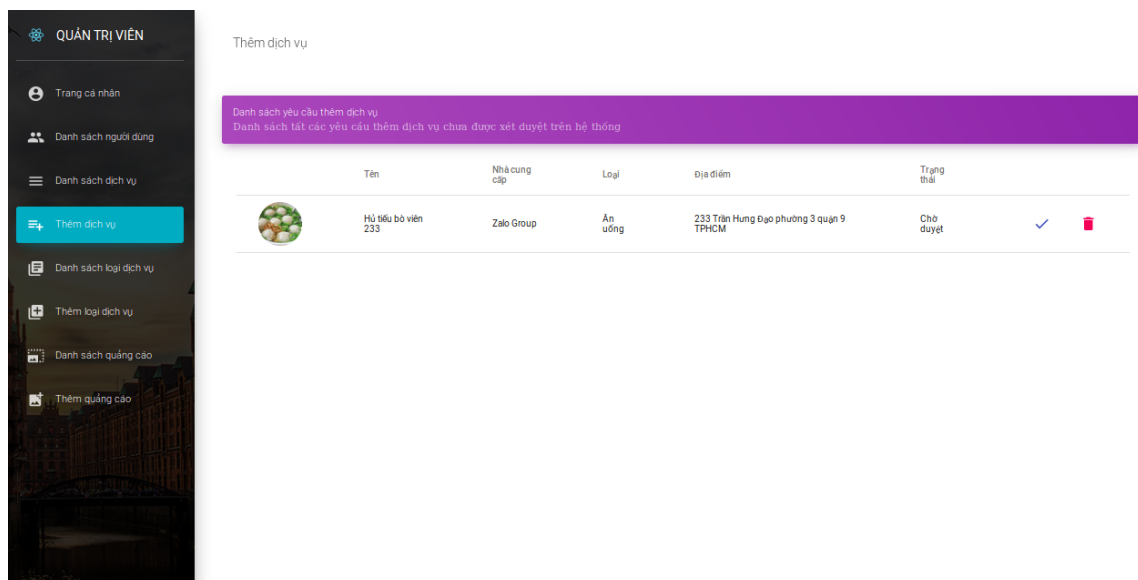
## Business Rules

Không có.

## Acceptance Criteria

- Admin chấp nhận các yêu cầu thêm mới cần xử lý bằng cách click vào các ô  bên cạnh mỗi yêu cầu.
- Admin từ chối các yêu cầu thêm mới cần xử lý bằng cách click vào các ô  bên cạnh mỗi yêu cầu.

Item	Action	Description
Nút 	Click	Chấp nhận yêu cầu thêm mới loại dịch vụ.
Nút 	Click	Từ chối yêu cầu thêm mới loại dịch vụ.





Hình 21. Các yêu cầu thêm mới dịch vụ



Là một quản trị viên (admin) của trang web, tôi muốn xét duyệt các yêu cầu thêm mới dịch vụ gửi về từ phía nhà cung cấp dịch vụ.

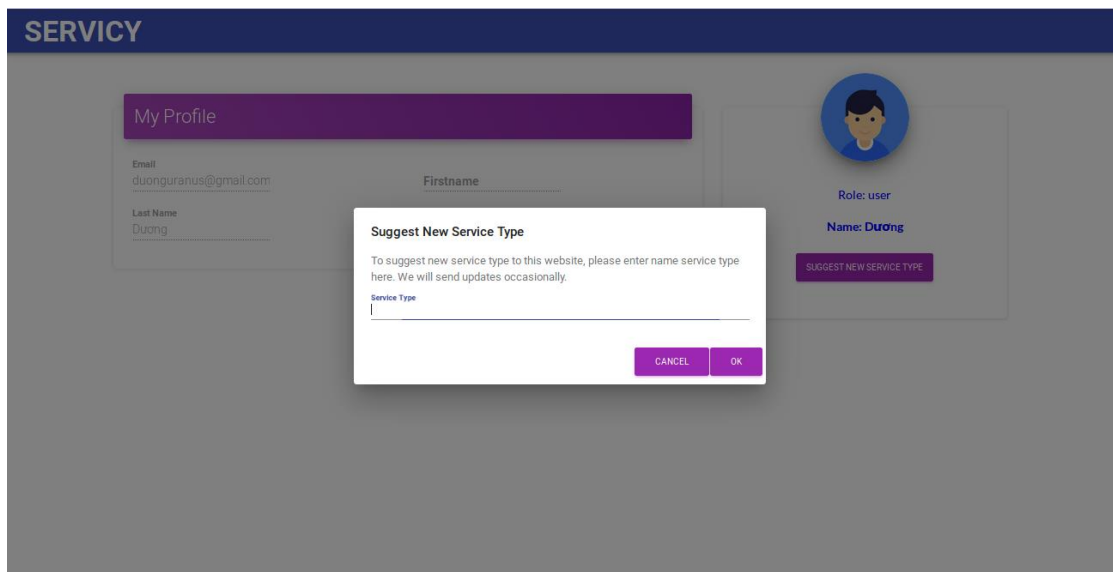
## Business Rules

Không có.

## Acceptance Criteria

1. Admin chấp nhận các yêu cầu thêm mới cần xử lý bằng cách click vào các ô  bên cạnh mỗi yêu cầu.
2. Admin từ chối các yêu cầu thêm mới cần xử lý bằng cách click vào các ô  bên cạnh mỗi yêu cầu.

Item	Action	Description
Nút 	Click	Chấp nhận yêu cầu thêm mới dịch vụ.
Nút 	Click	Từ chối yêu cầu thêm mới dịch vụ.



Hình 22. Yêu cầu thêm một loại dịch vụ mới (User)

Là một User, tôi muốn gửi yêu cầu thêm mới loại dịch vụ đến admin.

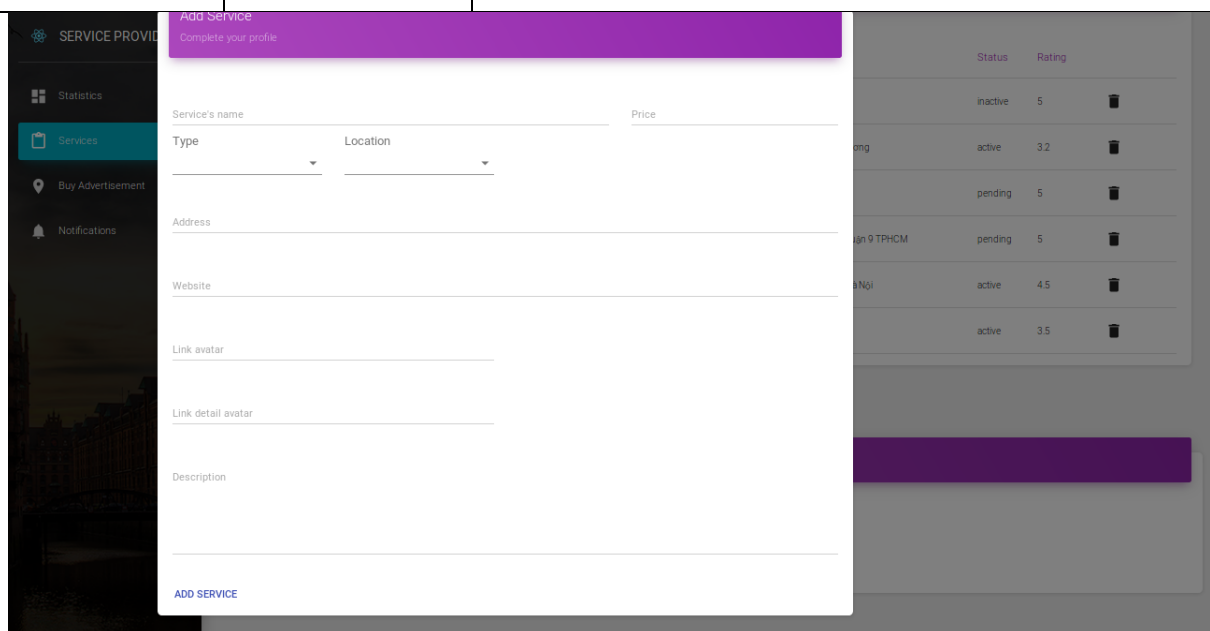
## Business Rules

Không có.

## Acceptance Criteria

1. User có thể nhập tên loại dịch vụ muốn thêm vào InputText "Service Type".
2. User có thể nhấn nút "OK" để gửi yêu cầu.
3. User có thể nhấn nút "CANCEL" để thoát.

Item	Action	Description
InputText "Service Type"	Type	User nhập tên loại dịch vụ muốn yêu cầu thêm vào.
Nút "OK"	Click	Gửi yêu cầu thêm mới loại dịch vụ cho admin.
Nút "CANCEL"	Click	Hủy quá trình yêu cầu thêm mới loại dịch vụ.



Hình 23: Thêm mới một dịch vụ mới

Là một nhà cung cấp dịch vụ, tôi có nhu cầu hoặc kinh doanh một dịch vụ mới trên trang web và muốn gửi yêu cầu thêm mới dịch vụ đến admin.

## Business Rules

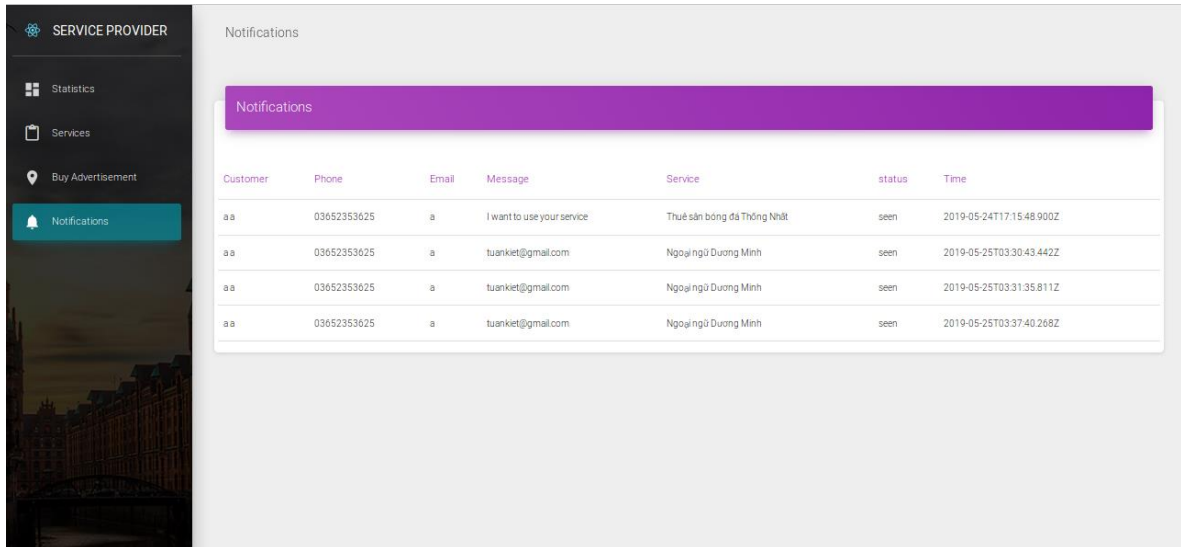
Không có.

### Acceptance Criteria

1. Nhà cung cấp dịch vụ có thể bấm vào thanh dropdown để chọn loại dịch vụ, location của dịch vụ.
2. Nhà cung cấp dịch vụ có thể nhập vào các InputText thông tin còn lại để nhập thông tin về dịch vụ.

Item	Action	Description
Dropdown "Service Type"	Select	Chọn loại dịch vụ cho dịch vụ muốn thêm.
Dropdown "Service Type"	Select	Chọn địa điểm cho dịch vụ muốn thêm.
Các InputText nhập thông tin	Type	Nhập các thông tin còn lại về dịch vụ.

#### 4.4 Nhánh chức năng: Đăng ký tìm hiểu dịch vụ, đánh giá chất lượng dịch vụ



The screenshot shows a web application interface for a service provider. On the left is a dark sidebar with navigation links: SERVICE PROVIDER, Statistics, Services, Buy Advertisement, and Notifications (highlighted). The main content area is titled 'Notifications' and contains a table with the following data:

Customer	Phone	Email	Message	Service	status	Time
aa	03652353625	a	I want to use your service	Thuê sân bóng đá Thống Nhất	seen	2019-05-24T17:15:48.900Z
aa	03652353625	a	tuankiet@gmail.com	Ngoại ngữ Dương Minh	seen	2019-05-25T03:30:43.442Z
aa	03652353625	a	tuankiet@gmail.com	Ngoại ngữ Dương Minh	seen	2019-05-25T03:31:35.811Z
aa	03652353625	a	tuankiet@gmail.com	Ngoại ngữ Dương Minh	seen	2019-05-25T03:37:40.268Z

Hình 24. Thông tin người theo dõi dịch vụ

Đây là màn hình hiện ra lúc nhà cung cấp dịch vụ xem thông tin người dùng đăng ký tìm hiểu dịch vụ, nó sẽ hiển thị thông tin về email, số điện thoại, thời gian đăng kí, vv của người dùng về dịch vụ của họ.

### Business Rules

Không có.

### Acceptance Criteria

Không có.

MC Donald



Hình 25. Component để người dùng đánh giá bằng số sao

Trong màn hình hiển thị chi tiết dịch vụ, có một component để người dùng đánh giá dịch vụ theo số sao.

### Business Rules

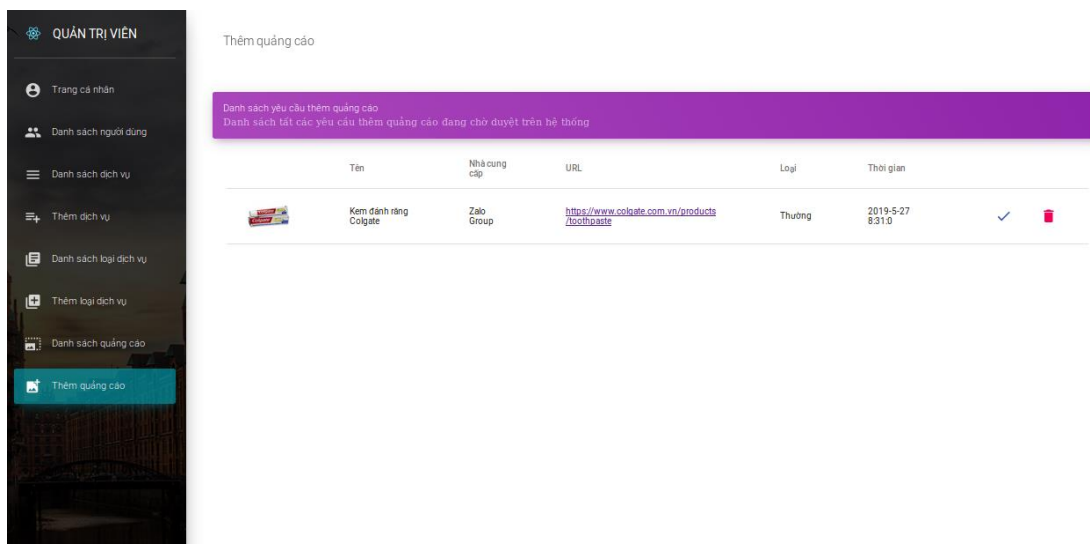
Không có.

### Acceptance Criteria

1. Người dùng có thể nhấn vào component theo số sao để đánh giá dịch vụ.

Item	Action	Description
Component "Rating"	Click	Người dùng sẽ chọn từ 1 sao đến 5 sao để thể hiện mức độ hài lòng.



## 4.5 Nhánh chức năng: hiển thị quảng cáo, đăng ký quảng cáo và theo dõi quảng cáo





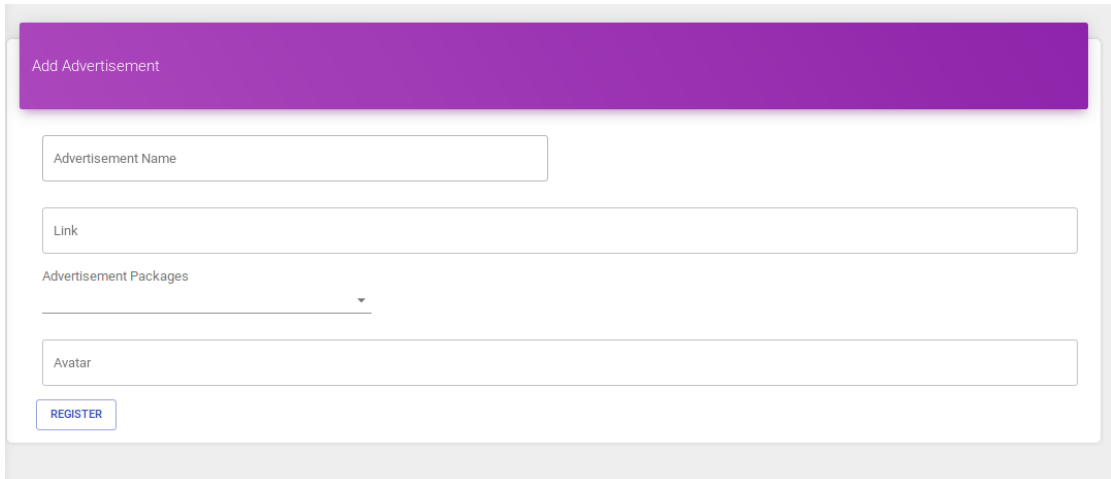
Hình 26. Accept quảng cáo

Đây là màn hình khi admin hệ thống đã đăng nhập và kiểm tra các yêu cầu đăng ký quảng cáo mới.

### Acceptance Criteria

1. Admin có thể nhấn nút , để cho phép quảng cáo đã chọn được chạy.
2. Admin có thể nhấn nút , để xóa yêu cầu đăng ký quảng cáo.

Item	Action	Description
Nút 	Click	Cho phép các quảng cáo được chọn được chạy trên hệ thống.
Nút 	Click	Xóa yêu cầu đăng ký quảng cáo



Hình 27. Mua quảng cáo

Đây là màn hình khi nhà cung cấp dịch vụ đã đăng nhập và muốn mua quảng cáo bằng cách lựa chọn mục “Buy Advertisement” ở thanh điều hướng.

## Business Rules

Không có.

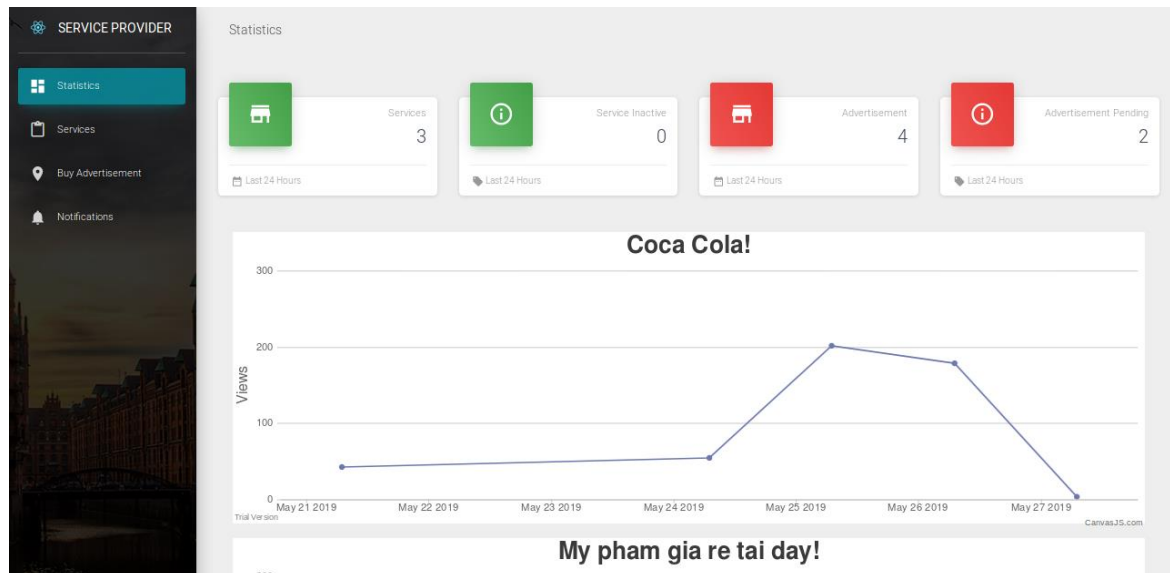
## Acceptance Criteria

1. Nhà cung cấp dịch vụ nhập vào các InputText về thông tin của quảng cáo mà mình muốn đăng ký.
2. Nhà cung cấp dịch vụ chọn Dropdown "Advertisement Packages" của quảng cáo mà mình muốn đăng ký.
3. Nhà cung cấp dịch vụ nhấn nút "REGISTER" để gửi yêu cầu đăng ký quảng cáo đến Admin.

Item	Action	Description
------	--------	-------------



Dropdown "Advertisem ent Packages"	Select	Chọn loại quảng cáo muốn đăng ký (Normal/Enterprise)
Các InputText thông tin	Type	Nhập các thông tin về quảng cáo như là avatar, link,vv...
Nút "REGISTER "	Click	Nhấn vào để gửi yêu cầu đăng ký quảng cáo lên Admin.



Hình 28. Theo dõi quảng cáo

Đây là màn hình khi nhà cung cấp dịch vụ đã đăng nhập và theo dõi các quảng cáo đã mua.

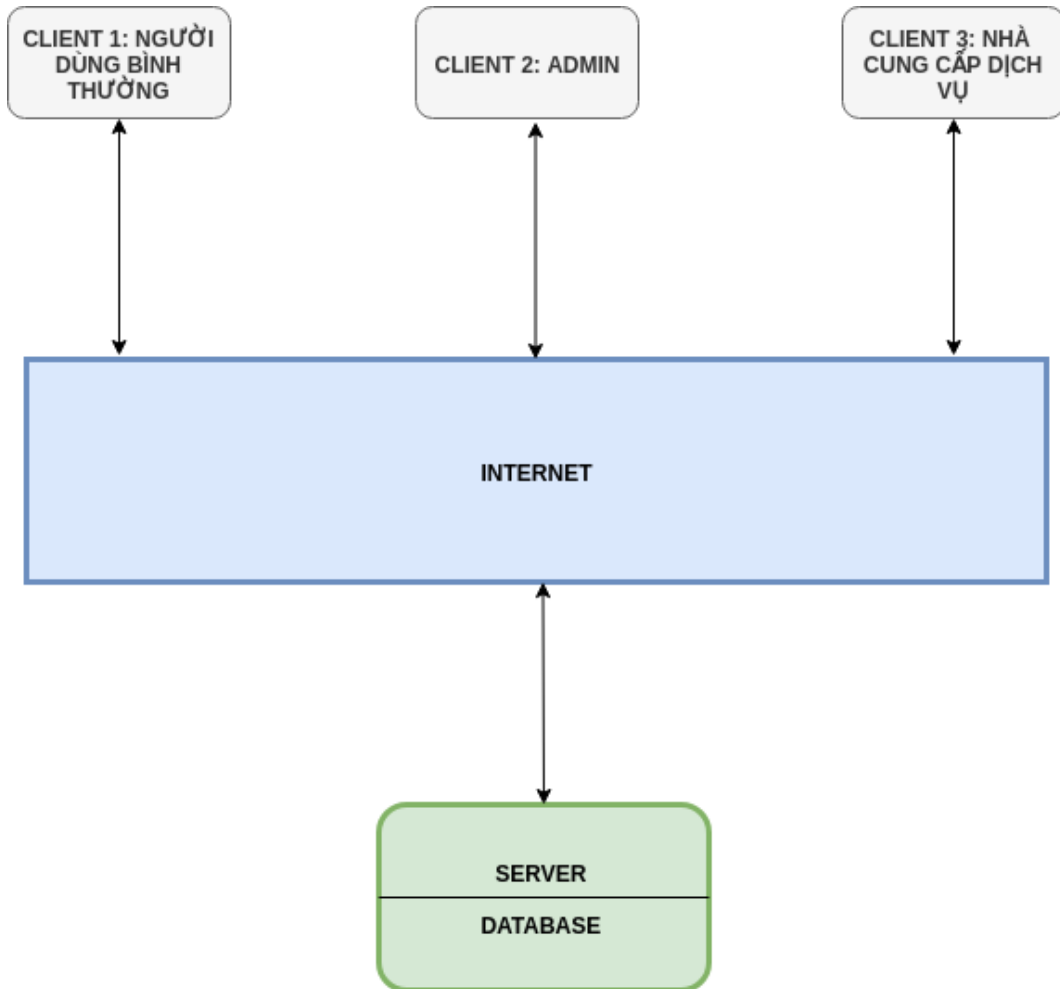
## Business Rules

### Acceptance Criteria

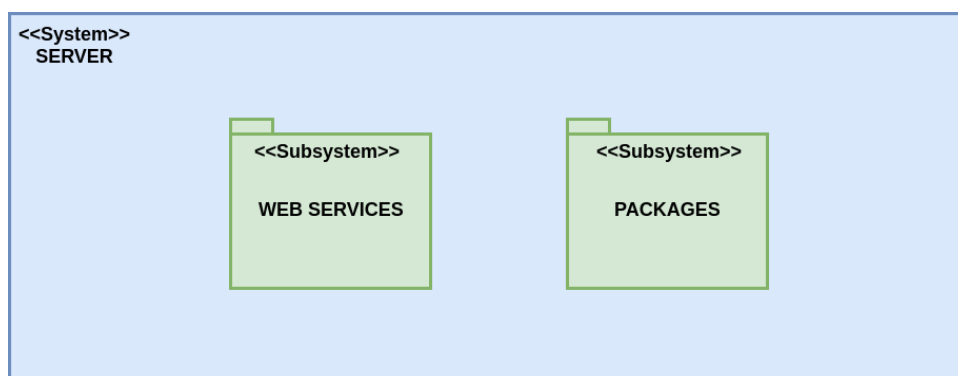
1. Nhà cung cấp theo dõi biểu đồ của từng quảng cáo đã được admin cho chạy của mình theo số View và số Click.

## 5. Kiến trúc hệ thống

### 5.1 Thiết kế kiến trúc



Hình 29. Kiến trúc tổng quan hệ thống



Hình 30. Kiến trúc bên trong Server

## 5.2 Mô tả kiến trúc

### 5.2.1 Người dùng bình thường (User)

**Name:** Người dùng bình thường

**Type:** Client

**Node:** Không có

**Description:** Người dùng bình thường có thể sử dụng email và password của mình để đăng nhập và sử dụng hệ thống, đồng thời có thể yêu cầu thêm 1 loại dịch vụ mới, bình luận dịch vụ, trả lời bình luận, đăng ký theo dõi dịch vụ, gửi phản hồi dịch vụ và tìm kiếm.

**Attributes:**

- id
- email
- password
- firstname
- lastname
- phone
- avatar

**Resources:** Server và Database.

**Operations:**

- sendRequestAdKindService(kind)
- comment(comment)
- replyComment(reply)
- registerService(Service,message)
- search(filter)
- viewAd()
- viewService(id)
- viewBestService()
- rateService(rating)
- changeInfo(info)

### 5.2.2 Admin

**Name:** Admin

**Type:** Client

**Node:** Không có

**Description:** Admin có thể dùng email và password của mình để đăng nhập vào hệ thống, ngoài ra còn có thể xóa người dùng, sửa vai trò của người dùng, xóa/chấp thuận/ từ chối dịch vụ, chấp thuận/ từ chối loại dịch vụ từ User/Nhà cung cấp dịch vụ.

**Attributes:**

- id
- email
- password
- firstname
- lastname
- phone
- avatar

**Resources:** Server và Database.

**Operations:**

- addUser(user)
- deleteUser(id)
- deleteService(id)
- acceptService(id)
- refuseService(id)
- acceptKindService(id)
- refuseKindService(id)
- acceptAd(id)
- refuseAd(id)
- comment(comment)
- replyComment(reply)
- changeInfo(info)



### 5.2.3 Provider

**Name:** Provider

**Type:** Client

**Node:** Không có

**Description:** Provider (nhà cung cấp dịch vụ) có thể dùng email và password của mình để đăng nhập vào hệ thống, ngoài ra còn có thể đăng ký quảng cáo, lấy danh sách các user đăng ký theo dõi dịch vụ, thêm dịch vụ, loại dịch vụ, bình luận, trả lời bình luận cũng như lấy thông tin của quảng cáo.

**Attributes:**

- id
- email
- password
- firstname
- lastname
- phone
- avatar

**Resources:** Server và Database.

**Operations:**

- buyAdvertise(advertise)
- getInterestingCustomer()
- addService(service)
- replyComment(reply)
- getAdvertiseInfo(advertiselId)
- comment(comment)
- viewAdvertise()
- viewService()
- sendRequestAdKindService(kind)



- changInfo(info)

#### 5.2.4 Server

**Name:** Server

**Type:** System

**Node:** Không có

**Description:** Server dựa vào request của User gửi mà query vào Database để lấy thông tin cần thiết, bao gồm các tác vụ thêm/xóa/sửa thông tin User, quảng cáo (Advertise), dịch vụ (Service), loại dịch vụ (Service Type) và lấy tất cả các User/Advertise/Service hiện có trong dịch vụ, đồng thời lấy các dịch vụ tốt nhất hiện có trên hệ thống.

**Attributes:** Không có

**Resources:** Database.

**Operations:**

- getView(adId)
- insertView(view)
- getService(id)
- getService(condition)
- updateService(service)
- insertService(service)
- deleteService(service)
- getRequest(id)
- getRequest(providerId)
- insertRequest(request)
- getReply(commentId)



- insertReply(reply)
- insertRating(rating)
- getAllLocations()
- getLocation(id)
- getComment(serviceId)
- insertComment(comment)
- getClick(adId)
- insertClick(Click)
- getAllAdtype()
- getUser(id)
- updateUser(user)
- insertUser(user)
- deleteUser(id)
- getAdvertise(name)
- updateAdvertise(name)
- insertAdvertise(advertise)
- deleteAdvertise(id)
- getAllUser()
- getAllService()
- getAllAdvertise()
- getBestService()
- getRandomAdvertise()
- getAdvertiseByProvider(provider)
- insertKindOfService(servicetype)



- deleteKindOfService(servicetype)
- updateKindOfService(servicetype)
- getAllKindOfService()
- getRating(serviceId)

### 5.2.5 Database

**Name:** Database

**Type:** System

**Node:** Không có

**Description:**

**Attributes:** Không có

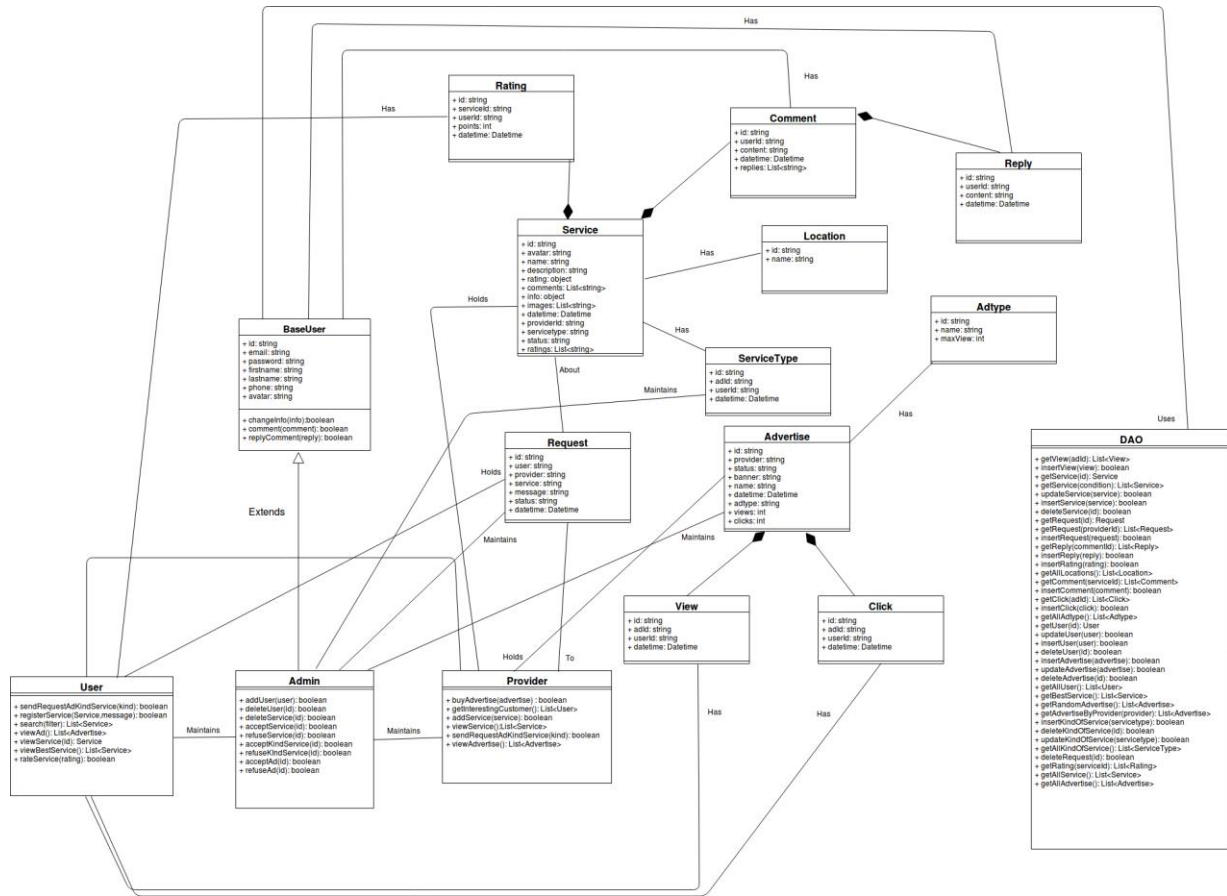
**Resources:** Không có.

**Operations:** thêm/xóa/sửa/select các bảng dữ liệu

## 6. Module Design

### 6.1 Class Diagram





Hình 31. Class Diagram hệ thống

## 6.2 Method Specification

### 6.2.1 BaseUser Class

## changeInfo

- Mục đích: thay đổi thông tin của user trong bảng users.
- Prototype: public boolean changeInfo(object info)
- Input:

info: chứa thông tin thay đổi.

- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi người dùng muốn thay đổi thông tin cá nhân.
- Cách gọi: `.changeInfo(info)`
- Algorithm:

- Lấy các trường thông tin mới trong object info thêm vào đối tượng user mới.
- Gọi hàm `.updateUser(user)`.
- Nếu thành công trả về True, ngược lại False.

## **comment**

- Mục đích: thêm một comment từ phía userpage.
- Prototype: `public boolean comment(Comment comment)`
- Input:

comment: đối tượng Comment muốn thêm.

- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi người dùng muốn bình luận bài viết nào đó.
- Cách gọi: `.comment(obj, str)`
- Algorithm:
  - Gọi hàm `insertComment(comment)`.
  - Nếu thành công trả về True, ngược lại False.

## **replyComment**

- Mục đích: thêm một reply từ phía userpage.
- Prototype: `public boolean replyComment(Reply reply)`
- Input:

reply: đối tượng Reply muốn thêm.

- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi người dùng muốn trả lời bình luận.
- Cách gọi: `.replyComment(reply)`
- Algorithm:
  - Gọi hàm `insertReply(reply)`.
  - Nếu thành công trả về True, ngược lại False.

## 6.2.2 User Class

### **sendRequestAdKindService**

- Mục đích: thêm một servicetype vào bảng ServiceType từ userpage hoặc providerpage.
- Prototype: public boolean sendRequestAdKindService(String kind)
- Input:  
  
kind: loại dịch vụ muốn đề xuất.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi User muốn đề xuất loại dịch vụ.
- Cách gọi: .sendRequestAdKindService(kind)
- Algorithm:
  - Tạo đối tượng servicetype với kind.
  - Gọi hàm .insertKindOfService(servicetype).
  - Nếu thành công trả về True, ngược lại False.

### **registerService**

- Mục đích: thêm một servicetype vào bảng ServiceType từ userpage hoặc providerpage.
- Prototype: public boolean registerService(Service Service,String message)
- Input:  
  
Service: dịch vụ muốn đăng ký tìm hiểu.  
  
message: thông điệp muốn gửi đến.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi User muốn đăng ký dịch vụ.
- Cách gọi: .registerService(Service)
- Algorithm:
  - Tạo đối tượng Request với message và Service.

- Gọi hàm `.insertRequest(request)`.
- Nếu thành công trả về `True`, ngược lại `False`.

## **search**

- Mục đích: tìm List các dịch vụ thỏa yêu cầu.
- Prototype: `public List<Service> search(object filter)`
- Input:  
  
filter: object chứa các thông tin tìm kiếm.
- Output: List các Service thỏa yêu cầu.
- Gọi khi: khi người dùng muốn tìm kiếm dịch vụ.
- Cách gọi: `.search(filter)`
- Algorithm:
  - Gọi hàm `.getService(filter)`.
  - Thêm vào List và trả về.

## **viewAd**

- Mục đích: gọi các quảng cáo ngẫu nhiên .
- Prototype: `public List<Advertise> viewAd()`
- Input: Không có.
- Output: List các quảng cáo trả về.
- Gọi khi: khi User xem quảng cáo ở userpage.
- Cách gọi: `.viewAd()`
- Algorithm:
  - Gọi hàm `.getRandomAdvertise()`.
  - Thêm vào List và trả về.

## **viewService**

- Mục đích: lấy Service dựa vào id.
- Prototype: `public Service viewService(String id)`
- Input:

id: id của Service cần lấy.

- Output: đối tượng Service tương ứng id.
- Gọi khi: khi người dùng muốn xem chi tiết dịch vụ nhấn vào.
- Cách gọi: `.viewService(id)`
- Algorithm:
  - Gọi hàm `.getService(id)`.
  - Trả về đối tượng Service.

### **viewBestService**

- Mục đích: lấy các Service tốt nhất.
- Prototype: `public List<Service> viewBestService()`
- Input: Không có.
- Output: List các Service tốt nhất.
- Gọi khi: khi người dùng xem các dịch vụ tốt nhất.
- Cách gọi: `.viewBestService()`
- Algorithm:
  - Gọi hàm `.getBestService()`.
  - Trả về List kết quả.

### **rateService**

- Mục đích: thêm đối tượng rating mới.
- Prototype: `public boolean rateService(Rating rating)`
- Input:

rating: đối tượng Rating cần thêm vào.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi User đánh giá dịch vụ bằng số sao.
- Cách gọi: `.rateService(rating)`
- Algorithm:
  - Gọi hàm `.insertRating(rating)`.
  - Nếu thành công trả về `True`, ngược lại `False`.

### 6.2.3 Admin Class

#### **addUser**

- Mục đích: thêm mới một người dùng.
- Prototype: `public boolean addUser(User user)`.
- Input: class `User`: chứa thông tin cần thiết về `User`.
- Output: `True` nếu thêm thành công, `False` ngược lại.
- Gọi khi: `User` đăng ký tham gia hệ thống.
- Cách gọi: `.addUser(user)`.
- Algorithm:
  - Gọi hàm `.insertUser(user)`.
  - Nếu thành công trả về `True`, ngược lại `False`.

#### **deleteUser**

- Mục đích: xóa một người dùng.
- Prototype: `public boolean deleteUser(String id)`
- Input:  
id: id của `User` muốn xóa.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: admin muốn xóa tài khoản người dùng.
- Cách gọi: `.deleteUser(id)`
- Algorithm:
  - Gọi hàm `.deleteUser(id)`.
  - Nếu thành công trả về `True`, ngược lại `False`.

#### **deleteService**

- Mục đích: xóa một dịch vụ.
- Prototype: `public boolean deleteService(String id)`
- Input:  
id: id của `Service` muốn xóa.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: admin muốn xóa bỏ một dịch vụ ra khỏi hệ thống.
- Cách gọi: `.deleteService(id)`



- Algorithm:
  - Gọi hàm `.deleteService(id)`.
  - Nếu thành công trả về `True`, ngược lại `False`.

### **acceptService**

- Mục đích: chấp nhận thêm một dịch vụ
- Prototype: `public boolean acceptService(String id)`
- Input:  
id: id của dịch vụ muốn chấp thuận.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: admin muốn thêm một dịch vụ được đề xuất.
- Cách gọi: `.acceptService(id)`.
- Algorithm:
  - Tạo đối tượng mới service với status là "active".
  - Gọi hàm `updateService(service)`.
  - Nếu thành công trả về `True`, ngược lại `False`.

### **refuseService**

- Mục đích: từ chối thêm một dịch vụ
- Prototype: `public boolean refuseService(String id)`
- Input:  
id: id của dịch vụ muốn từ chối.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: admin muốn từ chối một dịch vụ được đề xuất.
- Cách gọi: `.refuseService(id)`.
- Algorithm:
  - Tạo đối tượng mới service với status là "inactive".
  - Gọi hàm `updateService(service)`.
  - Nếu thành công trả về `True`, ngược lại `False`.

## **acceptKindService**

- Mục đích: chấp nhận thêm một loại dịch vụ
- Prototype: `public boolean acceptKindService(String id)`
- Input:  
id: id của loại dịch vụ muốn chấp thuận.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: admin muốn chấp thuận một loại dịch vụ được đề xuất.
- Cách gọi: `.acceptKindService(id)`.
- Algorithm:
  - Tạo đối tượng mới `servicetype` với status là "active".
  - Gọi hàm `.updateKindOfService(servicetype)`.
  - Nếu thành công trả về True, ngược lại False.

## **refuseKindService**

- Mục đích: từ chối thêm một loại dịch vụ
- Prototype: `public boolean refuseKindService(String id)`
- Input:  
id: id của loại dịch vụ muốn từ chối.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: admin muốn từ chối một loại dịch vụ được đề xuất.
- Cách gọi: `.refuseKindService(id)`.
- Algorithm:
  - Tạo đối tượng mới `servicetype` với status là "inactive".
  - Gọi hàm `.updateKindOfService(servicetype)`.
  - Nếu thành công trả về True, ngược lại False.

## **acceptAd**

- Mục đích: chấp thuận thêm quảng cáo.
- Prototype: `public boolean acceptAd(String id)`
- Input:



id: id của quảng cáo muốn chấp thuận.

- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: admin muốn chấp nhận một quảng cáo được đề xuất.
- Cách gọi: `.acceptAd(id)`.
- Algorithm:
  - Tạo đối tượng mới advertise với status là "running".
  - Gọi hàm `.updateAdvertise(advertise)`.
  - Nếu thành công trả về True, ngược lại False.

### **refuseAd**

- Mục đích: từ chối thêm quảng cáo.
- Prototype: `public boolean refuseAd(String id)`
- Input:

id: id của quảng cáo muốn từ chối.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: admin muốn từ chối một quảng cáo được đề xuất.
- Cách gọi: `.refuseAd(id)`.
- Algorithm:
  - Tạo đối tượng mới advertise với status là "inactive".
  - Gọi hàm `.updateAdvertise(advertise)`.
  - Nếu thành công trả về True, ngược lại False.

## **6.2.4 DAO Class**

### **getView**

- Mục đích: truy xuất view của một advertise thông qua id.
- Prototype: `public List<View> getView(String adId)`
- Input:

adId: Id của advertise cần lấy view.

- Output: Danh sách các view của advertise.
- Gọi khi: khi Provider muốn xem view của quảng cáo.
- Cách gọi: . getView(adId)
- Algorithm:
  - Truy xuất trong bảng view các View mà có adId bằng với adId.
  - Thêm vào List và trả về.

### **insertView**

- Mục đích: thêm vào bảng view một đối tượng view mới.
- Prototype: public boolean insertView(View view)
- Input:

view: đối tượng View cần thêm vào.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi người dùng nhấn vào quảng cáo tương ứng.
- Cách gọi: . insertView(view)
- Algorithm:
  - Thêm đối tượng view mới vào bảng view.
  - Nếu thành công return True, ngược lại False.

### **getService**

- Mục đích: truy xuất một Service thông qua id của nó.
- Prototype: public Service getService(String id)
- Input:

id: Id của service cần lấy thông tin.
- Output: Đối tượng Service.
- Gọi khi: khi người dùng nhấn vào một service cụ thể.
- Cách gọi: . getService(id)
- Algorithm:
  - Truy xuất trong bảng services Service có id tương ứng.



- Trả về đối tượng Service.

### **getService**

- Mục đích: truy xuất các Service thỏa điều kiện condition.
- Prototype: public List<Service> getService(Object condition)
- Input:  
  
condition: object chứa các điều kiện tìm kiếm.
- Output: Danh sách các Service thỏa điều kiện tìm kiếm.
- Gọi khi: khi người dùng muốn tìm kiếm dịch vụ.
- Cách gọi: .getService(condition)
- Algorithm:
  - Truy xuất trong bảng services các Service thỏa điều kiện trong condition.
  - Thêm vào List và trả về.

### **updateService**

- Mục đích: cập nhật một Service.
- Prototype: public boolean updateService(Service service)
- Input:  
  
service: đối tượng Service mới cần update.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi Admin muốn cập nhật dịch vụ.
- Cách gọi: .updateService(service)
- Algorithm:
  - Cập nhật service trong bảng services.
  - Nếu thành công trả về True, ngược lại False.

### **insertService**

- Mục đích: thêm vào một Service mới.
- Prototype: public boolean insertService(Service service)
- Input:  
  
service: đối tượng Service mới cần thêm.
- Output: giá trị boolean thành công hay không thành công.

- Gọi khi: khi Provider muốn yêu cầu thêm dịch vụ mới.
- Cách gọi: `.insertService(service)`
- Algorithm:
  - Thêm đối tượng service vào bảng services.
  - Nếu thành công trả về True, ngược lại False.

### **deleteService**

- Mục đích: xóa một Service với id tương ứng.
- Prototype: `public boolean deleteService(String id)`
- Input:

id: Id của Service cần xóa.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi Admin muốn xóa Service.
- Cách gọi: `.deleteService(id)`
- Algorithm:
  - Xóa Service trong bảng services có id tương ứng.
  - Nếu thành công trả về True, ngược lại False.

### **getRequest**

- Mục đích: truy xuất Request với id tương ứng.
- Prototype: `public boolean Request getRequest(String id)`
- Input:

id: Id của Request cần truy xuất.
- Output: Request.
- Gọi khi: khi Provider muốn lấy Request.
- Cách gọi: `.getRequest(id)`
- Algorithm:
  - Truy xuất trong mảng requests request có id là id.
  - Trả về Request.

### **getRequest**

- Mục đích: truy xuất các Request có providerId tương ứng.
- Prototype: `public List<Request> getRequest(String providerId)`
- Input:

providerId: Id của provider cần lấy Request.

- Output: Danh sách các Request của Provider.
- Gọi khi: khi Provider muốn xem các Request của mình.
- Cách gọi: .getRequest(providerId)
- Algorithm:
  - Truy xuất trong bảng requests các Request có trường provider là providerId.
  - Thêm vào List và trả về.

### **insertRequest**

- Mục đích: thêm một Request mới vào Database.
- Prototype: public boolean insertRequest(Request request)
- Input:

request: đối tượng Request mới cần thêm vào.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi User muốn gửi request cho Provider.
- Cách gọi: .insertRequest(request)
- Algorithm:
  - Thêm vào bảng requests đối tượng request .
  - Nếu thành công trả về True, ngược lại False.

### **getReply**

- Mục đích: lấy các Reply của commentId tương ứng.
- Prototype: public List<Reply> getReply(String commentId)
- Input:

commentId: id của Comment cần lấy Reply.
- Output: Danh sách các Reply của commentId.
- Gọi khi: khi người dùng muốn xem trả lời bình luận.
- Cách gọi: .getReply(commentId)
- Algorithm:
  - Truy xuất trong bảng replies các Reply tương ứng với commentId.
  - Thêm vào List và trả về.

## insertReply

- Mục đích: thêm một đối tượng Reply mới.
- Prototype: public boolean insertReply(Reply reply)
- Input:  
  
reply: đối tượng Reply mới cần thêm vào.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi người dùng muốn trả lời bình luận.
- Cách gọi: .insertReply(reply)
- Algorithm:
  - Thêm vào bảng replies đối tượng reply mới.
  - Nếu thành công trả về True, ngược lại False.

## insertRating

- Mục đích: thêm một đối tượng Rating mới.
- Prototype: public boolean insertRating(Rating rating)
- Input:  
  
rating: đối tượng Rating mới cần thêm vào.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi người dùng muốn đánh giá dịch vụ.
- Cách gọi: .insertRating(rating)
- Algorithm:
  - Thêm vào bảng ratings đối tượng rating mới.
  - Nếu thành công trả về True, ngược lại False.

## getAllLocations

- Mục đích: truy xuất danh sách toàn bộ Location.
- Prototype: public List<Location> getAllLocations()
- Input: Không.
- Output: danh sách các Locations.
- Gọi khi: người dùng xem bộ lọc địa điểm.
- Cách gọi: .getAllLocations().
- Algorithm:

- Truy xuất toàn bộ dữ liệu bảng locations.
- Trả về List kết quả.

### **getComment**

- Mục đích: truy xuất danh sách toàn bộ Location.
- Prototype: `public List<Comment> getComment(String serviceId)`
- Input:
  - serviceId: id của Service cần lấy comment.
- Output: danh sách các Comments.
- Gọi khi: người dùng xem bình luận.
- Cách gọi: `.getComment(serviceId)`
- Algorithm:
  - Truy xuất bảng comments các dữ liệu tương ứng serviceId.
  - Trả về List kết quả.

### **insertComment**

- Mục đích: thêm mới một đối tượng Comment.
- Prototype: `public boolean insertComment(Comment comment)`
- Input:
  - comment: đối tượng Comment cần thêm mới.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi người dùng bình luận.
- Cách gọi: `.insertComment(comment)`
- Algorithm:
  - Thêm đối tượng comment vào bảng comments.
  - Nếu thành công trả về True, ngược lại False.

### **getClick**

- Mục đích: truy xuất click của một advertise thông qua id.
- Prototype: `public List<Click> getClick(String adId)`

- Input:  
adId: Id của advertise cần lấy click.
- Output: Danh sách các click của advertise.
- Gọi khi: khi Provider muốn xem click của quảng cáo.
- Cách gọi: `.getClick(adId)`
- Algorithm:
  - Truy xuất trong quảng clicks các Clicks mà có adId bằng với adId.
  - Thêm vào List và trả về.

### **insertClick**

- Mục đích: thêm mới đối tượng Click vào bảng clicks.
- Prototype: `public boolean insertClick(Click click)`
- Input:  
click: đối tượng click cần thêm mới.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi người dùng nhấp vào một quảng cáo.
- Cách gọi: `.insertClick(click)`
- Algorithm:
  - Thêm đối tượng click vào bảng clicks.
  - Nếu thành công trả về True, ngược lại False.

### **getAllAdtype**

- Mục đích: truy xuất danh sách toàn bộ Adtype.
- Prototype: `public List<Adtype> getAllAdtype()`
- Input: Không.
- Output: danh sách các Adtype.
- Gọi khi: provider đăng ký quảng cáo.
- Cách gọi: `.getAllAdtype()`
- Algorithm:
  - Truy xuất toàn bộ dữ liệu bảng adtypes.





- Trả về List kết quả.

### **getUser**

- Mục đích: truy xuất User với id tương ứng.
- Prototype: `public User getUser(String id)`
- Input:  
  
id: Id của User cần truy xuất.
- Output: đối tượng User.
- Gọi khi: khi người dùng muốn sửa thông tin.
- Cách gọi: `.getUser(id)`
- Algorithm:
  - Truy xuất trong bảng users User có id tương ứng.
  - Trả về đối tượng User.

### **updateUser**

- Mục đích: cập nhật đối tượng user trong bảng users.
- Prototype: `public boolean updateUser(User user)`
- Input:  
  
user: đối tượng User cập nhật.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi người dùng cập nhật thông tin.
- Cách gọi: `.updateUser(user)`
- Algorithm:
  - Cập nhật đối tượng user trong bảng users.
  - Nếu thành công trả về True, ngược lại False.

### **insertUser**

- Mục đích: thêm đối tượng user mới vào bảng users.
- Prototype: `public boolean insertUser(User user)`
- Input:  
  
user: đối tượng cần thêm mới.



- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi User đăng ký tài khoản mới.
- Cách gọi: `.insertUser(user)`
- Algorithm:
  - Thêm mới đối tượng user vào bảng users.
  - Nếu thành công trả về True, ngược lại False.

### **deleteUser**

- Mục đích: xóa đối tượng user trong bảng users.
- Prototype: `public boolean deleteUser(String id)`
- Input:

id: id của đối tượng User cần xóa.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: Admin xóa User.
- Cách gọi: `.deleteUser(id)`
- Algorithm:
  - Xóa đối tượng User trong bảng users với id.
  - Nếu thành công trả về True, ngược lại False.

### **insertAdvertise**

- Mục đích: thêm một advertise vào bảng ads.
- Prototype: `public boolean insertAdvertise(Advertise advertise)`
- Input:

advertise: Advertise cần thêm .
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi Provider đăng ký quảng cáo.
- Cách gọi: `.insertAdvertise(advertise)`
- Algorithm:
  - Thêm đối tượng advertise vào bảng ads.
  - Nếu thành công trả về True, ngược lại False.

### **deleteAdvertise**

- Mục đích: xóa một advertise trong bảng ads.
- Prototype: public boolean deleteAdvertise(String id)
- Input:

id: id của Advertise cần xóa .

- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi Admin xóa quảng cáo.
- Cách gọi: .deleteAdvertise(id)
- Algorithm:
  - Xóa đối tượng Advertise với id trong bảng ads.
  - Nếu thành công trả về True, ngược lại False.

### **updateAdvertise**

- Mục đích: cập nhật một advertise trong bảng ads.
- Prototype: public boolean updateAdvertise(Advertise advertise)
- Input:

advertise: Advertise cần cập nhật .

- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi Admin chấp thuận/từ chối quảng cáo.
- Cách gọi: .updateAdvertise(advertise)
- Algorithm:
  - Cập nhật đối tượng Advertise trong bảng ads.
  - Nếu thành công trả về True, ngược lại False.

### **getBestService**

- Mục đích: lấy 3 dịch vụ tốt nhất trong bảng services.
- Prototype: public List<Service> getBestService()
- Input: Không
- Output: danh sách 3 Service tốt nhất.
- Gọi khi: khi người dùng xem các dịch vụ tốt nhất.
- Cách gọi: .getBestService()
- Algorithm:
  - Truy xuất 3 đối tượng có rating tốt nhất trong bảng services.
  - Thêm vào List và trả về.

## getRandomAdvertise

- Mục đích: lấy 4 Advertise một cách ngẫu nhiên trong bảng ads.
- Prototype: `public List<Advertise> getRandomAdvertise()`
- Input: không
- Output: 4 đối tượng Advertise ngẫu nhiên.
- Gọi khi: khi User xem quảng cáo.
- Cách gọi: `.getRandomAdvertise()`
- Algorithm:
  - Truy xuất 4 đối tượng Advertise ngẫu nhiên trong bảng ads.
  - Thêm vào List và trả về.

## getAdvertiseByProvider

- Mục đích: lấy các Advertise của một Provider.
- Prototype: `public List<Advertise> getAdvertiseByProvider(String provider)`
- Input:  
provider: id của provider cần lấy quảng cáo.
- Output: danh sách Advertise của Provider.
- Gọi khi: khi Provider muốn xem các quảng cáo của mình.
- Cách gọi: `.getAdvertiseByProvider(String provider)`
- Algorithm:
  - Truy xuất các đối tượng Advertise với provider trong bảng ads.
  - Thêm vào List và trả về kết quả.

## insertKindOfService

- Mục đích: thêm một ServiceType trong bảng servicetypes.
- Prototype: `public boolean insertKindOfService(ServiceType servicetype)`
- Input:  
servicetype: ServiceType cần thêm .
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi User/Provider muốn thêm mới loại dịch vụ.
- Cách gọi: `.insertKindOfService(servicetype)`
- Algorithm:

- Thêm đối tượng servicetype vào bảng servicetypes.
- Nếu thành công trả về True, ngược lại False.

### **deleteKindOfService**

- Mục đích: xóa một ServiceType trong bảng servicetypes.
- Prototype: public boolean deleteKindOfService(String id)
- Input:  
id: id của ServiceType cần xóa .
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi Admin xóa loại dịch vụ.
- Cách gọi: .deleteKindOfService(id)
- Algorithm:
  - Xóa đối tượng ServiceType với id trong bảng servicetypes.
  - Nếu thành công trả về True, ngược lại False.

### **updateKindOfService**

- Mục đích: cập nhật một ServiceType trong bảng servicetypes.
- Prototype: public boolean updateKindOfService(ServiceType servicetype)
- Input:  
servicetype: ServiceType cần cập nhật .
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi Admin cập nhật tình trạng của loại dịch vụ.
- Cách gọi: .updateKindOfService(ServiceType servicetype)
- Algorithm:
  - Cập nhật đối tượng ServiceType trong bảng servicetypes.
  - Nếu thành công trả về True, ngược lại False.

### **getAllKindOfService**

- Mục đích: truy xuất danh sách toàn bộ ServiceType.
- Prototype: public List<ServiceType> getAllKindOfService()
- Input: Không.

- Output: danh sách toàn bộ ServiceType.
- Gọi khi: khi hiển thị các bộ lọc loại dịch vụ.
- Cách gọi: `.getAllKindOfService()`
- Algorithm:
  - Truy xuất toàn bộ dữ liệu bảng servicetypes.
  - Trả về List kết quả.

### **getRating**

- Mục đích: truy xuất các Rating với id tương ứng.
- Prototype: `public List<Rating> getRating(String serviceId)`
- Input:

serviceId: id của Service cần lấy các Rating.
- Output: List đối tượng Rating.
- Gọi khi: khi tính toán rating trung bình.
- Cách gọi: `.getRating(serviceId)`
- Algorithm:
  - Truy xuất trong bảng ratings các Rating tương ứng serviceId.
  - Trả về List các kết quả.

## **6.2.5 Provider Class**

### **buyAdvertise**

- Mục đích: thực hiện mua một quảng cáo
- Prototype: `public boolean buyAdvertise(Advertise advertise)`
- Input:

Advertise advertise: thông tin về quảng cáo
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi nhà cung cấp đăng ký quảng cáo.
- Cách gọi: `.buyAdvertise(advertise)`

- Algorithm:
  - Tạo đối tượng advertise với status "pending".
  - Gọi hàm insertAdvertise(advertise).
  - Nếu thành công trả về True, ngược lại False.

### **getInterestingCustomer**

- Mục đích: lấy danh sách khách hàng quan tâm dịch vụ
- Prototype: public List<User> getInterestingCustomer()
- Input: Không có.
- Output: danh sách khách hàng quan tâm dịch vụ.
- Gọi khi: Nhà cung cấp muốn lấy thông tin người dùng quan tâm dịch vụ.
- Cách gọi: .getInterestingCustomer()
- Algorithm:
  - Dựa vào id của provider lấy danh sách các User trong bảng users.
  - Trả về List kết quả.

### **addService**

- Mục đích: yêu cầu thêm mới một service.
- Prototype: public boolean addService(Service service)
- Input:

Service service: dịch vụ cần thêm mới
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: nhà cung cấp yêu cầu thêm mới dịch vụ của mình vào hệ thống.
- Cách gọi: .addService(service)
- Algorithm:
  - Tạo một service với status là "pending".
  - Gọi hàm insertService(service).

- Nếu thành công trả về True, ngược lại False.

### **viewService**

- Mục đích: xem các dịch vụ của nhà cung cấp.
- Prototype: public List<Service> viewService()
- Input: Không có.
- Output: danh sách các dịch vụ của nhà cung cấp.
- Gọi khi: nhà cung cấp xem các dịch vụ của mình.
- Cách gọi: .viewService()
- Algorithm:
  - Lấy id của đối tượng nhà cung cấp.
  - Gọi hàm .getService(providerid).
  - Trả về List kết quả.

### **sendRequestAdKindService**

- Mục đích: thêm một servicetype vào bảng ServiceType từ userpage hoặc providerpage.
- Prototype: public boolean sendRequestAdKindService(String kind)
- Input:  
kind: loại dịch vụ muốn đề xuất.
- Output: giá trị boolean thành công hay không thành công.
- Gọi khi: khi Provider muốn đề xuất loại dịch vụ.
- Cách gọi: .sendRequestAdKindService(kind)
- Algorithm:
  - Tạo đối tượng servicetype với kind.
  - Gọi hàm .insertKindOfService(servicetype).
  - Nếu thành công trả về True, ngược lại False.

### **viewAdvertise**

- Mục đích: xem các quảng cáo của nhà cung cấp.





- Prototype: public List<Advertise> viewAdvertise()
- Input: Không có.
- Output: danh sách các quảng cáo của nhà cung cấp.
- Gọi khi: nhà cung cấp xem các quảng cáo của mình.
- Cách gọi: .viewAdvertise()
- Algorithm:
  - Lấy id (provider) của đối tượng nhà cung cấp.
  - Gọi hàm .getAdvertiseByProvider(provider).
  - Trả về List kết quả.

## 7. Github Repository

<https://github.com/ititandev/service-recommender-system>