**HỒ CHÍ MINH CITY UNIVERSITY OF TECHNOLOGY AND EDUCATION**

**FACULTY FOR HIGHT QUALITY TRAINING**



**GRADUATION THESIS**

**BUILD A WEBSITE PUBLISHED**

**AND FIND MOTEL**

**Student:**

**CHU MINH HOÀNG 17110139**

**PHẠM HUỲNH THANH LÂM 17110168**

**Course: 2017 – 2021**

**Major: INFORMATION TECHNOLOGY**

**Instructor: MSc TRẦN CÔNG TÚ**

Ho Chi Minh City, \_\_ July 20\_\_

|  |  |
| --- | --- |
|  | SOCIALIST REPUBLIC OF VIETNAM  **Independence – Freedom – Happiness**  ----\*\*\*----  *Ho Chi Minh City, \_\_\_ July 20\_\_* |

**MISSION ABOUT GRADUATION PROJECT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Full name: Chu Minh Hoàng | | | Student ID: 17110139 | |
| Full name: Phạm Huỳnh Thanh Lâm | | | Student ID: 17110168 | |
| Major: Information technology \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | | Class: 17110CLST3 - ST1 | |
| Instructor: MSc TRẦN CÔNG TÚ | | |  | |
| Date received the topic: | | | Thesis submission date: | |
| 1. Name of topic: Build a website published and find motel. | | | | |
| 2. Original data and documents:   * Algorithm implementation using HTML, CSS, JS, Bootstrap, ASP.NET CORE 3.x, SQL Server, Angular, Flutter, Flask Python, Signal-R. * Design a publishing website and find motel. * Refer to w3school's documentation. | | | | |
| 3. Content of the project:   * Research and design a website to publish information about motel with full features such as: publishing, managing news, finding motel, paying after publishing, managing services, managing employees, ... * Study the design of a basic Android application to view, find motels and contact published directly. | | | | |
| 4. Product:  Design and build a website and mobile android app to advertise and search for motel with full basic functions. | | | | |
| **INTRUCTOR LEADER** | | | **INTRUCTOR** | |
|  | | |  | |
|  | SOCIALIST REPUBLIC OF VIETNAM  **Independence – Freedom – Happiness**  ----\*\*\*----  *Ho Chi Minh City, \_\_\_ July 20\_\_* | |

**TEACHER'S COMMENTS QUESTIONS**

Full name: Chu Minh Hoàng Student ID: 17110139

Full name: Phạm Huỳnh Thanh Lâm Student ID: 17110168

Major: Information technology

Name of topic: Build a website published and find motel.

Name of instructor: MSc TRẦN CÔNG TÚ

**COMMENT**

1. Regarding the topic content & implementation volume:

.................................................................................................................................

.................................................................................................................................

.................................................................................................................................

1. Advantage:

.................................................................................................................................

.................................................................................................................................

.................................................................................................................................

1. Disadvantage:

.................................................................................................................................

.................................................................................................................................

.................................................................................................................................

1. Recommend for protection or not?

.................................................................................................................................

1. Type rating:

.................................................................................................................................

1. Score: .................. (Text....................................................................................)

*Ho Chi Minh City \_\_\_ July 20\_\_*

**INSTRUCTOR**

*(Sign and write full name)*

|  |  |
| --- | --- |
|  | SOCIALIST REPUBLIC OF VIETNAM  **Independence – Freedom – Happiness**  ----\*\*\*----  *Ho Chi Minh City, \_\_\_ July 20\_\_* |

**REVIEWER TEACHER COMMENTS QUESTION**

Full name: Chu Minh Hoàng Student ID: 17110139

Full name: Phạm Huỳnh Thanh Lâm Student ID: 17110168

Major: Information technology.

Name of topic: build a website published and find motel.

Name of reviewer: ..................................................................................................

**COMMENT**

1. Regarding the topic content & implementation volume:

.................................................................................................................................

.................................................................................................................................

.................................................................................................................................

1. Advantage:

.................................................................................................................................

.................................................................................................................................

.................................................................................................................................

1. Disadvantage:

.................................................................................................................................

.................................................................................................................................

.................................................................................................................................

1. Recommend for protection or not?

.................................................................................................................................

1. Type rating:

.................................................................................................................................

1. Score: ............. (Text ........................................................................................)

*Ho Chi Minh City \_\_\_ July 20\_\_*

**REVIEWER**

*(Sign and write full name)*

# **ACKNOWLEDGEMENT**

First, the implementation team would like to thank the Faculty of High-Quality Training - Ho Chi Minh City University of Technology and Education for creating all the most favorable conditions for the implementation team to have the opportunity to have free access to, and participate in.

The most respectful thanks that the team implements would like to go to **Mr. Tran Cong Tu** - who used all his enthusiasm and knowledge, accompanied, and directly guided and created all conditions to help the team develop. develop their full potential as well as cultivate knowledge throughout the study process, especially during the preparation and implementation from specialized essay to graduate thesis.

Finally, the implementation team wishes you good health and faith to continue your noble career of passing on knowledge to future generations.

Group sincerely thank you!

# **INTRODUCTION**

With the current world situation, the covid pandemic is complicated. Many industries are in trouble due to the pandemic, but in terms of technology, it is not trending down but growing very strongly. Technology makes it as easy as ordering through websites or ordering application, can text and call through social networks, etc.

In addition, the demand for motel of students and office workers is increasing, especially for students who have just entered the university, it is very necessary to find and rent houses. As a student group, I want to create a website that can be easily searched and published motel for users. Therefore, the group chose the topic "Building a website to published and search motel".

# **CONTENT**

[**ACKNOWLEDGEMENT** i](#_Toc77616188)

[**INTRODUCTION** ii](#_Toc77616189)

[**CONTENT OF ACRONYMS** iii](#_Toc77616191)

[**CONTENT TABLES** iv](#_Toc77616192)

[**CONTENT IMAGE AND DIAGRAM** v](#_Toc77616193)

[**CHAPTER 1 OVERVIEW** 1](#_Toc77616194)

[**1.1** **The reason for choosing the topic.** 1](#_Toc77616195)

[**1.2** **Target** 1](#_Toc77616196)

[**1.3** **Technology** 1](#_Toc77616197)

[**CHAPTER 2 THEORETICAL BASIS** 3](#_Toc77616198)

[**2.1** **Angular framework** 3](#_Toc77616199)

[**2.2** **ASP.NET API** 4](#_Toc77616200)

[**2.3** **ASP.NET Entity Framework** 4](#_Toc77616201)

[**2.4** **Microsoft SQL 2017** 4](#_Toc77616202)

[**2.5** **Firebase** 5](#_Toc77616203)

[**2.6** **Online chat software tawk.to** 5](#_Toc77616204)

[**2.6.1** **Introduce** 5](#_Toc77616205)

[**2.6.2** **Purpose** 5](#_Toc77616206)

[**2.7** **Flutter** 5](#_Toc77616207)

[**2.7.1** **Define** 5](#_Toc77616208)

[**2.7.2** **Why use flutter** 6](#_Toc77616209)

[**2.7.3** **Flutter's Structure** 6](#_Toc77616210)

[**2.7.4** **What is Flutter's feature*?*** 6](#_Toc77616211)

[**2.8** **Flash-Python** 7](#_Toc77616212)

[**2.8.1** **What is Flash python?** 7](#_Toc77616213)

[**2.8.2** **Ideas to use** 7](#_Toc77616214)

[**2.9** **Leaflet** 8](#_Toc77616215)

[**2.10** **Viet-Map API** 8](#_Toc77616216)

[**2.11** **Signal-R** 8](#_Toc77616217)

[**CHAPTER 3 ANALYSIS AND DESIGN** 10](#_Toc77616218)

[**3.1** **Surveying website** 10](#_Toc77616219)

[**3.1.1** **Reference, and survey website ‘phongtro123’** 10](#_Toc77616220)

[**3.1.2** **Reference, and survey website: ‘Cho tot’** 10](#_Toc77616221)

[**3.1.3** **Reference, and survey website: ‘Mogi’** 11](#_Toc77616222)

[**3.1.4** **Reference, and survey website: ‘Homedy’** 11](#_Toc77616223)

[**3.1.5** **Conclude** 12](#_Toc77616224)

[**3.2** **Modeling required** 16](#_Toc77616225)

[**3.2.1** **Design Use-case diagram** 16](#_Toc77616226)

[**3.2.2** **Use-case specification** 18](#_Toc77616227)

[**3.3** **Sequence diagram** 40](#_Toc77616228)

[**3.4** **Database design** 46](#_Toc77616229)

[**3.4.1** **Entity relationship diagram** 46](#_Toc77616230)

[**3.4.2** **A relational model of tables in a database** 47](#_Toc77616231)

[**3.5** **Design the theme.** 49](#_Toc77616232)

[**3.5.1** **Map search screen** 49](#_Toc77616233)

[**3.5.2** **Forum screen** 49](#_Toc77616234)

[**3.5.3** **Recommend motel screen** 50](#_Toc77616235)

[**3.5.4** **Home page app screen** 50](#_Toc77616236)

[**CHAPTER 4 PRODUCT INSTALLATION** 51](#_Toc77616237)

[**4.1** **Font end** 51](#_Toc77616238)

[**4.4.1** **Application structure website** 51](#_Toc77616239)

[**4.4.2** **The main word processing in font-end** 52](#_Toc77616240)

[**4.2** **Back end** 53](#_Toc77616241)

[**4.2.1** **Project API structure** 53](#_Toc77616242)

[**4.2.2** **Several Functions, and Procedures** 54](#_Toc77616243)

[**4.3** **Technology** 58](#_Toc77616244)

[**4.3.1** **Flash-Python** 58](#_Toc77616245)

[**4.3.2** **Leaflet** 60](#_Toc77616246)

[**4.3.3** **Signal-R** 60](#_Toc77616247)

[**4.3.4** **Viet-Map** 61](#_Toc77616248)

[**4.4** **AppFlutter** 63](#_Toc77616249)

[**CHAPTER 5 CONCLUSION** 66](#_Toc77616250)

[**5.1** **Result** 66](#_Toc77616251)

[**5.2** **Advantages** 68](#_Toc77616252)

[**5.3** **Defect** 69](#_Toc77616253)

[**5.4** **Disadvantage** 69](#_Toc77616254)

[**5.5** **Lessons Learned** 69](#_Toc77616255)

[**5.6** **Future work** 70](#_Toc77616256)

[**REFERENCES** 71](#_Toc77616257)

[**APPENDIX** 72](#_Toc77616258)

[**Appendix I: Development environment settings** 72](#_Toc77616259)

[**Appendix II: Creating an Angular application** 72](#_Toc77616260)

[**Appendix III: Creating an App Flutter** 73](#_Toc77616261)

# **CONTENT OF ACRONYMS**

CSDL: Database

ORM: Object Relational Mapping

ĐT: Good coin

HCM: Ho Chi Minh

BDS: Real estate

# **CONTENT TABLES**

[**Table 3.1:** Use-case specification login phone number 18](#_Toc77200833)

[**Table 3.2:** Use-case specification login google account. 19](#_Toc77200834)

[**Table 3.3:** Use-case specification is registered by phone number 21](#_Toc77200835)

[**Table 3.4:** Use-case specification registered with Google account. 22](#_Toc77200836)

[**Table 3.5:** Use-case specification to search motel by type, city, price, district 24](#_Toc77200837)

[**Table 3.6:** Use-case specification publish motel 25](#_Toc77200838)

[**Table 3.7:** Use-case specification see the detailed motel 27](#_Toc77200839)

[**Table 3.8:** Use-case specification changes the password 28](#_Toc77200840)

[**Table 3.9:** Use-case specification for chat 30](#_Toc77200841)

[**Table 3.10:** Use-case specification for logged out 31](#_Toc77200842)

[**Table 3.11:** Use-case specification for add employee 33](#_Toc77200843)

[**Table 3.12:** Use-case specification for edit admin information 34](#_Toc77200844)

[**Table 3.13:** Use-case specification extension motel news 36](#_Toc77200845)

[**Table 3.14:** Use-case specification search map 37](#_Toc77200846)

[**Table 3.15:** Use-case specification log account 39](#_Toc77200847)

[**Table 4.1:** Table jaw Flask-Python 58](#_Toc77200848)

[**Table 4.2:** Viet-map autocomplete parameter 62](#_Toc77200849)

# **CONTENT IMAGE AND DIAGRAM**

[**Figure 3.1:** Use case diagram 16](#_Toc77200900)

[**Figure 3.2:** Login sequence diagram 40](#_Toc77200901)

[**Figure 3.3:** Sequence diagram of browsing management 41](#_Toc77200902)

[**Figure 3.4:** Sequence diagram register diagram 42](#_Toc77200903)

[**Figure 3.5:** Search sequence diagram 43](#_Toc77200904)

[**Figure 3.6:** Publishing sequence diagram 44](#_Toc77200905)

[**Figure 3.7:** Sequence diagram of phone number change 45](#_Toc77200906)

[**Figure 3.8:** Sequence diagram of bill management 45](#_Toc77200907)

[**Figure 3.9:** Entity association model 46](#_Toc77200908)

[**Figure 3.10:** Relational model of tables in a database 48](#_Toc77200909)

[**Figure 3.11:** Map search screen 49](#_Toc77200910)

[**Figure 3.12:** Forum screen 49](#_Toc77200911)

[**Figure 3.13:** Recommend motel screen 50](#_Toc77200912)

[**Figure 3.14:** App home and detail screen 50](#_Toc77200913)

[**Figure 4.1:** Script to embed third-party widget tawk. to 52](#_Toc77200914)

[**Figure 4.2:** The function loads the image on firebase and uploads the URL link 53](#_Toc77200915)

[**Figure 4.3:** Function to send code to the phone 53](#_Toc77200916)

[**Figure 4.4:** API get motel for search by city, province, district, street, type and price 55](#_Toc77200917)

[**Figure 4.5:** API put account with hash password 56](#_Toc77200918)

[**Figure 4.6:** API get motel for app 57](#_Toc77200919)

[**Figure 4.7:** API get motel for map screen 57](#_Toc77200920)

[**Figure 4.8:** Import Leaflet API to Angular 60](#_Toc77200921)

[**Figure 4.9:** Package version Signal-R 60](#_Toc77200922)

[**Figure 4.10:** Config Signal-R 61](#_Toc77200923)

[**Figure 4.11:** Config link connect Signal-R 61](#_Toc77200924)

[**Figure 4.12:** Version Signal-R in angular 61](#_Toc77200925)

[**Figure 4.13:** Constructor Viet-map API 62](#_Toc77200926)

[**Figure 4.14:** Viet-map autocomplete API 62](#_Toc77200927)

[**Figure 4.15:** Script to embed third-party widget tawk.to 63](#_Toc77200928)

# **CHAPTER 1 OVERVIEW**

## **The reason for choosing the topic.**

With the increasing demand of rental and rental motel, especially students, or office workers and motel owners want to trust to rent out their rooms. Understanding the necessity of the web "build a website published and find motel." was formed. In addition, users can search and view information about motel through a simple app.

## **Target**

The goal of the product is to create a website and app. Divided into two main users: tenants and published. Tenants can go to the website or app to perform a search, see details, contact methods to publish the message, and can also search at the locations that the tenant wants to make it convenient. for tenants. Advertisers will be able to publish about their motel and manage their publishes.

## **Technology**

To build a suitable application for the theoretical problem, the topic will explore, and research the following areas:

* + Website:
* Learn about Angular, the JavaScript programming language, and website application libraries.
* Studying on SQL databases, Firebase for data storage.
* Learn about bootstrap, HTML, and CSS for interface design.
* Learn ASP.Net to write Back-end.
* Apply Technology: Leaflet, Flash-Python, Flutter, Viet-Map-API, Signal-R.
* Find out how to pay via Papal (Visa card).
  + Mobile app:
* Learn about Flutter, the Dart programming language, and app package to use.

# **CHAPTER 2 THEORETICAL BASIS**

## **Angular framework**

Angular is considered a free open source or framework dedicated to web design. Angular has been in development since 2009 and is maintained by Google. These frameworks are considered the most powerful front-end frameworks dedicated by advanced HTML programmers. [2]

Angular is widely used to build Single Page Applications (SPA) projects. Currently, Angular version stable is Angular 9 (released on February 7, 2020) with TypeScript 3.6, and 3.7.

* Highlights:
* Angular is the solution for easy single-page applications.
* Front-end code is usually very friendly thanks to the ability to Binding data on HTML platforms to create all sorts of great operations.
* You can easily Unit test.
* Components can be reused more easily.
* Support for programmers who can write less code with more functionality.
* You can run Angular in a variety of browsers, either on a PC or mobile.
* The basic features:
* Used for JavaScript-based development.
* Angular can create client-side applications based on the MVC model.
* Angular possesses high compatibility and can automatically handle JavaScript code so that it is suitable for most browsers.
* When free, and open-source code is available, it will be used more widely.
* Usually, the architecture of an Angular application is based on the ideas related to Components. Each Angular application usually starts with the same top levels called Root Components.

## **ASP.NET API**

API term stands for Application Programming Interface - Application Programming Interface. ASP.NET Web API is a framework, provided by Microsoft, that makes it easy to build Web API, i.e. services based on HTTP protocol. The ASP.NET Web API is the ideal platform for building Restful services on top of the .NET Framework. These Web API services can be used by many different clients, such as: browser, mobile application, desktop application, IOT, etc. [1]

## **ASP.NET Entity Framework**

Entity Framework is an ORM. Entity Framework

ORM is a set of technologies that allow working with Relational Database Management System (RDBMS) from object-oriented languages ​​and without needing it directly.

ORMs like Entity Framework support programmer to map (two dimensions): class with table structure; object with records (or rows) in the table; property with the column of the table; sets objects with sets of records; references to objects with relationships between tables. During this process, all SQL queries are generated, and executed automatically by the ORM. The programmer only needs to work with familiar concepts of the programming language. [4]

## **Microsoft SQL 2017**

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product whose primary function is to store, and retrieve data required by other software applications. Can run on the same computer or on another computer on the network (including the Internet).

Microsoft markets at least a dozen different versions of Microsoft SQL Server, aimed at different audiences, and for different workloads, from small single-machine applications to large Internet applications with many concurrent users.[10]

## **Firebase**

Firebase is a real-time database service provided by Google and operated in the cloud. It helps programmers to rapidly develop mobile applications by simplifying database operations, and storing, and synchronizing real-time user data. Applications that support this feature can store and retrieve data from the server very quickly. The data is stored in NoSQL database system and is placed on cloud server platform. Data is written and read with the lowest time in milliseconds. This platform supports the synchronization of user data even when there is no network connection. Creates a seamless experience regardless of the user's internet connection. [7]

## **Online chat software tawk.to**

### **Introduce**

Tawk.to is a free online chat software to help customers interact with website owners. Helping to answer customer questions as quickly as possible is to help businesses increase sales, and revenue quickly. [8]

### **Purpose**

Integrating tawk.to helps to support communication between customers, and website, helps to answer customers' questions, and at the same time helps employees proactively approach, and communicate with customers.

## **Flutter**

### **Define**

Flutter is a cross-platform application development platform for iOS and Android developed by Google that is used to create native apps for google.

Flutter consists of 2 important components:

* An SDK (Software Development Kit): A collection of tools that will help you develop your applications. This includes tools to compile your code into native machine code (code for iOS and Android).
* A Framework (UI Library based on widgets): A set of reusable user interface (UI) components (buttons, text inputs, sliders, etc.) only me.

### **Why use flutter**

Rapid app development: Hot reload makes it quick and easy to test, build user interfaces, add features, and fix bugs faster. Experience a second reload, without losing state, on emulators, simulators and devices for iOS and Android.

Beautiful and attractive user interface: Delight your users with Flutter's beautiful built-in widgets in Material Design and Cupertino (iOS-flavor), rich motion application programming interfaces (APIs), scrolling is naturally smooth and platform-aware.

Modern Framework: Easily create your user interfaces with Flutter's modern framework and rich set of platforms, layouts, and widgets. Solve your tough user interface challenges with powerful and flexible APIs for 2D, animation, gestures, effects and more.

### **Flutter's Structure**

Flutter has created a "new" concept that is widgets. Widgets are the foundation of Flutter, a widget that describes a part of the user interface. Every component like text, image, button or animation, theme, layout or even app is a widget. In Flutter all widgets or interfaces are coded in dart

When a widget changes state, such as due to a user click or an animation, the widget rebuilds itself to the new state. This saves developer time because UI can be described as a state function. We do not have to write extra code to only update the UI when the state changes.

### **What is Flutter's feature*?***

* Flutter is now and is a react framework.
* Using Dart programming language is simple and easy to learn.
* Fast application development.
* Nice and flexible user interface.
* Supports a lot of different widgets.
* Show the same UI on multiple platforms.
* High performance application.

## **Flash-Python**

### **What is Flash python?**

Flask is a web framework; it belongs to the type of micro-framework built in Python programming language. Flask allows you to build simple to complex web applications. It can build small APIs, web applications such as websites, blogs, wiki pages or a time-based website or even a commercial website. Flask provides you with the tools, libraries, and technologies to help you do the above.

Flask is a micro-framework. This means that Flask is a standalone environment with little use of other external libraries. Therefore, Flask has the advantage of being lightweight, has very few errors due to less dependencies, and is easy to detect and handle security errors.

### **Ideas to use**

Based on the title, it will predict related motel to suggest to users.

Steps to take:

* + - First the data processing:
      * Take the description and the title of the message to form a table.
      * Then delete unnecessary words or words that do not matter to the results.
    - Next, we deal with the correlation between the data.
      * Calculate the correlation: convert the description and headlines that form a table into a numeric matrix.
      * Calculate the correlation of the number matrix.
    - Next compare the correlation and make the appropriate suggestion.

## **Leaflet**

Leaflet is an open-source JavaScript library for building an interactive map application. This is a lightweight library, only about 38KB for the script but has all the features that most developers need.

Leaflet is designed with simplicity, performance, and usability in mind. It also works well on both desktop and mobile platforms, can be extended with dozens of plugins, plus it has a nice API document page, simple but easy to read.

## **Viet-Map API**

Viet-Map API is providing static maps, Maps API interacts with Viet-map's huge data warehouse. Developers can create great experiences for users on many different application platforms.

Viet-map Application Co., Ltd (Viet-map Co., Ltd) was established in 2006 by highly qualified experts, many years of experience in the field of GIS (Geographic Information System - geographic information system). and information technology.

## **Signal-R**

ASP.NET Signal-R is a library for ASP.NET developers to add real-time web functionality to their applications. Real-time web functionality is the ability to have server-side code push content to the connected clients as it happens, in real-time.

Signal-R takes advantage of several transports, automatically selecting the best available transport given the client's and server's capabilities. Signal-R takes advantage of WebSocket, and HTML5 API that enables bi-directional communication between the browser and server. Signal-R will use WebSocket under the covers when it is available, and gracefully fall back to other techniques and technologies when it is not, while the application code remains the same.

Signal-R also provides a simple, high-level API for doing server-to-client RPC (call JavaScript functions in a client's browser from server-side .NET code) in an ASP.NET application, as well as adding useful hooks for connection management, such as connect/disconnect events, grouping

connections, authorization.[6]

# **CHAPTER 3 ANALYSIS AND DESIGN**

## **Surveying website**

### **Reference, and survey website ‘phongtro123’**

#### Advantages

* Support customers to publish news, have a service table showing how and where to publish information on the website.
* There are ways to display eye-catching users such as hot, VIP 1, VIP 2, VIP 3 news and regular news.
* Payment is made through various methods: MoMo and international wallets such as visa, MasterCard, etc.

#### Defect

* User cannot enter a desired address.
* Publish that have expired are still displayed on the website.
* Publish are always charged, not free, limited depending on membership level.
* Don't have notice when there is new news, activities.

### **Reference, and survey website: ‘Cho tot’**

#### Advantages

* Notice when there is new news, activities.
* The saved message feature helps users review the news that needs to be reviewed.
* The website friends feature supports making friends and making chat with each other as a messenger platform.
* You can create your certificate page using the logged-in account.
* Create an on-page ad campaign.

#### Defect

* Unable to contact the page admin to respond directly if there are questions.
* Showing too little information on the published motel news.

### **Reference, and survey website: ‘Mogi’**

#### Advantages

* When entering the address, the location will appear on a google map help see any location near there.
* The map function shows the motels or realms that are zoned and marked.
* Support Language for English and Vietnamese

#### Defect

* The number of motel stars cannot be taken from finder.
* New motel published are always charged without free publishing, the limit time to published motel depends on membership level.

### **Reference, and survey website: ‘Homedy’**

#### Advantages

* Real estate for those who need to invest to consider a house or place to rent.
* Support to provide information about places in two cities Ho Chi Minh and Ha Noi.
* See feng shui for motel or house.
* See feng shui.
* Community: help publish questions, allow communication between people on the website.

#### Defect

* Information authentication creates trustworthiness.
* There is no interaction between the tenant, and the admin page to respond if there is a problem.
* When publishing a message, undergo many steps of operation many times (can perform 1 step aggregating information to be published).
* There is no multilingual integration, just use for Vietnamese people.

### **Conclude**

#### System description

##### *User*

* Website management
* Administrator: manage user information, create accounts, provide cost information for publishing information, display system data, manage user publishes.
* Management staff.
* Marketing advertising: provide the latest information.
* A user searching for news: Are students, people wishing to rent motel, or businesses wishing to buy, and sell. Help look up information about the motel quickly when going to an unfamiliar place.
* Publish information:
* Personal: The published can publish their news to advertising, based on the cost they pay for the publish, can cancel, or upgrade it. Update your information.
* Organizations: can register to publish information about motels, and rental locations they own.

##### *Function*

###### Register / Login

Users can log in with his / her permission to published news on the system.

Login / Registration can be by Facebook or google.

When registering, will perform confirmation by phone number.

Forgot your password when sending your password via voice message.

###### Search

Search can be by:

* Search bar.
* Search filters: City, county, price, area, geographic location.
* Filter by price, nearest location, latest news, latest news.
* Private or collective news published.

###### List of publishes

* List of news items searched: Picture, the extent of news, name of motels, location, area, date of publishing, price, and some basic information.
* Pagination (possible): based on the number of news that shows through different pages.
* Hot news, news has just been published.

###### Private details

* Information, and data about the motel.
* List of photos of the motel.
* Detailed location of the motel.
* Contact information (phone number, Zalo, Facebook, ...)
* Google map indicates the location of the motel news.
* There is also a list of news regarding the motel's location.

###### Publish news

* Publish information on the system through the steps of entry.
* Select the geographical location of the motel on google map.
* Payment by various methods for news.
* List of levels of the news published.
* Manage to publish (cancel news, upgrade news, ...)

###### Support

* Through chat-box reply message.
* Basic guide to publishing.

###### History

* Transaction publishing of users.
* Fee-charged transactions.

###### Recharge

Users recharge their accounts via the form of payments.

Various forms of payment with domestic cards and VISA, Mastercard, etc.

###### Verify users by phone.

Confirm registration, confirm phone number change, or forget password via SMS.

##### *Some other information*

* Manual for the website for both publishing information, logging in / registering, recharging, managing information.
* Service table, and information about the different service levels of publishing.
* Management of personal information.

##### *Other function*

* Friends: The main purpose is to chat with friends connected through the messenger platform, in the ‘Cho tot’ app we can interact directly with friends on messenger and share the good news or hot news. Only registered account holders can use the friend's feature.
* Chat: This feature allows tenants to interact directly with the person who published, and vice versa, the published can interact with the tenant. The chat feature appears when users view a room rental message in addition to how to make a call, they can text to the published if the phone is busy. A tenant can send a text message to agree on a meeting time to check the motel, negotiate the price, and the rules of the motel.
* Comment on asking questions on the forum page: exchange information, questions, and news.
* Google map: regional search, zoning search, and displaying basic information on google map.
* Phone number verification

#### Major

Customer registration will enter the full name, email address, phone number, password, and confirm password. The system will check if it is valid, it will send a 6-digit code to the phone number of the customer who entered, and the customer will confirm the code. The system will again check if it is valid, it will return to the home page, or the false message will not be reported.

Searching for guests or users can choose to use the search function, search by the keyword you enter, by country, by district, or by type of motel. The system will check and return related motel.

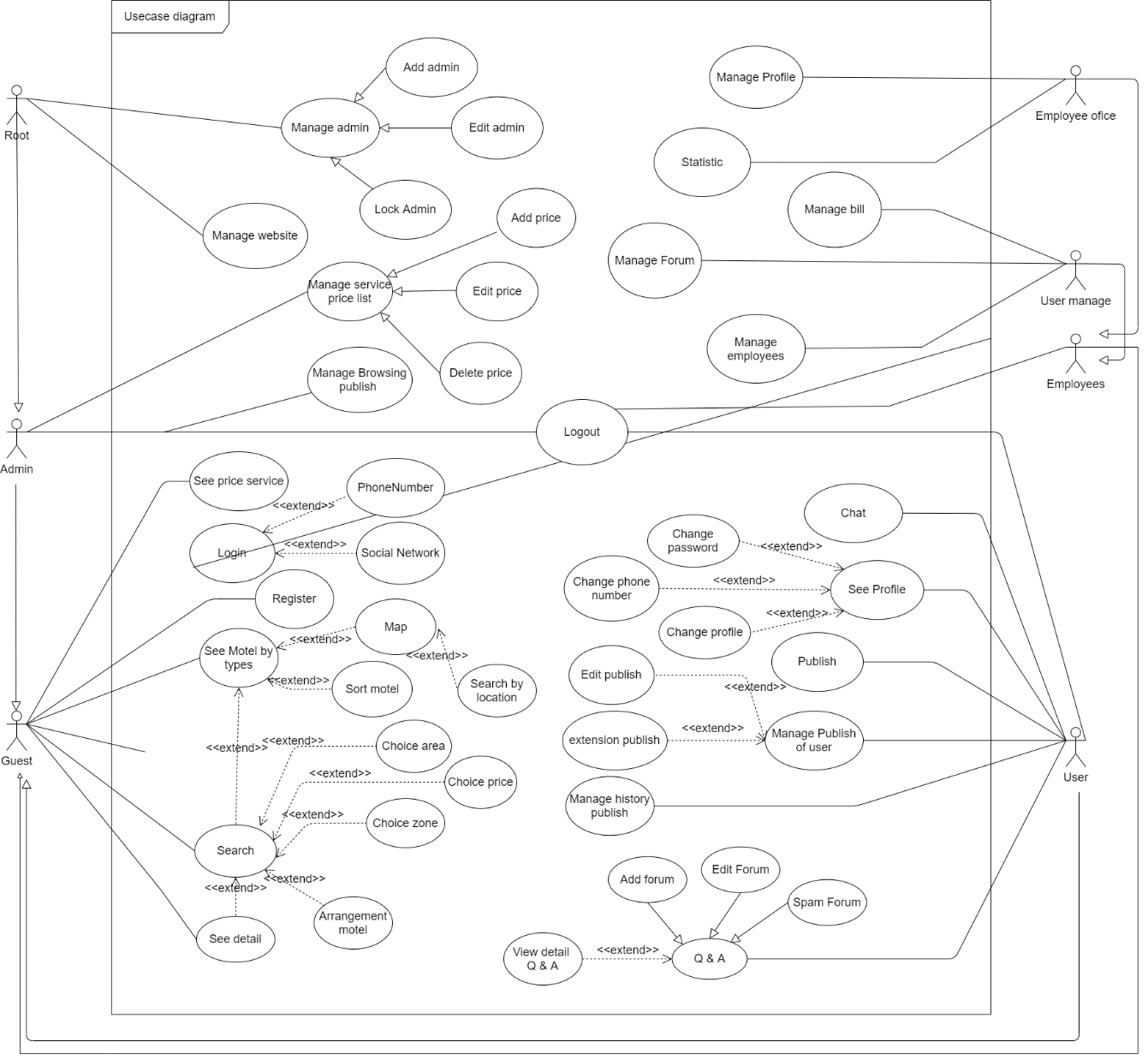
Customer login service will log in with Google, Facebook, or enter a phone number. The system will check if the guest is the user who will move to the home page, the admin or the employee will move to the website management page.

Service of publishing information, the user will select the publishing function, and choose the category of the house, then enter the property information (house name, city name, district address, address, and phone number), and then enter the information. Information to publish (rent, house area, direction, legal, news title, property description), then enter the details (a form of the house (front house, original house, alley, ...)., number of rooms, number of bedrooms), next will add pictures from the machine (can choose many pictures), next will choose the price to publish by day, month or week, final payment.

In the browsing part, after the user the news, the news will be added to the queue and the staff will approve the news for the published. If the published provides the wrong content, the staff will respond. If the publish is valid, the staff will approve the publish for the published and the status will be updated and the message will be published.

## **Modeling required**

### **Design Use-case diagram**



**Figure 3.1:** Use case diagram

A request to build a website where information can be published, and the motel's information searched.

There are 4 The actors involved in the problem: message viewer, user-user (who has an account), employee, manager:

* Private viewers who do not have an account can visit the website to view news, register an account, and comment on forums.
* User users who are registered users can view published news, buy gold coins, log out, publish news, and comment on forums.
* Staff is the user manager, and publish, revenue statistics for management.
* The manager is the last reviewer to publish on the site.
* When the user publishes the message according to the steps and confirms with the success message, the published message will be in line and wait for the approval staff, the staff will go to the browsing site to check the information and browse for it. Manager, the manager is the last person to approve, agree, the message will be published on the website. If the publish is not valid, it will notify the user to correct it.
* All will be able to view and comment on the talk page of the website.
* Login users can log in with an account and can log in using social networks like Google.
* Viewers, when creating a new account, can also easily register through 2 ways: using the personal phone number, after entering the phone number, the message will appear as a 6-digit code and enter 6-digits. In the confirmation section, a message will appear. When creating an account with a social network, the viewer will enter the account, and the password, the system will check, and create an account.
* Search on the map: The user enters the place to find, and the map will show the location that the user wants to find with related motel within the allowed radius.

### **Use-case specification**

**Table 3.1:** Use-case specification login phone number

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-1 | | | |
| Use Case No. | DT-1 | Use Case Version | 1.0 |
| Use Case Name | Sign in with phone number | | |
| Author | Chu Minh Hoàng | | |
| Date | 6/2021 | Priority | High |
| The actor:   * Guest   Summary:   * Use this case allows visitors to log into the website to use the advanced services of the site.   Goal:   * The account's role will be confirmed.   Triggers:   * Click the "Login" button.   Preconditions:   * User account has been created. * User account has been authorized. * The actor's device was connected to the internet when logging in   Publish conditions:   * Success: The account role is changed, and login to the user page * Fail: Display the message "Your password is not correct".   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | The actor selects the login button | The system will redirect users to the login page | | 2 | The actor enters phone number and password | The data shows up on the information that guests have entered | | 3 | The actor will select the login button | The system notifies the actor "Login successfully."  The system moves to "Home" screen |   Alternative Scenario:  Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | 1 | The actor re-enter login information | The system returns to login screen |   **Relationships:**  Business Rules:   * Employees must log in as the user role. | | | |

**Table 3.2:** Use-case specification login google account.

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-2 | | | |
| Use Case No. | DT-2 | Use Case Version | 1.0 |
| Use Case Name | Sign in with the Google account | | |
| Author | Chu Minh Hoàng | | |
| Date | 6/2021 | Priority | High |
| The actor:   * Guest   Summary:   * Use this case allows visitors to log into the website to use the advanced services of the site.   Goal:   * The account's role will be confirmed.   Triggers:   * Click on the button "Sign in with Google”.   Preconditions:   * User account has been created. * User account has been authorized. * The user's device was connected to the internet when logging in   Publish conditions:   * Success: The account role is changed, and login to the user page * Fail: Display the message "Your password is not correct".   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | The actor selects the login button | The system will redirect user to login page | | 2 | The actor chooses to sign in with Google | The system will redirect to the Google page | | 4 | The actor enter account google | The system will display account and check account | | 3 | The actor will select the "Continue" button. | The system notifies the user "Login successfully", and redirects the user to the home page |   Alternative Scenario:  Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | 1 | The actor re-enter login information | The system returns to login screen |   **Relationships:**  Business Rules:   * Employees must log in as the user role. | | | |

**Table 3.3:** Use-case specification is registered by phone number

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-3 | | | |
| Use Case No. | DT-3 | Use Case Version | 1.0 |
| Use Case Name | Sign up for an account by phone number | | |
| Author | Chu Minh Hoàng | | |
| Date | 6/2021 | Priority | High |
| The actor:   * Guest   Summary:   * Use this case allows actors to register for an account to log into the account.   Goal:   * The account's role will be confirmed.   Triggers:   * Click the button "Register”.   Preconditions:   * The user's device was connected to the internet when logging in.   Publish conditions:   * Success: The account data is saved to the database * Fail: Displays the message "Your account has not been created".   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | The actor selects the button "Register" | The system will redirect user to the “Register” page. | | 2 | The actor enters their full name, phone number, password, re-enter the password | The system will display information on the web | | 3 | The actor chooses the register button | The system authenticates the phone number and sends a text message to the phone number with a 6-digit code | | 4 | The actor enters confirmation code | The code is displayed on the system, and the system returns the message "You have successfully created an account".  The system will go to "Home" screen |   Alternative Scenario:  Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | first | actor re-enter login information | The system returns to login screen |   **Relationships:**  Business Rules:   * Employees must log in as the user role | | | |

**Table 3.4:** Use-case specification registered with Google account.

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-4 | | | |
| Use Case No. | DT-4 | Use Case Version | 1.0 |
| Use Case Name | Sign up for an account with the Google account | | |
| Author | Chu Minh Hoàng | | |
| Date | 12/2020 | Priority | High |
| The actor:   * Guest   Summary:   * Use this case allows guests to register for an account to log into the account.   Goal:   * The account's role will be confirmed.   Triggers:   * Click the “Register” button.   Preconditions:   * The user's device was connected to the internet when logging in.   Publish conditions:   * Success: The account data is saved to the database. * Fail: Displays the message "Your account has not been created".   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | The actor selects the button "Register" | The system will redirect user to the “Register” page. | | 2 | The actor selects the button "Register with Google" | The system will redirect to the Google page | | 3 | The actor enters Google account name, and password | The system receives a response from the Google page and displays a message | | 4 | The actor select "Continue" | The system redirects to the home page |   Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | first | Guest re-enter login information | The system returns to login screen |   **Relationships:**  Business Rules:   * Employees must log in as the user role. | | | |

**Table 3.5:** Use-case specification to search motel by type, city, price, district

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-5 | | | |
| Use Case No. | DT-5 | Use Case Version | 1.0 |
| Use Case Name | Search motel by type, city, price, district | | |
| Author | Chu Minh Hoàng | | |
| Date | 6/2021 | Priority | High |
| The actor:   * Guest, User   Summary:   * Use this case for the motel search.   Goal:   * The actors can search motel by type, city, price, district to get information   Triggers:   * The actor selects the type you want to search, click "Search".   Preconditions:   * The actor 's device was connected to the internet when logging in.   Publish conditions:   * Success: The search data will be displayed on the search page. * Fail: Displays the message "No matching data found".   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | The actor enters information to search for in the search box | The system displays data on the search box | | 2 | The actor enters in the search motel by type, city, price, county | The system will display the data on the website | | 3 | The actor selects the button "Search" | The system will redirect the guest to the search page |   Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | 1 | The actor enters in the search bar | The system will display the data on the website |   **Relationships:**  Business Rules: | | | |

**Table 3.6:** Use-case specification publish motel

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-6 | | | |
| Use Case No. | DT-6 | Use Case Version | 1.0 |
| Use Case Name | Publish motel | | |
| Author | Chu Minh Hoàng | | |
| Date | 6/2021 | Priority | High |
| The actor:   * User   Summary:   * Use this case publish motel for the house rental.   Goal:   * Publish motel for the house rental.   Triggers:   * Actor selects publish button.   Preconditions:   * The actor's device was connected to the internet when logging in. * Log in as a User.   Publish conditions:  Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | Actor selects publish button. | The system displays publish page | | 2 | Actor fill information in step 1 and select next | The system moves to step 2 | | 3 | Actor fill information in step 2 and select next | The system moves to step 3 | | 4 | Actor fill information in step 3 and select next | The system moves to step 4 | | 5 | Actor fill information in step 4 and select next | The system moves to step 5 | | 6 | Actor fill information in step 5 and select next | The system moves to step 6 | | 7 | Actor choose payment with PayPal | The system moves to PayPal site to payment |   Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | 1 | The system will return the details page but will appear the message execution fail |  |   **Relationships:**  Business Rules:   * When the step is completed, the user can return to the completed step * About steps: * Step 1: Enter basic information about address, district, province, ... * Step 2: Enter information about title, price, description, … * Step 3: Detailed information about location, room number, … * Step 4: Information about the motel * Step 5: Confirm package registration. * Step 6: Review published information * Step 7: Payment * User can add or remove photos | | | |

**Table 3.7:** Use-case specification see the detailed motel

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-7 | | | |
| Use Case No. | DT-7 | Use Case Version | 1.0 |
| Use Case Name | View for details | | |
| Author | Chu Minh Hoàng | | |
| Date | 6/2021 | Priority | High |
| The actor:   * User, Guest   Summary:   * This case view for detail motel.   Goal:   * The actor sees detailed information about the motel.   Triggers:   * The actor selects the motel you want to view and clicks on the title or image.   Preconditions:   * The actor's device was connected to the internet when logging in.   Publish conditions:   * Success: The detailed search data for motel information is displayed on the details page. * Fail: Displays the message "No matching data found".   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | The actor chooses one of motel, if they want to see detail | The system moves to the details page |   Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | 1 | The system will return the details page but will appear the message "No matching data found". |  |   **Relationships:**  Business Rules: | | | |

**Table 3.8:** Use-case specification changes the password

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-8 | | | |
| Use Case No. | DT-8 | Use Case Version | 1.0 |
| Use Case Name | Change the password | | |
| Author | Phạm Huỳnh Thanh Lâm | | |
| Date | 6/2021 | Priority | High |
| The actor:   * User   Summary:   * Using this case allows the user to change the password.   Goal:   * Actor can change passwords.   Triggers:   * User selects "Personal Information." And choose “Change Password.”   Preconditions:   * The actor's device was connected to the internet when logging in. * Log in as Actor.   Publish conditions:   * Success: Successfully changed password. * Fail: message "Password change failed"   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | Actor selects "Personal Information" | The system redirects to the personal information page | | 2 | Actor selects "Change password" | The system moves to the password change page | | 3 | Actor enter the old password, and the new password, re-enter the new password | The system displays the information just entered | | 4 | Actor selects "Change password" | The system will confirm, and save it to the database, redirect the user to the login page again |   Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | 1 | Actor selects "Change password" | The system says "Password does not match" |   **Relationships:**  Business Rules: | | | |

**Table 3.9:** Use-case specification for chat

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-9 | | | |
| Use Case No. | DT-9 | Use Case Version | 1.0 |
| Use Case Name | Chat | | |
| Author | Phạm Huỳnh Thanh Lâm | | |
| Date | 6/2021 | Priority | High |
| The actor:   * User   Summary:   * Use this case allows the user to chat with the site administrator.   Goal:   * Actor can communicate with staff of the site.   Triggers:   * Actor selects widget in the bottom right corner of the web page.   Preconditions:   * The actor's device was connected to the internet when logging in. * Log in as Actor.   Publish conditions:   * Success: the chat window will appear, and the user performs the chat * Fail: display the frame to send the message via mail, the user can respond by mail   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | Actor selects widget in the bottom right corner of the web page. | The system displays widget chat | | 2 | Actor enters the message into the message box | Chat system displays information | | 3 | User selects "Send" or press Enter | The chat system will display the chat, and send it to the admin |   Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | 1 | Actor selects widget in the bottom right corner of the web page | The system displays the frame to send the message via mail, the user can respond via mail. |   **Relationships:**  Business Rules: | | | |

**Table 3.10:** Use-case specification for logged out

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-10 | | | |
| Use Case No. | DT-10 | Use Case Version | 1.0 |
| Use Case Name | Log out | | |
| Author | Phạm Huỳnh Thanh Lâm | | |
| Date | 6/2021 | Priority | High |
| The actor:   * User, admin, employee   Summary:   * Use this case to allow the actor to log out.   Goal:   * The actors are outputting outside of the website.   Triggers:   * Actors select "Log out."   Preconditions:   * The actor's device was connected to the internet when logging in. * Login as the actor   Publish conditions:   * Success: successfully logged out * Fail: Return to the homepage with the login image still displayed   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | Actors select the avatar icon. | The system displays information containing functions | | 2 | Actors select the function "Log out" | The system validates and redirects the page to the homepage |   Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | 1 | Actors select the avatar icon. | The system displays information containing functions | | 2 | Actors select the function "Log out" | The system returns the home page with the login image still displayed. |   **Relationships:**  Business Rules: | | | |

**Table 3.11:** Use-case specification for add employee

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-11 | | | |
| Use Case No. | DT-11 | Use Case Version | 1.0 |
| Use Case Name | Add employee | | |
| Author | Phạm Huỳnh Thanh Lâm | | |
| Date | 6/2021 | Priority | High |
| The actor:   * Root   Summary:   * Using this case allows Root to add the employee account.   Goal:   * Root adds admin to the website.   Triggers:   * Actor selects "Manage employee."   Preconditions:   * The actor's device was connected to the internet when logging in. * Log in as Root.   Publish conditions:   * Success: successfully added employee * Fail: add failed   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | Actor select the function to add employee | The system moved to the employee add-on page | | 2 | Actor enter the employee information, and password | The system validates and updates the database again |   Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | 1 | Actor select the function to add admin | The system moved to the employee add-on page | | 2 | Actor enter the employee information, and password | The system verifies, and updates the database again, after returning the main page, the employee information has not been added |   **Relationships:**  Business Rules: | | | |

**Table 3.12:** Use-case specification for edit admin information

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-12 | | | |
| Use Case No. | DT-12 | Use Case Version | 1.0 |
| Use Case Name | Edit employee information | | |
| Author | Phạm Huỳnh Thanh Lâm | | |
| Date | 6/2021 | Priority | High |
| The actor:   * Root   Summary:   * Using this case allows actor to edit employee information.   Goal:   * The actor edits employee information on the website   Triggers:   * The actor selects "Manage employee."   Preconditions:   * The actor's device was connected to the internet when logging in. * Log in as actor.   Publish conditions:   * Success: successfully edited admin information * Fail: Failure to fix admin information   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | The actor selects the function to edit employee information | The system moved to edit employee information | | 2 | Root edit employee information | The system validates, and updates the database again, goes to the admin page, and updated data |   Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | 1 | Root select the function to edit employee information | The system moved to edit employee information | | 2 | Root edit employee information | The system validates and updates the database again, moves to the admin management page, and the data has not been updated |   **Relationships:**  Business Rules: | | | |

**Table 3.13:** Use-case specification extension motel news

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-13 | | | |
| Use Case No. | DT-13 | Use Case Version | 1.0 |
| Use Case Name | Extension motel news | | |
| Author | Phạm Huỳnh Thanh Lâm | | |
| Date | 6/2021 | Priority | High |
| The actor:   * User   Summary:   * Use this case allows extension the motel news if the user's want.   Goal:   * Actor can communicate extension the motel.   Triggers:   * Actor must be able to access user dashboard.   Preconditions:   * The actor's device was connected to the internet when logging in. * Log in as a actor.   Publish conditions:  Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | The actor clicks manage publish button. | The system displays manage publish page | | 2 | The actor clicks extension button | The system moves to extension screen | | 3 | The actor edits the motel information, choose the service of motel and click extension button | The system displays manage public page |   Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | 1 | The system will return the details page but will appear the message "No matching data found". |  |   **Relationships:**  Business Rules:   * Newly extension motel must be approved by the staff before they can be displayed. | | | |

**Table 3.14:** Use-case specification search map

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-14 | | | |
| Use Case No. | DT-14 | Use Case Version | 1.0 |
| Use Case Name | Search Map | | |
| Author | Chu Minh Hoàng | | |
| Date | 6/2021 | Priority | High |
| The actor:   * Guest, User   Summary:   * Actor can search with Map.   Goal:   * Easier search for actor   Triggers:   * Actor selects type, city, province, district, street, price, distance and search their address they want.   Preconditions:  Publish conditions:   * Success: Show location in map and area motel * Fail: Map show difference location   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | The actor click type motel in the header or search motel | The system will direct the actor to the search screen | | 2 | The actor clicks “Map” button | The system will direct the actor to the map screen | | 3 | The actor can choose the city, province, district, street, price, type of motel  Or input location name | The system will display data the actor has choose in the screen | | 4 | The actor clicks find button | The system will render page and load list data the actor needs |   Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response | | 1 | People search Map | Don’t show location in map or wrong location |   **Relationships:**  Business Rules: | | | |

**Table 3.15:** Use-case specification log account

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE - DT-15 | | | |
| Use Case No. | DT-15 | Use Case Version | 1.0 |
| Use Case Name | Log account | | |
| Author | Phạm Huỳnh Thanh Lâm | | |
| Date | 6/2021 | Priority | High |
| The actor:   * Root   Summary:   * Using this case allows the actor to lock the account.   Goal:   * Root lock the account into the website   Triggers:   * Actor selects "Manage account."   Preconditions:   * The actor's device was connected to the internet when logging in. * Log in as actor.   Publish conditions:   * Success: Successfully log account * Fail: Failure log account   Main Success Scenario:   |  |  |  | | --- | --- | --- | | Step | The actor Action | System Response | | 1 | The actor select account they want to log | The system will display this account for the actor to see | | 2 | The actor clicks button log | The system will send request to back end to log this account |   Exceptions:   |  |  |  | | --- | --- | --- | | No | The actor Action | System Response |   **Relationships:**  Business Rules: | | | |

## **Sequence diagram**



**Figure 3.2:** Login sequence diagram



**Figure 3.3:** Sequence diagram of browsing management



**Figure 3.4:** Sequence diagram register diagram



**Figure 3.5:** Search sequence diagram



**Figure 3.6:** Publishing sequence diagram



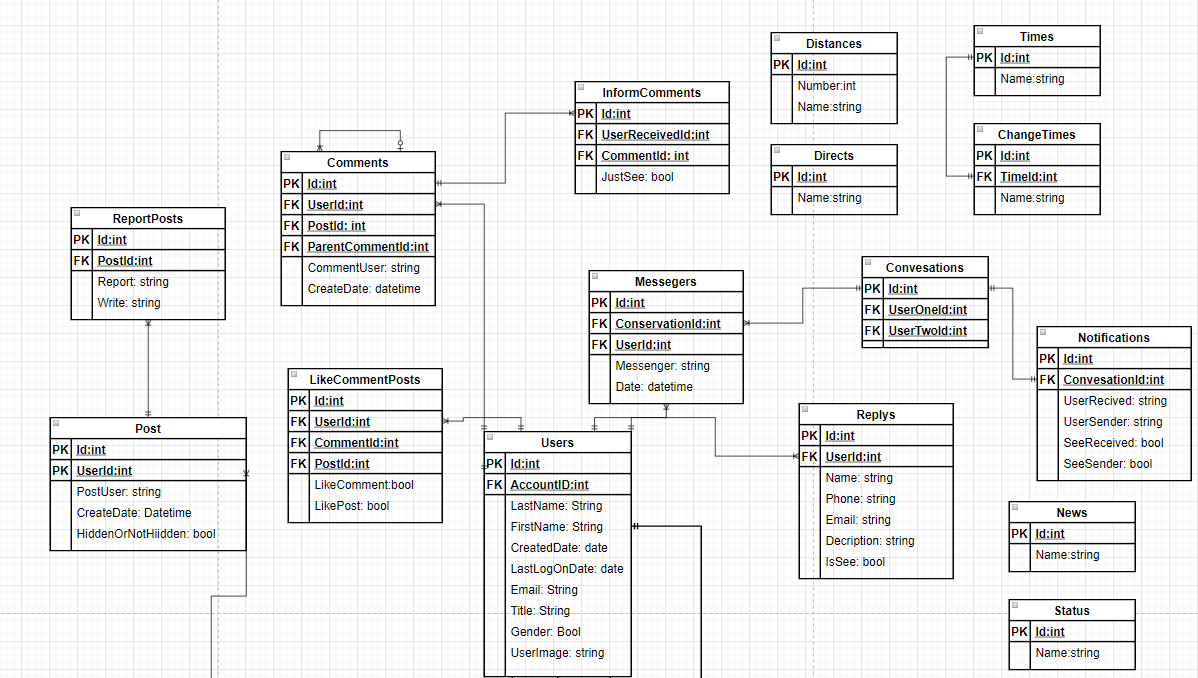
**Figure 3.7:** Sequence diagram of phone number change



**Figure 3.8:** Sequence diagram of bill management

## **Database design**

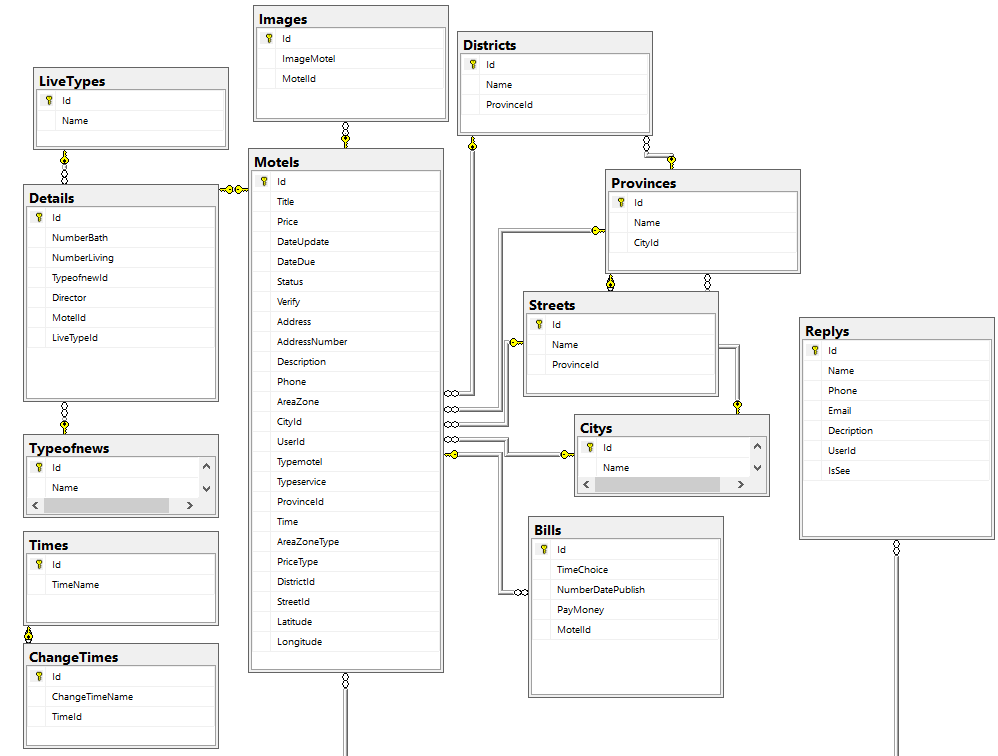
### **Entity relationship diagram**

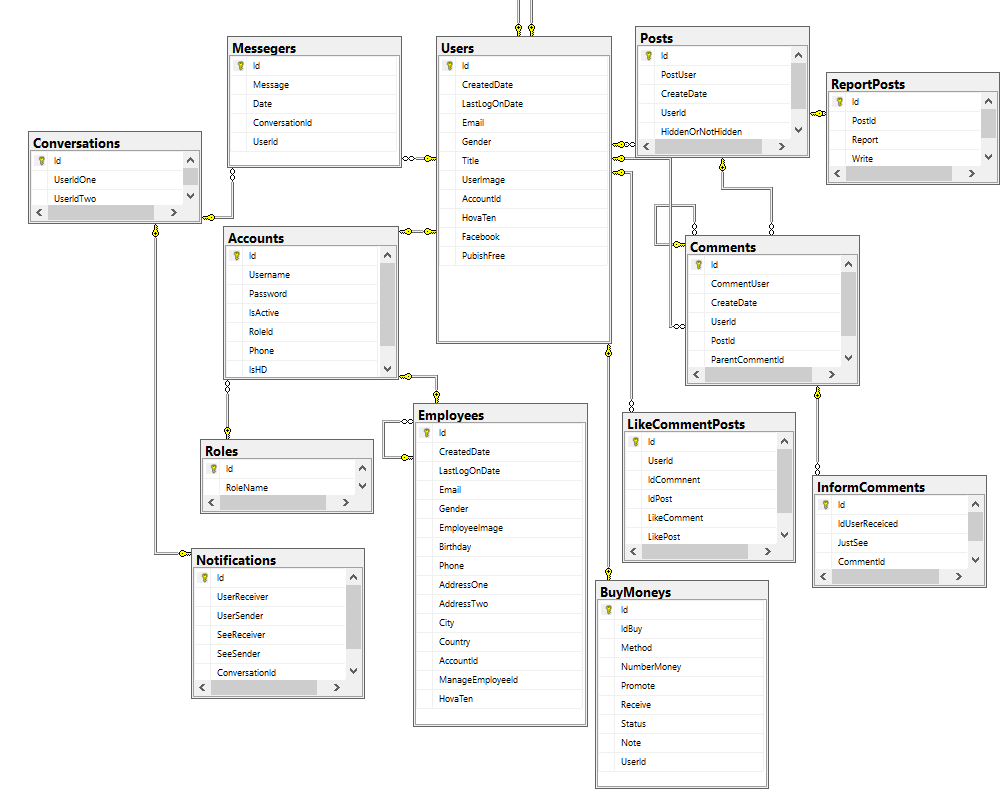




**Figure 3.9:** Entity association model

### **A relational model of tables in a database**





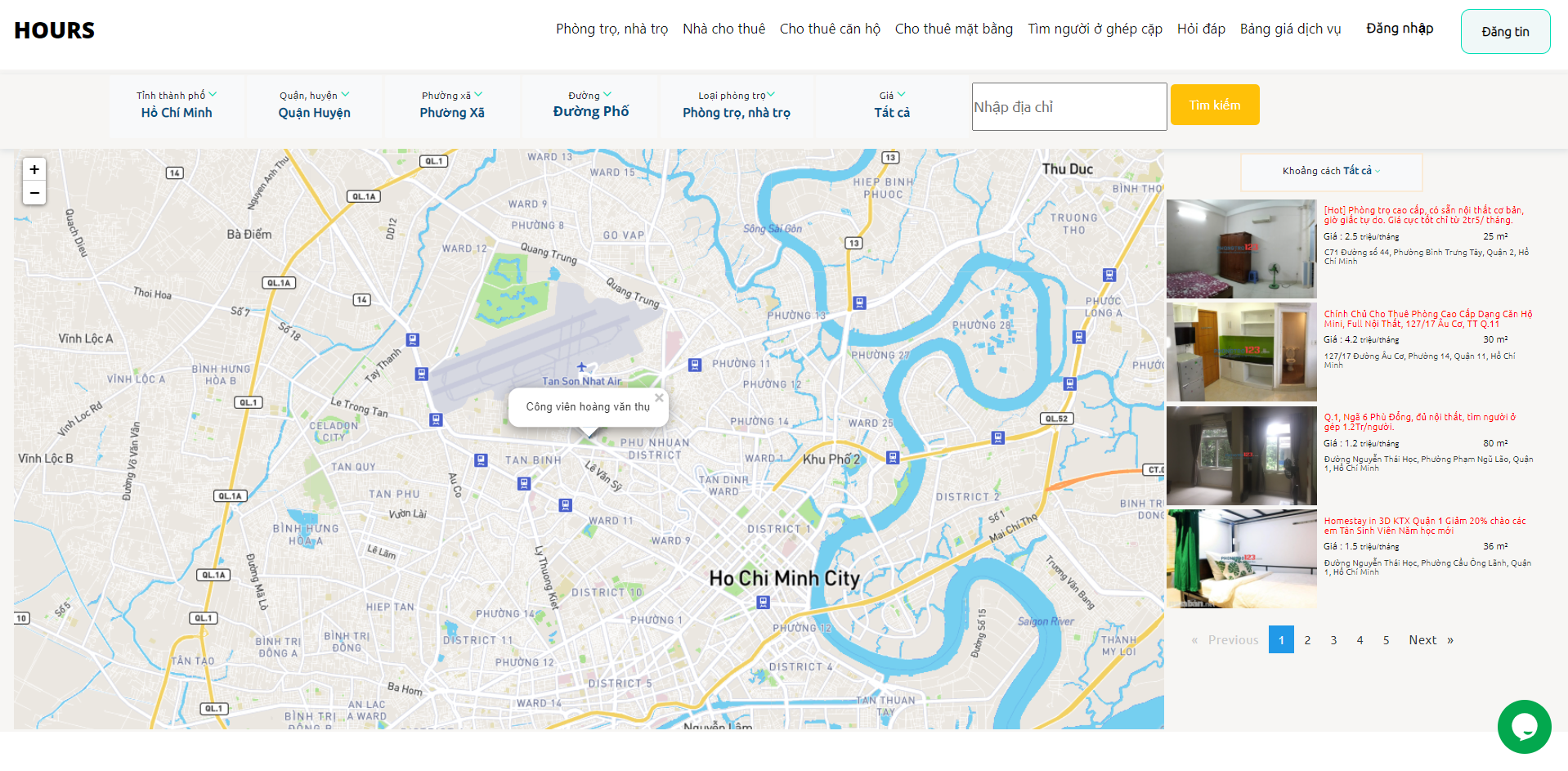
Graphical user interface

Description automatically generated

**Figure 3.10:** Relational model of tables in a database

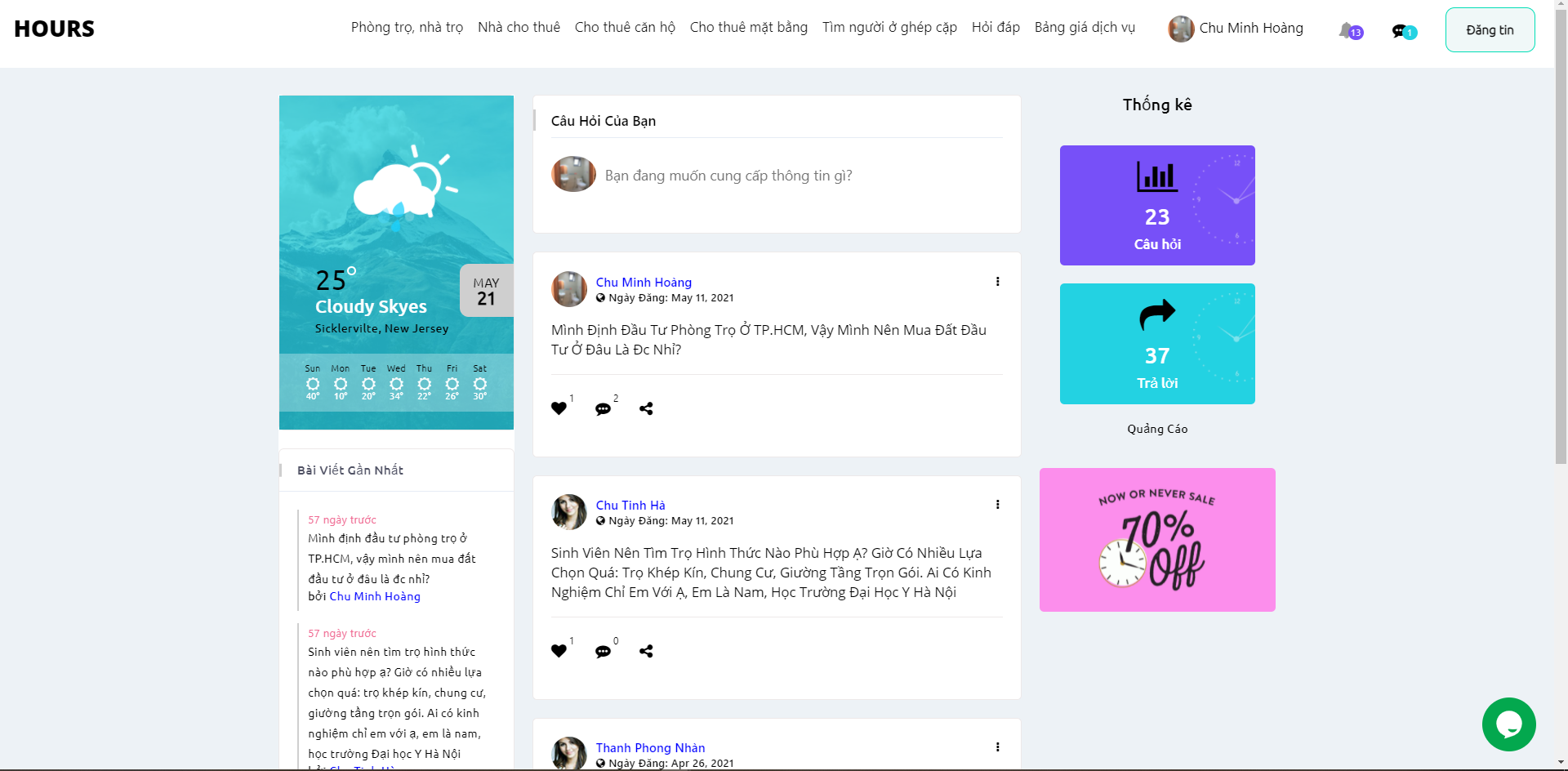
## **Design the theme.**

### **Map search screen**



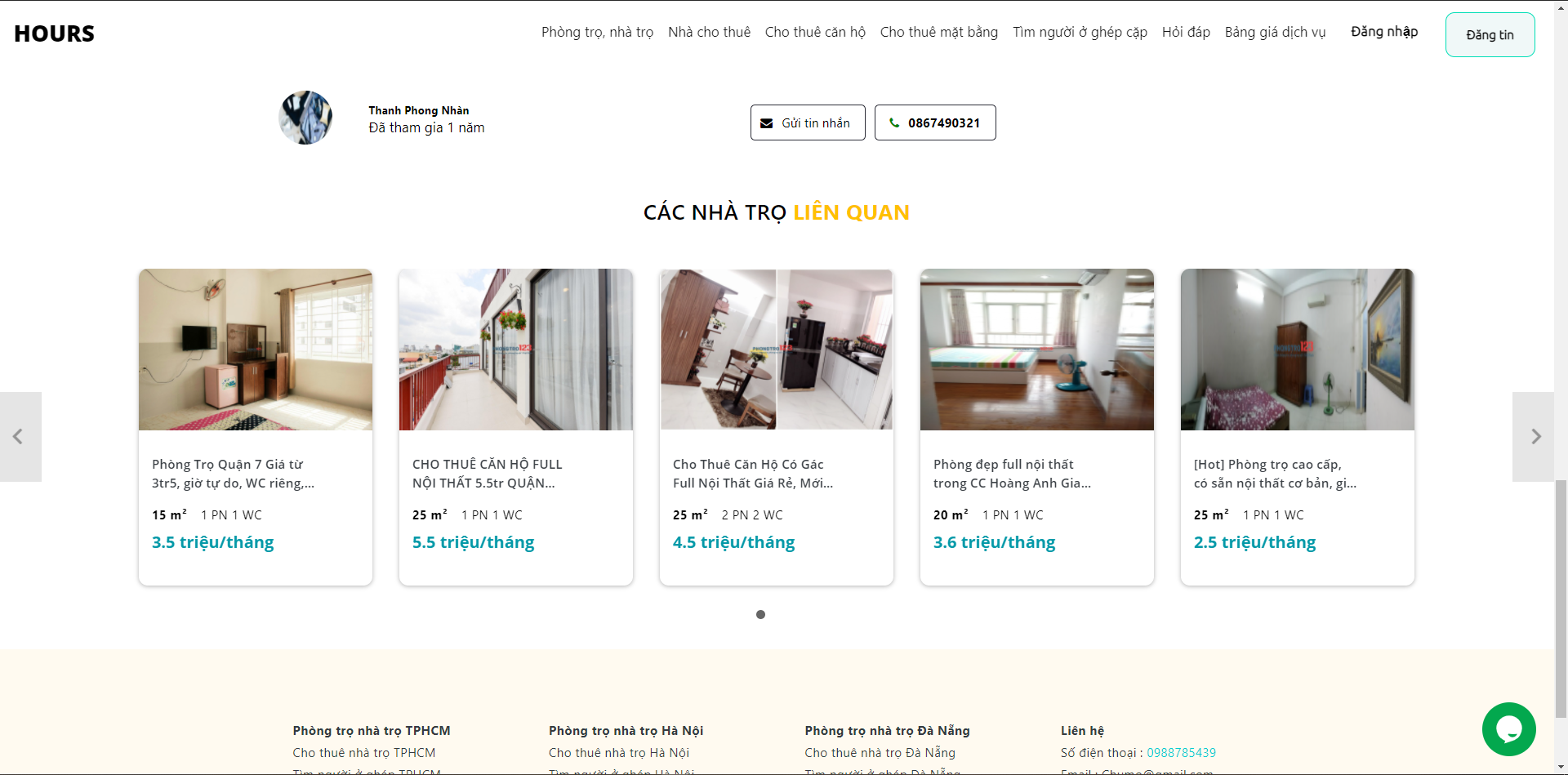
**Figure 3.11:** Map search screen

### **Forum screen**



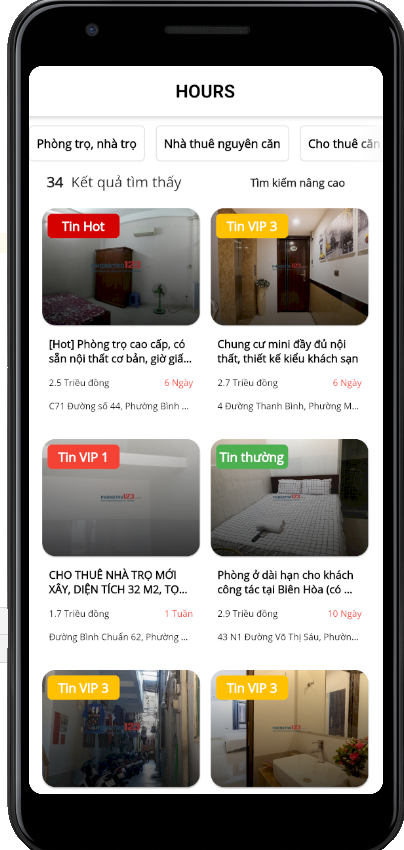
**Figure 3.12:** Forum screen

### **Recommend motel screen**



**Figure 3.13:** Recommend motel screen

### **Home page app screen**



**Figure 3.14:** App home and detail screen

# **CHAPTER 4 PRODUCT INSTALLATION**

## **Font end**

### **Application structure website**

* **e2e** (end to end): the directory used for testing.
* **node\_modules**: packages to be installed for use in the project.
* **src**: the project's code will be here. In the following section, we will learn more about this directory:
* **app**: The directory containing the application's main module.
* **assets**: directory containing resource files such as CSS, js, image.
* **environment**: a folder containing the environment setup files.
* **fav.ico**: file icon, display the icon (logo) of the website on the title bar of the browser.
* **index.html**: main file of the program.
* **main.ts**: This is the first file to be run when your application is run.
* **polyfill.ts**: file used for backward compatibility.
* **style.css**: file style for the application
* **test.ts**: The file contains the test cases for the project.
* **tsconfig.app.json**: file that contains the compiled configuration, "instructions" for the compiler code.
* **typings.d.ts**: file that manages definitions in Typescript.
* **angular.json**: contains project information
* **.gitignore**: file contains a list of files, folders not in the commit directory. You can learn more about git-ignore here (<https://git-scm.com/docs/gitignore>)
* **karma.conf.js**- The file contains the project requirements for unit testing.
* **package.json:** The file contains information about the libraries used in the project
* **tsconfig.json:** Contains extra rules when the project is translated like where (outer), what is the base path (base URL), what is the target version (target),

### **The main word processing in font-end**

* PayPal: allows creating payments with the amount of money from the website when the user publishes the news and connects to the payment page of PayPal.

Text

Description automatically generated

**Figure 4.1:** Script to embed third-party widget tawk. to

* Save the Firebase image: The image will be uploaded to the firebase storage, and then execute get download URL as the URL.

Text

Description automatically generated **Figure 4.2:** The function loads the image on firebase and uploads the URL link

* Sending messages to phones: thanks to firebase supports sending 6-digit code codes to the phone messages that users subscribe to.

Text

Description automatically generated

**Figure 4.3:** Function to send code to the phone

## **Back end**

### **Project API structure**

**Dependencies**: List of NuGet library packages to be installed and used in the project. Even the simplest empty project must use certain library packages.

**Properties**: contains project configuration information. If opened with Visual Studio, you will encounter the familiar graphical interface inherent in all C # projects. In ASP.NET Core, this information is stored in a JSON file that you can directly modify. This new way of saving is useful if you are using another code editor (such as Visual Studio Code).

**appsettings.json**: contains configuration information for application operations, such as connection strings, environment variables, command line parameters, etc.

**Program.cs**: This file contains the Program class that is responsible for the infrastructure configuration of the application. This class also contains the application's entry point. In general, the configuration inside the Program is almost unchanged in every project. If there are no special requirements, you will not need to make any adjustments in the Program class and the Program.cs file.

**Startup.cs**: This file contains classes with configuration methods for application operations, for example, which middleware you will use, what order is arranged in the pipeline middleware. You can also configure what types of services to use, like Dependency Injection, Logging. You will see some simple configurations in the next part of this lesson. Details on how to create a configuration class will see you in another lesson.

**Controllers**: Department is responsible for handling user requests made through the view. From there, the Controller gives appropriate data to the user. Besides, Controller also has the function to connect with the model.

**Migrations**: create a format that binds data to interact with a database

**Models**: This is the part that stores all of the application's data. The model is presented in the form of a database or sometimes simply one [XML file](https://monamedia.co/file-xml-la-gi/) normal. The model clearly shows operations with the database such as allowing viewing, accessing, processing data, ... [7]

### **Several Functions, and Procedures**

* The API methods:
* Get: get the data for search in website, filter by city, province, district, street, price, type motel

A picture containing table

Description automatically generated

**Figure 4.4:** API get motel for search by city, province, district, street, type and price

* Publish: get the json data of the model account, and then save it to the model data, then save it to the database

Text

Description automatically generated

**Figure 4.5:** API put account with hash password

* Get: get the JSON data for list motel API in app

Text

Description automatically generated with medium confidence

**Figure 4.6:** API get motel for app

* Get: Get JSON data by distance

Graphical user interface, text, application, email

Description automatically generated

**Figure 4.7:** API get motel for map screen

## **Technology**

### **Flash-Python**

* Structure of python flash
  + App.py: Home directory to initialize.
  + Data.py: Get data from backend via API.
  + Models.py: Declare the entity used as Motels.
  + Utils.py: Functions to process data.
  + remove\_stopwords: remove redundant words.
  + count: calculate the similarity of the data.
  + recommend: recommend motel.
  + Requirement.txt: Announcement of libraries and their versions
  + Env: Initialize the environment
  + \_\_pycache\_\_: When you run a program in python, the interpreter compiles it to byte code first (this is an oversimplification) and stores it in the \_\_pycache\_\_ directory.
* Libraries used
  + import re: The library is used to process Vietnamese strings.

**Table 4.1:** Table jaw Flask-Python

| **Jaw** | **Describe** |
| --- | --- |
| Find-all | Returns a list of matches |
| search | Returns a Match object if any of the values ​​in the string match |
| split | Returns a split list of strings at position match |
| sub | Replace positions that match the expression with another string |

* + import pandas as pd: The “pandas” library in python is an open-source library that effectively supports data manipulation. It is also a powerful data processing and analysis toolkit of the python programming language.
  + import nltk: NLTK or Natural Language Toolkit - Natural Language Toolkit, is a library written in Python that supports natural language processing. By providing common language processing mechanisms and techniques, it makes natural language processing easier and faster.
  + from nltk.corpus import stop-words: NLTK ships with stop-word lists for most languages. Is a library of unrelated words also known as "stop words".
  + from sklearn.metrics.pairwise, import cosine\_similarity: Calculate the cosine similarity between the samples
  + from sklearn.feature\_extraction.text, import CountVectorizer: Convert text document collection to token count matrix
  + import datetime: Python dates are not a data type of their own, but we can import a module named datetime to work with dates as date objects.
  + import requests: The request module allows you to send HTTP requests using Python.
  + from flask\_cors import CORS, cross\_origin: This package exposes a Flask extension that by default enables CORS support on all routes, for all origins and methods. It allows parameterization of all CORS headers on a per-resource level
  + from flask import Flask, jsonify, request: jsonify serializes data to JavaScript Object Notation (JSON) format, wrapping it in a Response object with application/json mime type.

### **Leaflet**

To use leaflet users will have to install leaflet library After installing the library, declare the scripts to use to display the map.[3]

Text

Description automatically generated

**Figure 4.8:** Import Leaflet API to Angular

Create a component to display the map.

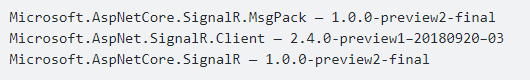
Main file format:

* + map.component.css: Contains methods for rendering called in HTML files.
  + map.component.html: Show display
  + map.component.spect.ts: Test code Contains
  + map.component.ts: Contains methods to dump data and display the map

### **Signal-R**

Signal-R is a free and open library that can be used to integrate real-time functionality in web applications. In the site apply Signal-R to do direct messaging with each other Signal-R is divided into 2 parts: font-end and back-end:

* Back-end: To use Signal-R, users must install the Signal-R package.



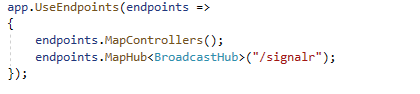
**Figure 4.9:** Package version Signal-R

Hub instantiation: Class Hub is client like angular will listen for different events Update Startup: Use CORS to agree to connect.



**Figure 4.10:** Config Signal-R

To connect font-end use endpoint



**Figure 4.11:** Config link connect Signal-R

* Font-end: To use Signal-R, users must install the Signal-R package.



**Figure 4.12:** Version Signal-R in angular

Create model, update app.component to listen from backend

### **Viet-Map**

The team uses the API provided by Viet-Map to get the location of the customer they want to search. The structure of the API uses [8]

A screenshot of a computer

Description automatically generated

**Figure 4.13:** Constructor Viet-map API

The method used is autocomplete:



**Figure 4.14:** Viet-map autocomplete API

Parameter:

**Table 4.2:** Viet-map autocomplete parameter

|  |  |  |  |
| --- | --- | --- | --- |
| Parameters | Style | Obligatory | Example |
| text | string | Yes | Tran Phu |
| focus.point.lat | floating point number | No | 10.756271 |
| focus.point.lon | floating point number | No | 106.66128 |
| boundary.rect.min\_lon | floating point number | No | 106.3564 |
| boundary.rect.max\_lon | floating point number | No | 107.01279 |
| boundary.rect.min\_lat | floating point number | No | 11.160291 |
| boundary.rect.max\_lat | floating point number | No | 10.376715 |
| boundary.circle.lat | floating point number | No | 10.756271 |
| boundary.circle.lon | floating point number | No | 106.66128 |
| boundary.circle.radius | floating point number | No | 35 |
| layers | string | No | address, venue |
| categories | string | No | [6000,6001](https://maps.vietmap.vn/docs/files/vietmap-poi-category.xlsx) |

Use get method to get parameters:

Text

Description automatically generated

**Figure 4.15:** Script to embed third-party widget tawk.to

Using scripts to embed third-party widgets tawk.to into the website, to help customers exchange to the website.

## **AppFlutter**

Packages used in pubspec.yaml

* + http: ^0.12.1: Use to read APIfrom backend via http.
  + flutter\_spinkit: "^4.0.0": A collection of animations with vibrations.
  + carousel\_slider: ^1.3.0: Using carousel slider in flutter.
  + url\_launcher: ^5.1.2: A Flutter plugin to launch a URL. Supports iOS, Android, web, Windows, macOS and Linux.
  + dropdownfield: ^0.0.2: Using dropdown fields in flutter.
  + dropdown\_search: ^0.3.1: Using dropdown search in flutter.
  + cupertino\_icons: ^0.1.3: content contains the default set of icon assets used by Flutter's Cupertino widgets.
  + google\_fonts: ^1.1.0: Google fonts are used.
  + flutter\_html: ^1.0.0-pre.1: Used to read HTML in data output

Includes dart files: API.dart, data.dart, detail.dart, filter.dart, Finding.dart, loading.dart, main.dart, search,dart.

API.dart(Call api from back end):

* + getMotels: Get data motel from back end.
  + getCities(): Take the city from behind.
  + getProvinces(int id): Get the district from the back end.
  + getTypes(): Get type message from back end.

Data.dart(Entity declarant)

* + Property: Declare property.
  + City: Declare the city.
  + Province: Declare province.
  + Type ion model: Type declaration.

Detail.dart: screen shows the details of the motel

* + Widget build: Main widget to run on detail screen.
  + Widget buildFeature: Widget to add image and text to the icon.

Filter.dart: filter display screen

* + \_getCity(): Get city data from API.
  + \_getType():Get data of news type from API.
  + \_getProvince(int id): Get county data from API.
  + Widget build: Main widget to run on filter screen.
  + \_sendDataToSecondScreen: Method of sending data to another screen.
  + Widget buildOption(String text): Button creation widget.

Finding.dart: Find Screen

* + \_getType(): Get data of news type from API.
  + \_getMotel: Get data of the motel from the API.
  + void initState(): Change status.
  + Widget build: Main widget to run on Finding trên screen.
  + List<Widget> buildFilter(): Horizontal scroll shows the type of motel.
  + List<Widget> buildProperties(): buildProperties creates a list of publishes.
  + Widget buildProperty(Property property, int index): buildProperty generates listing details by prooerty and index.
  + void \_showBottomSheet(): displays the sheet from the bottom of the screen to display filter information.

Loading.dart: The loading screen before entering the main screen

* + \_startTimer(): Set screen time to run.
  + void initState(): Change status.
  + Widget build: Main Widget to run on Loading screen.

Search.dart: search screen

* + \_getType(): Get data of news type from API.
  + \_getMotel: Get data of the motel from the API.
  + void initState(): Change status.
  + Widget build: Main widget to run on Desktop Search.
  + List<Widget> buildFilter(): Horizontal scrolling shows the type of motel.
  + List<Widget> buildProperties(): “buildProperties” creates a list of publishes.
  + Widget buildProperty(Property property, int index, String variable): BuildProperty generates the details of the list by property and index, variable.
  + void \_showBottomSheet(): displays the sheet from the bottom of the screen to show filter information.

# **CHAPTER 5 CONCLUSION**

## **Result**

1. Specialized essay

* Functional website

Guest:

* Guest is a person who has not logged in to become a user. Guests have basic functions such as search motel, view motel detail, view the service price table for publish motel, view list motel by type of them, view list motel under area, Filter motels, support guest.
* The website has many types of motels, so guests can choose motel to each type of motel they want to see and see details of each type of motel inside. When choosing the type of motel, there will be a place for users to enter the province and district in the search bar and can search more accurately through the filter.
* In addition, guests also know the website's publishing services and if they have any questions, they can contact the website's consultants.

User:

* User is the person who is logged in to the system that has guest functions and adds other functions such as login/register/logout, manage publish motel, transaction history, management personal information.
* Users can update their personal information on the website such as modifying personal information, changing login phone number or changing password. After published, they can manage their stories with manage publish motel page where they can see the new they Admin:
* Admin has main functions such as see the statistic the website, edit service for published motel, manage employee,
* Admin after logging in will be redirected to the main screen to see the website's statistics such as the top 5 people who publish the most motel, which type of motel has the most publishing, statistics by publishing revenue. In addition, the admin can edit the price of the service when publishing

Employee:

* Employees have main functions such as manage browsing publishes new and billing information
* Motel after user registration will be put in a queue for employee to review and browse, after employee browse, it will be displayed on the homepage. Also, employees can track website revenue

1. Graduation thesis

Continuity of specialized essay to graduate thesis some improvements were made. The interface of the website is fine-tuned in terms of display, the news is clearly displayed containing outstanding content according to different types of news, which catches the eye of users and especially complements and improves the functions:

* Website functions:

Guest:

* Guest refines search functionality makes searching easier by automatically suggesting results. In addition to expanding the search function with motel search with a map, guests can enter desired locations and the website will display motels near the address they are looking for.
* Guest creates an additional forum for users to ask questions or answer, guests can see questions and comments if they want to answer, they must log in to the system.
* The relevant motel can be suggested through the motel's title or description

User:

* On the forum users can comment or answer other questions.
* When the user's motel news expires, the user can renew the motel in manage publish motel.

Employee:

Employee have an additional forum management, where employee can monitor news in forum reflected by users and then make review on whether to hide the publish in forum or not.

* Mobile function:

About the phone will increase the user's interaction with the motel on the website. Users can view motel right on their phones and filter motel according to their wishes. You can view the detailed information of the motel along with the contact method, there are two contact methods are direct call and text message exchange via phone number.

## **Advantages**

* User/guest can search for motels via filter or advanced search (Search by direction and area of the motel), visually through the map, and can see the relevant motel nearby within the allowed radius.
* The website has integrated PayPal electronic payment service so that customers can pay easily.
* Guests who consider the motel satisfactory and intend to rent a motel can contact us directly through the website's messaging function or call phone number.
* In case the information about the motel expires, it can be renewed through the user's renewal function.
* The App can track and call or text the person who published the message.
* In website, suggestions for related motels through the motel selected by the user.

## **Defect**

* Website isn’t integrated with other popular e-wallets in the country.
* The site focuses a lot on the user and guest array, the functions of admin and employee are not developed much.
* Website display execution time is slow, sometimes it takes a while to display data.
* Features on the website are not synced with the mobile app.

## **Disadvantage**

* Website need to register a business account to integrate bank cards in Vietnam.
* Administration features, do not understand what the administrative functions of a website need.
* The security of the website has not been paid attention.

## **Lessons Learned**

* Team need allocate the execution time accordingly.
* The cohesion between team members plays an extremely important role. During the project, there was an unexpected event that was the Covid epidemic, two people in the team must communicate with each other more due to working online.
* Team must calmly solve the problem.
* Team must learn more deeply the algorithms applied in the subject.

## **Future work**

* Website must complete functions such as finding and publishing about motel.
* Website can top-up via different payment platforms both at home and abroad, payment platforms such as MoMo wallet, Moca wallet, Zalo-Pay, MasterCard, etc.
* Mobile complete synchronize website to make it easy for users to use. Currently the functions are only focused on guests, the features of the publish have not been implemented on the phone. Functions such as publish motel, managing publish, etc. can extend the use of the app to the person publishing the motel.
* Website must improve search by map, helping users to search more accurately.
* Website must improve response time (data, image, etc.).

# **REFERENCES**

[1] Đang Kim Thy, “Giới thiệu về ASP.NET Web API”

https://www.codelean.vn/2020/02/gioi-thieu-ve-aspnet-web-api.html

[2] Hieu Tran, “Angular là gì? Đặc trưng và tính năng cơ bản của Angular”

https://blog.itnavi.com.vn/angular-la-gi/

[3] Leaflet, “An open-source JavaScript library for mobile-friendly interactive maps”

<https://leafletjs.com/>

[4] Mai Chi, “Hướng dẫn tự học lập trình ADO.NET và Entity Framework”

<https://tuhocict.com/gioi-thieu-tong-quan-kien-truc-cai-dat-entity-framework>

[5] Mai Chi, “Cấu trúc dự án, cấu hình ứng dụng, middleware trong ASP.NET Core”

https://tuhocict.com/cau-truc-du-an-va-cau-hinh-ung-dung-aspnet-core/

[6] Phạm Minh Tuấn, “Sử dụng Realtime bằng Signalr trong .NET Core”

https://viblo.asia/p/su-dung-realtime-bang-signalr-trong-net-core-jvElaoJzKkw

[7] TopDev, “Firebase là gì? Có nên dùng cho các ứng dụng lớn?”

<https://topdev.vn/blog/firebase-la-gi/>

[8] Tawk.to, “Tawk.To – Phần Mềm Chat Trực Tuyến Hiệu Quả”

https://appnet.edu.vn/tawk-to-phan-mem-chat-truc-tuyen-hieu-qua/

[9] Vietmap, Tài liệu tham khảo API

https://maps.vietmap.vn/web/docs/

[10] Wikipedia, “Microsoft SQLServer”

<https://vi.wikipedia.org/wiki/Microsoft_SQL_Server>

# **APPENDIX**

## **Appendix I: Development environment settings**

To download, and install Visual Studio Code, follow these steps:

first. [Download the installation file of Visual Studio Code for Windows](https://quantrimang.com/url?q=aHR0cHM6Ly9kb3dubG9hZC5jb20udm4vdmlzdWFsLXN0dWRpby1jb2RlLTg2MzYw) (installation link https://code.visualstudio.com/).

2. Save the file on the device.

3. Double-click the file to launch the installation of Visual Studio Code on Windows 10.

4. Confirm the terms of the agreement.

5. Click the Next button.

6. Use the default installation location and click the Next button.

7. Use the default Start menu settings and click the Next button.

8. (Optional) Check the option Add “Open with code” action to Windows explorer file context menu.

9. (Optional) Check the option Add “Open with code” action to Windows Explorer directory context menu.

## **Appendix II: Creating an Angular application**

* Step 1: Install Angular libraries and tools: Install Angular CLI, type command at terminates # npm install -g @ angular / CLI.
* Step 2: Create an Angular project.

Create a directory containing your projects, for example, angular- learning, then run VS Code open that folder to work with (you can also work without openings Code right). From VS CODE type CTRL “+” to open the terminate, from there type the following command, to create the basic initialization project # ng new <Project name>.

* Step 3: Go to the project# cd <Project Name>.
* Step 4: Run the newly created project# ng serve.

## **Appendix III: Creating an App Flutter**

* Step 1: Install flutter
* Download the flashing zip file from the homepage and extract it.
* Add environment variable.

Graphical user interface, application

Description automatically generated

* Step 2: Create an application project.
* Install plugin flutter and dart.

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

* Create virtual server.

Graphical user interface, text, application

Description automatically generated

* To create a new Flutter project, click the Create New Project button on the welcome screen. In the New Project dialog box, select Flutter and click Next.
* Step 3: Run the project by pressing the Run button.