

|  |
| --- |
| **Project Document** |
| **Car Rental Service Website** |

|  |  |
| --- | --- |
| **Group 1** | |
| **Group Members** | **Hoang Anh - C2010G0141**  **Vu Duc Anh - C2108L0272**  **Nguyen Dinh Duy  - C2105L0183**  **Pham Dang Duc - C2108L0255**  **Nguyen Trung Duc - C2108L0265** |
| **Supervisor** | **Dao Quang Linh** |
| **Project code** | **CRSW** |

**- Hanoi, 12/2023 -**

**Table of Contents**

[Acknowledgement 4](#_heading=h.30j0zll)

[Definition and Acronyms 4](#_heading=h.3znysh7)

[**1**](#_heading=h.3znysh7) **Project Introduction 5**

[1.1](#_heading=h.2et92p0) Project Information 5

[1.2](#_heading=h.tyjcwt) Project Team 5

[1.3](#_heading=h.3dy6vkm) Product Background 5

[1.4](#_heading=h.1t3h5sf) Existing Systems 5

[1.5](#_heading=h.4d34og8) The proposed system 5

[1.6](#_heading=h.2s8eyo1) Project Scope & Limitations 5

[**2**](#_heading=h.17dp8vu) **Project Management Plan 6**

[2.1](#_heading=h.3rdcrjn) The purpose and target of a project 6

[2.2](#_heading=h.35nkun2) Management Approach 6

[2.3](#_heading=h.2jxsxqh) Responsibility Assignments 7

[2.4](#_heading=h.z337ya) Project Communications 7

[2.5](#_heading=h.3j2qqm3) Project Work Plan 7

[2.6](#_heading=h.4i7ojhp) Configuration Management 7

[**3**](#_heading=h.2bn6wsx) **Software Requirement Specification 8**

[3.1](#_heading=h.qsh70q) User Requirements 8

[3.2](#_heading=h.3as4poj) Functional Requirements 9

[3.3](#_heading=h.49x2ik5) Non-Functional Requirements 10

[3.4](#_heading=h.2p2csry) Other Requirements… 10

[**4**](#_heading=h.147n2zr) **Software Design Description 11**

[4.1](#_heading=h.3o7alnk) 1. System Design 11

[4.2](#_heading=h.32hioqz) 2. Database Design 11

[4.3](#_heading=h.1hmsyys) 3. Detailed Design 11

[**5**](#_heading=h.2grqrue) **Software Testing Documentation 12**

[5.1](#_heading=h.vx1227) 1. Scope of Testing 12

[5.2](#_heading=h.3fwokq0) 2. Test Strategy 12

[5.3](#_heading=h.19c6y18) 3. Test Plan 12

[5.4](#_heading=h.28h4qwu) 4. Test Cases 12

[5.5](#_heading=h.37m2jsg) 5. Test Reports 12

[**6**](#_heading=h.1mrcu09) **Release Package & User Guides 13**

[6.1](#_heading=h.46r0co2) 1. Deliverable Package 13

[6.2](#_heading=h.111kx3o)  2. Installation Guides 13

# **Acknowledgement**

During the time of studying, we have well provided and imparted all the necessary and highly valuable professional knowledge by our teacher in Aptech. We have our faithful, careful style and creative mind in studying in the future.

Project is a good opportunity to apply appropriate knowledge and subjects. While performing projects, we have actual and valuable experience. After one term, we were getting all one’s thoughts together and working on a project. During executing this project, we have received a lot of helping from our guider, teachers.

Especially, we wish to thank our teacher - Mr: Dao Quang Linh - for his enthusiastic, thoughtful, and careful guidance.

# **Project Introduction**

## **Project Information**

* **Project Information**
* Project Name (English): Car Rental Service Website
* Project Name (Vietnamese): Ứng dụng website dịch vụ cho thuê xe ô tô
* Project Code: CRSW
* Project Type: Web API Project
* Timeline: From 27th September to 8th November

## **Project Team**

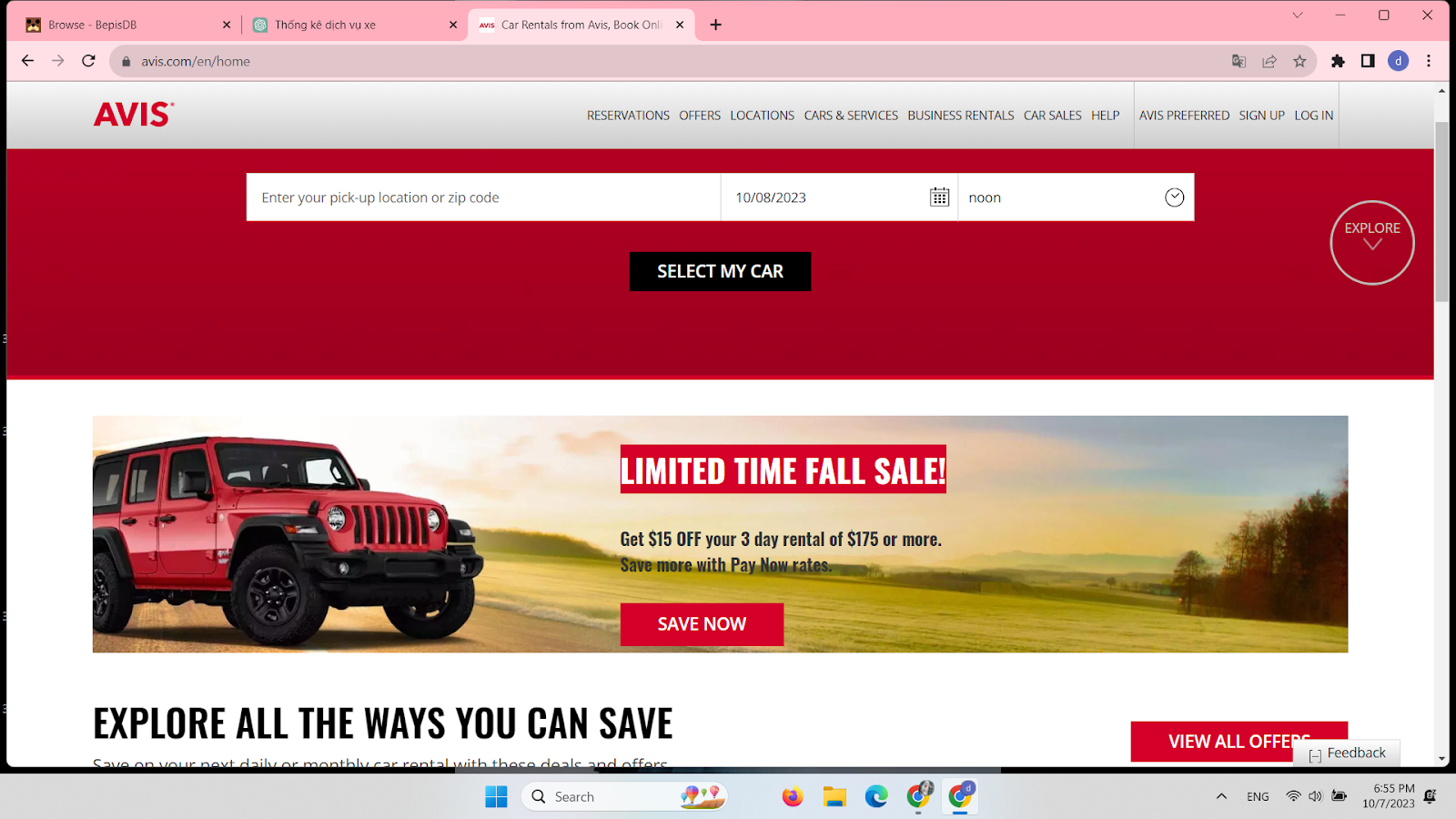
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Full name** | **Roll number** | **Position** | **Phone Number** | **Email** |
| Hoàng Anh | C2010G0141 | Member | 0988869520 | anh.h.953@aptechlearning.edu.vn |
| Vũ Đức Anh | C2108L0272 | Member | 0975502334 |  |
| Nguyễn Đình Duy | C2105L0183 | Leader | 0368415649 | duy.nd.1849@aptechlearning.edu.vn |
| Phạm Đăng Đức | C2108L0255 | Member | 0704125993 | duc.pd.1930@aptechlearning.edu.vn |
| Nguyễn Trung Đức | C2108L0265 | Member | 0836710765 | duc.nt.1940@aptechlearning.edu.vn |

## **Product Background**

* For each person and each family, a car is not only a means of transportation but also a major family asset and not every family can afford it. Not only that, some families, although well-off and able to buy a car, still choose to use self-driving car rental services because it brings many conveniences.
* Nowadays, car prices are increasing very rapidly, compared to the average income of each person, owning a car is very difficult. Based on the news website (https://baotintuc.vn), car prices in 2023 have increased 2.5 times compared to the market price of previous years, leading to workers gradually shifting to renting self-driving cars. According to Mordor Intelligence, the self-driving car rental market in Vietnam is valued at 463.19 million USD in 2021, and is expected to reach 884.71 million USD in 2027 with a compound annual growth rate of 13.82% in the period 2022 - 2027. Meanwhile, according to Statista, the revenue of the self-driving car rental market in Vietnam in 2023 is expected to reach 0.92 billion USD. In addition, the revenue of this market is expected to reach 1.19 billion USD by 2027 with a compound annual growth rate of 6.57% in the period 2023 - 2027.

## **Existing Systems**

* We have some overviews of similar existing solutions, overview of existing methods, limitations of some existing applications.
* Avis.com



|  |  |
| --- | --- |
| Pros:   * Easy to use. * Support for search owners’s names; informations of owners, car….; times-setting * Personalized. * Information is accurate and reliable. | Cons:   * Users and car owners must download apps from the web to fill in their necessary information. * The interface is rough and looks unfriendly * No support Vietnamese |

* Xego.vn



|  |  |
| --- | --- |
| Pros:   * Personalized. * Information is accurate and reliable. * Support Vietnamese | Cons:   * Users and car owners must download app of web to fill in their necessary informations. * Less informations. |

## **The proposed system**

* Online car rental service is a step forward from traditional car rental service. Although building an online car rental service has overcome many limitations of traditional services, the number of car rental websites with a user-friendly interface on the web is not many. The avis.com and xego.vn websites, as listed as examples, are among the largest service websites in this field, but there are still limitations related to personal information registration. In the immediate future, we want to create a website that has the function to fill in important information of individual car renters and owners. Besides, we want the website to have a convenient, user-friendly interface.
* The application will have functions to help users in their work most effectively. Functions such as:   
   1. **Display images and information of rented vehicles** -> help customers and users access a variety of models and types of vehicles that customers want instead of the limited number of models compared to traditional car rentals.  
   2. **Convenient and proactive** -> help customers and users proactively plan and book calendars at any time and in the most comfortable position.  
   3. **Reputation and quality** -> the app is an intermediary that accurately verifies both the customer and the owner of the rental car. Become a tool to ensure the quality and transparency of car leases and vehicle documents.  
   4. **Safety and privacy** -> legal support application, including customer support teams and vehicle owners when issues related to the agreement between the two parties occur.  
   5. **Reviews** -> show reviews of users to provide information to customers and consumers so that they can make wise rental/purchase decisions and have a comprehensive review of the product

## Project Scope & Limitations

**About our project scope:**

* **Consumer demand**
  + Help users and customers who want to rent self-driving cars access the service easily, flexibly and conveniently.
  + Create a platform that seamlessly connects the person who needs to rent a car with the owner of the rental car.
  + Become a tool to help car rental owners easily manage, do business, access more customer files.
* **Conditions and feasibility of the project:**
  + The project has a low investment, minimizes costs, helps users not to pay any additional fees from the website
  + The demand of customers in the market is high, the application has many opportunities to reach users
* **Limitations**
  + The website requires a stable internet connection to launch smoothly.
  + The website does not have an accompanying mobile application.
  + Verifying documents takes time and requires precise operations as instructed.

# **Project Management Plan**

## **The purpose and target of a project**

### **Project Objectives**

* This report is a guide to regulation and supervision of organization and plan of the project. It describes an overview of the project including the proposed system, boundaries and development environment of the capstone project. The ultimate purpose of Project management plan is to clearly deﬁne the roles, responsibilities and the processes of team members to assign tasks and deadlines.

### **Project Risks ( Risk Breakdown Structure (RBS)** )

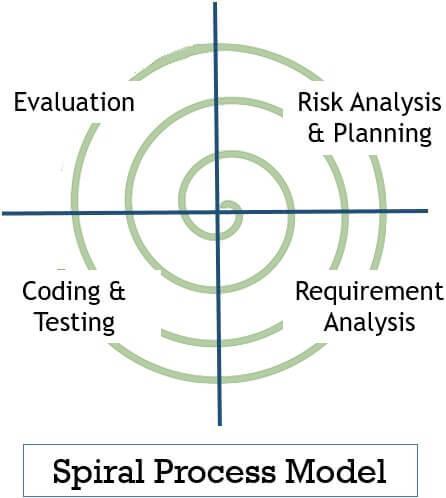
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Probability | Impact | | | |
|  | Low | Medium | High |
| Low |  |  |  |
| Medium |  |  |  |
| High |  | Risk 1 | Risk2 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Description | First Indicator | Prevention  Approach | Mitigation  Approach |
| 1 | Team members are spaced apart | * Teamwork’s efficiency decrease. * Hard to coordinate team members. * It is difficult for group members to meet each other |  |  |
| 2 | Lack of equipment and machinery for the project | * Slow down the completion of tasks * Difficulty sharing information and plans of the project * Unable to do work handed over |  |  |
| 3 |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## **Management Approach**

* ***Software Process Model***
* **We choose this Spiral Model, because it has many advantages:**
* Spiral model enables the better cost estimation
* Spiral model provides continuous and repeated development which helps in risk management.
* Spiral model provides the fast development and the features are added in a systematic manner.
* In a spiral model, client’s get the opportunity to see the software/product after every cycle.
* The spiral model in SDLC is considered as the most preferable model for large and complex projects/software.
* **There are some drawbacks which spiral model consists of:**
* The spiral model is expensive due to the high level of expertise required for risk analysis. Also, the projects take time to develop that causes the overall expenses.
* Due to its high cost, the spiral model is not suited for small projects.

### **Project Process**



* 1. **Planning**

This phase includes the planning process, tasks, resource defining, team planning, timelines and gathering of other project related information. Planning phase includes the estimating costs, schedule for iteration. Once the planning finalizes the team proceed to the next step i.e., Risk Analysis.

* 1. **Risk Analysis**

In Risk Analysis phase, the project prototype is put up for the solution. All the planning which is being planned in the planning phase proceeds with the risk analysis phase to overcome the problems and risks at the beginning stage of project development. Technical and management risk are the factors that are included in the risk analysis phase.

* 1. **Engineering & Execution**

In this phase, the execution part is being done by engineers and developers. The planning and risk analysis once finalized is being processed in the execution phase where all the coding, testing and deploying of software takes place.

* 1. **Evaluation**

In the evaluation phase, the product is being assessed by the client and provided with the revert if any changes required from client side. Evaluation phase includes all the above phases whether it is about planning, risk analysis, engineering & execution the client goes through each phase for evaluating the product or software.

### **Quality Management**

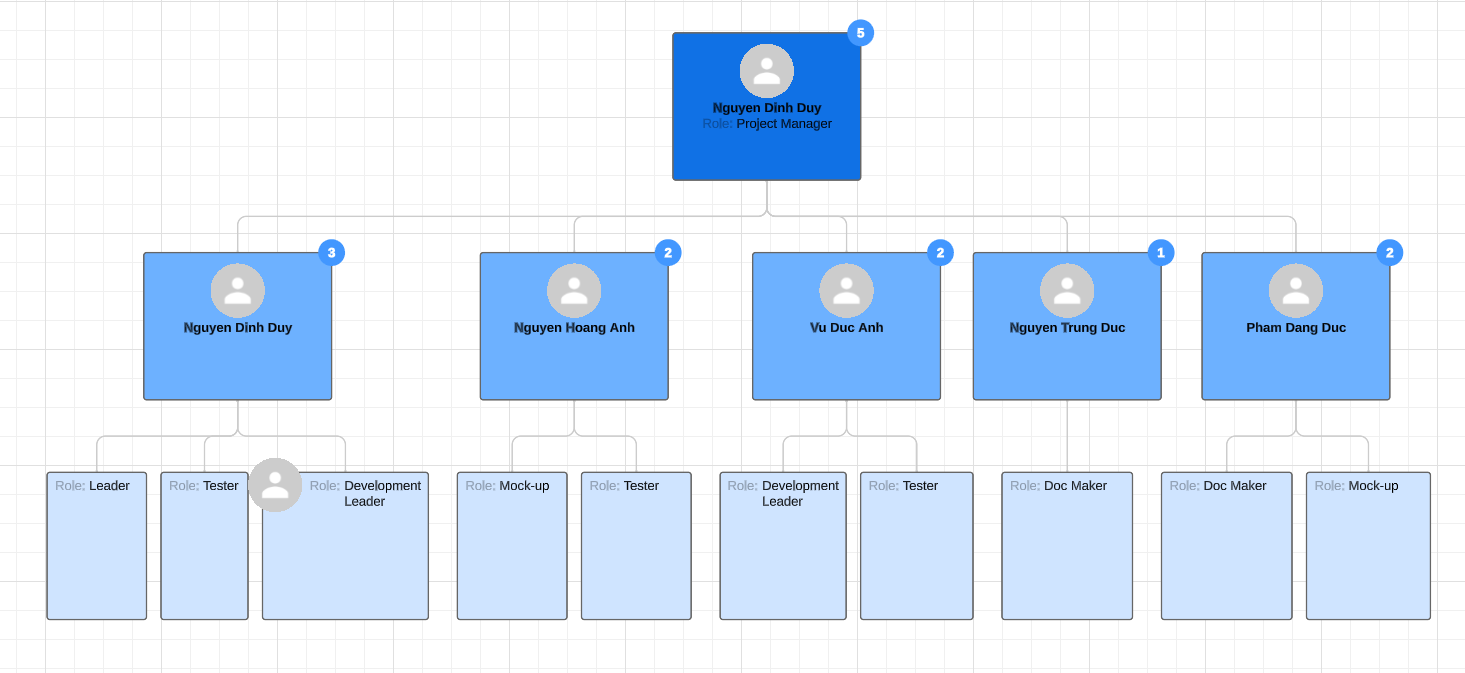
*[Provide the approach you would apply to improve the project quality, reach the project quality objectives*

### **Training Plan**

*[You need to plan the training activities in case any of your team member lack of knowledge/skills to handle the project works]*

*=>*

## **Responsibility Assignments**



|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Name** | **Roles** | **Responsibilities** |
| **1** | **Nguyen Dinh Duy** | * Project Manager. * Technical Leader. * Development Team Leader. | * Managed and lead the team project. * sAnalyzed and built materials for the project implementation process. * Developed project plan and implementation schedule. * Assigned responsibilities and tasks to members. * Maintained an organization’s work flow and kept its lines of communication open. * Monitored the progress of project implementation * Identified, assessed and mitigated project risks for successful project completion. * Communicated with other members to guarantee the project was on schedule and within scope. * Reviewed and supported requirement analysis. * Led requirement analysis team. * Understood clearly about project requirements and wrote SRS. * Maintained all documents by preparing standard templates, managing the process of creating, reviewing and updating documents in a team. * Researched algorithms and learned new techniques and architecture to use for projects. * Guided other members in the team to understand and to apply new techniques to the project. * Supported team members. |
| **2** | **Vu Duc Anh** | * Development Team Member * Design Team Leader | * Reviewed and supported requirement analysis. * Understood clearly about the project requirements and wrote SRS. * Identified certified solutions to meet the requirements. * Had a basic understanding of technology used for the project. * Guided other members to understand and to apply the technology to the project. * Summarized and made the project weekly report * Supported other team members. |
| **3** | **Hoang Anh** | * Development Team Member. * Design Team Member | * Had a basic understanding of technology used for the project. * Confirmed developing plan and assigned coding functions to other members. * Supported team in developing some functions. * Supported other team members. |
| **4** | **Nguyen Trung Duc** | * Development Team Member. * Test Team Member. * Document Team Member | * Had a basic understanding of technology used for the project. * Read & understand SRS and deploy document. * Optimized coding. * Executed testing. * Supported other team members. |
| **5** | **Pham Dang Duc** | * Test Team Leader. * Document Team leader. | * Supported requirement analysis. * Executed test cases, recorded test case results, documented and tracked defects, and performed test coverage analysis. * Supported team to develop some functions. |

## **Project Communications**

* Team meetings are held twice a week, on Wenesday and Saturday. Face-to-face meetings are the most effective way to understand requirements and communicate on the project. The outcomes of the meeting are the report of the completed and not completed work to improve and corresponding solutions for the project.
* All members will perform the task assigned by the leader on Thursday. On Monday, members will report what they have done in a week. After that, the team will work together to solve the problem and have plan for the coming week.
* **Communication Channels:**
* Email.
* Facebook.
* Phone.
* Face-to-face meetings
* Zalo
* **File Management:**
* Google Drive.
* Github

## **Project Work Plan**



## **Configuration Management**

### **Document Management**

* We manage project document by using Microsoft Word 2016
* Microsoft Word is an editing program that allows users to work on text documents with many effects such as fonts, text colors, graphic effects, multimedia effects, audio, video.
* Through Microsoft Word formatting tools, you can create professional quality documents effectively. Besides, Word also includes editing and revision tools that help you collaborate with people easily.
* Microsoft Word has the following functions: Drafting contracts and minutes; Drafting reports; Create CV; Create mail; Create envelopes; Watermark; Autocorrect; Document protection; Count from; Set display mode; …

### **Source Code Management**

* We manage project source code by using Git/GitHub
* **Manage source code easily**
* When you create a repo, the entire source code of that repo is saved on GitHub. Here, you can review your progress through comments after each commit. And the beauty here is that many people can make the same repo.
* The first benefit is that you know who committed and what. Next, your source can develop into many branches. Principles for working with branches like this: You can branch many branches to develop the project. But in the end, you have to merge back into the MASTER branch to get the complete project.
* **Tracking changes across versions**
* When there are many members working on a project, it is quite complicated to keep track of revisions - who changed what, when and where those files are stored. Don't worry because GitHub has taken this into account for you, by always saving the changes you push to the repository. Similar to Microsoft Word or Google Drive, you have a version history in case previous versions are lost or not saved.
* **Github is a great resource**
* With the Explore function, you can track and search for open source projects according to the technology pattern you like. Github supports code search regardless of whether it is a separate project or a website. In addition, this platform also has quite good SEO so users can search for any code string shared publicly.

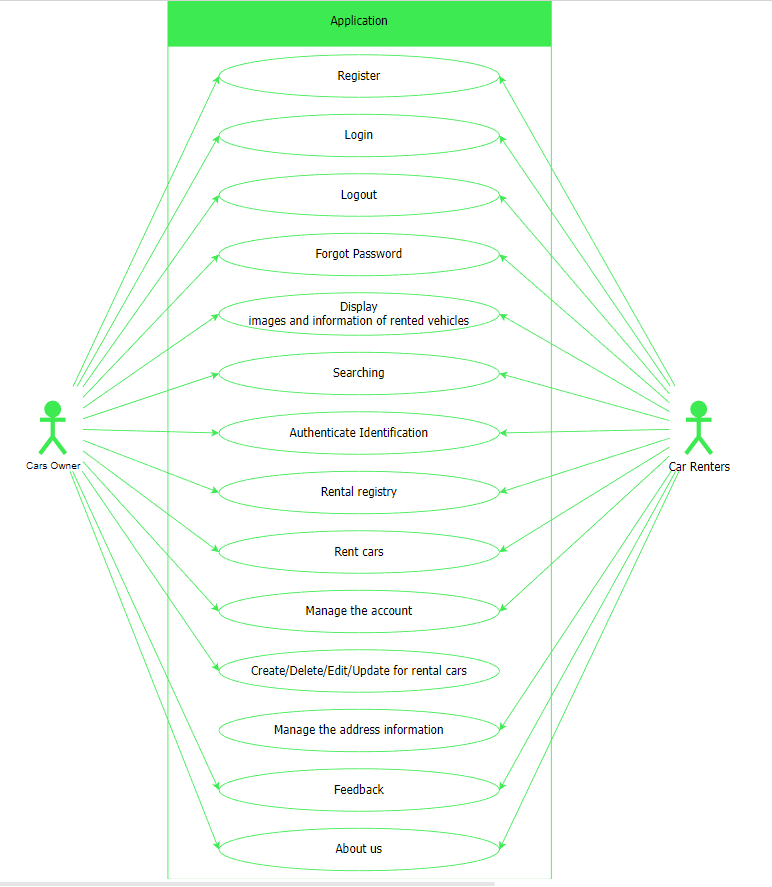
### **Tools & Infrastructures**

* 1. **Tools**
* **Version Control Systems:** Git 2.13.0 or higher.
* **Source Code Management Systems:** GitHub.
* **Project Management Tools:** Microsoft Office 2016 (Word, Excel, PowerPoint).
* **Diagram Tools:** https://app.diagrams.net ( website ).
* **Design Tools:** Photoshop CC 2019.
* ETC….
  1. **Techniques**
* **Programming Language**
* ***English***
* ***Vietnamese***

# Software Requirement Specification

## User Requirements

*[Provide specification of the user requirement: actor, use case diagram(s), use case description, etc.]*



***FIRGURE 3.2.1.2*** *Use Case Diagram*

**UC-01 : Register:**

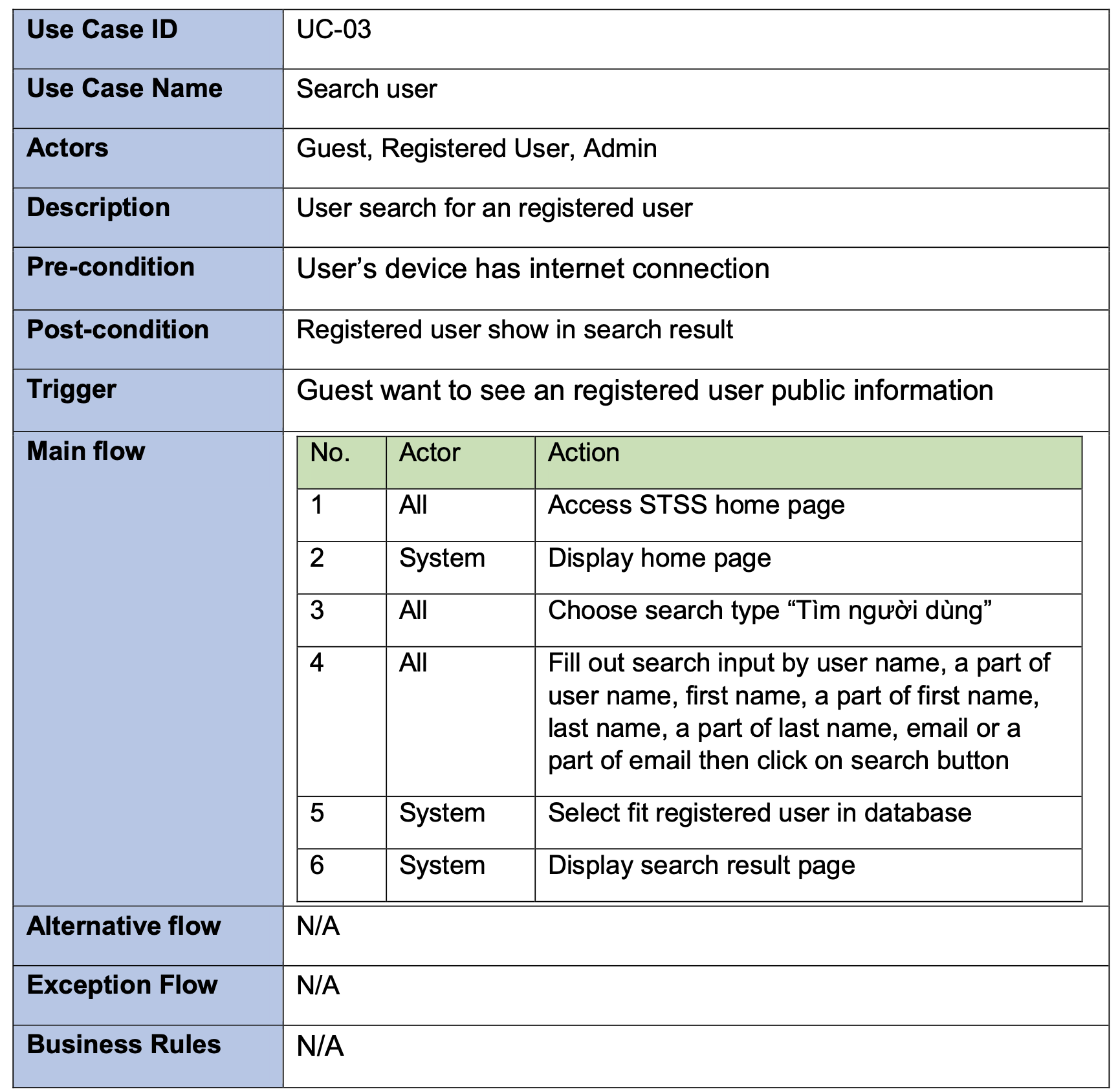
|  |  |
| --- | --- |
| Use Case ID | UC - 01 |
| Use Case Name | Register |
| Actors | Guest |
| Description | Register an account in the system. |
| Pre-Condition | * User is in login screen. * Internet permission |
| Post-Condition | * An account is registered. * Go to home screen. |
| Trigger | * Tap on “Sign Up” button. * Tap on email field. * Tap on name field. * Tap on password field. * Tap on “Register” button. |
| Main flow | |  |  | | --- | --- | | Actors | System | | 1.User click on “Sign Up” button |  | |  | 2. A resignter form will show up | | 3.User input form: Phonenum/Email, Name, Password, re-Password. Click and check box |  | |  | 4. The system updates user information -> adds the account to the database | | 5. Back to application interface |  | |
| Alternative flow | * When the user does not enter the required information in the registration form -> There are required fields that validated the user to fill in. * The “Confirm register” button will not available to click -> need to refresh the website |
| Exception flow | User enter not validated name, email or password. |
| Business rule | In order to log in to the system, user name and password are required. |

**UC-02: Login:**

|  |  |
| --- | --- |
| Use Case ID | UC – 02 |
| Use Case Name | Login |
| Actors | All user |
| Description | - This use case allows users who has username and password to login to the system to use system’s functions. Function that users can use must be allowed by admin.  - Allow user to login. |
| Pre-Condition | * User is in Home Screen * User have not logged in yet. * Internet permission. |
| Post-Condition | * Logged account. * Go to Home Screen. |
| Trigger | * Tap on navigator icon. * Tap on “Login”. * User enters email/phonenum and passwords. * Tap on “Login” button. |
| Main flow | |  |  | | --- | --- | | Actors | System | | 1. Click on header - > “Sign in” button |  | |  | 2. Show “Login” screen with: Email, Password. | | 3. Enter email, password. Then, tap on “Sign In” button. |  | |  | 4. Show screen “Home Screen” with data of this account. | |
| Alternative flow | N/A |
| Exception flow | Not fill email and password to field or fill wrong email and password. |
| Business rule | In order to log in to the system, user name and password are required existed. |

**UC-03:**

|  |  |
| --- | --- |
| Use Case ID | UC – 02 |
| Use Case Name | Login |
| Actors | All user |
| Description | - This use case allows users who has username and password to login to the system to use system’s functions. Function that users can use must be allowed by admin.  - Allow user to login. |
| Pre-Condition | * User is in Home Screen * User have not logged in yet. * Internet permission. |
| Post-Condition | * Logged account. * Go to Home Screen. |
| Trigger | * Tap on navigator icon. * Tap on “Login”. * User enters email/phonenum and passwords. * Tap on “Login” button. |
| Main flow | |  |  | | --- | --- | | Actors | System | | 1. Click on header - > “Sign in” button |  | |  | 2. Show “Login” screen with: Email, Password. | | 3. Enter email, password. Then, tap on “Sign In” button. |  | |  | 4. Show screen “Home Screen” with data of this account. | |
| Alternative flow | N/A |
| Exception flow | Not fill email and password to field or fill wrong email and password. |
| Business rule | In order to log in to the system, user name and password are required existed. |



## Functional Requirements

### System Functional Overview

*[Provide functionality overview of software system: screen flow, screen descriptions, system user roles, screen authorization, non-screen functions, ERD]*

# 3.2.1.1 Mockup

* Screen: Application Interface

***FIRGURE* *3.2.1.1***

**Description:**

## Non-Functional Requirements

### Reliability

-This is a communication website, so reliability is very important. The information from this website must be real and exactly (phone number, store address)

### Availability

Systems created using the platform should be generally available to users.

### 3.2.3.3 Security

The security is necessary for almost website. Protect user information is important

### Maintainability

The platform should contain interfaces that are extensible to allow new communication protocols to be added with ease.

### Portability

We can connect website by every device have internet

## Other Requirements…

# Software Design Description

*[Provide final software design information follow the template as part II in the Report #4]*

## 1. System Design

### 1.1 System Architecture

*[The content of this section includes the overall diagram which includes the sub-systems, the external systems, and the relationship/connection among them. You need also provide the explanation for each of the diagram components (modules, sub-systems, external systems, etc.)].*

### 1.2 Package Diagram

*[Provide the package diagram for each sub-system. The content of this section includes overall package diagram(s) and the explanation for each package (or namespace)]*

## 2. Database Design

4.2.2.1 Class Diagram

A screenshot of a computer

Description automatically generated

* **Table detail**
* **Table car**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data type** | **Constraint** | **Description** |
| car\_id | Integer | -primary key  -unsigned | Car id |
| price | Float |  | Car price |
| year\_manufacture | Integer |  | Year of car’s manufacture |
| image | Varchar(255) |  | Car image |
| is\_delete | Integer |  | car status info |
| number\_plate | Varchar(255) |  | Car number plate |
| description | Text |  | Car description |
| Address | Varchar(225) |  | Car address |
| custumer\_id | Integer |  | Custumer id |
| brand\_id | Integer |  | Car brand id |
| Model\_id | Integer |  | Car model id |
| car\_type\_id | Integer |  | Car type id |
| car\_status\_id | Integer |  | Car status id |

* **Table customer**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data type** | **Constraint** | **Description** |
| custumer\_id | Integer | -primary key  -unsigned | Custumer id |
| phone | Varchar(50) |  | Custumer phone number |
| email | Varchar(255) |  | Custumer email address |
| name\_display | Varchar(50) |  | Custumer name display on system |
| full\_name | Varchar(50) |  | Custumer full name |
| birthday | Varchar(255) |  | Custumer birthday |
| password | Varchar(50) |  | Custumer account password |
| role\_name | String |  | Custumer role |
| id\_number | Varchar(255) |  | Custumer national id |
| id\_fontside | Varchar(255) |  | Custumer national id fontside |
| id\_backside | Varchar(255) |  | Custumer national id backside |
| Verify\_flg | Integer |  | Verify status |

* **Table booking**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data type** | **Constraint** | **Description** |
| booking\_id | Varchar(20) | -primary key  -unsigned | Booking id |
| start\_date | Varchar(20) |  | Booking start date |
| start\_time | Varchar(20) |  | Booking start time |
| end\_date | Varchar(20) |  | Booking end date |
| end\_time | Varchar(20) |  | Booking end time |
| Total | Integer |  | Booking total |
| Complete\_flg | Integer |  | Booking status |
| status\_id | Intger |  | Booking status id |
| create\_by | Integer |  | Booking create by |
| car\_id | Integer |  | Car id |

* **Table feedback**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data type** | **Constraint** | **Description** |
| feedback\_id | Integer | -primary key  -unsigned | Feedback id |
| rate | Integer |  | Feedback star rate |
| comment | Text |  | Feedback content |
| create\_at | Varchar(20) |  | Feedback create at |
| create\_by | Integer |  | Feedback create by |
| car\_id | integer |  | Car id |

* **Table favourite\_car**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data type** | **Constraint** | **Description** |
| favourite\_car\_id | integer | -primary key  -unsigned | Favourite car id |
| custumer\_id | Integer |  | Custumer id |
| car\_id | Integer |  | Car id |

* **Table address**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data type** | **Constraint** | **Description** |
| address\_id | integer | -primary key  -unsigned | Address id |
| address\_name | Varchar(255) |  | Address name |
| customer\_id | integer |  | Custumer id |

* **Table booking\_status**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data type** | **Constraint** | **Description** |
| booking\_status\_id | Integer | -primary key  -unsigned | Booking status id |
| Booking\_status\_name | Varchar (50) |  | Booking status name |

* **Table car\_status**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data type** | **Constraint** | **Description** |
| car\_status\_id | Integer | -primary key  -unsigned | Car status id |
| car\_status\_name | Varchar (50) |  | Car status name |

* **Table brand**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data type** | **Constraint** | **Description** |
| Brand\_id | Integer | -primary key  -unsigned | Brand id |
| Brand\_name | Varchar (255) |  | Brand name |

* **Table model**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data type** | **Constraint** | **Description** |
| model\_id | Integer | -primary key  -unsigned | Model id |
| model\_name | Varchar(255) |  | Model name |
| brand\_id | Integer |  | Brand id |

* **Table vehicle\_registration**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data type** | **Constraint** | **Description** |
| vehicle\_registration\_id | Integer | -primary key  -unsigned | Vehicle registration id |
| vehicle\_registration\_image | Varchar(255) |  | Vehicle registration image |
| vehicle\_inspection\_image | Varchar(255) |  | Vehicle inspection image |
| car\_id | Integer |  | Car id |

* **Table car\_type**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data type** | **Constraint** | **Description** |
| car\_type\_id | Integer | -primary key  -unsigned | Car type id |
| car\_type\_name | Varchar(255) |  | Car type name |
| number\_seats | Integer |  | Car number seats |

*[Provide the files description, database table relationship & table descriptions]*

## 3. Detailed Design

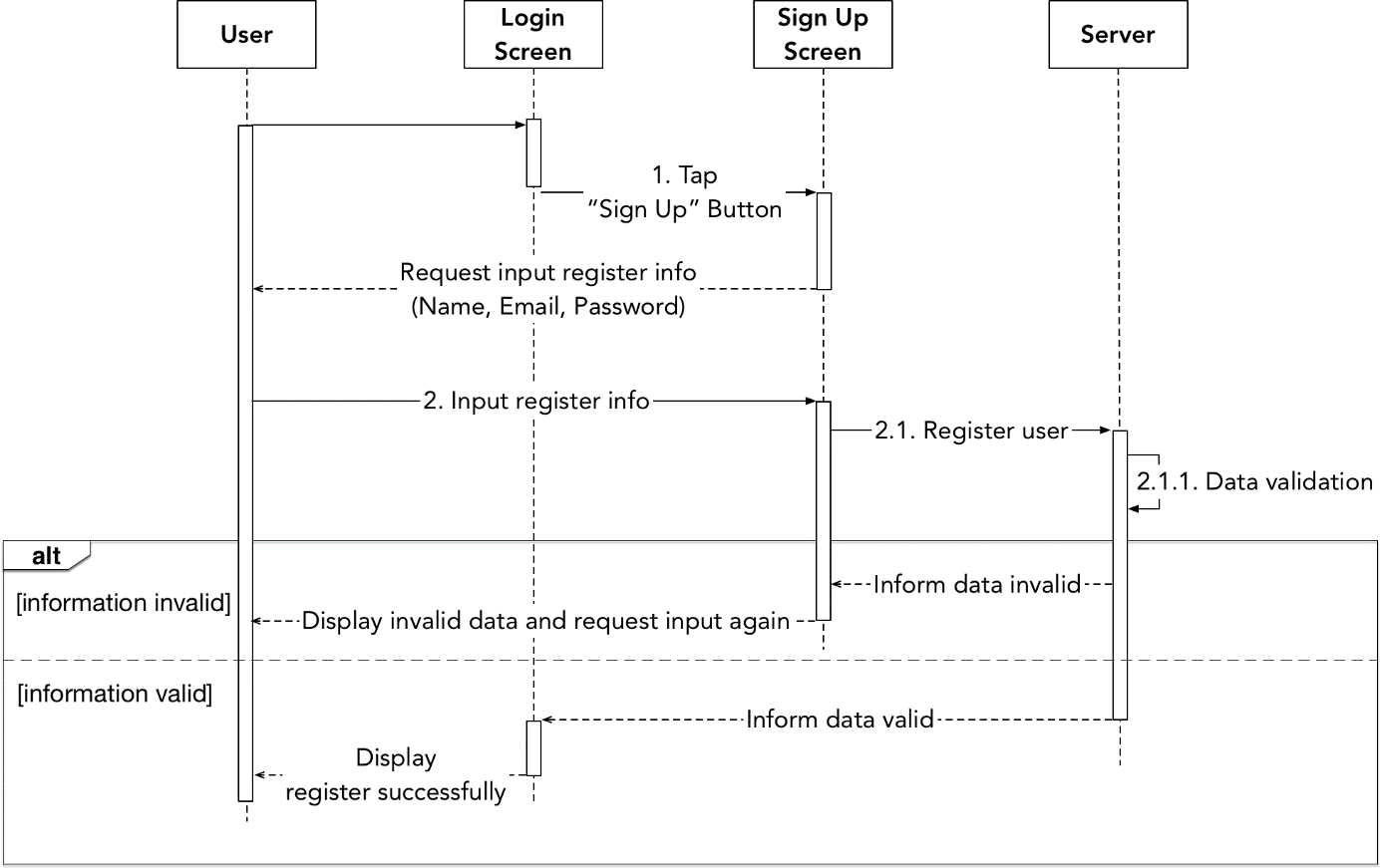
### 3.1 <Sequence Diagram>

*[Provide the detailed design for the feature <Feature Name1>. It includes Class Diagram, Class Specifications, and Sequence Diagram(s);* ***For the features/functions with the same structure of class & sequence diagrams, you need to provide the diagrams once for one feature/function and refer to those diagrams from other features/functions****]*

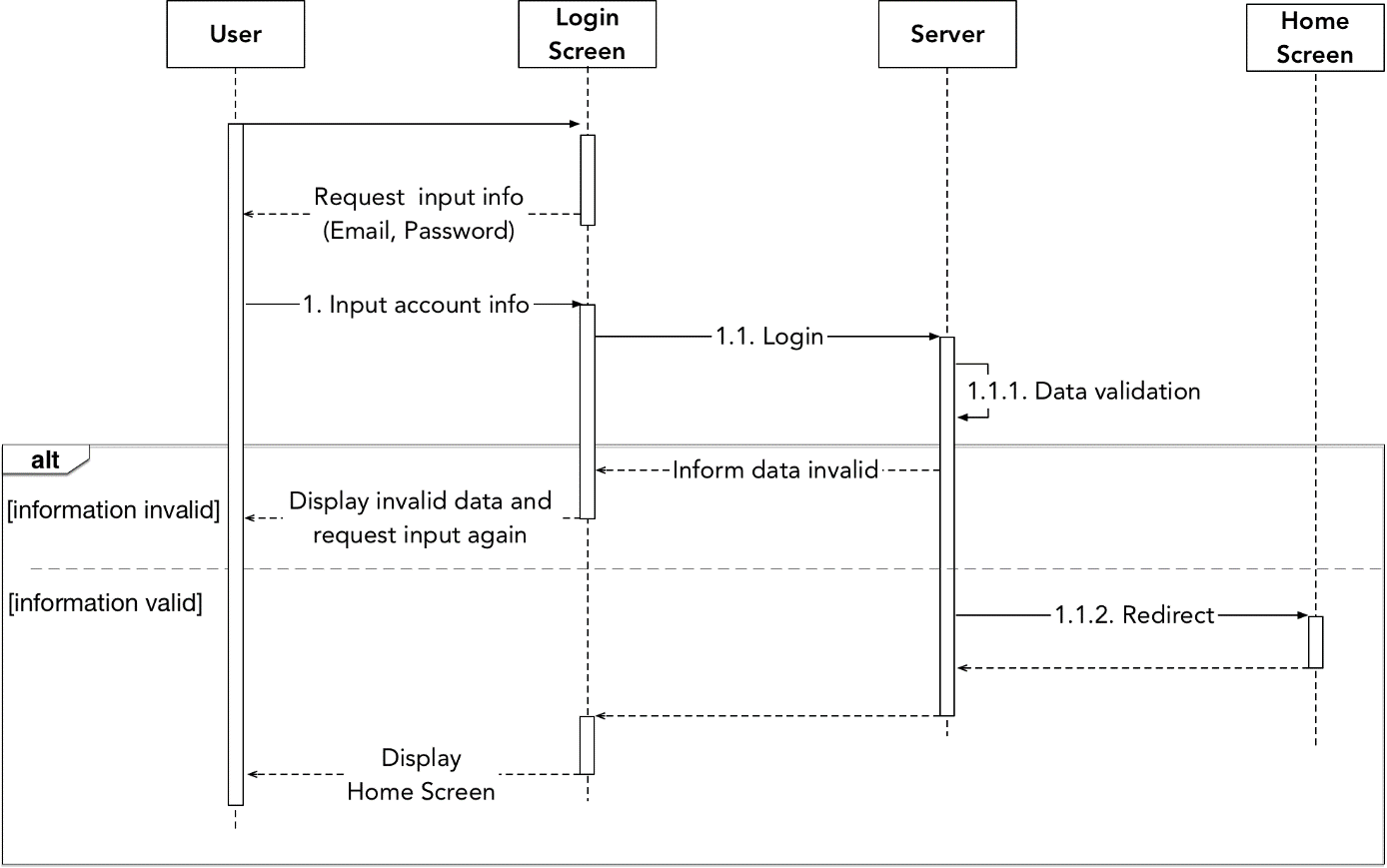
#### 3.1.1 Class Diagram

*[This part presents the class diagram for the relevant feature]*

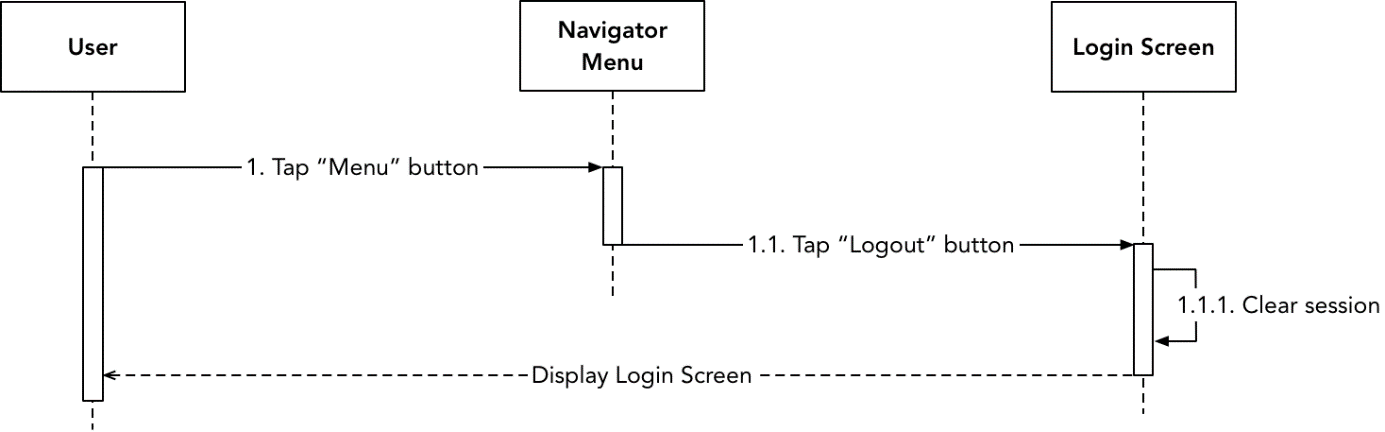
***4.3.1.1 <Sequence diagram for Use case “Register”>***

**

***4.3.1.2 <Use case “Login”>***



***4.3.1.3 < Sequence diagram for Use case “Logout”>***

******

***4.3.1.3 < Sequence diagram for Use case “Forgot password”>***

***4.3.1.3 < Sequence diagram for Use case “Display”>***

*A diagram of a car

Description automatically generated*

***4.3.1.3 < Sequence diagram for Use case “Searching”>***

***A diagram of a model

Description automatically generated***

### 3.2 <Feature/Function Name2>

# Software Testing Documentation

*[Provide final software testing information follow the template as part II in the Report #5]*

## 1. Scope of Testing

*[Describe the scopes of the test. Those include the target-of-test’s features, functions, and non-functional requirements that will or will not be tested.*

*Describe the stages/levels of testing that would be applied to your project - Unit, Integration, or System test. Each includes the in-charge, inputs/time, focuses, acceptance criteria.*

*List any constraints or assumptions made during the development of this document that may impact the design, development or implementation of testing]*

## 2. Test Strategy

*[List out and describe all testing types (you can refer the test types listed below or any other test types to selected the suitable ones for the project; for each selected test types you need to provide the following information: test objective, technique, completion criteria, etc.), test levels that those test types would be performed, & the details of test supporting tools would be used in the project]*

### 2.1 Testing Types

*[List out and describe here the testing types which you would apply in your project. You need to mention following information for each type of testing: objective, technique, completion criteria]*

### 2.2 Test Levels

*<List out and describe here the testing levels which you would execute in your project. Besides, clearly state the test types which are performed in each test level that you plan for this project>*

### 2.3 Supporting Tools

*<List of the test supporting tools which will be employed for this project>*

## 3. Test Plan

### 3.1 Human Resources

*[List and provide the details on roles and responsibilities of the project members who would involve in testing works]*

### 3.2 Test Environment

*[List and provide the details about the tools (software, hardware, infrastructure) which the project would use for testing]*

### 3.3 Test Milestones

*[Separate test milestones, which should be identified to communicate project status accomplishments]*

## 4. Test Cases

*[Prepare the details on the test cases following the provided template*

* *Unit Test Cases: Report5\_Unit Test.xls*
* *Other Test Cases (IT, ST, AT): Report5\_Test Report.xls]*

## 5. Test Reports

*[Provide the test result, statistics and the relevant test analysis for your testing in the project]*

# Release Package & User Guides

*[Provide final software testing information follow the template as part II in the Report #6]*

## 1. Deliverable Package

*[The section will list all source programs, scripts, documents with version number in this release. You can see the example following table for reference, can customize or delete if not using belong to each project characteristics]*

| **No.** | **Deliverable Item** | **Description** |
| --- | --- | --- |
| 1 | Schedule/Task Tracking |  |
| 2 | Project Backlog |  |
| 3 | Source Codes |  |
| 4 | Database Script(s) |  |
| 5 | Final Report Document |  |
| 6 | Test Cases Document |  |
| 7 | Defects List |  |
| 8 | Issues List |  |
| 9 | Slide |  |

## 2. Installation Guides

### 2.1 System Requirements

*[Define any system requirements necessary to support the application, including the software and relevant configurations]*

### 2.2 Installation Instruction

*[Includes installation instructions and configuration guidelines]*

## 3. User Manual

### 3.1 Overview

*[Descript the overview of the application and if could, insert the features workflow to help user has the overview of all the features in this application]*

### 3.2 Workflow 1

*[Describe the purpose of this workflow, draw workflow diagram and other relevant diagrams]*

*[Describe the detailed guides for the workflow by providing the brief description, step by step guides (attached with user interface) of how to use that function]*

### 3.3 Workflow 2