|  |
| --- |
| **FPT- aptech computer education** |
| eProject Report |
| Online Bus Ticket Reseveration System |
|  |
| |  |  | | --- | --- | | **404 Error Team** | | | **Group Member** | LE HOANG MINH – TH2002022 | | **Semester** | 3 | |
|  |

|  |
| --- |
| - Hanoi, November 2021 - |

Contents

1 - Acknowledgements

2 - Introduction

3 - Problem Definition

4 - Requirement and Business flows

5 – Design

6 - System Prototype

7 - Management and project Planning

8 - Screen shots

9 User Guide

# Acknowledgements

First, i would like to thank the teachers who supported me during this project. This is the first project I've done by myself, it's a good opportunity for me to improve my skills.

# Introduction

In this document i will explain and guide people about this project. This project will help people use transportation services more conveniently.

# Problem Definition

## Problem Abstraction

After reading and learning about the needs of SRC, we realized that there was a need for the company to build a website to manage the sale of bus tickets, instead of using the traditional method.

## The Current System

The current system that the SRC company is using is ticket sales at the counter, managing everything manually, it lacks tools to support ticket management and personnel.

The old way of doing things is not strict, leading to confusion, which can cause loss of sales and make it difficult for businesses to scale up or improve service quality.

## The Proposed System

To solve the above problems, we propose a solution is to build a system to manage the effects in the company's bus ticketing activities.

Includes the following sections:

- Vehicle management

- Ticket management

- HRM

## Boundaries of the System

The scope of the system depends on the level of utilization by the company. Since this is a company providing domestic transportation services, the scope of the system is to provide domestic rides that the company manages.

This website and audience are those who have travel needs within the country, regardless of foreigners or natives.

## Development Environment

To operate the system, it is necessary to install an internet network system, computers can connect to the internet.

Configurable computers (minimum):

- CPU: Intel Code i3

- Ram: 8GB

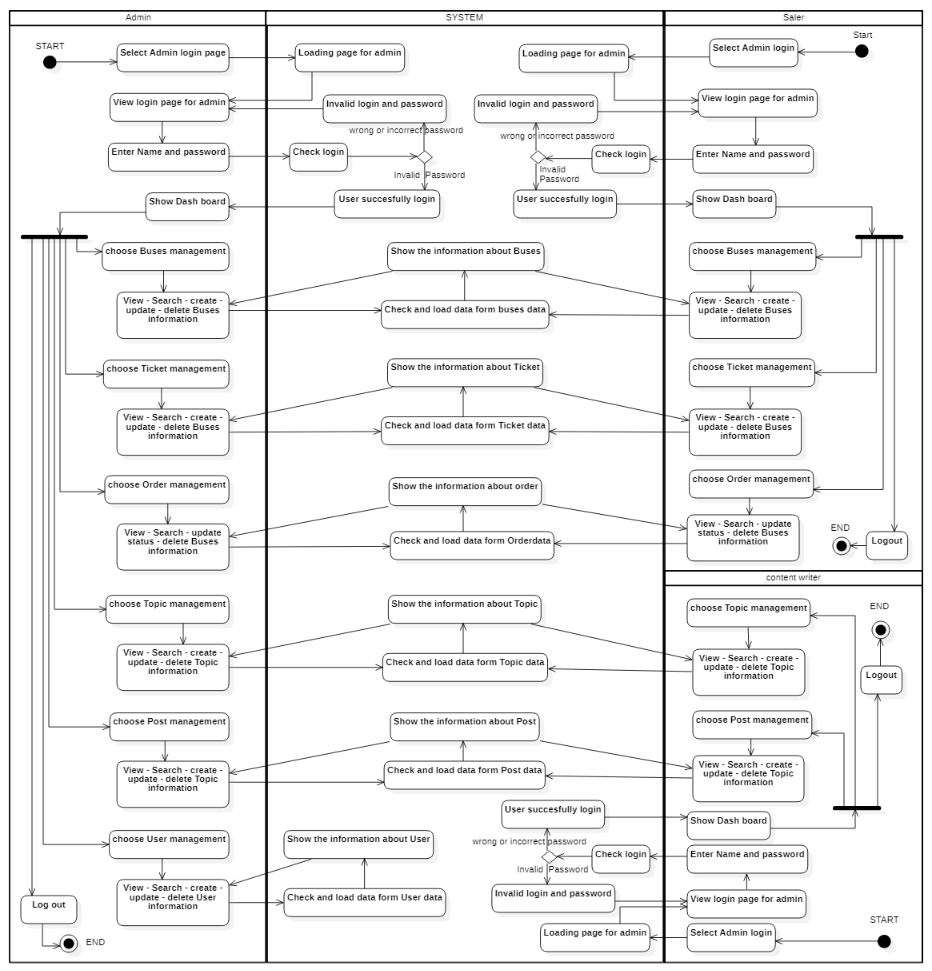
- Operating system: Windows 10

# Requirements and Business Flows

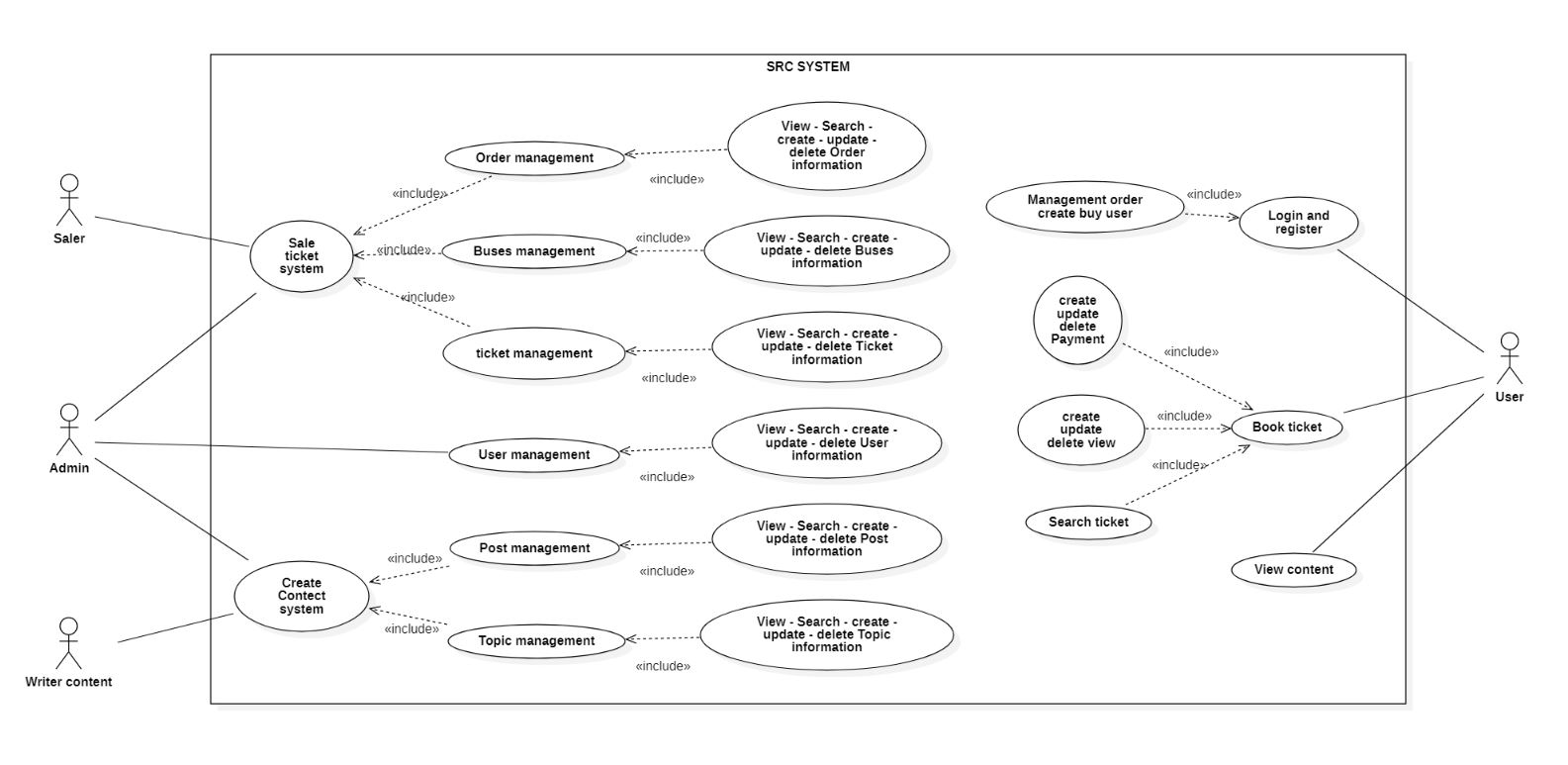
## Customer Requirements

SRC company wants us to build a website that can book and manage tasks in their business model, including: Tickets, garage, human resources.

## Activity Diagram



## Use Case Diagram



## Other Concerns

*<You can list here all other concerns about the business or the requirements if needed>*

# Design

## System Architecture

## Database Design

## Data structure

### *- Category table*

Table 5.1 Category table (category)

|  |  |  |
| --- | --- | --- |
| **Field** | **Data type** | **Must have** |
| ID | int | Y |
| Name | nvarchar(255) | Y |
| slug | varchar(255) | Y |
| parentid | int | Y |
| orders | int | Y |
| metakey | nvarchar(150) | N |
| metadesc | nvarchar(150) | N |
| created\_at | smalldatetime | Y |
| created\_by | int | Y |
| updated\_at | smalldatetime | Y |
| updated\_by | int | Y |
| status | int | Y |

### *Category table is used to record information, list of categories of motorcycles. Each product will be in a category, and a category can have multiple products. Customers will be able to find something like this. The management management function allows the manager to create categories as well as change the information and delete those categories. Categories all have a record of when they were created, when they were last edited, and of course, information about who made the changes. Category information will also include pictures and descriptions of the genre.*

### *- Topic table*

Table 5.2 topic table

|  |  |  |
| --- | --- | --- |
| **Field** | **Data type** | **Must have** |
| ID | int | Y |
| Name | nvarchar(255) | Y |
| slug | varchar(255) | Y |
| parentid | int | Y |
| orders | int | Y |
| metakey | nvarchar(150) | N |
| metadesc | nvarchar(150) | N |
| created\_at | smalldatetime | Y |
| created\_by | int | Y |
| updated\_at | smalldatetime | Y |
| updated\_by | int | Y |
| status | int | Y |

The article topic table is used to record information, list the topics of the article. Each article will be under a topic and a topic can have many articles. Customers will be able to find articles of their favorite genre faster if the database has statistics on topics like this. The author management function allows administrators to create topics as well as change information and delete topics. Topics all have a record of when they were created, when they were last edited, and of course, information showing who made the name change. Topic information will also have pictures and descriptions of the topic.

### *- Post table*

Table 5.3 Post table

|  |  |  |
| --- | --- | --- |
| **Field** | **Data type** | **Must have** |
| ID | int | Y |
| topid | int | Y |
| title | nvarchar(MAX) | Y |
| slug | varchar(255) | Y |
| detail | ntext | Y |
| img | varchar(255) | Y |
| type | varchar(50) | Y |
| metakey | nvarchar(150) | N |
| metadesc | nvarchar(150) | N |
| created\_at | smalldatetime | Y |
| created\_by | int | Y |
| updated\_at | smalldatetime | Y |
| updated\_by | int | Y |
| status | int | Y |

Post table is used to record information, list of articles. Each article will come with information such as title name, details, image. The post will also be dated and who created it similarly to when and who edited it. Status will show if this product is in stock or not, views will show how many times customers clicked to view the post.

### *Ticket table*

Table 5.4 ticket table

|  |  |  |
| --- | --- | --- |
| **Field** | **Data type** | **Must have** |
| ID | int | Y |
| Name | nvarchar(MAX) | Y |
| description | nvarchar(255) | N |
| flightCode | char(50) | Y |
| airline | nvarchar(255) | Y |
| airport | nvarchar(255) | Y |
| ticketType | nvarchar(255) | N |
| img | nvarchar(255) | Y |
| departure\_address | nvarchar(255) | N |
| arrival\_address | nvarchar(255) | Y |
| price | float | y |
| priceSale | float | y |
| Sold | int | y |
| created\_at | smalldatetime | y |
| created\_by | int | y |
| updated\_at | smalldatetime | y |
| updated\_by | int | y |
| status | int | y |

### *Order table*

Table 5.5 Order table

|  |  |  |
| --- | --- | --- |
| **Field** | **Data type** | **Must have** |
| ID | int | Y |
| Update\_At | smalldatetime | Y |
| Update\_By | int | N |
| DeliveryName | nvarchar(100) | N |
| DeliveryPhone | varchar(255) | N |
| UserID | int | Y |

Orders table is used to record information and list of orders. Each order will come with the information of the person placing the order. The order will also be dated and the person who created it is the same as the time and who edited it. The delivery status will show the shipping status of the order and the order status will indicate the order is completed, canceled or confirmed by the admin.

### *Order details table*

Table 5.6 Order details table

|  |  |  |
| --- | --- | --- |
| **Field** | **Data type** | **Must have** |
| OrderID | int | Y |
| ProductID | int | Y |
| Number | int | Y |
| Price | decimal(18,2) | Y |

Order details table will record the information of the order including which products with quantity and unit price when ordering them. Any change to the order details will result in the time and person who made the edit on the order sheet will be updated.

### *User table*

Table 5.7 User table

|  |  |  |
| --- | --- | --- |
| **Field** | **Data type** | **Must have** |
| ID | int | Y |
| UserName | varchar(255) | Y |
| Password | varchar(64) | Y |
| FullName | nvarchar(255) | Y |
| Gender | nvarchar(5) | N |
| Address | nvarchar(250) | Y |
| Phone | nvarchar(250) | Y |
| Email | nvarchar(255) | Y |
| CreatedAt | smalldatetime | Y |
| CreatedBy | int | N |

User table will store information, list of users of the site. Each user will belong to any user group. The user will also be credited with the time and person who created it as well as the time and the person who edited it.

Each user will have an account with its username, password, avatar, user group, and permissions for that group.

- Admin: full system rights, manage all activities on the system including adding new users (this account will belong to the user group without group), changing the profile and password of the account itself, position, add, remove permissions for the group and set random new passwords for users, then the information will be sent to that user's email. It is possible to add new products, product types, articles as well as delete and change their information. Admin also has the right to change information about orders and is the only group that has the right to delete accounts.

- Mod (Employee): only has the right to log in to the system and can change information about products, product types, new posts as well as delete, order, change profile and account password. self. This group will not have delete permissions at all.

- Member: customers will not have the right to access the management page and can only place orders, view their information as well as change their profile and password.

### *Table of permissions*

Table 5.9 Table of permissions

|  |  |  |
| --- | --- | --- |
| **Field** | **Data type** | **Must have** |
| ID | nvarchar(128) | Y |
| Name | nvarchar(250) | Y |

The table permissions will store a list of information system permissions. Permissions will be added to the group user system, the permissions detailing the group user's table will be where this information is stored. The administrator can view the list of system permissions for better management.

### *User group permissions details table*

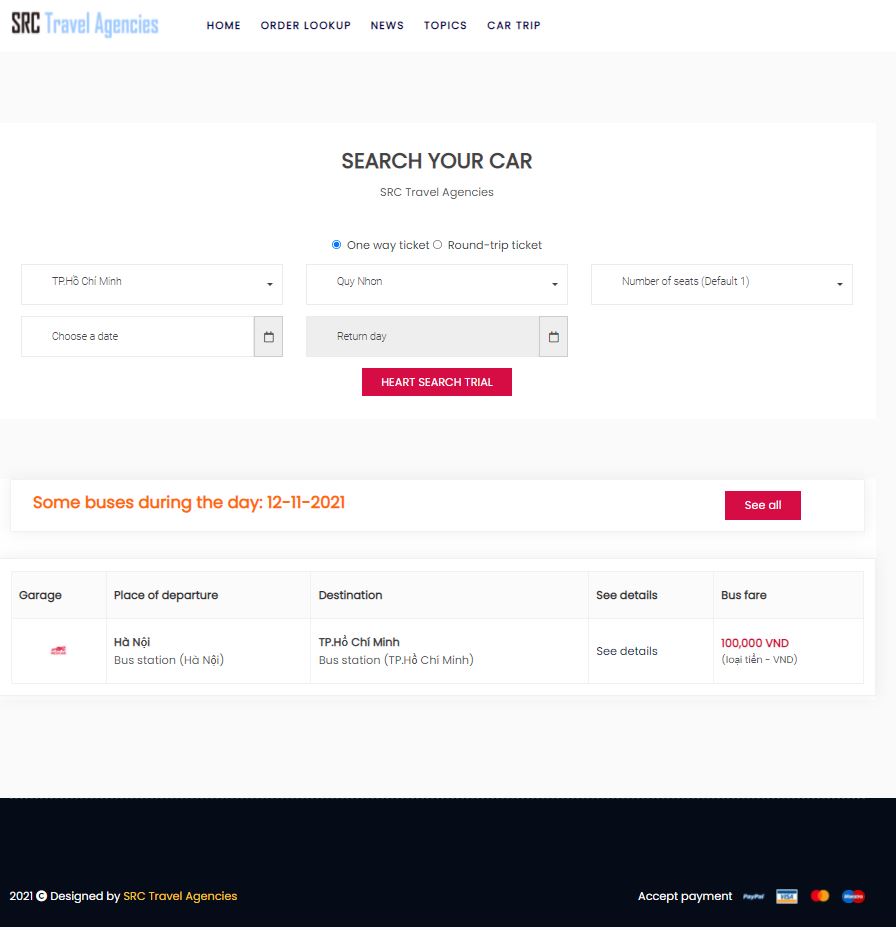
Table 5.10 User group permissions details table

|  |  |  |
| --- | --- | --- |
| **Field** | **Data type** | **Must have** |
| UserGroupID | nvarchar(128) | Y |
| RoleID | nvarchar(128) | Y |

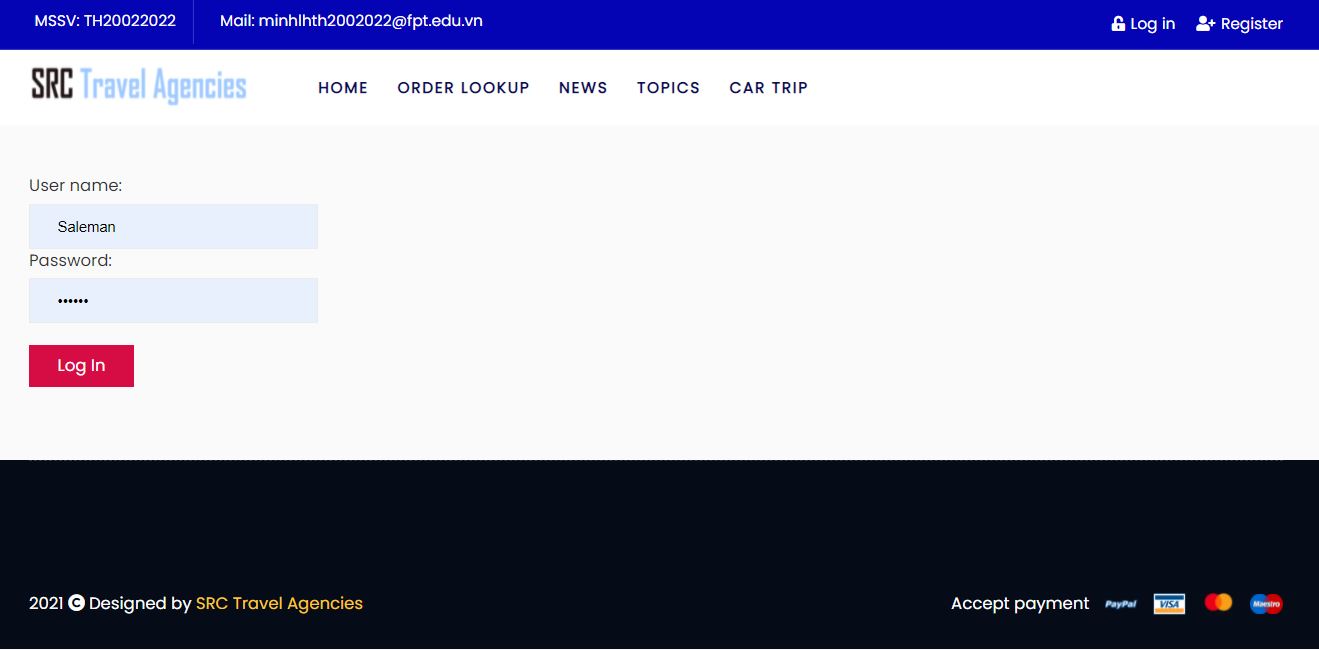
The user group permissions details table will store a detailed list of permissions for each user group. Administrators can change the permissions of user groups in this table, and there will be restrictions on the permissions the user groups can have. For example, the customer group will not be able to have access to the admin page.

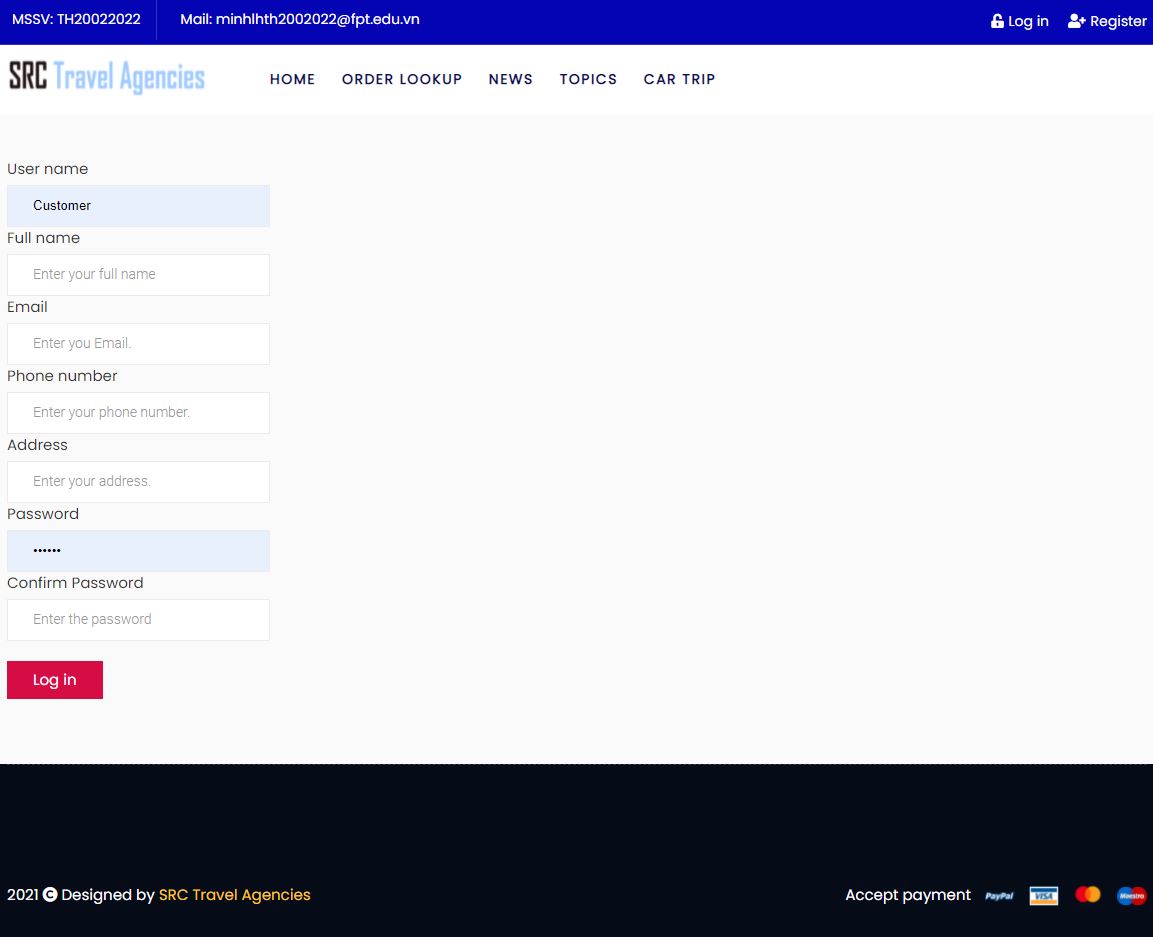
# System Prototype

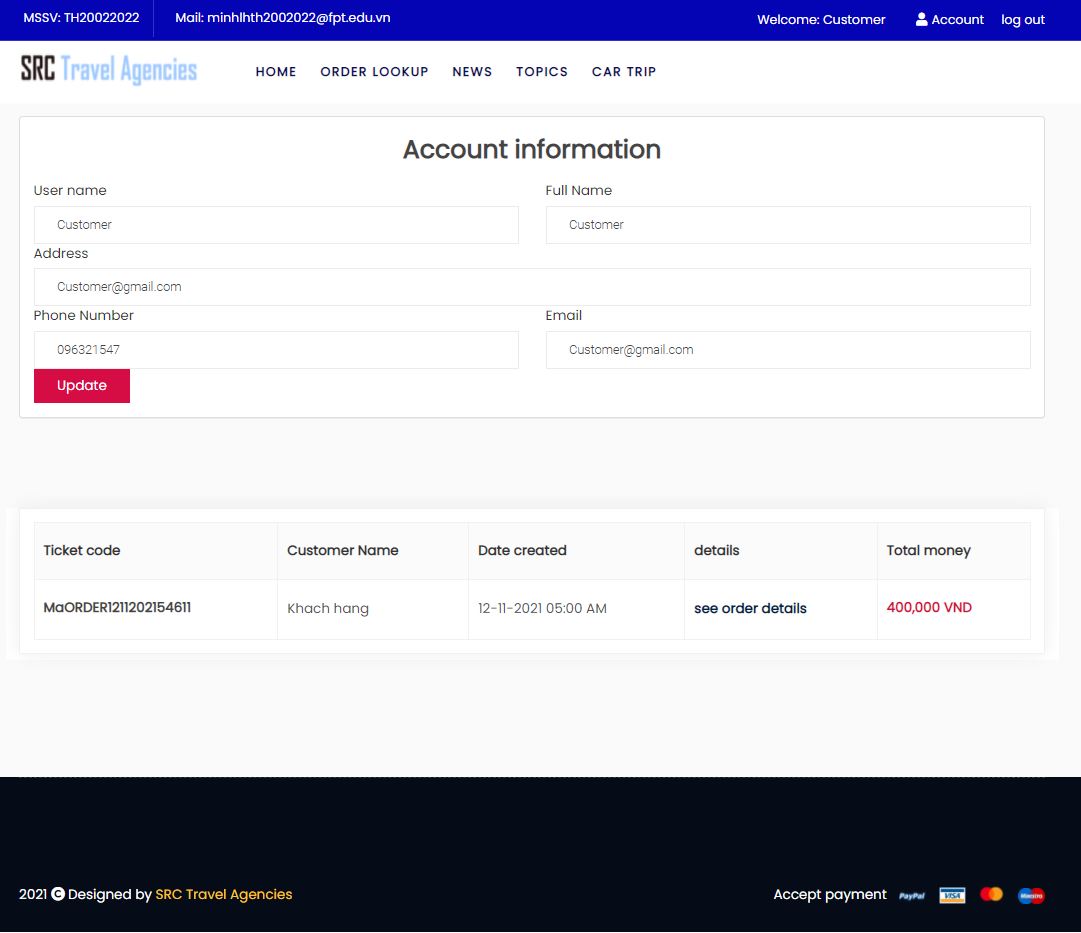
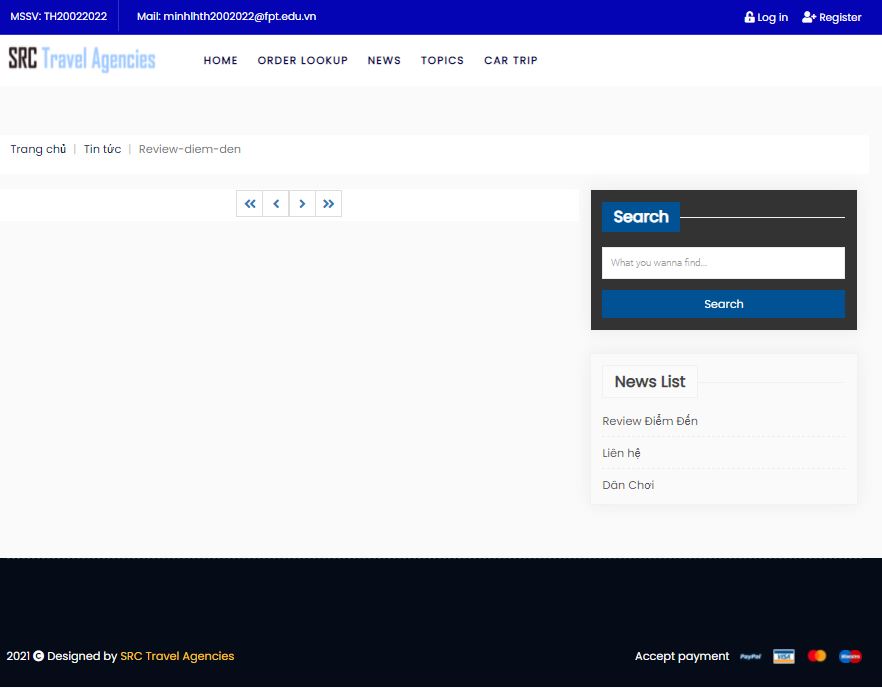
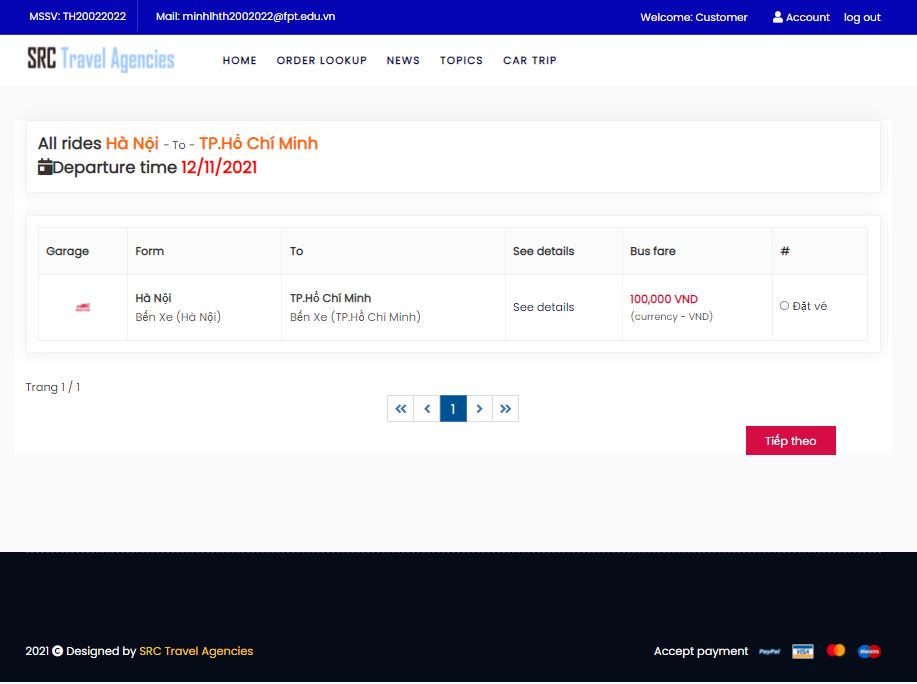
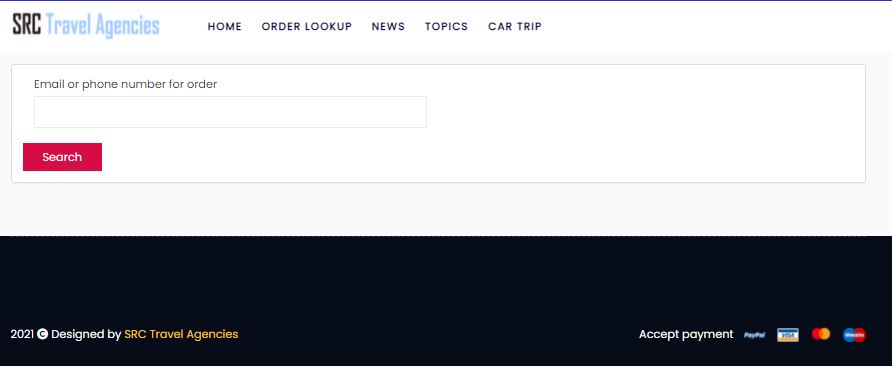
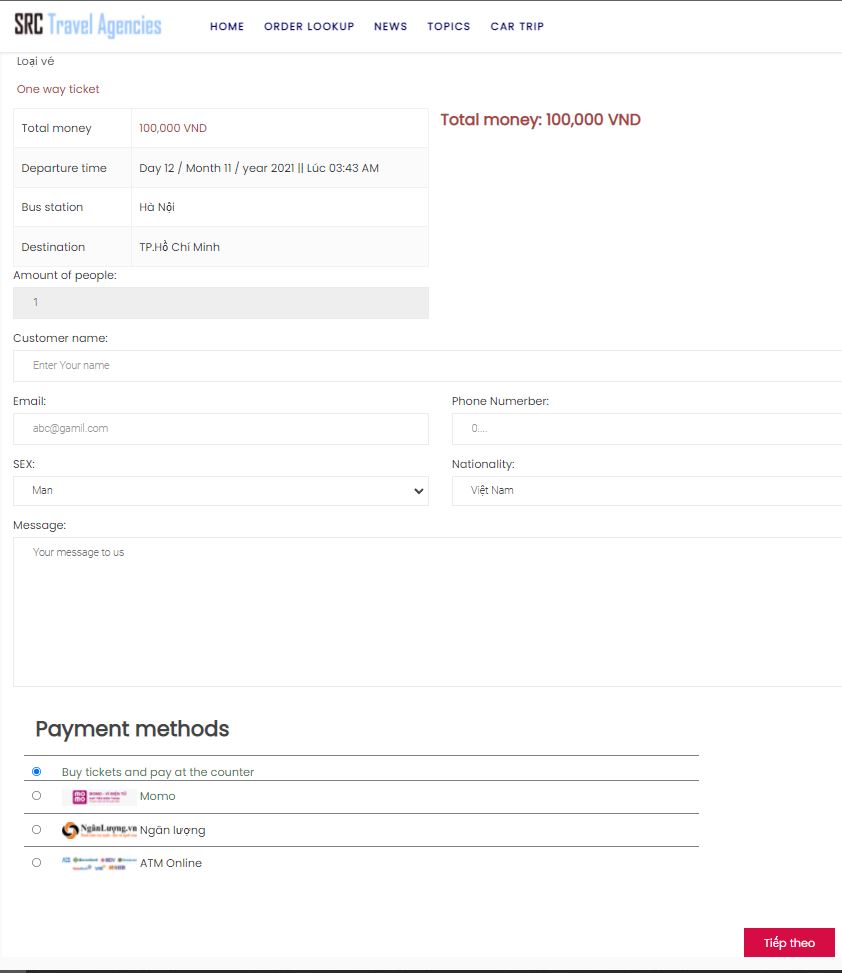
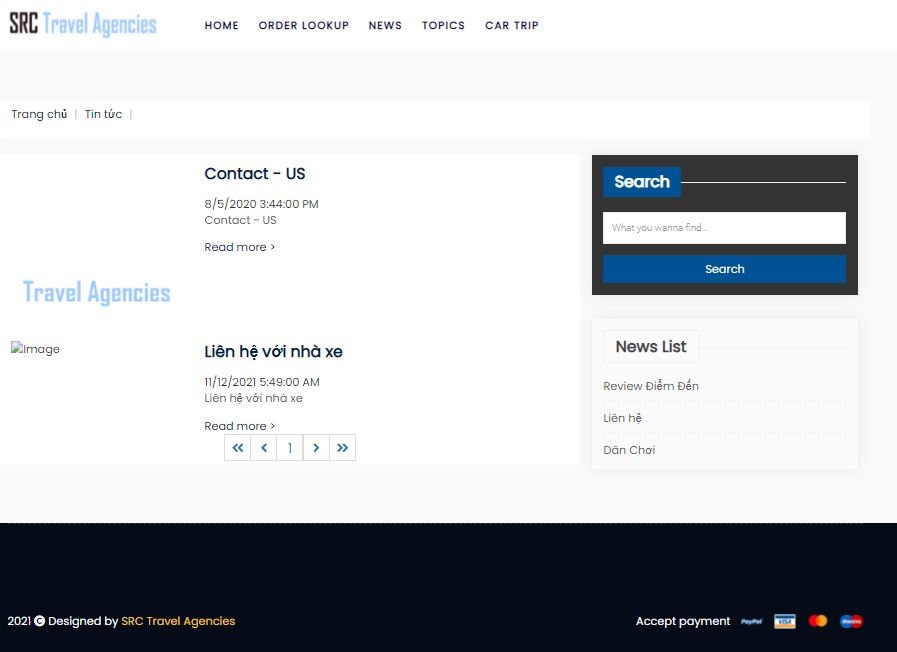
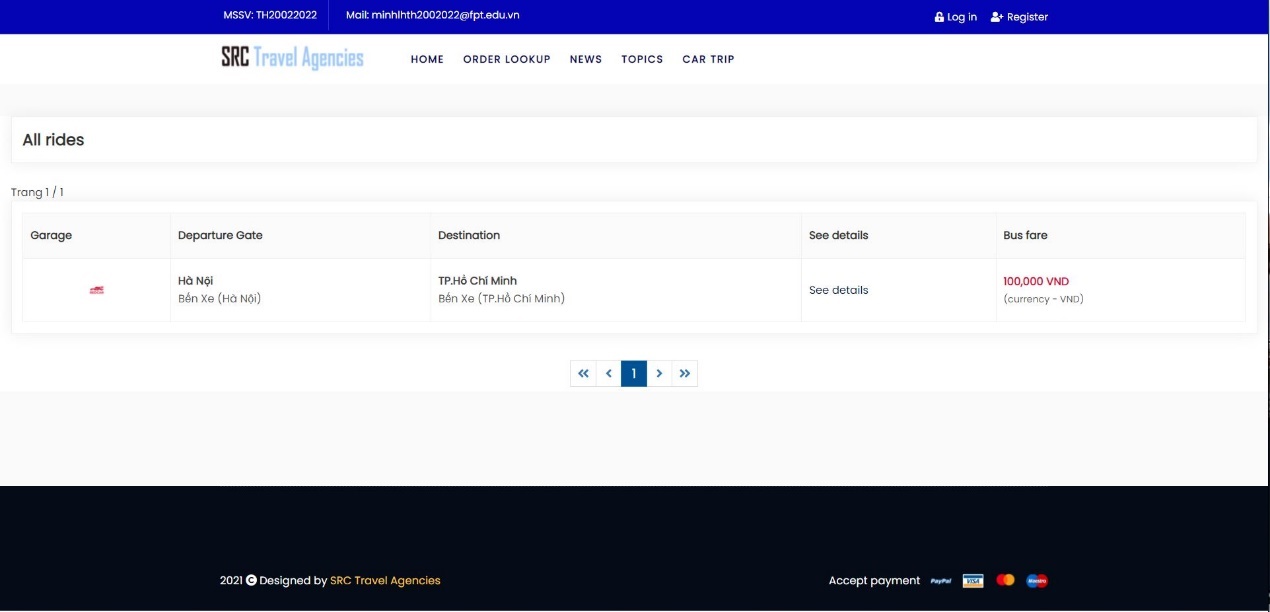
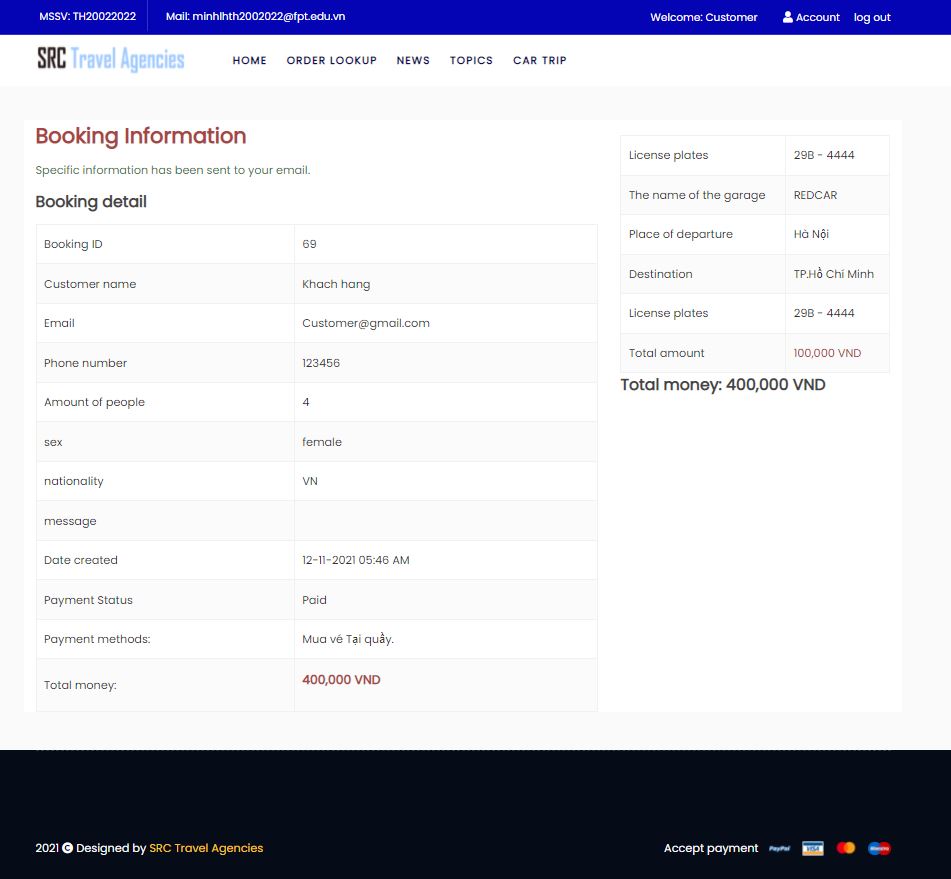
1. Home page



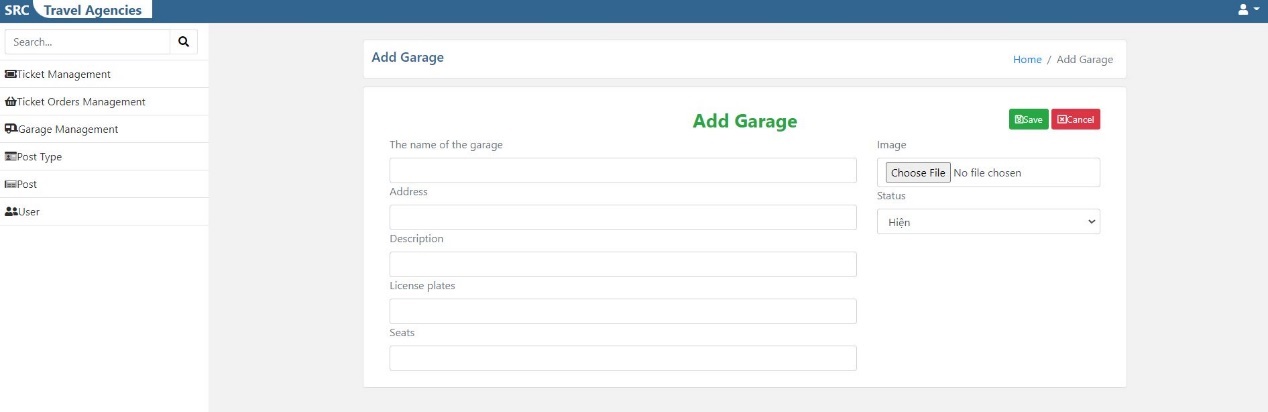
2 . login page

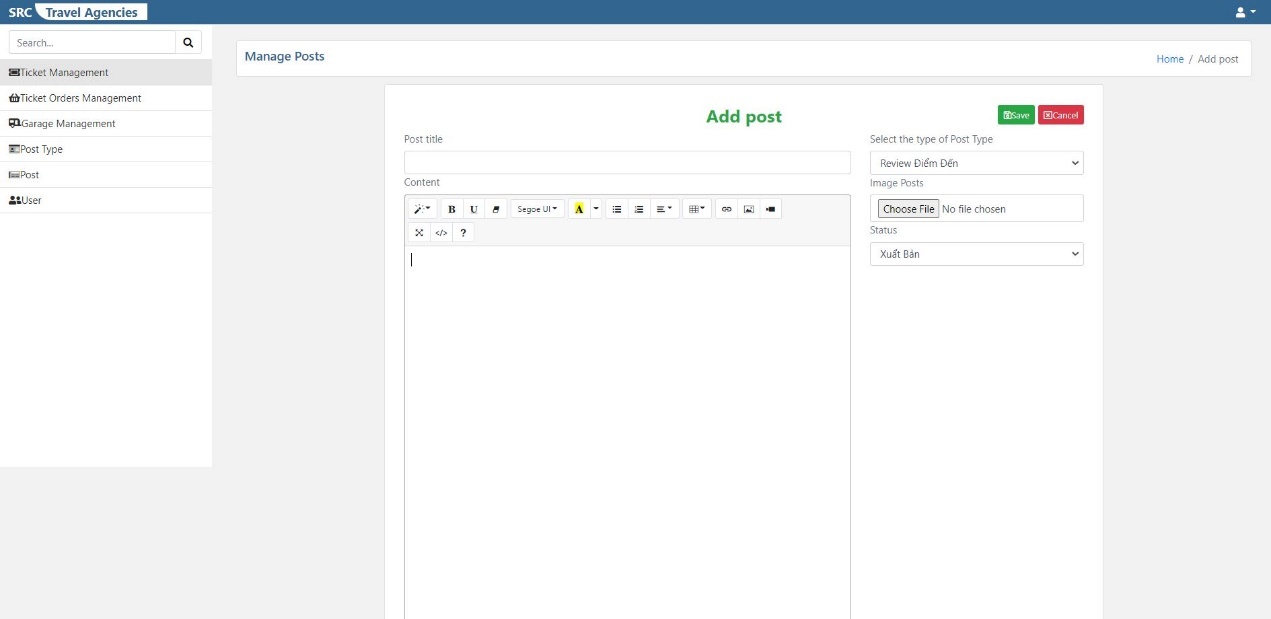


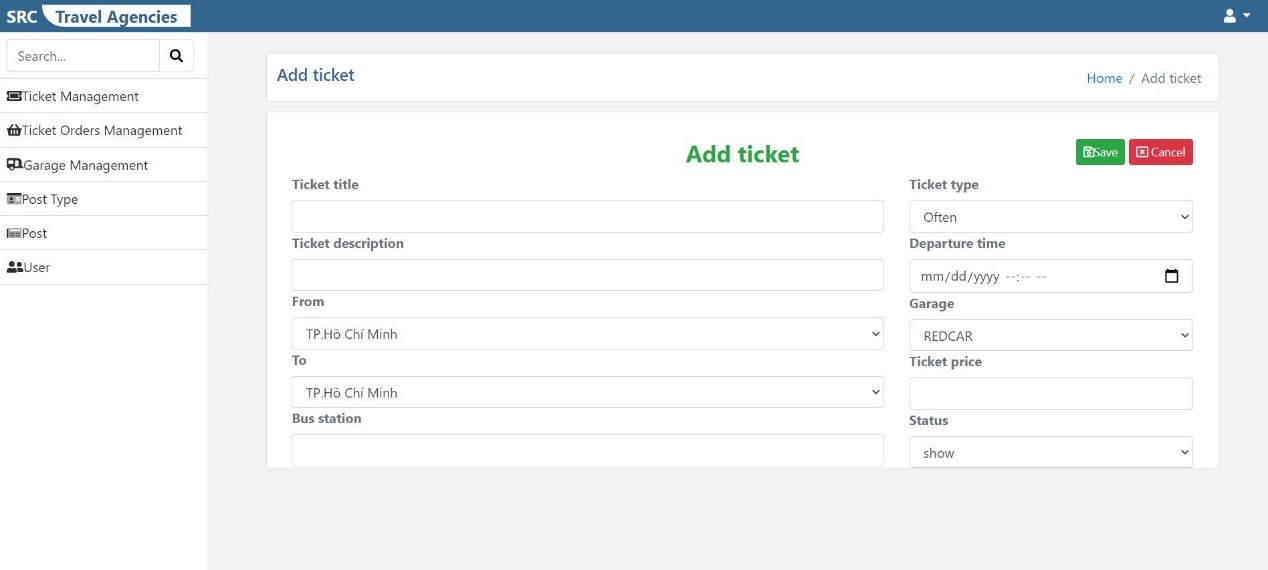


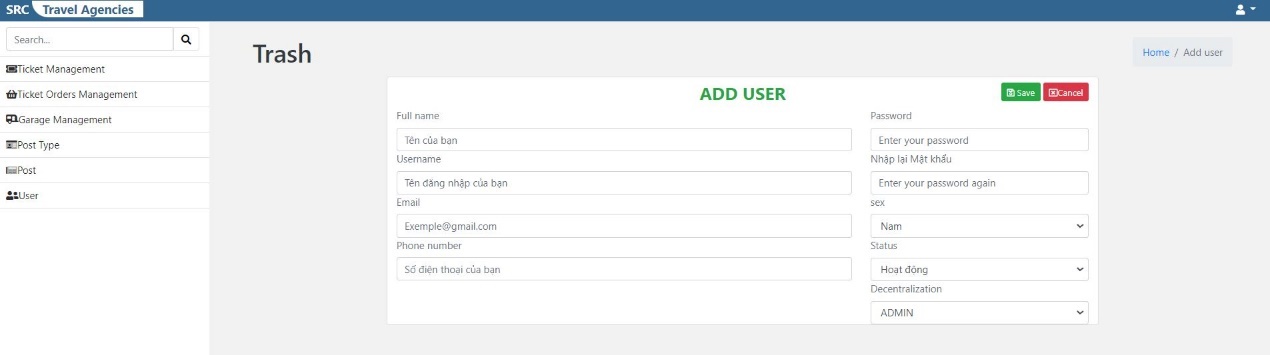


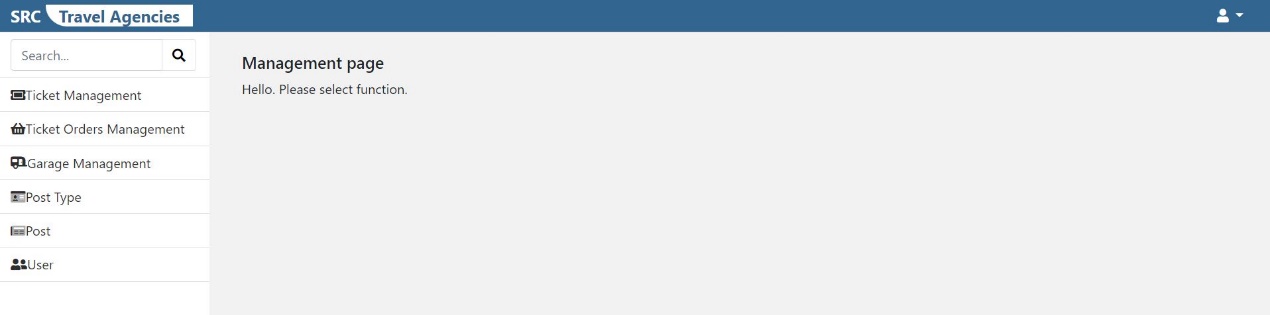
#Admin dashboard

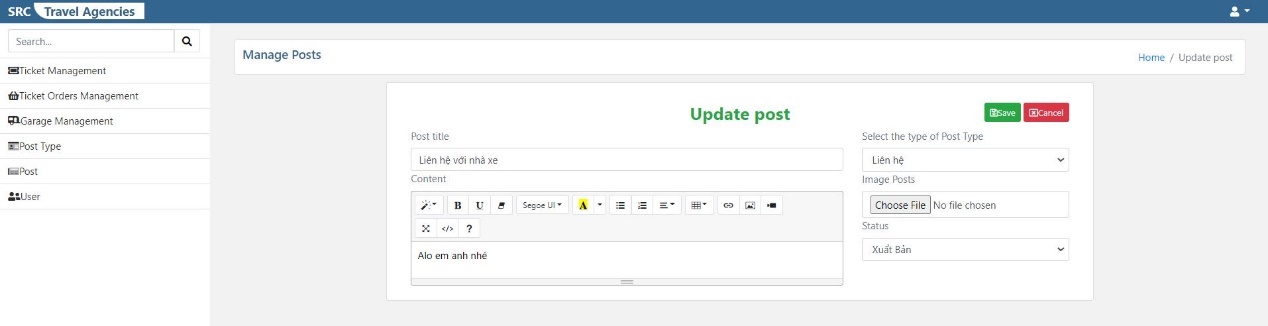


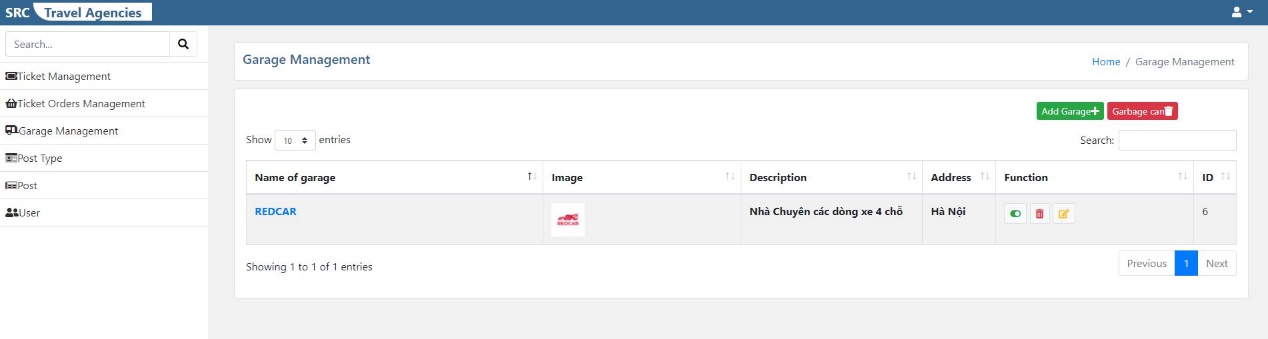


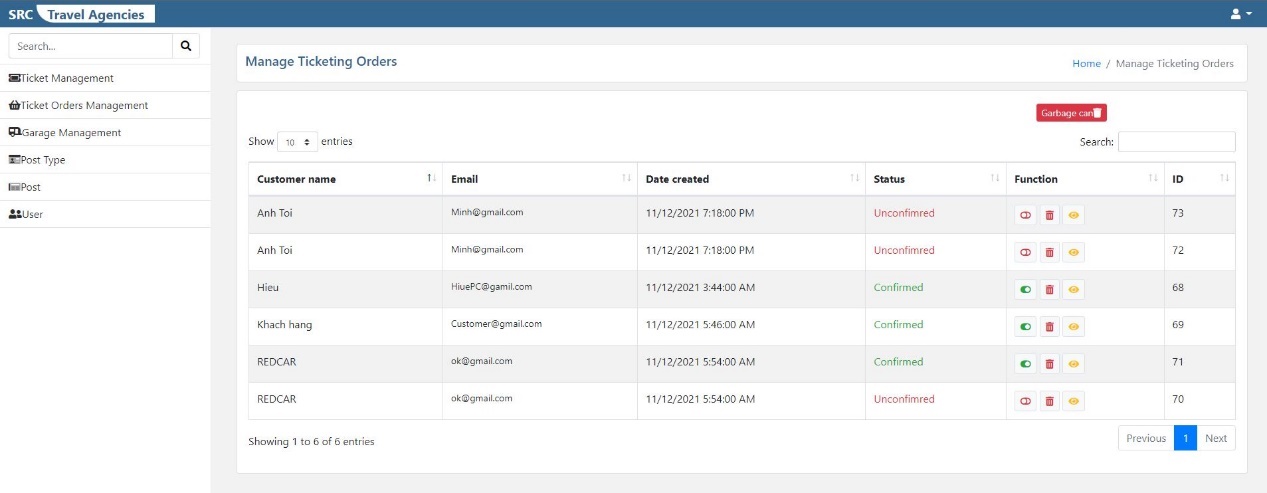
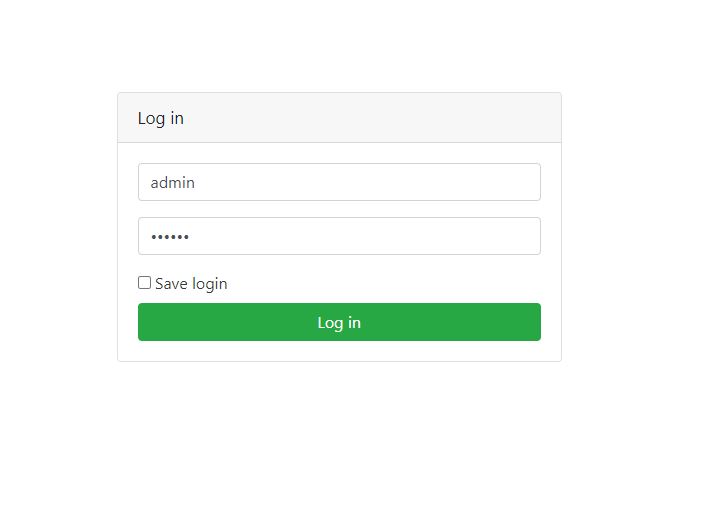
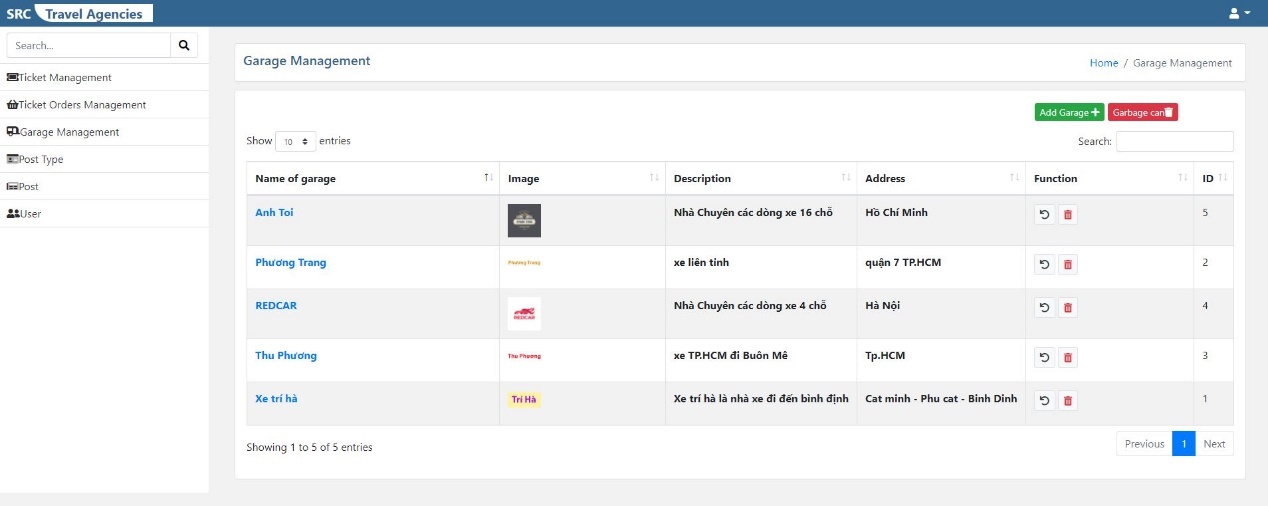


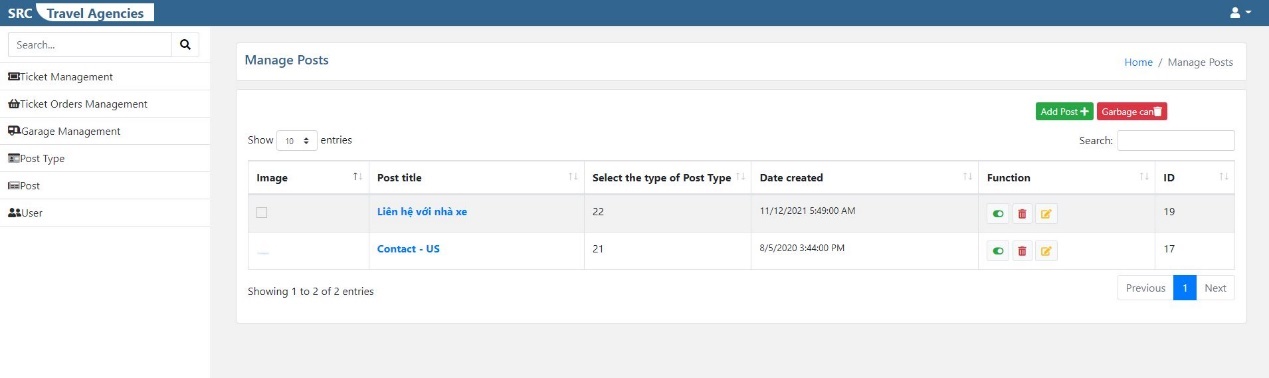


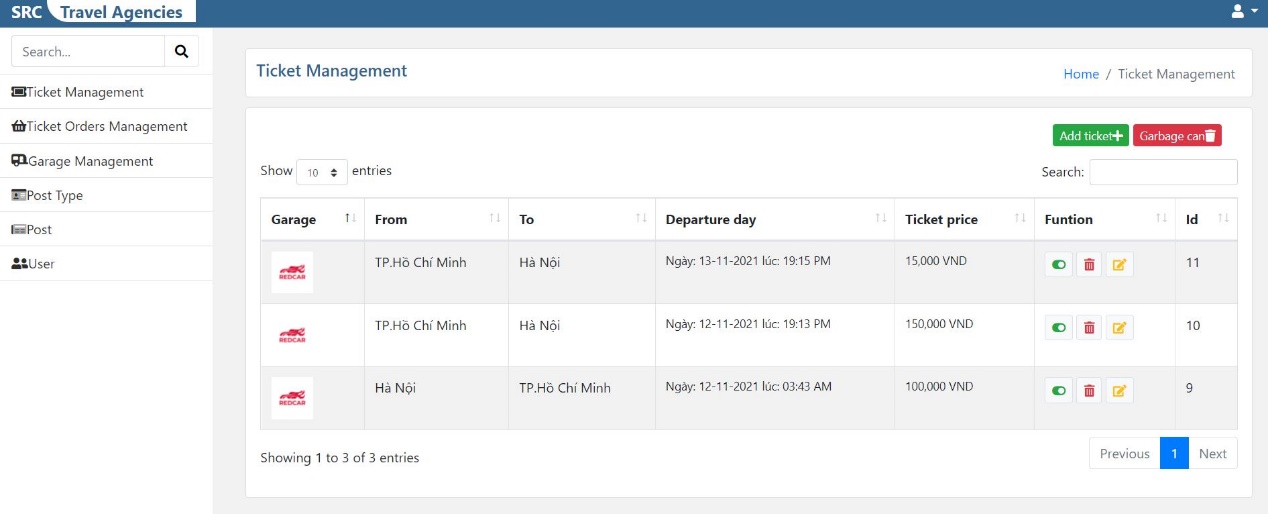


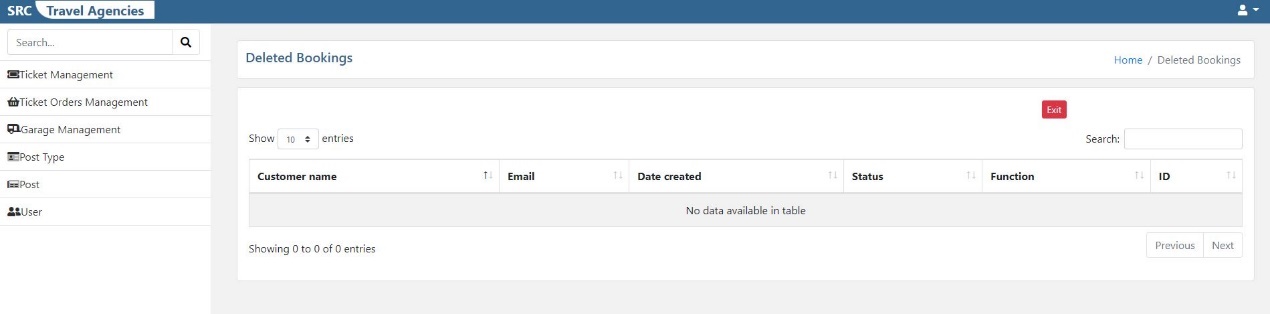


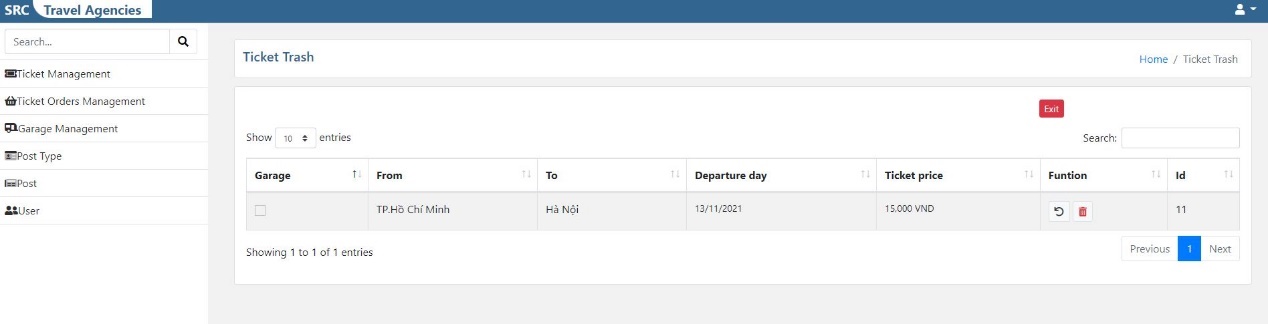


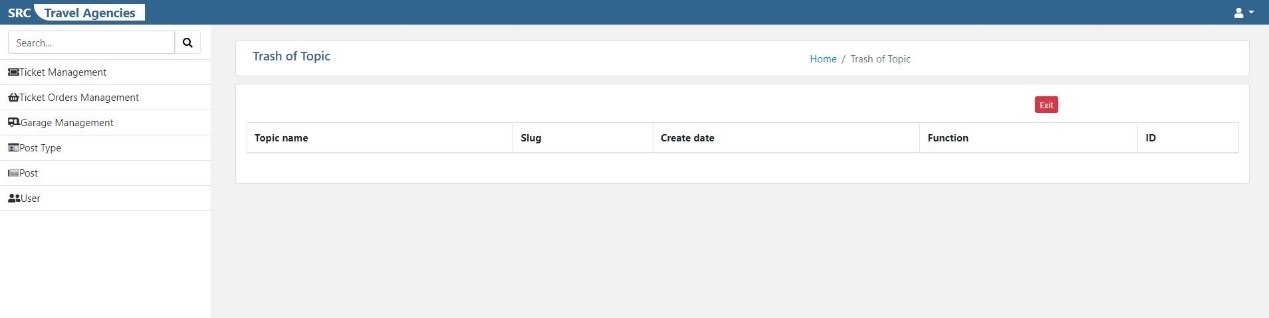


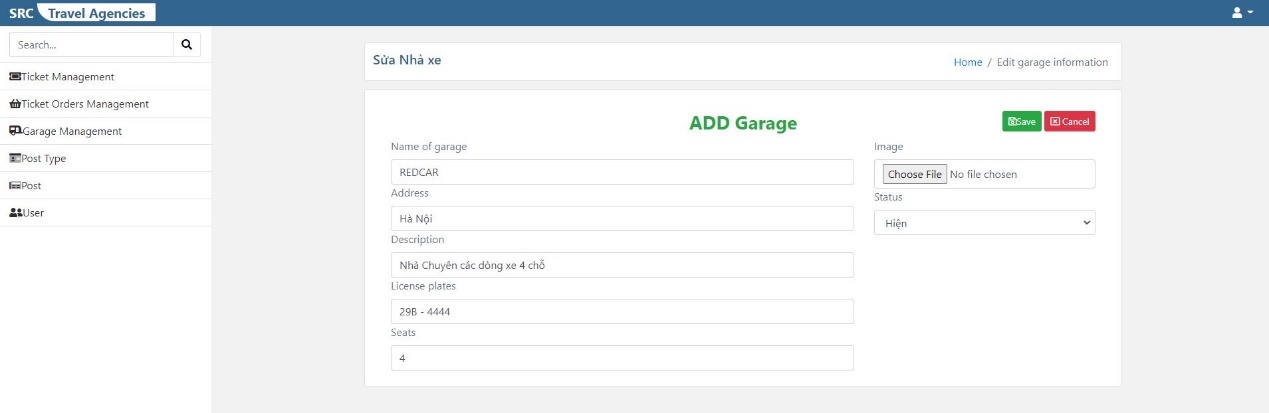


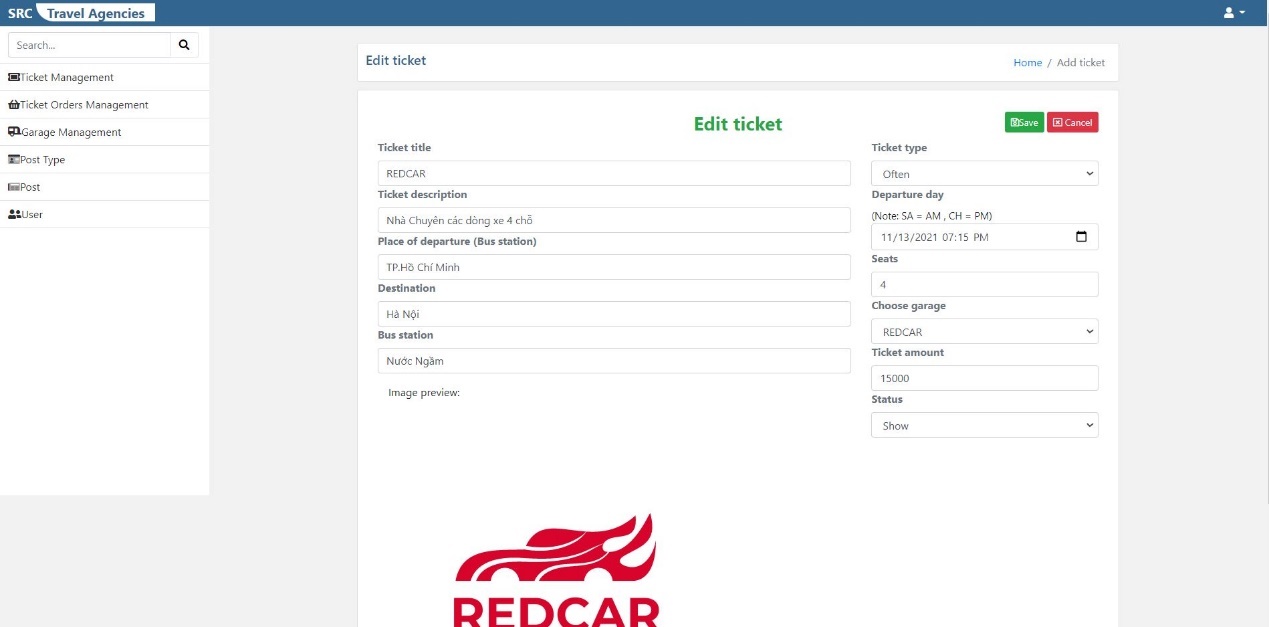


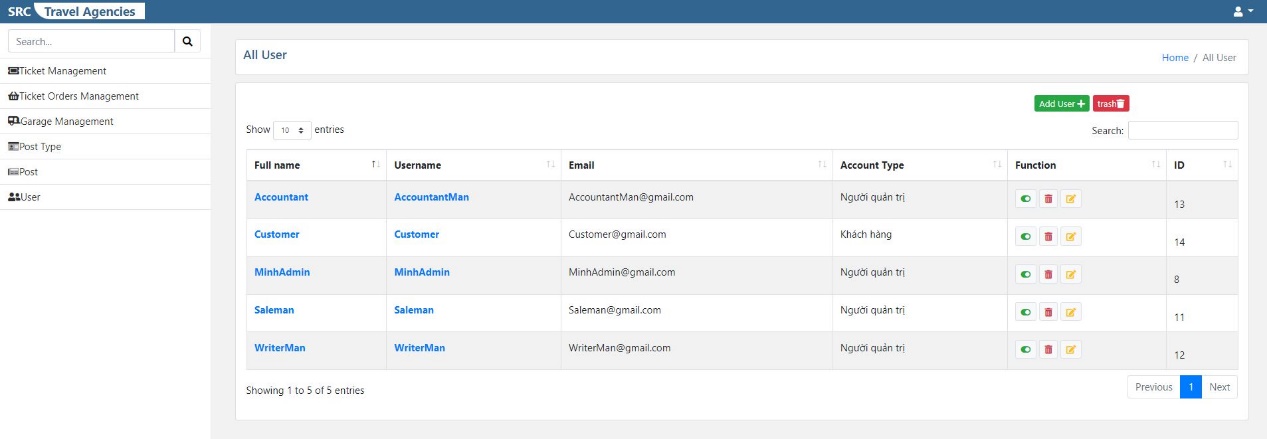












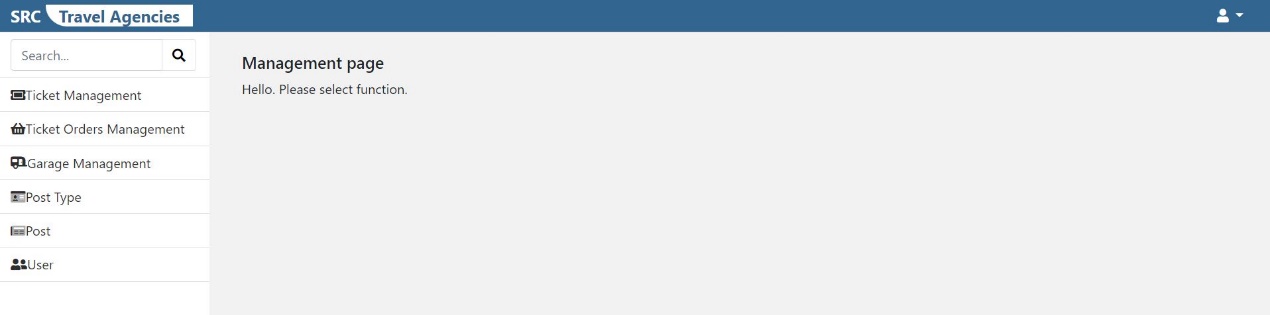
# 7.Management and Project Planning

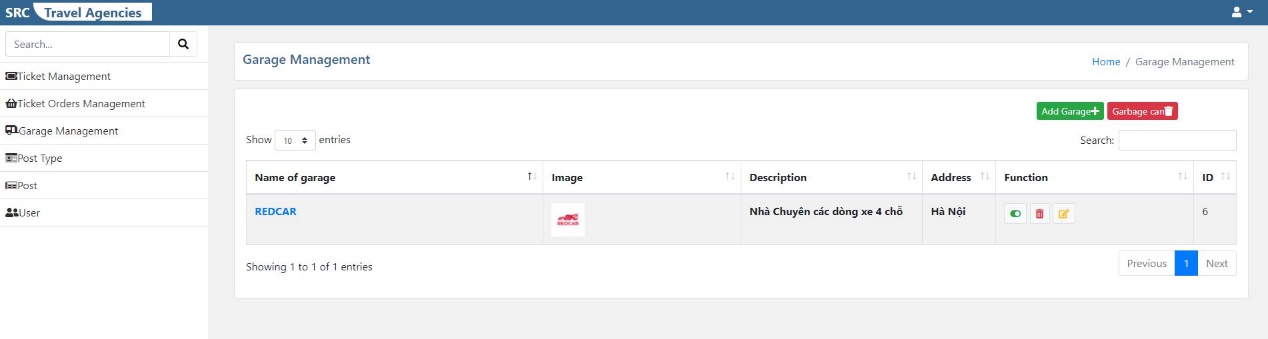
## 7.1Management Approach

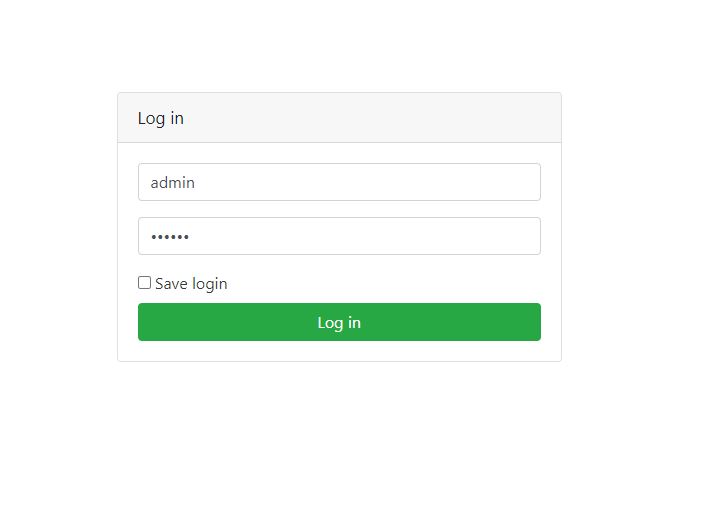
Since this is not a big project and only one person does it, I choose the simplest management method, which is to create the first basic tasks and then execute it as planned.

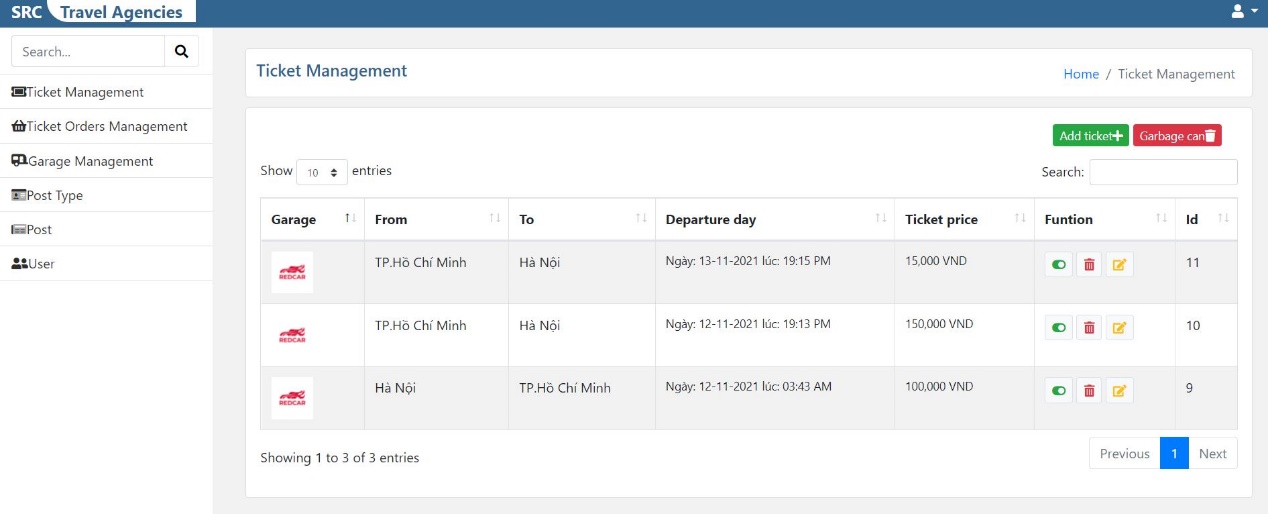
# 8Screenshots

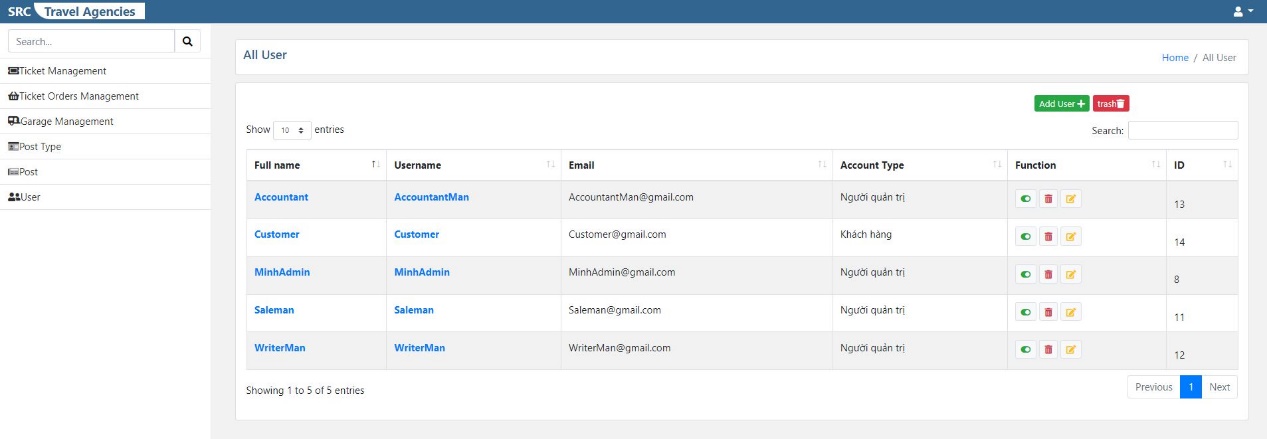
Admin



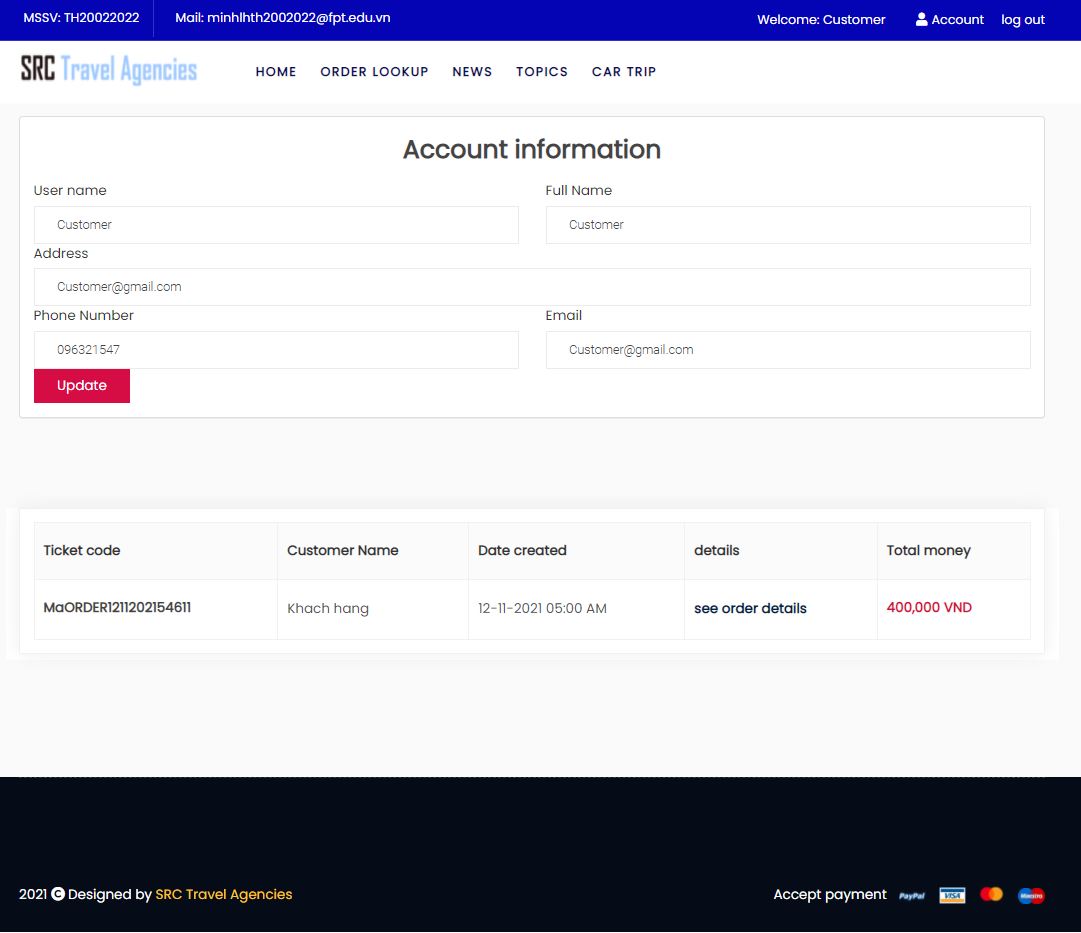
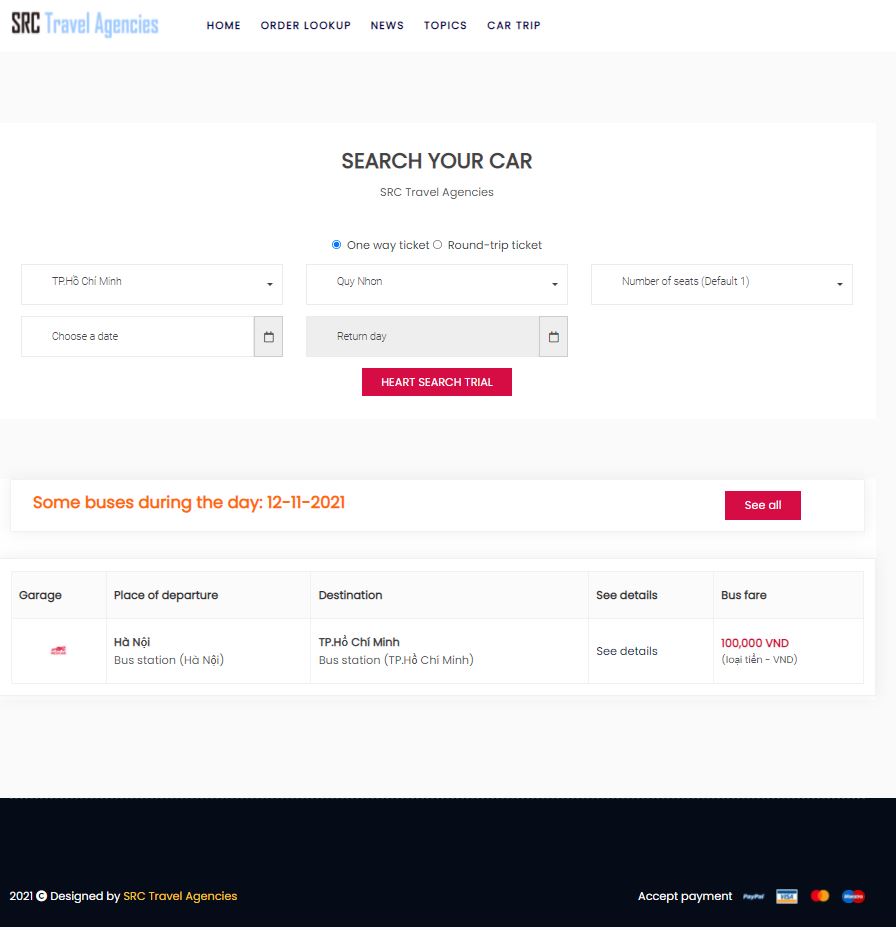








Home page



# 9User Guide

Watch video in this link to know how to use system:

https://youtu.be/D46zZiw-984