**FPT ACADEMY INTERNATIONAL**



**FPT – APTECH COMPUTER EDUCATION**

****

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Center Name** | | | |  | ACE-HCMC-2-FPT |  |
|  |  |  |  |  | Address: 590 Cach Mang Thang 8, District 3, Ho Chi Minh City, Viet Nam | |
| **Instructor** | | |  |  | Mrs. Nguyen Ha Vy |  |
| **Semester** | | |  |  | 3 |  |
| **Class** | | |  |  | T1.2209.E0 |  |
|  |  |  |  |  | **Name** | **Student ID** |
|  |  |  |  |  | Pham Huy Hoang | Student1421362 |
| **Group members** | | | |  | Tran Gia Toan | Student1416122 |
|  |  |  |  |  | Le Pham Tran Phu | Student1414240 |
|  |  |  |  |  | Bui Quoc Viet | Student1420715 |
| **Duration** |  |  | **Start date** |  | Jan 3rd 2024 |  |
|  |  | **End date** |  | Feb 5th 2024 |  |

Table of Contents

[REVIEW 1 3](#_Toc156913778)

[I. Acknowledge 4](#_Toc156913779)

[II. Introduction 4](#_Toc156913780)

[III. Synopsis 4](#_Toc156913781)

[IV. Problem definition 5](#_Toc156913782)

[V. Customer's Requirement Specifications 5](#_Toc156913783)

[VI. Requirement analysis 7](#_Toc156913791)

[TASK SHEET REVIEW 1 9](#_Toc156913792)

[REVIEW 2 10](#_Toc156913793)

[I. Architecture & Design of the Project 11](#_Toc156913794)

[II. Algorithms – Data Flowchart 12](#_Toc156913796)

[III. Use case diagram for users and admin 14](#_Toc156913797)

[TASK SHEET REVIEW 2 40](#_Toc156913798)

[REVIEW 3 41](#_Toc156913799)

[I. Site map 42](#_Toc156913800)

[II. Graphical User Interface (GUI) design 44](#_Toc156913803)

[TASK SHEET REVIEW 3 83](#_Toc156913824)

# **REVIEW 1**

## Acknowledge

First and foremost, we extend our sincere appreciation to our dedicated instructor, Mrs. Nguyen Ha Vy. Her insightful suggestions and unwavering encouragement were instrumental in helping us overcome not only technological challenges but also psychological tension. Without her guidance, our work would not have been successfully completed.

Furthermore, the seamless collaboration among our team members played a pivotal role in the successful execution and completion of this project. Despite encountering numerous obstacles along the way, our collective hard work, strong commitment, and camaraderie ultimately led us to achieve our goals.

Last but not least, we express our gratitude to Aptech's eProject assignment, which provided us with a golden opportunity to apply theoretical knowledge to real-world scenarios. The hands-on experience gained through this project is invaluable for our future careers.

## Introduction

Welcome to our online bus ticket booking platform! If you're a travel enthusiast seeking the convenience of reserving bus seats effortlessly from the comfort of your home, you've arrived at the perfect destination. Our website offers a reliable and user-friendly interface for you to search, explore, and book bus tickets, connecting you to various destinations with ease.

Embracing the convenience of modern technology, you no longer have to endure long lines or worry about seat availability. With just a few clicks, you can access a diverse range of bus routes and select the ones that best suit your travel preferences. We bring you not only convenient booking but also a seamless and enjoyable journey.

Designed with simplicity and user-friendliness in mind, our website allows you to effortlessly search for bus routes based on locations, operators, or departure times. Additionally, we provide comprehensive information for each route, including details about the bus operators, journey duration, and user reviews to help you make informed decisions.

Visit our website now to embark on a journey of hassle-free bus travel. Explore a variety of routes, check schedules, and book your bus tickets with ease. Prepare to experience the joy of seamless travel planning, ensuring a comfortable and enjoyable journey every time. Welcome to the future of online bus ticket booking – where convenience meets the excitement of exploration!

## Synopsis

Currently, the type of system being used at the counter is an internal system which is manually used in selling the bus tickets. The problems facing the company are that customers have to go to the counter to buy bus ticket or ask for bus schedule, customers will also have to queue up for a long time in order to secure a bus ticket and will also need to pay cash when they buy the bus ticket.

## Problem definition

1. **User-Friendly and Easy-to-Use Interface for Bus Booking**

Our online bus booking platform aims to feature an intuitive and user-friendly interface, ensuring that users can effortlessly find information about available bus routes, schedules, seat availability, and ticket prices. The website will be designed to organize information in a clear and logical manner, enhancing user experience and making the booking process straightforward.

1. **Flexible Bus Search and Schedule Options**

Our platform will provide an intelligent search tool that allows users to search for bus services based on various criteria such as departure/arrival locations, bus companies, departure times, and amenities. Additionally, detailed bus schedules will be available, enabling users to easily select convenient timings based on their travel preferences and plans.

1. **Easy Seat Selection and Booking Process**

For a hassle-free experience, our platform will offer a simple and convenient seat selection process. Users will have access to detailed bus layouts, allowing them to choose their preferred seats based on visual seating arrangements. After selecting seats, users can proceed with a secure and efficient online payment system, ensuring a seamless booking process.

By addressing these key points, our goal is to create a user-friendly and efficient online bus booking platform that caters to the needs of our customers, making their travel planning and booking experience convenient and enjoyable.

## Customer's Requirement Specifications

### Home Page:

#### Header – All Page for client will extend the header:

* Logo on the left side
* On the right side of navbar will show All Service Categories includes 4 Tab:
* News
* Become Our Driver (Sign up/Sign in for driver)
* Login(Register)

#### Body:

* Tab “Bus Schedule” show input form:
* Leaving from
* Going to
* Journey day (if customer do not select any date, default to the current date)
* Below form have a button “Search buses” can link to List Bus Routes Page.
* Tab “Check Ticket” show form Search to customer input ticket number. Beside form have a button “Check Ticket” can show information of customer include:
* Name
* Seat List
* Route
* Departure Time
* Arrival Time
* Bus Number
* Status (Notify the total remaining time from now until departure)
* Section Popular bus routes will show popular bus schedule in whole Vietnam. On each item bus route can link to List Bus Routes Page.
* Section introduction for PHTV Online Bus
* Section FAQ’s : A set of common questions that tend to be asked again and again.

#### Footer – All Page for client will extend the Footer:

* Show all information of owner ( owner email, all services, Quick links)
* At the end of the footer, will have a bar, on the left will show Copyright info, on the right will show Developed info.

### Login(Register) Page:

* This page will show the form to register or login.

### List Bus Routes Page:

* Will has sections about bus route, and every section will show information as bus type, time, number of seat, button “book now”
* Also customer can search for bus routes in this page.
* Sort and Filters: show the content of each customer's unique needs for the trip as below:
* Earliest departure
* Latest departure
* Lowest price
* Highest price
* Bus type
* Price

### News Page:

* This page will show news about tourism throughout Vietnam.

### Driver Login:

* This page will show the driver registration form.

### Authorize:

#### Admin:

* + Home Page
  + User Profile (Here the profile of the user is to be displayed. In this page itself the user should be able to edit the profile).
  + Bus management
  + Station management
  + Trip management
  + Promote trip management
  + User management
  + Driver management
  + Revenue tracking
  + Should be able to view, update or delete the informations that has uploaded on the site.

#### The user Authorize:

* + Home Page
  + User Profile (Here the profile of the user is to be displayed. In this page itself the user should be able to edit the profile).
  + Can view all the ticket booking history

#### The general user who is not the member of site Authorize:

* + Home Page
  + Can register with the site on the Login/Register Page
  + Should be able to view and search the bus routes
  + Can send feedback

### Non-functional Requirement Specifications

* + Good using Interface and Experience
  + Clean code
  + Try hard to improve the performance
  + Try limit bugs on this website

## Requirement analysis

1. **List of inputs to the system**
   * User login credentials (username, password).
   * Selected bus routes and departure/arrival details.
   * Seat preferences and booking details.
   * Payment information.
   * User feedback and ratings for bus services.
2. **List of outputs expected from the system**
   * Fast loading and smooth page redirection for an efficient booking process.
   * User dashboard displaying booked trips, upcoming schedules, and personalized recommendations.
   * Booking confirmation and e-tickets.
3. **Overview of Processes involved in the system**
   * User Authentication and Profile Management.
   * Bus Route and Schedule Management.
   * Seat Selection and Booking Process.
   * Payment Processing.
   * User Feedback and Ratings System.
4. **Hardware and Software required for implementing the project**
   * **Hardware:**
   * Intel Core i5 Processor or higher 8 GB RAM or above, Color SVGA
   * 500 GB Hard Disk space Mouse Keyboard
   * **Software:**
   * Frontend: Javascript, ReactJS, jQuery, HTML/CSS, Bootstrap, Antdesign
   * Backend: ASP.NET core Web API
   * Data store: SQL Server Management Studio (SSMS)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Ref. No.:**  **eP/Advertisement** **Portal Management** **System/01** | | **Project Title:** | **Activity Plan** **Prepared** **By:** | **Date of Preparation of Activity Plan:** | | | |
| **Sr.No.** | **Task** | **Actual Start** **Date** | **Actual** **Days** | **Team** **Mate** **Names** | **Status** |
| 1 | Acknowledge | PHTVBUS | All member | 12/01/2024 | 1 | Pham Huy Hoang | Completed |
| 2 | Introduction | 12/01/2024 | 1 | Tran Gia Toan | Completed |
| 3 | Statement of problem | 12/01/2024 | 1 | Le Pham Tran Phu | Completed |
| 4 | Problem definition | 12/01/2024 | 1 | Bui Quoc Viet | Completed |
| 5 | Customer's Requirement Specifications | 12/01/2024 | 1 | Pham Huy Hoang | Completed |
| 6 | Requirement analysis | 12/01/2024 | 1 | Bui Quoc Viet | Completed |

## TASK SHEET REVIEW 1

# **REVIEW 2**

## Architecture & Design of the Project

## A diagram of a data flow Description automatically generated

1. **Presentation**
   * Exposes interaction capabilities for the end users or applications / systems wanting to interact with it.
   * Interacts with the Business Logic layer.
   * Can interact with other applications (through their presentation layers).
2. **Business Logic**
   * Expresses domain knowledge, to be utilized from the Presentation layer.
   * Interacts with the Data Access layer.
   * Can interact with other applications (through their presentation layers).
3. **Data Access**
   * Acts as a data persistence, to be utilized from Business Logic.
   * Can interact with other applications (through their presentation layers).

## Algorithms – Data Flowchart

1. **The waterfall development Methodology**

A diagram of a design

Description automatically generated

1. **Context View of Online Bus Ticket Reservation System**

A diagram of a bus reservation

Description automatically generated

1. **Admin view of Online Bus Ticket Reservation System**

A diagram of a login and table

Description automatically generated

1. **User view of Online Bus Ticket Reservation System**

A diagram of a work flow

Description automatically generated

## Use case diagram for users and admin

A use case is a description of a system’s behaviour as it responds to a request that originates from outside of that system (the user). In figure 3.4, a use case of the activities in a bus transport system is shown.

A diagram of a person's diagram

Description automatically generated

In other words a use case describes “who” can do “what” with the system in question. The use case technique is used to capture a system’s behavioural requirements by detailing scenario-driven threads through the functional requirements.

1. **User**
2. **Login as user:**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Login | |
| **Actors** | Registered Customer | |
| **Description** | Customer who has registered an account can login | |
| **Requirements** | Customer provides username and password | |
| **Pre-conditions** | Customer must register an account | |
| **Post-conditions** | Success: Customer is logged in to website | |
| Fail: Refill information | |
|  | **Actor’s actions:** | **System’s responses:** |
|  | 1. Visitor clicks ‘Login’ button on Home  Page. | 2. System redirects to Login Page with the following controls: |
|  |  | * “Username” text field |
| **Basic flow** |  | * “Password” text field |
|  |  | * “Login” button |
|  | 3. Actor inputs Username and Password, then click the ‘Login’ button. | 4. System checks the information. |
|  |  | 5. System redirects to Customer’s homepage. |
| **Exceptions** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor inputs invalid username and password. | System redirects to Login page with the following controls:   * “User” text field * “Password” text field * “Login” button   System shows message: “Invalid username and password”. |

1. **Register as user:**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Register as user | |
| **Actors** | New Customer | |
| **Description** | Customer who has not registered an account | |
| **Requirements** | Customer registers an account on the website and becomes a member | |
| **Pre-conditions** | Customer provides their register information | |
| **Post-conditions** | Success: Customer creates a User account in this website | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Visitor clicks “Register” button on  Home Page. | 2. System redirects to Register page and displays Registration form with the following controls:   * “Username” text field * “Password” text field * “Email” text field * “Phone” text field |

|  |  |  |
| --- | --- | --- |
|  | 3. Customer inputs register’s information and click “Register” button. | * “Name” text field * “Register” button  1. System validates the information. 2. System inserts the account into database 3. System show success message. |
| **Exceptions** | **Actor’s actions:** | **System’s responses:** |
| * Actor left “Username” text field blank. * Username has already existed. * Actor left “Password” text field blank. * Actor left “Confirm Password” text field blank. * Password and confirm password don’t match. * Actor left “Email” text field blank. * Email has already existed. * Actor left “Phone” text field blank. * Input phone is not numeric format or phone length is not between 8 and 14 characters. * Actor left “Email” text field blank or input an invalid email address. | System displays message to customer:   * “Username cannot be blank”. * “This Username has already existed”. * “Password cannot be blank”. * “Confirm Password cannot be blank”. * “Confirm Password does not match password.”. * “Email cannot be blank”. * “This Email has already existed”. * “Phone cannot be blank”. * “The phone number should be numeric only and must be between 8 and 14 digits”. * “Email is invalid”. |

1. **Bus schedule**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Bus schedule | |
| **Actors** | Registered Customer/ New Customer | |
| **Description** | Customer can view bus schedule and bus trip detail | |
| **Requirements** | Customer select bus schedule and then select bus trip to show bus route’s details | |
| **Pre-conditions** | N/A | |
| **Post-conditions** | N/A | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor select bus schedule | 2. System shows list of bus routes |

1. **Buy ticket:**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Buy ticket | |
| **Actors** | Registered customer | |
| **Description** | Customer can order their ticket for the bus trip | |
| **Requirements** | Customer select bus, seats to buy tickets | |
| **Pre-conditions** | Customer must log into website and only can buy ticket for the seats which are available. | |
| **Post-conditions** | Success: Buy ticket successfully. | |
| Fail: Ticket is canceled. | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
|  | 1. Actor chooses “Buy ticket” to select bus they want. | 2. System redirects to showtime page. |
|  | 3. Actor chooses “seats” on the seating plan. | 4. System redirects displays a list of selected seat and total price(includes setting prices for each age group) |
|  | 5. Actor selects click “Payment” to confirm buying ticket | 6. System validates inputs by bus’s name of ticket |
|  |  | 7. System change status seating plan in database. |
|  |  | 8. System displays message : successful or error. |

1. **View account information:**

|  |  |  |
| --- | --- | --- |
| **Use case name** | View account information | |
| **Actors** | Registered Customer | |
| **Description** | Customer can view account information | |
| **Requirements** | Customer login to this website | |
| **Pre-conditions** | Customer provides their register information | |
| **Post-conditions** | Success: Visitor creates a User account in this website | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor select personal information | 2. System shows details of personal information |

1. **View booking history:**

|  |  |  |
| --- | --- | --- |
| **Use case name** | View booking history | |
| **Actors** | Registered Customer | |
| **Description** | Customer can view booking detail | |
| **Requirements** | Customer select booking history to show booking’s detail | |
| **Pre-conditions** | Customer provides their register information | |
| **Post-conditions** | Success: Customer creates a User account in this website | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor select booking history | 2. System shows detail of booking history |

1. **View bus detail:**

|  |  |  |
| --- | --- | --- |
| **Use case name** | View bus detail | |
| **Actors** | Registered customer | |
| **Description** | Customer can view bus detail | |
| **Requirements** | Customer select bus to show detail | |
| **Pre-conditions** | Customer must login if they want to comment on bus. | |
| **Post-conditions** | Success: Customer creates a User account in this website | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor select bus detail | 2. System shows information of bus. |

1. **Log out:**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Log out | |
| **Actors** | Registered customer | |
| **Description** | Customer back to home page as new customer | |
| **Requirements** | Customer must has at least 1 account | |
| **Pre-conditions** | Customer must log in before log out | |
| **Post-conditions** | Success: Actor log out successfully | |
| N/A | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
|  | 1. Actor click “Log out” button | 1. System displays message : log out successful. 2. System redirect customer to homepage. |
|  | | |

1. **Check ticket**

|  |  |  |
| --- | --- | --- |
| **Use case name** | View ticket detail | |
| **Actors** | Registered Customer/ New Customer | |
| **Description** | Customer can view ticket detail | |
| **Requirements** | Customer input ticket number to show each ticket’s detail | |
| **Pre-conditions** | N/A | |
| **Post-conditions** | N/A | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
|  | 1. Actor input ticket’s number | 2. System shows details of ticket |

1. **News**

|  |  |  |
| --- | --- | --- |
| **Use case name** | View news, review related to bus trip in whole Viet Nam | |
| **Actors** | Registered Customer/ New Customer | |
| **Description** | Customer can view news, review related to bus trip in whole Viet Nam | |
| **Requirements** | Customer select news button to show news, review related to bus trip | |
| **Pre-conditions** | N/A | |
| **Post-conditions** | N/A | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
|  | 1. Actor select news | 2. System shows news, review related to bus trip |

1. **Admin**
2. **Login**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Login | |
| **Actor** | Admin | |
| **Description** | When admin wants to manage member , news and bus | |
| **Requirements** | Admin provides username and password | |
| **Pre-conditions** | Admin must be exist on database | |
| **Post-conditions** | Success: Admin is logged in to website admin | |
| Fail: refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor clicks on ‘Login’ button.  3. Actor inputs Username and Password, then click the ‘Login’ button. | 2. System redirects to Login Page with the following controls:   * 'Username' text field * 'Password' text field * 'Login' button  1. System validates the information. 2. System redirects to Admin website |
| **Exceptions** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor inputs invalid email and password | System redirects to Login page with the following controls:   * ‘Username’ text field * ‘Password’ text field * ‘Login’ button. |

1. **Manage member**
   * + **Create new member**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Create new member | |
| **Actor** | Admin | |
| **Description** | When admin wants to create a new member | |
| **Requirements** | Actor provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: new product is added into the database | |
| Fail: refill information | |
|  | **Actor’s actions:** | **System’s responses:** |
| **Basic flow** | 1. Actor chooses Bus Management Tab on menu.  3. Actor chooses Bus Management/ Bus type management Tab on menu.  5. Actor clicks ‘Add New Bus/ Bus type’ button.  7. Actor clicks ‘Add User’ button.  9. Actor inputs information to create new product | 2. System displays a form with the  following fields:  - Bus management  - Bus type management  4. System redirects to ‘Bus/ Bus type’ page and display a list of existing bus/ bus type.  6. System displays a form with the  following fields:  - ID  - Name  - Description  - Note  8. System validates inputs.  10. System adds member into database.  11. System displays confirmation message or error message. |

* + - **Update member**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Update member’s detail | |
| **Actor** | Admin | |
| **Description** | When admin wants to update member | |
| **Requirements** | Admin provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: member detail is updated in the database | |
| Fail: refill information | |
|  | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses Member Tab on menu.  3. Actor clicks on ‘View Details’ on a member.  5. Actor chooses the criteria they want to update and input information. | 2. System redirects to ‘Member Details’ page and display a list of existing member.  4. System displays a form with the following fields.   * username * name * email * phone number * image   6. System validates inputs.  7. System update information member into database.  8. System displays confirmation message or error message.  9. System redirects to member’s detail page. |

* + - **Delete member**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Delete member | |
| **Actor** | Admin | |
| **Description** | When admin wants to delete member from the database | |
| **Requirements** | Admin provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: product is delete in the database | |
| Fail: product is not delete from the database | |
|  | **Actor’s actions:** | **System’s responses:** |
|  | 1. Actor chooses Member Tab on menu. | 2. System redirects to ‘Member List’ page and display a list of existing member. |
|  | 3. Actor clicks ‘Delete’ button. | 4. System delete member in the database |
| **Basic flow** |  | 5. System displays confirmation or error. |

1. **Manage Bus**
   * + **Create new bus**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Create new bus | |
| **Actor** | Admin | |
| **Description** | When admin wants to create a new bus | |
| **Requirements** | Actor provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: new product is added into the database | |
| Fail: refill information | |
|  | **Actor’s actions:** | **System’s responses:** |
| **Basic flow** | 1. Actor chooses Bus Management Tab on menu.  3. Actor chooses Bus Management/ Bus type management Tab on menu.  5. Actor clicks ‘Add New Bus/ Bus type’ button.  7. Actor clicks ‘Add User’ button.  9. Actor inputs information to create new product | 2. System displays a form with the  following fields:  - Bus management  - Bus type management  4. System redirects to ‘Bus/ Bus type’ page and display a list of existing bus/ bus type.  6. System displays a form with the  following fields:  - ID  - Name  - Description  - Note  8. System validates inputs.  10. System adds member into database.  11. System displays confirmation message or error message. |

* + - **Update Bus**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Update bus’s detail | |
| **Actor** | Admin | |
| **Description** | When admin wants to update bus | |
| **Requirements** | Admin provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: bus is updated in the database | |
| Fail: refill information | |
|  | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses bus Tab on menu.  3. Actor clicks on ‘View Details’ on a bus.  5. Actor chooses the criteria they want to update and input information. | 2. System redirects to ‘ Details’  page and display a list of existing product.  4. System displays a form with the following fields.  - ID  - Name  - Description  - Function  6. System validates inputs.  7. System update information bus into database.  8. System displays confirmation message or error message. |

* + - **Delete bus**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Delete bus | |
| **Actor** | Admin | |
| **Description** | When admin wants to delete bus from the database | |
| **Requirements** | Admin provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: bus is delete in the database | |
| Fail: bus is not delete from the database | |
|  | **Actor’s actions:** | **System’s responses:** |
| **Basic flow** | 1. Actor chooses Bus Tab on menu.  3. Actor clicks ‘Delete’ button. | 2. System redirects to ‘Bus List’ page and display a list of existing bus. |
|  |  | 4. System delete bus in the database |
|  |  | 5. System displays confirmation or error. |

1. **Manage Station**
   * + **Create new Station**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Create new Station | |
| **Actor** | Admin | |
| **Description** | When admin wants to create a new Station | |
| **Requirements** | Actor provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: new product is added into the database | |
| Fail: refill information | |
|  | **Actor’s actions:** | **System’s responses:** |
| **Basic flow** | 1. Actor chooses Station Management Tab on menu.  3. Actor clicks ‘Add New Station  button.  5. Actor clicks ‘Station management’ button.  7. Actor inputs information to create new product | 2. System redirects to ‘Station page  and display a list of existing Station.  4. System displays a form with the  following fields:  - ID  - Name  - Description  - Function  6. System validates inputs.  8. System adds Station into database.  9. System displays confirmation message or error message. |

* + - **Update Station**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Update Station’s detail | |
| **Actor** | Admin | |
| **Description** | When admin wants to update Station | |
| **Requirements** | Admin provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: Station is updated in the database | |
| Fail: refill information | |
|  | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses Station Tab on menu.  3. Actor clicks on ‘View Details’ on a Station.  5. Actor chooses the criteria they want to update and input information. | 2. System redirects to ‘ Details’  page and display a list of existing product.  4. System displays a form with the following fields.  - ID  - Name  - Description  - Function  6. System validates inputs.  7. System update information Station into database.  8. System displays confirmation message or error message. |

* + - **Delete Station**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Delete Station | |
| **Actor** | Admin | |
| **Description** | When admin wants to delete Station from the database | |
| **Requirements** | Admin provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: Station is delete in the database | |
| Fail: Station is not delete from the database | |
|  | **Actor’s actions:** | **System’s responses:** |
| **Basic flow** | 1. Actor chooses Station Tab on menu.  3. Actor clicks ‘Delete’ button. | 2. System redirects to ‘Station List’ page and display a list of existing Station. |
|  |  | 4. System delete Station in the database |
|  |  | 5. System displays confirmation or error. |

1. **Manage Trip**
   * + **Create new Trip**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Create new Trip | |
| **Actor** | Admin | |
| **Description** | When admin wants to create a new Trip | |
| **Requirements** | Actor provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: new product is added into the database | |
| Fail: refill information | |
|  | **Actor’s actions:** | **System’s responses:** |
| **Basic flow** | 1. Actor chooses Trip Management Tab on menu.  3. Actor clicks ‘Add New Trip  button.  5. Actor clicks ‘Trip management’ button.  7. Actor inputs information to create new product | 2. System redirects to ‘Trip page  and display a list of existing Trip.  4. System displays a form with the  following fields:  - ID  - Name  - Description  - Time  - Price  - Function  6. System validates inputs.  8. System adds Trip into database.  9. System displays confirmation message or error message. |

* + - **Update Trip**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Update Trip’s detail | |
| **Actor** | Admin | |
| **Description** | When admin wants to update Trip | |
| **Requirements** | Admin provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: Trip is updated in the database | |
| Fail: refill information | |
|  | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses Trip Tab on menu.  3. Actor clicks on ‘View Details’ on a Trip.  5. Actor chooses the criteria they want to update and input information. | 2. System redirects to ‘ Details’  page and display a list of existing product.  4. System displays a form with the following fields.  - ID  - Name  - Description  - Time  - Price  6. System validates inputs.  7. System update information Trip into database.  8. System displays confirmation message or error message. |

* + - **Delete Trip**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Delete Trip | |
| **Actor** | Admin | |
| **Description** | When admin wants to delete Trip from the database | |
| **Requirements** | Admin provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: Trip is delete in the database | |
| Fail: Trip is not delete from the database | |
|  | **Actor’s actions:** | **System’s responses:** |
| **Basic flow** | 1. Actor chooses Trip Tab on menu.  3. Actor clicks ‘Delete’ button. | 2. System redirects to ‘Trip List’ page and display a list of existing Trip. |
|  |  | 4. System delete Trip in the database |
|  |  | 5. System displays confirmation or error. |

1. **Revenue**

|  |  |  |
| --- | --- | --- |
| **Use case name** | Revenue tracking | |
| **Actor** | Admin | |
| **Description** | When admin wants to check revenue | |
| **Requirements** | Actor provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | N/A | |
| N/A | |
|  | **Actor’s actions:** | **System’s responses:** |
| **Basic flow** | 1. Actor chooses Revenue Tab on menu.  3. Actor chooses Monthly revenue / Revenue by route Tab on menu.    . | 2. System displays a form with the  following fields:  - Monthly revenue  - Revenue by route  4. System redirects to ‘Revenue by route / Revenue by route’ page and display a list of revenue  5. System displays a form with the  following fields:  - ID  - Name  - Description  - Note  -Revenue |

1. **Driver**
2. **Login as driver:**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Login | |
| **Actors** | Registered driver | |
| **Description** | Driver who has registered an account can login | |
| **Requirements** | Driver provides username and password | |
| **Pre-conditions** | Driver must register an account | |
| **Post-conditions** | Success: Driver is logged in to website | |
| Fail: Refill information | |
|  | **Actor’s actions:** | **System’s responses:** |
|  | 1. Driver clicks ‘Login’ button ‘Become Our Driver’ | 2. System redirects to Login Page with the following controls: |
|  |  | * “Username” text field |
| **Basic flow** |  | * “Password” text field |
|  |  | * “Login” button |
|  | 3. Actor inputs Username and Password, then click the ‘Login’ button. | 4. System checks the information. |
|  |  | 5. System redirects to Driver’s homepage and displays informations include:  - Trip code  - Image  - Station list  - Time  - Ticket price  - Assigned driver  - Assigned bus |
| **Exceptions** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor inputs invalid username and password. | System redirects to Login page with the following controls:   * “User” text field * “Password” text field * “Login” button   System shows message: “Invalid username and password”. |

1. **Register as Driver:**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Register as Driver | |
| **Actors** | New Driver | |
| **Description** | Driver who has not registered an account | |
| **Requirements** | Driver registers an account on the website and becomes a employee | |
| **Pre-conditions** | Driver provides their register information | |
| **Post-conditions** | Success: Customer creates a Driver account in this website | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Driver clicks “Register” button ‘Become Our Driver’ | 2. System redirects to Register page and displays Registration form with the following controls:   * “Username” text field * “Password” text field * “Email” text field * “Phone” text field |

|  |  |  |
| --- | --- | --- |
|  | 3. Customer inputs register’s information and click “Register” button. | * “FullName” text field * “Driver license” text field * “Place of birth” text field * Image file for avatar * “Register” button  1. System validates the information. 2. System inserts the account into database 3. System show success message. |
| **Exceptions** | **Actor’s actions:** | **System’s responses:** |
| * Actor left “Username” text field blank. * Username has already existed. * Actor left “Password” text field blank. * Actor left “Confirm Password” text field blank. * Password and confirm password don’t match. * Actor left “Email” text field blank. * Email has already existed. * Actor left “Phone” text field blank. * Input phone is not numeric format or phone length is not between 8 and 14 characters. * Actor left “Email” text field blank or input an invalid email address. | System displays message to customer:   * “Username cannot be blank”. * “This Username has already existed”. * “Password cannot be blank”. * “Confirm Password cannot be blank”. * “Confirm Password does not match password.”. * “Email cannot be blank”. * “This Email has already existed”. * “Phone cannot be blank”. * “The phone number should be numeric only and must be between 8 and 14 digits”. * “Email is invalid”. |

## TASK SHEET REVIEW 2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Ref. No.:**  **eP/Advertisement** **Portal Management** **System/01** | | **Project Title:** | **Activity Plan** **Prepared** **By:** | **Date of Preparation of Activity Plan:** | | | |
| **Sr.No.** | **Task** | **Actual Start** **Date** | **Actual** **Days** | **Team** **Mate** **Names** | **Status** |
| 1 | Architecture & design of the project | PHTVBUS | Pham Huy Hoang | 22/01/2024 | 1 | All member | Completed |
| 2 | Algorithms – data flowchart | 22/01/2024 | 1 | All member | Completed |
| 3 | Use case diagram for users and admin | 22/01/2024 | 1 | All member | Completed |

# **REVIEW 3**

## Site map

### Client site map

A diagram of a customer

Description automatically generated

### Dashboard site map

A diagram of a diagram

Description automatically generated

## Graphical User Interface (GUI) design

### Client GUI

## 

## A screenshot of a web page Description automatically generated

### Home page (Index)

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Process/Module Description and Details:** overview of popular Bus Routes in whole Viet Nam, search and booking trip, Check the tickets which the user has just added to the booking list.

### Bus Schedule

### A screenshot of a computer Description automatically generated

**Process/Module Description and Details:** search and booking trip, advanced search with sort and filter functions, proceed to pay for tickets.

### Check Ticket

### A screenshot of a computer Description automatically generated

**Process/Module Description and Details:** overview of popular Bus Routes in whole Viet Nam, search the tickets which the user has just added to the booking list.

### News page

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

**Process/Module Description and Details:** this page will display all latest news of bus trip on the world/ shows that uploaded.

### Dashboard GUI

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### Login

### Sign in

### A login screen with a building and trees Description automatically generated

### Sign up

### A screenshot of a login form Description automatically generated

### Forgot password

### A login screen with a building and a river Description automatically generated

### A screenshot of a login page Description automatically generated

### A screenshot of a computer Description automatically generated

**Process/Module Description and Details:** these pages show authentication system includes login for existing users, registration for new users, and a password recovery feature for those who forget their passwords. Users log in with their credentials, register by providing necessary details, and can recover passwords through a secure process such as email verification or temporary passwords.

### User profile

### A screenshot of a computer Description automatically generated

### A screenshot of a video game Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

**Process/Module Description and Details:** show user information, booking history, and the ability to update user details. Users can view and manage their personal information, check their booking history, and update their profile details such as name, email, or password. This provides a centralized hub for users to interact with and control their account on the platform.

### Admin page

### Bus Management

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### Bus Type Management

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### Station Management

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### Trip Management

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### Promote Trip Management

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### Offer Management

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### Moderator management

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### Customer Management

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### Driver Management

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### Register Driver Management

### A screenshot of a computer Description automatically generated

### Revenue by Month

### A screenshot of a computer Description automatically generated

### Revenue by Route

### A screenshot of a computer Description automatically generated

### FAQ Management

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### News Management

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

**Process/Module Description and Details:** showing page for managing a bus-related website encompasses various functionalities, including:

* Bus Management:

Add, edit, or remove buses from the system.

View and update information about each bus, such as model, capacity, and status.

* Bus Station Management:

Manage bus station details, including location, facilities, and operating hours.

Add, edit, or remove bus stations.

* Trip Management:

Schedule, modify, or cancel bus trips.

Track and manage real-time information about ongoing trips.

* Popular Trips:

Monitor and manage the popularity of bus trips.

Highlight or feature the most popular routes for users.

* Discounts and Promotions:

Create, modify, or deactivate special discounts or promotions for bus fares.

* User Management:

View, edit, or deactivate user accounts.

Access user details, booking history, and feedback.

* Driver Management:

Add, edit, or remove information about bus drivers.

Track driver schedules and performance.

* Revenue Management:

Monitor and analyze revenue streams.

Access reports on ticket sales, profits, and financial performance.

* Frequently Asked Questions (FAQ) Management:

Create, edit, or remove FAQs to address common user queries.

Organize FAQs into categories for easy navigation.

* News Management:

Post and manage news articles on the homepage.

Highlight important announcements or updates.

* Ticket Verification:

Verify and check ticket codes

The admin page provides a centralized dashboard for administrators to efficiently manage and oversee the entire bus transportation system, ensuring smooth operations, user satisfaction, and up-to-date information on the website.

### Driver Sign up

### A screenshot of a computer Description automatically generated

### Driver Sign in

### A screenshot of a computer Description automatically generated

### Driver page

### 

**Process/Module Description and Details:** these pages show authentication system for drivers includes a login page for existing drivers, a password recovery feature for forgotten passwords, and a registration page for new drivers. Additionally, there's a dedicated page displaying detailed information about each driver after they have logged in or registered. This setup ensures a seamless and secure experience for driver authentication, password recovery, registration, and accessing driver profiles.

### Booking page

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

**Process/Module Description and Details:** showing page for booking ticket including:

* Select Trip:

### Users can choose their desired trip from the available list of options.

### Detailed information about the departure point, destination, departure time, and ticket price will be displayed.

* Choose Seat:

### Users can select an available seat on the chosen trip from the seat layout.

### The layout visually represents the location of each seat, allowing users to pick their preferred seat.

* Select Passenger Age:

### Users need to provide information about the age of the passenger.

### The discount policy applies: free for children under 5 years old, a 50% discount for children aged 5 to 12, and a 30% discount for passengers over 50 years old.

* Discount Code:

### Users can input a discount code (if available) to enjoy special benefits before proceeding to payment.

* Online Payment:

### Users proceed with the online payment using a credit card.

### The system ensures the safety of personal information and transactions.

### This ticket booking process allows users flexibility in selecting trips, seats, and supports discount policies based on passenger age. Additionally, it provides an option to use discount codes and facilitates online payments, optimizing the ticket booking experience on the website.

### About us

### A screenshot of a computer Description automatically generated

### A screenshot of a computer Description automatically generated

**Process/Module Description and Details:** this page provides information about PHTVBUS, products, and services we offer. Additionally, it helps readers gain a deeper understanding of the mission, vision, and development direction of the website. There is also an introduction to website development members.

## TASK SHEET REVIEW 3

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Ref. No.:**  **eP/Advertisement** **Portal Management** **System/01** | | **Project Title:** | **Activity Plan** **Prepared** **By:** | **Date of Preparation of Activity Plan:** | | | |
| **Sr.No.** | **Task** | **Actual Start** **Date** | **Actual** **Days** | **Team** **Mate** **Names** | **Status** |
| 1 | Site Map | PHTVBUS | Pham Huy Hoang | 23/01/2024 | 1 | All member | Completed |
| 2 | Graphical User Interface (GUI) design | 23/01/2024 | 1 | All member | Completed |