**FPT ACADEMY INTERNATIONAL**



**FPT – APTECH COMPUTER EDUCATION**

**Airlines Reservation System**

|  |  |  |  |
| --- | --- | --- | --- |
| **Center Name:** | FPT – Aptech – HCM2 | | |
|  | Address:590 Cach Mang Thang Tam,  District 3, Ho Chi Minh City | | |
| **Instructor:** | Mr. Hoang Duc Quang | | |
| **Class:** | T1.2008.M0 | | |
| **Group:** | 02 | | |
| **Group members** | Name  Huỳnh Trần Thế Hùng | | Student ID  Student1273503 | |
|  | Nguyễn Văn Nhân | | Student1273005 | |
|  | Nguyễn Ngọc Sơn Tùng  Hoàng Hải Long | | Student1273494  Student1273511 | |
|  |  | |  | |
| **Project Duration** | Start date | 1st Dec 2021 | | |
|  | End date | 31th Dec 2021 | | |

**This is to certify that**

**Mr./Ms.:**

Huỳnh Trần Thế Hùng

Nguyễn Văn Nhân

Nguyễn Ngọc Sơn Tùng

Hoàng Hải Long

Has successfully designed & developed

Project: Airlines Reservation System

**Submitted by:**

Huỳnh Trần Thế Hùng

**Date of issue:**  Dec 1st 2021

**Authorized Signature:**

Contents

[REVIEW 1 4](#_Toc90038658)

[1. ACKNOWLEDGE 5](#_Toc90038659)

[2. INTRODUCTION 5](#_Toc90038660)

[3. PROBLEM DEFINITION 6](#_Toc90038661)

[FROM VIEWER'S VIEWPOINTS 6](#_Toc90038662)

[FROM THE POINT OF THE SMALL ADMINISTRATIVE 6](#_Toc90038663)

[4. CUSTOMER'S REQUIREMENTS SPECIFICATIONS (CRS) 6](#_Toc90038664)

[CLIENT: Airlines Reservation System 6](#_Toc90038665)

[BUSINESS/PROJECT OBJECTIVE: 6](#_Toc90038666)

[HARDWARE/ SOFTWARE REQUIREMENTS: 8](#_Toc90038667)

[5. TASK SHEET REVIEW 1 10](#_Toc90038668)

[REVIEW 2 12](#_Toc90038669)

[1. USECASE 13](#_Toc90038670)

[2. DATA FLOW DIAGRAM (DFD) 31](#_Toc90038671)

[Sequence Diagram for Airline Reservation System Design 33](#_Toc90038672)

[ Customer: 33](#_Toc90038673)

[ Admin: 33](#_Toc90038674)

[Class diagram 34](#_Toc90038675)

[ENTITY RELATIONSHIP DIAGRAM (ERD) 34](#_Toc90038676)

[ENTITY PROPERTIES 35](#_Toc90038677)

[3. TABLE DESIGN 37](#_Toc90038678)

[4.TASK SHEET REVIEW 2 41](#_Toc90038679)

# REVIEW 1

## ACKNOWLEDGE

Our team would like to dedicate this chapter to express our sincere gratitude to all those who have provided us with valuable support during the implementation of this Electronic Project.

First, we are indebted to our dedicated guide, Mr. Hoang Duc Quang, who gave us advice and encouragement to help us overcome not only technological challenges but also psychological stresses. Without him, our work could not have been completed brilliantly.

Second, we appreciate the eProject Team at Headquarters, which gave us a great opportunity to put what we've learned into practice. We are sure that the experience that we gain from this project will greatly facilitate our future career.

Third, we would also like to thank all the staff at my center who not only gave me the opportunity to work with them on this project, but also supported and encouraged them throughout the process.

Finally, more importantly, we would like to personally thank all of our dear classmates and team members who were always willing to support us openly and enthusiastically.

## INTRODUCTION

We are in an era where information technology makes buying and selling easier than ever. People can order anything on their phones, and everything is completed electronically, interactively in real time. So we are developing a website called Airlines Reservation System to catch up with today's online shopping trend.

Airlines Reservation System is for an organization that wants to develop an online store where their user can be someone who doesn't have a lot of time to go to the store or someone who has a habit of shopping online or simply being in too far from independent stores. It will provide users with the following features.

For Customer:

• Customers can register an account.

• Customers can login and manage their accounts.

• Customers can search for any flight such as Local (Dosmestic), International .

• Customers can search for products by price.

• Customers can view flight introduction before buying.

• Customers can purchase products after adding products to cart.

• Customers can view cart history.

• Customers can interact with the website by asking questions, giving feedback on the contact us function.

## PROBLEM DEFINITION

After analyzing the specifications of the Airlines Reservation System, in order to meet the customer's requirements correctly, we have concluded that for the best results, we should address the following issues. following query:

### FROM VIEWER'S VIEWPOINTS

• A home page will display the main products, the appropriate logo, display the logo on all pages in the site.

• A menu will include all the functions performed in the website.

• Customers can purchase products, add products to cart, view cart history and checkout after logging into their account

### FROM THE POINT OF THE SMALL ADMINISTRATIVE

• The login page will grant administrator access to the Product.

• Administrators can view, add, update and delete products.

• Administrators can view, add, update, delete, find, lock and unlock customer accounts.

• Admin can manage news and orders.

## CUSTOMER'S REQUIREMENTS SPECIFICATIONS (CRS)

### **CLIENT:** Airlines Reservation System

### ****BUSINESS/PROJECT OBJECTIVE:****

Website design project to sell products online: international flights and domestic flights

**Input to the system:**

**The product store administrator needs to provide:**

• Information needed to create an account to access the administrative system.

• The product will be displayed on the website.

**Product store customers need to provide:**

Personal information to purchase products.

**Output from the system:**

**Administrators will be able to:**

* Login / Log out of the administrative system.
* List Flights.
* Arrange Flights.
* Add Flights.
* Schedule Flights.
* Add Tickets
* Manage Tickets
* Checked Book Tickets.
* Update Flights
* Manage Price

**Anonymous will be able to:**

* Register for a customer account.
* Search Flights
* View Flights
* Book the Tickets after adding information.
* View the Tickets they have selected.

**Customers will be able to:**

* Signup As Customer.
* Log in / Log out of your account.
* Edit Profile
* Search Flights
* View Flights
* Book Tickets.
* Print Ticket.
* Check PNR.
* View order history.

**SCOPE OF WORK**

The application should perform the following functions:

**Login:**

* There are 1 types of account in the application (“admin”). Administrator will login as “admin”.
* For “admin”: can view the ticketsid, name, add new, update, delete.
* For “admin”: can view the customer id, name, add new, update, delete, lock, unlock.
* For “admin”: can view and manage view.
* For “admin”: can view and manage order.

**Buying products:**

* When customer choose a product and the selected product will be added to the cart.
* Customer can buy tickets at shopping cart after pressing the “Checkout” button and fill in the information.

**Database:**

* The information about tickets, customers, news, orders will be stored in the database.
* The information from the contact us page.

### HARDWARE/ SOFTWARE REQUIREMENTS:

**Hardware**

##### **Web Server**

|  |  |
| --- | --- |
| **Processor** | Intel Core I5 or higher. |
| **Memory** | 4 GB RAM or greater. |
| **Modem/ADSL** | Internet access is required. |

##### **Client**

|  |  |
| --- | --- |
| **Processor** | Intel Core I3 or higher. |
| **Memory** | 2 GB RAM or greater. |
| **Monitor** | Super VGA (1024x768) or higher resolution. |
| **Modem/ADSL** | Internet access is required. |

#### ****Software****

##### **Web Server**

|  |  |
| --- | --- |
| **Operation System** | Window home. |
| **Browser** | Google Chrome version 35. |
| **Database** | MySQL. |
| **Software** | Internet Information Service |
| ASP.NET CORE MVC |

##### **Client**

|  |  |
| --- | --- |
| **Operation System** | Window 7 or higher. |
| **Browser** | Google Chrome version 35. |

#### ****Development software****

* Microsoft Windows 10.
* MySQL.
* Visual Studio Code.
* Google Chrome.
* XAMPP.

#### ****Technology****

* ASP.NET MVC.
* JQuery.
* Bootstrap.
* HTML/CSS

## TASK SHEET REVIEW 1

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Project Ref. No.: Organic Product Store | | Project Name | Activity Plan Prepared By | Date of Preparation of Activity Plan: | | | | |
| Sr./No. | Task | Actual Start Date | Actual Days | Team member name | Status |
| 1 | Acknowledgment | AirLine  Reservation  System | All Team | 1-Dec-21 | 1 | Hùng | Completed |
| 2 | Introduction | 2-Dec-21 | 1 | Hùng | Completed |
| 3 | Problem Definition | 3-Dec-21 | 1 | Tùng | Completed |
| 4 | Customer Requirement Specification (CRS) | 4-Dec-21 | 3 | Tùng | Completed |
| 5 | Scope of Work | 5-Dec-21 | 2 | Nhân | Completed |
| 6 | Hardware/Software Requirement | 6-Dec-21 | 2 | Nhân | Completed |
| 7 | Task Sheet | 7-Dec-21 | 1 | Long | Completed |

|  |  |
| --- | --- |
| Date: 8 Dec 2021 | |
| Signature of Instructor:  **Mr. Hoang Duc Quang** | Signature of Team Leader:  **Nguyễn Văn Nhân** |

# REVIEW 2

## USECASE

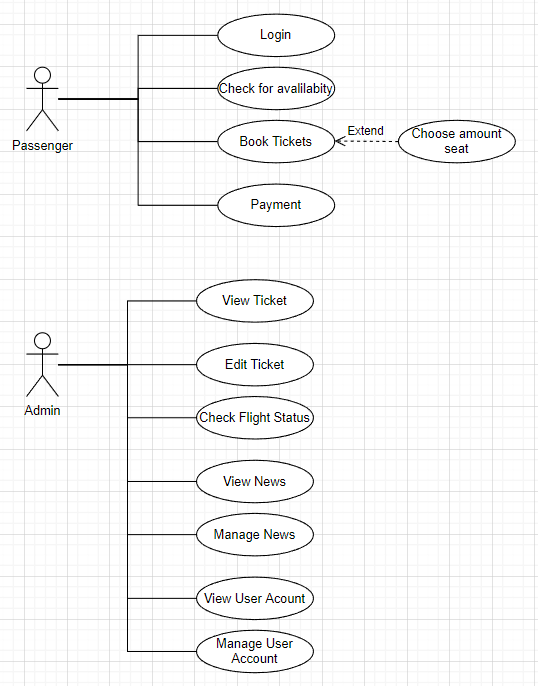
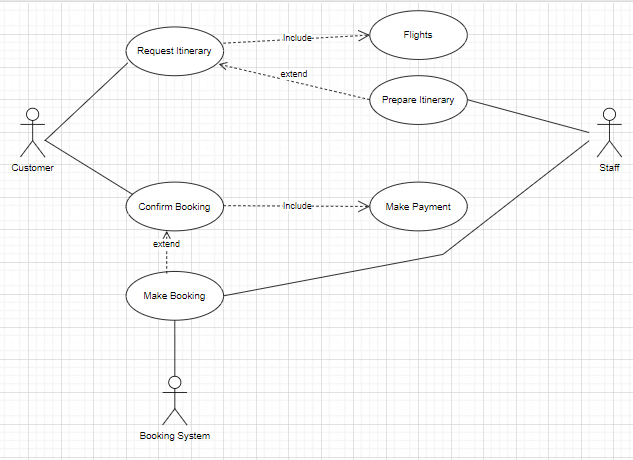
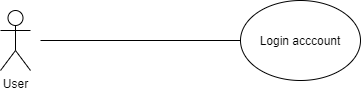


Fig 1.1 Airline Reservation System



**CUSTOMER**

****

**LOGIN AS CUSTOMER**

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | Login | |
| **Actors** | 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 | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Customer clicks ‘Login’ button on Home Page.  3. Actor inputs UserAccount and Password, then click the ‘Login’ button. | 2. System redirects to Login Page with the following controls:  - “UserAccount” text field  - “Password” text field  - “Login” button  4. System checks the information.  5. System redirects to Customer’s dashboard. |
| **Exceptions** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor inputs invalid UserAccount and password. | System redirects to Login page with the following controls:  - “UserAccount” text field  - “Password” text field  - “Login” button  System shows message: “Account or password is not precision”. |

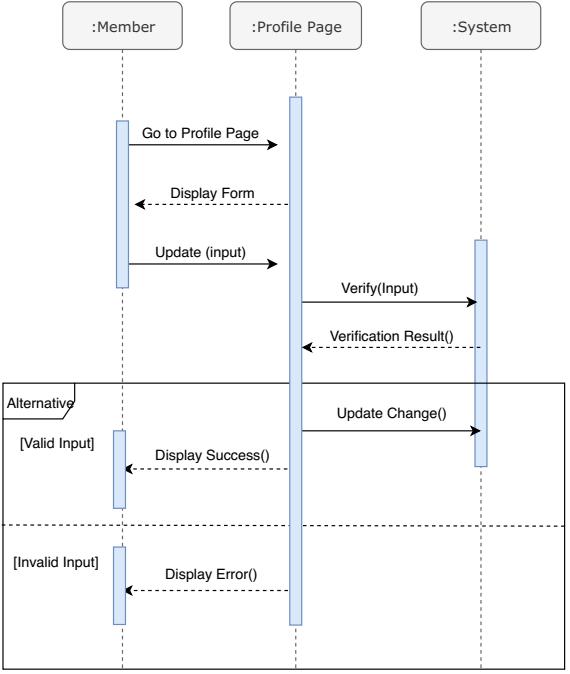
**REGISTER AS CUSTOMER:**

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | Register as Customer | |
| **Actors** | Customer | |
| **Description** | Customer who has registered an account can login | |
| **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. Customer clicks Hompage => Login => “Register” button on LoginTable  3. Customer inputs register’s information and click “Register” button. | 2. System redirects to Register page and displays Registration form with the following controls:  - “Full-name” text field  - “Username” text field  - “Password” text field  - “Confirm Password” text field  - “Email” text field  -“Address” text field  - “Phone” text field  - “Register” button  - “Reset or Delete” button  4. System validates the information.  5. System inserts the account into database  6. 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.”.  - “Address cannot be blank”.  - “This Address has already existed”.  - “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”. |

#### VIEW PROFILE

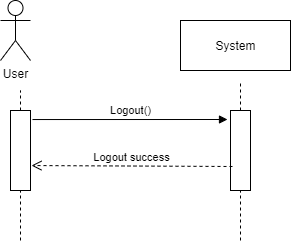
|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | View profile | |
| **Actors** | Customer | |
| **Description** | Customer who want to view their profile | |
| **Requirements** | Customer provides their personal information | |
| **Pre-conditions** | Customer is logged in | |
| **Post-conditions** | Success: Actor can view their profile | |
| Fail: Actor cannot view their profile | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor clicks “Account” button on Menu Bar. | 2. System redirects to Account page and displays Account form with the following controls:  - “Full-name” text field  - “Address” text field  - “Email” text field  - “Phone” text field |

#### EDIT PROFILE:



|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | Edit profile | |
| **Actors** | Customer | |
| **Description** | Customer who want to update their profile | |
| **Requirements** | Customer provides their personal information | |
| **Pre-conditions** | Customer is logged in | |
| **Post-conditions** | Success: Actor can edit their profile | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor clicks “Account” button on Menu Bar.  3. Actor inputs personal information and clicks “Submit”. | 2. System redirects to Account page and displays Account form with the following controls:  - “Full-name” text field  - “Address” text field  - “Email” text field  - “Phone” text field  - “Update” button  4. System validates password then sends to database.  5. System shows message: “You must want to change your personal information!” =>” OK ”. |
| **Exceptions** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor login an account which does not exist in database. | System shows message: “Invalid account”. |

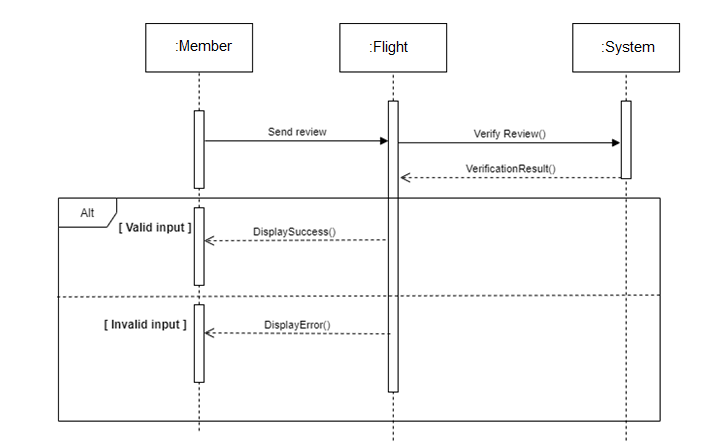
**LOGOUT**

****

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | Logout | |
| **Actors** | Customer | |
| **Description** | Customer clicks on “Logout” button and their session is terminated. | |
| **Requirements** | N/A | |
| **Pre-conditions** | 1. Customer is logged in  2. Customer no longer wants to be logged in | |
| **Post-conditions** | Success: Customer can logout | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor clicks “Log out” button on Menu Bar. | 2. The system logs the actor out and invalidates the session.  3. The system redirects to the default logout page. |
| **Exceptions** | Actor’s actions: | System’s responses: |
| 1. Actor cannot logout | System shows message: “Not logged in”. |

**Payment**

**VIEW FLIGHT DETAIL**

****

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | View Flight | |
| **Actors** | Registered Customer/ New Customer | |
| **Description** | Customer can view Flight detail | |
| **Requirements** | Customer select flight to show flight’s detail | |
| **Pre-conditions** | N/A | |
| **Post-conditions** | Success: Display flight detail matching the selected flight | |
| Fail: No flight is shown | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor select Flight | 2. System shows flight’s detail matching selected product |

**VIEW CART**

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | View cart | |
| **Actors** | Registered customer | |
| **Description** | Customer can view their cart | |
| **Requirements** | N/A | |
| **Pre-conditions** | Customer must log in before view cart | |
| **Post-conditions** | Success: Customer can view their cart | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor click on “Cart” icon. | 2. System shows cart and request cancel booking. |
| **Alternative flow** | 1. Actor clicks “x”– Close button- on table  2. Actor change quantity of a product  3. Actor clicks “x” – close button – in the right side of each product. | 1. System redirects to current page.  2. System removes this booking from cart. |

**CHECKOUT**

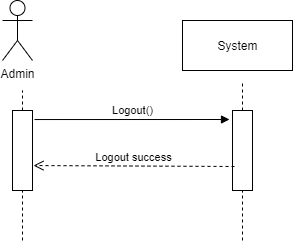
|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Check out and Place order | |
| **Actors** | Registered customer | |
| **Description** | Customer can order flights | |
| **Requirements** | Customer must has at least 1 flights in cart | |
| **Pre-conditions** | Customer must log in before adding flights | |
| **Post-conditions** | Success: Actor order successfully | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor click “Add to cart” button or “Cart” icon.  3. Actor clicks “Check out with” button.  5. Actor inputs order information then click “Place Order” button.  6. Actor inputs Card owner’s name, card number and security code if actor choose payment by credit card.  9. Actor views order’s status in Order page. | 2. System shows cart and added flights  4. System shows all flights in cart with calculated total price and show following controls:  - “Customer name” text field.  - “Phone” text field.  - “Address” text field.  - “Note” text field  - “Payment” table.  Systems shows Payment tab type according to type of payment actors choose.  5. System validates information.  6. System redirects to Order page and show order’s status. |
| **Alternative flow** | 1. Actor clicks “x”– Close button- on table  3. Actor clicks “X” button on each product.  5. Actors clicks “Home” button. | 1. System redirects to cart page..  2. System redirect to Home pages. |
| **Exceptions** | **Actor’s action** | **System’s response** |
| 1. Actor inputs more flights than there are in stock.  2. Actor left “Billing name” text field blank.  3. Actor left “Address” text field blank.  6. Actor left “Payment” dropdown blank.  7. Actor inputs invalid card information. | 8. System shows messages:  - “Flight out of stock”.  - “Order name must not be blank”.  “Address must not be blank”.  “Must choose a delivery date”.  “Must choose a delivery time frame”.  “Must choose a payment method”.  “Invalid card”. |

**VIEW ORDER HISTORY**

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | View order history | |
| **Actors** | Registered customer | |
| **Description** | Customer can view their order history | |
| **Requirements** | N/A | |
| **Pre-conditions** | Customer must log in to website | |
| **Post-conditions** | Success: Customer can view their order. | |
| Fail: Customer cannot see their order. | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses “Account” icon on Menu Bar  3. Actor chooses “Your row” on sides bar.  5. Actor selects the order that they want to view in “Purchase history” and clicks “View” Button. | 2. System redirects to Account page.  4. System redirects to “Your row” page and displays a list of existing orders and their status.  6. System validates inputs by id of Order.  7. System displays all order’s detail and its status. |

**ADMIN**

**LOGIN**

****

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Login | |
| **Actor** | Admin | |
| **Description** | When admin wants to manage Flights, news, views orders and orders detail | |
| **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 Admin-Acount and Password, then click the ‘Login’ button. | 2. System redirects to Login Page with the following controls:  - ' Admin-Acount' text field  - 'Password' text field  - 'Login' button  4. System validates the information.  5. 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:  - ‘ Admin-Acount ’ text field  - ‘Password’ text field  - ‘Login’ button. |

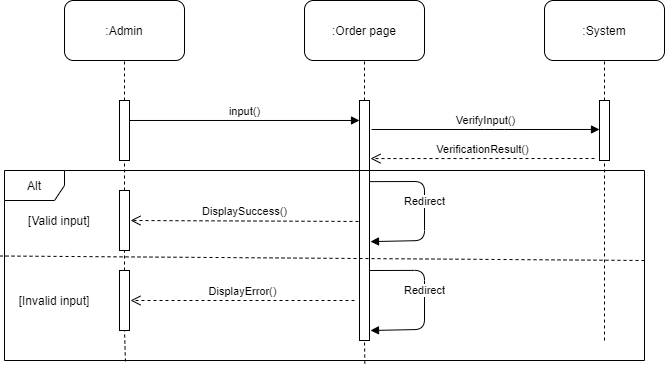
**Manage Customer**

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Manage Customer | |
| **Actor** | Admin | |
| **Description** | When admin wants to control Status customer ‘s cart | |
| **Requirements** | Admin provides New content | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: News content is added into the website | |
| Fail: admin can’t create | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor clicks ‘Edit-icon’ button.  2. Actor inputs information to update customer ‘s cart | System displays a form with the following fields:  - id  - name  - user-account  - password  - email  - phone  - address  -status  Finally, click update to repair . |

**Manage Flights**

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Show Flights | |
| **Actor** | Admin | |
| **Description** | When admin wants to follow menu | |
| **Requirements** | Admin provides News Content | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: News is updated in the website | |
| Fail: admin can’t update | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor clicks ‘Manage Flights ’ button. | 2. System displays a form with the following fields:  - image  - code  - name ‘s product  - description  - price  - type |

**Order Management**

****

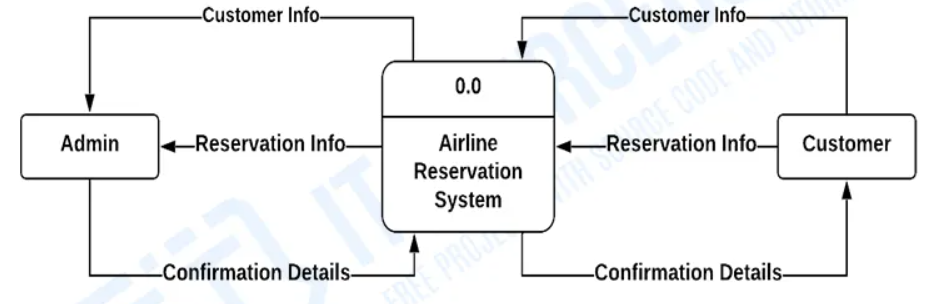
|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Check status ‘s cart | |
| **Actor** | Admin | |
| **Description** | When admin wants to Check status ‘s cart | |
| **Requirements** | Admin provides News Content | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: product is delete in the website | |
| Fail: product is not delete from the website | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor clicks ‘Order Management’ button. | 2. System displays list of cart .  3. System displays a form with the following fields:  - Code order  - Code Customer  - Date  - Price  - Status  - Action  4.Clik ‘ show ’ button to see more. |

**Employee management**

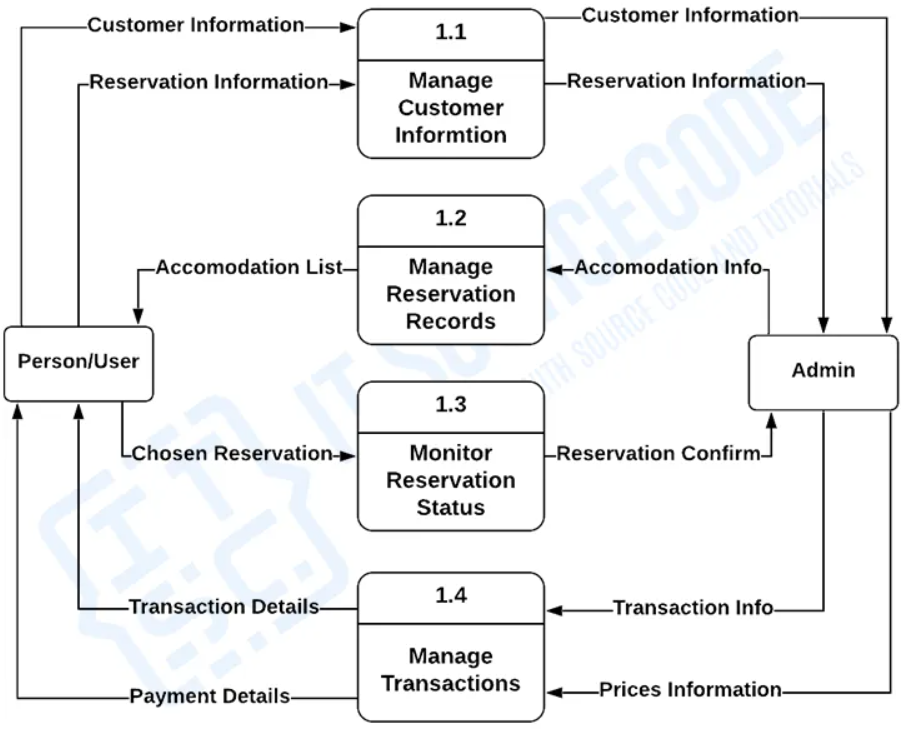
|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Empoyee manager | |
| **Actor** | Admin | |
| **Description** | When admin wants to position adjustment | |
| **Requirements** | Admin has at least 2 employee | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success : employee is shown in the website | |
| Fail: Page cannot see from the website | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses ‘Employee Manager’ on menu. | 2. System show Employee in the website.  3. System displays a form:  - avatar  - id employee  -fullname  -empaccount  - email  - phone  - Position  - Action  4.Clik ‘Edit-icon’ on action colum to update position ‘s employee |

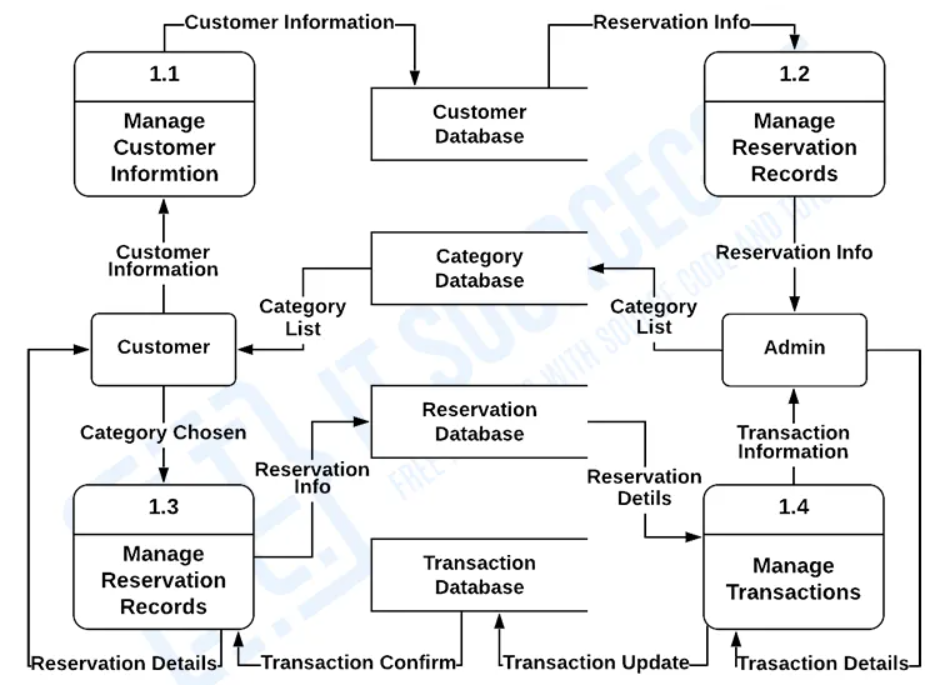
## DATA FLOW DIAGRAM (DFD)

**LEVEL 0**



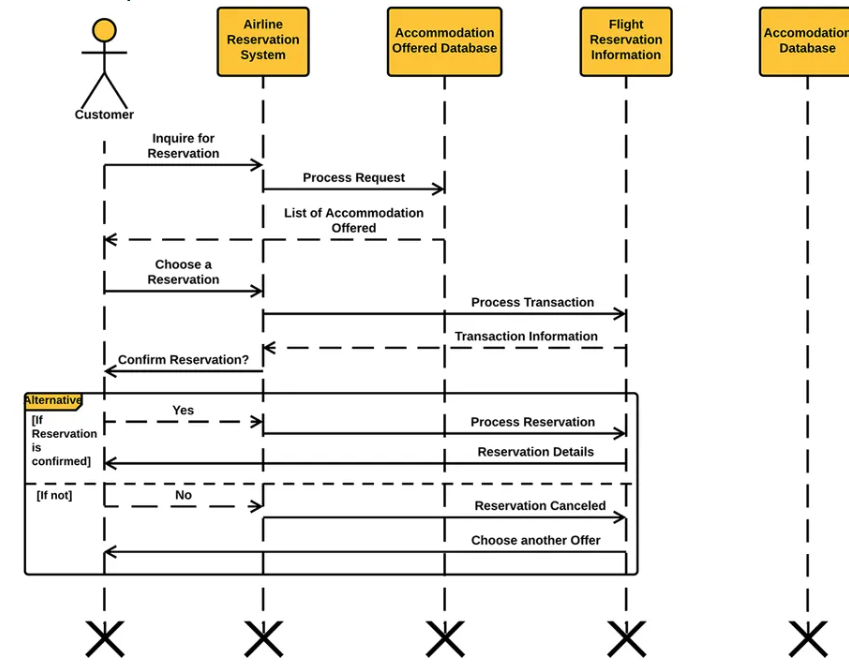
**LEVEL 1**



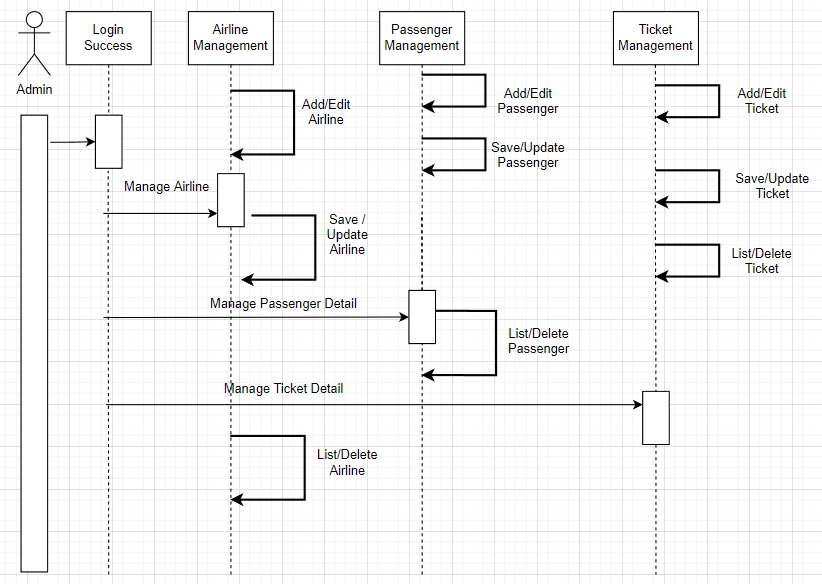


**Sequence Diagram for Airline Reservation System Design**

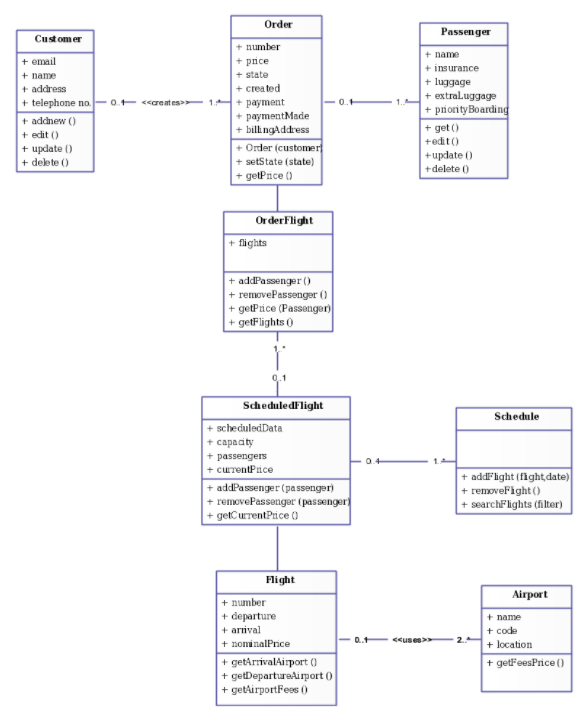
* **Customer:**



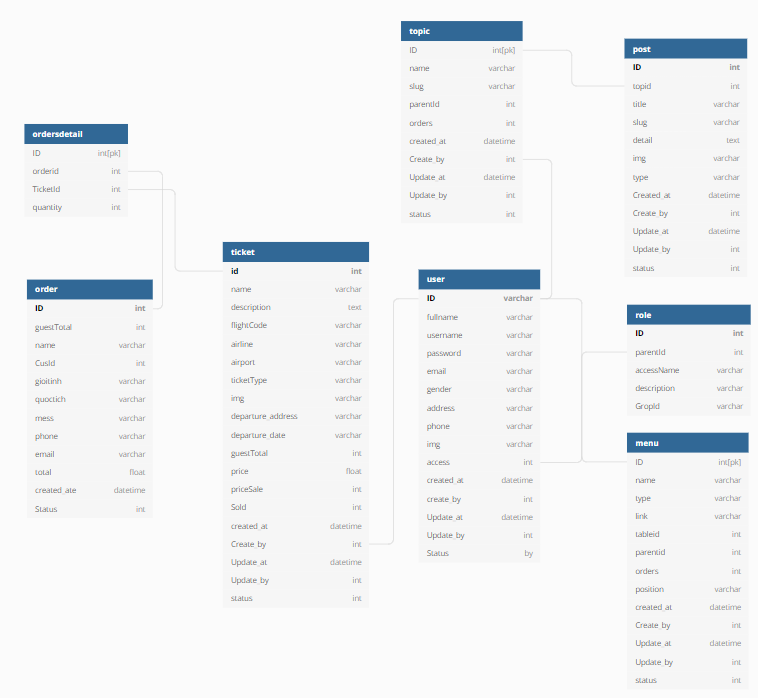
* **Admin:**



### Class diagram

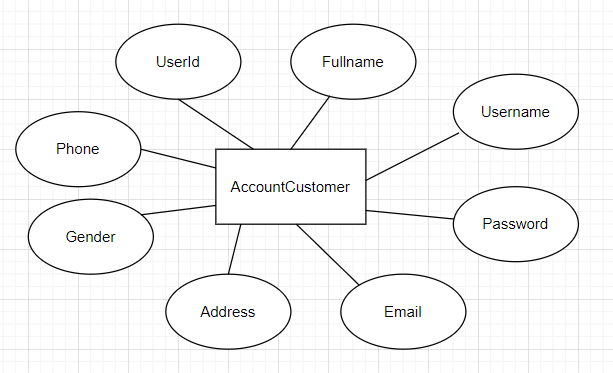


### ENTITY RELATIONSHIP DIAGRAM (ERD)

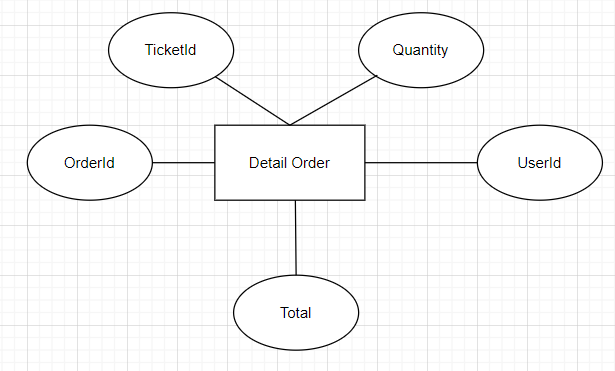


### ENTITY PROPERTIES

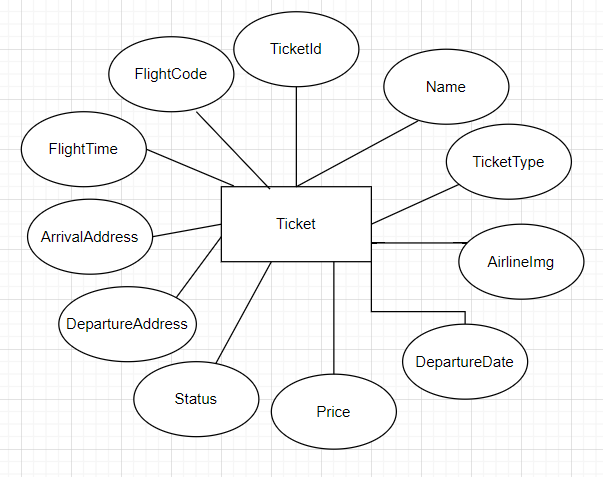
#### Customer



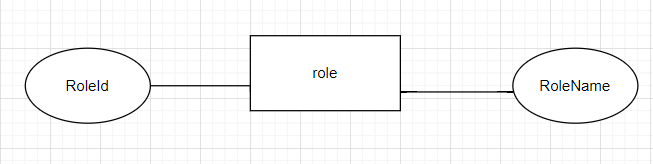
#### Detail\_Order



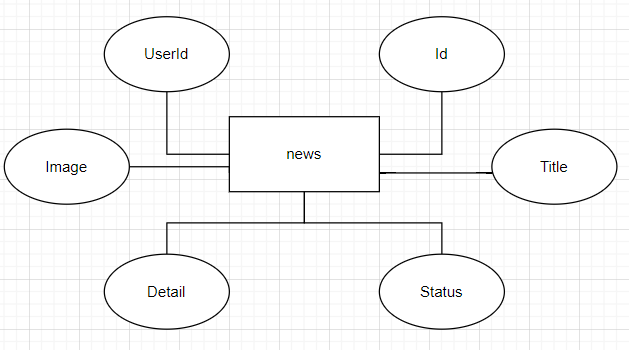
#### Ticket



#### Role



#### News



## TABLE DESIGN

**Ticket**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: Ticket** | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | TicketId | VARCHAR(225) | NOT NULL  PRIMARY KEY | Store the ID |
| 2 | Name | VARCHAR(225) | NOT NULL | Store the Name |
| 3 | TicketType | VARCHAR(225) | NOT NULL | Store the TicketType |
| 4 | AirlineImg | VARCHAR(225) | NOT NULL | Store the AirlineImg |
| 5 | DepartureDate | VARCHAR(225) | NOT NULL | Store DepartureDate |
| 6 | Price | float | NOT NULL | Store the Price |
| 7 | Status | int | NOT NULL | Store Status |
| 8 | DepartureAddress | VARCHAR(225) | NOT NULL | Store DepartureAddress |
| 9 | ArrivalAddress | VARCHAR(225) | NOT NULL | Store ArrivalAddress |
| 10 | FlightTime | int | NOT NULL | Store FlightTime |
| 11 | FlightCode | VARCHAR(225) | NOT NULL | Store FlightCode |
| 12 | Create\_at | VARCHAR(225) | NOT NULL | Store Create\_at |
| 13 | Create\_by | VARCHAR(225) | NOT NULL | Store Create\_by |
| 14 | Update\_at | VARCHAR(225) | NOT NULL | Store Update\_at |
| 15 | Update\_by | VARCHAR(225) | NOT NULL | Store Update\_by |
| 16 | UserId | VARCHAR(225) | NOT NULL | Store UserId |

**account**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: employee** | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | UserId | varchar(255) | NOT NULL  PRIMARY KEY | Store the ID. |
| 2 | Fullname | varchar(255) | NOT NULL | Store the Fullname |
| 3 | Username | varchar(255) | NOT NULL | Store the Username |
| 4 | Password | varchar(255) | NOT NULL | Stores the Password |
| 5 | Email | varchar(255) | NOT NULL | Stores the Email |
| 6 | Address | varchar | NOT NULL | Stores the Address |
| 7 | Phone | varchar | NOT NULL | Stores the phone |
| 8 | Gender | varchar(255) | NOT NULL | Stores gender |
| 9 | Image | varchar(255) | NOT NULL | Stores Image |
| 10 | Role | int | NOT NULL | Store Role |
| 11 | Create\_at | varchar(255) | NOT NULL | Store Create\_at |
| 12 | Create\_by | varchar(255) | NOT NULL | Store Create\_by |
| 13 | Update\_at | varchar(255) | NOT NULL | Store Update\_at |
| 14 | Update\_by | varchar(255) | NOT NULL | Store Update\_by |
| 15 | Status | int | NOT NULL | Store Status |
| 16 | Passport | varchar(255) | NOT NULL | Store Passport |
| 17 | Birthday | varchar(255) | NOT NULL | Store Birthday |

**post**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: post** | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | Id | Int | NOT NULL  PRIMARY KEY | Store the ID |
| 2 | Title | varchar(255) | NOT NULL  FOREIGN KEY | Store the title |
| 3 | Create\_at | varchar(255) | NOT NULL | Store the Create\_at |
| 4 | Create\_by | varchar(255) | NOT NULL | Store the Create\_by |
| 5 | Update\_at | varchar(255) | NOT NULL | Store the Update\_at |
| 6 | Update\_by | varchar(255) | NOT NULL | Store the Update\_by |
| 7 | Status | varchar(255) | NOT NULL | Store the status |
| 8 | Detail | text | NOT NULL | Store the Detail |
| 9 | Image | nvarchar(255) | NOT NULL | Store the Image |
| 10 | UserId | varchar | NOT NULL | Store the UserId |

**order**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: order** | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | OrderId | Int | NOT NULL  PRIMARY KEY | Store the ID |
| 2 | TicketId | Varchar | NOT NULL | Store the |
| 3 | Quantity | int | NOT NULL | Store the quantity |
| 4 | UserId | varchar | NOT NULL | Store price of UserId |
| 5 | Total | float | NOT NULL | Store price total |

**role**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: role** | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | RoleId | Int | NOT NULL  PRIMARY KEY | Store the ID of the role |
| 2 | RoleName | varchar | NOT NULL | Store the name of role. |

## 4.TASK SHEET REVIEW 2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Project Ref. No.: Organic Product Store | | Project Name | Activity Plan Prepared By | Date of Preparation of Activity Plan: | | | |
| Sr./No. | Task | Actual Start Date | Actual Days | Team member name | Status |
| 1 | Use Case | Organic Product Store | All team | 2-Dec-21 | 3 | Tùng | Completed |
| 2 | Data Flow Diagram (DFD) | 3-Dec-21 | 2 | Hùng | Completed |
| 3 | Entity Relationship Diagram (ERD) | 4-Dec-21 | 3 | Nhân | Completed |
| 4 | Entity Properties | 5-Dec-21 | 2 | Long | Completed |
| 5 | Table Design | 6-Dec-21 | 1 | Hùng | Completed |
| 6 | Class Diagram | 7-Dec | 1 | Tùng | Completed |
| 7 | Sequence Diagram | 7-Dec-21 | 1 | Nhân | Completed |

|  |  |
| --- | --- |
| Date: 8 Dec 2021 | |
| Signature of Instructor:  Mr. Hoang Duc Quang | Signature of Team Leader:  Nguyễn Văn Nhân |