# Report No. 4 Software Design Description

## Design Overview

* The architectural design describes the overall architecture of the system and the architecture of each main component and subsystem.
* The detailed design describes static and dynamic structure for each component and functions. It includes class diagrams, class explanations and sequence diagrams for each use cases.
* The database design describes the relationships between entities and details of each entity.
* Document overview:
* Section 2: gives an overall description of the system architecture design.
* Section 3: gives component diagrams that describe the connection and integration of the system.
* Section 4: gives the detail design description, which includes class diagram, class explanation, and sequence diagram to details the application functions.
* Section 5: gives the interface design description, which includes component interface, web application interface, and mobile application design.
* Section 6: describe a fully attributed Entity Relationship Diagram.
* Section 7: describe the algorithms that apply in the system.

## System Architecture Design

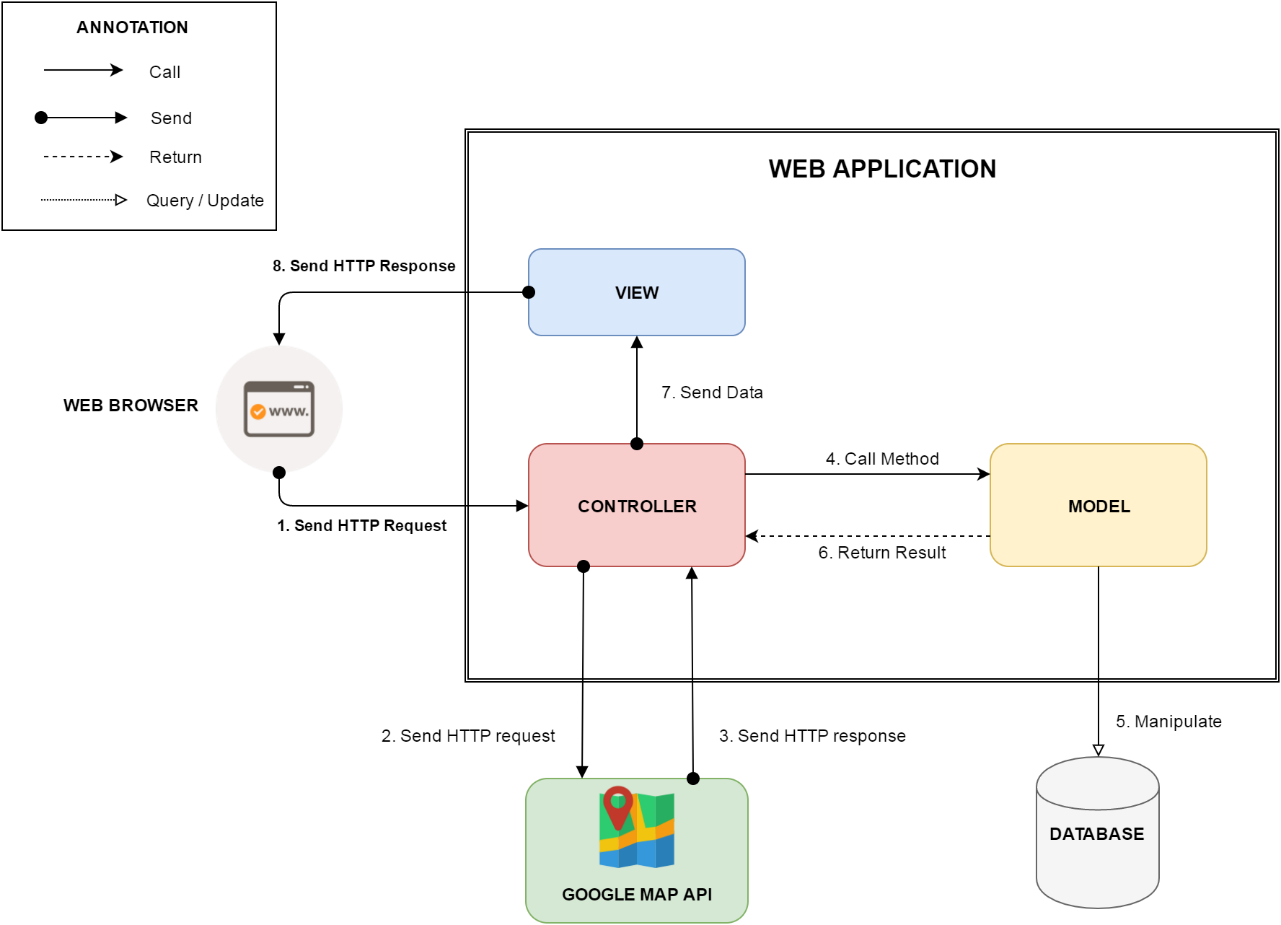


Figure 1: Web Application Architecture

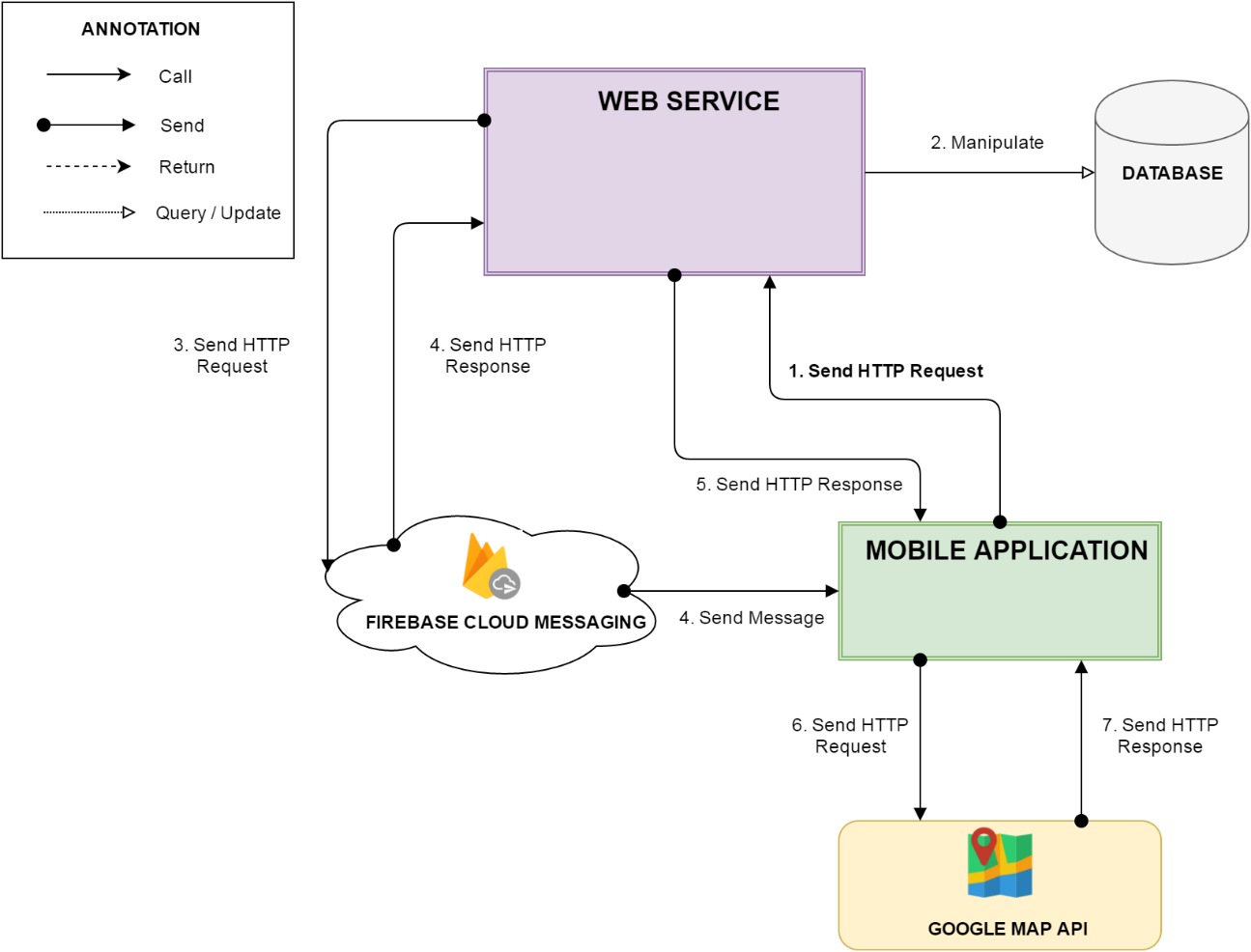


Figure 2: Mobile Application Architecture

### Web Application Architecture Description

In Web Application, the system is developed under MVC architecture style. We choose this architecture for Web application because of following advantages:

* Web application contains Web services with MVC architecture, we can separate business code with Controller and View. So we can use the business code in web service without repeat the code.
* We can organize the code better for maintainability, extensibility, reusability so we can expand the scope to other kind of illnesses such as flu, fever…
* In scope of 3-members team, MVC architecture makes it easier to split the big project into small modules and make it easier to assign each module for members in our team.

### Android Application Architecture Description

In our Android application, the application is developed under MVC architecture style. We choose this architecture for Android application because of following advantages:

* We can organize the code better for maintainability, extensibility and reusability.

## Component Diagram



Figure 3: Component Diagram

|  |  |
| --- | --- |
| Component Dictionary: Describes components | |
| Mobile application | Mobile application package |
| Web service | Provide API for mobile application to interact with the system. |
| Service | Common component is used to handle system’s business operations. |
| Admin Component | Component to handler admin activities in the system |
| Passenger Component | Component to handler passenger activities in the system |
| Staff Component | Component to handler staff activities in the system |
| Manager Component | Component to handler manager activities in the system |
| Repository | Component is used to handle interaction between the system and database. |
| Auto Handler Component | Component is used to handle scheduler in the system |
| Model | Entity framework |
| Google Firebase Service | Handle push notification with Google Firebase Message |

Table 1: Component Dictionary

## Detailed Description



### Class Diagram



Figure 4: Class Diagram

|  |  |  |
| --- | --- | --- |
| **Class Dictionary: Describe Classes** | | |
| **Class name** | **Mapping column with conceptual diagram** | **Description** |
| Subscription | Subscription | Contains the information of subscription. |
| User Subscription | Subscription | Contain the subscription information of user who subscribe an offer. |
| User | User | Contains the information of user. |
| Card | Card | Contains the information of card. |
| Ticket | Ticket | Contains the information of ticket. |
| Ticket Type | Ticket Type | Contains the information of ticket type. |
| Bus Route | Bus Route | Contains the information of bus route. |
| Credit Plan | Credit Plan | Contain the the information of credit plan which is used to add credit to card in the system. |
| Payment Transaction | N/A | Contains transaction history of card. |
| Promotion | Promotion | Contains the information of promotion. |
| Scratch Card | Scratch Card | Contains the information of scratch card. |

Table 46: Describe class

### Class diagram explanation

* + 1. Ticket Type

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Id | Integer | Private | Unique identifier of a ticket type |
| Name | String | Private | Name of the ticket type |
| Price | Integer | Private | Price of the ticket type |
| Description | String | Private | Description of the ticket type |
| Status | Integer | Private | Status of the ticket type |

Table 47: Ticket Type Attributes

**Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| Getter | Attribute type | Public | Get attribute value |
| Setter | Void | Public | Set value of attribute |

Table 48: Ticket Type Methods

* + 1. Ticket

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Id | Integer | Private | Unique identifier of a ticket ticket |
| CardID | Integer | Private | Unique identififer of the NFC card |
| TicketTypeId | Integer | Private | Unique identifier of the ticket type |
| BusRouteId | Integer | Private | Unique identifier of the bus route |
| Total | Integer | Private | The ticket price |
| BoughtDated | DateTime | Private | The date time that ticket is bought |
| IsNoCard | Boolean | Private | Ticket is bought by card or cash |

Table 49: Ticket Attributes

**Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| Getter | Attribute type | Public | Get attribute value |
| Setter | Void | Public | Set value of attribute |

Table 50: Ticket Methods

1. Bus Route

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Id | Integer | Private | Unique identifier of a bus route |
| Code | String | Private | Unique code of the bus route |
| Name | String | Private | Name of the bus route |

Table 51: Bus Routes Attributes

**Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| Getter | Attribute type | Public | Get attribute value |
| Setter | Void | Public | Set value of attribute |

Table 52: Bus Routes Methods

1. Card

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Id | Integer | Private | Unique identifier of an card |
| UniqueIdentifier | String | Private | Manufacture UID NFC card |
| CardName | String | Private | Name of the NFC card |
| RegistrationDate | DateTime | Private | Date of registration of the NFC card |
| Balance | Double | Private | Balance of the NFC card |
| Status | Integer | Private | Status of the NFC card |
| UserId | Integer | Private | Indentifier of the NFC’card holder |
| DataVersion | Long | Private | The version of data on card |

Table 53: Card Attributes

**Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| Getter | Attribute type | Public | Get attribute value |
| Setter | Void | Public | Set value of attribute |

Table 54: Card Methods

1. Payment Transaction

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Id | Integer | Private | Unique identifier of a payment transaction |
| CardId | Integer | Private | Identifier of the NFC card |
| CreditPlanId | Date | Private | Identifier of the credit plan |
| TransactionId | String | Private | Identifier of the payment transaction |
| Total | Double | Private | Total cost of the payment transaction |
| PaymentDate | DateTime | Private | Date of payment |

Table 55: Payment Transaction Attribute

**Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| Getter | Attribute type | Public | Get attribute value |
| Setter | Void | Public | Set value of attribute |

Table 56: Payment Transaction Methods

1. User

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Id | Integer | Private | Unique identifier of a user |
| PhoneNumber | String | Private | User’s phone number |
| Password | String | Private | User account’s password |
| Fullname | String | Private | User’s fullname |
| Status | Integer | Private | User’s statatus |
| RoleId | Integer | Private | User’s role identifier |
| DeviceToken | String | Private | Firebase notification token of a passenger’s mobile. |
| MinBalance | Integer | Private | Minimum balance of passenger’s cards that the system will notify them. |

Table 57: User Attributes

**Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| Getter | Attribute type | Public | Get attribute value |
| Setter | Void | Public | Set value of attribute |

Table 58: User Methods

1. Promotion

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Id | Integer | Private | Unique identifier of a promotion |
| Name | String | Private | Name of the promotion |
| Description | String | Private | Description of the promotion |
| Status | Integer | Private | Status of the promotion |
| CreatedDate | DateTime | Private | Date of creation |
| ExpiredDate | DateTime | Private | Date of expiration |

Table 59: Promotion Attributes

**Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| Getter | Attribute type | Public | Get attribute value |
| Setter | Void | Public | Set value of attribute |

Table 60: Methods Methods

1. Credit Plan

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Id | Integer | Private | Unique identifier of a credit plan |
| Name | String | Private | Name of the credit plan |
| Price | Integer | Private | Price of the credit plan |
| Status | Integer | Private | Status of the credit plan |

Table 61: Credit Plan Attributes

**Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| Getter | Attribute type | Public | Get attribute value |
| Setter | Void | Public | Set value of attribute |

Table 62: Credit Plan Methods

1. Role

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Id | Integer | Private | Unique identifier of a role |
| Name | String | Private | Name of the role |

Table 63: Role Attributes

**Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| Getter | Attribute type | Public | Get attribute value |
| Setter | Void | Public | Set value of attribute |

Table 64: Role Methods

1. Scratch Card

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Id | Integer | Private | Unique identifier of a scratch card |
| Code | String | Private | Code of a scratch card |
| Price | Integer | Private | Price of of a scratch card |
| Status | Integer | Private | Status of a scratch card |
| ExpiredDate | DateTime | Private | Date of expiration |

Table 65: Scratch Card Attributes

**Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| Getter | Attribute type | Public | Get attribute value |
| Setter | Void | Public | Set value of attribute |

Table 66: Scratch Card Methods

1. Offer Subscription

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Id | Integer | Private | Unique identifier of an offer subscription |
| Name | String | Private | Name of an offer subscription |
| Code | String | Private | Code of an offer subscription |
| Price | Integer | Private | Price of an offer subscription |
| DiscountPercent | Integer | Private | Percentage of discount on each ticket of an offer subscription |
| TicketNumber | Integer | Private | Number discounted ticket of an offer subscription |
| Status | Integer | Private | Status of an offer subscription |

Table 67: Offer Subscription Attributes

**Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| Getter | Attribute type | Public | Get attribute value |
| Setter | Void | Public | Set value of attribute |

Table 68: Offer Subscription Methods

1. User Subscription

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Id | Integer | Private | Unique identifier of a user subscription |
| UserId | Integer | Private | Identifier of the user |
| SubscriptionId | Integer | Private | Identifier of the offer subscription |
| ExpiredDate | DateTime | Private | Date of expiration |
| IsActive | Boolean | Private | Auto extend the subscription or not |
| TicketRemaining | Integer | Private | Number discounted ticket remaining. |

Table 69: User Subscription Attributes

**Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| Getter | Attribute type | Public | Get attribute value |
| Setter | Void | Public | Set value of attribute |

Table 70: User Subscription Methods

### Interactive Diagram

* + 1. **Passenger**
       1. Add credit

Summary: This diagram shows process of passenger toping-up his/her card.

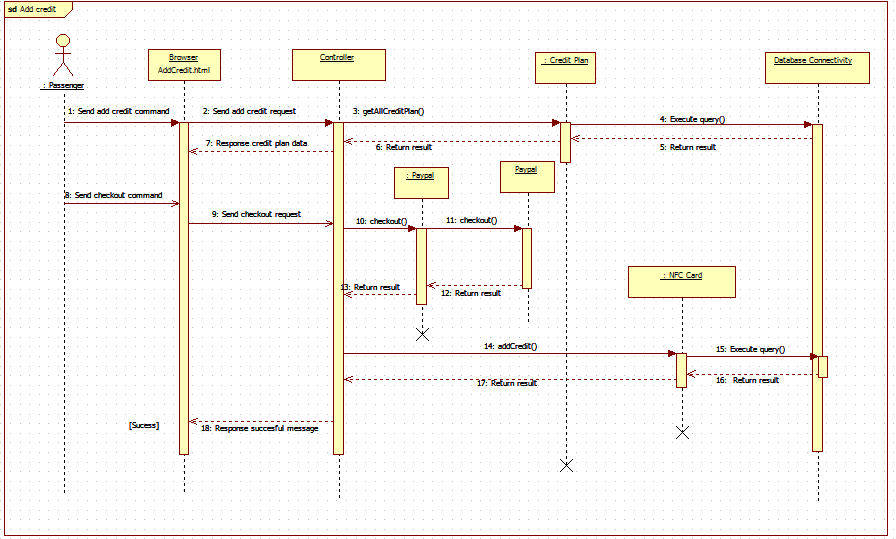


Figure 5: <Sequence Diagram> Add Credit

* + - 1. Get Cards

Summary: This diagram shows process of passenger getting his/her card information.

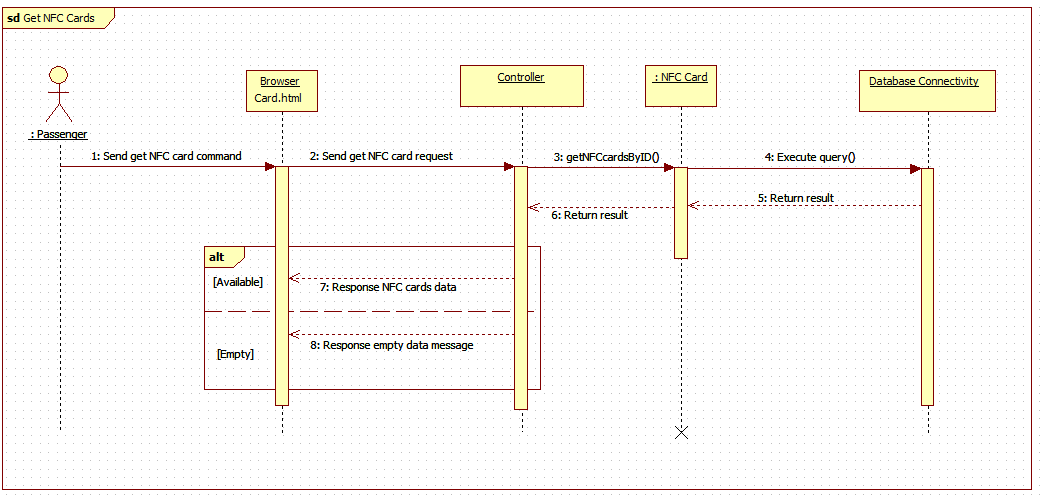


Figure 6: <Sequence Diagram> Get Cards

* + - 1. Find bus

Summary: This diagram shows process of passenger finding a bus’s information

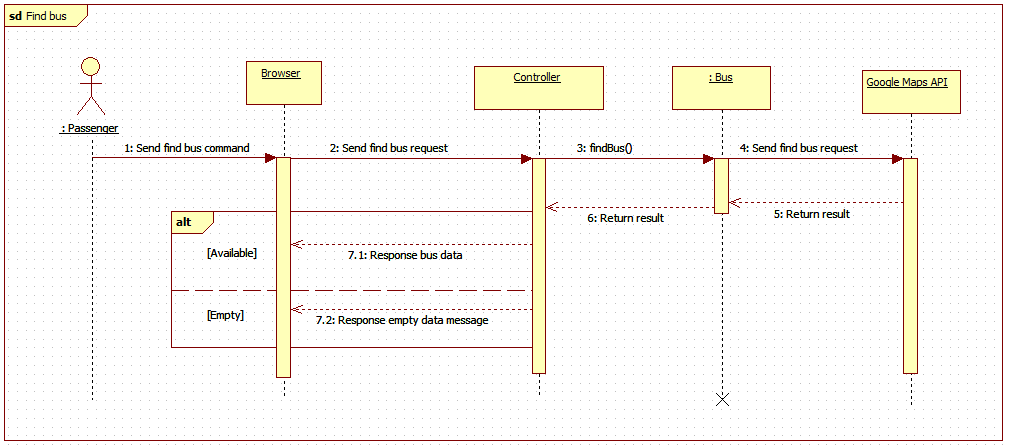


Figure 49: <Sequence Diagram> Find bus

* + - 1. Get outcome report

Summary: This diagram shows process of passenger getting his/her outcome report.

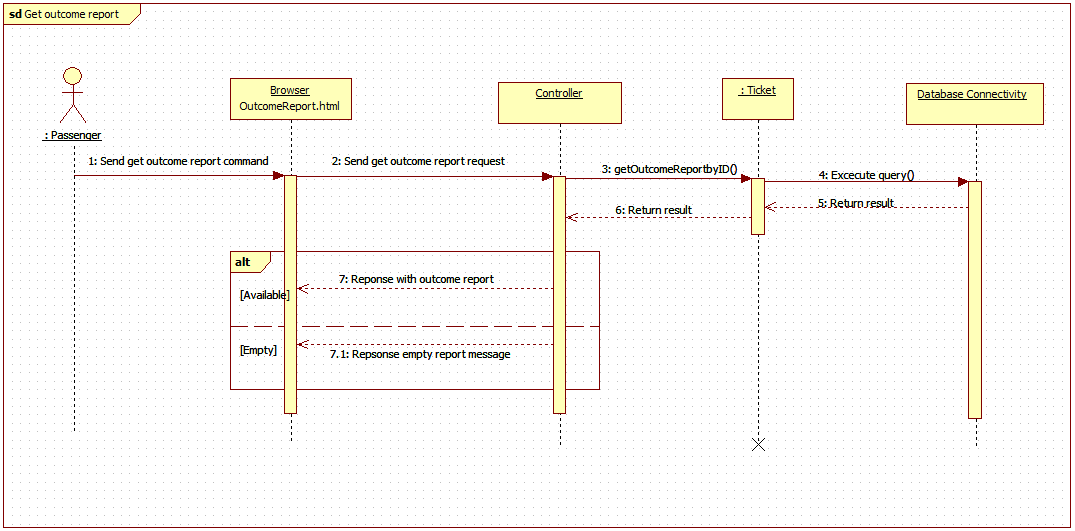


Figure 49: <Sequence Diagram> Get Income Report

* + 1. **Staff**

1. Edit Passenger

Summary: This diagram shows process of staff editing passenger information.

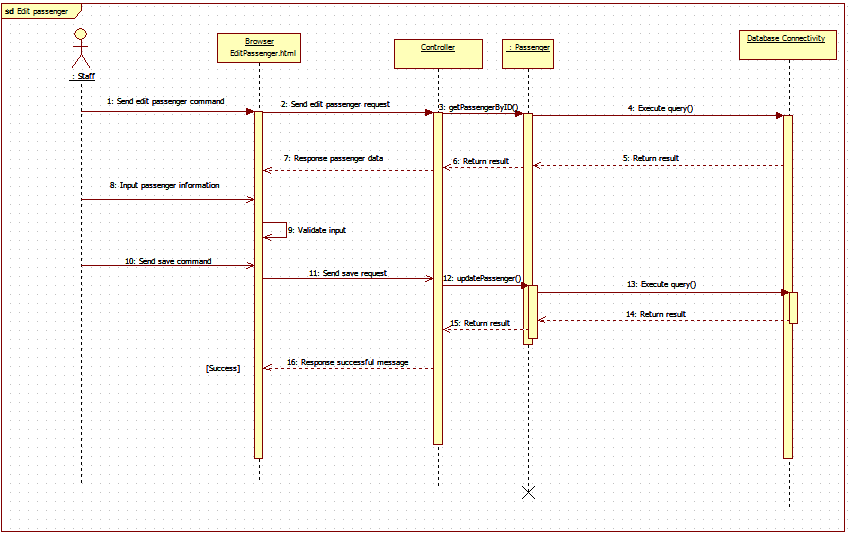


Figure 50: <Sequence Diagram> Edit Passenger

1. Activate/Deactivate Card

Summary: This diagram shows process of staff activating or deactivating passenger’s card

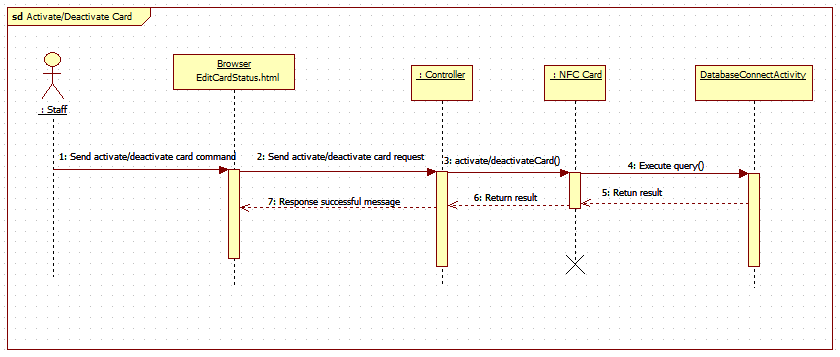


Figure 51: <Sequence Diagram> Activate/Deactivate Card

1. Add Card

Summary: This diagram shows process of staff adding new card

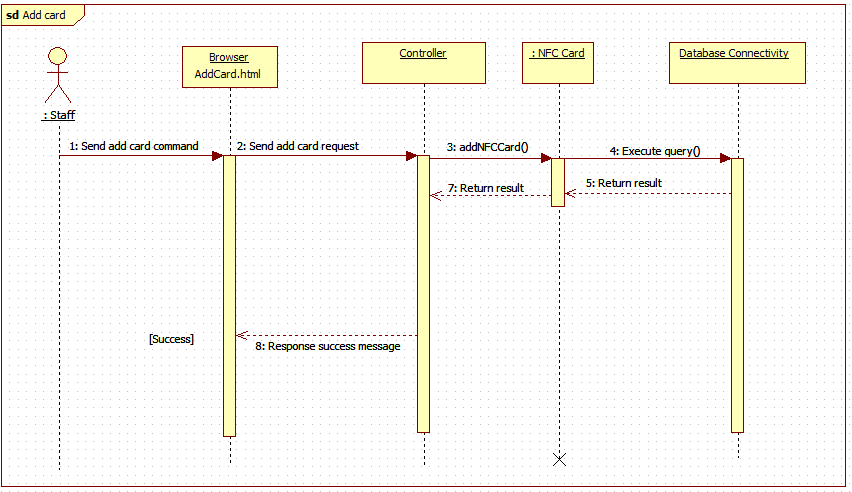


Figure 52: <Sequence Diagram>Add Card

* + 1. Manager

1. Get Income Report

Summary: This diagram shows process of manager getting income report

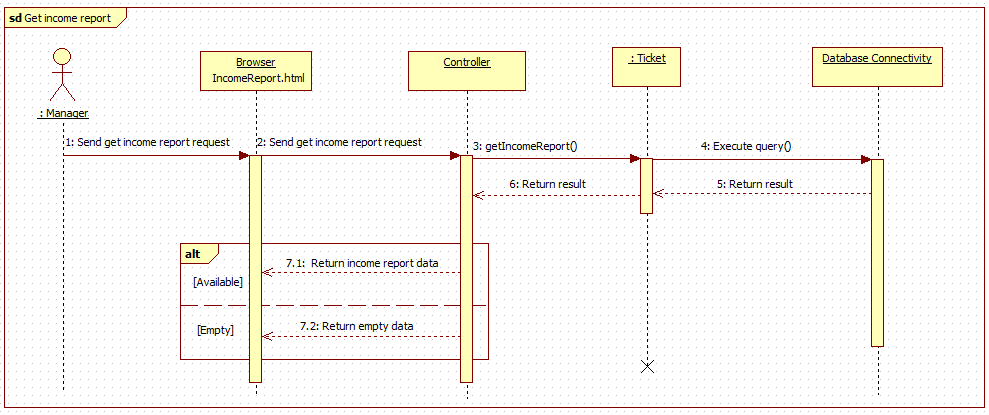


Figure 53: <Sequence Diagram>Get Outcome Report

1. Create promotion

Summary: This diagram shows process of manger creating a promotion

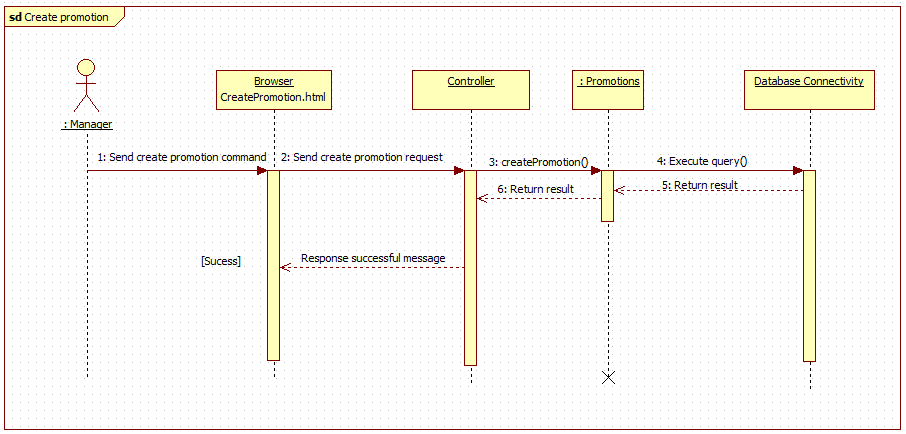


Figure 54: <Sequence Diagram>Create Promotion

1. **Mobile Application**
   * + 1. Add credit

Summary: This diagram shows process of passenger toping-up his/her card via mobile application.

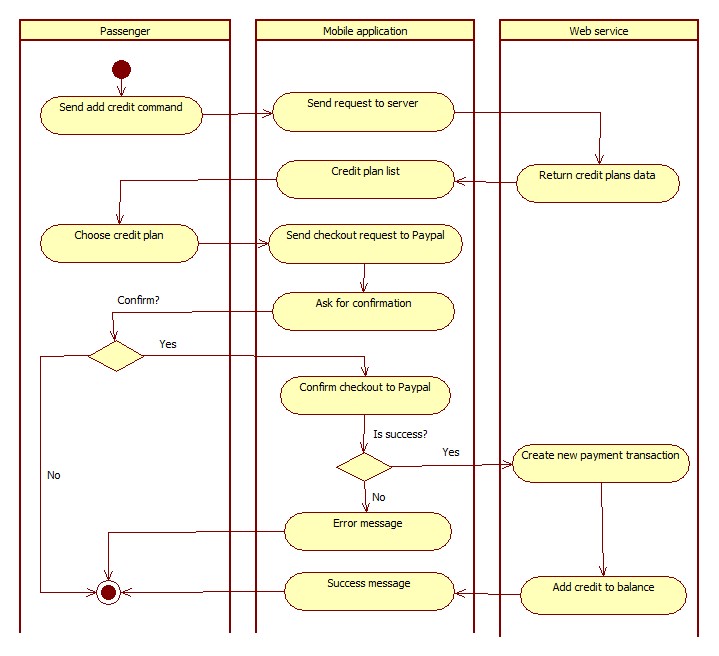


Figure 55: <Activity Diagram> Add Credit

* + - 1. Find Bus

Summary: This diagram shows process of passenger finding a bus route information via mobile application.

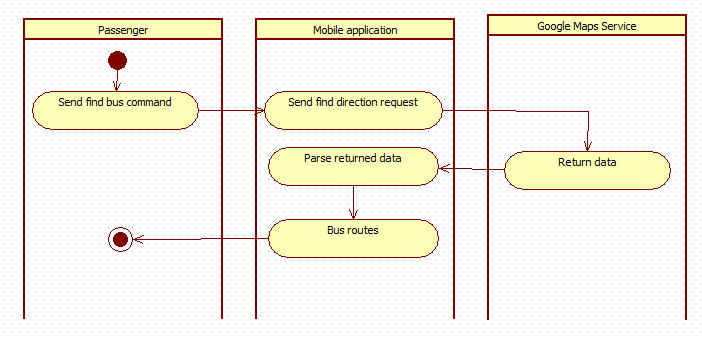


Figure 56: <Activity Diagram>Find Bus

* + - 1. Get Outcome Report

Summary: This diagram shows process of passenger getting his/her outcome report via mobile application.

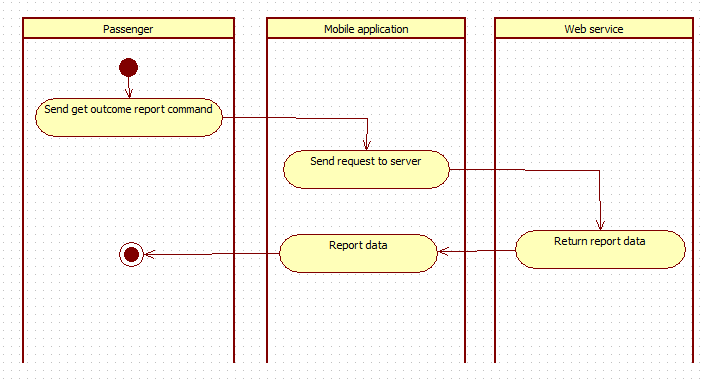


Figure 57: <Activity Diagram> Get Outcome Report

1. Send Notification

Summary: This diagram shows process of sending notification from system to passenger’s mobile phone.



Figure 58: <Activity Diagram> Send Notification

## User Interface

### Component Interface

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Signature | Description | Input | Output | Output format | Exception |
| /Api/PushOfflineData | Push offline ticket from emulator to server after connection lost. | key: string  cardId: string  ticketTypeId: int  routeCode: string  boughtDate: string | Json string of message object | Json | JsonException |
| /Api/GetBusStop | Get all bus stops of a bus route. | key: string  routeCode: string | Json string of message object | Json | JsonException |
| /Api/AddCard | Add new NFC card to the system | key: string  cardId: string | Json string of medical record object | Json | JsonException |
| /Api/ChangeCardName | Change name of a card | key: string  cardId: string  name: string | Json string of message object | Json | JsonException |
| /Api/Topup | Add credit to card by scratch card | key: string  cardId: string  code: string | Json string of pet object | Json | JsonException |
| /Api/GetCardInfo | Get information of a card | key: string  cardId: string | Json string of pet object | Json | JsonException |
| /Api/AddCardBalance | Add credit to card by Paypal | key: string  cardId: string  creditPlanId: int  transactionId: string | Json string of pet object | Json | JsonException |
| /Api/GetProfile | Get profile of an account | key: string  phone: string | Json string of pet object | Json | JsonException |
| /Api/GetAllCreditPlan | Get all credit plans information | key: string | Json string of pet object | Json | JsonException |
| /Api/GetAllTicketType | Get all ticket type information | key: string | Json string of pet object | Json | JsonException |
| /Api/ GetBusRouteByCode | Get a bus route details information by route code | key: string  routeCode: string | Json string of pet object | Json | JsonException |
| /Api/ GetAllBusRoutes | Get all bus routes information | key: string | Json string of pet object | Json | JsonException |
| /Api/ SellTicket | Sell ticket to emulator by card | key: string  cardId: string  ticketTypeId: int  routeCode: string  currentBalance: long  dataVersion: long | Json string of pet object | Json | JsonException |
| /Api/ RequestSellTicketToken | Get current server token | key: string | Json string of pet object | Json | JsonException |
| /Api/ SellTicketWithToken | Sell ticket to emulator by phone, server token is required | key: string  cardId: string  ticketTypeId: int  routeCode: string  currentBalance: long  dataVersion: long  token: string | Json string of pet object | Json | JsonException |
| /Api/ Login | Login to the system | key: string  phone: string  password: string | Json string of pet object | Json | JsonException |
| /Api/ UpdateProfile | Update account profile | key: string  phone: string  password: string  fullname: string  minBalance: int | Json string of pet object | Json | JsonException |
| /Api/ ActivateAccountByApp | Activate account by mobile application | key: string  phone: string  cardId: string | Json string of pet object | Json | JsonException |
| /Api/ GetAllCard | Get card list of an account | key: string  phone: string | Json string of pet object | Json | JsonException |
| /Api/ GetReport | Get outcome report | key: string  phone: string  beginDate: string  endDate: string | Json string of pet object | Json | JsonException |
| /Api/ RegisterNotificationToken | Register device token to get notification | key: string  phone: string  token: string | Json string of pet object | Json | JsonException |
| /Api/ GetPromotion | Get current available promotions | key: string | Json string of pet object | Json | JsonException |

Table 71: Component Interface

### 5.2 Web Application Design

5.2.1 Unauthorized User

5.2.1.1 Login



**Fields**

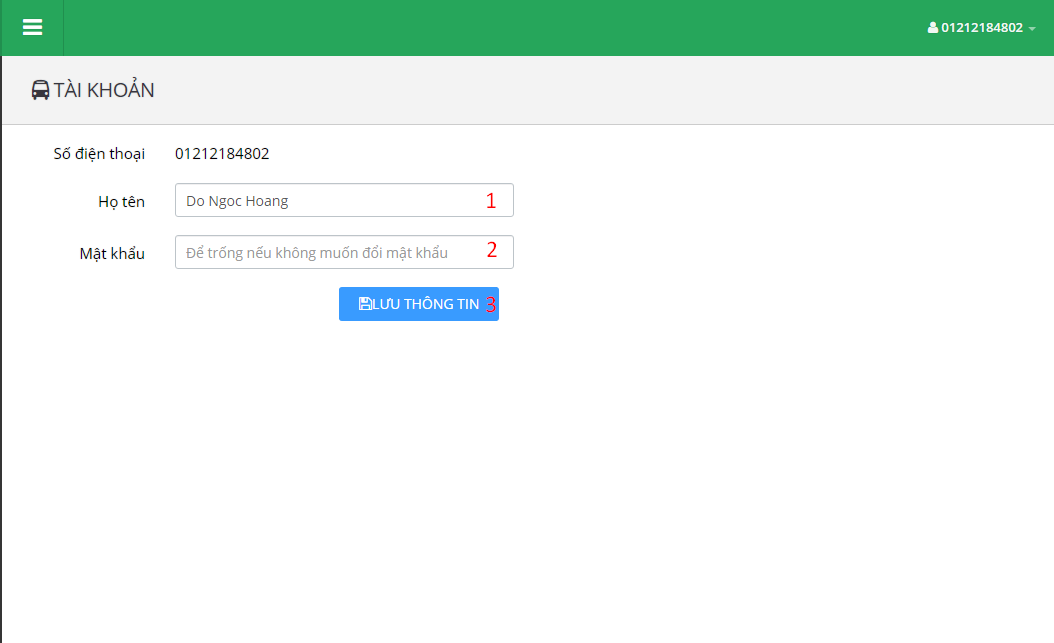
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | Phone | Account phone number | No | Yes | Textbox | String | 9-13 characters |
| 2 | Password | Account password | No | Yes | Textbox | String | 6-50 characters |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 3 | Login | Send login command | Validate PhoneNumber and Password | Open specific page with specific account |

5.2.2 Authorized User

5.2.2.1 Update profile



**Fields**

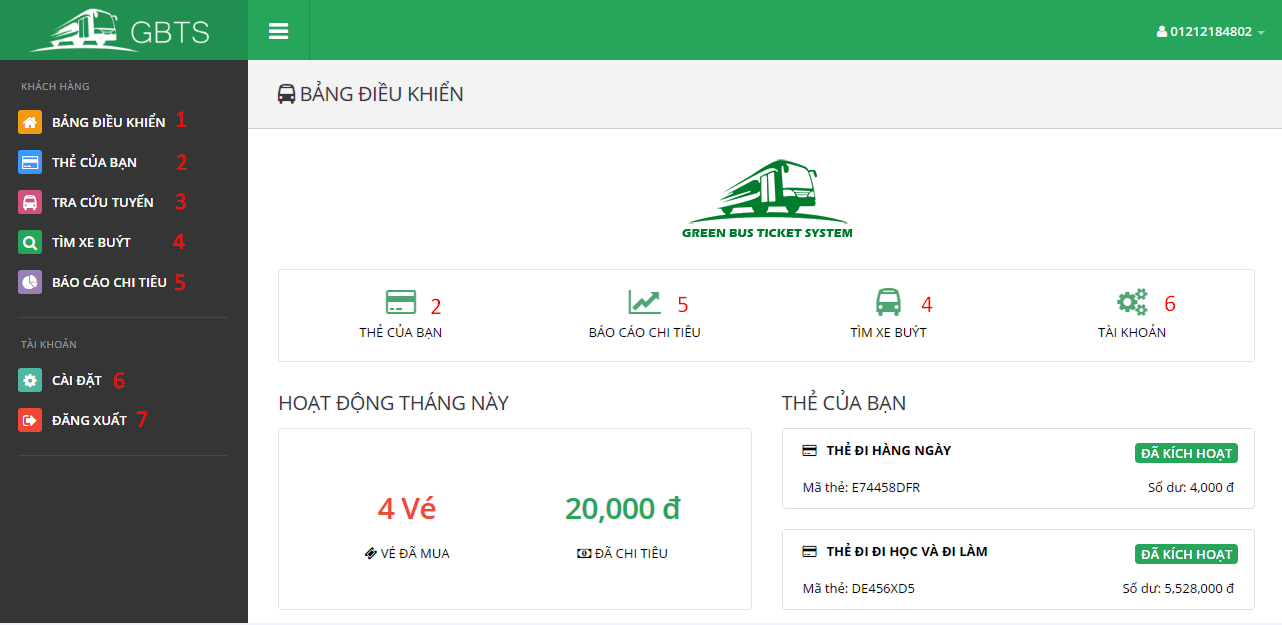
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | Fullname | Account full name | No | No | Textbox | String | 0-250 characters |
| 2 | Password | Account password | No | No | Textbox | String | 6-50 characters |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 3 | UpdateProfile | Send update profile command | Validate Fullname and Password | Display success message. |

5.2.3 Passenger

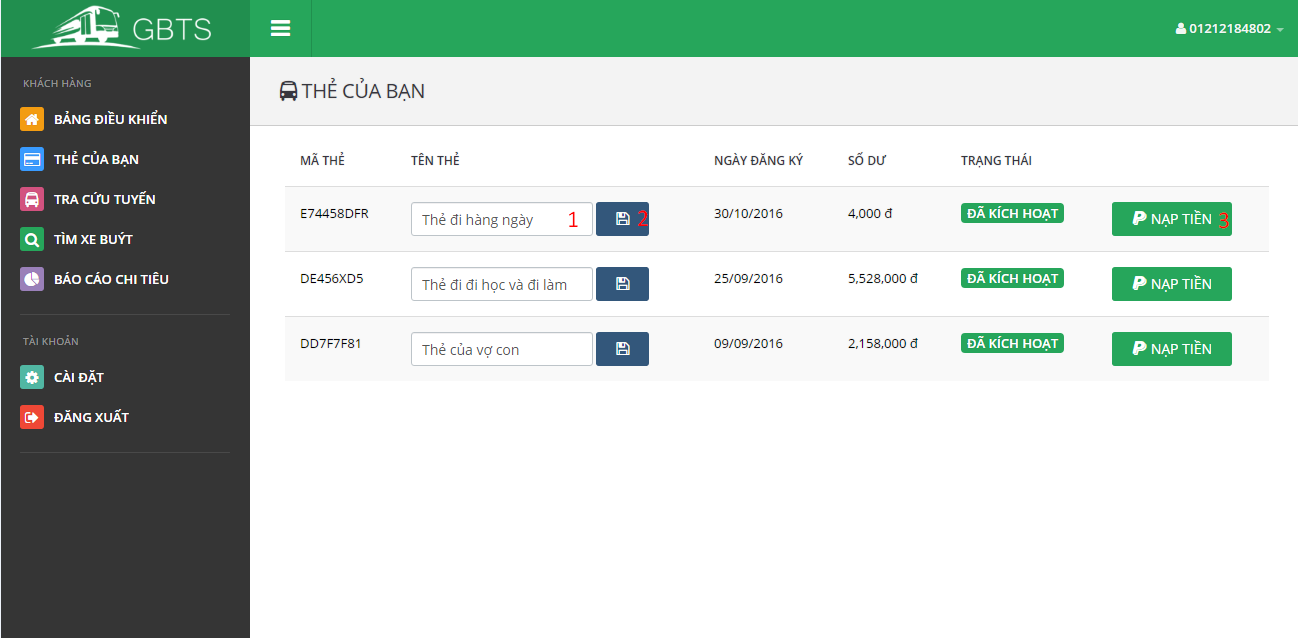
5.2.3.1 Dashboard



**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 1 | ViewDashboard | View dashboard page | N/A | Open dashboard page |
| 2 | GetCards | Send get cards command | N/A | Open card list page |
| 3 | ViewLookupBusRoutePage | View lookup bus route page | N/A | Open lookup bus route page |
| 4 | ViewFindBusPage | View find bus page | N/A | Open find bus page |
| 5 | ViewOutcomeReportPage | View outcome report page | N/A | Open get outcome report page |
| 6 | ViewProfile | View profile page | N/A | Open profile page |
| 7 | Logout | Send logout command | N/A | Destroy current session and redirect to login page. |

5.2.3.2 Get cards



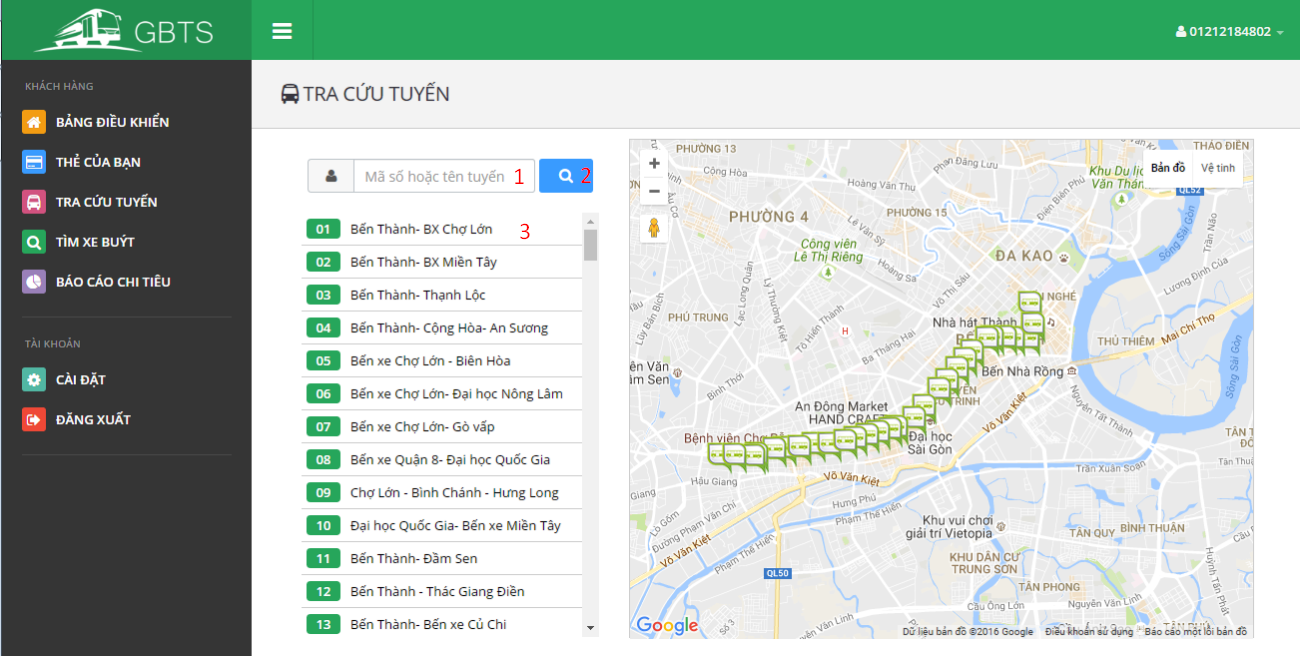
**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | CardName | Card name | No | No | Textbox | String | 0-250 characters |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 2 | UpdateCardName | Send update card name command | Validate CardName | Display success message. |
| 3 | AddCredit | Send add credit command | Validate card’s status. Disable button if card is deactivated. | Show add credit plan view. |

5.2.3.3 Lookup bus route



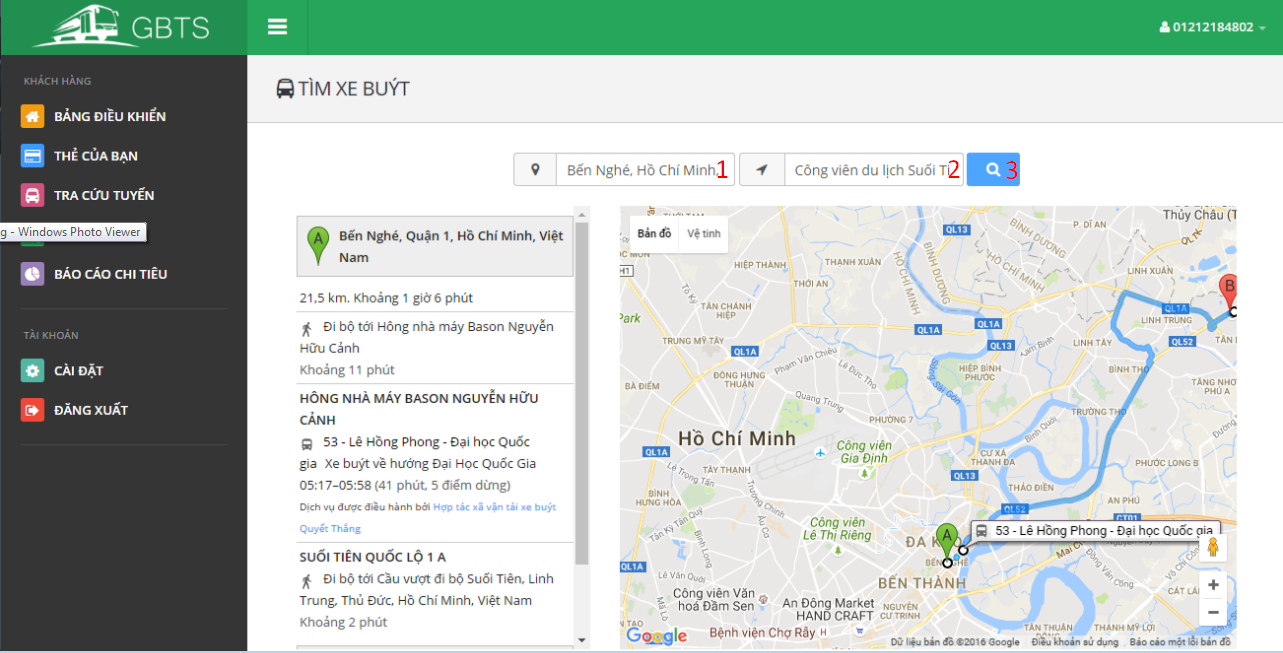
**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | Term | Route code or route name | No | No | Textbox | String | 0-250 characters |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 2 | Lookup | Send lookup bus route command | N/A | Display bus routes match searching term |
| 3 | ViewBusStop | Send view bus stop command | N/A | Display bus stops belong to selected term on map view. |

5.2.3.4 Find bus



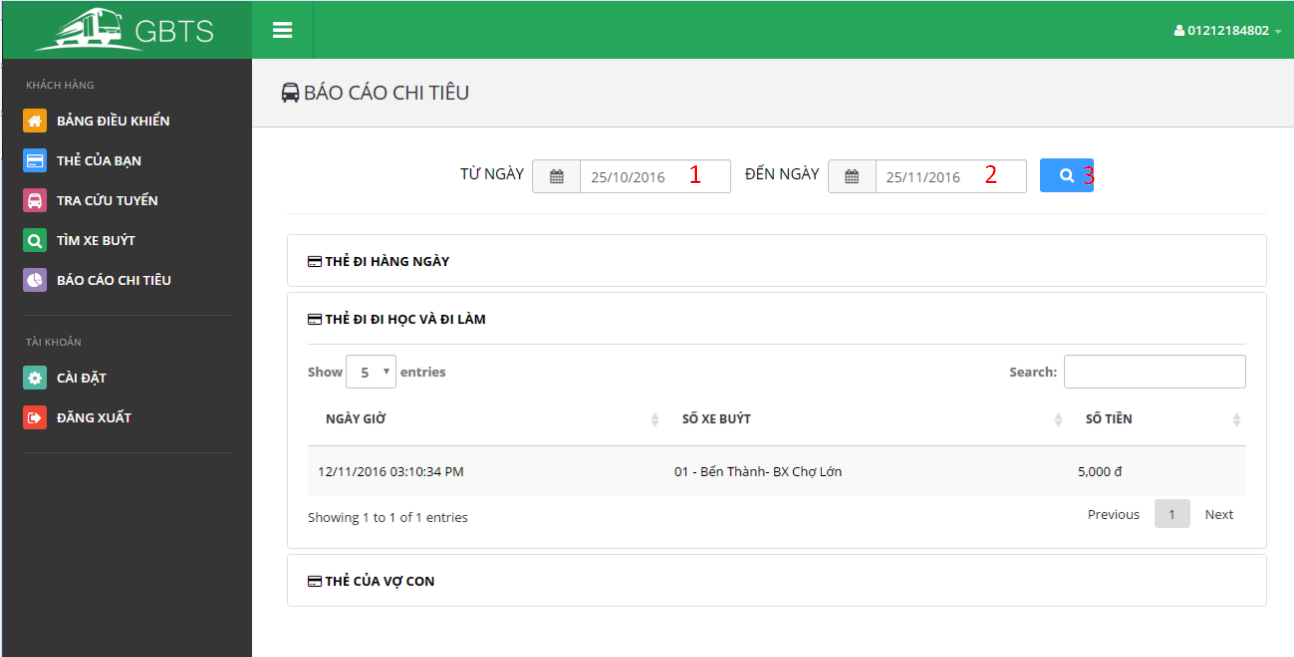
**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | StartingAddress | Starting address | No | No | Textbox | String | N/A |
| 2 | DestinationAddress | Destination address | No | No | Textbox | String | N/A |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 3 | GetRoute | Send get route command | N/A | Display bus routes match starting address and destination address on map view. |

5.2.3.4 Find bus



**Fields**

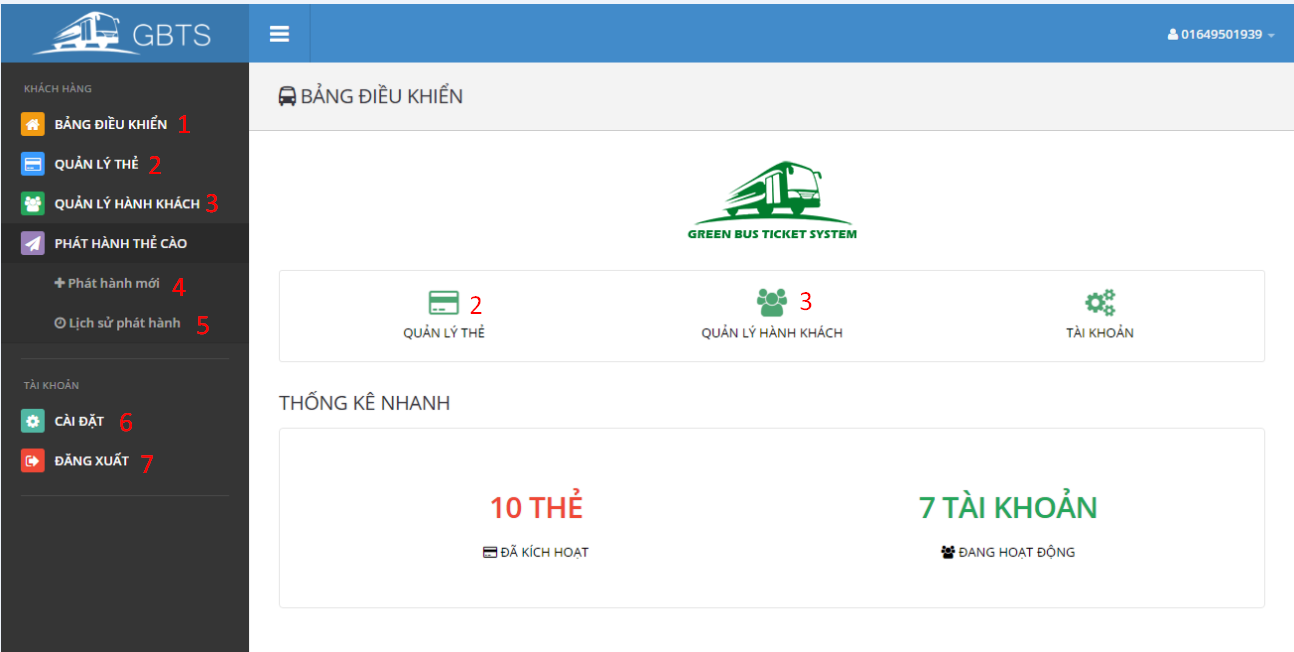
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | BeginDate | Begin date | No | No | Date select box | String | N/A |
| 2 | EndDate | End date | No | No | Date select box | String | N/A |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 3 | GetOutcomeReport | Send get outcome report command | N/A | Display outcome report according to selected begin date and end date. |

5.2.4 Staff

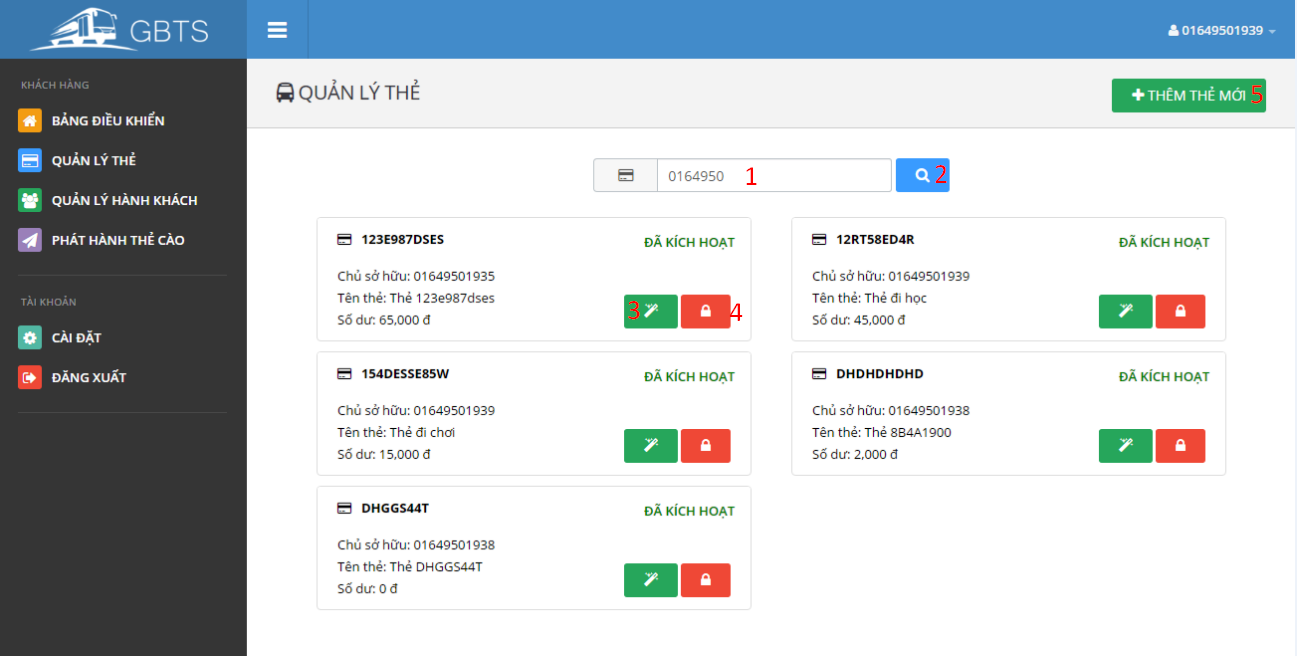
5.2.4.1 Dashboard



**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 1 | ViewDashboard | View dashboard page | N/A | Open dashboard page |
| 2 | ManageCards | View manage cards page | N/A | Open manage card page |
| 3 | ManagePassengers | View manage passengers page | N/A | Open manage passengers page |
| 4 | PublishScratchCard | Send publish scratch card command | N/A | Publish scratch card view is showed. |
| 5 | PublishHistory | Send publish history command | N/A | Publish history view is showed. |
| 6 | ViewProfile | View profile page | N/A | Open profile page |
| 7 | Logout | Send logout command | N/A | Destroy current session and redirect to login page. |

5.2.4.2 Manage Cards



**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | Term | Owner phone number or card UID | No | No | Textbox | String | N/A |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 2 | SearchCard | Send search card command | N/A | Display cards match searching term. |
| 3 | TransferCard | Send transfer card command | N/A | Transfer card view is showed. |
| 4 | Activate/DeactivateCard | Send activate/deactivate card command | N/A | Selected card is activated/deactivated |
| 5 | Add card | Send add card command | N/A | Add card view is showed. |

5.2.4.3 Manage Passengers



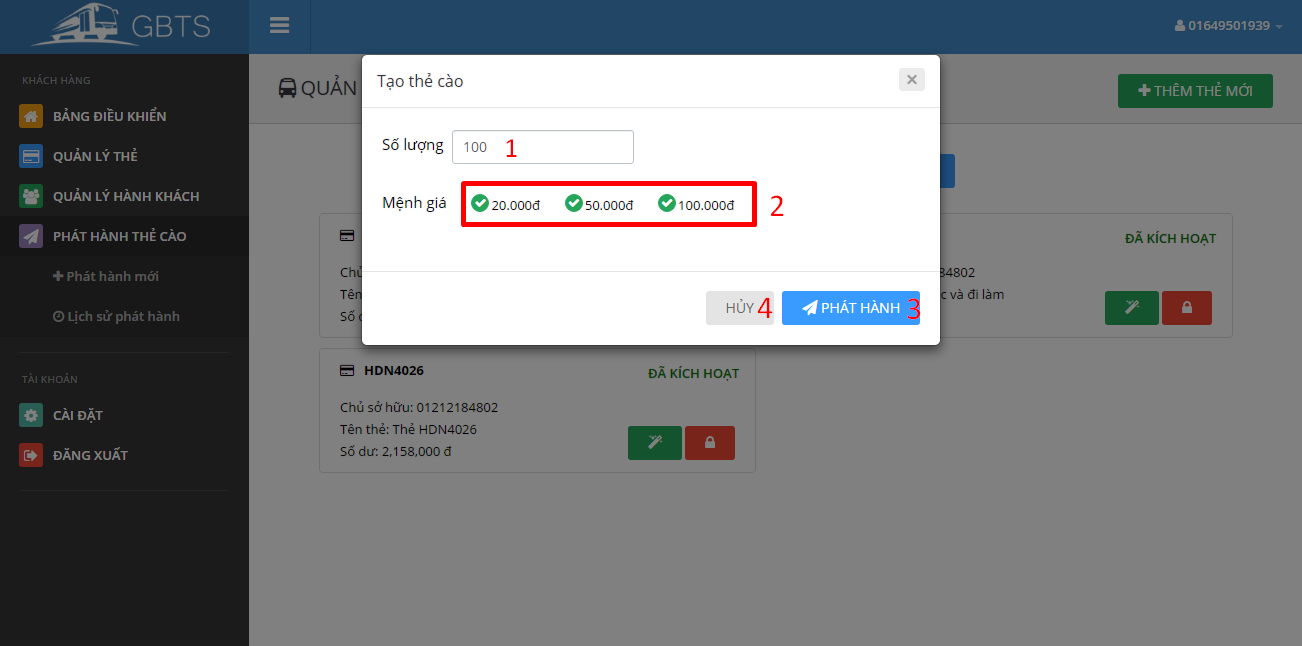
**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | Term | Passenger phone number or full name | No | No | Textbox | String | N/A |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 2 | SearchPassenger | Send search passenger command | N/A | Display passengers match searching term. |
| 3 | EditPassenger | Send edit passenger command | N/A | Edit passenger view is showed. |

5.2.4.4 Publish scratch card



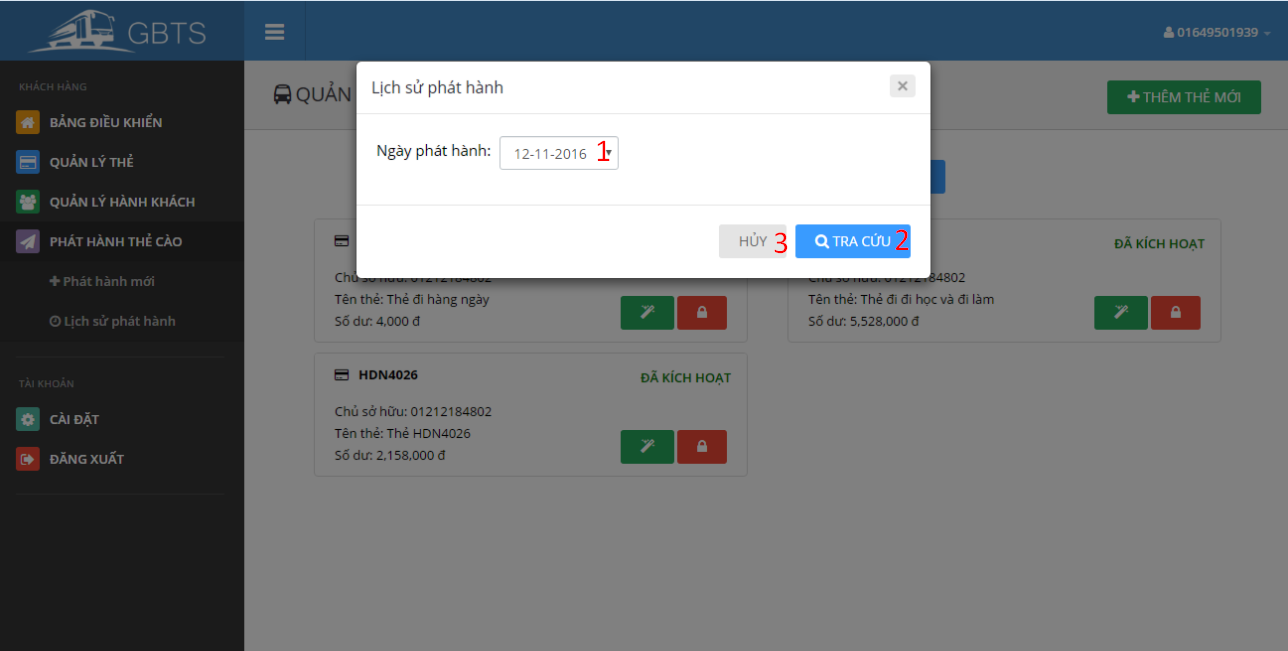
**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | Quantity | Scratch card quantity. | No | No | Textbox | Number | N/A |
| 2 | ScratchCardType | Type of scratch card | No | No | Checkbox | Number | N/A |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 3 | Publish | Send publish scratch card command | N/A | Open published scratch card list on a new page. |
| 4 | Cancel | Close publish scratch card view | N/A | Publish scratch card view is closed. |

5.2.4.5 Publish history



**Fields**

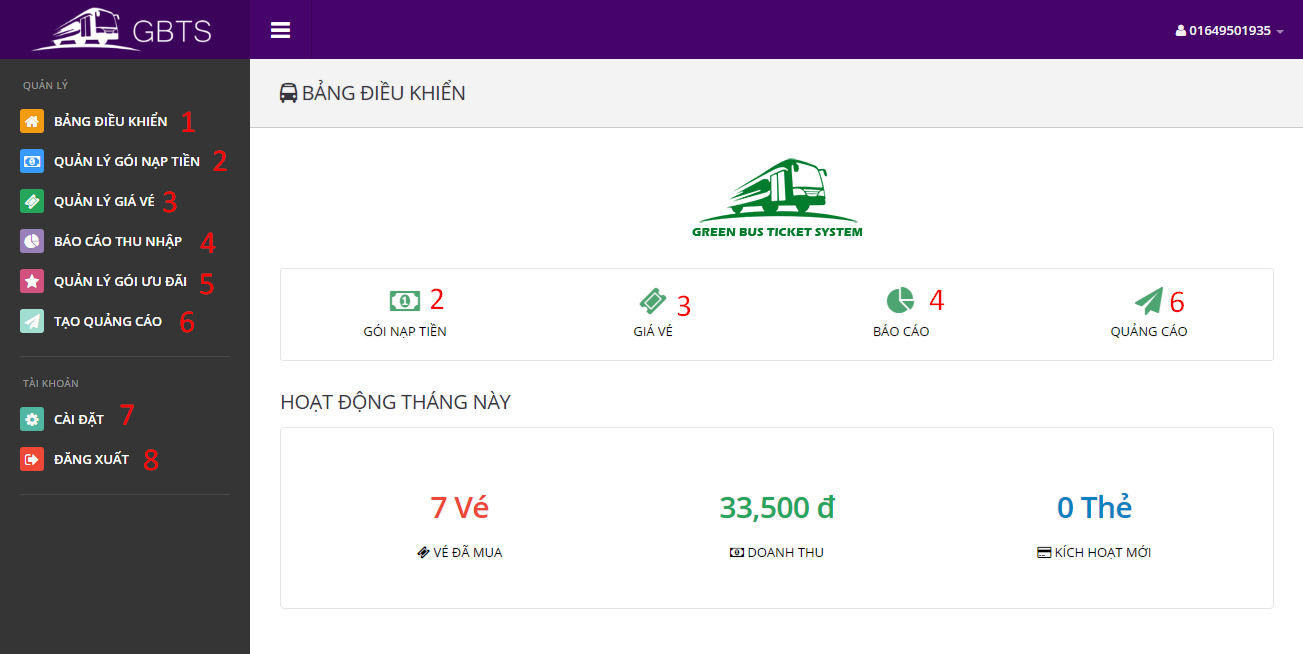
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | PublishedDate | Scratch card published dated | No | No | Selectbox | String | N/A |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 3 | GetPublishedHistory | Send get published history command | N/A | Open published scratch card list of selected published date on a new page. |
| 4 | Cancel | Close publish history view | N/A | Publish history view is closed. |

5.2.5 Manager

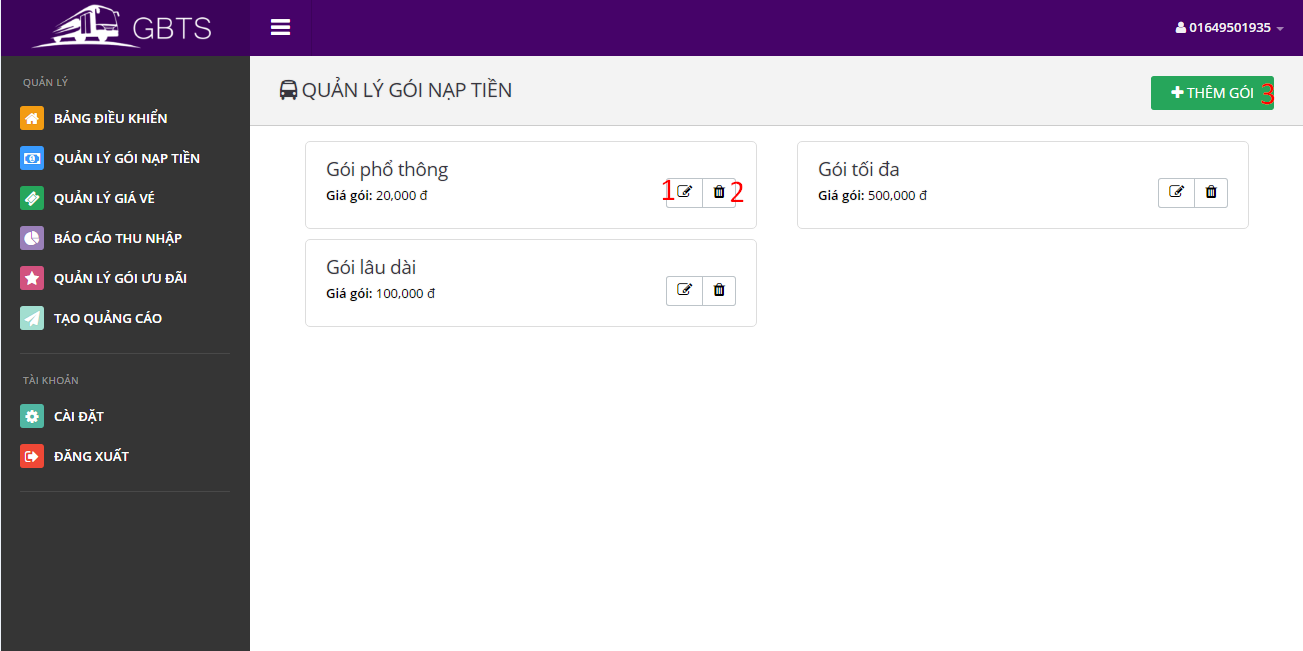
5.2.5.1 Dashboard



**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 1 | ViewDashboard | View dashboard page | N/A | Open dashboard page |
| 2 | ManageCreditPlans | View manage credit plans page | N/A | Open manage credit plans page |
| 3 | ManageTicketTypes | View manage ticket types page | N/A | Open manage ticket types page |
| 4 | ViewIncomeReportPage | View income report page | N/A | Open income report page |
| 5 | ManageSubscriptions | View manage subscriptions page | N/A | Open manage subscriptions page |
| 6 | CreatePromotionPage | View create promotion page | N/A | Open create promotion page |
| 7 | ViewProfile | View profile page | N/A | Open profile page |
| 8 | Logout | Send logout command | N/A | Destroy current session and redirect to login page. |

5.2.5.2 Manage Credit Plans



**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 1 | EditCreditPlan | Send edit credit plan command | N/A | Edit credit plan view is showed. |
| 2 | DeleteCreditPlan | Send delete credit plan command | N/A | A delete confirmation is show. |
| 3 | AddCreditPlan | Send add credit plan command | N/A | Add credit plan view is showed. |

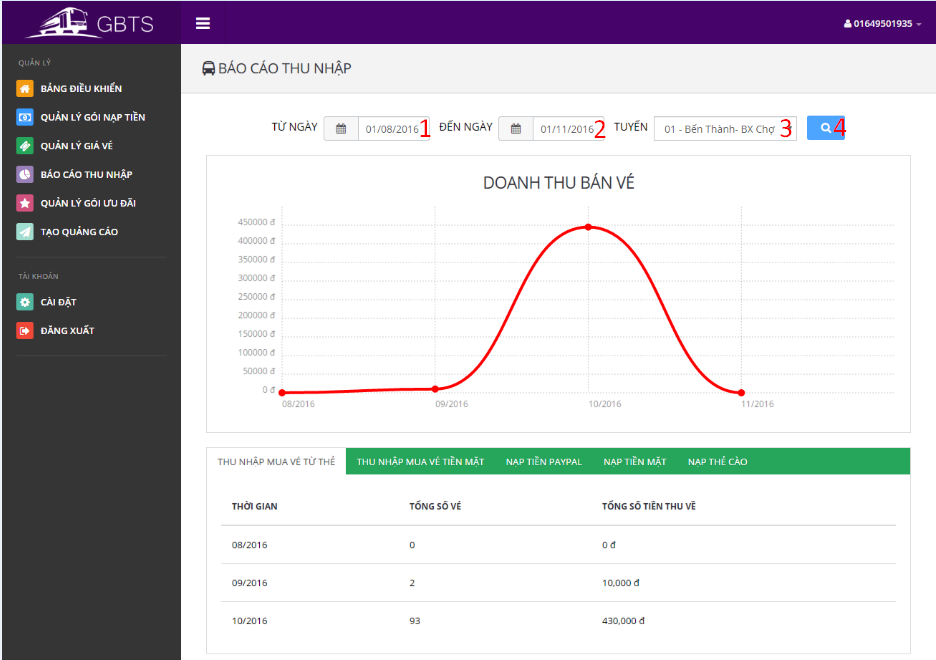
5.2.5.3 Manage Ticket Type



**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 1 | EditTicketType | Send edit ticket type command | N/A | Edit ticket type view is showed. |
| 2 | DeleteTicketType | Send delete ticket type command | N/A | A delete confirmation is show. |
| 3 | AddTicketType | Send add ticket type command | N/A | Add ticket type view is showed. |

5.2.5.4 Get income report



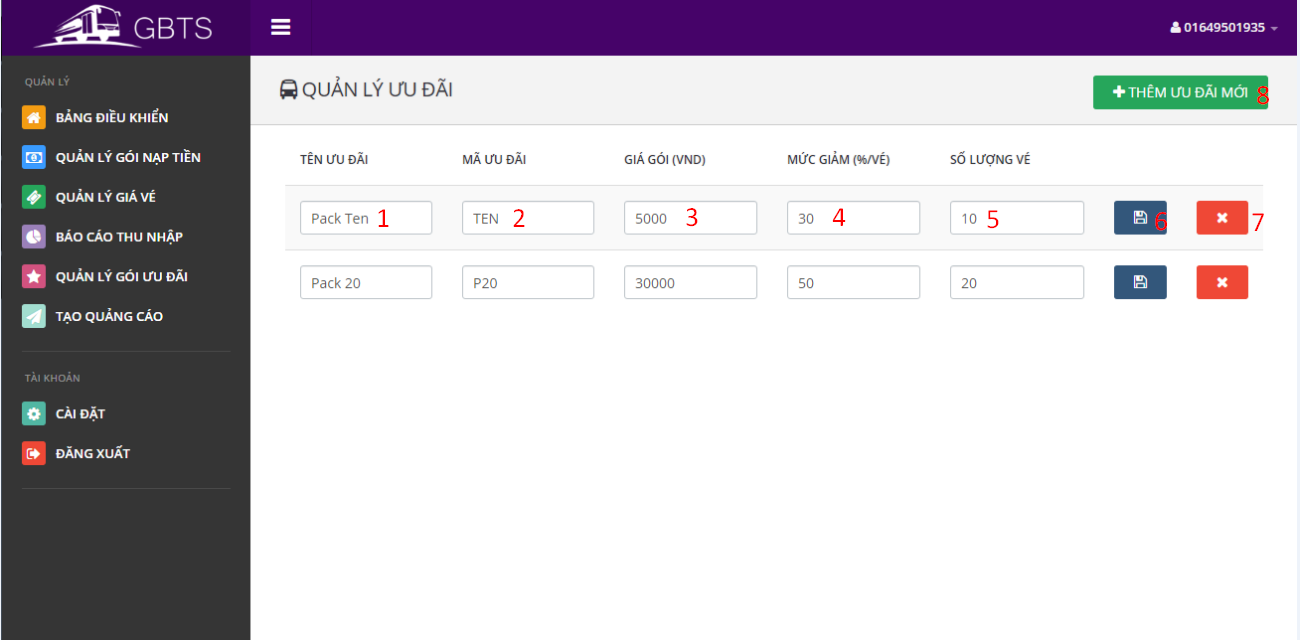
**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | BeginDate | Begin date | No | No | Date selection box | String | N/A |
| 2 | EndDate | End date | No | No | Date selection box | String | N/A |
| 3 | BusRoute | Bus route | No | Yes | Selectbox | String | N/A |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 4 | GetOutComeReport | Send get outcome report command | N/A | Report according to begin date and end date is showed. |

5.2.5.5 Manage Subscriptions



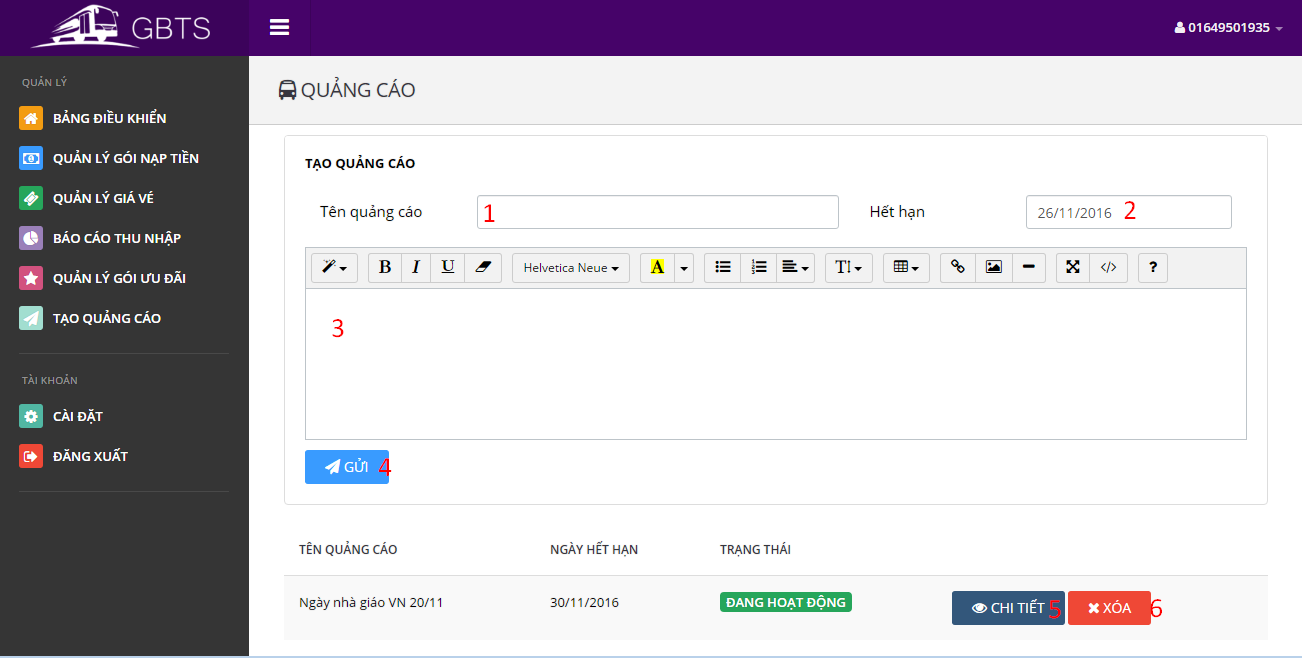
**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | SubscriptionName | Subscription name | No | Yes | Textbox | String | 3-250 characters |
| 2 | SubscriptionCode | Subscription code | No | Yes | Textbox | String | 3-6 characters |
| 3 | SubscriptionPrice | Subscription price | No | Yes | Textbox | Number | N/A |
| 4 | DiscountPercent | Subscription discount percent per ticket | No | Yes | Textbox | Number | N/A |
| 5 | TicketNumber | Number of applied ticket. | No | Yes | Textbox | Number | N/A |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 6 | SaveSubscription | Send save subscription command | All fields are required. | Success message is showed |
| 7 | DeleteSubscription | Send delete subscription command | N/A | A confirmation box is showed. |
| 8 | AddSubscription | Send add subscription command | N/A | New blank subscription is added to subscription list. |

5.2.5.6 Create promotion



**Fields**

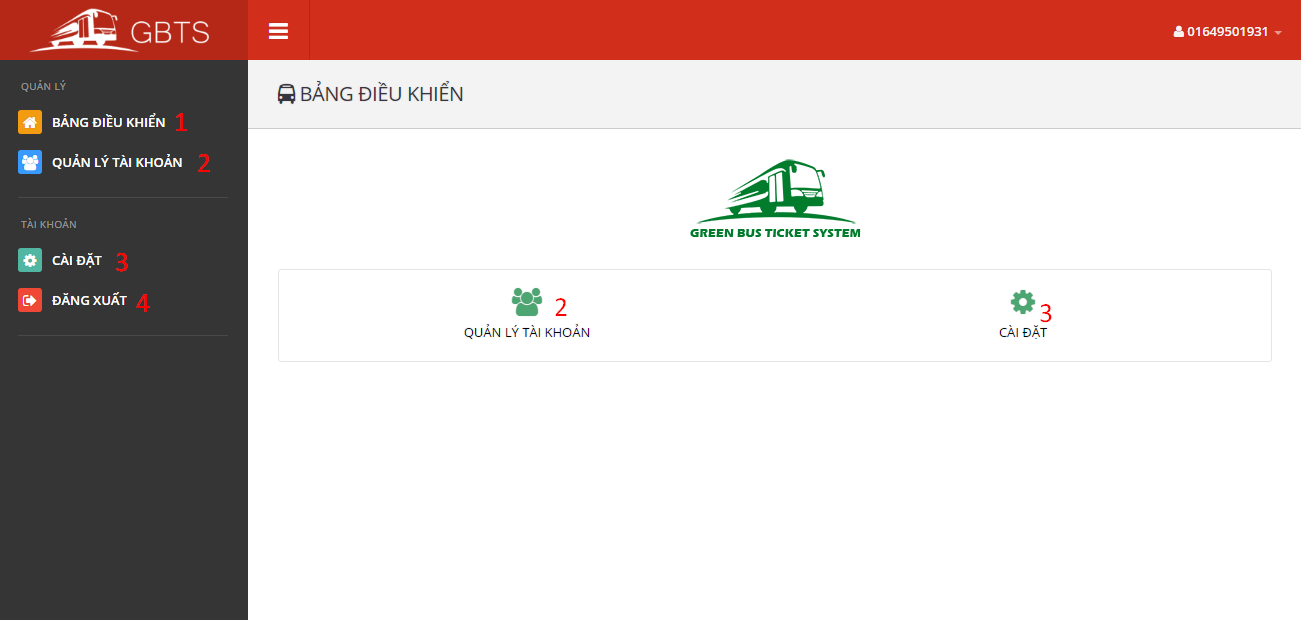
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | PromotionName | Promotion name | No | Yes | Textbox | String | 3-250 characters |
| 2 | ExpiredDate | Promotion expired date | No | Yes | Date selection box | String | N/A |
| 3 | PromotionContent | Promotion content | No | Yes | Text area | String | At least 20 characters |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 4 | CreatePromotion | Send create promotion command | All fields are required. | Success message is showed. New promotion is added to promotion list. |
| 5 | ViewPromotion | Send view promotion command | N/A | Promotion content is showed. |
| 6 | DeletePromotion | Send delete promotion command | N/A | A confirmation box is showed. |

5.2.6 Admin

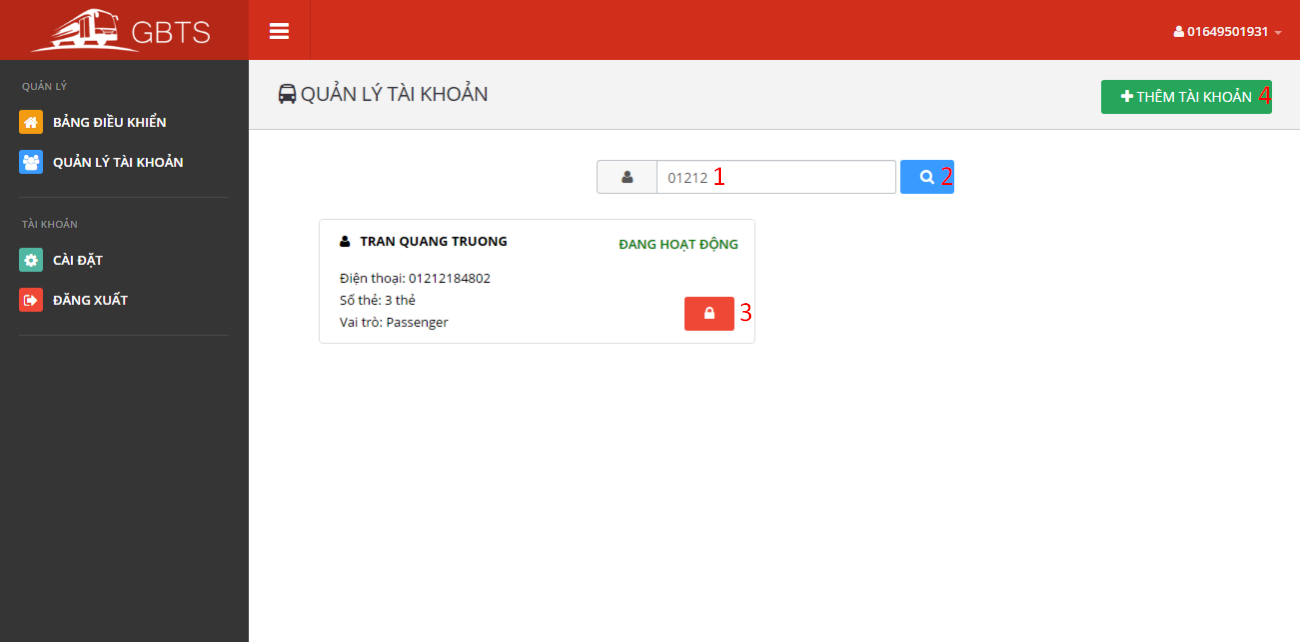
5.2.6.1 Dashboard



**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 1 | ViewDashboard | View dashboard page | N/A | Open dashboard page |
| 2 | ManageAccounts | View manage accounts page | N/A | Open manage accounts page |
| 3 | ViewProfile | View profile page | N/A | Open profile page |
| 4 | Logout | Send logout command | N/A | Destroy current session and redirect to login page. |

5.2.6.2 Manage Accounts



**Fields**

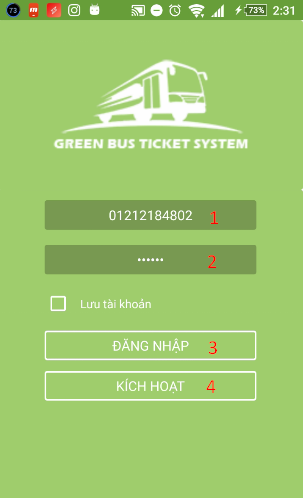
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type | Length |
| 1 | Term | Account phone number or full name | No | No | Textbox | String | 3-250 characters |

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 2 | Search account | Send search account command | N/A | Accounts match searching term is showed. |
| 3 | Activated/Deactivated account | Send activated/deactivated account command | N/A | Account is activated/deactivated |
| 4 | AddAccount | Send add account command | N/A | Add account view is showed. |

### 5.2 Mobile Application Design

#### 5.2.1 Login



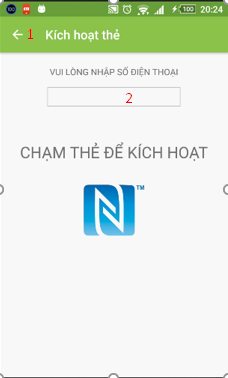
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type |
| 1 | Phone number | Allow user to input their registered phone number. | No | Yes | Number input | String |
| 2 | Password | Allow user to input their registered password | No | Yes | Text input | String |

**Fields**

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 3 | Login | Perform login function | Yes | Open view main screen or display error messages |
| 4 | Activate | Allow user active account by phone number. | Yes | Open view active screen |

#### 5.3.2 Active Card



**Fileds**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type |
| 1 | Phone number | Allow user to input their registered phone number. | No | Yes | Text input | String |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 1 | Back | Navigate back 1 screen | No | Open previous screen in the activity stack |

**Button/Hyperlinks**

‘

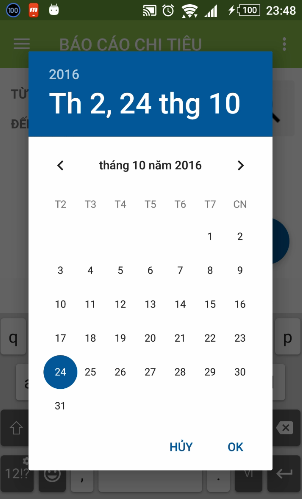
#### 5.3.3 Main screen



**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 1 | Navigation Drawer | Navigation Drawer | No | Open/Close Navigation Drawer |
| 2 | Menu item | Let user choose a function | No | Open fast function |
| 3 | Search | Open search screen | No | Open search screen |

#### 5.3.4 View Report



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type |
| 1 | Date Start | Allow user to input date start. | No | Yes | Text input | Datetime |
| 2 | Date End | Allow user to input date end. | No | Yes | Text input | Datetime |

**Fileds**

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 3 | View report | View report of card | No | Open/Close Navigation Drawer |
| 4 | Navigation Drawer | Navigation Drawer | No | Open/Close Navigation Drawer |

#### 5.3.5 Manage Account



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field Name | Description | Read only | Mandatory | Control type | Data type |
| 2 | Full Name | Allow user input name. | No | Yes | Text Input | String |
| 3 | Old password | Allow user input password. | No | Yes | Text | String |
| 4 | New password | Allow user input password | No | Yes | Text | String |
| 5 | Confirm password | Allow user input password | No | Yes | Text | String |
| 6 | Notification | Allow user min balance to get notification. | No | Yes | Text | Number |

**Fields**

**Button/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validation | Outcome |
| 1 | Navigation Drawer | Navigation Drawer | No | Open/Close Navigation Drawer |
| 7 | Save | Send request to server | No | Change data on this screen |

## Database design

### 6.1 Entity Relationship Diagram

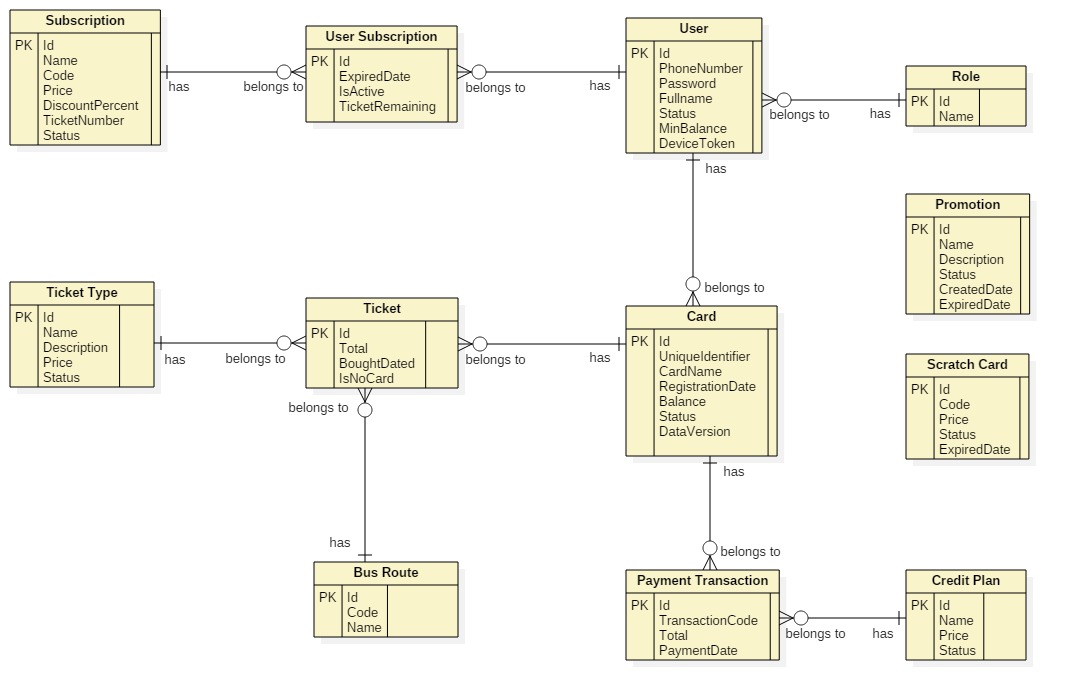


Figure 59: <Database design> Entity Relation Diagram

### 6.2 Entity Dictionary

|  |  |
| --- | --- |
| Entity Name | Description |
| User | Contains all properties about user. |
| Passenger | Contains all properties about passenger info. |
| Admin | Contains all properties about admin info. |
| Staff | Contains all properties about staff info. |
| Manager | Contains all properties about manager info. |
| Card | Contains all properties about NFC card. |
| Ticket Type | Contains all properties about ticket type for based on bus route, each ticket type has diffirent price. |
| Bus Route | Contains all properties about bus route in Ho Chi Minh City. |
| Ticket | Contains all properties about ticket to specify which card buy ticket belongs to which ticket type on which bus route. |
| Credit Plan | Contains all info about credit plan which will be chosen while adding credit to card. |
| Payment Transaction | Contains all info about transaction when adding credit to card via cash, credit plan or scratch card. |
| Promotion | Contains all info about promotion. |
| Scratch Card | Contains all info about scratch card. |
| Subscription | Contains all info about offer subscription. |
| User Subscription | Contains all info about user subscription to specify who subscribe which offer subscription. |

Table 72: Entity Dictionary

## Algorithms

### 7.1 AES Encryption

#### 7.1.1 Definition

AES (acronym of Advanced Encryption Standard) is a symmetric encryption algorithm. The algorithm was developed by two Belgian cryptographer Joan Daemen and Vincent Rijmen. AES was designed to be efficient in both hardware and software, and supports a block length of 128 bits and key lengths of 128, 192, and 256 bits.

#### 7.1.2 Define Problem

For convevient purposes of allowing passenger to buy ticket in case of no internet connection from emulator to server, we save card’s balance and data version to NFC card. We need to encrypt these information to make sure no one can change the original information.

#### 7.1.3 Solution

In order to secure these informations, we use AES Encryption Algorithm to encrypt informations before saving to NFC card.

Algorithm reference: <http://howtodoinjava.com/security/java-aes-encryption-example/>

### 7.2 Token Authentication

#### 7.2.1 Definition

Token authentication is a technique to prove that the passengers is who they claim to be. Requests which have the right token will be processed.

#### 7.2.2 Define problem

When the passengers use mobile application to buy ticket instead of their cards. Mobile application sends some informations to emulator for verification, we have to make sure that these informations only valid in a specific time for buying ticket, anyone who can capture these informations will not be able to use it and buy ticket later.

#### 7.2.3 Solution

Token we use in this solution is a GUID. GUIDs are stored as 128-bit values, and are displayed as 32 hexadecimal digits with groups separated by hyphens.

GUID Reference: <https://en.wikipedia.org/wiki/Globally_unique_identifier>

To solve this problem, we should follow these steps:

* Server creates a token if there is no token on the server.
* Server change the token frequently (every 30 minutes).
* When mobile application is used to buy ticket, it asks the token from server.
* Then mobile application sends a request along with this token to emulator.
* Emulator sends to server. Server verifies the received token, if it matchs current server token, request will be processed.

1. Complexity: O(n)
2. Flowchart

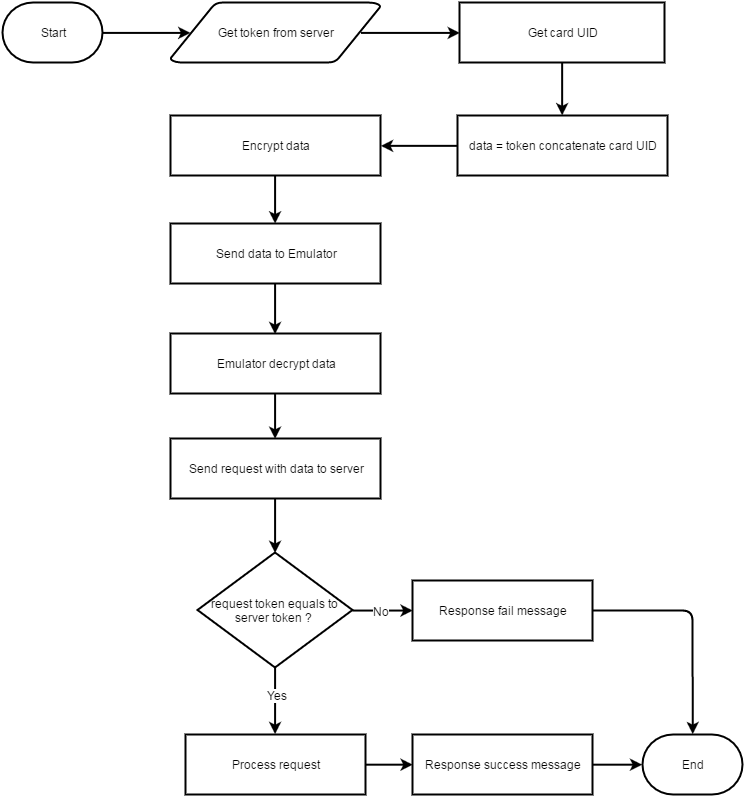


Figure 60: <Algorithms> Entity Relation Diagram

**7.3 Daily Usage Prediction**

1. Definition

Daily Usage Prediction is the way to predict how much a passenger need to spend for buying ticket in the next day base on the average bought tickets in the past.

1. Define Problem

Passenger sometimes forget to add credit to their cards. Each passenger has a diffirent travel demand. We should notify them in the morning if the remaining balance in their card is not enough for spending in the next day.

1. Solution

To solve this problem, we should follow these steps:

* Everyday, at a specific time, Auto Handler will filter the current passenger list to get passengers who have already installed mobile application, bought at least 30 tickets before and the last bought ticket must not exceed 7 days from the current date.
* Calculate the average spending for each customer and on each card of them:

**Average spending** =

**Total Spending**: Total money spent on a card in last 30 days.

**Number of dates**: Number of distinct dates which have at least one ticket was bought in last 30 days.

* If the current card’s balance less than the average spending above, a notification will be send to passenger’s mobile for reminding them.

1. Complexity: O(n2)
2. Flowchart

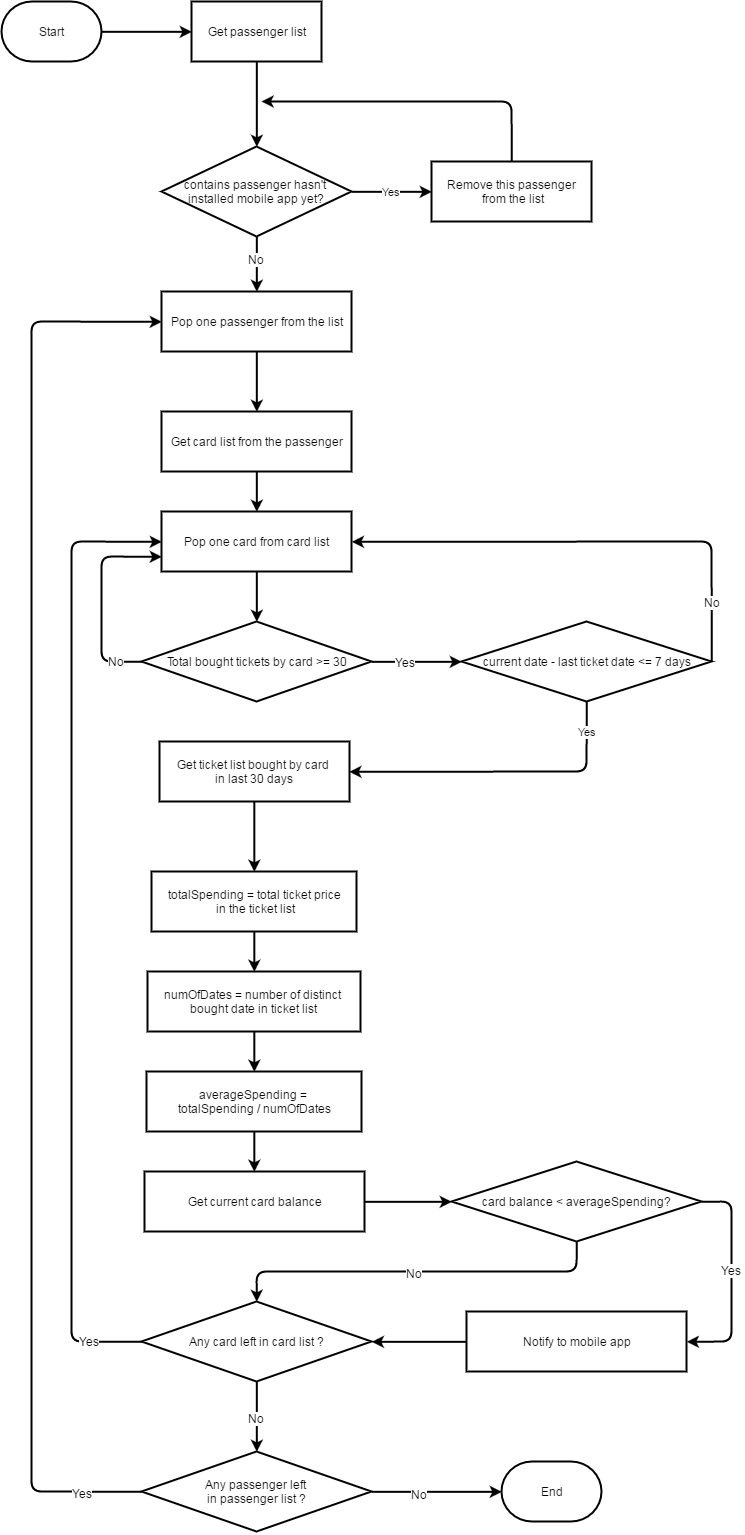


Figure 61: <Algorithms> Daily Usage Prediction