# Report No. 3 Software Requirement Specification

## User Requirement Specification

### Unauthorized User Requirement

Unauthorized is user does not login to this system. Unauthorized only has two functions.

* Sign up.
* Login.

### Authorized User Requirement

Authorized User is user who has logged into system. Authorized User only has two functions.

* Edit profile.
* Logout.

### Borrower Requirement

Borrower can use some following functions:

* + Check in/out
* Add wish list book
* View borrowed list

### Librarian Requirement

Librarian is the user who interacts directly to borrower, they can use some following functions:

* Search for borrower
* Edit borrower
* Manage books

### Admin Requirement

Admin is the user who responsible for managing accounts for the whole system, they can use some following functions:

* Manage account

### Emulator Requirement

Emulator is the device which can interact with mobile phone, it can use some following functions:

* Read & write RFID tags
* Verify user by NFC/QR code

### Auto Handler Requirement

Auto Handler can use some following functions:

* + Suggest promotions
  + Auto extend subscription
  + Crawl bus routes
  + Auto generate server token

## System Requirement Specification



### External Interface Requirement

#### User Interface

* The user interface uses English language in both web admin and mobile application.
* The user interface displays best on 1366x768 resolutions on desktop, and 1080x1920 on mobile.

#### Hardware Interface

* Desktop:
  + RAM: 2GB
  + CPU: 2.5Ghz
* Android Smartphone:
  + RAM: 1GB
  + CPU: 1.7 Ghz

#### Software Interface

* Web admin: Works with Firefox (v30 or above), Google Chrome (v25 or later) or with any web browser that supports HTML5 & CSS3.
* Mobile application: Android operating system (v4.0 or later)

#### Communication Protocol

* Use HTTP 1.1 to communicate between the web browser and the web server.
* Use HTTP 1.1 to communicate between the mobile application and the web service.

### System Overview Use Case

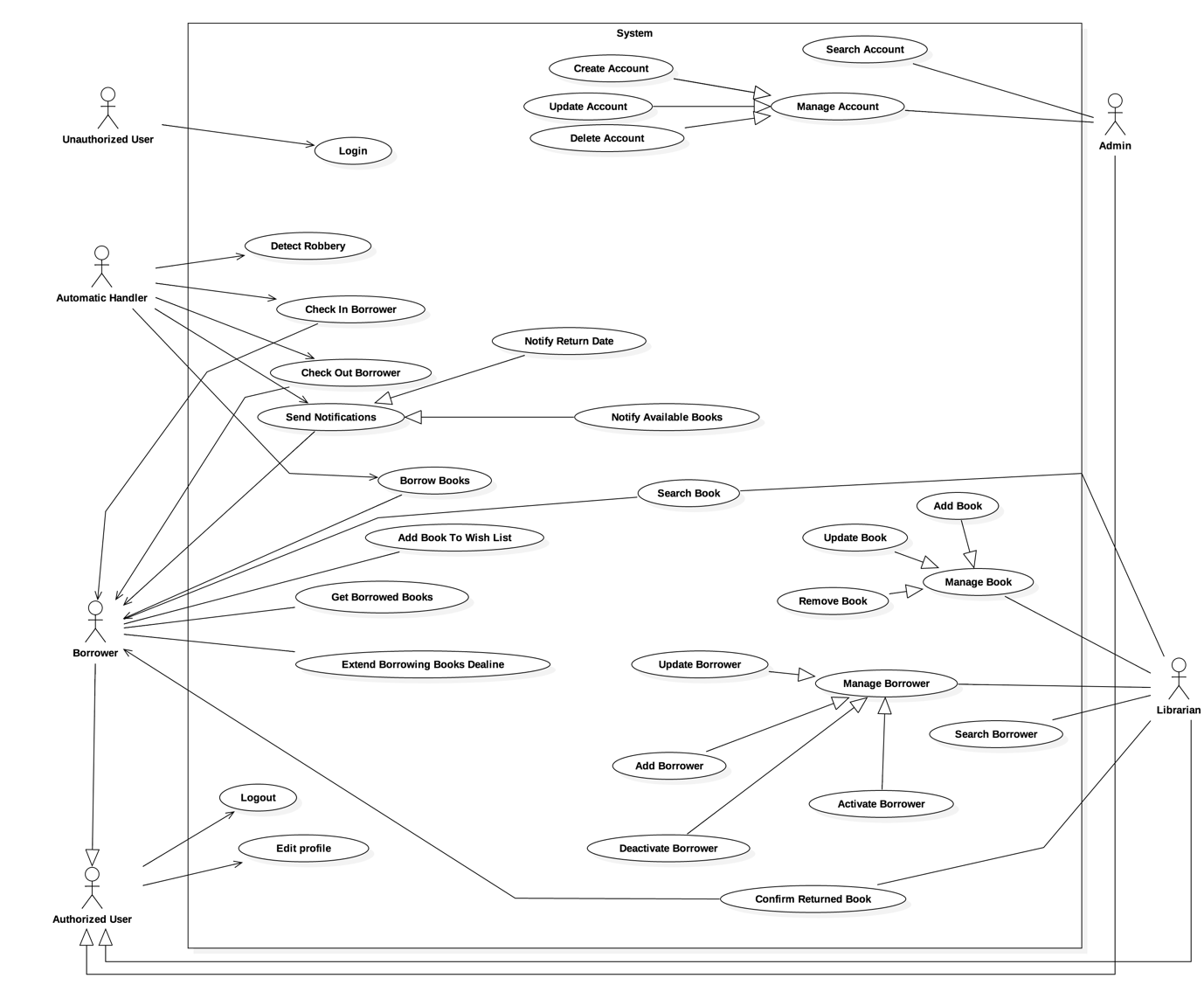


Figure 1 : <Use case> System overview



#### Web Application

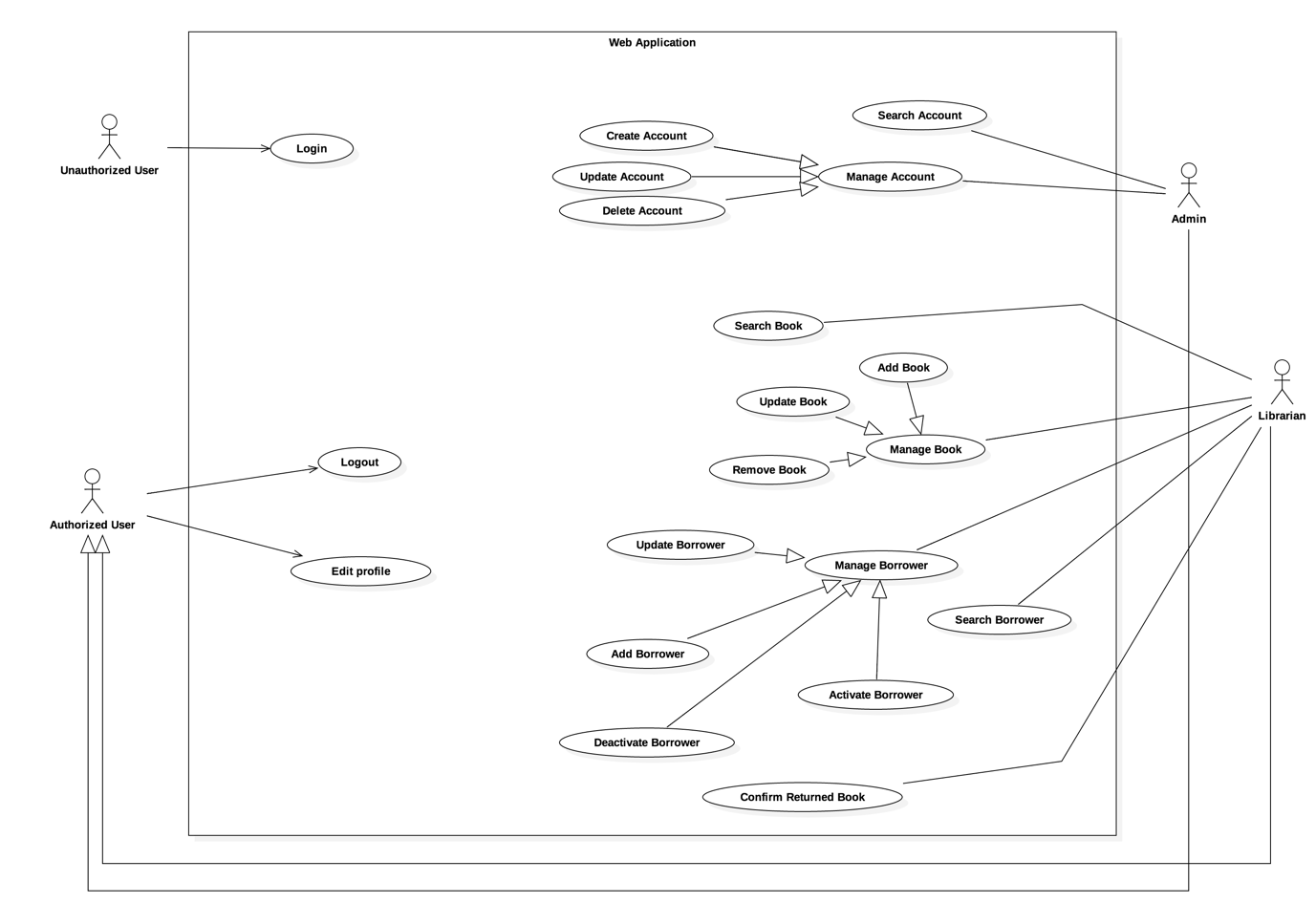


Figure 2 : <Use case> Web application overview

#### Mobile Application

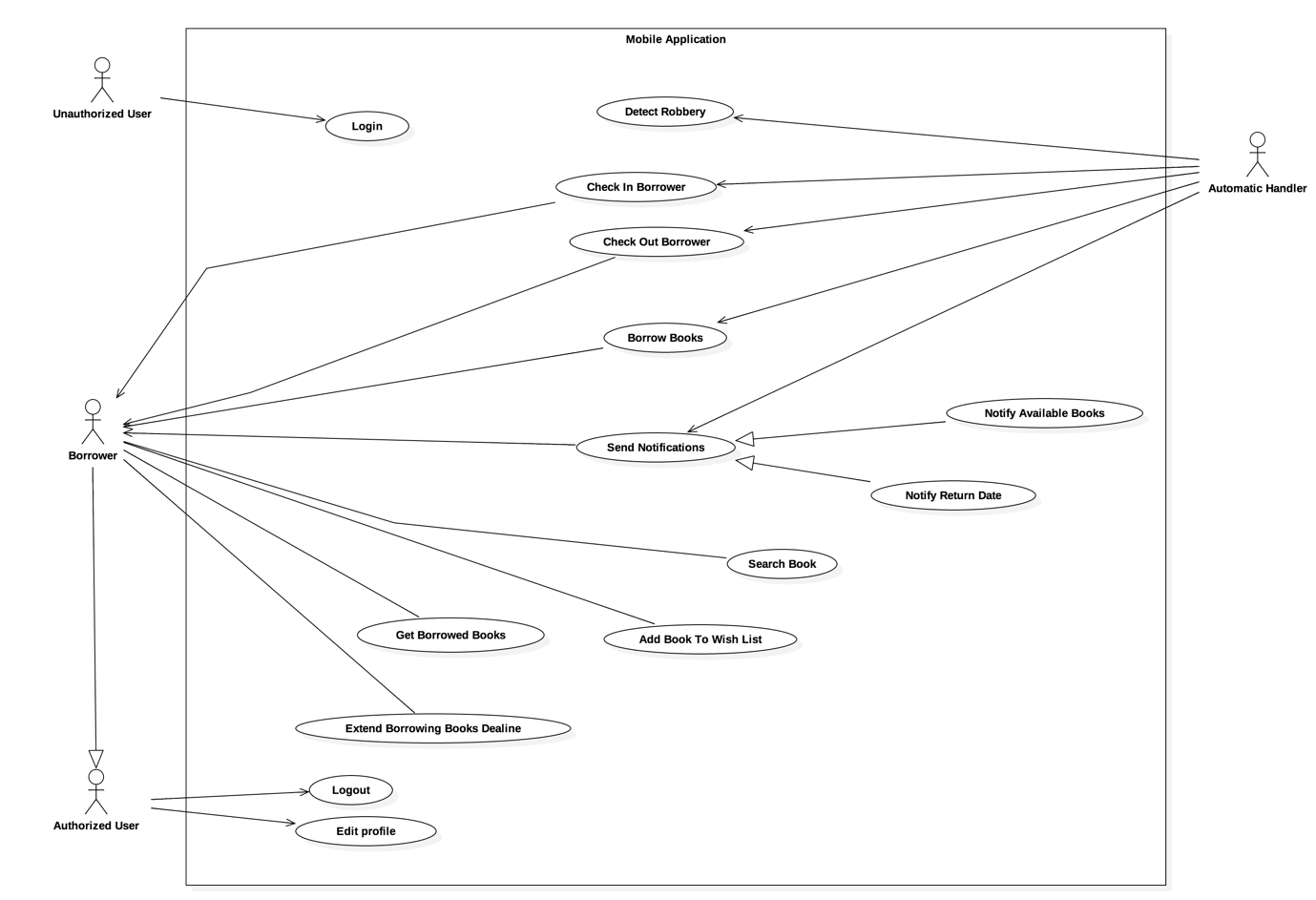


Figure 3: <Use case> Mobile application overview



### List of Use Case

#### Common Use Case

##### Unauthorized User

###### <Unauthorized User> Overview use case



Figure 4: <Unauthorized User> Overview Use Case

###### <Unauthorized User> Login



Figure 5: <Unauthorized User> Login

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – JWL01** | | | |
| **Use Case No.** | JWL01 | **Use Case Version** | 1.0 |
| **Use Case Name** | Login | | |
| **Author** | Nguyen Tuan Anh | | |
| **Date** | January 16th, 2017 | **Priority** | High |
| **Actor:**   * Unauthorized User.   **Summary:**   * This use case allows Unauthorized User to log into the system with a specific role.   **Goal:**   * Authenticate user, redirect user to a specific view based on the role.   **Triggers:**   * Unauthorized User presses Login button.   **Preconditions:**   * Valid username and password are input.   **Post Conditions:**   * **Success:** Unauthorized User is logged into the system as Authorized User. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 2 | Unauthorized User inputs username and password, then presses Login button.  [Exception 1] | Unauthorized User will be logged into System with their specified role.  The system redirects to the role’s view.  [Alternative 1] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Unauthorized user leaves the username and password field empty. | System displays an error message. |   **Exception:**  [Exception 1]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Unauthorized User inputs wrong username and password. | System displays an error message. |   **Relationships: N/A**  **Business Rules:**   * System authenticates user by checking the username and password. * Unauthorized User enters password in a hidden text field and the password must be encrypted before sending to server. * After logged in to the system, Unauthorized User will be redirected to a specific view: * If the role is “Admin”, the system will display Admin view. * If the role is “Librarian”, the system will display Librarian view. * If the role is “Borrower”, the system will display Borrower view. | | | |

##### Authorized User

###### <Unauthorized User> Overview Use Case

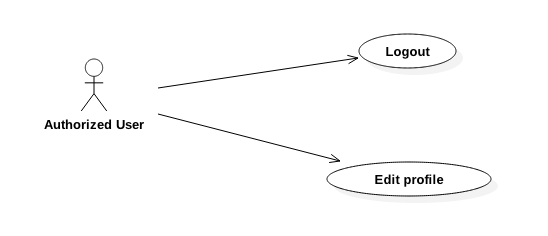


Figure 6: <Authorized User> Overview Use Case

###### <Unauthorized User> Logout

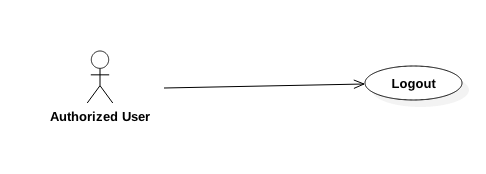


Figure 7: <Authorized User> Logout

###### <Unauthorized User> Edit Profile

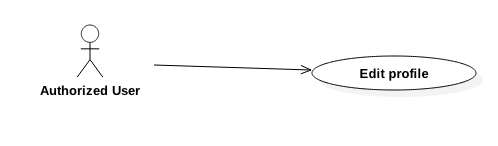


Figure 8: <Authorized User> Edit Profile

#### Web Application

##### Admin

###### <Admin> Overview Use Case

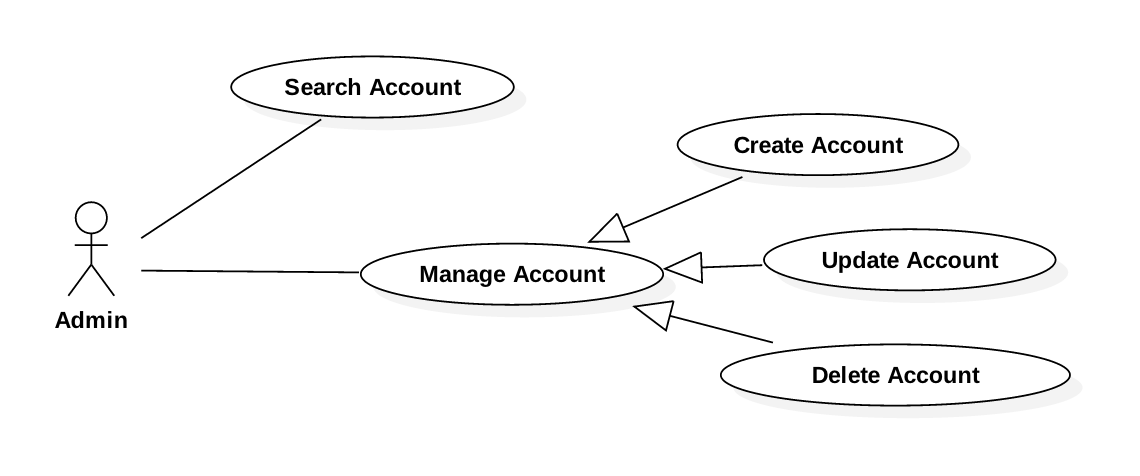


Figure 9: <Admin> Overview Use Case

###### <Admin> Search Account

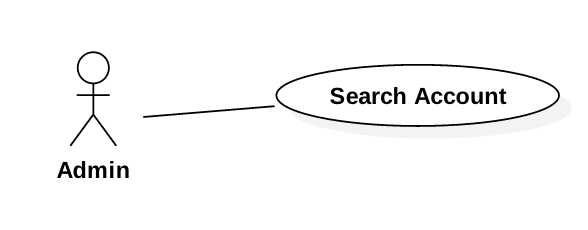


Figure 10: <Admin> Search Account

###### <Admin> Create Account

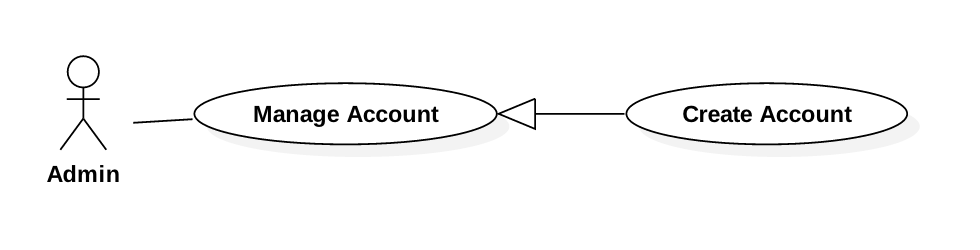


Figure 11: <Admin> Create Account

###### <Admin> Update Account

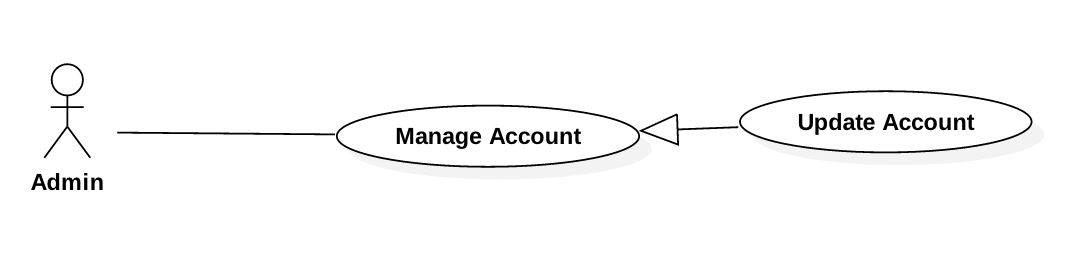


Figure 12: <Admin>Update Account

###### <Admin> Delete Account

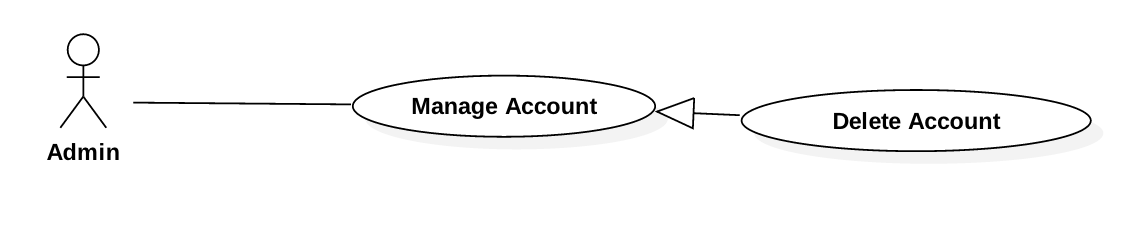


Figure 13: <Admin> Delete Account

##### Librarian

###### <Librarian> Overview Use Case

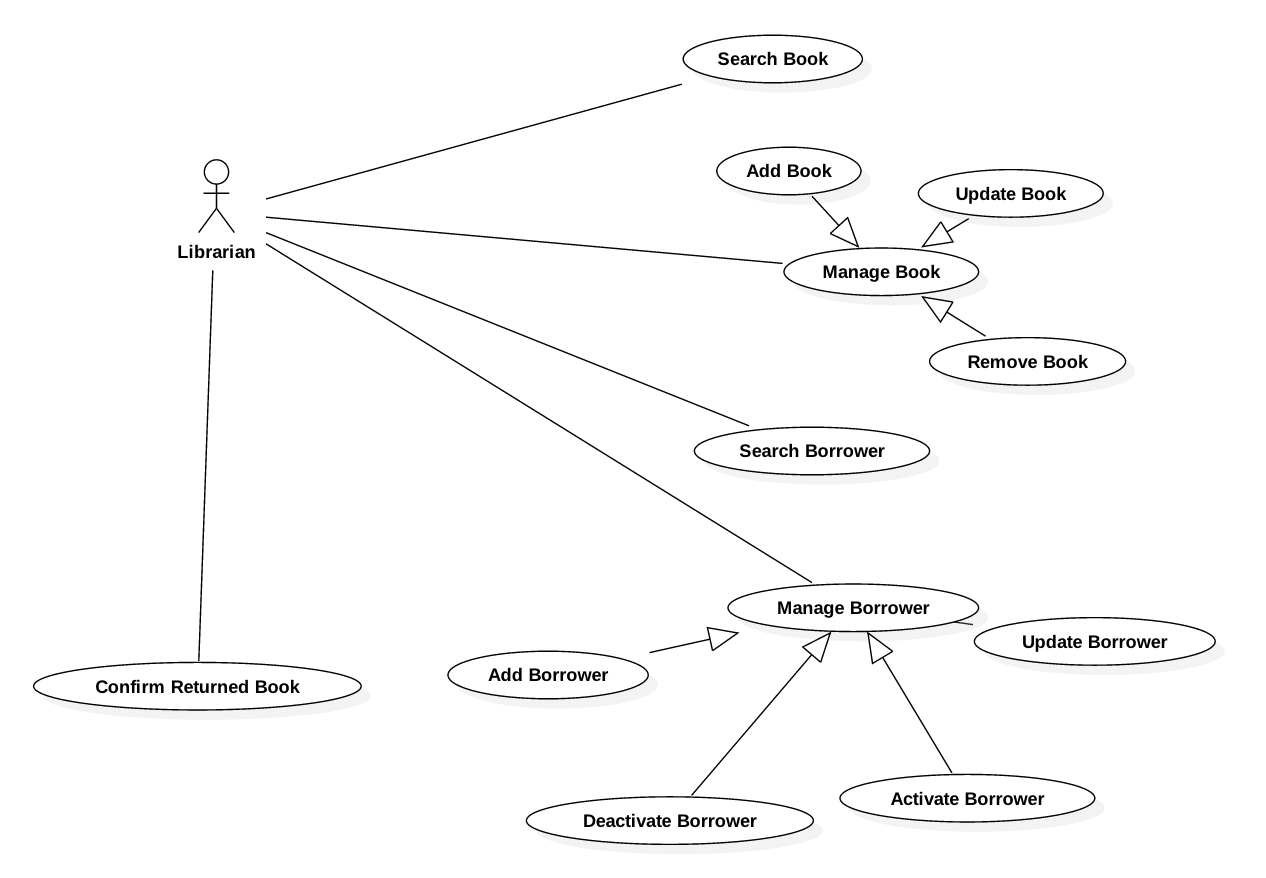


Figure 14: <Librarian> Overview Use Case

###### <Librarian> Search Book



Figure 15: <Librarian> Search Book

###### <Librarian> Add Book

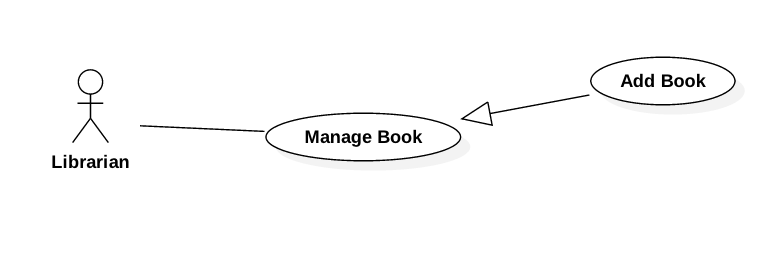


Figure 16: <Librarian> Add Book

###### <Librarian> Update Book

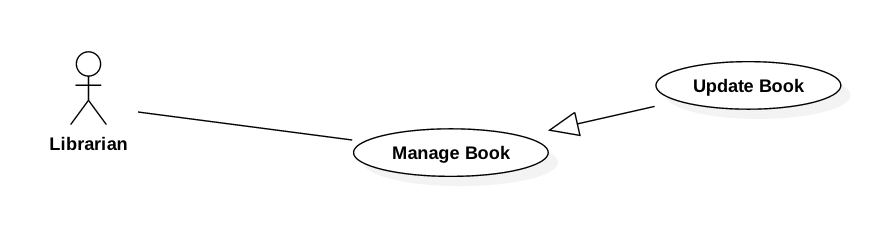


Figure 17: <Librarian> Update Book

###### <Librarian> Remove Book

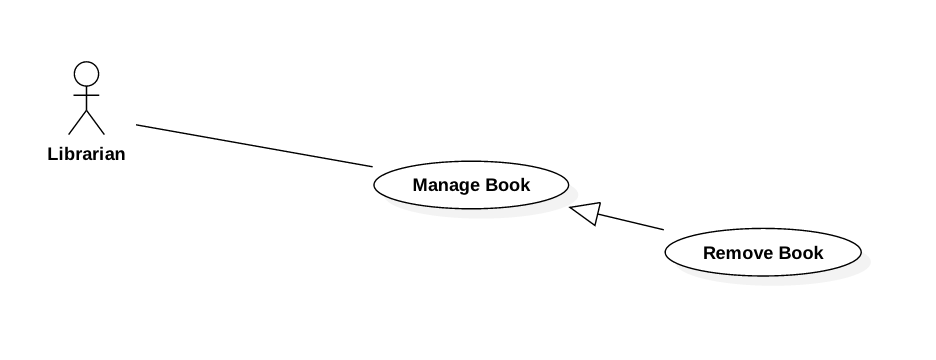


Figure 18: <Librarian> Remove Book

###### <Librarian> Add Borrower



Figure 19: <Librarian> Add Borrower

###### <Librarian> Update Borrower

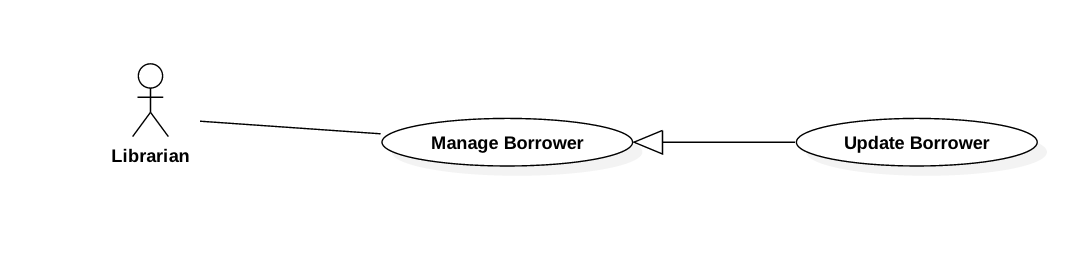


Figure 20: <Librarian> Update Borrower

###### <Librarian> Activate/Deactivate Borrower

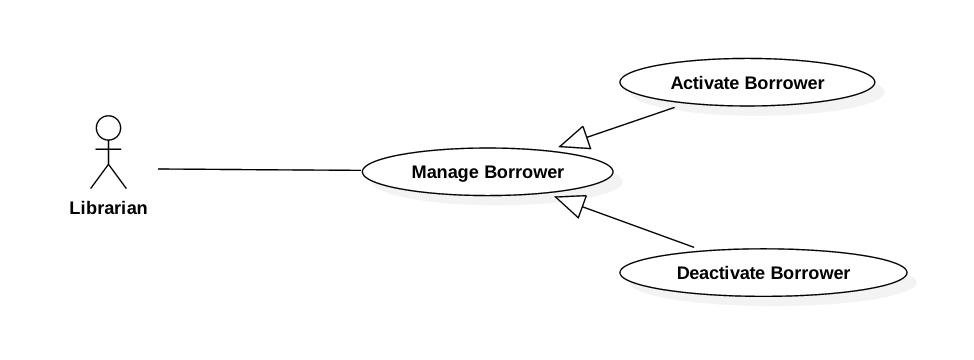


Figure 21: <Librarian> Activate/Deactivate Borrower

###### <Librarian> Search Borrower



Figure 22: <Librarian> Search Borrower

###### <Librarian> Confirm Returned Book

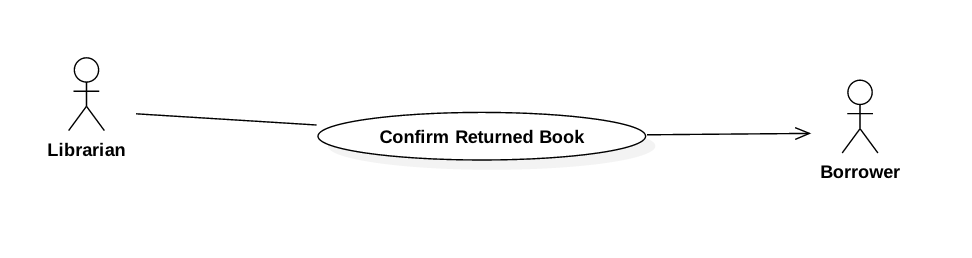


Figure 23: <Librarian> Confirm Returned Book

#### Mobile Application

##### Borrower

###### <Borrower> Overview Use Case

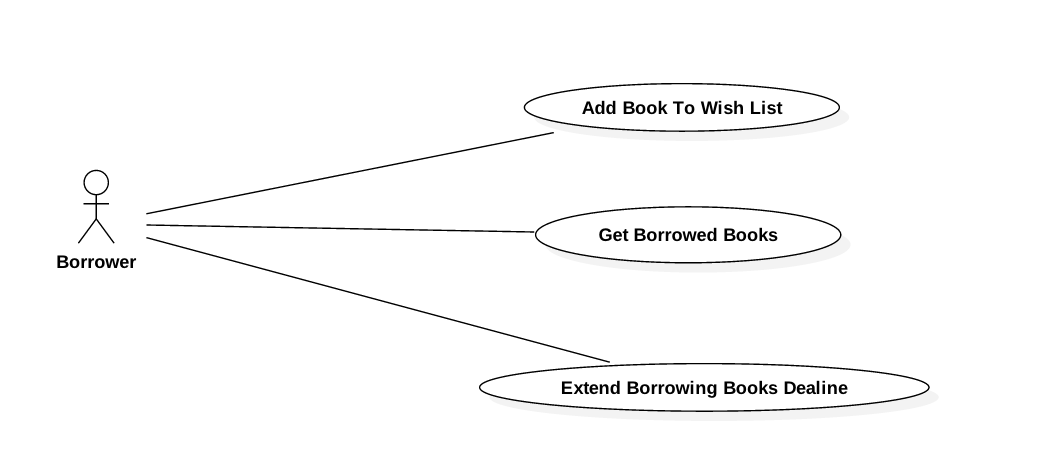


Figure 24: <Borrower> Overview Use Case

###### <Borrower> Search Book



Figure 25: <Borrower> Search Book

###### <Borrower> Add Book To Wish List



Figure 26: <Borrower> Add Book To Wish List

###### <Borrower> Get Borrowed Books



Figure 27: <Borrower> Get Borrowed Books

###### <Borrower> Extend Borrowed Book Deadline



Figure 28: <Borrower> Extend Borrowed Book Deadline

##### Automatic Handler

###### <Automatic Handler> Overview Use Case

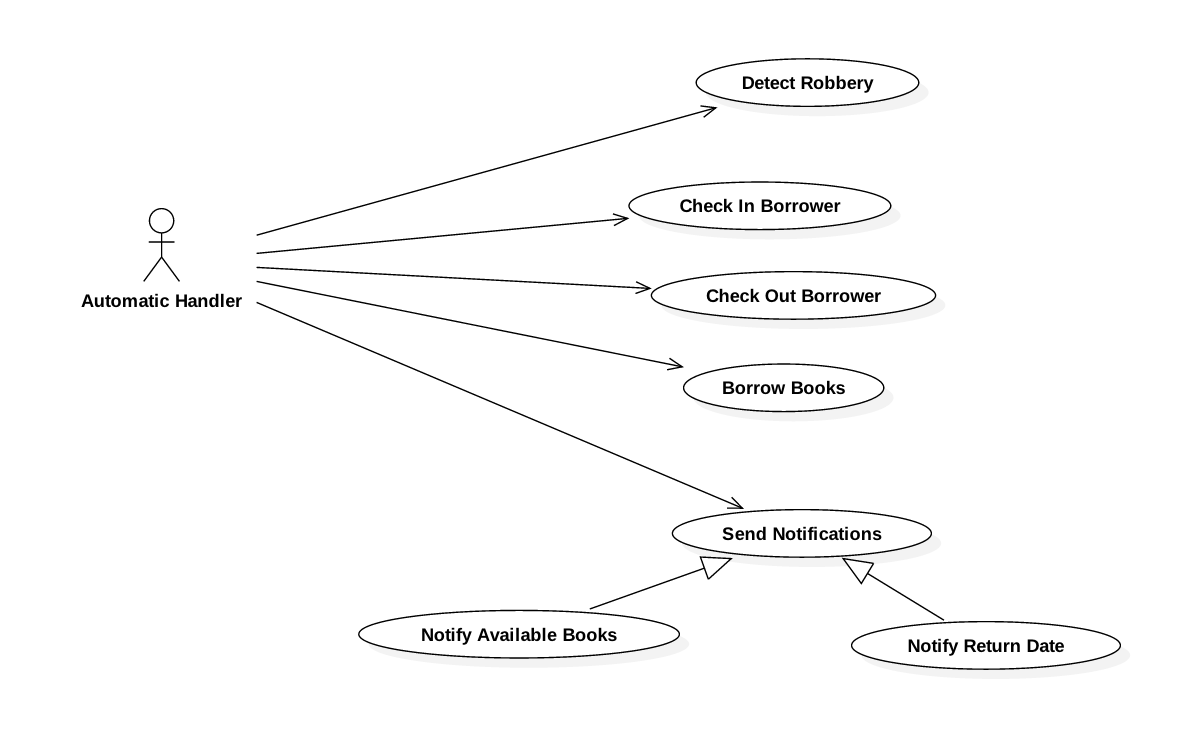


Figure 29: <Automatic Handler> Overview Use Case

###### <Automatic Handler> Detect Robbery

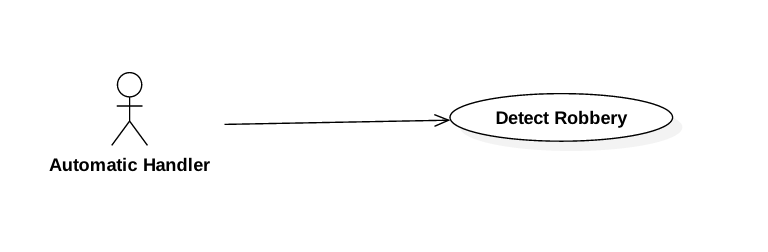


Figure 30: <Automatic Handler> Detect Robbery

###### <Automatic Handler> Check In Borrower

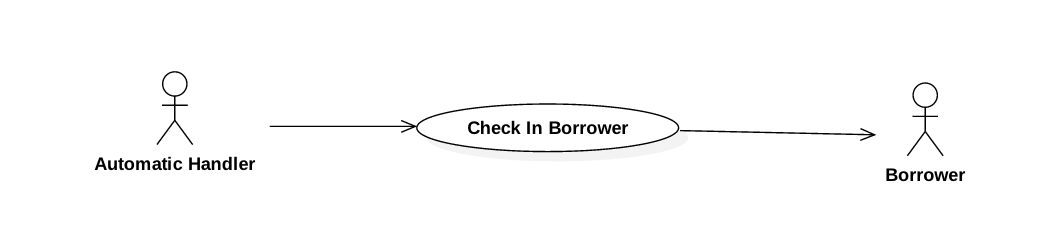


Figure 31: <Automatic Handler> Check In Borrower

###### <Automatic Handler> Check Out Borrower

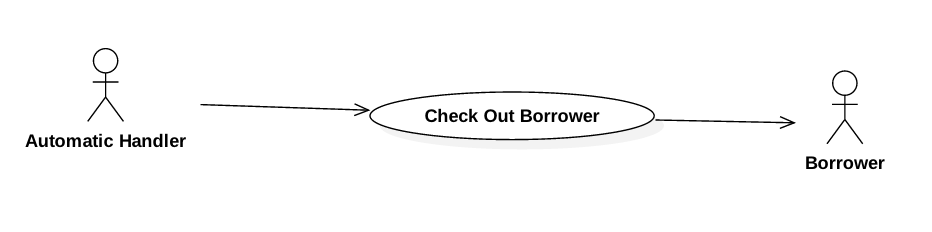


Figure 32: <Automatic Handler> Check Out Borrower

###### <Automatic Handler> Borrow Books

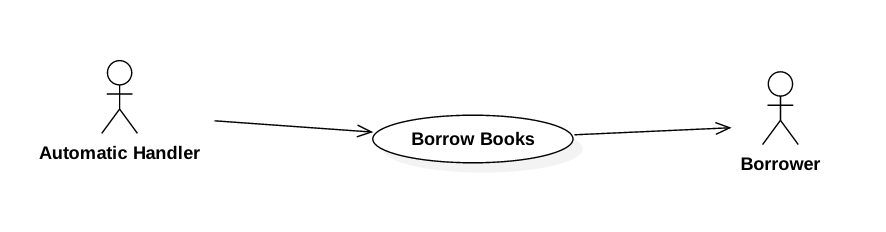


Figure 33: <Automatic Handler> Borrow Books

###### <Automatic Handler> Notify Available Books

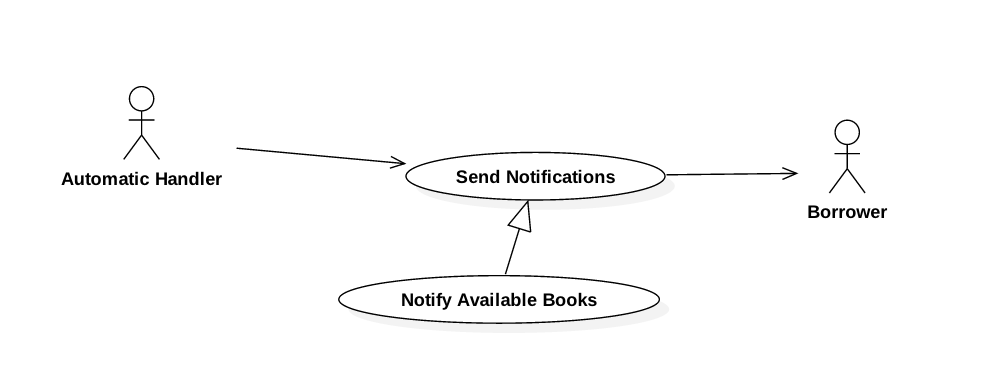


Figure 34: <Automatic Handler> Notify Available Books

###### <Automatic Handler> Notify Return Date

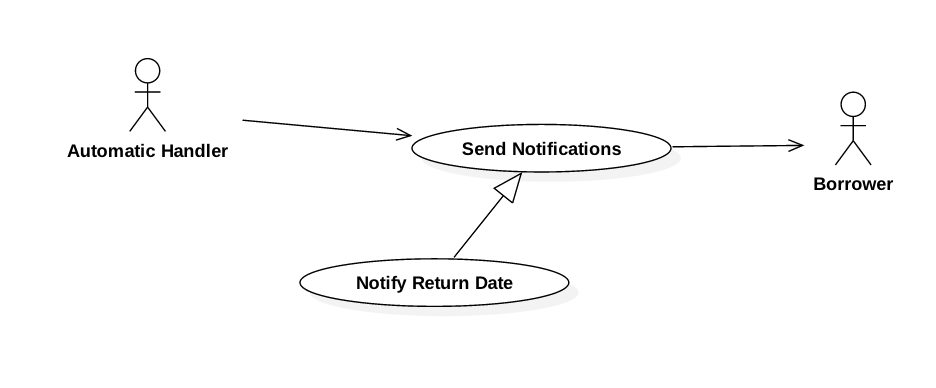


Figure 35: <Automatic Handler> Notify Return Date

##### Unauthorized User

###### <Unauthorized User> Overview use case



Figure 3: <Unauthorized User> Overview use case

###### <Unauthorized User> Login



Figure 4: <Unauthorized User> Login

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS01** | | | |
| **Use Case No.** | GBTS01 | **Use Case Version** | 1.0 |
| **Use Case Name** | Login | | |
| **Author** | Tran Quang Truong | | |
| **Date** | September 18, 2016 | **Priority** | High |
| **Actor:**   * Unauthorized User   **Summary:**   * This use case allows unauthorized user to log into the system with a specific role   **Goal:**   * Authenticate user, redirect user to a specific view base on the role.   **Triggers:**   * Unauthorized User sends Login account command.   **Preconditions:**   * Valid phone number and password is inputted.   **Post Conditions:**   * **Success:** Unauthorized User is logged into the system as an Authorized User. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 2 | Unauthorized User inputs information then send Login command.  [Exception 1] | Unauthorized user will login system with their specified role.  The system redirects to the role’s view.  [Alternative 1] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Unauthorized user leaves phone number and password field empty | System display an error message. |   **Exception:**  [Exception 1]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Unauthorized user inputs wrong phone number and password | System displays an error message. |   **Relationships: N/A**  **Business Rules:**   * System authenticate user by checking the phone number and password. * User enter password in the hidden text field and password must be encrypted before sending to server. * After logged in to the system, actor will be redirect to a specific view: * If the role is “Hành khách”, the system will display Passenger view. * If the role is “Quản lý”, the system will display Bus Manager view. * If the role is “Nhân viên”, the system will display Bus Manager view. * If the role is “Quản trị”, the system will display Administrator view. | | | |

Table 1: <Unauthorized User> Login

###### <Unauthorized User> Activate account



Figure 5: <Unauthorized User> Activate account

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS02** | | | |
| **Use Case No.** | GBTS02 | **Use Case Version** | 1.0 |
| **Use Case Name** | Activate account | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Unauthorized User   **Summary:**   * This use case allows Unauthorized User to activate a new account.   **Goal:**   * Unauthorized User can activate a new account with an NFC card to become a Passenger.   **Triggers:**   * Unauthorized User sends “Activate account” command.   **Preconditions:**   * Actor logged in system with role “Unauthorized User”. * Actor has a NFC card which isn’t activated yet.   **Post Conditions:**   * **Success:** New account is created. * **Fail:** Send error SMS.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Unauthorized User sends “Activate account” command by sending an SMS with their NFC card id to the system. | System gets and verifies the sender’s phone number and card id from the SMS.  System creates a new account with the phone number and maps the NFC card which has card id from SMS with this account.  System sends a SMS contains account login detail back to user’s phone number.  [Exception 1] |   **Alternative Scenario:** N/A  **Exceptions:**  [Exception 1,2]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Unauthorized User sends SMS with wrong format. | System sends an error back to user’s phone number: “Tin nhan sai cu phap, vui long kiem tra lai.” |   **Relationships:** N/A  **Business Rules:**   * New account will be created with input information and its status is “True”. * New account role is Passenger. * After activating account successful, user can login to the system with account login detail. | | | |

Table 2: <Unauthorized User> Activate account

###### Authorized User

###### < Authorized User> Overview use case



Figure 6: < Authorized User> Overview use case

###### < Authorized User> Edit profile



Figure 7: < Authorized User> Overview use case

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS03** | | | |
| **Use Case No.** | GBTS03 | **Use Case Version** | 1.0 |
| **Use Case Name** | Edit profile | | |
| **Author** | Tran Quang Truong | | |
| **Date** | September 18, 2016 | **Priority** | Low |
| **Actor:**   * Authorized User   **Summary:**   * This use case allows authorized users to edit their personal information.   **Goal:**   * Authorized user’s personal information is updated when they want to change their password or full name.   **Triggers:**   * Authorized User sends Edit profile command.   **Preconditions:**   * User must have logged in.   **Post Conditions:**   * **Success:** Authorized user’s personal information is updated in the database. * **Fail:** Error message is displayed.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Authorized User sends Edit profile command.  [Alternative 1] | System list out information of account in system and components:   * “Số điện thoại”: label * “Họ tên”: free text input, length 6-50 * “Password:: free password input, length 6-50 * Update command. | | 2 | Authorized User updates information. |  | | 3 | Authorized User sends Update command. | System validates the input then updates the account’ profile.  System displays a successful message: “Cập nhật thành công”  [Exeption 1] |   **Alternative Scenario: N/A**  **Exception:**   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Authorized User input password field with wrong format. | System displays error message: “Password từ 6-50 ký tự” |   **Relationships: N/A**  **Business Rules:**   * User update account profile in case of changing full name or password * Password must be encrypted before saving to the system. * User cannot update phone number. | | | |

Table 3: < Authorized User> Overview use case

###### < Authorized User> Log out



Figure 8: < Authorized User> Log out

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS04** | | | |
| **Use Case No.** | GBTS04 | **Use Case Version** | 1.0 |
| **Use Case Name** | Logout | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Authorized User   **Summary:**   * This use case allows Authorized User to logout.   **Goal:**   * Authorized User can logout of the system.   **Triggers:**   * Authorized User sends “Logout” command.   **Preconditions:**   * Actor logged in system with role “Authorized User”.   **Post Conditions:**   * **Success:** Authorized User is logged out. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Authorized User sends “Logout” command. | System clears the current session of the user then redirects to home page. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * After logging out of the system, user is no longer has permission to access any authorized function of the system. | | | |

Table 4: < Authorized User> Log out

##### Passenger

###### < Passenger> Overview use case



Figure 9: < Passenger> Overview use case

###### < Passenger> Get NFC card



Figure 10: < Passenger> Get NFC card

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS05** | | | |
| **Use Case No.** | GBTS05 | **Use Case Version** | 1.0 |
| **Use Case Name** | Get NFC card | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Passenger   **Summary:**   * This use case allows Passenger to get their NFC cards list.   **Goal:**   * The passenger can view all their NFC cards and the information such as card id, credit remaining of a specific card.   **Triggers:**   * Passenger sends “Get NFC cards” command.   **Preconditions:**   * Actor logged in system with role “Passenger”. * Passenger has at least one NFC card in the system.   **Post Conditions:**   * **Success:** The NFC card list is show. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Passenger sends “Get NFC cards” command. | System gets NFC card list which belong to current logged account then display “Card list” view, each card has following information:  + “Mã thẻ”: label  + “Tên thẻ”: free text input, length 3-50, required  + “Ngày đăng kí”: label  + “Số dư”: label  + “Trạng thái”: label  + “Add credit” command.  [Alternative 1] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Passenger hs no card on the system | System shows a message "Bạn chưa có thẻ nào trên hệ thống." |   **Exceptions:** N/A  **Relationships:** Add credit   * Extension point: Add credit * Condition: Passenger selects a NFC card in card list and sends “Add credit” command.   **Business Rules:**   * Card status is one of the following status: “Đã kích hoạt”, “Tạm khóa”, “Chưa kích hoạt” * Cards with status “Đã kích hoạt” or “Tạm khóa” will be showed to Passenger. | | | |

Table 5: < Passenger> Get NFC card

###### < Passenger> Add credit



Figure 11: < Passenger> Add credit

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS06** | | | |
| **Use Case No.** | GBTS06 | **Use Case Version** | 1.0 |
| **Use Case Name** | Add credit | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Passenger   **Summary:**   * This use case allows Passenger to add more credit to their NFC card.   **Goal:**   * Passenger can add more credit to their specific NFC card for using this card to buy ticket when traveling by bus.   **Triggers:**   * Passenger sends “Add credit” command of one card on their NFC card list.   **Preconditions:**   * Actor logged in system with role “Passenger”. * Passenger has at least one NFC card in the system. * The NFC card which passenger wants to add credit must have status “Đã Kích Hoạt”   **Post Conditions:**   * **Success:** Credits is added to corresponding card balance. * **Fail:** Show an error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Passenger sends “Add credit” command of one card on their NFC card list. | System shows “Add credit” view with following information:   * “Gói nạp tiền”: select one of the options such as “50.000đ”, “100.000đ”, etc. based on Credit plans created by Manager. * “Checkout” command. * “Cancel” command. | | 2 | Passenger selects one option.  [Alternative 1] |  | | 3 | Passenger sends “Checkout” command. | System redirects to Paypal payment gateway. | | 4 | Pssenger processes the PayPal payment. | If the payment is succeed: Credits will be added to corresponding card balance. System redirects to “Card list” view and display successful message: “Thanh toán thành công!”  [Exception 1] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Passenger sends cancel command | System returns to “Card list” view. |   **Exceptions:**  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Payment failed | System returns to “Card list” view and displays error message: “Thanh toán không thành công!” |   **Relationships:** Get NFC cards   * Extension point: Add credit * Condition: Passenger selects a NFC card in card list and sends “Add credit” command.   **Business Rules:**   * Credit in the system mean the card’s balance which will be used to buy ticket. * After payment successfully, credits will be added to corresponding card balance and payment transaction will be saved to database for tracking by Bus Manager. * Credit plans and its price are loaded from system. These plans are created and managed by Bus Manager. | | | |

Table 6: < Passenger> Add credit

###### < Passenger> Edit card name



Figure 12: < Passenger> Edit card name

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS07** | | | |
| **Use Case No.** | GBTS07 | **Use Case Version** | 1.0 |
| **Use Case Name** | Edit card name | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Passenger   **Summary:**   * This use case allows Passenger edit their NFC card name.   **Goal:**   * Passenger can update new name for their card which make them easy to identify each card in their card list.   **Triggers:**   * Passenger changing name of a card in card list.   **Preconditions:**   * Actor logged in system with role “Passenger”. * Passenger has at least one NFC card in the system.   **Post Conditions:**   * **Success:** New card name is updated. * **Fail:** Show an error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Passenger changing name of a card in card list. | System waits for user inputting. | | 2 | Passenger stops inputting. | System updates new name for corresponding card.  [Exception 1] |   **Alternative Scenario:** N/A  **Exceptions:**  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Passenger inputs card name with wrong format. | System displays error message: “Vui lòng nhập tên thẻ từ 3-50 ký tự ”. |   **Relationships:** Get NFC cards   * Extension point: Edit card name * Condition: Passenger edit card name of a card on card list.   **Business Rules:**   * New default card name when a new card is added to the system is “Thẻ {Card id}” * Card name will be used in outcome report for representing which card paid for ticket. | | | |

Table 7: < Passenger> Edit card name

###### < Passenger> Get outcome report



Figure 13: < Passenger> Get outcome report

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS08** | | | |
| **Use Case No.** | GBTS08 | **Use Case Version** | 1.0 |
| **Use Case Name** | Get outcome report | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Passenger   **Summary:**   * This use case allows Passenger get their outcome report.   **Goal:**   * Passenger can view how much they spend for bus in a specific date range and also the details list of bought tickets.   **Triggers:**   * Passenger sends “Get outcome report” command.   **Preconditions:**   * Actor logged in system with role “Passenger”.   **Post Conditions:**   * **Success:** Outcome report is showed. * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Passenger sends “Get outcome report” command. | System shows “Get outcome report” view with following information:   * “Từ ngày”: date input, default date is 1st of current month, required. * “Đến ngày”: date input, default date is the current date, required. * “Get report” command. | | 2 | Passenger inputs date for both two field. |  | | 3 | Passenger sends “Get report” command. | System verifies input information. Get outcome report of current passenger based on input date range. Display report as a table, each row has following information:   * “Tên thẻ”: label * “Số vé đã mua”: label * “Tổng tiền”: label * “Tuyến thường đi”: label   [Exception 1,2,3,4]  [Alternative 1] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | If there is no data to show. | System shows a message “Không có dữ liệu trong khoảng thời gian đã chọn”. |   **Exceptions:**  [Exception 1,2,3,4]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Passenger does not input required fields. | System notices that user need to input all required field.   * “Từ ngày”: System displays warning message “Vui lòng chọn ngày bắt đầu”. * “Đến ngày”: System displays warning message “Vui lòng chọn ngày kết thúc”. | | 2 | Passenger inputs start date greater than end date. | System displays error message: “Ngày bắt đầu phải nhỏ hơn ngày kết thúc.” | | 3 | Passenger inputs date range over 30 days. | System displays error message: “Khoảng thời gian không vượt quá 30 ngày.” | | 4 | Passenger inputs end date greater than current date. | System displays error message: “Ngày bắt đầu phải nhỏ hơn ngày kết thúc.” |   **Relationships:** N/A  **Business Rules:**   * Start date and end date format are “dd/mm/yyyy”. * End date must not be earlier than start date. * End date must not greater than current date. * Date range for getting report must be within 30 days. * System allows Passenger to search and sort report on all fields. | | | |

Table 8: < Passenger> Get outcome report

###### < Passenger> Find bus



Figure 14: < Passenger> Find bus

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS09** | | | |
| **Use Case No.** | GBTS09 | **Use Case Version** | 1.0 |
| **Use Case Name** | Find bus | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Passenger   **Summary:**   * This use case allows Passenger to find bus routes.   **Goal:**   * Passenger can find bus routes based on starting point and destination for getting instructions to get to the destination.   **Triggers:**   * Passenger sends “Find bus” command.   **Preconditions:**   * Actor logged in system with role “Passenger”.   **Post Conditions:**   * **Success:** Bus routes result list is showed. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Passenger sends “Find bus” command. | System shows “Find bus” view with following information:   * “Điểm khởi hành”: free text input. * “Điểm đến”: free text input. * “Find” command. | | 2 | Passenger inputs information. |  | | 3 | Passenger sends “Find” command. | System verifies input information then finds bus routes. System displays routes list, each route has following information:   * “Số xe”: label * “Thời gian ước tính”: label   [Alternative 1] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | No route found. | If there is no route as the result, system shows a message "Không tìm thấy kết quả nào phù hợp". |   **Exceptions:** N/A  **Relationships:** Get route   * Condition: Passenger chooses a route in route result list. * Extension point: Get route   **Business Rules:**   * If Passenger doesn’t input starting point or destination, system will use the current location automatically. * System finds and display result based on starting point and destination, top 5 results will be displayed to Passenger. | | | |

Table 9: < Passenger> Find bus

###### < Passenger> Get route



Figure 15: < Passenger> Get route

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS10** | | | |
| **Use Case No.** | GBTS10 | **Use Case Version** | 1.0 |
| **Use Case Name** | Get route | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Passenger   **Summary:**   * This use case allows Passenger to get a bus route.   **Goal:**   * Passenger can view a bus route details and instruction for getting bus to get to the destination.   **Triggers:**   * Passenger sends “Get route” command of one route on bus route result list.   **Preconditions:**   * Actor logged in system with role “Passenger”. * The bus route result list has at least one route item.   **Post Conditions:**   * **Success:** Bus route detail is show. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Passenger send “Get route” command of one route on bus route result list.  [Alternative 1] | System gets correspond route details then shows “Route details” view:   * Details instructions * Route preview on map. |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Passenger sends back command | System returns to bus route result list. |   **Exceptions:** N/A  **Relationships:** Find bus   * Condition: Passenger chooses a route in route result list. * Extension point: Get route   **Business Rules:**   * Route instruction will be display step by step to make sure it’s easy to follow. * Route instruction format:   {Estimated time} {Route Number List}  {Action 1}: {Destination1 or Route Number1}  {Action 2}: {Destination2 or Route Number2}  {Action 3}: {Destination3 or Route Number3} | | | |

Table 10: < Passenger> Get route

###### < Passenger> Look up bus route



Figure 16: < Passenger> Get route

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS10** | | | |
| **Use Case No.** | GBTS11 | **Use Case Version** | 1.0 |
| **Use Case Name** | Look up bus route | | |
| **Author** | Doan Minh Duc | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Passenger   **Summary:**   * This use case allows Passenger to get a specific bus route’s information.   **Goal:**   * Passenger can view a bus route’s stops by route code.   **Triggers:**   * Passenger sends “Look up bus route” command of one route on bus route result list.   **Preconditions:**   * Actor logged in system with role “Passenger”. * The bus route result list has at least one route item.   **Post Conditions:**   * **Success:** Bus route detail is show. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Passenger send “Look up bus route” command of one route on bus route result list.  [Alternative 1] | System gets correspond route details then shows “Route details” view:   * Stops name * Route preview on map. |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Passenger sends back command | System returns to bus route result list. |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * Bus stops name will be display in a list. | | | |

Table 11: < Passenger> Get route

###### < Passenger> Register subscription



Figure 17: < Passenger> Get route

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS10** | | | |
| **Use Case No.** | GBTS12 | **Use Case Version** | 1.0 |
| **Use Case Name** | Register subscription | | |
| **Author** | Doan Minh Duc | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Passenger   **Summary:**   * This use case allows Passenger to register a subscription package.   **Goal:**   * Registered subscription passengers can buy ticket with cheaper price.   **Triggers:**   * Passenger sends “Register subscription” command.   **Preconditions:**   * Actor logged in system with role “Passenger”.   **Post Conditions:**   * **Success:** System sends an SMS with successful message to passenger. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Passenger sends “Register subscription” command by sending an SMS with their NFC card id and subscription name to the system. | System sends an SMS with successful message to passenger.  [Exception 1] |   **Alternative Scenario:** N/A  **Exceptions:** N/A  [Exception 1]   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Passenger sends SMS with wrong format. | System sends an error back to user’s phone number: “Tin nhan sai cu phap, vui long kiem tra lai.” |   **Relationships:** N/A  **Business Rules:**   * Subscription will be automatically extend until passenger sends cancel command. * With subscription, passengers can buy ticket at cheap price with limited amount of tickets. | | | |

Table 12: < Passenger> Get route

###### < Passenger> Cancel subscription



Figure 18: < Passenger> Get route

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS10** | | | |
| **Use Case No.** | GBTS13 | **Use Case Version** | 1.0 |
| **Use Case Name** | Cancel subscription | | |
| **Author** | Doan Minh Duc | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Passenger   **Summary:**   * This use case allows Passenger to cancel a registerd subscription package.   **Goal:**   * Passenger cancels a specific subscription package.   **Triggers:**   * Passenger sends “Cancel subscription” command.   **Preconditions:**   * Actor logged in system with role “Passenger”.   **Post Conditions:**   * **Success:** System sends an SMS with successful message to passenger. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Passenger sends “Cancel subscription” command by sending an SMS with their NFC card id and subscription name to the system. | System sends an SMS with successful message to passenger.  [Exception 1] |   **Alternative Scenario:** N/A  **Exceptions:** N/A  [Exception 1]   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Passenger sends SMS with wrong format. | System sends an error back to user’s phone number: “Tin nhan sai cu phap, vui long kiem tra lai.” |   **Relationships:** N/A  **Business Rules:**   * After cancelling a subscription, passengers buy ticket at listed price. | | | |

Table 13: < Passenger> Get route

##### Staff

###### < Staff > Overview use case



Figure 19: < Staff > Overview use case

###### < Staff > Add card



Figure 20: < Staff > Add card

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS11** | | | |
| **Use Case No.** | GBTS14 | **Use Case Version** | 1.0 |
| **Use Case Name** | Add card | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Staff.   **Summary:**   * This use case allows Staff to add a new NFC card to the system.   **Goal:**   * Staff can add a new NFC card to the system to make this card available to use or activate by Passenger.   **Triggers:**   * Staff sends “Add card” command.   **Preconditions:**   * Actor logged in the system with role “Staff”.   **Post Conditions:**   * **Success:** New NFC card added to the system. * **Fail:** System displays error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Staff sends “Add card” command. | System displays “Add card” view with following information:   * “Mã thẻ”: free text input, length 6 – 250, required. * “Số dư”: number input, positive integer number from 0 to 10000000. * “Save” command. * “Cancel” command. | | 2 | Staff taps NFC card on the NFC reader then inputs information.  [Alternative 1] |  | | 3 | Staff sends “Save” command. | System verifies input information, adds new card to the system then displays success message: “Thêm thẻ thành công”.  [Exception 1,2,3] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Staff sends cancel command. | System return to main view. |   **Exceptions:**  [Exception 1,2,3]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Staff does not input required fields. | System notices that user need to input all required field.   * “Mã thẻ”: System displays warning message “Vui lòng nhập mã thẻ”. | | 2 | Staff inputs some fields with wrong format. | System notices that user need to input all required field.   * “Mã thẻ”: System displays warning message “Vui lòng nhập mã thẻ từ 6-250 ký tự”. * “Số dư”: System displays warning message “Vui lòng nhập số dư từ 0 – 10.000.000”. | | 3 | Card ID is already existed in the system. | System displays warning message “Mã thẻ đã tồn tại trên hệ thống”. |   **Relationships: N/A**  **Business Rules:**   * New card will be added to the system with input information with status “Chưa kích hoạt”. * Tap card on NFC reader to get card’s ID. * If no card balance is input. New card balance is 15000. * New card is not belonged to any passenger but is still valid for using to buy ticket, but not allow adding more credit. * Bus company publishes many card for selling to temporary Passenger or in case of forgetting card. | | | |

Table 14: < Staff > Add card

###### < Staff > Activate/Deactivate card



Figure 21: < Staff > Activate/Deactivate card

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS12** | | | |
| **Use Case No.** | GBTS15 | **Use Case Version** | 1.0 |
| **Use Case Name** | Activate / Deactivate card | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Staff.   **Summary:**   * This use case allows Staff to activate / deactivate NFC card in the system.   **Goal:**   * Staff can activate NFC card which will allow using this card to buy ticket. * Staff can deactivate NFC card which will lock this card from buying ticket.   **Triggers:**   * Staff sends “Activate/Deactivate” command of a card on card list.   **Preconditions:**   * Actor logged in the system with role “Staff”.   **Post Conditions:**   * **Success:** Target card status is changed. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Staff sends “Activate/Deactivate” command of a card on card list | If selected card is currently new and not activated yet. System will do nothing and display message “Thẻ này chưa kích hoạt!”.  If selected card is currently activated. System will deactivate corresponding card and display message “Đã khóa thẻ”.  If selected card is currently deactivated. System will activate corresponding card and display message “Đã mở khóa thẻ”. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** Activate/Deactivate card   * Condition: Staff is editing a passenger, select a card and choose Activate/Deactivate card. * Extension point: Activate/Deactivate card.   **Business Rules:**   * System activate NFC card by changing card status to “Đã kích hoạt” which will allow using this card to buy ticket. * System deactivate NFC card by changing card status to “Tạm khóa” which will lock this card from using to buy ticket. * System cannot activate/deactivate new card with status “Chưa kích hoạt”. | | | |

Table 15: < Staff > Activate/Deactivate card

###### < Staff > Search passenger



Figure 22: < Staff > Search passenger

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS13** | | | |
| **Use Case No.** | GBTS16 | **Use Case Version** | 1.0 |
| **Use Case Name** | Search passenger | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Staff.   **Summary:**   * This use case allows Staff to search a passenger.   **Goal:**   * Staff can get passenger list match input phone number and view the information of a specific passenger.   **Triggers:**   * Staff changes content of the search text input.   **Preconditions:**   * Actor logged in the system with role “Staff”. * There is at least one passenger in the system with status “True”.   **Post Conditions:**   * **Success:** Passenger list match searching criteria is showed. * **Fail:** N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Staff inputs phone number to search text input field. | System loads the search results each time the search query changed. Information for each result:   * “Số điện thoại”: label * “Họ và tên: label * “Edit passenger” command.   [Alternative 1] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | There is no passenger match searching criteria. | System shows message: “Không tìm thấy khách hàng nào.” |   **Exceptions: N/A**  **Relationships: N/A**  **Business Rules:**   * Each passenger has a unique phone number. * Any passengers with status “False” will be ignore from searching process and will not be display in search result. | | | |

Table 16: < Staff > Search passenger

###### < Staff > Edit passenger



Figure 23: < Staff > Edit passenger

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS14** | | | |
| **Use Case No.** | GBTS17 | **Use Case Version** | 1.0 |
| **Use Case Name** | Edit passenger | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Staff.   **Summary:**   * This use case allows Staff to edit a passenger.   **Goal:**   * Staff can update a passenger details and activate / deactivate their card in the system.   **Triggers:**   * Staff sends “Edit passenger” command of a passenger on passenger list.   **Preconditions:**   * Actor logged in the system with role “Staff”.   **Post Conditions:**   * **Success:** Passenger’s detail is updated to the system. * **Fail:** System displays error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Staff sends “Edit passenger” command of a passenger on passenger list. | System displays “Edit passenger” view with following information:   * “Số điện thoại”: label, read-only. * “Họ tên”: free text input, length 3 – 80, required. * NFC card list, each card has following information:   + “Mã thẻ”: label   + “Số dư”: label   + “Trạng thái”: select one of the options such as “Đã kích hoạt”, “Tạm khóa”.   + “Activate/Deactivate” command * “Save” command. * “Cancel” command. | | 2 | Staff edits information.  [Alternative 1,2] |  | | 3 | Staff sends “Save” command. | System verifies input information. System updates passenger details to the system then displays success message: “Cập nhật thông tin khách hàng thành công”.  [Exception 1,2] |   **Alternative Scenario:**  [Alternative 1,2]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Staff sends cancel command. | System returns to main view. | | 2 | Staff sends “Activate/Deactivate” command of a NFC card on card list. | If selected card is currently activated. System will deactivate corresponding card and display message “Đã khóa thẻ”.  If selected card is currently deactivated. System will activate corresponding card and display message “Đã mở khóa thẻ”. |   **Exceptions:**  [Exception 1,2]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Staff does not input required fields. | System notices that user need to input all required field.   * “Họ tên”: System displays warning message “Vui lòng nhập họ tên”. | | 2 | Staff inputs some fields with wrong format. | System notices that user need to input again.   * “Họ tên”: System displays warning message “Vui lòng nhập họ tên từ 3-80 ký tự”. |   **Relationships:** Activate/Deactivate card   * Condition: Staff is editing a passenger, select a card and choose Activate/Deactivate card. * Extension point: Activate/Deactivate card   **Business Rules:**   * Passenger details will be updated to the system with input information. Passenger status won’t change. * Each passenger has a unique phone number. The system does not allow Staff to edit phone number. * System activate NFC card by changing card status to “Đã kích hoạt”. * System deactivate NFC card by changing card status to “Tạm khóa”. | | | |

Table 17: < Staff > Edit passenger

###### < Staff > Publish scratch cards



Figure 24: < Staff > Edit passenger

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS14** | | | |
| **Use Case No.** | GBTS18 | **Use Case Version** | 1.0 |
| **Use Case Name** | Publish scratch cards | | |
| **Author** | Doan Minh Duc | | |
| **Date** | 1/11/2016 | **Priority** | Normal |
| **Actor:**   * Staff.   **Summary:**   * This use case allows Staff to publish scratch cards.   **Goal:**   * Staff can publish scratch cards for sale. Passenger can buy scratch card to top up card   **Triggers:**   * Staff sends “Publish scratch cards” command of a passenger on passenger list.   **Preconditions:**   * Actor logged in the system with role “Staff”.   **Post Conditions:**   * **Success:** Newscratch cards are generated in the database. * **Fail:** System displays error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Staff sends “Publish scratch cards” command. | System displays “Publish scratch cards” view with following information:   * “Số lượng”: number input, required. * “Mệnh giá”: select at least one of price options. * “Publish” command. * “Cancel” command. | | 2 | Staff input information.  [Alternative 1,2] |  | | 3 | Staff sends “Save” command. | System generates the scratch cards  [Exception 1] |   **Alternative Scenario:**  [Alternative 1,2]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Staff sends cancel command. | System returns to main view. |   **Exceptions:**  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Staff does not input required fields. | System notices user need to input all required field. |   **Relationships:** N/A  **Business Rules:**   * Each scratch card must have the expired date. * Each scratch card can only be used once. * Scratch card’s code is case sensitive. | | | |

Table 18: < Staff > Edit passenger

##### Manager

###### < Manager > Overview use case



Figure 25: < Manager > Overview use case

###### < Manager > Add ticket type



Figure 26: < Manager > Add ticket type

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS15** | | | |
| **Use Case No.** | GBTS19 | **Use Case Version** | 1.0 |
| **Use Case Name** | Add ticket type | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Manager.   **Summary:**   * This use case allows Manager to add a ticket type.   **Goal:**   * Manager can create and add a ticket type to the system with specific ticket price that is suitable for correspond route distance.   **Triggers:**   * Manager sends “Add ticket type” command.   **Preconditions:**   * Actor logged in the system with role “Manager”.   **Post Conditions:**   * **Success:** New ticket type is added to the system. * **Fail:** System displays error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends “Add ticket type” command. | System displays “Add ticket type” view with following information:   * “Loại vé”: free text input, length 6-250, required. * “Mô tả”: free text area, required. * “Giá vé”: number input, positive integer number from 500 to 100000, required. * “Save” command. * “Cancel” command. | | 2 | Manager inputs information.  [Alternative 1] |  | | 3 | Manager sends “Save” command. | System verifies input information, adds new ticket type to the system then displays success message: “Thêm loại vé thành công”.  [Exception 1,2,3] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends cancel command. | System return to main view. |   **Exceptions:**  [Exception 1,2,3]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Manager does not input required fields. | System notices that user need to input all required field.   * “Tên loại vé”: System displays warning message “Vui lòng nhập tên loại vé”. * “Mô tả”: System displays warning message “Vui lòng nhập mô tả”. * “Giá vé”: System displays warning message “Vui lòng nhập giá vé”. | | 2 | Manager inputs some fields with wrong format. | System notices that user need to input again.   * “Tên loại vé”: System displays warning message “Vui lòng nhập tên loại vé từ 6-250 ký tự”. * “Giá vé”: System displays warning message “Vui lòng nhập giá vé từ 500-100000”. | | 3 | Ticket type name is already existed in the system. | System displays warning message “Tên loại vé không được trùng”. |   **Relationships: N/A**  **Business Rules:**   * New ticket type will be added to the system with input information with status “True”. * Ticket type and its price depend on route distance and will be specify by company. * Each emulator on bus will be configured to a corresponding ticket type. * Passenger pays ticket price according to the ticket type of the bus they use. * Ticket type name is unique to make sure no ticket type is duplicated. | | | |

Table 19: < Manager > Add ticket type

###### < Manager > Edit ticket type



Figure 27: < Manager > Edit ticket type

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS16** | | | |
| **Use Case No.** | GBTS20 | **Use Case Version** | 1.0 |
| **Use Case Name** | Edit ticket type | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Manager.   **Summary:**   * This use case allows Manager to edit a ticket type in the system.   **Goal:**   * Manager can update a ticket type details in the system for adapting the ticket price change.   **Triggers:**   * Manager sends “Edit ticket type” command of one ticket type on ticket type list.   **Preconditions:**   * Actor logged in the system with role “Manager”. * There is at least one ticket type in the system.   **Post Conditions:**   * **Success:** Ticket type detail is updated to the system. * **Fail:** System displays error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends “Edit ticket type” command of one ticket type on ticket type list. | System displays “Edit ticket type” view and loads bus detail to a form with following information:   * “Loại vé”: free text input, length 6-250, required. * “Mô tả”: free text area, required. * “Giá vé”: number input, positive integer number from 500 to 100000, required. * “Save” command. * “Cancel” command. | | 2 | Manager edits information.  [Alternative 1] |  | | 3 | Manager sends “Save” command. | System verifies input information. Update ticket type detail to the system then displays success message: “Sửa thông tin loại vé thành công”.  [Exception 1,2,3] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends cancel command. | System return to main view. |   **Exceptions:**  [Exception 1,2,3]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Manager does not input required fields. | System notices that user need to input all required field.   * “Tên loại vé”: System displays warning message “Vui lòng nhập tên loại vé”. * “Mô tả”: System displays warning message “Vui lòng nhập mô tả”. * “Giá vé”: System displays warning message “Vui lòng nhập giá vé”. | | 2 | Manager inputs some fields with wrong format. | System notices that user need to input again.   * “Tên loại vé”: System displays warning message “Vui lòng nhập tên loại vé từ 6-250 ký tự”. * “Giá vé”: System displays warning message “Vui lòng nhập giá vé từ 500-100000”. | | 3 | Ticket type name is already existed in the system. | System displays warning message “Tên loại vé không được trùng”. |   **Relationships: N/A**  **Business Rules:**   * Ticket type detail will be updated to the system with input information. * Ticket type and its price depend on route distance and will be specify by company. * Each emulator on bus will be configured to a corresponding ticket type. * Passenger pays ticket price according to the ticket type of the bus they use. * Ticket type name is unique to make sure no ticket type is duplicated. | | | |

Table 20: < Manager > Edit ticket type

###### < Manager > Delete ticket type



Figure 28: < Manager > Delete ticket type

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS17** | | | |
| **Use Case No.** | GBTS21 | **Use Case Version** | 1.0 |
| **Use Case Name** | Delete ticket type | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Manager.   **Summary:**   * This use case allows Manager to delete a ticket type from the system.   **Goal:**   * Manager can delete a ticket type from the system.   **Triggers:**   * Manager sends “Delete ticket type” command of one ticket type on ticket type list.   **Preconditions:**   * Actor logged in the system with role “Manager”. * There is at least one ticket type in the system. * The target ticket type has status “True”   **Post Conditions:**   * **Success:** Ticket type is deleted from the system. * **Fail:** System displays error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends “Delete ticket type” command of one ticket type on ticket type list. | System displays confirmation message: “Bạn có chắc chắn muốn xóa?” | | 2 | Manager sends “Yes” command.  [Alternative 1] | System deletes target ticket type from the system and displays success message: “Xóa thành công”; |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends cancel command. | System return to main view. |   **Exceptions: N/A**  **Relationships: N/A**  **Business Rules:**   * The target ticket type must have status “True”. * If target ticket type has not been used yet. System delete this ticket type completely, otherwise system set its status to “False”. | | | |

Table 21: < Manager > Delete ticket type

###### < Manager > Add credit plan



Figure 29: < Manager > Add credit plan

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS18** | | | |
| **Use Case No.** | GBTS22 | **Use Case Version** | 1.0 |
| **Use Case Name** | Add plan | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Manager.   **Summary:**   * This use case allows Manager to add a new credit plan.   **Goal:**   * Manager can create and add a new credit plan to the system. Passenger can add credit to their card by buying these credit plans.   **Triggers:**   * Manager sends “Add plan” command.   **Preconditions:**   * Actor logged in the system with role “Manager”.   **Post Conditions:**   * **Success:** New credit plan is added to the system. * **Fail:** System displays error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends “Add plan” command. | System displays “Add plan” view with following information:   * “Tên gói”: free text input, length 3 – 250, required. * “Mô tả”: free text area, optional. * “Giá gói”: number input, positive integer number from 50000 to 10000000, required. * “Save” command. * “Cancel” command. | | 2 | Manager inputs information.  [Alternative 1] |  | | 3 | Manager sends “Save” command. | System verifies input information, adds new plan to the system then displays success message: “Thêm gói thành công”.  [Exception 1,2,3] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends cancel command. | System return to main view. |   **Exceptions:**  [Exception 1,2,3]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Manager does not input required fields. | System notices that user need to input all required field.   * “Tên gói”: System displays warning message “Vui lòng nhập tên gói”. * “Giá gói”: System displays warning message “Vui lòng nhập giá gói”. | | 2 | Manager inputs some fields with wrong format | System notices that user need to input all required field.   * “Tên gói”: System displays warning message “Vui lòng nhập tên gói từ 3-250 ký tự”. * “Giá gói”: System displays warning message “Vui lòng nhập giá gói từ 50.000-10.000.000”. | | 3 | Credit plan name is already existed in the system. | System displays warning message “Tên gói không được trùng, vui lòng chọn tên gói khác.”. |   **Relationships: N/A**  **Business Rules:**   * New credit plan will be added to the system with input information with status “True” and will available to buy when Passenger want to add credit to their card. * Credit plan name is unique to make sure no plan is duplicated. | | | |

Table 22: < Manager > Add credit plan

###### < Manager > Edit credit plan



Figure 30: < Manager > Edit credit plan

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS19** | | | |
| **Use Case No.** | GBTS23 | **Use Case Version** | 1.0 |
| **Use Case Name** | Edit plan | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Manager.   **Summary:**   * This use case allows Manager to edit a credit plan.   **Goal:**   * Manager can update a credit plan details in the system for adapting price change.   **Triggers:**   * Manager sends “Edit plan” command of one plan on plan list.   **Preconditions:**   * Actor logged in the system with role “Manager”. * There is at least one credit plan in the system with status “True”   **Post Conditions:**   * **Success:** Credit plan details updated to the system. * **Fail:** System displays error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends “Edit plan” command of one plan on plan list. | System displays “Edit plan” view and loads plan detail to a form with following information:   * “Tên gói”: free text input, length 3 – 250, required. * “Mô tả”: free text area, optional. * “Giá gói”: number input, positive integer number from 50000 to 10000000, required. * “Save” command. * “Cancel” command. | | 2 | Manager edits information.  [Alternative ] |  | | 3 | Manager sends “Save” command. | System verifies input information, updates credit plan details then displays success message: “Sửa thông tin gói thành công”.  [Exception 1,2,3] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends cancel command. | System returns to main view. |   **Exceptions:**  [Exception 1,2,3]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Manager does not input required fields. | System notices that user need to input all required field.   * “Tên gói”: System displays warning message “Vui lòng nhập tên gói”. * “Giá gói”: System displays warning message “Vui lòng nhập giá gói”. | | 2 | Manager inputs some fields with wrong format | System notices that user need to input all required field.   * “Tên gói”: System displays warning message “Vui lòng nhập tên gói từ 3-250 ký tự”. * “Giá gói”: System displays warning message “Vui lòng nhập giá gói từ 50.000-10.000.000”. | | 3 | Credit plan name is already existed in the system. | System displays warning message “Tên gói không được trùng, vui lòng chọn tên gói khác.”. |   **Relationships: N/A**  **Business Rules:**   * New credit plan detail will be updated to the system with input information. No change on plan status. * Credit plan name is unique to make sure no plan is duplicated. | | | |

Table 23: < Manager > Edit credit plan

###### < Manager > Delete credit plan



Figure 31: < Manager > Delete credit plan

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS20** | | | |
| **Use Case No.** | GBTS24 | **Use Case Version** | 1.0 |
| **Use Case Name** | Delete plan | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Manager.   **Summary:**   * This use case allows Manager to delete a credit plan.   **Goal:**   * Manager can delete a credit plan from the system to make it unavailable to buy in any situation.   **Triggers:**   * Manager sends “Delete plan” command of one plan on plan list.   **Preconditions:**   * Actor logged in the system with role “Manager”. * There is at least one credit plan in the system with status “True”   **Post Conditions:**   * **Success:** Credit plan is deleted from the system. * **Fail:** System displays error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends “Delete plan” command of one plan on plan list. | System displays confirmation message: “Bạn có chắc chắn muốn xóa?” | | 2 | Manager sends “Yes” command.  [Alternative 1] | System deletes target plan from the system and displays success message: “Xóa thành công”; |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends cancel command. | System return to main view. |   **Exceptions: N/A**  **Relationships: N/A**  **Business Rules:**   * Plans which have status “False” will be unavailable for being bought by Passenger. * If a plan has not been related to any ticket, system delete this plan completely, otherwise, system sets its status to “False”. | | | |

Table 24: < Manager > Delete credit plan

###### < Manager > Get income report



Figure 32: < Manager > Get income report

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS21** | | | |
| **Use Case No.** | GBTS25 | **Use Case Version** | 1.0 |
| **Use Case Name** | Get income report | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Manager   **Summary:**   * This use case allows Manager can get outcome report.   **Goal:**   * Manager can view total income from selling ticket in a specific date range and also the details list of bought tickets.   **Triggers:**   * Manager sends “Get income report” command.   **Preconditions:**   * Actor logged in system with role “Manager”.   **Post Conditions:**   * **Success:** Income report is showed. * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends “Get income report” command. | System shows “Get income report” view with following information:   * “Từ ngày”: date input, default date is 1st of current month, required. * “Đến ngày”: date input, default date is the current date, required. * “Get report” command. | | 2 | Manager inputs date for both two field. |  | | 3 | Manager sends “Get report” command. | System verifies input information, gets income report based on input date range. System displays report as a table, each row has following information:   * “Ngày giờ”: label * “Thẻ thanh toán”: label * “Khách hàng”: label * “Số tiền”: label   [Exception 1,2,3,4]  [Alternative 1] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Cause | Syste Response | | 1 | If there is no data to show. | System shows a message “Không có dữ liệu trong khoảng thời gian đã chọn”. |   **Exceptions:**  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Manager does not input required fields. | System notices that user need to input all required field.   * “Từ ngày”: System displays warning message “Vui lòng chọn ngày bắt đầu”. * “Đến ngày”: System displays warning message “Vui lòng chọn ngày kết thúc”. | | 2 | Manager inputs start date greater than end date. | System displays error message: “Ngày bắt đầu phải nhỏ hơn ngày kết thúc.” | | 3 | Manager inputs date range over 30 days. | System displays error message: “Khoảng thời gian không vượt quá 30 ngày.” | | 4 | Manager inputs end date greater than current date. | System displays error message: “Ngày bắt đầu phải nhỏ hơn ngày kết thúc.” |   **Relationships:** N/A  **Business Rules:**   * Start date and end date format are “dd/mm/yyyy”. * End date must not be earlier than start date. * Total income would be calculated from all bought tickets match input date range. * End date must not greater than current date. * Date range for getting report must be within 30 days. * System allows Manager to search and sort report on all fields. | | | |

Table 25: < Manager > Get income report

###### < Manager > Create promotion



Figure 33: < Manager > Create promotion

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS22** | | | |
| **Use Case No.** | GBTS26 | **Use Case Version** | 1.0 |
| **Use Case Name** | Create promotions | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Manager   **Summary:**   * This use case allows Manager to create promotion and offer for suggesting Passenger.   **Goal:**   * Manager can create promotion in the system. * Passenger will receive latest promotion from the system when the Auto Handler sends it to them.   **Triggers:**   * Manager sends “Create promotions” command.   **Preconditions:**   * Actor logged in system with role “Manager”.   **Post Conditions:**   * **Success:** New promotion is added to the system. * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Manager sends “Create promotion” command. | System shows “Create promotion ” view with following information:   * “Tên quảng cáo”: free text input, length 6-250, required. * “Nội dung”: html input area, required. * “Save” command. | | 2 | Manager inputs information. |  | | 3 | Manager sends “Save” command. | System verifies input information. Create new promotion with input information and add to system.  [Exception 1,2] |   **Alternative Scenario:** N/A  **Exceptions:**  [Exception 1,2]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Manager does not input required fields. | System notices that user need to input all required field.   * “Tên quảng cáo”: System displays warning message “Vui lòng nhập quảng cáo”. * “Nội dung”: System displays warning message “Vui lòng nhập nội dung”. | | 2 | Manager inputs some fields with wrong format | System notices that user need to input all required field.   * “Tên quảng cáo”: System displays warning message “Vui lòng nhập tên quảng cáo từ 6-250 ký tự”. |   **Relationships:** N/A  **Business Rules:**   * New promotion will be created with input information and its status is “True”. * After Auto Handler sends promotion successfully, the status will be change to “False”. | | | |

Table 26: < Manager > Create promotion

##### Admin

###### < Admin> Overview use case



Figure 34: < Admin> Overview use case

###### < Admin> Search account



Figure 35: < Admin> Search account

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS23** | | | |
| **Use Case No.** | GBTS27 | **Use Case Version** | 1.1 |
| **Use Case Name** | Search account | | |
| **Author** | Doan Minh Duc | | |
| **Date** | September 13, 2016 | **Priority** | High |
| **Actor:**   * Administrator.   **Summary:**   * This use case allows admin to search for accounts.   **Goal:**   * Admin can quickly navigate and view the information of accounts by phone number.   **Triggers:**   * Admin sends Search command.   **Preconditions:**   * Actor logged in the system with role “Administrator”.   **Post Conditions:**   * **Success:** System shows the result. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Admin inputs the phone number then sends Search command. | System displays matched account in a table with these following information for each account:  +“Số điện thoại”: label  +“Vai trò”: label  +”Họ tên”: label  +”Trạng thái”: label  [Alternative 1] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Admin input a none-existed phone number | System shows no results matched. |   **Exceptions: N/A**  **Relationships: N/A**  **Business Rules:**   * Each account has a unique phone number. * System display top 10 account on each page. | | | |

Table 27: < Admin> Search account

###### < Admin> Add account

Figure 36: < Admin> Add account

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS24** | | | |
| **Use Case No.** | GBTS28 | **Use Case Version** | 1.1 |
| **Use Case Name** | Add account | | |
| **Author** | Doan Minh Duc | | |
| **Date** | September 13, 2016 | **Priority** | High |
| **Actor:**   * Administrator.   **Summary:**   * This use case allows admin to create an account in the system.   **Goal:**   * Admin can create new account with specific role such as Admin, Staff, Manager or Passenger.   **Triggers:**   * Admin sends creating account command.   **Preconditions:**   * User logged in the system with the role “Administrator”.   **Post Conditions:**   * **Success:** New account is inserted into the system. * **Fail:** System displays appropriate error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Admin sends creating account command. | System list information of account in system and components:   * “Số điện thoại”: number input, required, length 10-15 * “Vai trò”: select one of the options such as “Quản trị”, “Quản lý”, “Hành khách”, “Nhân viên” ,required. * “Họ tên”: free text input, length 3-80. * Register command. * Cancel command. | | 2 | Admin fills in information for each account.  [Alternative 1, 2] | System validates input information. | | 3 | Admin sends Register command. | System displays a successful message: “Tạo tài khoản thành công”  [Exception 2] |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | 1 | Admin clicks on Cancel button to cancel the register process. | System redirects to admin’s main page. |   [Alternative 2]   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Admin inputs invalid data. | System displays an error message and asks admin to fill the form again. |   **Exception:**  [Exception 2]   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Admin inputs an existed phone number. | System displays an error message “Số điện thoại đã tồn tại trong hệ thống”. |   **Relationships: N/A**  **Business Rules:**   * Phone number is unique on each account. * “Vai trò” is the account’s role with 4 options, “Admin”, “Quản lý”,”Nhân viên”, “Hành khách”. “Admin” will act as an Administrator role in the system. “Quản lý” will act as Manager in the system. “Nhân viên” will act as a Staff role in the system. “Hành khách” will act as a Passenger role in the system. * New account will be created with the status “Đang hoạt động”. * The auto-generated password will be sent to registered phone number. | | | |

Table 28: < Admin> Add account

###### < Admin> Activate/Deactivate account



Figure 37: < Admin> Edit account

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS25** | | | |
| **Use Case No.** | GBTS29 | **Use Case Version** | 1.2 |
| **Use Case Name** | Activate/Deactivate account | | |
| **Author** | Doan Minh Duc | | |
| **Date** | September 13, 2016 | **Priority** | High |
| **Actor:**   * Administrator.   **Summary:**   * This use case allows admin to activate or deactivate an account.   **Goal:**   * The status of an account is updated in the system.   **Triggers:**   * Admin sends Activate/Deactivate account command.   **Preconditions:**   * User logged in the system with the role “Administrator”. * The account must be available in the database.   **Post Conditions:**   * **Success:** Anaccount’s status is updated in the database. * **Fail:** System displays appropriate error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Admin sends “Activate/Deactivate account” command on a specific account | System activates/deactivates the account |   **Alternative Scenario: N/A**  **Exceptions: N/A**  **Relationships: N/A**  **Business Rules:**   * An account’s status can be “Đang hoạt động” or “Đã khóa”. “Đang hoạt động” means the account is activated, it can be used to access the system. “Đã khóa” means the account is disabled, it cannot be used to access the system. | | | |

Table 29: < Admin> Edit account

##### Auto Handler

###### < Auto Handler > Overview use case



Figure 38: < Auto Handler > Overview use case

###### < Auto Handler > Parse bus route



Figure 39: < Auto Handler > Parse bus route

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS27** | | | |
| **Use Case No.** | GBTS30 | **Use Case Version** | 1.0 |
| **Use Case Name** | Parse bus route | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Auto Handler   **Summary:**   * This use case allows Auto Handler to get newest bus route list.   **Goal:**   * Auto Handler can get newest bus route list for adapting changes of bus route in real life.   **Triggers:**   * The current time hit configured time.   **Preconditions:**   * Configured time has been set.   **Post Conditions:**   * **Success:** Newest bus route list is updated to the system. * **Fail:** Write error log.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Auto Handler checks the current time. If the current time hit configured time. Auto Handler sends “Parse bus route” command. | System parses and get newest route list and updates to database.  Write to log file.  [Exception 1] |   **Alternative Scenario:** N/A  **Exceptions:**  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Auto Handler is interrupted | No promotion will be sent. Error details will be tracked in a log file. |   **Relationships:** N/A  **Business Rules:**   * System will parse route list from <http://www.buyttphcm.com.vn/TTLT.aspx> * Sent log file structure:   File name: Promotions.log   |  |  |  |  | | --- | --- | --- | --- | | No | Time | Promotion Name | Result | |  |  |  |  | |  |  |  |  |  * Error log file structure:   File name: AutoHandler\_Error.log   |  |  |  |  | | --- | --- | --- | --- | | No | Time | Auto Handler Name | Error Details | |  |  |  |  | |  |  |  |  | | | | |

Table 30: < Auto Handler > Parse bus route

###### < Auto Handler > Suggest promotion



Figure 40: < Auto Handler > Suggest promotion

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS28** | | | |
| **Use Case No.** | GBTS31 | **Use Case Version** | 1.0 |
| **Use Case Name** | Suggest promotions | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Auto Handler   **Summary:**   * This use case allows Auto Handler to get latest promotions then suggest to Passenger.   **Goal:**   * Passenger will receive latest promotions and offers from system such as credit plans sale off, free ticket, etc.   **Triggers:**   * The current time hit configured time.   **Preconditions:**   * Configured time has been set. * There is at least one promotion with status “True”.   **Post Conditions:**   * **Success:** Passenger receives latest promotion and offer. * **Fail:** Write error log.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Auto Handler checks the current time. If the current time hit configured time. Auto Handler send “Suggest promotions” command. | System gets latest promotions and sends to all passengers. Change related promotion and offer status to “False”.  Write to log file.  [Exception 1] |   **Alternative Scenario:** N/A  **Exceptions:**  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Auto Handler is interrupted | No promotion will be sent. Error details will be tracked in a log file. |   **Relationships:** N/A  **Business Rules:**   * The promotion which will be sent must have status “True”. * After Auto Handler sends promotion & offer successfully, the status will be change to “False”. * If the current time doesn’t hit configured time or no promotion have status “True”, do nothing. * If the current time hit configured time. Auto Handler get latest promotion which have status “True” then send to passengers. * Promotion will be send from 7pm – 9pm for not harassing passengers. * Sent log file structure:   File name: Promotions.log   |  |  |  |  | | --- | --- | --- | --- | | No | Time | Promotion Name | Result | |  |  |  |  | |  |  |  |  |  * Error log file structure:   File name: AutoHandler\_Error.log   |  |  |  |  | | --- | --- | --- | --- | | No | Time | Auto Handler Name | Error Details | |  |  |  |  | |  |  |  |  | | | | |

Table 31: < Auto Handler > Suggest promotion

###### < Auto Handler > Send notification



Figure 41: < Auto Handler > Send notification

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS29** | | | |
| **Use Case No.** | GBTS32 | **Use Case Version** | 1.0 |
| **Use Case Name** | Send notification | | |
| **Author** | HoangDN | | |
| **Date** | 13/09/2016 | **Priority** | Normal |
| **Actor:**   * Auto Handler   **Summary:**   * This use case allows Auto Handler to notify Passenger.   **Goal:**   * Passenger will receive notifications from system when their card’s balance is running out.   **Triggers:**   * Auto Handler sends “Send notification” command.   **Preconditions:**   * Passenger has recently bought a ticket. * Passenger’s card balance is running out.   **Post Conditions:**   * **Success:** Passenger receives notification from system. * **Fail:** Write error log.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Auto Handler is waiting for message. If it get any message from system about one card’s balance is running out. Auto Handler sends “Send notification” command. | System sends notification to Passenger: “Thẻ {Card ID} của bạn sắp hết tiền, vui lòng nạp thêm tiền vào thẻ!”  Write to log file.  [Exception 1] |   **Alternative Scenario:** N/A  **Exceptions:**  [Exception 1]   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Auto Handler is interrupted | No notification will be sent. Error details will be tracked in a log file. |   **Relationships:** N/A  **Business Rules:**   * Card’s balance is running out when the balance is less than 5.000đ * System notifies Passenger one time after they bought a ticket if their card’s balance is running out. * Sent log file structure:   File name: Notifications.log   |  |  |  |  | | --- | --- | --- | --- | | No | Time | Card ID | Result | |  |  |  |  | |  |  |  |  |  * Error log file structure:   File name: AutoHandler\_Error.log   |  |  |  |  | | --- | --- | --- | --- | | No | Time | Auto Handler Name | Error Details | |  |  |  |  | |  |  |  |  | | | | |

Table 32: < Auto Handler > Send notification

#### Emulator

##### < Emulator > Overview use case



Figure 42: < Emulator > Overview use case

##### < Emulator > Write card

Figure 43: < Emulator > Write card

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS30** | | | |
| **Use Case No.** | GBTS33 | **Use Case Version** | 1.1 |
| **Use Case Name** | Write NFC card | | |
| **Author** | Doan Minh Duc | | |
| **Date** | September 13, 2016 | **Priority** | High |
| **Actor:**   * Emulator.   **Summary:**   * This use case allows emulator to insert new card into the system to use the system and to be managed.   **Goal:**   * New UID in inserted into the system, available for using or activating.   **Triggers:**   * Emulator send write card command.   **Preconditions:**   * Valid card.   **Post Conditions:**   * **Success:** Emulator displays a successful message. * **Fail:** Emulator displays an error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Emulator reads the card ID then send write card command | System checks the existence of the UID in the database  Systems insert the card ID into the system then display successful message along with an appropriate sound.  [Exception 1] |   **Alternative Scenario: N/A**  **Exceptions:**  [Exception 1]   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Card ID is already existed in the system | System displays an error message along with an appropriate sound. |   **Relationships: N/A**  **Business Rules:**   * Card ID is unique on each card. * Card ID is inserted into the system with the default status “Chưa kích hoạt”. * Only written card can be used in the system. * Emulator’s screen turns green and plays a single beep sound to indicate a success. * Emulator’s screen turns red and plays a triple beep sound to indicate an error. | | | |

Table 33: < Emulator > Write card

##### < Emulator > Verify ticket

Figure 44: < Emulator > Verify ticket

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – GBTS31** | | | |
| **Use Case No.** | GBTS34 | **Use Case Version** | 1.1 |
| **Use Case Name** | Verify ticket | | |
| **Author** | Doan Minh Duc | | |
| **Date** | September 13, 2016 | **Priority** | High |
| **Actor:**   * Emulator.   **Summary:**   * This use case allows emulator to verify passenger’s card.   **Goal:**   * Verify card by checking the balance then decide to grant access or not.   **Triggers:**   * Passenger taps his/her NFC card on the emulator.   **Preconditions:**   * N/A   **Post Conditions:**   * **Success:** Emulator charges card’s holder for boarding. * **Fail:** Emulator display an error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Emulator reads the ID of the card then send verify ticket command. | System checks the balance mapped with the UID.  +If the balance is equal or more than the cost fare of the bus route, the system will charge the card then display a successful message.  +If the balance is less than the cost fare of the bus route, the system will display an error message. |   **Alternative Scenario:** **N/A**  **Exceptions: N/A**  **Relationships: N/A**  **Business Rules:**   * Card ID is unique on each card. * If the passenger forgets his/her card, he/she can buy a single trip card on the bus or pay in cash. * Emulator’s screen turns green and plays a single beep sound indicated a passenger can pass. * Emulator’s screen turns red and plays a triple beep sound indicated a passenger cannot pass. * Auto Handler pushes a notification to remind passenger to top-up when the money left in the card is less than 5.000 VND. | | | |

Table 34: < Emulator > Verify ticket

## Software System Attribute



### Usability

* The Android application for emulator requires 10 minutes training for staff.
* The Android application for passenger will take 10-30 minutes to get used to the mobile system completely

### Reliability

* The number of sending notification failure is 1 time per 1000 notifications.
* Timer tasks run at configured time with 100% execution rate.

### Availability

* System is available 24 hours per day and 7 days per week.
* System should take at most 5 hours per month for backup or repairing.

### Security

* Privacy: Each role of user has a specific permission to interact with system.
* System requires SMS verification for all users.
* System always checks authorization and authenticated before doing anything.
* Only admin can grant permission to other roles.

### Maintenanability

* The system is divided into separated modules.
* The code is easy to maintain and upgrade.

### Portability

* The web application is running on Windows Server 2008 or above.
* Mobile application for passenger and emulator runs on Android API greater than 4.1.
* Providing easy installation.

### Performance

* Web appication handles the task within 10 seconds.
* System response time of SMS message depend on telecomunications infrastructurs and server.
* Emulator reads data from NFC card in less than 5 second within the distance between 0 centimeter and 1 centimeter in the condition with no obstacle.

## Conceptual Diagram

1. 

Figure 45: Conceptual Diagram

**Data 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. |
| Offer Subscription | Contains all info about offer subscription. |
| User Subscription | Contains all info about user subscription to specify who subscribe which offer subscription. |

Table 35: Conceptual Diagram Data Dictionary