**MINISTRY OF EDUCATION AND TRAINING**

**FPT UNIVERSITY**

Capstone Project Document

**iMuseum**

|  |  |
| --- | --- |
| **Group 01** | |
| **Group members** | Nguyễn Tấn Phát (Team Leader)  Trần Quang Tùng (Team Member)  Võ Thanh Hiếu (Team Member)  Phan Quốc Hùng (Team Member) |
| **Supervisor** | Mr. Kiều Trọng Khánh |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | iBM |

-Ho Chi Minh City, **09/05/2016**-

*This page is intentionally left blank*

# Report No.3 Software Requirement Specification

## User Requirement Specification

|  |  |
| --- | --- |
| **Actor** | **Definition** |
| **1. Guest** | Person uses mobile application or website but not login into system |
| **2. Authenticated User** | Person who has logged in into the system, users can use some functions base on their role |
| **3. Staff** | Security, who tracks historic items in a specific area |
| **4. Admin** | Administrator, who monitors, configures and manipulates the information in the system |
| **5. Visitor** | Person uses mobile application and want to visit the museum |
| **6. Auto Handler** | Is an actor, a part of the system, do every automatic function according to set up |

Table 11: User Definition

### Guest Requirement

* *Login:* guest uses usename and password to login into the system to become staff or admin and use some function base on the role after login.

### Staff Requirement

* *Scan Historic Item:* staff uses mobile application and go through which historic item needs to be tracked.
* *Track Historic Item:* staff uses mobile application for detect which historic item is moved.

### Admin Requirement

* *Get log of system:* admin can use web application for get all log of the system to review.
* **Manipulate historic item information**
  + *Insert historic item information:* add a new historic item information to system.
  + *Search historic item information:* admin can search the historic item information in the system for getting, updating or deleting that information.
  + *Update historic item information:* update an existed historic item information in system.
  + *Delete historic item information:* remove an existed historic item information in system.
* **Manipulate beacon information**
  + *Insert beacon information:* add a new beacon information to system. After adding a new beacon information, admin can map a beacon information with a historic item information.
  + *Search beacon information:* admin can search the beacon information in the system for getting, updating, deleting that information or mapping beacon information with a specific historic item information.
  + *Update beacon information:* update an existed beacon information in system.
  + *Delete beacon information:* remove an existed beacon information out of system.

### Visitor Requirement

* *Get museum map:* visitor can get the entire museum map images.
* *Get historic item information:* visitor can use mobile application and get the detail information of a historic item. The historic item information includes historic item name, description, related items and near-by items.
* *Report historic item information:* visitor can use the mobile application to feedback the admin about the historic item information.
* *Save historic item information:* store the information of historic item in local mobile storage for offline viewing.
* *Get favorite historic item information:* visitor can use mobile application and get the detail information of a historic item, which visitor saved before.
* *Get related item location:* get the location of a specific historic item in the museum

### Authenticated user Requirement

* *Logout:* when finish all activities at website staff and admin can log out of system.

### Auto Handler Requirement

* *Simulator generate ticket code:* system generate ticket code for visitor.
* *Define related historic item:* system define which historic item in the system relate to new historic item.
* *Detect in-range historic item:* system will detect all in-range historic item when visitor move in the museum.
* *Detect moved historic item:* system will detect which historic item is moved.
* *Notify the report of visitor for admin:* system notify for admin know that visitor has just send a feedback about a historic item.
* *Speak historic item information:* system will read aloud the description of historic item information.
* *Crawl historic item information:* Periodically, the system will get all historic item information from another website and save to database.

## System Requirement Specification

### External Interface Requirement

#### User Interface

* The user interface use English and Vietnamese language in website and android mobile application.
* The user interface for website display best on 1024x768 pixels – screen size.
* The user interface for android application is designed base on material design and display best on 480x800 pixels – screen size

#### Hardware Interface

* Android smartphone supports Bluetooth 4.0 low energy
  + OS: Android (above 4.3
  + Chipset: Snapdragon 400 1.7GHz Dual Core
  + RAM: 1GB
* Estimote beacons. (them reference)

#### Software Interface

* Web application: work with Chrome (v47 or above), Internet Explorer (v10 or above), Firefox (v43 or above)
* Mobile application: Android operation system (v4.4 or above) integrate with Estimote beacon SDK.

#### Communication Protocol

* Use HTTP protocol 1.1 for communication between the web browser and the web server
* Use HTTP protocol 1.1 for communication between the mobile application and the web server
* Use Bluetooth 4.0 (BLE – Bluetooth Less Energy) protocol for communication between the estimate beacons with the mobile application

### System Overview Use Case

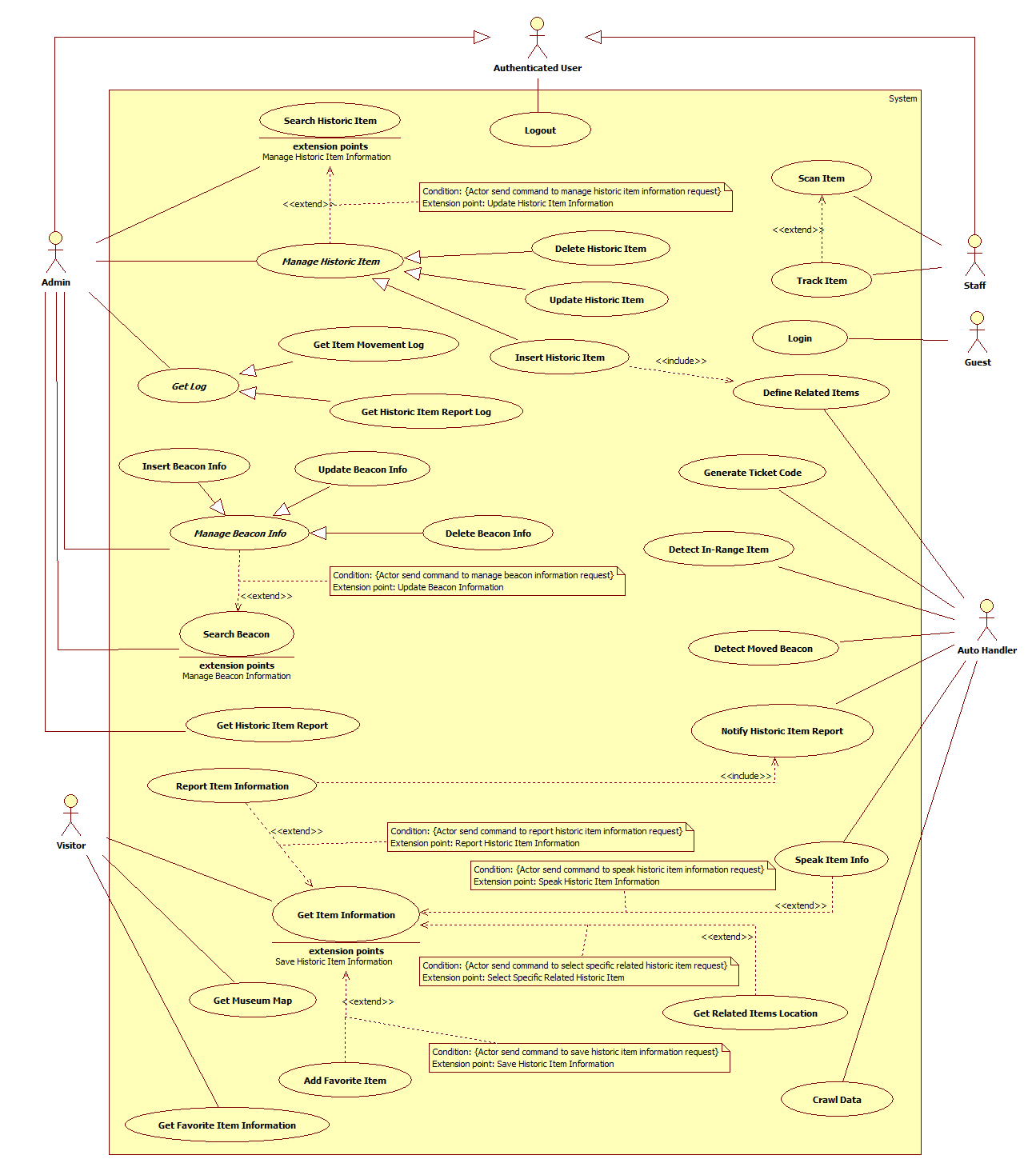


Figure 2: System Overview Use Case

### List of Use Case

#### <Guest> Overview use case

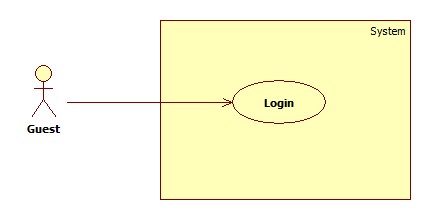


Figure 3: <Guest> Overview Use Case

##### <Guest> Login

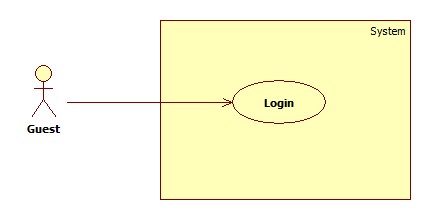


Figure 4 : <Guest> Login

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_G01** | | | |
| **Use Case No.** | UC\_G01 | **Use Case Version** | 1.0 |
| **Use Case Name** | Login | | |
| **Author** | HungPQ | | |
| **Date** | 23/05/2016 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use case allows guess to log in to system. * This use case applies for both web and mobile.   **Goal:**   * Guest logs in to system and be able to use functions based on role.   **Triggers:**   * Guest sends the login command.   **Preconditions:** N/A  **Post Conditions:**   * **Success:**    + If guest logged in as Staff, system will display Staff view   + If guest logged in as Admin, system will display Admin view   + If guess input wrong credential information, system will display message: “Invalid username or password!” * **Fail:** System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Guest goes to login view. | System requires information:   * *Username* * *Password* | | 2 | Guest inputs information | System show the input information of guest in suitable text input. | | 3 | Guest sends login command. | Guest will login system with their specified role.  [Alternative 1]  [Alternative 2]  [Exception 1]  [Exception 2] |   **Alternative Scenario 1:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Guest inputs wrong credential information | System show message “Invalid username or password!” |   **Alternative Scenario 2:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 2 | Guest missed to input information | System show message “Please input username and password!” |   **Exceptions 1:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Guest send login command without Internet connection | System shows error message “Please check your Internet connection!” |   **Exceptions 2:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Server error | System shows error message “Temporary cannot login, please try again later!” |   **Relationships:** N/A  **Business Rules:**   * Logged in users have 2 roles: Staff and Admin. * Staff can log in at mobile application for Staff. * Admin can log in at both mobile application for Staff and BOM website. | | | |

Table 12: Login specification

#### <Visitor> Overview use case

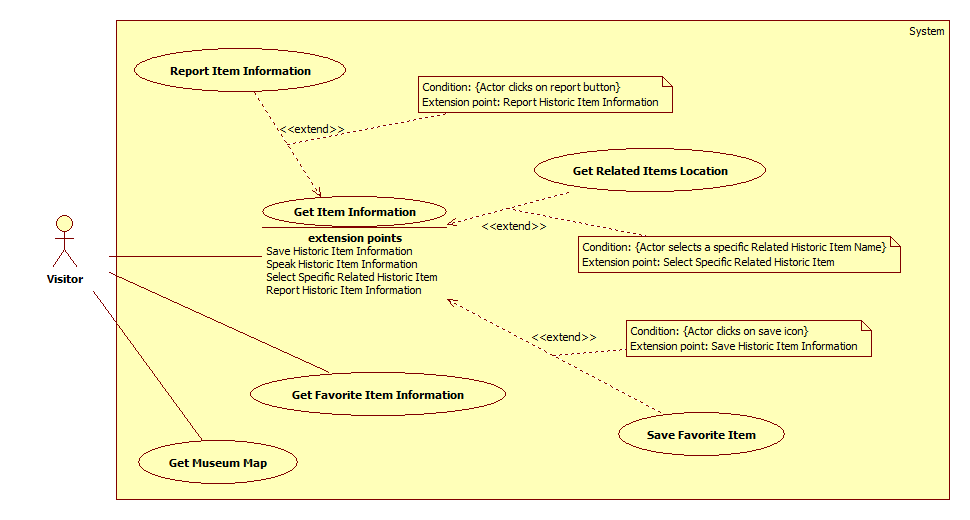


Figure 7: <User> Overview Use Case

##### <Visitor> Get Museum Map

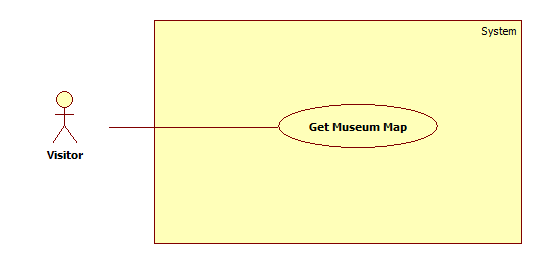


Figure 8: <Visitor> Get Museum Map

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_V01** | | | |
| **Use Case No.** | UC\_V01 | **Use Case Version** | 1.0 |
| **Use Case Name** | Get Museum Map | | |
| **Author** | PhatNT | | |
| **Date** | 24/05/2016 | **Priority** | Normal |
| **Actor:**   * Visitor   **Summary:**   * This use case allows user to get all overview map of the museum. * This use case applies for Visitor – Android Mobile Application.   **Goal:**   * Visitor can get and see all the image map of all floor in the museum.   **Triggers:**   * Visitor send “Get Museum Map” command.   **Preconditions:**   * Actor has accessed the system under Visitor role.   **Post Conditions:**   * **Success:**    + If there are image maps in the system, image of all floors in the museum will be displayed.   + If there is no image map in the system, message “The museum has no map” will be shown. * **Fail:** fail message will be shown depend on a specific exception case.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Visitor selects Museum Map segment. |  | | 2 | Visitor sends Get Museum Map command. (There is at least an image map in the system) | - The progress dialog will be shown.  - System will get and show the list of floor images into tabs.  - The progress dialog will be dismissed.  - One tabs include:   * Floor image (image) * Floor name (label)   [Alternative 1]  [Alternative 2]  [Exception 1]  [Exception 2] |   **Alternative Scenario 1:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Visitor sends Get Museum Map command. (There is no image map in the system) | - The progress dialog will be shown.  - The message “There is no image map” will be shown.  - The progress dialog will be dismissed. |   **Alternative Scenario 2:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Visitor cancels Get Museum Map while system is processing. | - The progress dialog will be dismissed.  - System stops get the list of floor images.  - The message “You have cancelled get museum map” will be shown. |   **Exceptions 1:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Visitor selects Museum Map segment and send Get Museum Map command without Internet Connection. | - The message “Please check your internet connection!” will be shown. |   **Exceptions 2:**   |  |  |  | | --- | --- | --- | | Step. | Cause | System Response | | 1 | Server error | - System shows error message “Temporary cannot process you request, please try again later!” |   **Relationships:** N/A  **Business Rules:**   * System will download and save all museum map in internal memory. * System will not download all museum map in the next time request if museum map not change. | | | |

Table 13: Get Museum Map Specification

##### <Visitor> Get Favorite Item Information

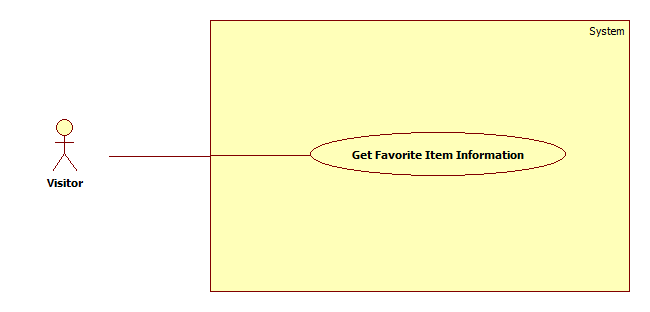


Figure 9: <Visitor> Get Favorite Item Information

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_V05** | | | |
| **Use Case No.** | UC\_V05 | **Use Case Version** | 1.0 |
| **Use Case Name** | Get Favorite Item Information | | |
| **Author** | PhatNT | | |
| **Date** | 28/05/2016 | **Priority** | Normal |
| **Actor:**   * Visitor   **Summary:**   * This use case allows user to get all information of a specific historic item, which visitor saved before. * This use case applies for Visitor – Android Mobile Application.   **Goal:**   * Visitor can get and see all information of a specific historic item that they saved before.   **Triggers:**   * Visitor send “Get Favorite Item Information” command.   **Preconditions:**   * Actor has accessed the system under Visitor role. * There is at least one favorite item.   **Post Conditions:**   * **Success:** all historic item information will be displayed. * **Fail**: message will be shown depend on each specific exception case.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Visitor selects a historic item and send Get Favorite Item Information command. | - The progress dialog will be shown.  - System will get all historic item information and show them on the screen.  - The progress dialog will be dismissed.  - All historic item information include:   * Historic item image * Historic item name * Historic item description * List of related Historic item. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * When the visitor moves in the region of the historic item, the system will cache the name of historic item to show at a next time without sending a request to get name of historic item. | | | |

Table 14: Get Item Information Specification

#### <Admin> Overview use case

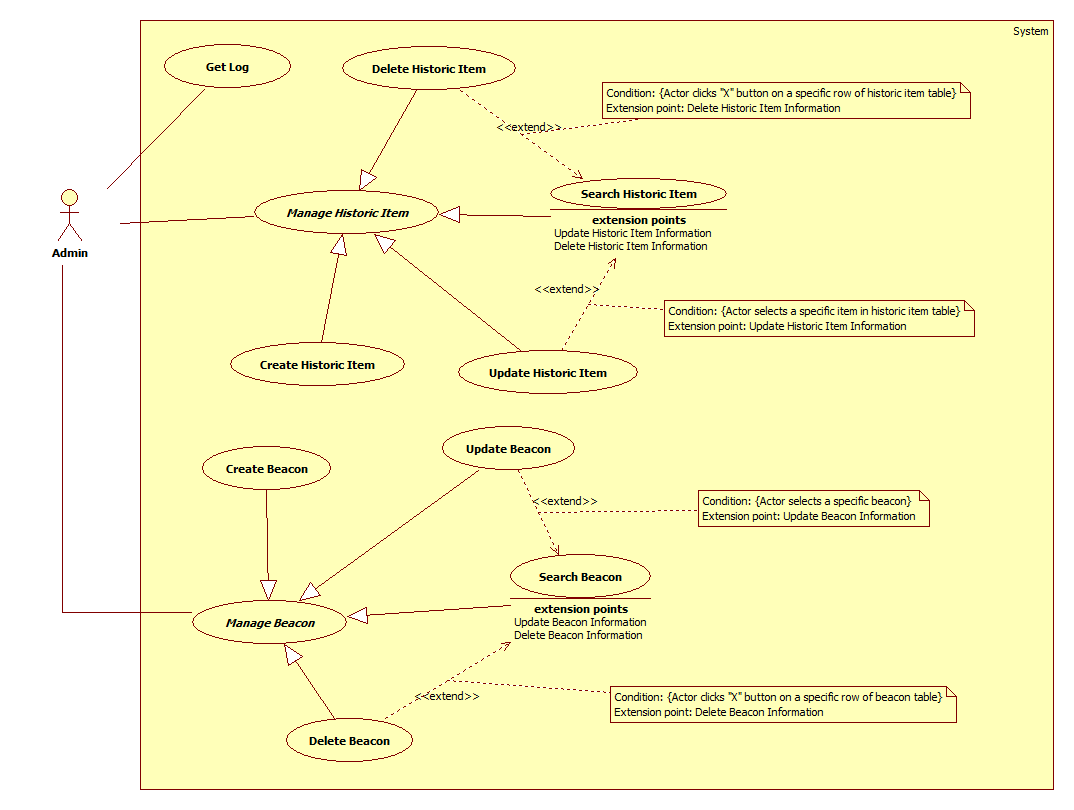


Figure 11: <Admin> Overview Use case

##### <Admin> Get Beacon Log

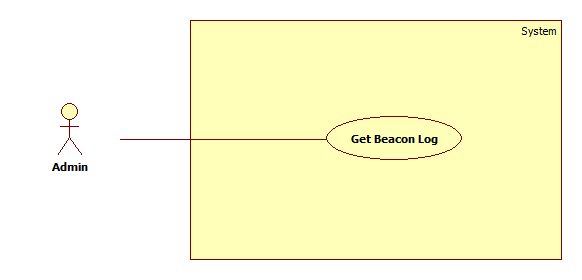


Figure 12: <Admin> Get Beacon Log

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_A01** | | | |
| **Use Case No.** | UC\_A01 | **Use Case Version** | 1.0 |
| **Use Case Name** | Get Beacon Log | | |
| **Author** | HungPQ | | |
| **Date** | 23/05/2016 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows Admin to get the list of Beacon movements log. * This use case applies for Admin web application.   **Goal:**   * Admin can see the list of Beacon movements’ log.   **Triggers:**   * Admin sends “Get beacon movements log” command.   **Preconditions:**   * Actor logged in system under Admin role.   **Post Conditions:**   * **Success:** Beacon movements Log is successfully shown to Admin. * **Fail:** System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Admin sends Get Beacon Movements Log command. | - System gets and shows the log of Beacons movements.  One record in list include (in order):   * Beacon ID (label) * Beacon Name (label) * Beacon status (label) * Notification time (label)   [Alternative 1**]**  [Exception 1]  [Exception 2] |   **Alternative Scenario 1:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Beacon movements log is empty. | System displays message “There is no problem to show. Everything looks fine!” |   **Exceptions 1:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Admin send Get Beacons Movement Log command without Internet Connection. | System displays message “Please check your internet connection!” |   **Exceptions 2:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Server error | System shows error message “Temporary cannot process your request, please try again later!” |   **Relationships:** N/A  **Business Rules:**   * The list is sorted in descending order by time. | | | |

Table 17: Get Beacon Log specification

##### <Admin> Create Historic Item

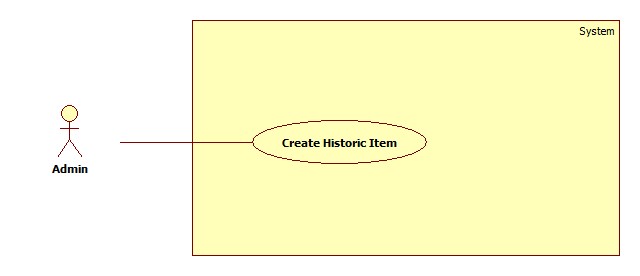


Figure 12: <Admin> Create Historic item

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_A03** | | | |
| **Use Case No.** | UC\_A03 | **Use Case Version** | 1.0 |
| **Use Case Name** | Insert Historic Item | | |
| **Author** | HungPQ | | |
| **Date** | 24/05/2016 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows admin to insert a new item. * This use case applies for Admin web application.   **Goal:**   * Admin inserts a new item into system database.   **Triggers:**   * Admin sends “Insert historic Item” command.   **Preconditions:**   * Actor logged in under Admin role.   **Post Conditions:**   * **Success:** New item is successfully inserted in system database. * **Fail:** System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Admin accesses “Insert historic item” view. | System lists out information of historic items:   * *Item image URL*: text field, required. * *Item name*: text field, required * *Item description*: text area, required * *Item location*: drop-down list (show existed location in database), required. * *Beacon Name*: drop-down list (show available Beacons in database), required. * *Beacon UUID*: label (read-only), required, autocomplete when admin choose beacon name. * *Beacon Mac Address*: label (read-only), required, autocomplete when admin choose beacon name. * *Beacon major ID*: label (read-only), required, autocomplete when admin choose beacon name. * *Beacon minor ID*: label (read-only), required, autocomplete when admin choose beacon name. | | 2 | Admin enters information of items.  [Alternative 1]  [Exception 1]  [Exception 2]  [Exception 3]  [Exception 4]  [Exception 5] |  | | 3 | Admin sends “Insert historic Item” command. | System saves item in system database.  System redirects to “Historic Item List” page. |   **Alternative Scenario 1:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Admin cancels process. | System redirects to previous page and does nothing. |   **Exceptions 1:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Admin enters wrong image URL | System shows messages “The Item Image URL is not an image” |   **Exceptions 2:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Admin ignores one of required field and requests create historic item information. | System shows messages “Please input all required fields” |   **Exceptions 3:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | The name of the item is duplicated. | System shows messages “This item is already existed!” |   **Exceptions 4:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Admin sends create historic item command without Internet Connection. | System shows message “Please check your internet connection!” |   **Exceptions 5:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Server error | System shows error message “Temporary cannot process your request, please try again later!” |   **Relationships:** N/A  **Business Rules:**   * One item should only match with one beacon. * Beacon’s UUID, Major ID and Minor ID are set on Estimote Cloud. * A correct Image URL must contain “.jpg”, “.jpeg”, “.png”, “.tif” or “.gif”. | | | |

Table 17: Create Historic item specification

##### <Admin> Create Beacon

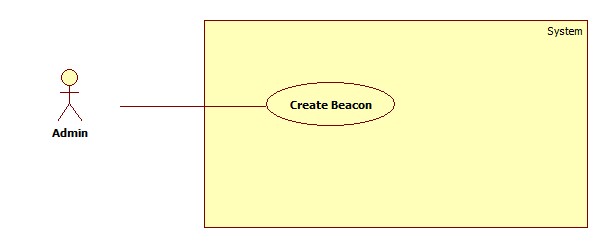


Figure 12: <Admin> Create Beacon

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_A07** | | | |
| **Use Case No.** | UC\_A07 | **Use Case Version** | 1.0 |
| **Use Case Name** | Insert Beacon | | |
| **Author** | HungPQ | | |
| **Date** | 24/05/2016 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows admin to Insert a new Beacon. * This use case applies for Admin mobile application.   **Goal:**   * Admin can add a new Beacon information into system database.   **Triggers:**   * Admin sends “Insert Beacon” command.   **Preconditions:**   * Actor logged in under Admin role.   **Post Conditions:**   * **Success:**    + Beacon is successfully added to system database.   + Message “Insert new beacon successful!” will be shown. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Admin accesses “In-range Beacon List”  [Alternative 1] | System shows the list of In-Range Beacons, sorted in ascending order by distance. | | 2 | Admin chooses a Beacon and accesses “Beacon details view” | System lists out information of Beacon:   * Beacon Name: text field * Beacon Mac Address: label label (read-only) * Beacon UUID: label label (read-only) * Beacon major ID: label label (read-only) * Beacon minor ID: label label (read-only) | | 3 | Admin enters Beacon name.  [Exception 1] |  | | 4 | Admin sends create command.  [Exception 2]  [Exception 3]  [Exception 4] | System saves Beacon in system database.  System redirects to “In-range Beacon List” page.  [Exception 2]  [Exception 3] |   **Alternative Scenario 1:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | There is no in-range Beacon | System shows messages “There is no Beacon around” |   **Exceptions 1:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | The Beacon Name field is blank | System shows messages “Beacon Name must not be blank.” |   **Exceptions 2:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | The entered Beacon is existed in system database | The “Update Beacon” function will be called to update Beacon new information. |   **Exceptions 3:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Admin sends create command without Internet Connection. | System shows message “Please check your internet connection!” |   **Exceptions 4:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Server error | System shows error message “Temporary cannot process your request, please try again later!” |   **Relationships:** N/A  **Business Rules:**   * The Beacon Mac address, UUID, major ID and minor ID is automatically fill when Admin chooses a Beacon. Those information was set on Estimote Beacon Management page. * The Beacon Name should present a historic item. | | | |

Table 17: Create Beacon specification

##### <Admin> Update Beacon

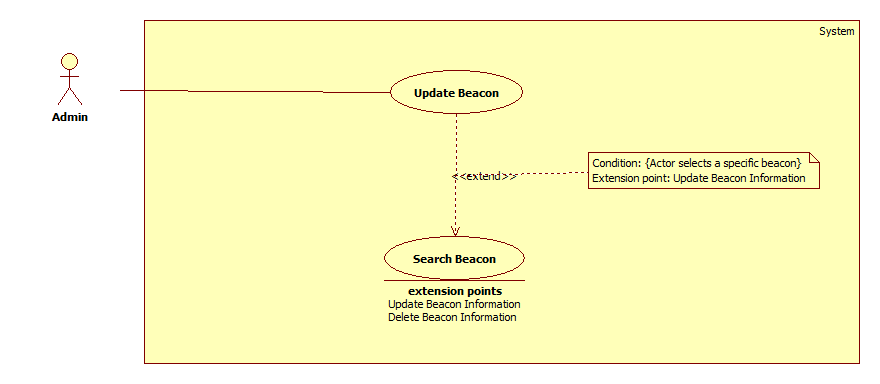


Figure 12: <Admin> Update Beacon

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_A08** | | | |
| **Use Case No.** | UC\_A08 | **Use Case Version** | 1.0 |
| **Use Case Name** | Update Beacon | | |
| **Author** | HungPQ | | |
| **Date** | 24/05/2016 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows admin to update an existed Beacon information. * This use case applies for Admin mobile application.   **Goal:**   * Admin can update the beacon information to system database.   **Triggers:**   * Admin sends “Update Beacon” command.   **Preconditions:**   * Actor logged in under Admin role. * Chosen beacon must be existed in system database.   **Post Conditions:**   * **Success:**    + Beacon information is successfully updated in system database   + “Update beacon successfully!” message will be shown. * **Fail:** System shows error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Admin accesses “In-range Beacon List”  [Alternative 1] | System show the list of In-Range Beacons, sorted in ascending order by distance. | | 2 | Admin choose a Beacon and get beacon details | System lists out information of Beacon:   * Beacon Name: text field * Beacon Mac Address: label (read-only) * Beacon UUID: label (read-only) * Beacon major ID: label (read-only) * Beacon minor ID: label (read-only) | | 3 | Admin edits Beacon Name.  [Exception 1] |  | | 4 | Admin sends update command. | System saves Beacon in system database.  System redirects to “In-range Beacon List” page.  [Exception 2]  [Exception 3] |   **Alternative Scenario 1:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | There is no in-range Beacon | System shows messages “There is no Beacon around” |   **Exceptions 1:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | The Beacon Name field is blank | System shows messages “Beacon Name must not be blank.” |   **Exceptions 2:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Admin sends create command without Internet Connection. | System shows message “Please check your internet connection!” |   **Exceptions 3:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Server error | System shows error message “Temporary cannot process your request, please try again later!” |   **Relationships:**  **Business Rules:**   * The Beacon Mac address, UUID, major ID and minor ID is automatically fill when Admin chooses a Beacon. Those information was set on Estimote Beacon Management page. * The Beacon Name should present a historic item. | | | |

Table 17: Update Beacon specification

#### <Authenticated user> Overview use case

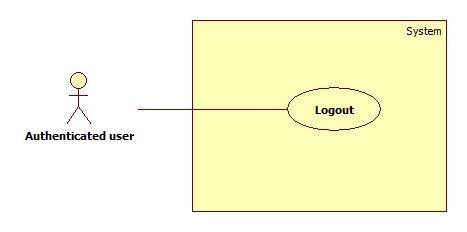


Figure 16: <Authenticated user> Overview Use Case

##### <Authenticated user> Logout

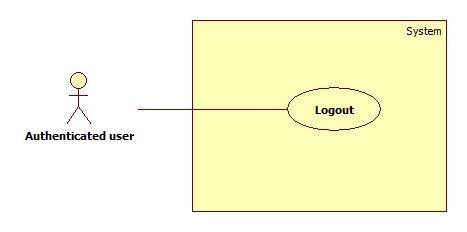


Figure 17: <Authenticated user> Logout

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_U01** | | | |
| **Use Case No.** | UC\_U01 | **Use Case Version** | 1.0 |
| **Use Case Name** | Logout | | |
| **Author** | HungPQ | | |
| **Date** | 23/05/2016 | **Priority** | Medium |
| **Actor:**   * Authenticated user (include Staff and Admin)   **Summary:**   * This use case allows authenticated user to log out of the system. * This use case applies for both web and mobile.   **Goal:**   * Authenticated user logs out the system, the session is killed.   **Triggers:**   * Authenticated User sends log out command. * Actor send command after killing session (for web)   **Preconditions:**   * User logged in the system.   **Post Conditions:**   * **Success:** System displays the login page. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User sends logout command. | - System clears session and log user out of the system.  - System displays login view.  [Alternative 1]  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User sends any command after not available for too long (for web application only) | System check the session state then takes user out of system.  System display login view. |   **Exceptions 1:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | User send logout command without Internet connection | - System shows error message “Please check your Internet connection!” |   **Relationships:** N/A  **Business Rules:**   * If authenticated user is not available longer than 30 minutes, they will see the login view when they be back. * After logout, role “Authenticated User” will become “Guest” | | | |

Table 20: Logout specification

#### <Auto Handler> Overview use case

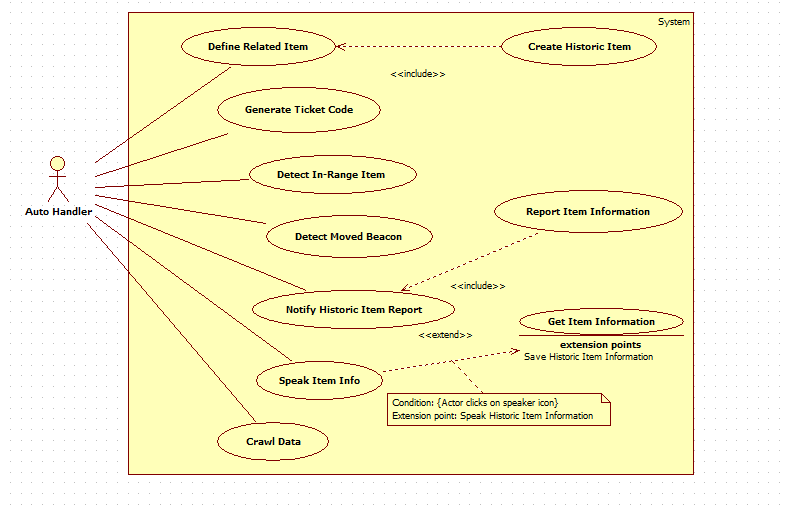
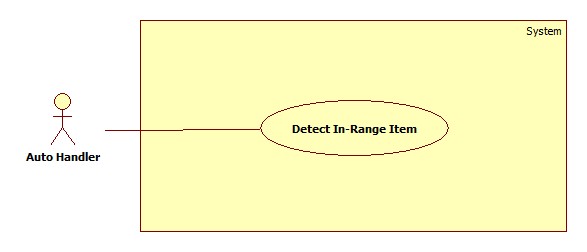


Figure 16: <Auto Handler> Overview Use Case

##### < Auto Handler > Detect In-Range Item



#### <Staff> Overview use case

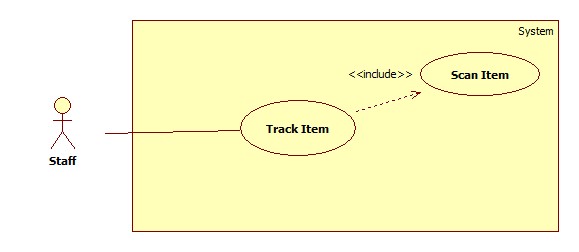


Figure 3: < Staff > Overview Use Case

##### < Staff> Track Item

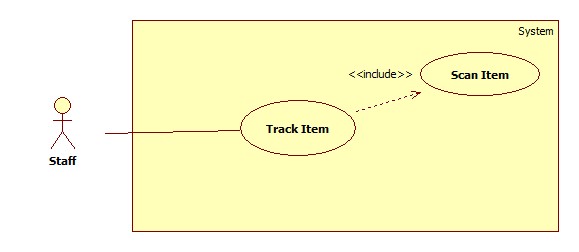


Figure 4: < Staff > Track Item

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC001** | | | |
| **Use Case No.** | 001 | **Use Case Version** | 2.0 |
| **Use Case Name** | Track Item | | |
| **Author** | HieuVT | | |
| **Date** | 25/05/2016 | **Priority** | Normal |
| **Actor:**   * Staff   **Summary:**   * This use case allows the staff who is working to track historic items which are managed by him when it is moved is allowed or isn’t allowed.   **Goal:**   * Staff tracks historic item when it is moved.   **Triggers:**   * System detected moved items and sent notification to staff.   **Preconditions:**   * Guest login in “iMuseum” mobile application with staff role. * Staff scanned all historic items which are managed by him.   **Post Conditions:**   * **Success:** Notification is sent to staff. * **Fail:** Error massage is sent to staff.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 |  | System show list of moved item in screen of staff. | | 2 | Staff click “Allow” button in screen. | System show “Item ”.  [Exception 1] | | 3 | Auto handler receives list of related historic items and send it to admin. |  |   **Exceptions 1:**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | System can’t not find set of related item base on new item’s name. | System log error message. |   **Relationships:** Create Item use case  **Business Rules:**   * After defining related item, all data will be stored in database of system. | | | |

Table 13: Register specification

## Software system attribute

### Usability

#### Graphic user interface

#### Usability

#### Installation

### Reliability

### Availability

### Security

### Maintainability

### Portability

### Performance

## Conceptual diagram

Figure 21: Conceptual diagram

|  |  |
| --- | --- |
| Entity Data dictionary: describe all content of all entities | |
| **Entity Name** | **Description** |
| Admin | Contain the admin information |
| Staff | Contain the staff information |
| Visitor | Contain the visitor information |

Table 25: Conceptual diagram data dictionary