

2.3.1<Member> Overview Use case



Figure 1: <Member> Overview Use case

**2.3.1.1 <Member>** **Edit profile Use Case Diagram**



Figure 2: **<Member> Edit profile Use case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – LRA001** | | | |
| **Use Case No.** | LRA001 | **Use Case Version** | 1.0 |
| **Use Case Name** | Edit profile | | |
| **Author** | HonNV | | |
| **Date** | 17/05/2015 | **Priority** | Normal |
| **Actor:**   * Member   **Summary:**   * This use case allows members to edit their profile in the system.   **Goal:**   * Profile will be updated to the system.   **Triggers:**   * Members wants to edit their profile. * From the sidebar:   + “Manage profile” tag -> “Edit profile” subtag.   **Preconditions:**   * User must login the system with Member or Admin role.   **Post Conditions:**   * **Success:** Profile will be updated to the database. * **Fail:** Show error message on the current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Manage profile” tag. | Navigate to “Manage profile” tag which contains:   * “Edit profile” tag. * “Change password” tag. | | 3 | Member clicks “Edit profile” tag. | Navigate to “Edit profile” page which contains:   * “Username”: label. * Username: textbox, min length: 5, max length: 100, disabled. * “Email”:label. * Email:texboxt, regular expression: /^[\_a-z0-9-]+(\.[\_a-z0-9-]+)\*@@[a-z0-9-]+(\.[a-z0-9-]+)\*(\.[a-z]{2,4})$/. * “Save change”: button. | | 4 | Member fills data to the form.  Member clicks “Save change” button. | Update profile to the database.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | |  | N/A | N/A |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Invalid email format. | “Email” textbox border color is red. |   **Relationships:** N/A  **Business Rules:**   * Member’s profile is updated. | | | |

Table1: <Member> Edit profile

**2.3.1.2 <Member>** **Change password Use Case Diagram**



Figure 3: <Authorized user> Change Password

Use Case Specification

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – LRA002** | | | |
| **Use Case No.** | LRA002 | **Use Case Version** | 1.0 |
| **Use Case Name** | Change password | | |
| **Author** | HonNV | | |
| **Date** | 17/05/2015 | **Priority** | Normal |
| **Actor:**   * Member.   **Summary:**   * This use case allows Member to change password.   **Goal:**   * Member changes password successfully.   **Triggers:**   * Member logins to website. * Member clicks “Change password” tag.   **Preconditions:** Member has register into the system.  **Post Conditions:**   * **Success:** New password has been updated into database. * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Member clicks “Change password” tag. | Show “Change password ” page with includes:   * Old passwrod: textbox, password box, min length: 6, max length: 20, required. * New password: textbox, password box, min length: 6, max length: 20, required. * Confirm new password: textbox, password box, min length: 6, max length: 20, required. * “Save change”: button. | |  | Member clicks “Save change” button [Alternative 1] | Show successful message and return to home page. [Exception 1,2, 3,4,5,6] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | |  | N/A | N/A |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Length of password is not in range. | Show error message: “Password length must be between 6-20 characters!” | | 2 | “Old password” is wrong. | Show error message:” Wrong password.” | | 3 | “Old password” is a blank. | Show error message:” Please enter a password” | | 4 | “New password” is a blank. | Show error message:” Please enter your new password” | | 5 | “Confirm new password” is a blank. | Show error message:” Please enter your confirm new password” | | 6 | “New password” and “Confirm new password” is not similar. | Show error message: “New password and Confirm new password do not match”. |   **Relationships:** Manage profile.  **Business Rules:**   * New password has been updated into database. | | | |

Table2: <Member> Change Password

**2.3.1.3 <Member>** **Retrieve password Use Case Diagram**



Figure 4: **<Member> Retrieve password Use case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – LRA003** | | | |
| **Use Case No.** | LRA003 | **Use Case Version** | 1.0 |
| **Use Case Name** | Retrieve password | | |
| **Author** | HonNV | | |
| **Date** | 17/05/2015 | **Priority** | Normal |
| **Actor:**   * Member   **Summary:**   * This use case allows member to retrieve password in the system.   **Goal:**   * password will be retrieve.   **Triggers:**   * Members wants to retrieve password. * Member clicks “Sent” button.   **Preconditions:**   * N/A   **Post Conditions:**   * **Success:** Password will be retrieved to member. -> New password will be created and sent to member. * **Fail:** Show error message on the current page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Forgot password” link. | Navigate to “Forgot password” page which contains:   * Email:regular expression: /^[\_a-z0-9-]+(\.[\_a-z0-9-]+)\*@@[a-z0-9-]+(\.[a-z0-9-]+)\*(\.[a-z]{2,4})$/ required. * “Sent”: button. | | 2 | Member fills data to the email.  Member clicks “Sent” button. | Insert new password into database. New Password is sent to email.  [Exception 1, 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | ctor Action | System Response | |  | N/A | N/A |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Invalid email format. | “Email” textbox border color is red. | | 2 | Email does not exist in the system. | “Email” textbox border color is red. |   **Relationships:** N/A  **Business Rules:**   * The new password is saved to the system. | | | |
|  | | | |

Table 3: < Member > Retrieve password

**2.3.1.4 <Member>** **Logout Use Case Diagram**



Figure 36: < Member > Log out

Use Case Specification

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – LRA004** | | | |
| **Use Case No.** | LRA004 | **Use Case Version** | 1.0 |
| **Use Case Name** | Logout | | |
| **Author** | HonNV | | |
| **Date** | 17/05/2015 | **Priority** | Normal |
| **Actor:**   * Member.   **Summary:**   * This use case allows Member to log out of the system.   **Goal:**   * Member logs out of the system. Member’s current session is destroyed.   **Triggers:**   * Member wants to log out. * Member clicks “Logout” link.   **Preconditions:**   * Member has logged in to the system.   **Post Conditions:**   * **Success**: Member’s current session is destroyed. Redirect to homepage. * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Member clicks “Logout” link. | * Destroy Member’s session. Redirect to homepage. [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | |  | N/A | N/A |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Fail to destroy Member’s session | Redirect to error page. |   **Relationships:** N/A  **Business Rules:**   * Member clicks “Logout” link, * Member’s role will be changed to guest. | | | |

Table4: < Member > Log out

##### <System> Auto Parse Data

Use Case Diagram



Figure 9: <System> Auto Parse Data

Use Case Specification

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – LRA005** | | | |
| **Use Case No.** | LRA005 | **Use Case Version** | 1.0 |
| **Use Case Name** | Auto Parse Data | | |
| **Author** | HonNV | | |
| **Date** | 17/05/2015 | **Priority** | High |
| **Actor:**   * System.   **Summary:**   * System can parse data automatically from many websites at specified time.   **Goal:**   * Get data from many websites.   **Triggers:**   * The time hits configured time.   **Preconditions:**   * Parse time has been configured.   **Post Conditions:**   * **Success:** New data is inserted to database. * **Fail:** Nothing is changed in the database.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Server checks the current time. If it hits configured time, parse process starts. | * Send request to the parsed link. * Fetch data from the response based on the inputted XPaths. * Validate data [Exception 1]. * If data is valid, insert to database [Alternative 1]. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Server checks the current time. If it hits configured time, parse process starts. | * If fetched product is already in the database, update its information. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Data is invalid. | * Don’t insert to database. |   **Relationships:** N/A  **Business Rules:**   * If product’s name is empty, consider it invalid. | | | |

Table 5: <System> Auto Parse Data