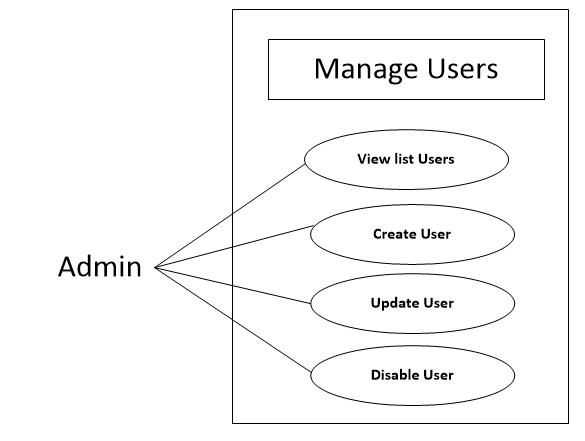
**REVIEW 2**

* 1. **Manage user**

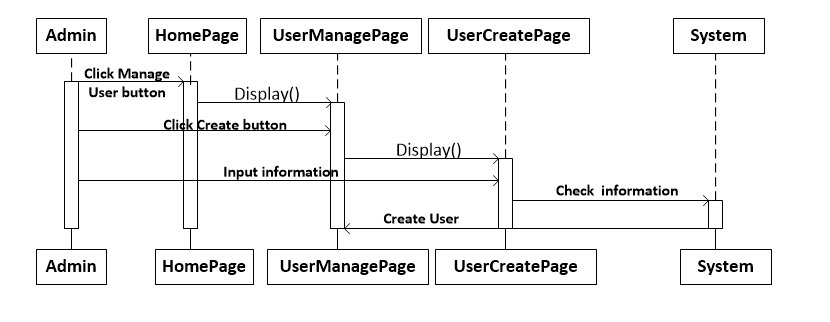
****

* 1. **View List, Detail user**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | View list user | |
| **Actors** | Admin | |
| **Description** | Actor can view list user account | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Return manage list user page | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Manage User’ button in Actor’s Home page menu.  3. Actor click Detail button | System Responses:  2. System redirects to UserManage page and displays List user form with the following control :  - ‘UserName’ label  - ‘FullName’ label  - ‘Email’ label  - ‘Role’ label  - ‘Department’ label  - ‘Detail’ button  4. System redirects to UserDetail page with the following control :  - ‘UserName’ label  - ‘FullName’ label  - ‘Birthday’ label  - ‘Gender’ label  - ‘Email’ label  - ‘Role’ label  - ‘Department’ label  - ‘status’ label  - ‘Edit’ Button |

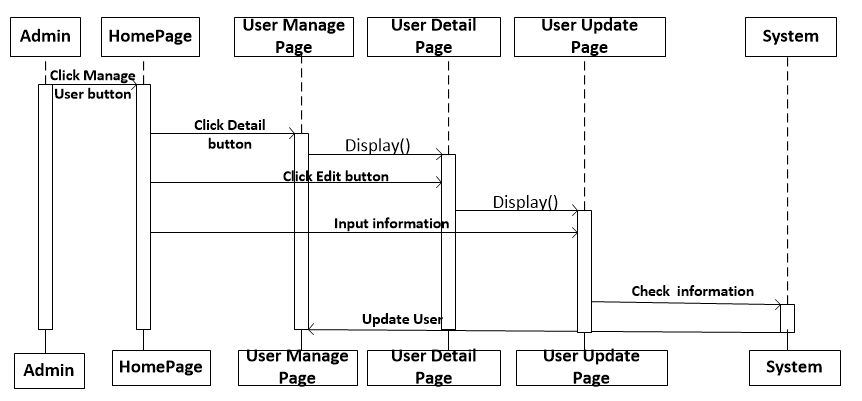
* 1. **Create user**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Create user | |
| **Actors** | Admin | |
| **Description** | Actor can Create user account | |
| **Requirements** | Actor provides profile information | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Return UserManage page  Fail: Output fail | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Create user’ button in Actor’s UserManage page  3. Actor inputs user’s information, and click ‘Create’ button. [Alternative 1] | System Responses:  2. System redirects to UserCreate page and displays Create user form with the following control :  - ‘UserName’ text field  - ‘Password’ text field  - ‘Password Confirm’ text field  - ‘FullName’ text field  - ‘Birthday’ datebox  - ‘Gender’ checkbox group  - ‘Email’ text field  - ‘Role’ listbox  - ‘Department’ listbox  - ‘Save’ button  4. System checks the information  [Exception 1]  5. System inserts the account into database with default status is true, show message: ‘Create Account Success’ and then redirects UserManage page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back’ button. | System Responses:  2. System returns to UserManage page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘UserName’ text field bank.  2. UserName has already exists.  3. Actor lefts input an invalid UserName.  4. Actor lefts ‘New Password’ text field blank.  5. Actor lefts ‘New Confirm Password’ text field blank.  6. New Password and New Password Confirm don’t match.  7. Actor lefts ‘Full Name’ text field blank.  8. Actor lefts ‘Birthday’ datebox blank.  9.Input age to ‘Birthday’ datebox less than 18.  10. Actor lefts ‘Email’ text field blank or input an invalid email address. | System Responses:  System displays message to Actor :  - ‘UserName cannot be blank’.  - ‘Username is exists’.  - ‘UserName is invalid’.  - ‘Password cannot be blank’’.  - ‘Confirm Password can’t be blank’.  - ‘Confirm password not match Password.’  - ‘FullName cannot be blank’.  - ‘Birthday cannot be blank’.  - ‘Age must over 18’.  - ‘Email is invalid or is exists’. |

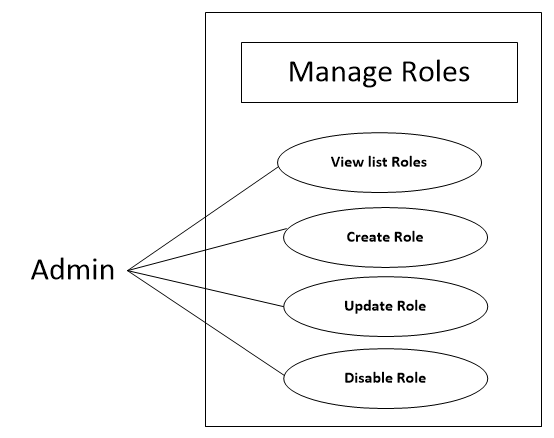
****

* 1. **Update user**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Update user’s information | |
| **Actors** | Admin | |
| **Description** | Admin can update user’s information | |
| **Requirements** | Actor provides account’s information | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Update Success.  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Edit’ button in UserDetail page.  3. Actor input information and click ‘Save’ button. [Alternative 1] | System Responses:  2. System redirects to UserUpdate page and displays Update form with the following control :  - ‘UserName’ text field (read only)  - ‘FullName’ text field  - ‘Birthday’ datebox  - ‘Gender’ checkbox group  - ‘Email’ text field  - ‘Role’ label  - ‘Department’ label  - ‘status’(Enable) checkbox  - ‘Reset Password’ checkbox  - ‘Save’ button  4. System checks the information  [Exception 1]  5. System update the account into database and then redirects to UserManage page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Home page. |
| **Exception** | Actor Actions:  [Exception 1]  7. Actor lefts ‘Full Name’ text field blank.  9. Actor lefts input an invalid email address.  9. Actor lefts ‘Email’ text field blank or input an invalid email address. | System Responses:  System displays message to Actor :  - ‘FullName cannot be blank’.  - ‘Email is invalid or is exists’. |

****

**2.Manage Role**

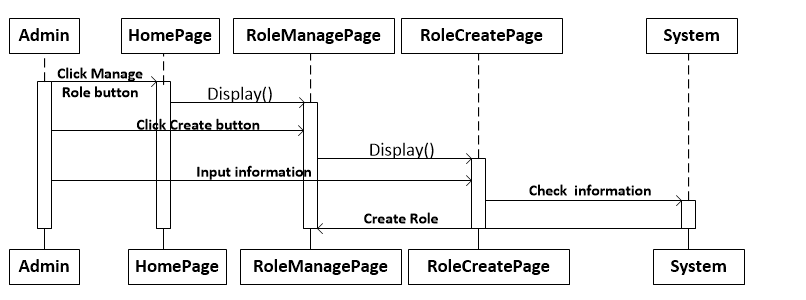
****

* 1. **View list Role**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | View list user | |
| **Actors** | Admin | |
| **Description** | Actor can view list user account | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Return RoleManage page | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Manage Role’ button in Actor’s Home page. | System Responses:  2. System redirects to RoleManage page and displays List Role with the following control :  - ‘Role’ label  - ‘Description’ label  - ‘Status’(Enable) label  - ‘Edit’ button |

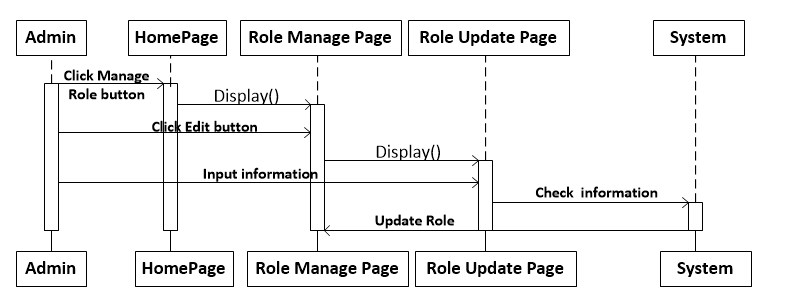
* 1. **Create Role**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Create Role | |
| **Actors** | Admin | |
| **Description** | Actor can Create Role on department | |
| **Requirements** | Actor provides Role’s information | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Return manage list user page  Fail: Output fail | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Create Role’ button in Actor’s CreateRole page  3. Actor inputs Role’s information, and click ‘Create’ button. [Alternative 1] | System Responses:  2. System redirects to RoleCreate page and displays Create user form with the following control :  - ‘RoleName’ text field  - ‘Description’ textarea  - ‘Create’ Button  4. System checks the information  [Exception 1]  5. System inserts the Role into database, show message: ‘Create Role Success’ and then redirects RoleManage page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back’ button. | System Responses:  2. System returns to RoleManage page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘RoleName’ text field bank.  2. RoleName has already exists. | System Responses:  System displays message to Actor :  - ‘RoleName cannot be blank’.  - ‘RoleName is exists’ |

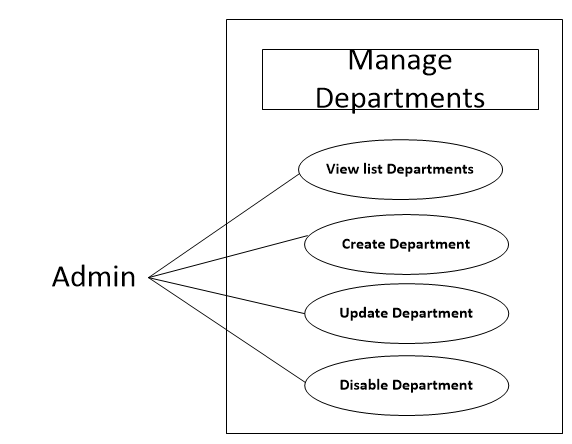
****

* 1. **Update Role**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Update Role’s information | |
| **Actors** | Admin | |
| **Description** | Admin can update Role’s information | |
| **Requirements** | Actor provides Role’s information | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Update Success then redirects to RoleManage page.  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Edit’ button in RoleManage page.  3. Actor input information and click ‘Save’ button. [Alternative 1] | System Responses:  2. System redirects to RoleUpdate page and displays Update form with the following control :  - ‘RoleName’ text field  - ‘Department’ listbox  - ‘Description’ textarea  - ‘Status’ checkbox  - ‘Save’ Button  4. System checks the information  [Exception 1]  5. System update the account into database and then redirects to RoleManage page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back’ button. | System Responses:  2. System returns to RoleManage page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘RoleName’ text field bank.  2. RoleName has already exists.  3. | System Responses:  System displays message to Actor :  - ‘RoleName cannot be blank’.  - ‘RoleName is exists’ |

****

**3.Manage Department**

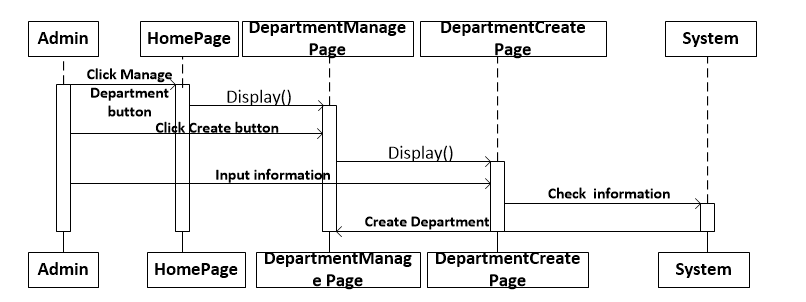
****

**3.1 View list Department**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | View list Department | |
| **Actors** | Admin | |
| **Description** | Actor can view list Department | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Return Department Manage page | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Manage Department’ button in Actor’s Home page. | System Responses:  2. System redirects to Department Manage page and displays Deparment List with the following control :  - ‘Department’ label  - ‘status’(Enable) label  - ‘Edit’ button |

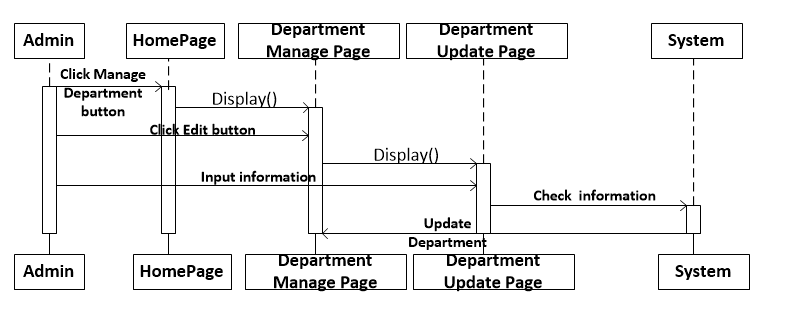
**3.2 Create Department**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Create Department | |
| **Actors** | Admin | |
| **Description** | Actor can Create Department | |
| **Requirements** | Actor provides Department’s information | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Return manage list Department page  Fail: Output fail | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Create Department’s’ button in Actor’s Create Department page  3. Actor inputs Department’s information, and click ‘Create’ button. [Alternative 1] | System Responses:  2. System redirects to CreateDepartment page and displays Create Department form with the following control :  - “Department’s Name” text field  - ‘Create’ Button  4. System checks the information  [Exception 1]  5. System inserts the Department into database, show message: ‘Create Department Success’ and then redirects Department Manage page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back’ button. | System Responses:  2. System returns to Department Manage page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Department Name’ text field bank.  2. Department Name has already exists. | System Responses:  System displays message to Actor :  - ‘Department Name cannot be blank’.  - ‘Department Name is exists’ |

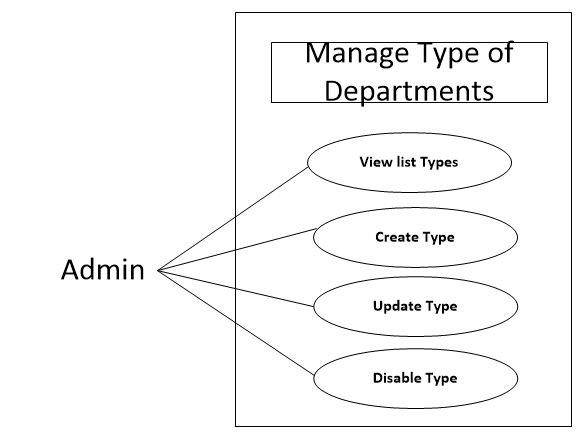
****

* 1. **Update Department**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Update Department’s information | |
| **Actors** | Admin | |
| **Description** | Admin can update Department’s information | |
| **Requirements** | Actor provides Department’s information | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Update Success then redirects to Department Manage page.  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Edit’ button in Department Manage page.  3. Actor input information and click ‘Save’ button. [Alternative 1] | System Responses:  2. System redirects to Department Update page and displays Update form with the following control :  - ‘Department Name’ text field  - ‘Status’(Enable) checkbox  - ‘Save’ Button  4. System checks the information  [Exception 1]  5. System update the Department into database and then redirects to DepartmentManage page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back’ button. | System Responses:  2. System returns to DepartmentManage page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘DepartmentName’ text field bank.  2. DepartmentName has already exists. | System Responses:  System displays message to Actor :  - ‘DepartmentName cannot be blank’.  - ‘DepartmentName is exists’ |

****

**5. Manage Type of document**

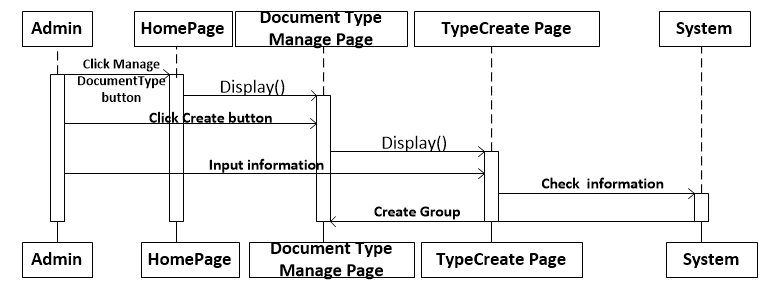
****

**5.1 View list Type of document**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | View list Type | |
| **Actors** | Admin | |
| **Description** | Actor can view list Type of document | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Return Type Manage page | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Manage Department’ button in Actor’s Home page. | System Responses:  2. System redirects to Type Manage page and displays Deparment List with the following control :  - ‘Type’ label  - ‘status’(Enable) label  - ‘Edit’ button |

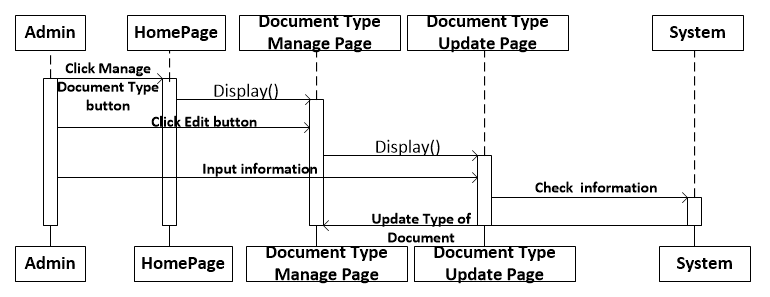
**5.2 Create Type of document**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Create Type | |
| **Actors** | Admin | |
| **Description** | Actor can Create type of document | |
| **Requirements** | Actor provides type’s information | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Return manage list type’s document page  Fail: Output fail | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Create Type’ button in Actor’s Manage Type of document page  3. Actor inputs Type’s information, and click ‘Create’ button. [Alternative 1] | System Responses:  2. System redirects to TypeCreate page and displays Create Type form with the following control :  - “Type Name” text field  - ‘Create’ Button  4. System checks the information  [Exception 1]  5. System inserts the type into database, show message: ‘Create Type of document Success’ and then redirects Type Manage page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back’ button. | System Responses:  2. System returns to Type Manage page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Type Name’ text field bank.  2. ‘Type Name’ has already exists. | System Responses:  System displays message to Actor :  - ‘Type Name cannot be blank’.  - ‘Type Name is exists’ |

****

* 1. **Update Type of document**

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Update Type’s information | |
| **Actors** | Admin | |
| **Description** | Admin can update Type ‘s information | |
| **Requirements** | Actor provides Type ‘s information | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Update Success then redirects to Type Manage page.  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Edit’ button in Type Manage page.  3. Actor input information and click ‘Save’ button. [Alternative 1] | System Responses:  2. System redirects to Type Update page and displays Update form with the following control :  - ‘Type Name’ text field  - ‘Status’(Enable) checkbox  - ‘Save’ Button  4. System checks the information  [Exception 1]  5. System update the Type into database and then redirects to Type Manage page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back’ button. | System Responses:  2. System returns to Type Manage page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Type Name’ text field bank.  2. Type Name has already exists. | System Responses:  System displays message to Actor :  - ‘Type Name cannot be blank’.  - ‘Type Name is exists’ |

****