



*Operating Lease Support System Development*

**For**

******

***Functional Specification***

***USER AND ROLE MANAGEMENT***

Revision 2.0

**Prepared by:**

**PT. Berlian Sistem Informasi**

KTB Annex Bld. 4th Floor, Jl. Jend. A. Yani. Pulomas,

Tel. +62 21 4786 7575 (Hunting) Fax. +62 21 4714964 [www.bsi.co.id](http://www.bsi.co.id)

Document Control

|  |  |
| --- | --- |
| **Author** | Achmad Fardani Rizki |
| **File Name** |  |
| **Path** | N/A |
| **Create Date** |  |
| **Last Edited** |  |
| **Number of Pages** |  |

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Revision** | **Description** | **Author** |
| 15-May-2015 | 0.0 | Initial creation | Achmad Fardani |
| 10-June-2015 | 1.0 | Update distribution list and document approval | Setiyono |
| 25-June-2015 | 2.0 | Update data structure and screen design | Setiyono |

Distribution List

|  |  |  |
| --- | --- | --- |
| **Name** | **Organization** | **Title** |
| Mr. Ahmad Fikri | Dipo Star Finance, PT | DSF ITD Head |
| Mr. Deni Kusumah | Dipo Star Finance, PT | DSF ITD |
| Mr. Mukhlis Ibrahim | Dipo Star Finance, PT | DSF ITD |
| Mr. Delta Riangga | Dipo Star Finance, PT | DSF Project Manager |
| Mr. Feri Irawan | Berlian Sistem Informasi, PT | BSI Multifinance Division Head |
| Mr. Asra Haryadi | Berlian Sistem Informasi, PT | BSI Operation Unit Manager |
| Mr. Devilosa Indra Kamal | Berlian Sistem Informasi, PT | BSI Service Manager |

Document Approval

By signing this document I acknowledge I have read the document and give the Project Management Team approval to proceed.

|  |  |  |  |
| --- | --- | --- | --- |
| **Role** | **Name** | **Signature** | **Date** |
| DSF ITD Head | Mr. Ahmad Fikri |  |  |
| DSF ITD | Mr. Deni Kusumah |  |  |
| DSF ITD | Mr. Mukhlis Ibrahim |  |  |
| BSI Operation Unit Manager | Mr. Asra Haryadi |  |  |
| BSI Service Manager | Mr. Devilosa Indra Kamal |  |  |

Table of Contents

Document Control 2

Revision History 2

Distribution List 2

Document Approval 2

1 Introduction 4

1.1 Purpose 4

1.2 Scope 4

1.3 Acronyms and abbreviations 5

1.4 References 5

2 Detail Specification 6

2.1 User 6

**2.1.1** **Use case** 6

**2.1.2** **Operations and scenarios** 6

**2.1.3** **Sitemap design** 8

**2.1.4** **Screen design** 9

**2.1.4.1** **User** 9

**2.1.4.2** **Add new user record** 9

**2.1.4.3** **Edit User record** 10

**2.1.4.4** **View detail user record** 10

**2.1.5** **Screen features** 11

**2.1.6** **Data structure** 11

2.2 Role 12

**2.2.1** **Use case** 12

**2.2.2** **Operations and scenarios** 12

**2.2.3** **Sitemap design** 14

**2.2.4** **Screen design** 15

**2.2.4.1** **Role information** 15

**2.2.4.2** **Add new Role record** 15

**2.2.4.3** **Edit Role record** 16

**2.2.4.4** **View detail role record** 17

**2.2.5** **Screen features** 18

**2.2.6** **Data structure** 19

2.3 Manage Permission 19

**2.3.1** **Use case** 19

**2.3.2** **Screen design** 19

**2.3.2.1** **Manage Permission** 19

**2.3.3** **Screen features** 20

**2.3.4** **Data structure** 20

3 Appendix 21

1. Introduction
   1. Purpose

Functional specification is to explain some of the following:

1. Describe who uses the application or system,
2. Describe the order in which steps or events or on the scenario are performed,
3. Show user interface design, and
4. What kind of data that will be maintained by functions.
   1. Scope

User and Role Management module explained about how to maintain user data and role data which is utilized in the system. On this module there’s no integration with another system (such like ACA and MFAPPL). This modules deal in the “to-be design” for developing the new system.

To keep the accuracy of entire design of the system we are using modeling approach for design and analysis. ***To define a model of functions we will determine use cases. In the use case, it can summarize who uses the application or system, and what they can do with it.***

Within a use case we can determine which use cases are supported by application. We can create use case diagram, activity diagram, use case scenario, data structure, and interface design.

A data structure can specified what kind information that will be required for the system. Data structure is represented database design of the system. A user interface design can imagine the user about the system as visual. User interface is designed based on prototype which already created. And, based on these things, we will develop the system.

The following is functionality of user management module:

* User information

1. Add new User record.

To add new user record by fill out add new form.

1. Edit user record.

To edit user record by fill out edit form. For submit a user record.

1. Submit user record.

To submit a user records by click event.

1. Filter and sort user record

To perform data filter and data sort of user records.

1. View detail user record

To inspect a detail user records by opens a detail form of user.

1. Changes status of user record

To update status of user record by click events.

* Role information

1. Add new role record.

To add new role record by fill out add new form.

1. Edit role record.

To edit role record by fill out edit form. For submit a role record.

1. Submit role record.

To submit a role records by click event.

1. Filter and sort role record

To perform data filter and data sort of role records.

1. View detail role record

To inspect a detail role records by opens a detail form of role.

1. Changes status of role record

To update status of role record by click events.

* Permission

1. Reload function.

To update the list of function from each module by click events.

1. Edit alias name.

To set alias name of function from each module, the alias name will be shown as assign permission a column name on role module.

1. Activate permission.

To update status of permission record by click events.

* 1. Acronyms and abbreviations
* ACA : Automatic Credit Approval System
* MFAPPL : Multi Finance Application
  1. References

[N/A]

1. Detail Specification
   1. User
      1. **Use case**

The figure below is summarizing who uses features of function, and what they can do with it.

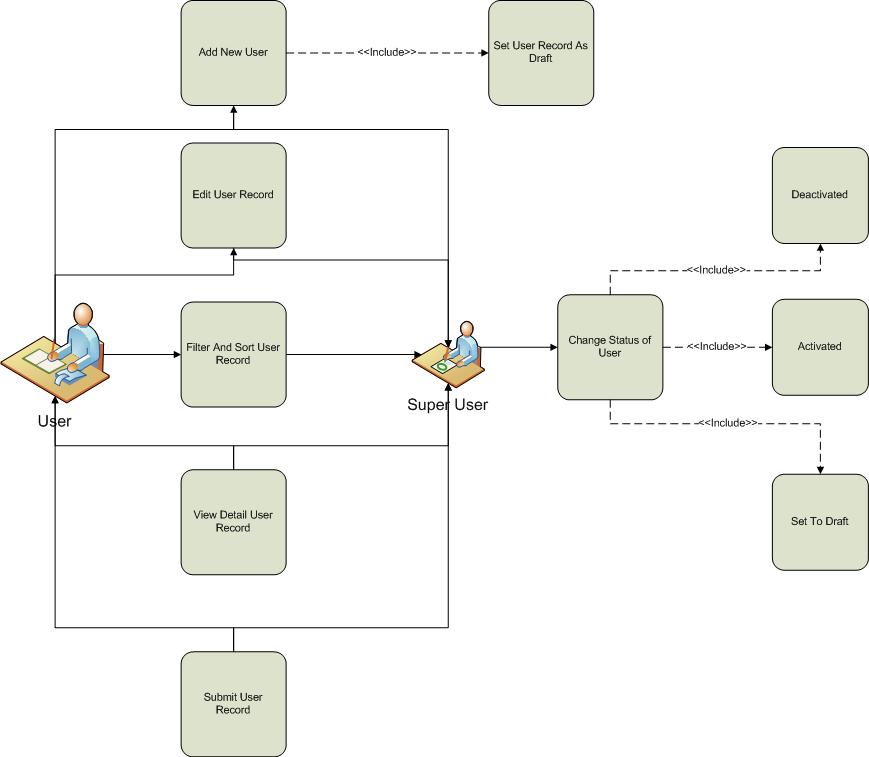


Figure 1 – Use case diagram of User Management

* + 1. **Operations and scenarios**

The figure below is describing operations or steps performed in a function interact with people (flow of work between users and the system).

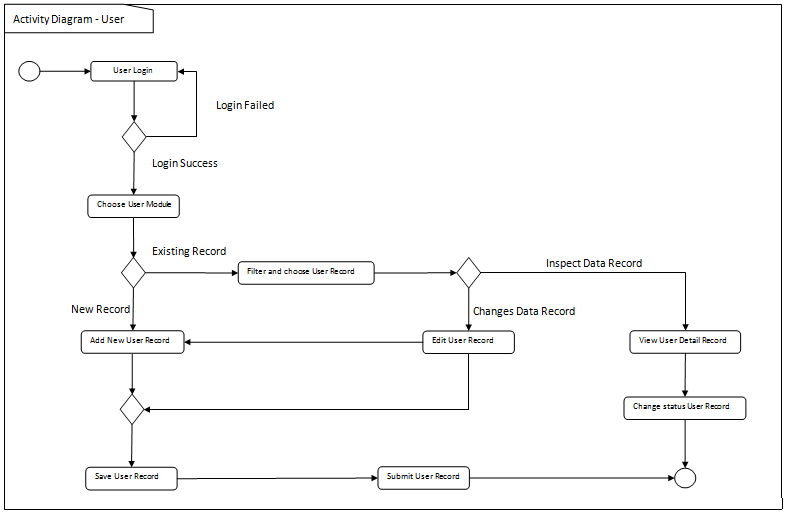


Figure 2 – Activity diagram of User Management

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Steps/ Operations** | **Actor/ People** | **Scenario** |
| 1 | Login | System Admin, ITD User | * Actor fills out login form by inputting user name and password. * Actor clicks [sign in] button. System will check the account of user; if actor account is already registered user will enter into the system. If doesn’t user can return the step. |
| 2. | Choose User module | System Admin, ITD User | * After login, actor can choose user module on the menu of system. After that, screen of user information will be opens. * After user screen opens, user list will be shown and existing user data will be displayed. |
| 3. | Add new user | System Admin, ITD User | * If user information screen opens, actor can click [add] button to add new records of user, and after that add new user screen will be opens. * Upon add new user screen, actor fill out the fields. * The mandatory fields cannot be blanks. |
| 4. | Save user record | System Admin, ITD User | * After fill out the fields, actor can click [save as draft] button, if want to make a records as **draft**. * After that, system will proceed to store a data into database and will be automatically check a validation of data (such like; data type, mandatory data, and data length) * If done, actor can click back button to return to the user list, then user data will be displayed. |
| 5 | Submit user record | System Admin, ITD User | * Once data is saved and data has been confirmed, actor can submit the record by click [submit] button. * After that system will set a user record as **submitted (or not draft)**. * If done, actor can click back button to return to the customer list. |
| 6. | Filter and sort user records | System Admin, ITD User | * Actor can then filter and sort the user record based on columns where displayed on the user list. * To filter user records, actor can enter keywords into the filter fields. After that press enter on the keyboards. If data founds, the result will be displayed on the customer list. If doesn’t customer list will be blanks. * To sort user records, actor can click header of column on the list table. After that the records will be sorted by ascending or descending, alternately. |
| 7. | Edit user records | System Admin, ITD User | * On the user information screen actor can open edit user screen to edit some kinds of data. * To edit data of user and if edit user screen opens, actor can edit a record by inputting a data on each of the fields. * The mandatory fields cannot be blanks. |
| 8. | View detail user record | System Admin, ITD User | * On the user information screen actor can open view detail user screen to inspect a data. * To view detail user record, click [view detail] button, after that system will displayed a screen of view detail user. * On this screen, actor cannot do something to edit of data. Only view detail of user record. |
| 9 | Change status of user record | System Admin, ITD User | * If view detail user screen opens, actor can change the status of user by click of [activated] button or [deactivated] button (depending by status that needs to change). * If data should be edited by actor, then records can be [set to draft] * After that, system will proceed automatically to change the status. * If status of user record is **active**, then [deactivated] will be visible. And opposite that, if status of customer record is **inactive**, then [activated] will be visible. In short, one of them will be visible or invisible. * If done, actor can click back button to return to the deactivated list. |

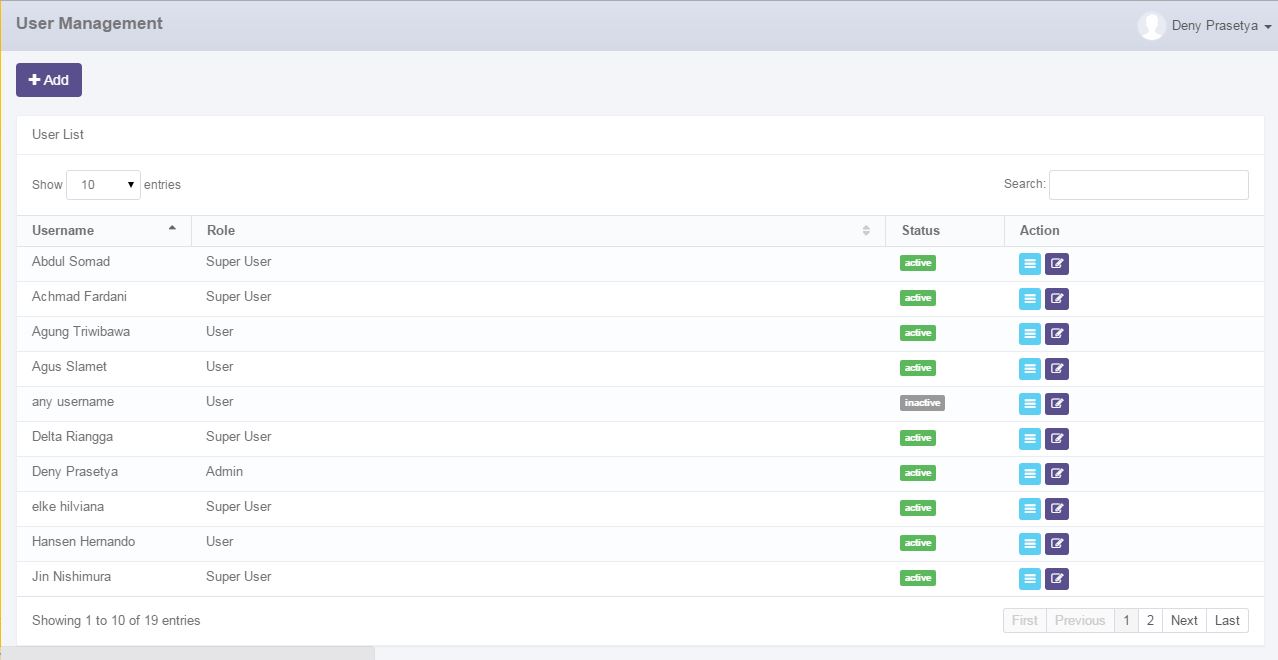
* + 1. **Sitemap design**

Describe a web page that lists the pages on a web site to users. Usually sitemap organized in hierarchical style.

Figure 3 – Sitemap design of customer information page

* + 1. **Screen design**
       1. **User**

This screen is designed to display a User list. On the User list actor can filter and sort a User records. And also, actor can open another screen to add new User record, edit User record, and view detail of User record.



Data paging information

To open paging of user records

To filter user record

To open edit or view detail user

To open add new user pages

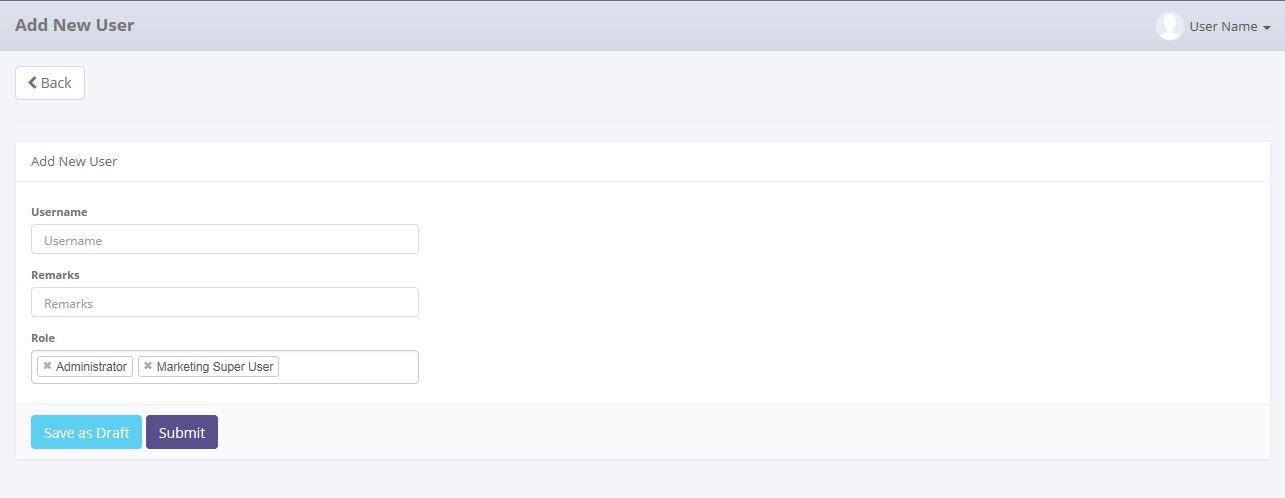
To filter a number of record filter button

Displaying user list

Image 1 – Screen of User Information

* + - 1. **Add new user record**

This screen is designed to entry a new User record. After fill out some data at the screen, actor can stored a data into the system by click the action button.



To save or submit of data

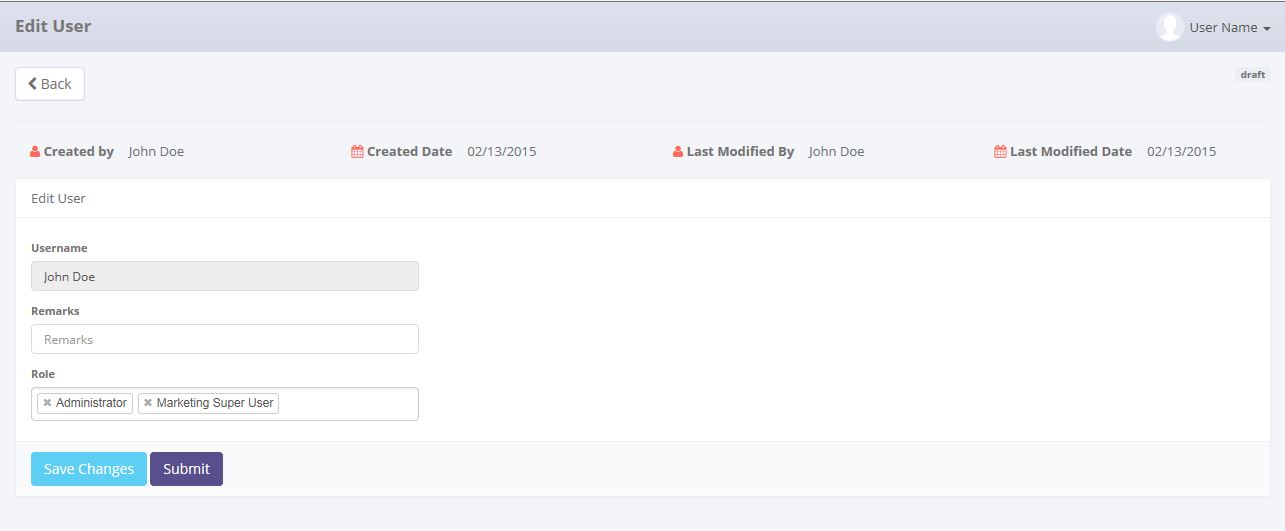
Return back to user list

To entry a data of user

Image 2 – Screen of add new User record

* + - 1. **Edit User record**

This screen is designed to edit User record. After changes some data at the screen, actor can stored a data into the system by click the action button.



Status of user record

History of data

Return back to user list

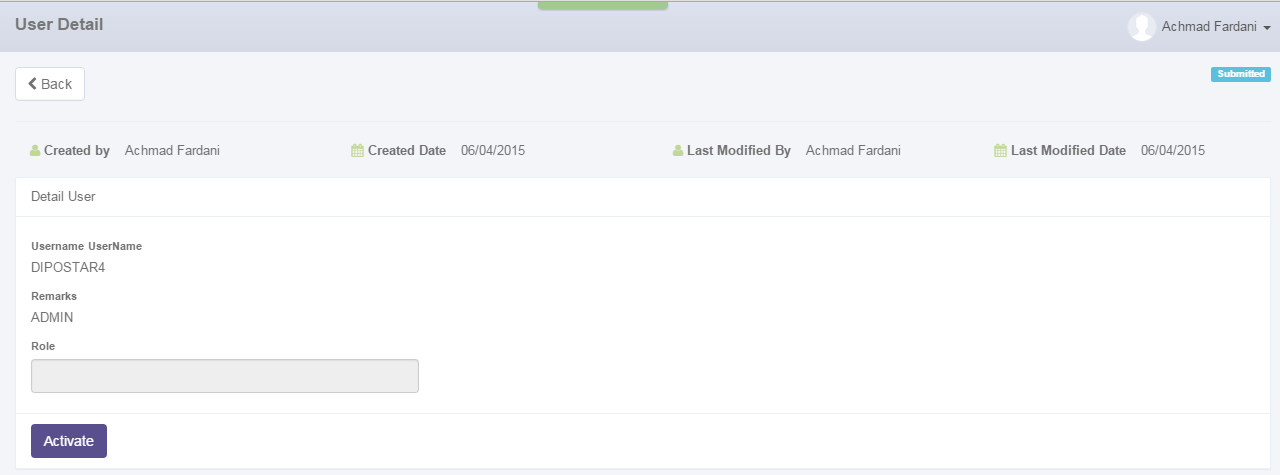
To save or submit of data

Entry data form

Image 3 – Screen of edit customer record

* + - 1. **View detail user record**

This screen is designed to view detail a User record. On this screen actor can changes status of a User record by click action button.



To activate data

History of data

Return back to user list

Detail information of user

Status of user record

Image 4 – Screen of view detail user record

* + 1. **Screen features**

|  |  |
| --- | --- |
| **Features** | **Description** |
| [Filter and sort] | To perform data filter and data sort of User records. |
| [Add new] | To add new User record by fill add new form. |
| [Save as draft] or [Set to draft] | To save a new User record as a draft. |
| [Submit] | To submit a User record as a final. |
| [Save changes] | To update of changes when editing User record. |
| [View detail] | To inspect a detail User records by opens a detail form of User. |
| [Edit] | To edit of User record. |
| [Activated] | To return back of status of User record as active. |
| [Deactivated] | To set a status of User record as inactive |

* + 1. **Data structure**

|  |  |  |
| --- | --- | --- |
| **Field** | **Data Type** | **Mandatory** |
| IdUser | int | Yes |
| UserName | varchar | Yes |
| Password | varchar | No |
| Remarks | varchar | No |
| CreateDate | datetime | Yes |
| CreateBy | varchar | Yes |
| LastModified | datetime | Yes |
| LastModifiedBy | varchar | Yes |
| Status | varchar | Yes |
| IsDraft | bit | Yes |
| IsSubmit | bit | Yes |
| IsActive | bit | Yes |
| IsDeleted | bit | Yes |

* 1. Role
     1. **Use case**

The figure below is summarizing who uses features of function, and what they can do with it.

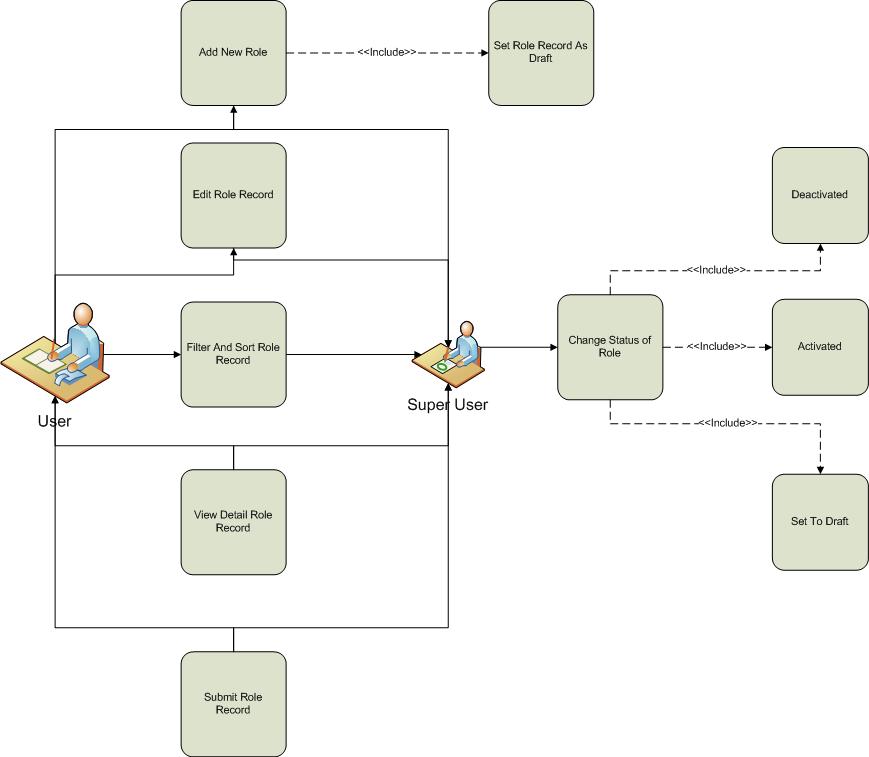


Figure 4 – Use case diagram of Role Management

* + 1. **Operations and scenarios**

The figure below is describing operations or steps performed in a function interact with people (flow of work between users and the system).

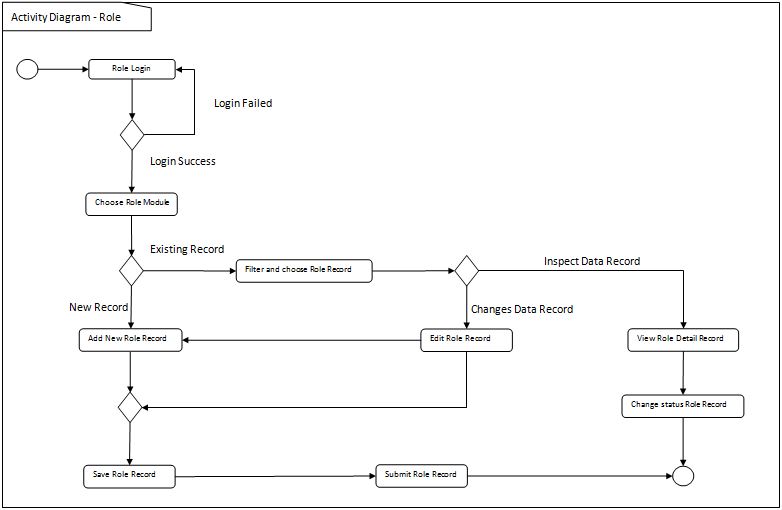


Figure 5 – Activity diagram of Role Information

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Steps/ Operations** | **Actor/ People** | **Scenario** |
| 1 | Login | System Admin, ITD User | * Actor fills out login form by inputting user name and password. * Actor clicks [sign in] button. System will check the account of user; if actor account is already registered user will enter into the system. If doesn’t user can return the step. |
| 2. | Choose Role module | System Admin, ITD User | * After login, actor can choose Role module on the menu of system. After that, screen of Role information will be opens. * After Role screen opens, Role list will be shown and existing Role data will be displayed. |
| 3. | Add new Role | System Admin, ITD User | * If Role information screen opens, actor can click [add] button to add new records of Role, and after that add new Role screen will be opens. * Upon add new Role screen, actor fill out the fields. * The mandatory fields cannot be blanks. |
| 4. | Save Role record | System Admin, ITD User | * After fill out the fields, actor can click [save as draft] button, if want to make a records as **draft**. * After that, system will proceed to store a data into database and will be automatically check a validation of data (such like; data type, mandatory data, and data length) * If done, actor can click back button to return to the Role list, then Role data will be displayed. |
| 5 | Submit Role record | System Admin, ITD User | * Once data is saved and data has been confirmed, actor can submit the record by click [submit] button. * After that system will set a Role record as **submitted (or not draft)**. * If done, actor can click back button to return to the customer list. |
| 6. | Filter and sort Role records | System Admin, ITD User | * Actor can then filter and sort the Role record based on columns where displayed on the Role list. * To filter Role records, actor can enter keywords into the filter fields. After that press enter on the keyboards. If data founds, the result will be displayed on the customer list. If doesn’t customer list will be blanks. * To sort Role records, actor can click header of column on the list table. After that the records will be sorted by ascending or descending, alternately. |
| 7. | Edit Role records | System Admin, ITD User | * On the Role information screen actor can open edit Role screen to edit some kinds of data. * To edit data of Role and if edit Role screen opens, actor can edit a record by inputting a data on each of the fields. * The mandatory fields cannot be blanks. |
| 8. | View detail Role record | System Admin, ITD User | * On the Role information screen actor can open view detail Role screen to inspect a data. * To view detail Role record, click [view detail] button, after that system will displayed a screen of view detail Role. * On this screen, actor cannot do something to edit of data. Only view detail of Role record. |
| 9 | Change status of Role record | System Admin, ITD User | * If view detail Role screen opens, actor can change the status of Role by click of [activated] button or [deactivated] button (depending by status that needs to change). * If data should be edited by actor, then records can be [set to draft] * After that, system will proceed automatically to change the status. * If status of role record is **active**, then [deactivated] will be visible. And opposite that, if status of role record is **inactive**, then [activated] will be visible. In short, one of them will be visible or invisible. * If done, actor can click back button to return to the deactivated list. |

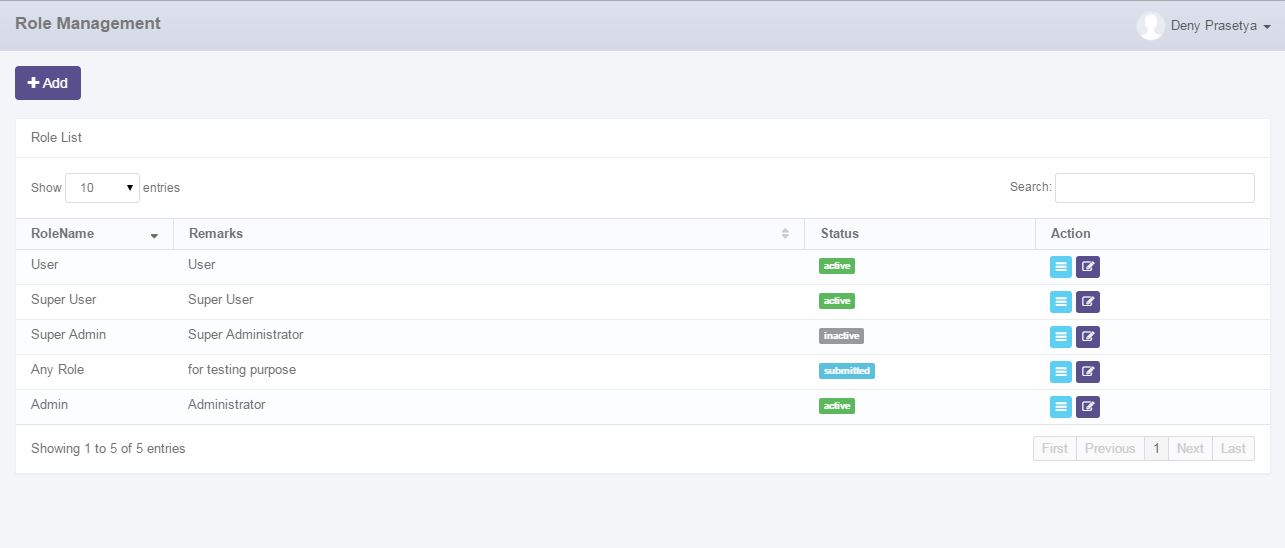
* + 1. **Sitemap design**

Describe a web page that lists the pages on a web site to users. Usually sitemap organized in hierarchical style.

Figure 6 – Sitemap design of role information page

* + 1. **Screen design**
       1. **Role information**

This screen is designed to display a Role list. On the Role list user can filter and sort a Role records. And also, user can open another screen to add new Role record, edit Role record, and view detail of Role record.



To open paging of role records

Data paging information

To open add new role pages

To open edit or view detail role

Displaying role list

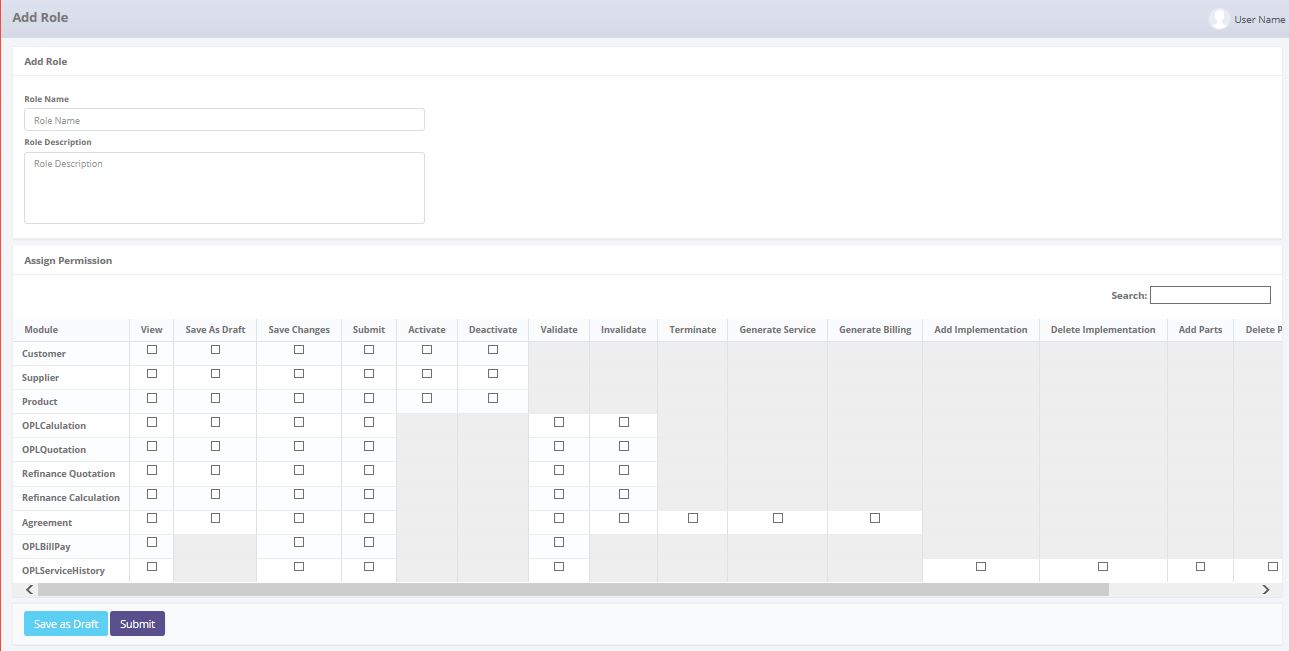
To filter role record

To filter a number of record filter button

Image 5 – Screen of Role information

* + - 1. **Add new Role record**

This screen is designed to entry a new Role record. After fill out some data at the screen, user can stored a data into the system by click the action button.



To search module name of permission

To assign permission to role

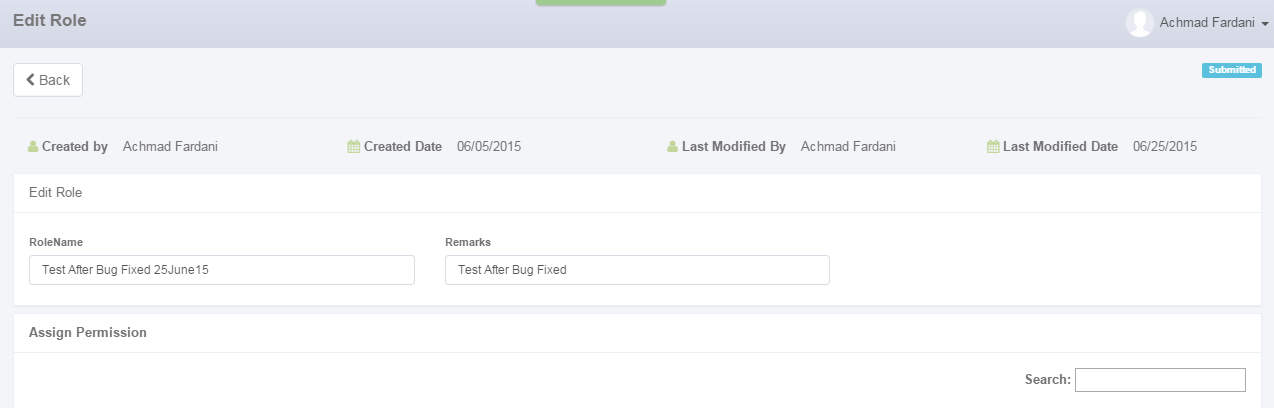
To save or submit of data

To entry a data of role

Image 6 – Screen of add new role record

* + - 1. **Edit Role record**

This screen is designed to edit role record. After changes some data at the screen, user can stored a data into the system by click the action button.



To search module name of permission

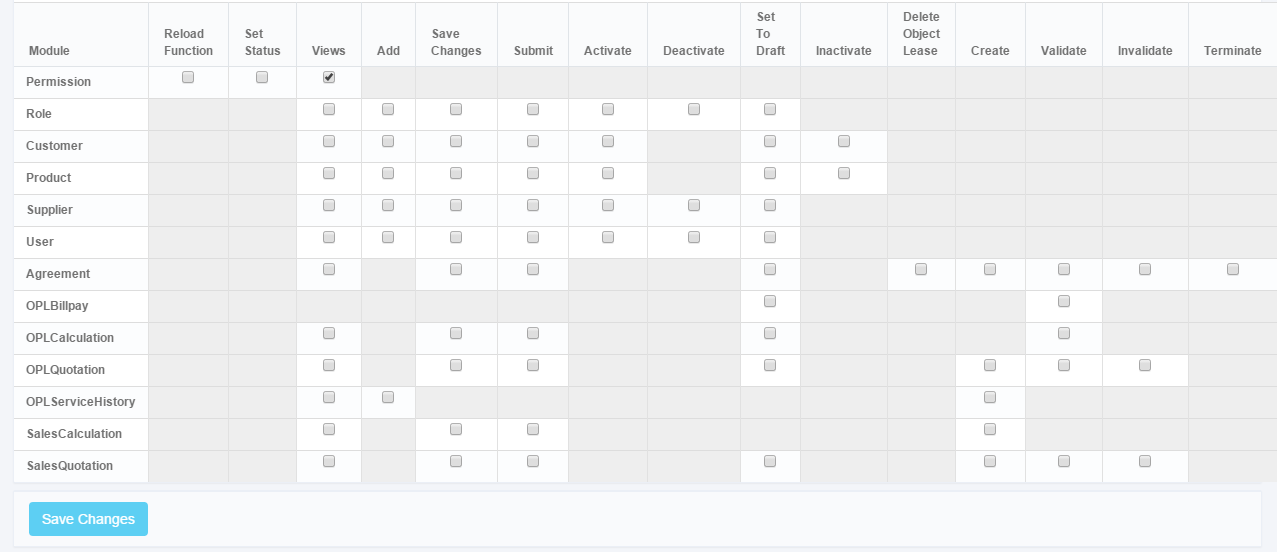
Entry data form

Status of role record

History of data

To assign permission to role

Return back to role list

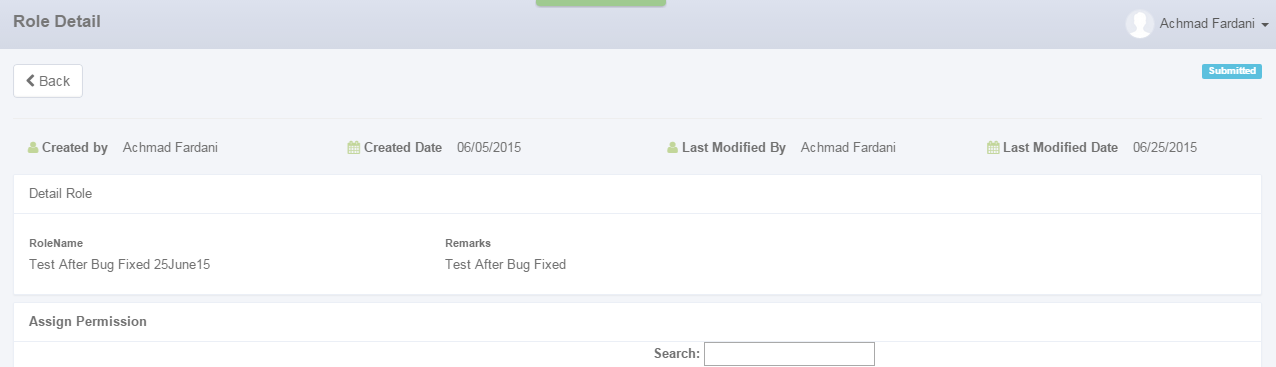


To save data

Image 7 – Screen of edit role record

* + - 1. **View detail role record**

This screen is designed to view detail a role record. On this screen user can changes status of a supplier record by click action button.



To search module name of permission

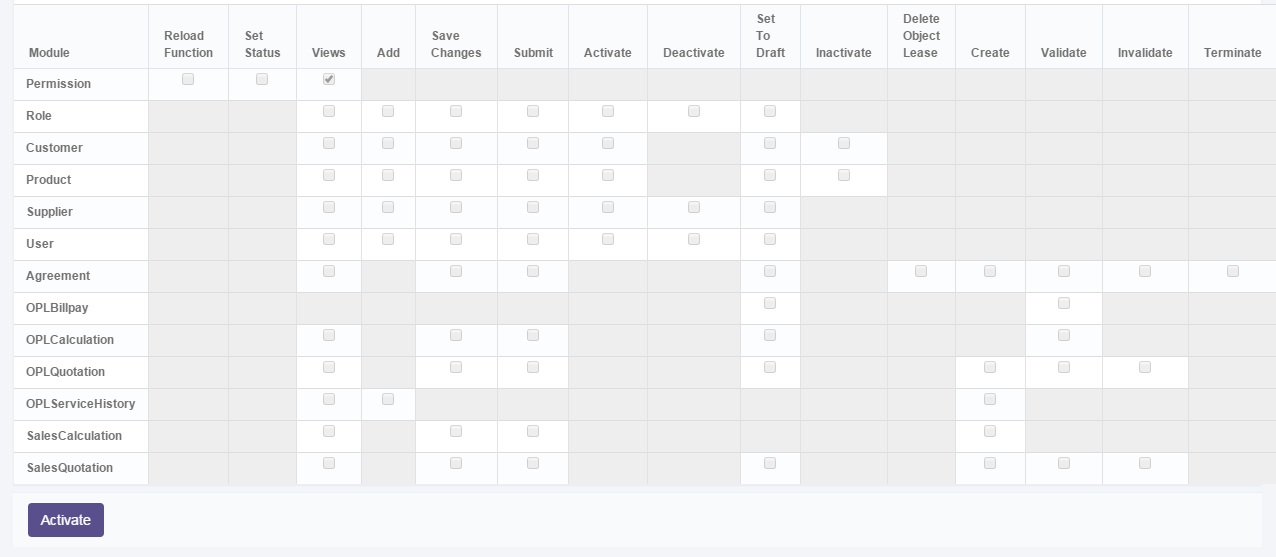
Role data

Status of role record

History of data

permission of role

Return back to role list



To activate data

Image 8 – Screen of view detail role record

* + 1. **Screen features**

|  |  |
| --- | --- |
| **Features** | **Description** |
| [Filter and sort] | To perform data filter and data sort of product records. |
| [Add new] | To add new supplier record by fill out add new form. |
| [Save as draft] or  [Set to draft] | To save a new supplier record as a draft. |
| [Submit] | To submit a supplier record as a final. |
| [Save changes] | To update of changes when editing supplier record. |
| [View detail] | To inspect a detail supplier records by opens a detail form of supplier. |
| [Edit] | To edit of supplier record. |
| [Activated] | To return back of status of supplier record as active. |
| [Deactivated] | To set a status of supplier record as inactive |

* + 1. **Data structure**

|  |  |  |
| --- | --- | --- |
| **Field** | **Data Type** | **Mandatory** |
| IdRole | int | Yes |
| RoleName | varchar | Yes |
| Remarks | varchar | No |
| IsDeleted | bit | Yes |
| Status | varchar | Yes |
| IsDraft | bit | Yes |
| IsSubmit | bit | Yes |
| IsActive | bit | Yes |
| CreatedDate | datetime | Yes |
| CreatedBy | nvarchar | Yes |
| LastModified | datetime | Yes |
| LastModifiedBy | varchar | Yes |

* 1. Manage Permission
     1. **Use case**

The figure below is summarizing who uses features of function, and what they can do with it.

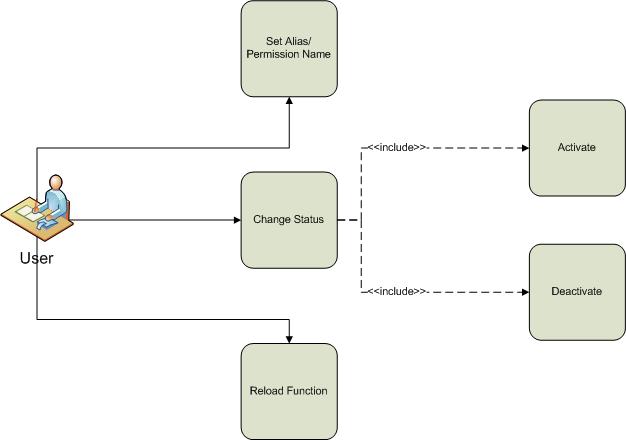
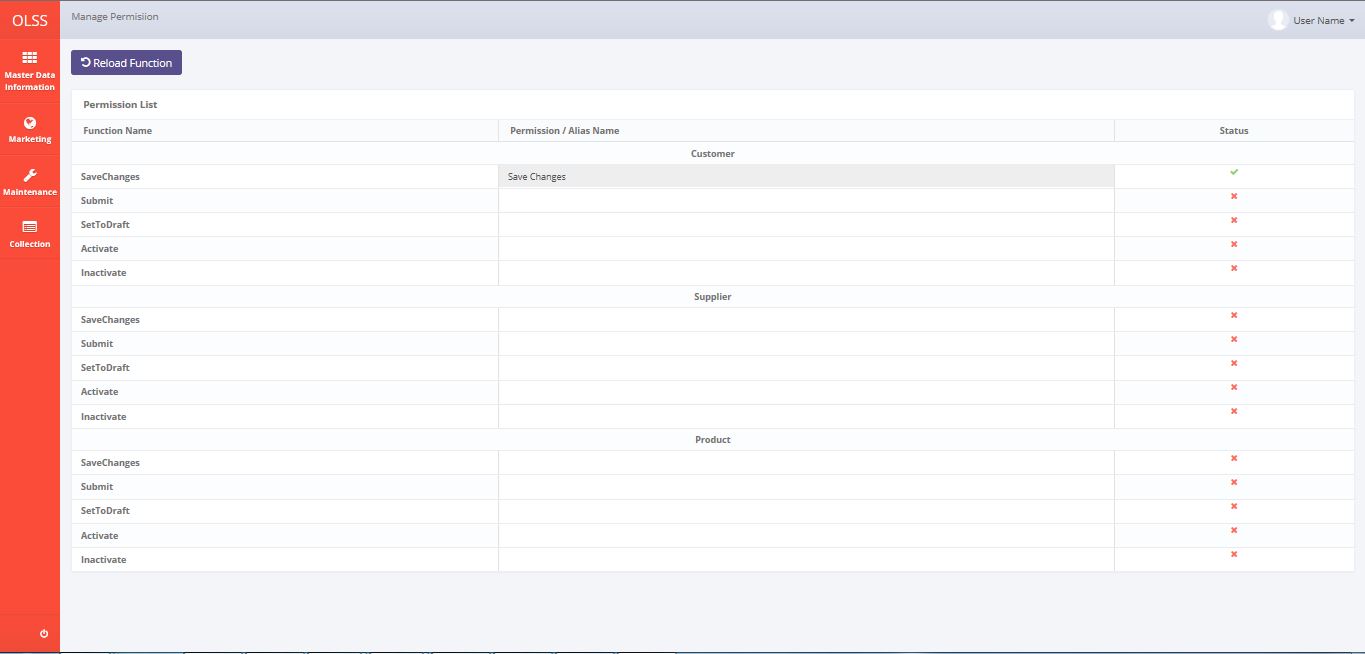


Figure 7 – Use case diagram of Manage Permission

* + 1. **Screen design**
       1. **Manage Permission**

This screen is designed to display, and manage the permission. On the permission list user can reload the list of olss function, set the alias, and activate or deactivate permission.



Activate/Deactivate Permission

Input for function aliases

List of function name

To reload the function list on olss application

Image 9 – Screen of Manage Permission

* + 1. **Screen features**

|  |  |
| --- | --- |
| **Features** | **Description** |
| [Reload Function] | To reload all function on olss assembly that has [OlssAtuthorization] Attributes. |
| [Activated / Deactivated] | To Set Permission as Active or Inactive Permission |
| [Set Alias] | To set alias of function name, the alias will shown as column name of assign permission on role module |

* + 1. **Data structure**

|  |  |  |
| --- | --- | --- |
| **Field** | **Data Type** | **Mandatory** |
| IdPermission | int | Yes |
| ModuleName | nvarchar | Yes |
| FunctionName | nvarchar | Yes |
| PermissionName | nvarchar | Yes |
| IsActive | bit | Yes |

1. Appendix

|  |
| --- |
| ***What is the use case?***   1. ***A use case is*** *a written description of how users will perform tasks on your website.  It outlines, from a user’s point of view, a system’s behavior as it responds to a request. Each use case is represented as a sequence of simple steps, beginning with a user's goal and ending when that goal is fulfilled. (*[*http://www.usability.gov/how-to-and-tools/methods/use-cases.html*](http://www.usability.gov/how-to-and-tools/methods/use-cases.html)*)* 2. ***A use case is*** *a list of steps, typically defining interactions between a role and a system. (*[*http://en.wikipedia.org/wiki/Use\_case*](http://en.wikipedia.org/wiki/Use_case)*)* 3. ***A use case is*** *a formal way of representing how a business interacts with its environment. it summarized into a single picture.* ([*http://romisatriawahono.net/lecture/sad/romi-sad-05-implementation-march2014.pptx*](http://romisatriawahono.net/lecture/sad/romi-sad-05-implementation-march2014.pptx)*)*   ***What are Benefits of use cases?***   1. ***Use cases*** *add value because they help explain how the system should behave and in the process, they also help brainstorm what could go wrong.  They provide a list of goals and this list can be used to establish the cost and complexity of the system. Project teams can then negotiate which functions become*[*requirements*](http://www.usability.gov/how-to-and-tools/methods/requirements.html)*and are built.*     *(*[*http://www.usability.gov/how-to-and-tools/methods/use-cases.html*](http://www.usability.gov/how-to-and-tools/methods/use-cases.html)*)*   1. *With the help of use case diagram, you can discuss and communicate:*  * *The scenarios in which your system or application interacts with people, organizations, or external systems.* * *The goals that it helps those actors achieve.* * *The scope of your system.*   *(*[*http://msdn.microsoft.com/en-us/library/dd409432.aspx*](http://msdn.microsoft.com/en-us/library/dd409432.aspx)*)* |