**Use Case Realization Report - Design**

Image result for instant edge logo

**The Enterprise Transformation Platform**

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
| Document Reference: | ISS\_IE\_UCRR-A\_2.0 |
| Client: | **I**nstant **E**dge |
| Project: | **I**nstant **E**dge- Manage Operations Module |
| Document Title: | Use case Realization Report - Design |
| Version | 2.0 |
| Date | 5/03/2017 |
| Author | Rameswari Mohanty |
| Approver | Mathias Behne |

**Amendment History:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Version | Review & Approval Form Number | Date | Brief description of amendments and affected pages, paragraph | Author | Reviewer |
| 1.0 | ISS/IE/UCRR-D/1.0 | 10/01/2017 | Initial version | Rameswari Mohanty | Vrinda Gupta |
| 2.0 | ISS/IE/UCRR-D/2.0 | 5/03/2017 | Changes in the Routing Incident Diagrams | Rameswari Mohanty | Zhao Pengcheng |

**Team Members:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Group** | **Matriculation No.** |
| Rameswari Mohanty | SE24/4FT | A0148532E |
| Vrinda Gupta | SE24/4FT | A0148518X |
| Vignesh Selvaraju | SE24/4FT | A0148576N |
| Zhao Pengcheng | SE24/4FT | A0148528W |

**Table of Contents**

**Analysis Model 3**

Analysis for Maintain Flowfieldcontrol 3

Class Diagram for Maintain Flowfieldcontrol diagram 3

Collaboration Diagram for Create Flowfieldcontrol diagram 3

<anonymous> 4

Collaboration Diagram for Delete Flowfieldcontrol diagram 8

Collaboration Diagram for Modify Flowfieldcontrol diagram 11

Architect 15

Architect 16

Architect 16

Flowfieldcontrol 17

FlowfieldcontrolController 18

FlowfieldcontrolList 19

FlowfieldcontrolScreen 20

Flowfieldgroup 20

FlowfieldgroupList 21

Flowmodel 21

FlowmodelList 22

Flowprocedure 22

FlowprocedureList 23

Main Screen 23

MainController 24

Analysis for Maintain Flowtype 52

Class Diagram for Maintain Flowtype diagram 52

Collaboration Diagram for Create Flowtype diagram 52

Collaboration Diagram for Delete Flowtype diagram 55

Collaboration Diagram for Modify Flowtype diagram 58

Architect 60

Flowtype 61

FlowtypeController 61

FlowtypeList 62

FlowtypeScreen 62

MainController 63

MainScreen 63

1. **Design Model**
   1. **Design for Maintain Flowfieldcontrol**
      1. **Class Diagram for Maintain Flowfieldcontrol diagram**



1. **Class Diagram for Maintain Flowfieldcontrol**
   * 1. **Sequence diagram for Create Flowfieldcontrol diagram**



1. **Sequence Diagram for Create Flowfieldcontrol**

To commence using the system, **the Architect is required to log-in** and will follow the below steps to create the Flowfieldcontrol :

* The **Architect** selecttheFlow field control option under the configuration on the **MainScreen** to be able to create a flowfieldcontrol.
* The control will be sent to the **MainController**. The **MainController** initiates the **FlowfieldcontrolController** to start the create process
* The **FlowfieldcontrolController** displays **Flowfieldcontrol Screen** to allow the **Architect** to input the Flowfieldcontrol field for creation of new flowfieldcontrol
* The **FlowfieldcontrolController** find out **Flow Procedure** from the **List of Flow Model**, and then gets the flow procedure associated with the model and shows on the screen.
* The Architect selects the procedure and search the flow field group in the screen.
* The **FlowfieldcontrolController** finds the flowfield group from the flowfieldgroup list and displays on the screen.
* The **Architect** enters the new **flowfieldcontrol** details on the screen and saves**.**
* The **FlowfieldcontrolController** checks if the **flowfieldcontrol** exists in the flow procedure it shows error**.**
* If the **flowfieldcontrol** doesn’t exists a new **flowfieldcontrol** is created and added to the flow procedure**.**
  + 1. **Sequence diagram for Modify/Delete Flowfieldcontrol diagram**



1. **Sequence Diagram for Modify/Delete Flowfieldcontrol**

To commence using the system, **the Architect is required to log-in** and will follow the below steps to delete the Flowfieldcontrol :

* The **Architect** selecttheFlow field control option under the configuration on the **MainScreen** to be able to create a flowfieldcontrol.
* The control will be sent to the **MainController**. The **MainController** initiates the **FlowfieldcontrolController** to start the create process
* The **FlowfieldcontrolController** displays **Flowfieldcontrol Screen** to allow the **Architect** to input the Flowfieldcontrol field for creation of new flowfieldcontrol
* The **FlowfieldcontrolController** find out **Flow Procedure** from the **List of Flow Model**, and then gets the flow procedure associated with the model and shows on the screen.
* The Architect selects the procedure and selects the flowfieldcontrol to be deleted and press delete.
* The **FlowfieldcontrolController** finds the flowfieldgroup and deletes it.
* The **FlowfieldcontrolController** removes the **flowfieldcontrol** from the corresponding **flow procedure** and display success message on the screen**.**
  1. **Design for Maintain Flowtype**

**1.2.1 Class Diagram for Maintain Flowtype diagram**



1. Class Diagram for Maintain Flowtype
   * 1. **Sequence Diagram for Create Flowtype diagram**



1. Collaboration Diagram for Create Flowtype

To commence using the system, **the Architect is required to log-in** and will follow the below steps to create the Flowfieldcontrol :

* The **Architect** selecttheFlow type option under the configuration on the **MainScreen** to be able to create a flowType.
* The control will be sent to the **MainController**. The **MainController** initiates the **FlowtypeController** to start the create process
* The **FlowtypeController** displays **FlowtypeScreen** to allow the **Architect** to input the Flowtype field for creation of new flowftype
* The **FlowtypeController** find out **Flowtype** from the **List of Flowtype**, and shows on the screen.
* The Architect enters the details of the new flow type on the screen.
* The **FlowtypeController** checks if the flow type exists displays error message on the screen.
* If the **flowtype doesn’t** exists a new **flowtype** is created and added to the flowtypelist**.**
  + 1. **Sequence Diagram for Modify/Delete Flowtype diagram**



1. Collaboration Diagram for Delete Flowtype

To commence using the system, **the Architect** is required to log-in and will follow the below steps to create the Flowtype :

* The **Architect** selecttheFlow type option under the configuration on the **MainScreen** to be able to delete a flowType.
* The control will be sent to the **MainController**. The **MainController** initiates the **FlowtypeController** to start the create process
* The **FlowtypeController** displays **FlowtypeScreen** to allow the **Architect** to select the Flowtype to be deleted
* The **FlowtypeController** find out **Flowtype** from the **List of Flowtype**, and shows on the screen.
* The Architect confirms the deletion of the selected flow Type.
* The **FlowtypeController** delete the **flow type** and removes it from the **flow type list**.
* The **FlowtypeController** displays success message on the screen**.**
  1. **Design for Push Notification**

1. **Class Diagram for Push Notification diagram**



1. Architecture Overview - Subsystem layering
   * 1. **Sequence Diagram for login diagram**



1. Sequence Diagram : Login

The precondition for the Log-in function is that the IE Mobile application is launched, the **Login Screen** is displayed and the Log-in option is enabled. The requirements for any **User** to log-in are as follows:

1. The **User** log-in the systems by selecting the Log-in option from the **LoginScreen**. When this occurs, the Login**Screen** instructs **AuthenticationController** to perform login operation.
2. The **AuthenticationController** initiates **IEAuthenticationControl** to perform login operation.
3. The **User** enters username and password into the **LoginScreen**.
4. The **LoginScreen** validates the format of the username and password.
5. If the format is invalid, the **LoginScreen** will display an error message. Repeat the steps to allow entering of userid and password.
6. If the format is valid, the **LoginScreen** will send the entered username and password to the **AuthenticationControl**.
7. The **AuthnticationControl** validates if a valid username and password is entered through searching the user information from **UserList via the IEAuthenticationControl in the Server**. The **UserList** gets the attribute password from the **User** object it is holding.
8. If the username and password are valid, the **IEAuthenticationControl** will return success.
9. If the validation of username and password failed, the **IEAuthenticationControl** will return an error and eventually the AuthenticationController will instruct the **LoginScreen** to display a log-in failure message. Repeat the steps to allow entering of userid and password.
10. If the validation of username and password is successful, the **AuthenticationController** will instruct the **LoginScreen** to display a successful message.

The following exceptions have been identified:

**Log-in Failed**. This is handled in the standard flow in para (9).

* + 1. **Sequence Diagram for logout diagram**



1. Sequence Diagram : Logout

The precondition for the Logout function is that the IE Mobile application is launched, the User has logged in. The requirements for any **User** to log-out are as follows:

1. The User selects the logout option in the BrowserScreen.
2. The BrowserScreen instructs the BrowserController of the logout option.
3. The BrowserController invokes the logout method from the AuthenticationController
4. The AuthenticationController performs the logout routines and returns the status to the BrowserController.
5. The BrowserController instructs the BrowserScreen to display the Success message.
   * 1. **Sequence Diagram for Browser Platform**



1. Sequence Diagram : Browse Platform

The precondition for the Logout function is that the IE Mobile application is launched, the User has logged in. The requirements for any **User** to Browse Platform are as follows:

1. The User selects the logout option in the BrowserScreen.
2. The BrowserScreen instructs the BrowserController of the logout option.
3. The BrowserController invokes the logout method from the AuthenticationController
4. The AuthenticationController performs the logout routines and returns the status to the BrowserController.
5. The BrowserController instructs the BrowserScreen to display the Success message.
   * 1. **Sequence Diagram for Deregister for Push Notifications diagram**



1. Sequence Diagram : Deregister for Push Notifications

The precondition for the Log-out function is that the User has successfully logged-in to PRMS. The requirements for any **User** to log-out are as follows:

1. The **User** selects the Log-out option from the **Browser Screen**.
2. The system instructs the AuthenticationController to perform logout routines.
3. The AuthenticationController sends the IEDeviceToken to the IEDeviceControl to deregister device
4. The IEDeviceControl reads the User from the UserList to get all devices for the User.
5. The Corresponding Device is read for the user and removed.
6. The IEDeviceControl sends success response to the AuthenticationController.
7. The AuthenticationController finishes the routine and instructs the Browser screen to display success.
   * 1. **Sequence Diagram for Register With FCM diagram**



1. Sequence Diagram : Register With FCM

The precondition for the Register with FCM function is that the User has allowed Remote Notifications for the Instant Edge Mobile App. The requirements for any **User** to Register with FCM are as follows:

1. The **User** opens the App and starts the registering routine.
2. The Login Screen is displayed and the Device token is generated by the AuthenticationController and sent to the Google's FCM Server.
3. The FCM Server saves the Device token against the App (registered) and generates a FCM token for the App.
4. The AuthenticationController retrieves the FCM token and saves it for future registration for push notifications with the Instant Edge server.
   * 1. **Sequence Diagram : Register for push notification diagram**



1. Sequence Diagram : Register With FCM
   * 1. **Sequence Diagram : Register for Push Notifications diagram**
2. Sequence Diagram : Register for Push Notifications

The precondition for the Send Push Notification function is that the **User** has logged in and allowed Remote Notifications for the Instant Edge Mobile App. The requirements for Instant Edge Platform to **Send Notification** are as follows:

1. The **IEPlatform** triggers a notification on any configured action.
2. The **FCMEndPoint** instructs the **IEDeviceControl** to send notification to respective users with the payload it mentions.
3. The **IEDeviceControl** reads the device of user and reads the **FcmToken** of the device from the **UserDevice**.
4. The **IEDeviceControl** proceeds to build the **FcmMessage** using the **FcmToken**, payload and other parameters and ends by sending the notification out.
   * 1. **Sequence Diagram : Send Notifications diagram**



1. Sequence Diagram : Send Notifications

The precondition for the Send Push Notification function is that the **User** has logged in and allowed Remote Notifications for the Instant Edge Mobile App. The requirements for Instant Edge Platform to **Send Notification** are as follows:

1. The **IEPlatform** triggers a notification on any configured action.
2. The **FCMEndPoint** instructs the **IEDeviceControl** to send notification to respective users with the payload it mentions.
3. The **IEDeviceControl** reads the device of user and reads the **FcmToken** of the device from the **UserDevice**.
4. The **IEDeviceControl** proceeds to build the **FcmMessage** using the **FcmToken**, payload and other parameters and ends by sending the notification out.
   1. **Design for Routing Incident**
      1. **Class Diagram for Routing Incident**



* + 1. **Sequence Diagram for Routing Incident**

