**Functional Specification Document**

**Project Name: MOU Client on Android**

**Project FP code: FP/PRJ**

**Project owner: Pankaj Choudhary**

**FSD Version No: 0.1**

|  |  |
| --- | --- |
| **DOCUMENT CONTRIBUTORS** | |
| **Role** | **Name** |
| Project Name | MOU Client on Android |
| Business Analyst | Neerav |
| Project Delivery Leader | Partha Sarthi |
| Project Manager | Pankaj Choudhary |

Table of Contents

[1. Business Overview / Introduction 5](#_Toc480977172)

[2. Abbreviation 6](#_Toc480977173)

[3. Scope 6](#_Toc480977174)

[3.1. In Scope 6](#_Toc480977175)

[3.2 Excluded from Scope 6](#_Toc480977176)

[4. Assumptions 6](#_Toc480977177)

[5. Target user/Actors 6](#_Toc480977178)

[6. Constraints 7](#_Toc480977179)

[7. Dependencies 7](#_Toc480977180)

[8. Functional Requirement 7](#_Toc480977181)

[8.1. Use Cases 8](#_Toc480977182)

[8.1.1. Use Case 1.0 8](#_Toc480977183)

[8.1.1.1. Use Case Diagram 8](#_Toc480977184)

[8.1.1.2. Use Case Description 8](#_Toc480977185)

[8.1.2. Use Case 2.0 10](#_Toc480977186)

[8.1.2.1. Use Case Diagram 10](#_Toc480977187)

[8.1.2.2. Use Case Description 10](#_Toc480977188)

[8.1.3 Use Case 3.0 12](#_Toc480977189)

[8.1.3.1 Use Case Diagram 12](#_Toc480977190)

[8.1.3.2 Use Case Description 12](#_Toc480977191)

[8.1.4 Use Case 4.0 14](#_Toc480977192)

[8.1.4.1 Use Case Diagram 14](#_Toc480977193)

[8.1.4.2 Use Case Description 14](#_Toc480977194)

[8.1.5 Use Case 5.0 15](#_Toc480977195)

[8.1.5.1 Use Case Diagram 15](#_Toc480977196)

[8.1.5.2 Use Case Description 15](#_Toc480977197)

[8.1.6 Use Case 6.0 20](#_Toc480977198)

[8.1.6.1 Use Case Diagram 20](#_Toc480977199)

[8.1.6.2 Use Case Description 20](#_Toc480977200)

[8.1.7 Use Case 7.0 21](#_Toc480977201)

[8.1.7.1 Use Case Diagram 21](#_Toc480977202)

[8.1.7.2 Use Case Description 21](#_Toc480977203)

[8.1.8 Use Case 8.0 26](#_Toc480977204)

[8.1.8.1 Use Case Diagram 26](#_Toc480977205)

[8.1.8.2 Use Case Description 26](#_Toc480977206)

[8.2 Entities Involved 30](#_Toc480977207)

[8.3 Screenshots and Mock-up headings 31](#_Toc480977208)

[9 Risk Section 34](#_Toc480977209)

[10 Hardware/Software Requirements 34](#_Toc480977210)

[11 Non Functional Requirement 34](#_Toc480977211)

[11.1 Performance 34](#_Toc480977212)

[11.2 Usability 35](#_Toc480977213)

[11.3 Security 35](#_Toc480977214)

[11.4 Backup/Recovery Policy 36](#_Toc480977215)

[11.5 Volumetric Requirements 36](#_Toc480977216)

[12 Application Impacts 36](#_Toc480977217)

[13 Issues 36](#_Toc480977218)

[Revision History 37](#_Toc480977219)

# Business Overview / Introduction

The Unique Identification Authority of India (UIDAI) provides a Unique Identity (Aadhaar) to all Indian residents. The UIDAI proposes to provide online authentication using demographic and biometric data.

For mobiles to be instruments of authentication for digital identity, mobile number should be unique, authenticable and fulfil requirements of non-repudiation. A possible way of achieving the same would be to link the mobile numbers of users to their Aadhaar and use Aadhaar OTP as one of the user authentication scheme. Aadhaar system already offers a mechanism to keep mobile number linked to Aadhaar identity and use it via mobile OTP (One Time Pin) based authentication.

Use of Aadhaar linked mobile OTP authentication allows the following:

* Enable remote and secure verification of an Aadhaar holder using mobile during mobile based services.
* Enable use of mobiles as an identity instrument and trusted authentication factor that is attached to Aadhaar, thereby simplifying online access to public services.

In addition, registered mobile numbers are used for resident communication, providing UIDAI mobile application for features such as HOTP, biometric locking, etc.

Such large scale usage assumes that mobile number in Aadhaar database is up to date. In addition to currently existing update mechanisms (permanent update centers, self-service update portal, etc.), if UIDAI offers a mobile update API for “trusted” authentication user agencies (such as specific Government AUAs, banks, etc.), few lakhs of biometric update touch points can be made available to residents for easy linking and updating of their mobile number within Aadhaar system.

This document is the proposed use Mobile Update API for use within trusted Enrollment Operators. A new mobile update Android client will be developed where the mobile update API will be integrated and a new operator type “MOU Operator “will be introduced .These operator can use the MOU client and can perform the mobile update enrollment’s from the new MOU Client.

# Abbreviation

|  |  |
| --- | --- |
| **Terms** | **Meanings** |
| UIDAI | Unique Identity Authority of India |
| RO | Regional Office |
| EA | Enrolment Agency |
| POI/POA/ POR | Proof of identity/ Proof of address/ proof of relationship |
| CIDR | Central Identification Data repository |

# Scope

## In Scope

The scope of this project includes the following requirements:

* MOU Client
* Integrated of MOU API with MOU Android client
* Integration of MOU Client with Registrar –PAC Validation API
* MOU Client –Operator Sync

## Excluded from Scope

Section not mentioned in ‘Included in Scope’ is considered Out of Scope.

# Assumptions

# Target user/Actors

|  |  |
| --- | --- |
| Sl. No. | Users/Actors |
| 1 | UIDAI Admin |
| 2 | Registrar |
| 3 | Enrolment Agency |
| 4 | Regional Office |
| 5 | MSP |

# Constraints

N/A

# Dependencies

# Functional Requirement

|  |  |  |
| --- | --- | --- |
| **Requirement ID** | **Requirement** | **Use Case Mapping** |
| Req01 | Client Registration | [**UC1.0**](#_Use_Case_1.0) |
| Req02 | Client De-Registration | [**UC2.0**](#_Use_Case_2.0) |
| Req03 | Operator On-boarding | [**UC3.0**](#_Use_Case_3.0) |
| Req04 | Operator Login | [**UC4.0**](#_Use_Case_4.0) |
| Req05 | Mobile Number/email update | [**UC5.0**](#_Use_Case_5.0) |
| Req06 | Successful Mobile Update EID list | [**UC6.0**](#_Use_Case_6.0) |
| Req07 | Payment Authorization Code (PAC validation) Hook - Integration of MOU Client with Registrar API | [**UC7.0**](#_Use_Case_7.0) |
| Req08 | Operator Sync | [**UC8.0**](#_Use_Case_8.0) |

## Use Cases

### Use Case 1.0

### Use Case Diagram

N/A

### Use Case Description

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name: Client Registration** | | | | | | |
| Use Case ID | | UC 1.0 | | | | |
| Actors | | UIDAI Admin | | | | |
| Pre-condition(s) | | 1. The internet facility should be available for client registration 2. The admin user should have logged-in user credentials | | | | |
| Trigger | | The actor clicks on the link “Client registration” | | | | |
| Description | | This use case would allow the admin user to register the client so that enrolment can be performed. The client details to be entered are Registrar ID, EA Name, Station ID, Location, Admin username and Password. All other icons (Enrolment, Operator, Upload, Registration, Settings and Find Aadhaar) will remain disabled until the client is registered on the server. | | | | |
| PII Data accessed | | Read | | | | |
| **Normal Flow** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
| 1 | Actor clicks on link “Client registration” | | | | System displays the screen “Client registration” with following details:   * Registration information * Registrar ID * Enrolment Agency * Station ID * Location * Portal Credentials * Admin username * Password | |
| 2 | Actor enters the details and clicks on “Register” button | | | | System sends the client data to server and then displays a success message | |
| 3 | Actor clicks on “Reset” button | | | | System clears the entered values. | |
| **Alternate Flow** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
|  |  | | | |  | |
| Post Conditions | | | 1. System saves the client registration details in the system and displays a success message. 2. Once client is registered, the following icons will get enabled:  * Enrolment, * Operator, * Upload, * Registration, * Settings and * Find Aadhaar | | | |
| Errors/Exception | | | Description | | | Error Code and Error Message |
| 1. If actor clicks on “register” without entering registrar ID 2. If actor clicks on “register” after not entering 3- digit value in registrar ID 3. If actor clicks on “register” without entering enrolment agency 4. If actor clicks on “register” without entering station ID 5. If actor clicks on “register” after not entering 5- digit value in station ID 6. If actor clicks on “register” without entering location 7. If actor clicks on “register” without entering username 8. If actor clicks on “register” without entering password 9. If actor clicks on “register” after entering value other than number in registrar ID or station ID | | | 1. “Please enter registration ID” 2. “Please enter 3-digit registrar ID.” 3. “Please enter enrolment agency” 4. “Please enter station ID” 5. “Please enter 5-digit station ID.” 6. “Please enter location” 7. “Please enter username” 8. “Please enter password” 9. “Please enter numeric value.” |
| Business Rules | | | 1. The following fields are mandatory and must be filled in:  * Registrar ID * Enrolment Agency * Station ID * Location * Username * Password  1. Registrar ID should be 3- digit value 2. Station ID should be 5- digit value | | | |
| Entities Involved | | | UIDAI Admin, Enrolment agency | | | |
| Assumptions | | | Client registration need to be done online as it is an online version of CEL. | | | |
| Issue List | | | Issues | Status | | |
|  |  | | |

### Use Case 2.0

### Use Case Diagram

N/A

### Use Case Description

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name: Client De-Registration** | | | | | | |
| Use Case ID | | UC 2.0 | | | | |
| Actors | | UIDAI Admin | | | | |
| Pre-condition(s) | | 1. The internet facility should be available for client registration 2. Client should be registered in the system | | | | |
| Trigger | | The actor clicks on the link “Client de-registration” | | | | |
| Description | | This use case would allow the admin user to de- register the client which is registered in the system. Once the user is de-registered, mobile update should not be allowed by that user. | | | | |
| PII Data accessed | | Read | | | | |
| **Normal Flow** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
| 1 | Actor clicks on link “Client de- registration” | | | | System displays the screen “Client de-registration” with following filled in details:   * Registration information * Registrar ID * Enrolment Agency * Station ID * Location * Portal Credentials * Admin username * Password | |
| 2 | Actor enters the details and clicks on “De-Register” button | | | | System sends the de-register request to server and display a success message | |
| **Alternate Flow** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
|  |  | | | |  | |
| Post Conditions | | | 1. System sends the request to de-register the client and displays a success message. | | | |
| Errors/Exception | | | Description | | | Error Code and Error Message |
|  | | |  |
| Business Rules | | | 1. Once the client is de-registered, no mobile update should be allowed by that client | | | |
| Entities Involved | | | UIDAI Admin, Enrolment agency | | | |
| Assumptions | | |  | | | |
| Issue List | | | Issues | Status | | |
|  |  | | |

### Use Case 3.0

### Use Case Diagram

N/A

### Use Case Description

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name: Operator On-boarding** | | | | | | |
| Use Case ID | | UC 3.0 | | | | |
| Actors | | UIDAI Admin | | | | |
| Pre-condition(s) | | The internet facility should be available to on-board the operator  The operator should be already on-boarded as per UIDAI guidelines in the Admin portal. | | | | |
| Trigger | | Actor clicks on link “Operator authorization” | | | | |
| Description | | The use case would be used to on-board the operator by biometric authentication. For on-boarding process, the name and Aadhaar of operator along with biometric is required. | | | | |
| PII Data accessed | | Read | | | | |
| **Normal Flow** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
| 1 | Actor clicks on the link “Operator authorization “ | | | | System displays the form with following fields:   * Operator ID * Operator name * Aadhaar number of operator * Biometric of operator | |
| 2 | Actor enters the details and clicks on “Add operator” button | | | | System verifies the details and displays a success message | |
| 3 | Actor clicks on “Reset” button | | | | System clears the entered values. | |
| **Alternate Flow** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
| 1 |  | | | |  | |
| Post Conditions | | | 1. System authenticates the operator details and displays a success message | | | |
| Errors/Exception | | | Description | | | Error Code and Error Message |
| 1. If operator Id has not been entered 2. If operator name has not been entered 3. If Aadhaar number has not been entered 4. If value other than number has been entered 5. If 12-digit value has not been entered 6. If biometric details has not been captured 7. If operator name is not same as written on Aadhaar 8. If biometric details does not match with the Aadhaar number | | | 1. “Please enter operator Id” 2. “Please enter operator name” 3. “Please enter Aadhaar Number” 4. “Only numeric value is allowed” 5. “Aadhaar number should be 12-digit value. Please enter valid Aadhaar Number” 6. “Please enter biometric details” 7. “Operator name should be same as written on mentioned Aadhaar number” 8. “Biometric details does not match against the mentioned Aadhaar number” |
| Business Rules | | | 1. The following fields are mandatory and must be filled in:  * Operator Id * Operator name * Aadhaar number of operator * Biometric details  1. Aadhaar number should be a 12- digit numeric value. 2. On-boarding process should be done against only registered operator at a time 3. Operator name should be same as written on mentioned Aadhaar number 4. The operator type should be of MOU Operator. 5. In case the operator authentication fails then the operator has to update his biometric from the standard client to work on enrolment. 6. Operator should be active on server and the status of Aadhaar should be valid for successful biometric authentication. | | | |
| Entities Involved | | | UIDAI Admin, Operator, Aadhaar data | | | |
| Assumptions | | | The operator should be having valid Aadhaar so that biometric can be authenticated successfully. | | | |
| Issue List | | | Issues | Status | | |
| NA | NA | | |

### Use Case 4.0

### Use Case Diagram

N/A

### Use Case Description

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name: Operator Login** | | | | | | |
| Use Case ID | | UC 4.0 | | | | |
| Actors | | Enrollment Operator | | | | |
| Pre-condition(s) | | The internet facility should be available for logging into the application | | | | |
| Trigger | | Actor access the MOU Client Android application | | | | |
| Description | | The use case would enable the operator to logged in into the application by entering Aadhaar number and biometric authentication | | | | |
| PII Data accessed | | Read | | | | |
| **Normal Flow** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
| 1 | Actor enters the MOU Client Android application | | | | System displays the login screen with following details:   * Aadhaar number * Biometric Details | |
| 2 | Actor enters the details and clicks on Login button | | | | System validates the login details and redirects the actor to home screen of application | |
| **Alternate Flow** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
|  |  | | | |  | |
| Post Conditions | | | 1. System validates the operator details and redirects to home screen. | | | |
| Errors/Exception | | | Description | | | Error Code and Error Message |
| 1. If Aadhaar number has not been entered 2. If invalid Aadhaar number s entered 3. If biometric details has not been captured 4. If biometric detail does not match against the entered Aadhaar number | | | 1. “Please enter Aadhaar number” 2. “Please enter 12-digit Aadhaar number” 3. “Biometric details should be captured.” 4. “Biometric detail does not match against the entered Aadhaar number.” |
| Business Rules | | | 1. The following fields are mandatory and must be filled in:  * Aadhaar number * Biometric detail  1. Aadhaar number should be 12- digit value. | | | |
| Entities Involved | | | Operator, Aadhaar data | | | |
| Assumptions | | | The operator should be registered on the Admin portal/server  The operator should have UID for authentication. | | | |
| Issue List | | | Issues | Status | | |
|  |  | | |

### Use Case 5.0

### Use Case Diagram

N/A

### Use Case Description

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name: Mobile number/ Email Update** | | | | | | |
| Use Case ID | | UC 5.0 | | | | |
| Actors | | Enrolment Operator | | | | |
| Pre-condition(s) | | 1. The internet facility should be available for mobile number update. 2. The enrolment operator should be logged in. 3. Valid Aadhaar Number of Resident and Enrolment operator should be available. | | | | |
| Trigger | | Actor will access the MOU Client Android application and clicks on “Mobile Update” link available on Home Screen. | | | | |
| Description | | MOU client will allow residents to update their mobile number or email ID or both which has been registered during enrolment. Once the request for modification of the mobile number/ email ID will be raised, then the same mobile number needs to be authenticated via Aadhaar/ OTP Authentication and the biometric of the enrolment operator and the resident. MOU Client Android application will consume Mobile Update API which will captures Aadhaar number, resident disclosure, new mobile number, verification code, and biometrics of the resident. In addition, optional attributes email ID can also be captured and updated. | | | | |
| PII Data accessed | | Read | | | | |
| **Normal Flow** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
| 1 | Actor Clicks on Link “Mobile update” available on home screen of the Application. | | | | System displays the generate OTP screen with following fields:-   * New Mobile (10 digits ,non-empty) * Generate OTP Button | |
| 2 | Actor enters the data and clicks on “Generate OTP” button. | | | | Actor receives an OTP on the mobile no. filled by in the last screen.  System displays the Mobile update screen with the following details:-   * Resident Aadhaar (**mandatory**) * Resident Biometric(**mandatory**) * Operator Aadhaar(**Pre populated non-editable field**) * Operator Biometric (**mandatory**) * OTP (**mandatory,6 digits**) * Email (optional, **editable field**) * New Mobile (**Pre populated non-Editable field**) * Resident Closure (checkbox) | |
| 3 | Actor enters the data and checked the “resident disclosure” | | | | System will display a pop up displaying the message :-  **Disclosure under section 3(2) of THE AADHAAR (TARGETED DELIVERY OF FINANCIAL AND  OTHER SUBSIDIES, BENEFITS AND  SERVICES) ACT, 2016:**  I confirm that I have been residing in India for at least 182 days in the preceding 12 months & information (including biometrics) provided by me to the UIDAI  is my own and is true, correct and accurate. I am aware that my information including biometrics will be used for update of Aadhaar and authentication. I understand that my identity information (except core biometric) may be provided to an agency only with my consent during authentication or as per the provisions of the Aadhaar Act. I have a right to access my identity information (except core biometrics) following the procedure laid down by UIDAI.  There will be YES button to accept the disclosure. | |
| 4 | Actor clicks on YES button and click on Submit button | | | | System validates the data and directs the user to Validate PAC Screen. | |
| **Alternate Flow :-** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
|  |  | | | |  | |
| Post Conditions | | | System validates the Aadhaar data and displays a success message.  Resident receives the URN message for the successfully created mobile/email request. | | | |
| Errors/Exception | | | Description | | | Error Code and Error Message |
| 1. If Resident’s mobile number has not been entered. 2. If mobile number is not a 10 digits value. 3. If email address is invalid. 4. If Resident’s Aadhaar number has not been entered. 5. If invalid Aadhaar number is entered. 6. If Aadhaar number is not a 12 digits value. 7. If biometric details of resident has not been captured. 8. If biometric details of operator has not been captured. 9. If Resident Disclosure is unchecked. 10. If OTP has not been entered. 11. If entered OTP is not 6-digit value 12. If OTP entered is incorrect. 13. If biometric details has not been matched with the entered Aadhaar number | | | 1. “Please enter mobile number” 2. “Mobile number should be of 10 digits.Please enter valid mobile number” 3. “Please enter valid email address” 4. “Please enter Aadhaar of Resident.” 5. “Aadhaar number should be 12-digit value. Please enter valid Aadhaar number” 6. “Aadhaar number should be 12-digit value. Please enter valid Aadhaar number ” 7. “ Please provide biometric details of Resident” 8. “ Please provide biometric details of Operator” 9. “Please check Resident consent.” 10. “Please enter OTP” 11. “OTP should be 6-digit value” 12. “Please enter correct OTP” 13. “Your biometric data is not matched with Aadhaar number” |
| Business Rules | | | 1. The following fields are displayed and need to be filled by user:  * New Mobile - mandatory * Resident Aadhaar- mandatory * Resident Biometric - mandatory * Operator Aadhaar - mandatory * Operator Biometric - mandatory * Resident Disclosure- mandatory * OTP – mandatory * New Email- optional  1. At the time of Mobile number/ email update both operator and Resident have to authenticate via Biometric authentication. 2. Unique OTP should be generated by generic OTP for each transaction. 3. Mobile number should be of 10 digit and Aadhaar number should be of 12 digits. 4. Update packet will be created only after mobile OTP authentication, Operator biometric authentication and Resident biometric authentication. 5. The OTP will be sent as follows:  * Mobile update (new mobile number, old email ID or no email ID): OTP will be sent on new mobile number * Email update (old/existing mobile number, new email ID): OTP will be sent on old/ existing mobile number * Mobile and email update (New mobile number, new email ID): OTP will be sent on new mobile number  1. In case only email ID need to be updated, system should consider it as mobile update packet where system will take existing mobile number as new mobile number 2. OTP should be 6-digit value 3. Auth Generic OTP will be used to send the OTP 4. Operator Aadhaar number should be non-editable or in read-only mode. The Aadhaar should be same from which operator has logged in into the CELC client. 5. New Mobile number on update mobile screen should be non-editable or in read-only mode. Mobile number should be same as provided in the Generate OTP screen. 6. For each MOU request operator authentication is mandatory. 7. For successful MOU update request from the client to MOU API actor needs to mandatorily accept the resident disclosure or resident needs to mandatorily accept the disclosure .If resident did not accept the disclosure mobile update request should not happen. 8. Only resident can update their mobile number/email ID from this application, child cannot use this application to update his mobile number and email ID. | | | |
| Entities Involved | | | Resident and Operator Aadhaar Data | | | |
| Assumptions | | | The update packet will be generated on the server side and be put on the update workflow for Aadhaar generation. | | | |
| Issue List | | | Issues | Status | | |
|  |  | | |

### Use Case 6.0

### Use Case Diagram

N/A

### Use Case Description

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name: Successful Mobile Update EID List** | | | | | | |
| Use Case ID | | UC 6.0 | | | | |
| Actors | | Enrolment Operator | | | | |
| Pre-condition(s) | | The enrolment operator should be logged in. | | | | |
| Trigger | | Actor will access the MOU Client Android application and clicks on “MOU List” link available on Home screen. | | | | |
| Description | | Operator can see the successful mobile update enrolment IDs using this tab. The report will show the top 1000 (according to latest response time) successful enrolment ID for mobile/email update done from the client/device. | | | | |
| PII Data accessed | | Read | | | | |
| **Normal Flow** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
| 1 | Actor Clicks on Link “MOU List” available on home screen. | | | | System displays the generate screen with following fields:-   * EID * Creation Time (Successful response time)   System would show the last 1000 results. | |
| **Alternate Flow** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
| 1 |  | | | |  | |
| Post Conditions | | | System display the the last 1000 successful EID, Creation Time (Successful response time) | | | |
| Errors/Exception | | | Description | | | Error Code and Error Message |
|  | | |  |
| Business Rules | | | * Historical data will be maintained in local client /device DB * Only 1000 successful EID will be displayed. | | | |
| Entities Involved | | | Resident and Operator Data | | | |
| Assumptions | | | NA | | | |
| Issue List | | | Issues | Status | | |
|  |  | | |

### Use Case 7.0

### Use Case Diagram

N/A

### Use Case Description

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name: Payment Authorization Code (PAC validation)** | | | | | | |
| Use Case ID | | UC 7.0 | | | | |
| Actors | | Enrolment Operators | | | | |
| Pre-condition(s) | | (1)Actor should have valid user credentials to access the application  (2) Client is configured with payment required flag YES. | | | | |
| Trigger | | Actor has reviewed the update data and then submits. | | | | |
| Description | | Once the payment is made for a mobile/email update, the registrar/CSC system backend or payment portal should issue a usable "payment authorization code - PAC" (it can be shown on the screen and/or send to operator mobile, etc.)  Client shall allow operator to type in the PAC before mobile update request. UIDAI client will do an HTTPS call to configured URL (this SHOULD NOT be editable by EA/REG/Operator) to validate the PAC. URL should be configurable at the backend at registrar/EA level which is downloaded as part of client configuration (non-modifiable). If "payment required" flag is enabled, client should mandate the operator to type in the PAC without which mobile update will not be completed. The registrar backend will validate PAC and return success/failure to back to client. If it is valid, Client would proceed to save the mobile update and complete mobile update flow. Otherwise, a valid PAC will have to be retyped. Operator can also "cancel" the mobile/email update request altogether and start again.  **Changes in client:**  Client configuration to have the following configuration entries (all non-modifiable by EA/REG/Operator):   * "Payment-api-url=<https://paymentportal/validatePAC>", * "Payment-service-api-key=", * “PAYMENT\_REQUIRED=TRUE/FALSE"   Based on the flag settings and update parameter value, client shall pop up the payment confirmation dialog (it should have a message, a text box for PAC entry, OK/Cancel button) before making and update request.  On OK button, validate PAC to be "non-empty" and call the payment API URL and pass input:-   * Organization: - UIDAI * API Token * Client Name: - MOU Client * Client Version * Registrar-code * EA-code * Operator-id   **Transaction Info:-**   * Tx ID (Unique) * Service-type: MOBILE-EMAIL-UPDATE * PAC   **Backend API should return the following:-**   * Return value: - “SUCCESS | FAILURE" * Tx id * Error-message   If success, proceed. Else show error message to operator and pop-up PAC dialog again. Of course, operator can click "cancel" button on the dialog and cancel the whole update flow if they wish. | | | | |
| PII Data accessed | | Read | | | | |
| **Normal Flow: When Payment required flag is enabled** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
| 1 | Actor enters information in Mobile update screen and submits | | | | * System directs actor to a screen which asks to enter PAC Code | |
| 2 | Actor enters the PAC code provided and clicks on Submit button | | | | * System will call the Registrar API URL (API URL is already configured in the client) and will pass the following parameters :-  1. Organization 2. API Token 3. Client Name 4. Client version 5. Registrar Code 6. EA code ,Operator ID 7. Tx ID, Service Type 8. PAC.  * Registrar API will validate the request and will return the following parameters:-  1. Success/Failure 2. Tx ID 3. Error Message (in case of failure).  * If PAC is validated, the client will prepare a mobile update request to MOU API. * If the MOU server authenticates the MOU request, the server will send the EID to the MOU client of that MOU update request. * This EID will be pushed from the client to Registrar API | |
| **Alternate Flow: When Payment required flag is disabled** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
| 1 | Actor review the data and clicks on Confirm button | | | | System displays the “Biometric confirmation screen” with following field:   * Resident Confirmation- biometric * Operator Confirmation -biometric * Introducer Confirmation- biometric * Attach Document * Submit * Cancel   Validate PAC is not displayed | |
| Post Conditions | | | System got the “SUCCESS” response from Registrar API for the PAC code .So PAC verification got successful and update packet will be created.  Once PAC is verified, Submit button will be enabled.  Each transaction will be logged in a log file which will have the parameters details that UCL client are sending to registrar API and the response receiving from API. | | | |
| Errors/Exception | | | Description | | | Error Code and Error Message |
| 1. If actor clicks on Submit button without entering PAC 2. If UCL client receives “FAILED”Status and Invalid PAC like message from Registrar API. | | | 1. “Please enter PAC”. 2. “Please enter correct PAC” |
| Business Rules | | | 1. Unique EID id –PAC (same PAC can be used) combination for each transaction will be generated by Client. 2. UIDAI client will do an HTTPS call to configured URL (this SHOULD NOT be editable by EA) to validate the PAC. URL should be configurable at the backend at registrar/EA level which is downloaded as part of client configuration (non-modifiable). If "payment required" flag is enabled, UCL client should mandate the operator to type in the PAC without which update would not be completed. 3. UIDAI client configuration to have the following configuration entries (all non-modifiable by EA/Operator):  * "Payment-api-url=<https://paymentportal/validatePAC>", * "Payment-service-api-key=", * "Demo-update-payment-required=TRUE/FALSE", * "Mobile-email-update-payment-required=TRUE/FALSE", * "Photo-update-payment-required=TRUE/FALSE", * "Bio-update-payment-required=TRUE/FALSE" * “PAYMENT\_REQUIRED=TRUE/FALSE"  1. A log file shall be created which will have the parameters details that UCL Client are sending to the registrar API and the response receiving from API. So a log file will be generated with below parameters in a file (file format can be of any type .txt,.csv,.PDF) on the operator’s local machine :-    1. Request parameters (i.e. what we are sending to CSC API/web service)    2. Response parameters (i.e. what we are receiving from CSC API/web service)    3. Time stamp (when we are sending /receiving) 2. Handling of scenarios where the PAC was approved but the update was rejected during the update processing will not be in scope for UCL client and should be handled at the registrar/CSC API layer. 3. Registrar server can send only 2 responses either FAILED OR SUCCESS, whatever text server will send as a response it will be displayed in the client. 4. Submit button should be enabled only when the registrar server sends SUCCESS response 5. “Validate PAC” button should be displayed only when payment required flag is YES | | | |
| Entities Involved | | | Operator, UIDAI | | | |
| Assumptions | | | * The PAC will be validated in CSC service and can be used multiple times as well. No control of validation of PAC will be done at the UCL client. Only validation of PAC filed in the client is that Maximum length of PAC will be 25 and can be of alphanumeric type. * Update of resident will be dependent on CSC API return message, on successful response, UIDAI will go ahead and process the update. On Failure response, UIDAI will not update the data and will show an error message about invalid PAC entered in the system. UCL application would not have any control of the CSC API responses. * The scenario of PAC is approved at UCL and later on packet got rejected would be out of scope for MSP perspective and should be handled by the registrar/CSC API. * Registrar site already has SSL domain certificate deployed. It is a server side certificate and nothing is needed on the client side. Client just need to make https call to CSC server. So no security certificates needs to be installed at client side. * If at the time of validating the PAC, the enrolment ID is already generated so in the case if validation of PAC got failed the sequence will get wasted. * There is no restriction on the retry‘s number i.e. number of times operator can re-try in in case of failed transactions. * Local Language Update will it be part of Demo-update-payment-required. * Handling of scenarios where the PAC was approved but the update was rejected during the update processing will not be in scope for client and should be handled at the registrar API layer. | | | |
| Issue List | | | Issues | Status | | |
|  |  | | |

### Use Case 8.0

### Use Case Diagram

N/A

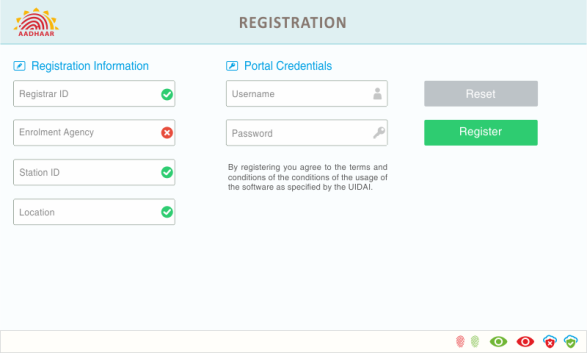
### Use Case Description

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name: Operator Sync** | | | | | | |
| Use Case ID | | UC 8.0 | | | | |
| Actors | | Enrolment Operators | | | | |
| Pre-condition(s) | | (1)Actor should have valid user credentials to access the application  (2) Client is configured with payment required flag YES. | | | | |
| Trigger | | Actor taps on Operator Sync Link on the home screen | | | | |
| Description | | Operator Sync will be required by operator to perform biometric authentication and Aadhaar OTP at least once in 10 days on the client in which he is on boarded. The on boarded operators who do not do operator sync within 10 days, they cannot perform the mobile update from that Client. This will also ensure that mobile number of operator is in operation & working.  It provides facility to update mobile update centers locations and operator’s details so during the operator sync i.e. operator’s Authentication (Biometric + OTP) on the same screen operator will sync the center and operator’s details at server end. So for this sync some additional parameters such as name of the center, type of the center (PEC/CAMP), complete address of the center, operator name & his mobile etc. will be taken in the ECMP client. This will ensure that list of enrolment centers are always updated at the server end. This may be implemented in all types of clients in future. Current scope is only at ECMP. | | | | |
| PII Data accessed | | Read | | | | |
| **Normal Flow: When Payment required flag is enabled** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
| 1 | Actor taps upon Operators Sync Link on Mobile Update Page | | | | * Operator sync screen will be arrive where operator will authenticate himself. * Operator Sync window displays following fields:-  1. Registrar (Automatic Populated, non-modifiable), 2. EA (Automatic Populated, non-modifiable) 3. Station ID (Automatic Populated, non-modifiable) 4. Name of center (To be input by operator, modifiable) 5. Address of center, (To be input by operator, modifiable) 6. Operator name (Automatic Populated, non-modifiable) 7. Operator ID (Automatic Populated, non-modifiable) 8. Operator Mobile Number (Automatic Populated, non-modifiable) 9. Operator Aadhaar Number (Automatic Populated, non-modifiable) 10. Type of center (Dropdown values: Permanent /Camp-Mode, to be selected by operator, modifiable). Should also provide a ‘specify others’ text box to enter the center name if it is not found in the drop-down 11. Center ID (As of now blank, will be disabled as of now) 12. GPS details (Automatic Populated, non-modifiable, in few cases it will be blank) 13. Whitelisted for BE (True/False) 14. Operator will click Sync button and all the details will be stored at the sever end. 15. ‘OK’ button, to submit | |
|  |  | | | | * On click to OK button, a new Authentication window for operator’s OTP Verification. | |
| 2 | Operator will enter OTP and give his/her biometric on authentication window and click on OK button. | | | | * On successful response,   operator sync is successful | |
| **Alternate Flow: When Payment required flag is disabled** | | | | | | |
| **Step Number** | **User Action** | | | | **System Response** | |
|  |  | | | |  | |
| Post Conditions | | | Operator sync is successful | | | |
| Errors/Exception | | | Description | | | Error Code and Error Message |
| On operator sync window:-   1. Name of center is left blank 2. Address of center is left blank 3. If no value is selected from the ‘Type of center’ and ‘specify others’ text box is also left blank | | | 1. “Please enter Name of the center”. 2. ““Please enter Address of the center”. 3. Please enter name of the center |
| Business Rules | | | * A new configurable parameter will be added called as “**No. of days for operator sync**” * OTP will be send to only registered mobile number against Aadhaar number, no input of mobile number should be allowed for OTP validation by the operator. Mobile number and operator’s Aadhaar number will be a read only field. * If operator do not perform the operator sync, after the sync date no enrollments should be allowed from that ECMP machine. * Sync date will be a configurable parameter. * If operator do not perform the center and operators details sync, after the sync date no mobile update should be allowed from that client. * Operator Authentication and center/operators details sync screen will be same, both sync will be performed simultaneously and both have same one configurable parameter called as “No. of days for operator sync” * If name of the center is selected from a dropdown, the textbox ‘specify others’ should be freeze and not should allow to enter text * If name of center is selected as ‘others’ in the dropdown, then it is mandatory to fill the text box ‘specify others’ | | | |
| Entities Involved | | | Operator, UIDAI | | | |
| Assumptions | | | The operator should have registered his mobile number with his Aadhaar for OTP Auth. | | | |
| Issue List | | | Issues | Status | | |
|  |  | | |

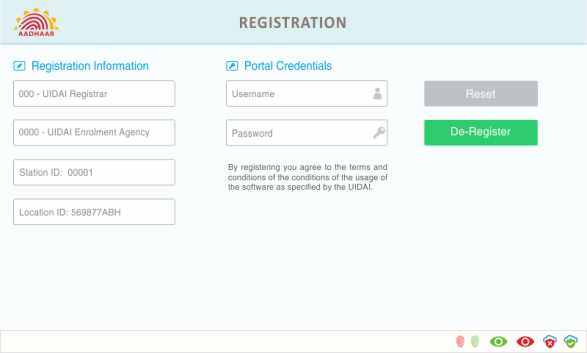
## Entities Involved

|  |  |  |
| --- | --- | --- |
| **Entity** | **Operation**  **(Create / Modify / Referred)** | **Internal / External** |
| Aadhaar data | Create | Internal |
| Aadhaar data of operator/ guardian | Referred |  |
|  |  |  |

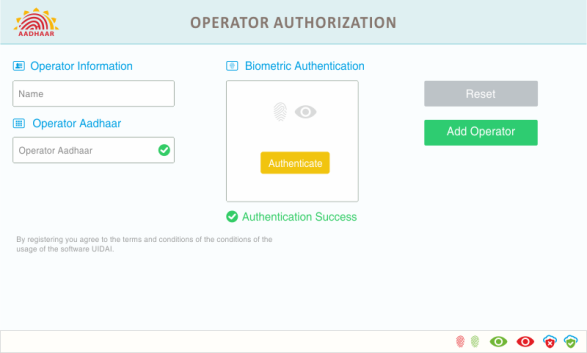
## Screenshots and Mock-up headings



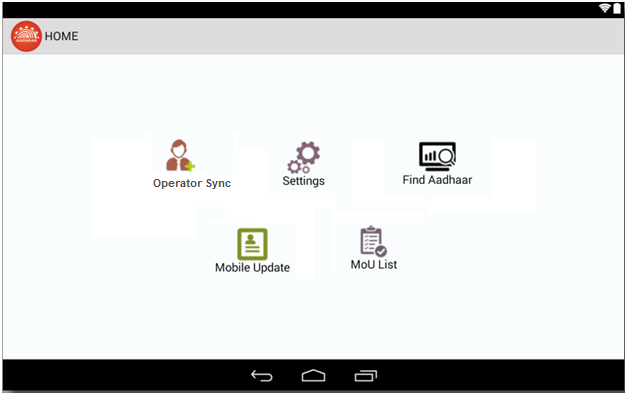
**Fig 1.0: Client registration**



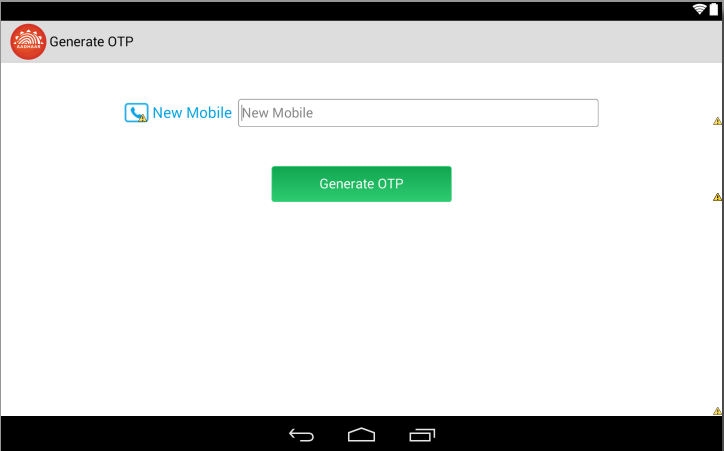
**Fig 2.0: Client de-registration**



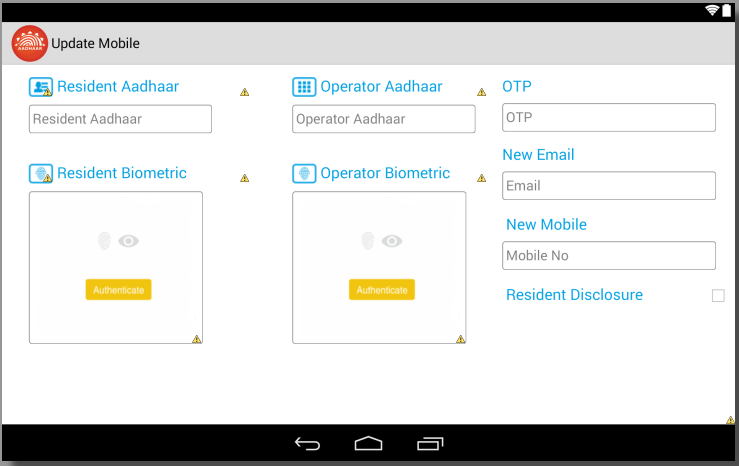
**Fig 3.0: Operator on-boarding**



**Fig 4.0: Mobile Update Link**

****

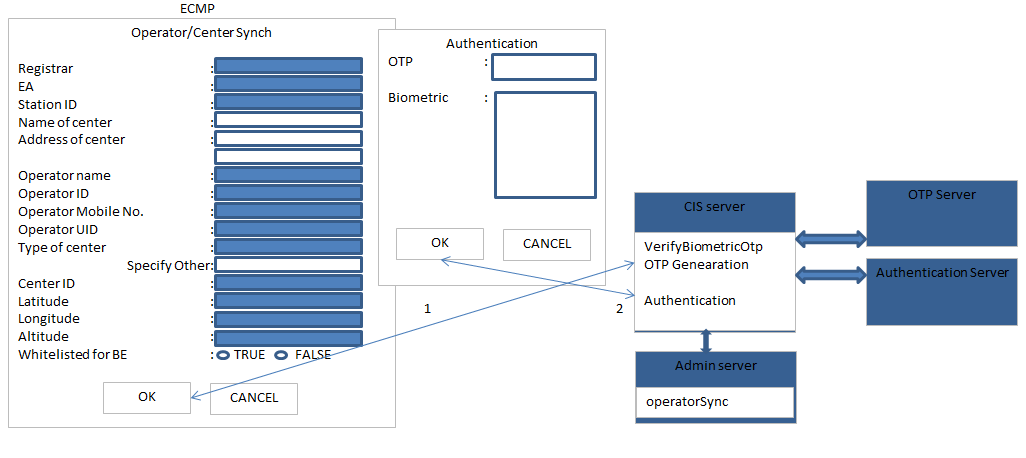
**Fig 5.0 Generate OTP Screen**

****

**Fig 6.0 Mobile Update Screen**

****

**Fig 7.0 Payment confirmation dialog Pop-Up**



**Fig 8.0 Operator Sync**

# Risk Section

# Hardware/Software Requirements

# Non Functional Requirement

### Performance

### Usability

### Security

* Necessary for the application/ database and identify the key parameters to be logged and audited.
* **Public/ Private Key Infrastructure**: If PKI will be used for secure information, protection of encryption/private key, secure distribution of the public key and appropriate encryption algorithm (RSA 2048, AES 256 & SHA 512) must be considered.
* Any Data at rest should be stored in Encrypted form.
* Mapping between Child Aadhaar Number and Parent’s Aadhaar number should be done in backend as to establish trust level and proper authentication.
* Application containing any stored passwords or sensitive data should be encrypted and stored.

### Backup/Recovery Policy

### Volumetric Requirements

# Application Impacts

NA

# Issues

N/A

# Revision History

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Version no.** | **Date** | **Author** | **Description/summary of changes** | **Reviewed by** | **Approved by (MSP)** | **Approved by (UIDAI)** |
| 0.1 | 27/04/2017 | Purav | Initial Draft | Neerav |  |  |
|  |  |  |  |  |  |  |