**Outreach Mobile Web App High Level Design**

Draft 0.5

Last Updated: 30-Dec-14

**Abstract**

This document describes the scope, high level design, database mappings and API specifications for the Outreach mobile web application.

Contents

2 Version History 4

3 References 4

4 Abbreviations and Acronyms 4

5 Assumptions, Issues and Risks 5

5.1 Assumptions 5

5.2 Issues 5

5.3 Risks 5

6 Introduction 6

6.1 Scope 6

6.2 Out of Scope 6

7 Solution Design 6

7.1 Architecture 6

7.1.1 Outreach Mobile UI 7

7.1.2 OAuth Provider 7

7.1.3 Service 7

7.1.4 Image Repository 7

7.1.5 Database 7

7.2 Security Model 7

7.2.1 OAuth 2.0 Implementation 7

7.3 Static Model 9

7.4 Database Design 10

7.5 Interface Specification and data mappings 11

7.5.1 Generic Response Type 11

7.5.2 Authentication and Authorization Service 11

7.5.3 Client Service 12

7.5.4 Enrollment Service 16

7.5.5 Disabilities Service 19

7.5.6 Date of Engagement Service 22

7.5.7 PATH Status Service 24

7.5.8 Referral Source Service 27

7.5.9 Services Service 29

7.5.10 LOV Service 32

7.6 Dynamic Design 33

7.6.1 E2E Flow 33

7.7 Deployment Design 33

8 Implementation Strategy 33

8.1 Technology Selection 33

9 Appendix A: LOV List 34

9.1 LOV Name to LOV ID mapping 34

9.2 LOV ID to LOV Value Mapping 35

# Version History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Description |
| Draft 0.1 | 18-Nov-2014 | Innoppl | Initial Draft |
| Draft 0.2 | 09-Dec-2014 | Innoppl | Updated Interface specification based on the database new database design. |
| Draft 0.3 | 18-Dec-2014 | Innoppl | Updated Client Service  Added LOV Service |
| Draft 0.4 | 26-Dec-2014 | Innoppl | Added Enrollment Service |

Table 1: Version History

# References

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.No. | Version | Source | Author | Document Name |
|  | v1.1 | Client Supplied Product | Saravanan  Innoppl | Outreach\_SRS\_v1.1.pdf  *(Software Requirements Specification Document)* |
|  | N/A | Client Supplied Product | Innoppl | ApprovedDesigns.zip  *(Mobile Phone Screen Mockups)* |
|  | v1 | Client Supplied Product | Saravanan  Innoppl | OutreachMobileAppAPISpecification.docx  *(Initial draft API Specification)* |
|  | N/A | Client Supplied Product | Innoppl | Database\_conceptual\_diagram.jpeg  *(Database )* |
|  | N/A | Client Supplied Product | Innoppl | erd.png |
|  | V4.0 | www.hudhdx.info | N/A | HMIS\_Logical\_Model.pdf |

Table 2: References

# Abbreviations and Acronyms

|  |  |
| --- | --- |
| Abbreviation/Acronym | Description |
| UI | User Interface |
| N/A | Not Available |
| E2E | End to End |
| HTML | HyperText Markup Language |
| HTTP | HyperText Transfer Protocol |
| SSL | Secure Socket Layer |
| ERD | Entity Relationship Diagram |
| HUD | U.S. Department of Housing and Urban Development |
| HMIS | Homeless Management Information Systems |
| SRS | Software Requirements Specification |
| API | Application Program Interface |
| AJAX | Asynchronous Javascript for XML |
| LOV | List of Values |

Table 3: Abbreviations and Acronyms

# Assumptions, Issues and Risks

## Assumptions

|  |  |  |  |
| --- | --- | --- | --- |
| S.No. | Assumption | Comments | Status |
|  | The layout specified in ApprovedDesigns.zip [Ref.2] will be used for both phone and tablet resolutions. Desktop layout is not in the current scope. | Based on discussion held with Saravanan on 14-Nov-2015, it is agreed that the layout provided by Innoppl will be used for both phone and tablet resolution. Desktop layout is not in our current scope. | Closed |
|  |  |  |  |
|  |  |  |  |

Table 4: Assumptions

## Issues

|  |  |  |  |
| --- | --- | --- | --- |
| S.No. | Assumption | Comments | Status |
|  |  |  |  |
|  |  |  |  |

Table 5: Issues

## Risks

|  |  |  |  |
| --- | --- | --- | --- |
| S.No. | Assumption | Comments | Status |
|  |  |  |  |
|  |  |  |  |

Table 6: Risks

# Introduction

Outreach Mobile is a mobile web application primarily focused in

## Scope

* All use cases specified in Outreach\_SRS\_v1.1.pdfdocument.
* The layout design is based on the approved design mockup. The same layout will be used for phone and tablet.
* Outreach Mobile Web application Web Services design, implementation and deployment.
* Outreach UI design, Implementation and deployment.

## Out of Scope

* Layout for desktop is not in the current scope of work.
* Database Design

# Solution Design

## Architecture

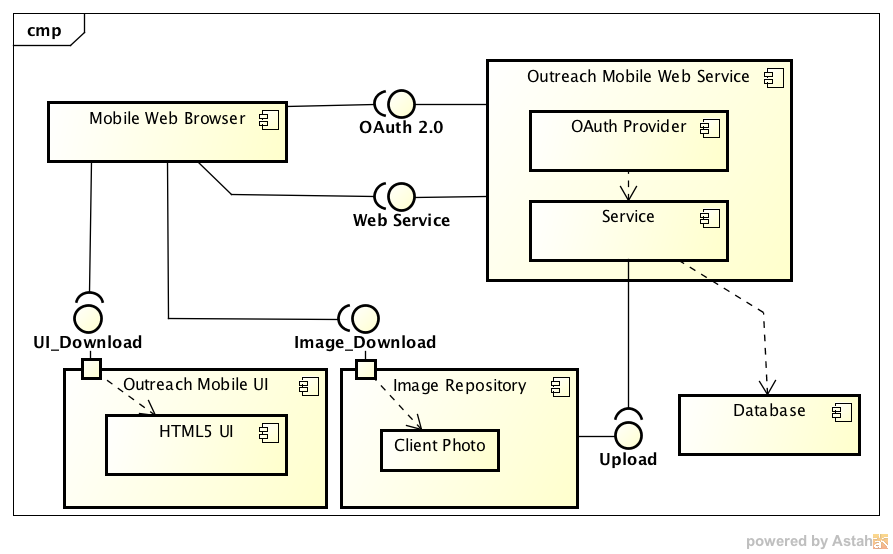


Figure 1: Outreach Mobile Web UML Component Diagram

The above figure (Figure 1) provides a high level detail of components in the Outreach mobile web application.

### Outreach Mobile UI

Web container to publish HTML screens.

### OAuth Provider

OAuth 2.0 provider.

### Service

Outreach Mobile Web Service container

### Image Repository

Web server to publish and retrieve client images.

### Database

MySQL Cloud database.

## Security Model

The authentication and authorization is handled using the [[1]](#footnote-2)OAuth 2.0 mechanism. OAuth 2.0 is an open authorization protocol specification defined by IETF OAuth WG (Working Group) which enables applications to access each other’s data.

Oauth 2.0 is selected because of the following fundamental considerations:

* The entire application is developed as RESTful webservices and accessed by the Outreach HTML 5 based UI via Ajax.Since it is stateless in nature, the mechanisms of securing these services are different from standard web application where it is easily handled by session management, but in the case of REST, no session can be maintained as the calling point may or may not be a web browser.
* The RESTful services will be accessed by 3rd party applications.
* The application can be accessed from native/hybrid mobile applications.

### OAuth 2.0 Implementation

Since the core functionalities are implemented as RESTful web services and accessed using Ajax from HTML 5 web pages, we will be using the OAuth 2.0 generic Implicit Grant mechanism. The following Figure 2gives a high-level overview of the OAuth 2.0 implicit grant mechanism. Since the access token is sent as part of the HTTP response, it is visible over the wire in Internet and so the transport has to be secured using SSL.

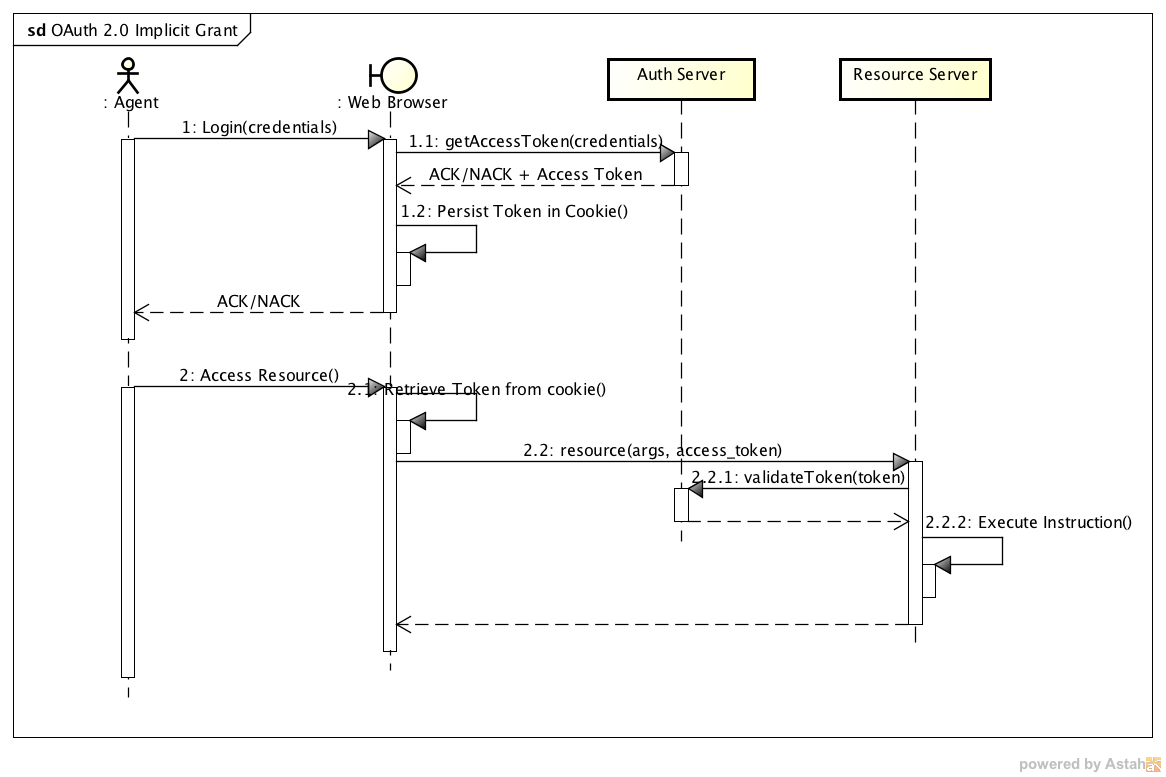


Figure 2: OAuth 2.0 Implicit Grant UML Sequence Diagram

## Static Model

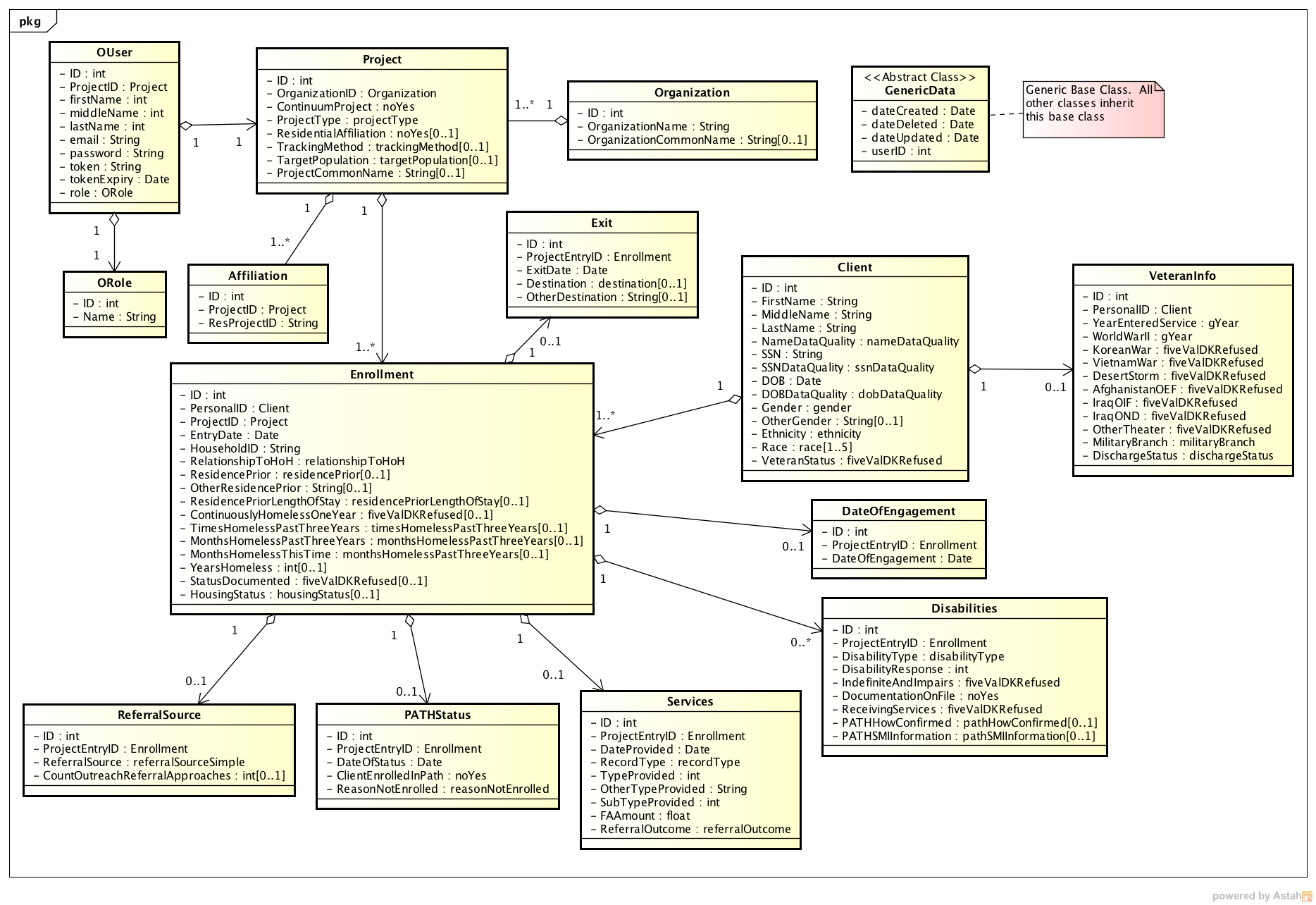
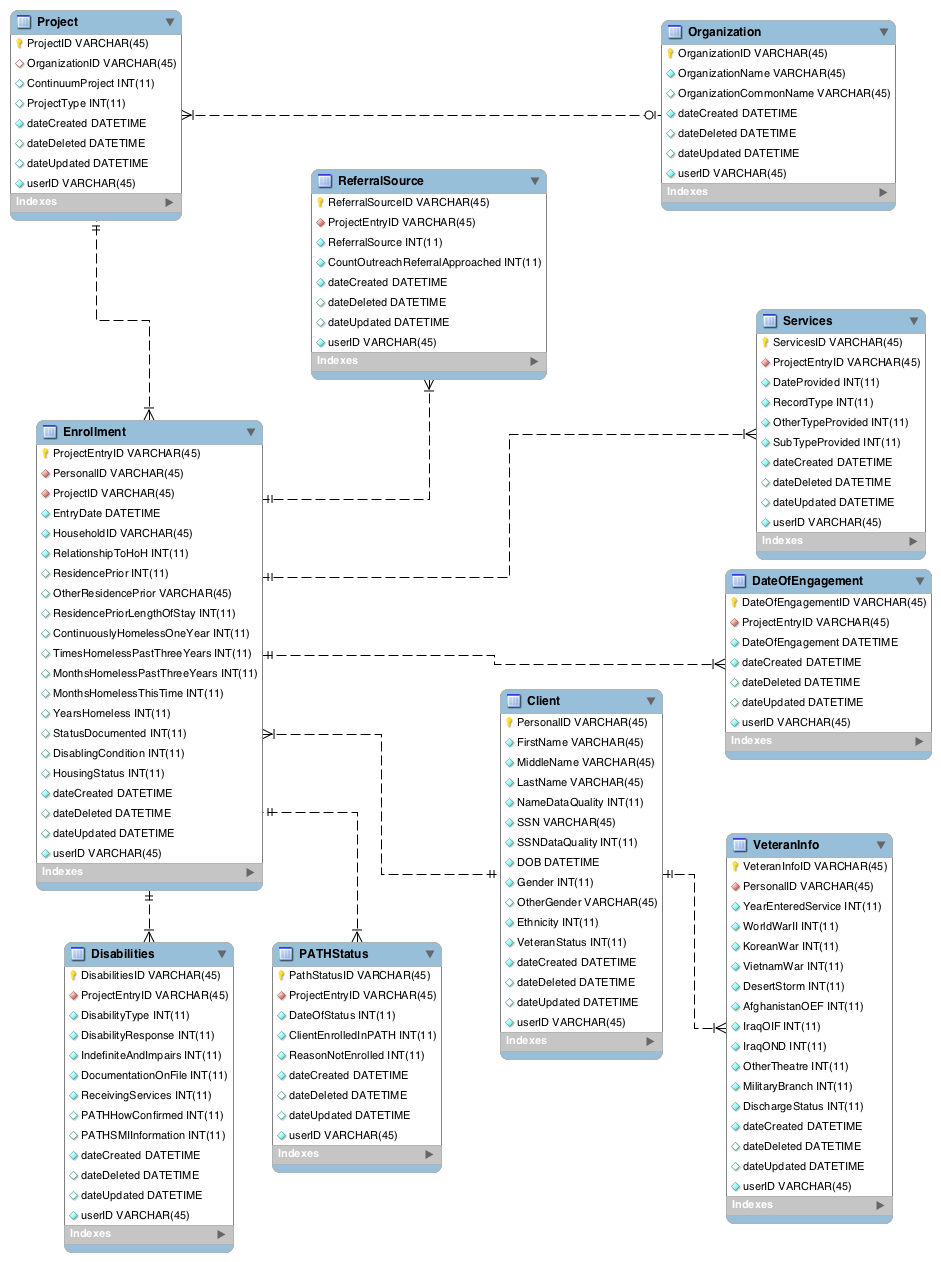


Figure 3: Outreach Data Model - UML Class Diagram

## Database Design

Ref.5



## Interface Specification and data mappings

### Generic Response Type

All response to the APIs in the subsequent sections will have the following structure for the web service response.

{

"header": {

"code": 0,

"message": "",

"total\_count": 234

},

"body": [

Array of Requested Data Structure

]

}

The code field identifies whether the response is success or failure. If the value of code is 0 then it is a success else it is a failure. The following is a sample failure response. Only header is sent for failure scenarios.

{

"header": {

"code": 1001,

"message": "No records found",

}

}

|  |  |  |
| --- | --- | --- |
| Field | Description | Mandatory(M)/Optional(O) |
| header | A generic response header | M |
| code | Response Code  0 – Success  >0 – Failure | M |
| message | Description of failure/error. | O |

Table 7: Generic Response Header

### Authentication and Authorization Service

All web services are authenticated and authorized using the OAuth 2.0 implicit grant flow. Please refer section 7.2 for the OAuth 2.0 implicit grant flow.

#### API to retrieveAccess Token

Use the following web service to get an access token.

|  |  |
| --- | --- |
|  |  |
| END POINT | /service/user/auth |
| HTTP METHOD | POST |
| TRANSPORT | SSL |
| AUTHENTICATION |  |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | {  "email":"a@b.com",  "password":"pa55w0rd"  } |
| SUCCESS RESPONSE | HTTP 200 OK  {     "access\_token": "94134def-1775-4d85-9b5d-a856e51",     "expires\_in": 1419939427717,     "token\_type": "Bearer"  } |
| FAILURE RESPONSE | HTTP 401Unauthorized |

### Client Service

#### Client DataStructure

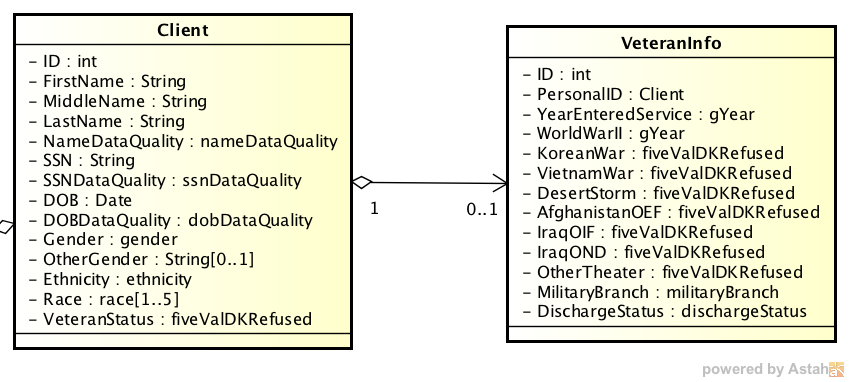


Figure 4: Client Data Structure

The following sample JSON represents a Client object (CLIENT\_OBJECT).

{

"ID": 101,

"FirstName": "Edward",

"MiddleName": "Barnaby",

"LastName": "Jones",

"NameSuffix": "Sr.",

"NameDataQuality": 9,

"SSN": "7xxx9xx8x",

"SSNDataQuality": 2,

"DOB": "2014-11-18",

"DOBDataQuality": 1,

"Gender": 3,

"OtherGender": "intersex",

"Ethnicity": 0,

"Race": 8,

"VeteranStatus": 9,

"VeteranInfo": {

"ID": 787,

"YearEnteredService": "2002",

"YearSeparated": "2004",

"WorldWarII": "0",

"KoreanWar": "99",

"VietnamWar": "1",

"DesertStorm": "0",

"AfghanistanOEF": "8",

"IraqOIF": "9",

"IraqOND": "99",

"OtherTheater": "8",

"MilitaryBranch": "8",

"DischargeStatus": "6"

}

}

#### WSAPI\_01: Add New Client

|  |  |
| --- | --- |
|  |  |
| END POINT | /service/client |
| HTTP METHOD | POST |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | CLIENT\_OBJECT [Ref: 7.5.3.1]  The following fields must**NOT** be specified  ID of Client  ID of VeteranInfo if applicable |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <CLIENT\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to add"  }  } |

#### WSAPI\_02: Update Existing Client Information

|  |  |
| --- | --- |
|  |  |
| END POINT | /service/client |
| HTTP METHOD | PUT |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | CLIENT\_OBJECT [Ref: 7.5.3.1]  The following fields must be specified  PersonalID  ProjectEntryID  VeteranInfoID if applicable |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <CLIENT\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to update"  }  } |

#### WSAPI\_03: Lookup Client by PersonalID

|  |  |
| --- | --- |
|  |  |
| END POINT | /service/client/<PersonalID> |
| HTTP METHOD | GET |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <CLIENT\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Not Found"  }  } |

#### WSAPI\_04: Search Client by Search Attributes

|  |  |
| --- | --- |
|  |  |
| END POINT | /service/client/search |
| HTTP METHOD | GET |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| Query Params | {  "firstName": "john",  "lastName": "smith",  "ssn": "123",  "sAge": 30,  "eAge": 50,  "start": 1,  "rows": 10  }  firstName - First Name  lastName - Last Name  ssn - SSN  sAge - Start Age  eAge - End Age  start - Start Index  rows – No. of rows to return. Maximum value can be 10 |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <CLIENT\_OBJECT>  },  {  <CLIENT\_OBJECT>  },  {  <CLIENT\_OBJECT>  },  ...  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Not Found"  }  } |

### Enrollment Service

#### Enrollment Data Structure

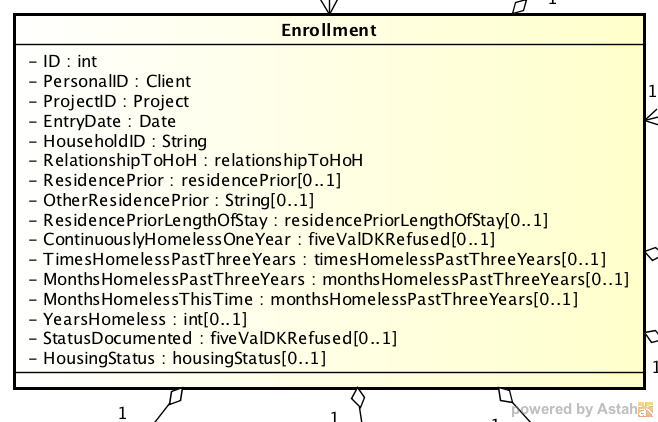


Figure 5: Enrollment Data Structure

The following sample JSON represents a Client object (CLIENT\_OBJECT).

{

"ID": 5642,

"PersonalID": {"ID": 563},

"EntryDate": "2016-04-12Z",

"HouseholdID": "a34563w",

"RelationshipToHoH": "1",

"ResidencePrior": "12",

"OtherResidencePrior": "some other residence text here",

"ResidencePriorLengthOfStay": "8",

"ContinuouslyHomelessOneYear": "8",

"TimesHomelessPastThreeYears": "0",

"MonthsHomelessPastThreeYears": "103",

"MonthsHomelessThisTime": "101",

"YearsHomeless": "3",

"StatusDocumented": "1",

"DisablingCondition": "9",

"HousingStatus": "6"

}

#### WSAPI\_01a: Add New Enrollment for a Client

|  |  |
| --- | --- |
|  |  |
| END POINT | /service/enrollment |
| HTTP METHOD | POST |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | ENROLLMENT\_OBJECT [Ref: 7.5.3.1]  The following fields must **NOT** be specified  ID |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <ENROLLMENT\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to add"  }  } |

#### WSAPI\_01b: Update Existing Enrollment Information

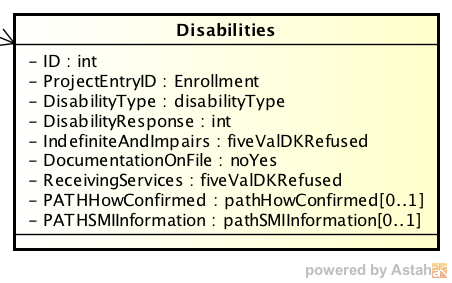
|  |  |
| --- | --- |
|  |  |
| END POINT | /service/enrollment |
| HTTP METHOD | PUT |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | CLIENT\_OBJECT [Ref: 7.5.3.1]  The following fields must be specified  ID |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <ENROLLMENT\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to update"  }  } |

#### WSAPI\_01c: Get all enrollments for PersonalID

|  |  |
| --- | --- |
|  | |
| END POINT | /service/enrollment/{PersonalID} |
| HTTP METHOD | GET |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | CLIENT\_OBJECT [Ref: 7.5.3.1]  The following fields must be specified  ID |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <ENROLLMENT\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to update"  }  } |

### Disabilities Service

#### Disabilities Data Structure



The following sample JSON represents a Disabilties object (DISABILITIES\_OBJECT).

{

"ID": 674,

"ProjectEntryID": {"ID": 343},

"DisabilityType": "10",

"DisabilityResponse": "2",

"IndefiniteAndImpairs": "9",

"DocumentationOnFile": "1",

"ReceivingServices": "99",

"PATHHowConfirmed": "1",

"PATHSMIInformation": "3"

}

#### WSAPI\_05: Add New Disability to existing client and enrollment

|  |  |
| --- | --- |
|  |  |
| END POINT | /disability |
| HTTP METHOD | POST |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | Array of DISABILITIES\_OBJECT [Ref: 7.5.5.1]  [  {  <DISABILITIES\_OBJECT>  },  {  <DISABILITIES\_OBJECT>  },  {  <DISABILITIES\_OBJECT>  },  ...  ]  The following fields must **NOT** be specified  ID  ProjectEntryID |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <DISABILITIES\_OBJECT>  },  {  <DISABILITIES\_OBJECT>  },  {  <DISABILITIES\_OBJECT>  },  ...  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to add"  }  } |

#### WSAPI\_06: Update Existing Disability Information

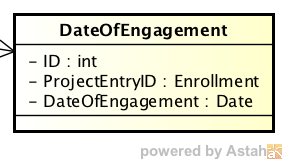
|  |  |
| --- | --- |
|  |  |
| END POINT | /disability |
| HTTP METHOD | PUT |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | Array of DISABILITIES\_OBJECT [Ref: 7.5.5.1]  [  {  <DISABILITIES\_OBJECT>  },  {  <DISABILITIES\_OBJECT>  },  {  <DISABILITIES\_OBJECT>  },  ...  ]  The following fields must be specified  ID  ProjectEntryID |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <DISABILITIES\_OBJECT>  },  {  <DISABILITIES\_OBJECT>  },  {  <DISABILITIES\_OBJECT>  },  ...  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to update"  }  } |

#### WSAPI\_07: Lookup Disabilities by DisabilitiesID

|  |  |
| --- | --- |
|  |  |
| END POINT | /disability/<DisabilitiesID> |
| HTTP METHOD | GET |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <DISABILITIES\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Not Found"  }  } |

### Date of Engagement Service

#### Date of Engagement Data Structure



The following sample JSON represents a Date of Engagement object (DOE\_OBJECT).

{

"ID": 4321,

"ProjectEntryID": {"ID": 343},

"DateOfEngagement": "2016-04-17"

}

#### WSAPI\_08: Add New Date of Engagement to existing client and enrollment

|  |  |
| --- | --- |
|  |  |
| END POINT | /doe |
| HTTP METHOD | POST |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | DOE\_OBJECT [Ref:7.5.6.1]  {  <DOE\_OBJECT>  }  The following fields must **NOT** be specified  ID |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  < DOE\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to add"  }  } |

#### WSAPI\_09: Update Existing Date of Engagement Information

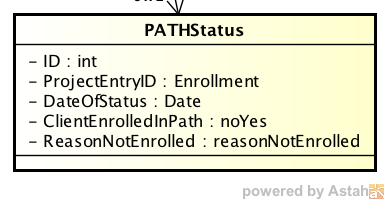
|  |  |
| --- | --- |
|  |  |
| END POINT | /doe |
| HTTP METHOD | PUT |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | DOE\_OBJECT [Ref: 7.5.6.1]  {  <DOE\_OBJECT>  }  The following fields must be specified  ID |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  < DOE \_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to update"  }  } |

#### WSAPI\_10: Lookup Date of Engagement by DateOfEngagementID

|  |  |
| --- | --- |
|  |  |
| END POINT | /doe/<DateOfEngagementID> |
| HTTP METHOD | GET |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <DOE\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Not Found"  }  } |

### PATH Status Service

#### PATH Status Data Structure



The following sample JSON represents a PATH Status object (PATH\_OBJECT).

{

"ID": 453332,

"ProjectEntryID": {"ID": 343},

"DateOfStatus": "2016-04-23",

"ClientEnrolledInPATH": "0",

"ReasonNotEnrolled": "2"

}

#### WSAPI\_11: Add New PATH Status to existing client and enrollment

|  |  |
| --- | --- |
|  |  |
| END POINT | /pathstatus |
| HTTP METHOD | POST |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | DOE\_OBJECT [Ref:7.5.7.1]  {  <PATH\_OBJECT>  }  The following fields must **NOT** be specified  ID |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <PATH\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to add"  }  } |

#### WSAPI\_12: Update Existing PATH Status Information

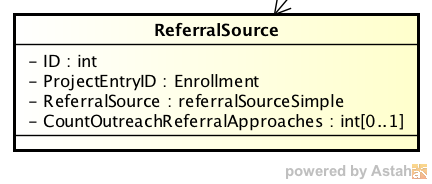
|  |  |
| --- | --- |
|  |  |
| END POINT | /pathstatus |
| HTTP METHOD | PUT |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | DOE\_OBJECT [Ref: 7.5.7.1]  {  <PATH\_OBJECT>  }  The following fields must be specified  ID |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <PATH\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to update"  }  } |

#### WSAPI\_13: Lookup PATH Status by PathStatusID

|  |  |
| --- | --- |
|  |  |
| END POINT | /pathstatus/<PathStatusID> |
| HTTP METHOD | GET |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <PATH\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Not Found"  }  } |

### Referral Source Service

#### Referral Source Data Structure



The following sample JSON represents a Referral Source object (REF\_SRC\_OBJECT).

{

"ID": 34343,

"ProjectEntryID": {"ID": 343},

"ReferralSource": "18",

"CountOutreachReferralApproaches": "3"

}

#### WSAPI\_14: Add New ReferralSource to existing client and enrollment

|  |  |
| --- | --- |
|  |  |
| END POINT | /referralsrc |
| HTTP METHOD | POST |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | REF\_SRC\_OBJECT [Ref:7.5.8.1]  {  <REF\_SRC\_OBJECT>  }  The following fields must **NOT** be specified  ID |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <REF\_SRC\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to add"  }  } |

#### WSAPI\_15: Update Existing Referral Source Information

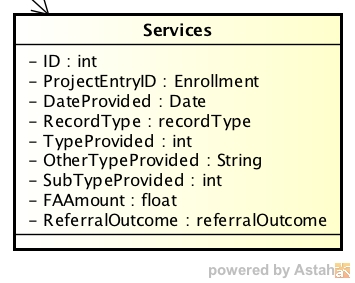
|  |  |
| --- | --- |
|  |  |
| END POINT | /referralsrc |
| HTTP METHOD | PUT |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | REF\_SRC\_OBJECT [Ref: 7.5.8.1]  {  <REF\_SRC\_OBJECT>  }  The following fields must be specified  ReferralSourceID |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <REF\_SRC\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to update"  }  } |

#### WSAPI\_16: Lookup Referral Source by ReferralSourceID

|  |  |
| --- | --- |
|  |  |
| END POINT | /referralsrc/<ReferralSourceID> |
| HTTP METHOD | GET |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <REF\_SRC\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Not Found"  }  } |

### Services Service

#### Services Data Structure



The following sample JSON represents a Services object (SVCS\_OBJECT).

{

"ID": "44gg",

"ProjectEntryID": {"ID": "343"},

"DateProvided": "2015-09-06",

"RecordType": "16",

"TypeProvided": "11",

"OtherTypeProvided": "some other type here",

"SubTypeProvided": "12",

"FAAmount": "1145.59",

"ReferralOutcome": "2"

}

#### WSAPI\_17: Add New Servicesto existing client and enrollment

|  |  |
| --- | --- |
|  |  |
| END POINT | /services |
| HTTP METHOD | POST |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | SVCS\_OBJECT [Ref:7.5.9.1]  {  <SVCS\_OBJECT>  }  The following fields must **NOT** be specified  ServicesID |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <SVCS\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to add"  }  } |

#### WSAPI\_18: Update Existing Services Information

|  |  |
| --- | --- |
|  |  |
| END POINT | /services |
| HTTP METHOD | PUT |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| REQUEST JSON | SVCS\_OBJECT [Ref: 7.5.9.1]  {  <SVCS\_OBJECT>  }  The following fields must be specified  ServicesID |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <SVCS\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Failed to update"  }  } |

#### WSAPI\_19: Lookup Services by ServicesID

|  |  |
| --- | --- |
|  |  |
| END POINT | /services/<ServicesID> |
| HTTP METHOD | GET |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <SVCS\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Not Found"  }  } |

### LOV Service

#### LOV Data Structure

The following sample JSON represents a Disabilties object (LOV\_OBJECT).

{

"id": 26,

"name": "No",

"value": "0"

}

#### WSAPI\_20: Get LOV by Type

|  |  |
| --- | --- |
|  |  |
| END POINT | /lov/<LOV NAME>  Refer Table 9in Section 9 for LOV Names  Eg: /lov/fiveValDKRefused |
| HTTP METHOD | GET |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <LOV\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Not Found"  }  } |

#### WSAPI\_21: Reverse Lookup LOV by Type and Value

|  |  |
| --- | --- |
|  |  |
| END POINT | /lov/<LOV NAME>/<LOV Value>  Refer Table 9 in Section 9 for LOV Names  Eg: /lov/fiveValDKRefused/0 |
| HTTP METHOD | GET |
| TRANSPORT | SSL |
| AUTHENTICATION | HTTP Header  Authorization: Bearer <BASE64 Encoded Access Token>  Refer section 7.5.2.1 to get the access token |
| HTTP HEADERS | Content-Type:application/json  Accept:application/json |
| SUCCESS RESPONSE | HTTP 200 OK  {  "header": {  "code": 0  },  "body": [  {  <LOV\_OBJECT>  }  ]  } |
| FAILURE RESPONSE | HTTP 200 OK  {  "header": {  "code": 1001,  "message": "Not Found"  }  } |

## Dynamic Design

### E2E Flow

## Deployment Design

# Implementation Strategy

## Technology Selection

All tools and technologies that we use are either Open Source or available for free.

| Business Functionality | Technology | Reason |
| --- | --- | --- |
| Solution Platform | Java7 / JEE6 | The recommendation solution involves a highly scalable architecture with concurrency support. This is feasible with Java. The other alternatives are PHP & .Net.  PHP does not support concurrency. And we will end up with scalability issues when trying to update huge number of book records.  Microsoft solutions are expensive to implement and scale. |
| Web Services | Spring MVC Framework | Spring is the de-facto standard for implementing scalable web services and it considerably reduces time.  Easy to test |
| OAuth Provider | Spring OAuth Provider | Since we are going to use Spring Framework, Spring OAuth Provider is the default choice as it is easy to integrate and supports OAuth 2.0 specification. |
| Automated Testing | Spring Testing Framework | Easy to test by deploying to web servers automatically. |
| Build Automation | Maven | Maven is mature and is most widely used build automation tool. |
| Database Access | Hibernate JPA | Database agnostic, scalable and high performance ORM engine. Highly compatible with Spring Framework. |
| Database | MySQL |  |
| JEE Web Server | Jetty | Simple, Lightweight and scalable web server. |
| Java Version | JDK 7.0 | Latest stable version of Java |
| JEE Version | JEE 6 | Latest stable version of Java Enterprise Edition |

Table 8: Recommendation Service Technology Selection

# Appendix A: LOV List

## LOV Name to LOV ID mapping

| LOV Name | LOV ID |
| --- | --- |
| addressDataQuality | 1 |
| disabilityType | 2 |
| dobDataQuality | 3 |
| employmentType | 4 |
| ethnicity | 5 |
| fiveValDKRefused | 6 |
| fysbReasonNoServices | 7 |
| gender | 8 |
| householdType | 9 |
| housingStatus | 10 |
| militaryBranch | 11 |
| monthsHomelessPastThreeYears | 12 |
| nameDataQuality | 13 |
| noYes | 14 |
| noYesRefused | 15 |
| pathHowConfirmed | 16 |
| pathSMIInformation | 17 |
| projectType | 18 |
| race | 19 |
| reasonNotEnrolled | 20 |
| recordType | 29 |
| referralOutcome | 21 |
| referralSourceSimple | 22 |
| relationshipToHoH | 23 |
| residencePrior | 24 |
| residencePriorLengthOfStay | 25 |
| ssnDataQuality | 26 |
| State | 27 |
| timesHomelessPastThreeYears | 28 |

Table 9: LOV Name to LOV ID mapping

## LOV ID to LOV Value Mapping

| LOV ID | Value | Display Name |
| --- | --- | --- |
| 1 | 1 | Full address reported |
| 1 | 2 | Incomplete or estimated address reported |
| 1 | 8 | Client doesn't know |
| 1 | 9 | Client refused |
| 1 | 99 | Data not collected |
| 2 | 5 | Physical Disability |
| 2 | 6 | Developmental Disability |
| 2 | 7 | Chronic Health Condition |
| 2 | 8 | HIV/AIDS |
| 2 | 9 | Mental Health Problem |
| 2 | 10 | Substance Abuse |
| 3 | 1 | Full DOB reported |
| 3 | 2 | Approximate or partial DOB reported |
| 3 | 8 | Client doesn’t know |
| 3 | 9 | Client refused |
| 3 | 99 | Data not collected |
| 4 | 1 | Full-timed |
| 4 | 2 | Part-time |
| 4 | 3 | Seasonal / sporadic (including day labor) |
| 4 | 99 | Data not collected |
| 5 | 0 | Non-Hispanic/Non-Latino |
| 5 | 1 | Hispanic/Latino |
| 5 | 8 | Client doesn’t know |
| 5 | 9 | Client refused |
| 5 | 99 | Data not collected |
| 6 | 0 | No |
| 6 | 1 | Yes |
| 6 | 8 | Client doesn't know |
| 6 | 9 | Client refused |
| 6 | 99 | Data not collected |
| 7 | 1 | Out of age range |
| 7 | 2 | Ward of the State – Immediate Reunification |
| 7 | 3 | Ward of the Criminal Justice System – Immediate Reunification |
| 7 | 4 | Other |
| 7 | 99 | Data not collected |
| 8 | 1 | Male |
| 8 | 2 | Transgender male to female |
| 8 | 3 | Transgender female to male |
| 8 | 4 | Other |
| 8 | 8 | Client doesn't know |
| 8 | 9 | Client refused |
| 8 | 99 | Data not collected |
| 9 | 1 | Households without children |
| 9 | 3 | Households with at least one adult and one child |
| 9 | 4 | Households with only children |
| 10 | 1 | Category 1 - Homeless |
| 10 | 2 | Category 2 - At imminent risk of losing housing |
| 10 | 3 | At-risk of homelessness |
| 10 | 4 | Stably Housed |
| 10 | 5 | Category 3 - Homeless only under other federal statutes |
| 10 | 6 | Category `4 - Fleeing domestic violence |
| 10 | 8 | Client doesn’t know |
| 10 | 9 | Client refused |
| 10 | 99 | Data not collected |
| 11 | 1 | Army |
| 11 | 2 | Air Force |
| 11 | 3 | Navy |
| 11 | 4 | Marines |
| 11 | 6 | Coast Guard |
| 11 | 7 | Other |
| 11 | 8 | Client doesn’t know |
| 11 | 9 | Client refused |
| 11 | 99 | Data not collected |
| 12 | 100 | 0 months |
| 12 | 101 | 1 month |
| 12 | 102 | 2 months |
| 12 | 103 | 3 months |
| 12 | 104 | 4 months |
| 12 | 105 | 5 months |
| 12 | 106 | 6 months |
| 12 | 107 | 7 months |
| 12 | 108 | 8 months |
| 12 | 109 | 9 months |
| 12 | 110 | 10 months |
| 12 | 111 | 11 months |
| 12 | 112 | 12 months |
| 12 | 7 | Other |
| 12 | 8 | Client doesn’t know |
| 12 | 9 | Client refused |
| 12 | 99 | Data not collected |
| 13 | 1 | Full name reported |
| 13 | 2 | Partial, street name, or code name reported |
| 13 | 8 | Client doesn’t know |
| 13 | 9 | Client refused |
| 13 | 99 | Data not collected |
| 14 | 0 | No |
| 14 | 1 | Yes |
| 14 | 99 | Data not collected |
| 15 | 0 | No |
| 15 | 1 | Yes |
| 15 | 9 | Client refused |
| 15 | 99 | Data not collected |
| 16 | 1 | Unconfirmed; presumptive or self-report |
| 16 | 2 | Confirmed through assessment and clinical evaluation |
| 16 | 3 | Confirmed by prior evaluation or clinical records |
| 16 | 99 | Data not collected |
| 17 | 0 | No |
| 17 | 1 | Unconfirmed; presumptive or self-report |
| 17 | 2 | Confirmed through assessment and clinical evaluation |
| 17 | 3 | Confirmed by prior evaluation or clinical records |
| 17 | 8 | Client doesn’t know |
| 17 | 9 | Client refused |
| 17 | 99 | Data not collected |
| 18 | 1 | Emergency Shelter |
| 18 | 2 | Transitional Housing |
| 18 | 3 | PH - Permanent Supportive Housing (disability required for entry) |
| 18 | 4 | Street Outreach |
| 18 | 6 | Services Only |
| 18 | 7 | Other |
| 18 | 8 | Safe Haven |
| 18 | 9 | PH – Housing Only |
| 18 | 10 | PH – Housing with Services (no disability required for entry) |
| 18 | 11 | Day Shelter |
| 18 | 12 | Homelessness Prevention |
| 18 | 13 | PH - Rapid Re-Housing |
| 18 | 14 | Coordinated Assessment |
| 19 | 1 | American Indian or Alaska Native |
| 19 | 2 | Asian |
| 19 | 3 | Black or African American |
| 19 | 4 | Native Hawaiian or Other Pacific Islander |
| 19 | 5 | White |
| 19 | 8 | Client doesn’t know |
| 19 | 9 | Client refused |
| 19 | 99 | Data not collected |
| 20 | 1 | Client was found ineligible for PATH |
| 20 | 2 | Client was not enrolled for other reason(s) |
| 20 | 99 | Data not collected |
| 21 | 1 | Attained |
| 21 | 2 | Not attained |
| 21 | 3 | Unknown |
| 22 | 1 | Self-Referral |
| 22 | 2 | Individual: Parent/Guardian |
| 22 | 3 | Individual: Relative or Friend |
| 22 | 4 | Individual: Other Adult or Youth |
| 22 | 5 | Individual: Partner/Spouse |
| 22 | 6 | Individual: Foster Parent |
| 22 | 7 | Outreach Project: FYSB |
| 22 | 10 | Outreach Project: Other |
| 22 | 11 | Temporary Shelter: FYSB Basic Center Project |
| 22 | 12 | Temporary Shelter: Other Youth Only Emergency Shelter |
| 22 | 13 | Temporary Shelter: Emergency Shelter for Families |
| 22 | 14 | Temporary Shelter: Emergency Shelter for Individuals |
| 22 | 15 | Temporary Shelter: Domestic Violence Shelter |
| 22 | 16 | Temporary Shelter: Safe Place |
| 22 | 17 | Temporary Shelter: Other |
| 22 | 18 | Residential Project: FYSB Transitional Living Project |
| 22 | 19 | Residential Project: Other Transitional Living Project |
| 22 | 20 | Residential Project: Group Home |
| 22 | 21 | Residential Project: Independent Living Project |
| 22 | 22 | Residential Project: Job Corps |
| 22 | 23 | Residential Project: Drug Treatment Center |
| 22 | 24 | Residential Project: Treatment Center |
| 22 | 25 | Residential Project: Educational Institute |
| 22 | 26 | Residential Project: Other Agency project |
| 22 | 27 | Residential Project: Other Project |
| 22 | 28 | Hotline: National Runaway Switchboard |
| 22 | 29 | Hotline: Other |
| 22 | 30 | Other Agency: Child Welfare/CPS |
| 22 | 31 | Other Agency: Non-Residential Independent Living Project |
| 22 | 32 | Other Project Operated by your Agency |
| 22 | 33 | Other Youth Services Agency |
| 22 | 34 | Juvenile Justice |
| 22 | 35 | Law Enforcement/ Police |
| 22 | 36 | Religious Organization |
| 22 | 37 | Mental Hospital |
| 22 | 38 | School |
| 22 | 39 | Other Organization |
| 22 | 8 | Client doesn’t know |
| 22 | 9 | Client refused |
| 22 | 99 | Data not collected |
| 23 | 1 | Self (head of household) |
| 23 | 2 | Head of household’s child |
| 23 | 3 | Head of household’s spouse or partner |
| 23 | 4 | Head of household’s other relation member (other relation to head of household) |
| 23 | 5 | Other: non-relation member |
| 24 | 1 | Emergency shelter, including hotel or motel paid for with emergency shelter voucher |
| 24 | 2 | Transitional housing for homeless persons (including homeless youth) |
| 24 | 3 | Permanent housing for formerly homeless persons (such as: CoC project; HUD legacy programs; or HOPWA PH) |
| 24 | 4 | Psychiatric hospital or other psychiatric facility |
| 24 | 5 | Substance abuse treatment facility or detox center |
| 24 | 6 | Hospital or other residential non-psychiatric medical facility |
| 24 | 7 | Jail, prison or juvenile detention facility |
| 24 | 8 | Client doesn’t know |
| 24 | 9 | Client refused |
| 24 | 12 | Staying or living in a family member’s room, apartment or house |
| 24 | 13 | Staying or living in a friend’s room, apartment or house |
| 24 | 14 | Hotel or motel paid for without emergency shelter voucher |
| 24 | 15 | Foster care home or foster care group home |
| 24 | 16 | Place not meant for habitation (e.g., a vehicle, an abandoned building, bus/train/subway station/airport or anywhere outside) |
| 24 | 17 | Other |
| 24 | 18 | Safe Haven |
| 24 | 19 | Rental by client, with VASH subsidy |
| 24 | 20 | Rental by client, with other ongoing housing subsidy |
| 24 | 21 | Owned by client, with ongoing housing subsidy |
| 24 | 22 | Rental by client, no ongoing housing subsidy |
| 24 | 23 | Owned by client, no ongoing housing subsidy |
| 24 | 24 | Long-term care facility or nursing home |
| 24 | 25 | Rental by client, with GPD TIP subsidy |
| 24 | 26 | Residential project or halfway house with no homeless criteria |
| 24 | 99 | Data not collected |
| 25 | 2 | More than one week, but less than one month |
| 25 | 3 | One to three months |
| 25 | 4 | More than three months, but less than one year |
| 25 | 5 | One year or longer |
| 25 | 8 | Client doesn’t know |
| 25 | 9 | Client refused |
| 25 | 10 | One day or less |
| 25 | 11 | Two days to one week |
| 25 | 99 | Data not collected |
| 26 | 1 | Full SSN reported |
| 26 | 2 | Approximate or partial SSN reported |
| 26 | 8 | Client doesn’t know |
| 26 | 9 | Client refused |
| 26 | 99 | Data not collected |
| 27 | AK | AK |
| 27 | AL | AL |
| 27 | AR | AR |
| 27 | AZ | AZ |
| 27 | CA | CA |
| 27 | CO | CO |
| 27 | CT | CT |
| 27 | DE | DE |
| 27 | FL | FL |
| 27 | GA | GA |
| 27 | HI | HI |
| 27 | IA | IA |
| 27 | ID | ID |
| 27 | IL | IL |
| 27 | IN | IN |
| 27 | KS | KS |
| 27 | KY | KY |
| 27 | LA | LA |
| 27 | MA | MA |
| 27 | MD | MD |
| 27 | ME | ME |
| 27 | MI | MI |
| 27 | MN | MN |
| 27 | MO | MO |
| 27 | MS | MS |
| 27 | MT | MT |
| 27 | NC | NC |
| 27 | ND | ND |
| 27 | NE | NE |
| 27 | NH | NH |
| 27 | NJ | NJ |
| 27 | NM | NM |
| 27 | NV | NV |
| 27 | NY | NY |
| 27 | OH | OH |
| 27 | OK | OK |
| 27 | OR | OR |
| 27 | PA | PA |
| 27 | RI | RI |
| 27 | SC | SC |
| 27 | SD | SD |
| 27 | TN | TN |
| 27 | TX | TX |
| 27 | UT | UT |
| 27 | VA | VA |
| 27 | VT | VT |
| 27 | WA | WA |
| 27 | WI | WI |
| 27 | WV | WV |
| 27 | WY | WY |
| 27 | DC | DC |
| 27 | AS | AS |
| 27 | FM | FM |
| 27 | GU | GU |
| 27 | MH | MH |
| 27 | MP | MP |
| 27 | PR | PR |
| 27 | PW | PW |
| 27 | VI | VI |
| 27 | AA | AA |
| 27 | AE | AE |
| 27 | AP | AP |
| 28 | 0 | 0 |
| 28 | 1 | 1 |
| 28 | 2 | 2 |
| 28 | 3 | 3 |
| 28 | 4 | 4 or more |
| 28 | 8 | Client doesn’t know |
| 28 | 9 | Client refused |
| 28 | 99 | Data not collected |
| 29 | 12 | Contact |
| 29 | 14 | Services Provided |
| 29 | 15 | Financial Assistance |
| 29 | 16 | Referrals Provided |
| 29 | 141 | PATH service |
| 29 | 142 | RHY service |
| 29 | 143 | HOPWA service |
| 29 | 144 | SSVF service |
| 29 | 151 | HOPWA financial assistance |
| 29 | 152 | SSVF financial assistance |
| 29 | 161 | PATH referral |
| 29 | 162 | RHY referral |
| 29 | 200 | Bed night |

Table 10: LOV Name Value Details

**<<END OF DOCUMENT>>**

1. For more information about OAuth 2.0 please refer <https://tools.ietf.org/html/draft-ietf-oauth-v2-30#section-1.3.2> [↑](#footnote-ref-2)