



# IH API INTEGRATION GUIDE

INTEGRATION GUIDE FOR USER MANAGEMENT AND PAM SURVEY

---

**TABLE OF CONTENTS**

Introduction .....4

Technology & Security .....4

    1. Technology.....4

    2. Security .....4

Implementation Overview .....4

Client Credentials.....**Error! Bookmark not defined.**

REST API Location .....5

URI Dictionary .....6

API Integration Scenarios.....7

API Integration .....7

    Process Recommendations.....7

    Support Response Time .....8

API Request and Response.....8

    1. Enroll User.....8

    2. Check If User Exist.....10

    3. Update User .....11

    4. Deactivate User .....12

    5. Submit Survey .....12

    PAM 6 .....13

    PAM 6 Statements .....14

    PAM 6 Test Cases .....14

    PAM 10 .....14

    PAM 10 Statements .....15

    PAM 10 Test Cases .....16

    PAM 13 .....16

    PAM 13 Statements .....17

PAM 13 Test Cases .....	19
Survey Response .....	19
6. Request PAM Level and Score .....	20
REST API Status Codes and Messages .....	21
Field level limits and parameter values.....	23
Language Codes .....	24
Sample Data.....	25
1. Enroll User.....	25
2. Submit Survey .....	25
C# .NET Code Example .....	26
Java Code Example.....	28
Document History.....	32

## DOCUMENT HISTORY

Date	Version	Description	Author
<date>	1.0	Initial version with UAT credentials.	IH

## INTRODUCTION

This document outlines the application programming interfaces (API) available to Insignia Health’s clients for interacting with Insignia Health systems. The document includes some examples that should be used for informational purposes.

## TECHNOLOGY & SECURITY

### 1. TECHNOLOGY

- Representational State Transfer (REST) architecture.
- Utilizes Standard Verbs (GET, POST, etc.)
- Stateless Communication: No client context is stored on the server between requests. Prior to generating a request, client is responsible for providing all necessary information.
- Exchange Format: XML.
- Caching: Client can choose to implement caching partially or completely in order to speed/improve the performance on the receiving end.
- Uniform Resource Identifier (URI): Utilized to access API interface.

### 2. SECURITY

- Authentication utilizes Insignia Health provided Client ID and Client Passkey
- Hypertext Transfer Protocol over SSL (HTTPS)

## IMPLEMENTATION OVERVIEW

This section describes Insignia Health’s process for setting up your organization with the Flourish platform.

1. Client signs license agreement with Insignia Health.
2. The client is created in Flourish UAT environment.
3. Insignia Health provides the client with an API guide that includes their UAT credentials. The specification will be password-protected and the password is sent in a separate email.
4. Client develops and tests the API in UAT environment. Once complete, the client informs Insignia Health that they are ready to proceed with the live PRODUCTION environment.

5. Insignia Health verifies the client implementation, including survey administration within client system and data verification in Insignia Health UAT environment.
6. Insignia Health deploys the client to PRODUCTION environment.
7. Insignia Health provides the client with an API guide that includes their PRODUCTION credentials. The specification will be password-protected and the password is sent in a separate email.
8. Insignia Health and the client verifies that API works in Production.

### Client Credentials

To access the API features you will need to pass valid credentials within each request packet as outlined below.

Key Type	Value	XML Example
<b>Client External ID</b>	***	<ClientExtID>***</ClientExtID>
<b>Client Passkey for UAT</b>	***	<ClientPassKey>***</ClientPassKey>
<b>Client Passkey for PROD</b>	***	
<b>Subgroup External ID on UAT</b>	568471853	<SubgroupExtID>568471853<SubgroupExtID>
<b>Survey Name</b>	PAM 13	<SurveyName>PAM 13</SurveyName>
<b>Survey Delivery Mode</b>	System Name – please provide the name of your system that is submitting API calls	<SurveyResponse SurveyDeliveryMode="System Name">

FOR v1.0: Insignia Health will send you passkeys for Production after your organization completes testing on UAT.

FOR v2.0: Client External ID and Subgroup External ID are the same in UAT and Production (IF NOT, PROVIDE IN THE TABLE).

### REST API LOCATION

You can access API on our testing environment by browsing the URI at the following locations:

US UAT environment:	<a href="https://servicesUAT.insigniahealth.com/services/XML/v1.0/ihapi.svc/ping">https://servicesUAT.insigniahealth.com/services/XML/v1.0/ihapi.svc/ping</a>
UKW UAT environment:	<a href="https://servicesUAT.insigniahealth.co.uk/services/XML/v1.0/ihapi.svc">https://servicesUAT.insigniahealth.co.uk/services/XML/v1.0/ihapi.svc</a>
AUS PROD environment:	<a href="https://servicesaus.insigniahealth.com/services/XML/v1.0/ihapi.svc">https://servicesaus.insigniahealth.com/services/XML/v1.0/ihapi.svc</a>

For backward compatibility with older systems that cannot connect with TLS Server Name Introduction (SNI), our UAT and Production environments have been configured with listeners on port 4433. They can be accessed by appending :4433 at the end of the web URL. Example:

<https://servicesUAT.insigniahealth.com:4433/services/XML/v1.0/ihapi.svc/enrolluser>

## URI DICTIONARY

Each individual API call can be accessed via a GET or POST verb available at the following URIs.

Operation	HTTP VERB	REST URI
<b>REST API Endpoint</b>	GET	https://<Web URL>/services/XML/v1.0/ihapi.svc
<b>Ping Check Service Availability</b>	GET	
<b>Check User Exists</b>	POST	https://<Web URL>/services/XML/v1.0/ihapi.svc/checkifuserexists
<b>Enroll User</b>	POST	https://<Web URL>/services/XML/v1.0/ihapi.svc/enrolluser
<b>Update User</b>	POST	https://<Web URL>/services/XML/v1.0/ihapi.svc/updateuser
<b>Deactivate User</b>	POST	http://<Web URL>/services/XML/v1.0/ihapi.svc/deactivateuser
<b>Submit PAM Survey</b>	POST	https://<Web URL>/services/XML/v1.0/ihapi.svc/submitusersurvey
<b>Retrieve PAM Score and Level</b>	POST	https://<Web URL>/services/XML/v1.0/ihapi.svc/getscorelevel

You can send the data to URIs listed above from any HTTP debugging tools such as Fiddler and see exchange of information working in a raw format. To see a code sample for implementing REST client using C# .NET please refer to the section [Accessing REST API via C# .NET](#).

You can visit the following URL in order to access Insignia Health’s help page that includes links to the XSD document:

<https://services.insigniahealth.com/services/XML/v1.0/help.html>

API INTEGRATION SCENARIOS

Scenario #1: Enroll User

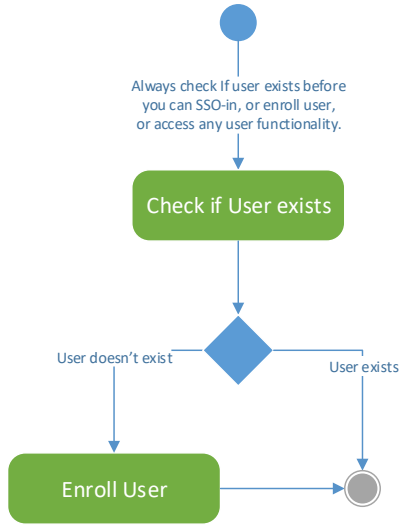


Figure 1. Enroll User

Scenario #2: Update User

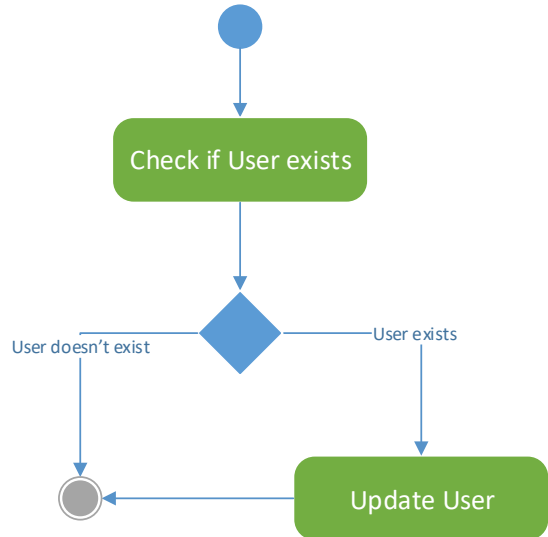


Figure 2. Update User

Scenario #3: Submit Survey

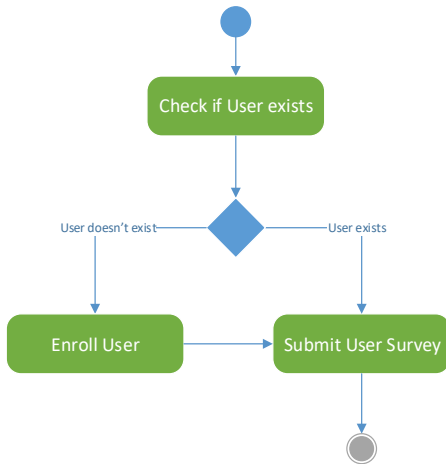


Figure 3. Submit Survey

Scenario #4: Request PAM level and score

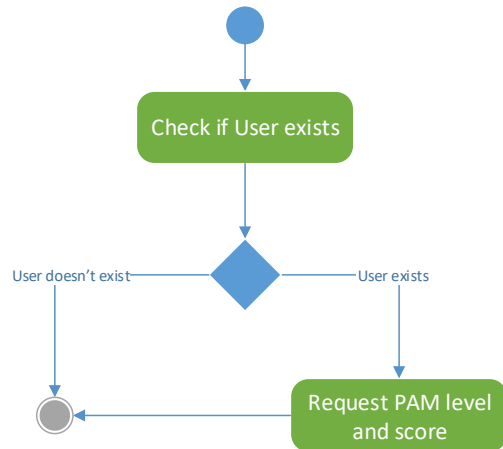


Figure 4. Request PAM level and score

API INTEGRATION

PROCESS RECOMMENDATIONS

- Attempt each service call a maximum of (x) times (recommend 3).
- If after (x) attempts, queue all unsuccessful transactions for later processing. Provide a graceful failure message to the end user.

- Generate an email that is sent to both Insignia Health (support@insigniahealth.com) and your technical support that includes information regarding the failed transaction.
- Once the API becomes available, process all queued transactions.

#### SUPPORT RESPONSE TIME

- Core Business Hours: 7:00 am – 5:00 pm PDT – 2 hours
- Non-Core Business Hours: 5:00 pm – 7:00 am PDT – 8 hours

#### UAT OFFLINE HOURS

- Weekly: Friday, 5:00 pm PST – Monday, 5:00 am PST
- During code release preparation (approximately every 9 weeks for 48 hours), communication will be sent prior to this scheduled downtime

#### UKW UAT OFFLINE HOURS

- Weekly: Friday, 17:00 PST – Sunday, 23:00 PST
- During code release preparation (approximately every 9 weeks for 48 hours), communication will be sent prior to this scheduled downtime

### API REQUEST AND RESPONSE

A typical http post request consists of a header and a request body that you send over https protocol. In order to access Insignia Health’s various API calls you will need to pass the information as outlined below.

For all REST API service calls you will need to pass the Request Header with the following information that tells REST service that incoming data is in XML format.

Request Header
Accept: application/xml Content-Type: text/xml

#### 1. ENROLL USER

Along with the request header information outlined above you will add following request body and POST it to URL https://<Web URL>/services/ihapi.svc/EnrollUser.

REST URI
http://<Web URL>/services/ihapi.svc/EnrollUser



Request Header
Accept: application/xml Content-Type: text/xml
Request Body
<pre>&lt;?xml version="1.0" encoding="utf-8" ?&gt; &lt;request&gt;   &lt;user&gt;     &lt;clientextid&gt;*****&lt;/clientextid&gt;     &lt;clientpasskey&gt;*****&lt;/clientpasskey&gt;     &lt;subgroupextid&gt;*****&lt;/subgroupextid &gt;     &lt;thirdpartyidentifier&gt;18071982001&lt;/thirdpartyidentifier&gt;     &lt;firstname&gt;John &lt;/firstname&gt;     &lt;lastname&gt;Doe&lt;/lastname&gt;     &lt;primaryemail&gt;johndoe@insigniahealth.com&lt;/primaryemail&gt;     &lt;queuesurvey&gt;&gt;false&lt;/queuesurvey&gt;     &lt;username&gt;testuser003&lt;/username&gt;     &lt;password&gt;Password1&lt;/password&gt;     &lt;dob day="18" month="07" year="1982"/&gt;     &lt;gender&gt;male&lt;/gender&gt;     &lt;preferredlanguage&gt;English&lt;/preferredlanguage&gt;     &lt;insurancestatus&gt;POS&lt;/insurancestatus&gt;     &lt;addressline1&gt;133 SW 2nd Avenue&lt;/addressline1&gt;     &lt;addressline2&gt; Suite 300&lt;/addressline2&gt;     &lt;city&gt;Portland&lt;/city&gt;     &lt;state&gt;OR&lt;/state&gt;     &lt;postalcode&gt;97204&lt;/postalcode&gt;     &lt;countrycode&gt;USA&lt;/countrycode&gt;     &lt;primaryphonenumber&gt;5032992800&lt;/primaryphonenumber&gt;     &lt;tcppaccepted&gt;Yes&lt;/tcppaccepted&gt;     &lt;emailoptin&gt;yes&lt;/emailoptin&gt;   &lt;/user&gt; &lt;/request&gt;</pre>
Highlighted fields are required.

The response you get back from REST API is in the XML format. A typical response from Insignia Health’s API will include a StatusCode and StatusMessage. StatusCode is a generic identifier of message contract while StatusMessage will be descriptive information associated with the status code for given operation.

Below is an example of the response for successful registration of a user. If registration was rejected, the response will show the error message. A complete list of error messages is provided in the section “REST API Status Codes and Messages”.

Response Data
---------------

```
<RegisterResponseData xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <StatusCode>Success_Register</StatusCode>
  <StatusMessage> User successfully registered.</StatusMessage>
  <ExternalUserID>160205-BD5L631M</ExternalUserID>
  <ThirdPartyIdentifier>18071982001</ThirdPartyIdentifier>
</RegisterResponseData>
```

## 2. CHECK IF USER EXISTS

Use this method to check if the user exists by the user name, third party ID or external user ID.

<b>REST URI</b>
http://<Web URL>/services/ihapi.svc/CheckIfUserExists
<b>Request Header</b>
Accept: application/xml Content-Type: text/xml
<b>Request Body</b>
<pre>&lt;?xml version="1.0" encoding="utf-8" ?&gt; &lt;request&gt;   &lt;user&gt;     &lt;clientextid&gt;*****&lt;/clientextid&gt;     &lt;clientpasskey&gt;*****&lt;/clientpasskey&gt;     &lt;username&gt; API_UN_2013053122135Z1OR8HX1H6RG&lt;/username&gt;     &lt;thirdpartyidentifier&gt;18071982006&lt;/thirdpartyidentifier&gt;     &lt;externaluserid&gt;201305312213F3S12VT3L0S5X&lt;/externaluserid&gt;   &lt;/user&gt; &lt;/request&gt;</pre>
Use only one of the three highlighted fields to identify a user.

Below is an example of the response for checking existing user.

<b>Response Data</b>
<pre>&lt;UserExistsResponseData xmlns:i="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;StatusCode&gt;Success_ThirdPartyIdentifier&lt;/StatusCode&gt;   &lt;StatusMessage&gt;User with third party identifier exists.&lt;/StatusMessage&gt;   &lt;ExternalUserID&gt;160202-84004FG3&lt;/ExternalUserID&gt;   &lt;ThirdPartyIdentifier&gt;testuser0202&lt;/ThirdPartyIdentifier&gt;   &lt;UserName&gt;User_2797&lt;/UserName&gt; &lt;/UserExistsResponseData&gt;</pre>

### 3. UPDATE USER

Along with the request header information outlined above you will add following request body and POST it to URL `https://<Web URL>/services/ihapi.svc/UpdateUser`

REST URI
<code>http://&lt;Web URL&gt;/services/ihapi.svc/UpdateUser</code>
Request Header
Accept: application/xml Content-Type: text/xml
Request Body
<pre>&lt;?xml version="1.0" encoding="utf-8" ?&gt; &lt;request&gt;   &lt;user&gt;     &lt;clientextid&gt;*****&lt;/clientextid&gt;     &lt;clientpasskey&gt;*****&lt;/clientpasskey&gt;     &lt;username&gt;testuser003&lt;/username&gt;     &lt;thirdpartyidentifier&gt;18071982003&lt;/thirdpartyidentifier&gt;     &lt;ExternalUserID&gt;201302132230325AWQNIJMGVX&lt;/ExternalUserID&gt;   &lt;/user&gt;   &lt;UpdateUserData&gt;     &lt;subgroupextid&gt;*****&lt;/subgroupextid &gt;     &lt;firstname&gt;John&lt;/firstname&gt;     &lt;lastname&gt;Doe&lt;/lastname&gt;     &lt;dob day="18" month="07" year="1982"/&gt;     &lt;gender&gt;male&lt;/gender&gt;     &lt;primaryemail&gt;johndoeNEW@insigniahealth.com&lt;/primaryemail&gt;     &lt;preferredlanguage&gt;English&lt;/preferredlanguage&gt;     &lt;insurancestatus&gt;POS&lt;/insurancestatus&gt;   &lt;/UpdateUserData&gt; &lt;/request&gt;</pre>
Highlighted text is required.
Only one of the three highlighted user identifiers needs to be passed.

Below is an example of the response for updating user's profile.

Response Data
<pre>&lt;UpdateResponseData xmlns:i="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;StatusCode&gt;Success_UpdateUser&lt;/StatusCode&gt;   &lt;StatusMessage&gt;User successfully updated.&lt;/StatusMessage&gt;   &lt;ExternalUserID&gt;160114-HD9KDXJF&lt;/ExternalUserID&gt;</pre>

```
<ThirdPartyIdentifier>18071982003</ThirdPartyIdentifier>
</UpdateResponseData>
```

#### 4. DEACTIVATE USER

To deactivate a user, please use the following request.

REST URI
http://<Web URL>/services/ihapi.svc/DeactivateUser
Request Header
Accept: application/xml Content-Type: text/xml
Request Body
<pre>&lt;?xml version="1.0" encoding="utf-8" ?&gt; &lt;request&gt;   &lt;user&gt;     &lt;clientextid&gt;*****&lt;/clientextid&gt;     &lt;clientpasskey&gt;*****&lt;/clientpasskey&gt;     &lt;username&gt; API_UN_2013053122135Z1OR8HX1H6RG&lt;/username&gt;     &lt;thirdpartyidentifier&gt;18071982006&lt;/thirdpartyidentifier&gt;     &lt;externaluserid&gt;160127-HSKDVEYP &lt;/externaluserid&gt;   &lt;/user&gt; &lt;/request&gt;</pre>
Use only one of the three highlighted fields to identify a user.

Below is an example of the response for updating user's profile.

Response Data
<pre>&lt;DeactivateResponseData xmlns:i="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;StatusCode&gt;Success_UserDeactivation&lt;/StatusCode&gt;   &lt;StatusMessage&gt;User successfully deactivated.&lt;/StatusMessage&gt;   &lt;MemberId&gt;160127-HSKDVEYP&lt;/MemberId&gt; &lt;/DeactivateResponseData&gt;</pre>

#### 5. SUBMIT SURVEY

Use these values for mapping user's answers:

User responded:	Value:
Disagree Strongly	1
Disagree	2

User responded:	Value:
Agree	3
Agree Strongly	4
N/A	0

PAM 6

If your organization is using PAM 6, please use the following request structure.

REST URI
http://<Web URL>/services/ihapi.svc/SubmitUserSurvey
Request Header
Accept: application/xml Content-Type: text/xml
Request Body
<pre> &lt;?xml version="1.0" encoding="utf-8" ?&gt; &lt;Request&gt;   &lt;user&gt;     &lt;clienttextid&gt;*****&lt;/clienttextid&gt;     &lt;clientpasskey&gt;*****&lt;/clientpasskey&gt;     &lt;username&gt;testuser003&lt;/username&gt;     &lt;thirdpartyidentifier&gt;18071982003&lt;/thirdpartyidentifier&gt;     &lt;ExternalUserID&gt;201302132230325AWQNIJMGVX&lt;/ExternalUserID&gt;   &lt;/user&gt;   &lt;Survey&gt;     &lt;Language&gt;en&lt;/Language&gt;     &lt;SurveyName&gt;PAM 6&lt;/SurveyName&gt;     &lt;Administration day="25" month="3" year="2010" /&gt;     &lt;SurveyResponse Age="" Gender="" SurveyDeliveryMode="System Name"&gt;       &lt;Answer ID="PA1"&gt;4&lt;/Answer&gt;       &lt;Answer ID="PA2"&gt;2&lt;/Answer&gt;       &lt;Answer ID="PA5"&gt;0&lt;/Answer&gt;       &lt;Answer ID="PA9"&gt;3&lt;/Answer&gt;       &lt;Answer ID="PA10"&gt;3&lt;/Answer&gt;       &lt;Answer ID="PA13"&gt;1&lt;/Answer&gt;     &lt;/SurveyResponse&gt;   &lt;/Survey&gt; &lt;/Request&gt; </pre>
<p>Highlighted text is required for client identification.</p> <p>Survey attributes are all required.</p> <p>Only one of the three highlighted user identifiers needs to be passed.</p>

PAM 6 STATEMENTS

#	Statement	Answer ID
1.	When all is said and done, I am the person who is responsible for taking care of my health.	PA1
2.	Taking an active role in my own health care is the most important thing that affects my health.	PA2
3.	I am confident that I can tell whether I need to go to the doctor or whether I can take care of a health problem myself.	PA5
4.	I know what treatments are available for my health problems.	PA9
5.	I have been able to maintain (keep up with) lifestyle changes, like eating right or exercising.	PA10
6.	I am confident that I can maintain lifestyle changes, like eating right and exercising, even during times of stress.	PA13

PAM 6 TEST CASES

PA1	PA2	PA5	PA9	PA10	PA13	PAM Score	PAM Level
Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	0.0	1
Disagree	Disagree	Disagree	Disagree	Disagree	Disagree	33.0	1
Agree	Agree	Agree	Agree	Agree	Agree	51.0	2
Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	100.0	4
N/A	N/A	N/A	N/A	N/A	N/A	51.0	2
Disagree	Disagree	Disagree	Disagree Strongly	Disagree Strongly	Disagree Strongly	22.2	1
Agree Strongly	Agree Strongly	Agree Strongly	Agree	Agree	Agree	74.7	4
Agree	Agree	Agree	Agree	Agree	N/A	57.1	3
N/A	N/A	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	51.0	2
Disagree Strongly	Disagree	Agree	Agree Strongly	Agree	N/A	47.2	2

PAM 10

For PAM 10, please use the following request structure.

<b>REST URI</b>
http://<Web URL>/services/ihapi.svc/SubmitUserSurvey
<b>Request Header</b>
Accept: application/xml Content-Type: text/xml

**Request Body**

```
<?xml version="1.0" encoding="utf-8" ?>
<Request>
  <user>
    <clienttextid>*****</clienttextid>
    <clientpasskey>*****</clientpasskey>
    <username>testuser003</username>
    <thirdpartyidentifier>18071982003</thirdpartyidentifier>
    <ExternalUserID>201302132230325AWQNIJMGVX</ExternalUserID>
  </user>
  <Survey>
    <Language>en</Language>
    <SurveyName>PAM 10</SurveyName>
    <Administration day="25" month="3" year="2010" />
    <SurveyResponse Age="" Gender="" SurveyDeliveryMode="System Name">
      <Answer ID="PA1">4</Answer>
      <Answer ID="PA2">2</Answer>
      <Answer ID="PA4">1</Answer>
      <Answer ID="PA5">0</Answer>
      <Answer ID="PA6">4</Answer>
      <Answer ID="PA7">3</Answer>
      <Answer ID="PA10">3</Answer>
      <Answer ID="PA11">2</Answer>
      <Answer ID="PA12">4</Answer>
      <Answer ID="PA13">1</Answer>
    </SurveyResponse>
  </Survey>
</Request>
```

Highlighted text is required for client identification.

Survey attributes are all required.

Only one of the three highlighted user identifiers needs to be passed.

PAM 10 STATEMENTS

#	Statement	Answer ID
1.	When all is said and done, I am the person who is responsible for taking care of my health.	PA1
2.	Taking an active role in my own health care is the most important thing that affects my health.	PA2
3.	I know what each of my prescribed medications do.	PA4
4.	I am confident that I can tell whether I need to go to the doctor or whether I can take care of a health problem myself.	PA5
5.	I am confident that I can tell a doctor concerns I have even when he or she does not ask.	PA6

6.	I am confident that I can follow through on medical treatments I may need to do at home.	PA7
7.	I have been able to maintain (keep up with) lifestyle changes, like eating right or exercising.	PA10
8.	I know how to prevent problems with my health.	PA11
9.	I am confident I can figure out solutions when new problems arise with my health.	PA12
10.	I am confident that I can maintain lifestyle changes, like eating right and exercising, even during times of stress.	PA13

PAM 10 TEST CASES

PA1	PA2	PA4	PA5	PA6	PA7	PA10	PA11	PA12	PA13	PAM Score	PAM Level
Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	0.0	1
Disagree	Disagree	Disagree	Disagree	Disagree	Disagree	Disagree	Disagree	Disagree	Disagree	33.6	1
Agree	Agree	Agree	Agree	Agree	Agree	Agree	Agree	Agree	Agree	51.0	2
Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	100.0	4
N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	51.0	2
Disagree	Disagree	Disagree	Disagree	Disagree	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	24.1	1
Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree	Agree	Agree	Agree	Agree	72.1	3
Disagree	Disagree	Disagree	Disagree	Disagree	Agree	Disagree	N/A	N/A	N/A	51.0	2
N/A	N/A	N/A	Agree	Agree	Agree	Agree	Agree	Agree	Agree	51.0	2
Disagree Strongly	Disagree	Agree	Agree Strongly	N/A	Disagree Strongly	Disagree Strongly	Agree	Agree Strongly	N/A	42.9	1

PAM 13

If your organization is using PAM 13, please submit your requests in the following format.

<b>REST URI</b>
http://<Web URL>/services/ihapi.svc/SubmitUserSurvey
<b>Request Header</b>
Accept: application/xml Content-Type: text/xml
<b>Request Body</b>
<?xml version="1.0" encoding="utf-8" ?> <Request> <user>



```

<clienttextid>*****</clienttextid>
<clientpasskey>*****</clientpasskey>
<username>testuser003</username>
<thirdpartyidentifier>18071982003</thirdpartyidentifier>
<ExternalUserID>201302132230325AWQNIJMGVX</ExternalUserID>
</user>
<Survey>
  <Language>en</Language>
  <SurveyName>PAM 13</SurveyName>
  <Administration day="25" month="3" year="2010" />
  <SurveyResponse Age="" Gender="" SurveyDeliveryMode="System Name">
    <Answer ID="PA1">4</Answer>
    <Answer ID="PA2">2</Answer>
    <Answer ID="PA3">1</Answer>
    <Answer ID="PA4">1</Answer>
    <Answer ID="PA5">0</Answer>
    <Answer ID="PA6">4</Answer>
    <Answer ID="PA7">3</Answer>
    <Answer ID="PA8">1</Answer>
    <Answer ID="PA9">1</Answer>
    <Answer ID="PA10">3</Answer>
    <Answer ID="PA11">2</Answer>
    <Answer ID="PA12">4</Answer>
    <Answer ID="PA13">1</Answer>
  </SurveyResponse>
</Survey>
</Request>

```

Highlighted text is required for client identification.

Survey attributes are all required.

Only one of the three highlighted user identifiers needs to be passed.

PAM 13 STATEMENTS

#	Statement	Answer ID
1.	When all is said and done, I am the person who is responsible for taking care of my health.	PA1
2.	Taking an active role in my own health care is the most important thing that affects my health.	PA2
3.	I am confident I can help prevent or reduce problems associated with my health.	PA3
4.	I know what each of my prescribed medications do.	PA4
5.	I am confident that I can tell whether I need to go to the doctor or whether I can take care of a health problem myself.	PA5
6.	I am confident that I can tell a doctor concerns I have even when he or she does not ask.	PA6
7.	I am confident that I can follow through on medical treatments I may need to do at home.	PA7
8.	I understand my health problems and what causes them.	PA8

9.	I know what treatments are available for my health problems.	PA9
10.	I have been able to maintain (keep up with) lifestyle changes, like eating right or exercising.	PA10
11.	I know how to prevent problems with my health.	PA11
12.	I am confident I can figure out solutions when new problems arise with my health.	PA12
13.	I am confident that I can maintain lifestyle changes, like eating right and exercising, even during times of stress.	PA13

PAM 13 TEST CASES

Survey #	PA1	PA2	PA3	PA4	PA5	PA6	PA7
1.	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly
2.	Disagree	Disagree	Disagree	Disagree	Disagree	Disagree	Disagree
3.	Agree	Agree	Agree	Agree	Agree	Agree	Agree
4.	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly
5.	N/A	N/A	N/A	N/A	N/A	N/A	N/A
6.	Disagree	Disagree	Disagree	Disagree	Disagree	Disagree	Disagree
7.	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly
8.	Disagree	Disagree	Disagree	Disagree	Disagree	Disagree	Disagree
9.	N/A	N/A	N/A	N/A	Agree	Agree	Agree
10.	Disagree Strongly	Disagree	Agree	Agree Strongly	N/A	Disagree Strongly	Disagree

Survey #	PA8	PA9	PA10	PA11	PA12	PA13	PAM Score	PAM Level
1.	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	0.0	1
2.	Disagree	Disagree	Disagree	Disagree	Disagree	Disagree	34.2	1
3.	Agree	Agree	Agree	Agree	Agree	Agree	51.0	2
4.	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	Agree Strongly	100.0	4
5.	N/A	N/A	N/A	N/A	N/A	N/A	51.0	2
6.	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	Disagree Strongly	26.1	1
7.	Agree	Agree	Agree	Agree	Agree	Agree	72.5	4
8.	Disagree	N/A	N/A	N/A	N/A	N/A	51.0	2
9.	Agree	Agree	Agree	Agree	Agree	Agree	51.0	2
10.	Agree	Agree Strongly	N/A	Disagree Strongly	Disagree	Agree	40.7	1

SURVEY RESPONSE

When submitting a request with survey responses you will receive back PAM score and PAM level. An example of this type of response is below.

```

Response with PAM score and PAM level
<UserSurveyResponseData xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <StatusCode>Success_SurveySubmit</StatusCode>
    
```

```
<StatusMessage>Submit User Survey success.</StatusMessage>
<SurveyEnteredDateTime>2015-03-25T00:00:00</SurveyEnteredDateTime>
<SurveyName>PAM 10</SurveyName>
<SurveyResult>
  <ResponseData>
    <Type id="PAMLevel" value="1"/>
    <Type id="PAMScore" value="35.50"/>
  </ResponseData>
</SurveyResult>
<ThirdPartyIdentifier>100120131015AVVel21</ThirdPartyIdentifier>
</UserSurveyResponseData>
```

## 6. REQUEST PAM LEVEL AND SCORE

Along with the request header information outlined above you will add following request body and POST it to URL <https://<Web URL>/services/ihapi.svc/GetScoreLevel>

<b>REST URI</b>
http://<Web URL>/services/ihapi.svc/GetScoreLevel
<b>Request Header</b>
Accept: application/xml Content-Type: text/xml
<b>Request Body</b>
<?xml version="1.0" encoding="utf-8" ?>         <request>           <user>             <clientextid>*****</clientextid>             <clientpasskey>*****</clientpasskey>             <thirdpartyidentifier>100120131015AVVel21</thirdpartyidentifier>           </user>         </request>

When submitting a request for PAM level and PAM score for existing user you will receive back PAM score and PAM level. An example of this type of response is below.

<b>Response with PAM score and PAM level for existing user</b>
<UserSurveyResponseData xmlns:i="http://www.w3.org/2001/XMLSchema-instance">         <StatusCode>Success_SurveyDetails</StatusCode>         <StatusMessage>Get Score/Level details success.</StatusMessage>         <SurveyEnteredDateTime>2015-03-25T00:00:00</SurveyEnteredDateTime>         <SurveyName>PAM10</SurveyName>

```

<SurveyResult>
  <ResponseData>
    <Type id="PAMLevel" value="1"/>
    <Type id="PAMScore" value="35.50"/>
  </ResponseData>
</SurveyResult>
<ThirdPartyIdentifier>100120131015AVVel21</ThirdPartyIdentifier>
</UserSurveyResponseData>
    
```

## REST API STATUS CODES AND MESSAGES

Following is a list of status codes available as of today's date. More status codes will continue to be added as the API evolves over time.

API Call	Status Code	Status Message
All calls	Error_GeneralError	Sorry. Something went wrong on our end. Please try again.
All calls	Error_InActiveClient	Sorry. You are not an active client.
EnrollUser	Error_InactiveEnrollmentAPI	Sorry. You do not have this feature available. Enrollment API failed.
SubmitUserSurvey	Error_InactiveSurveyAPI	Sorry. You do not have this feature available. Survey API failed.
CheckIfUserExists	Error_ExternalIdentifier	User with external identifier doesn't exist or invalid.
CheckIfUserExists	Success_ExternalIdentifier	User with external identifier exists.
CheckIfUserExists	Error_ThirdPartyIdentifier	User with third party identifier doesn't exist or invalid.
CheckIfUserExists	Success_ThirdPartyIdentifier	User with third party identifier exists.
CheckIfUserExists	Success_UserName	Username exists.
CheckIfUserExists	Error_CheckUserFailed	Invalid ClientExternalIdentifier or Input request. Check user exists failed.
CheckIfUserExists	Error_InvalidUserCredential	Sorry. We could not find user with the specified credentials.
CheckIfUserExists DeactivateUser SubmitUserSurvey	Error_InactiveUserSubGroup	The subgroup for the user specified is disabled.
EnrollUser	Success_Register	User successfully registered.
EnrollUser	Error_DuplicateThirdPartyIdentifier	Third Party Identifier is duplicate.
EnrollUser	Error_InvalidThirdPartyIdentifier	Third Party Identifier is invalid.
EnrollUser	Error_DuplicateUsername	UserName is either duplicate or invalid.

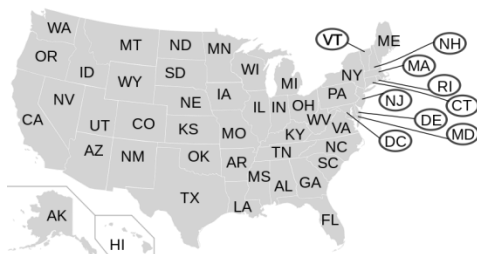
API Call	Status Code	Status Message
EnrollUser	Error_InvalidAccessCode	Access Code is invalid.
EnrollUser	Error_InvalidPrimaryEmail	Email is in invalid format.
EnrollUser	Error_DuplicatePrimaryEmail	Email is either duplicate or in invalid format.
EnrollUser	Error_RegisterFailed	Invalid ClientExternalIdentifier or Input request. Registration failed.
EnrollUser UpdateUser	Error_InvalidGender	Gender is invalid. User enrollment failed.
EnrollUser UpdateUser	Error_InvalidDOB	DOB is invalid. User enrollment failed.
EnrollUser UpdateUser	Error_InactiveSubGroup	This subgroup is disabled.
UpdateUser	Error_UpdateUserFailed	Invalid ClientExternalIdentifier or Input request. Update user failed.
UpdateUser	Error_UserUpdateFailed	Failed to update user.
UpdateUser	Success_UpdateUser	User successfully updated.
SubmitUserSurvey	Error_SurveySubmit	Submit User Survey failed.
SubmitUserSurvey	Success_SurveySubmit	Submit User Survey success.
SubmitUserSurvey	Error_SurveyInvalid	This survey is no longer valid. Submit User Survey failed.
SubmitUserSurvey	Error_SurveyAlreadyTaken	This survey has already been taken. Submit User Survey failed.
SubmitUserSurvey	Error_SurveyExpired	This survey is expired. Submit User Survey failed.
SubmitUserSurvey	Error_SurveyInactive	This survey is InActive. Submit User Survey failed.
SubmitUserSurvey	Error_SurveySchedule	Could not administer or schedule a survey for this user. Submit User Survey failed.
SubmitUserSurvey	Error_SurveyFailed	Invalid ClientExternalIdentifier or Input request. Submit user survey failed.
DeactivateUser	Error_DeactivateUserFailed	Invalid ClientExternalIdentifier or Input request. Deactivate user failed.
DeactivateUser	Success_UserDeactivation	User successfully deactivated.
DeactivateUser	Error_UserDeactivationFailed	Failed to deactivate user.
SubmitUserSurvey	Success_SurveyDetails	Get Score/Level details success.
SubmitUserSurvey	Error_PAMSurveyNotFound	Get Score/Level PAM Survey Response details not found.
SubmitUserSurvey	Error_SurveyDetails	Get Score/Level details failed.

API Call	Status Code	Status Message
SubmitUserSurvey	Error_InvalidSurveyDate	The date specified is not a valid date for this survey. Please select a date between <Start Date> and <End Date>.
SubmitUserSurvey	Error_InvalidLanguage	The language provided as a preferred language for a user or a survey language for a survey is not in the list of supported languages.

**FIELD LEVEL LIMITS AND PARAMETER VALUES**

The request input XML fields that you pass will need to conform to following field level limits and valid values.

Field	Limit	Data Type/ Valid Values
<username>	100	Alpha numeric
<thirdpartyidentifier>	50	Alpha numeric
<password>	255	Alpha numeric
<primaryemail>	100	Alpha numeric
<firstname>	35	Alpha numeric
<lastname>	60	Alpha numeric
<dob day="dd" month="mm" year="yyyy"/>	dd=2, mm=2, yyyy=4 digits	Numeric
<gender>		<ul style="list-style-type: none"> <li>- Undefined</li> <li>- Male</li> <li>- Female</li> <li>- Undetermined</li> </ul>
<addressline1>	80	Alpha numeric
<addressline2>	80	Alpha numeric
<addressline3>	80	Alpha numeric
<city>	50	Alpha numeric
<state>	50 <b>Postal code for US states</b> and free form value for other countries.	Alpha numeric 2 letter code value USA states. e.g. OR for Oregon <a href="https://www.usps.com/send/official-abbreviations.htm">https://www.usps.com/send/official-abbreviations.htm</a>
<postalcode>	15	Alpha numeric



Field	Limit	Data Type/ Valid Values
<countrycode>	3	Alpha numeric 3 letter codes for countries: e.g. USA, IND, CAN, CHN etc. Refer this link for values. <a href="http://en.wikipedia.org/wiki/ISO_3166-1">http://en.wikipedia.org/wiki/ISO_3166-1</a>
<primaryphonenumber>	15	Alpha numeric
<EmailOptIn>	3	- yes - no
<tcpaccepted >	3	- yes - no
<preferredlanguage>	15	Alpha numeric; all languages are listed in the table below. Submit language code or language name.
<insurancestatus>	25	- Dual Eligible - HMO - HSA: HDHP - Medicaid - Medicaid: Low Utilizer - Medicaid: Non Utilizer - Medicare - No Insurance - POS - PPO - Unknown
<Language> within EnrollUser and SubmitUserSurvey	15	Alpha numeric; all languages are listed in the table below. Submit language code or language name.

## LANGUAGE CODES

Please, use the following language codes when submitting language preferences for a user or submitting a survey.

#	Language Code	Language Name
1.	ara	Arabic
2.	ben	Bengali
3.	bul	Bulgarian
4.	sng	Chinese (Cantonese)
5.	cmn	Chinese (Mandarin)
6.	chk	Chuukese
7.	hat	Creole (Haitian)
8.	ces	Czech
9.	dan	Danish

#	Language Code	Language Name
10.	nid	Dutch
11.	enu	English - UK
12.	eng, en	English
13.	fil	Filipino
14.	fin	Finnish
15.	fra	French
16.	frc	French Canadian
17.	deu	German
18.	grk	Greek



#	Language Code	Language Name
19.	guj	Gujarati
20.	heb	Hebrew
21.	hin	Hindi
22.	hun	Hungarian
23.	ita	Italian
24.	jpn	Japanese
25.	kor	Korean
26.	msa	Malay
27.	mah	Marshallese
28.	nor	Norwegian
29.	oth	Other
30.	pol	Polish
31.	ptb	Portuguese - Brazil
32.	por	Portuguese - Portugal
33.	pan	Punjabi

#	Language Code	Language Name
34.	rom	Romanian
35.	rus	Russian
36.	smo	Samoan
37.	Slk	Slovak
38.	som	Somali
39.	esp	Spanish – Puerto Rico
40.	esm, es	Spanish - Mexico
41.	spa	Spanish – Spain
42.	swe	Swedish
43.	tgl	Tagalog
44.	tam	Tamil
45.	tur	Turkish
46.	ukr	Ukrainian
47.	urd	Urdu
48.	vit	Vietnamese

## SAMPLE DATA

### 1. ENROLL USER

Sample Test EnrollUserData.xml file used for registering with code above looks as below.

Please, note that this file is provided as a sample and final XML structure may vary over the period of development.

```

EnrollUserData.xml
<?xml version="1.0" encoding="utf-8" ?>
<request>
  <user>
    <clientextid>*****</clientextid>
    <clientpasskey>*****</clientpasskey>
    <subgroupextid>*****</subgroupextid >
    <thirdpartyidentifier>555111222</thirdpartyidentifier>
    <primaryemail>18079252013@insigniahealth.com</primaryemail>
  </user>
</request>
    
```

### 2. SUBMIT SURVEY

Sample SubmitSurveyData.xml file used for submitting survey is as below. Please, make sure to submit correct Survey Name and correct number of survey answers based on survey type: PAM 6, PAM 10 or PAM 13.

Note: <thirdpartyidentifier> is the same as what you submitted during EnrollUser call above.

**SubmitSurveyData.xml**

```

<?xml version="1.0" encoding="utf-8" ?>
<Request>
  <user>
    <clienttextid>*****</clienttextid>
    <clientpasskey>*****</clientpasskey>
    <thirdpartyidentifier>555111222</thirdpartyidentifier>
  </user>
  <Survey>
    <Language>en</Language>
    <SurveyName>PAM10</SurveyName>
    <Administration day="3" month="12" year="2013" />
    <SurveyResponse Age="" Gender="" SurveyDeliveryMode="System Name">
      <Answer ID="PA1">4</Answer>
      <Answer ID="PA2">4</Answer>
      <Answer ID="PA4">3</Answer>
      <Answer ID="PA5">0</Answer>
      <Answer ID="PA6">4</Answer>
      <Answer ID="PA7">3</Answer>
      <Answer ID="PA10">3</Answer>
      <Answer ID="PA11">3</Answer>
      <Answer ID="PA12">1</Answer>
      <Answer ID="PA13">2</Answer>
    </SurveyResponse>
  </Survey>
</Request>

```

**C# .NET CODE EXAMPLE**

Following sample code demonstrates how you could connect to REST API Interface provided by Insignia Health.

Create APITestHarness solution using visual studio and paste the code as below in the Main function.

**Create APITestHarness Console App project**

```

using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Net;
using System.Text;
using System.Xml;
using System.Xml.Linq;

namespace APITestHarness
{
  class Program
  {
    static void Main(string[] args)
    {
      HttpWebRequest req = null;
      HttpWebResponse res = null;
      bool _enrollUser = true;
      bool _submitSurvey = true;
      try
      {

```

```

Console.WriteLine("*** Enroll user using IH API.*** \n");
if (_enrollUser)
{
    String URI = "https://<Web URL>/services/ihapi.svc/EnrollUser";

    req = (HttpWebRequest)WebRequest.Create(URI);
    req.Method = "POST";
    req.ContentType = "application/xml; charset=utf-8";
    req.Timeout = 30000;
    req.Headers.Add("SOAPAction", URI);

    XmlDocument xmlDoc = new System.Xml.XmlDocument();
    xmlDoc.XmlResolver = null;
    xmlDoc.Load(@"..\..\EnrollUserData.xml");
    string sXML = xmlDoc.InnerXml;
    req.ContentLength = sXML.Length;
    System.IO.StreamWriter sw = new
System.IO.StreamWriter(req.GetRequestStream());
    sw.Write(sXML);
    sw.Close();

    res = (HttpWebResponse)req.GetResponse();

    XDocument xDoc = XDocument.Load(new
StreamReader(res.GetResponseStream()));

    System.Console.WriteLine(xDoc.Document);
}
Console.WriteLine("\n *** Now Submit survey for the user you just created
above. *** \n");
if (_submitSurvey)
{
    String URI ="https://<Web URL>/services/ihapi.svc/SubmitUserSurvey";

    req = (HttpWebRequest)WebRequest.Create(URI);
    req.Method = "POST";
    req.ContentType = "application/xml; charset=utf-8";
    req.Timeout = 30000;
    req.Headers.Add("SOAPAction", URI);

    XmlDocument xmlDoc = new System.Xml.XmlDocument();
    xmlDoc.XmlResolver = null;
    xmlDoc.Load(@"..\..\SubmitSurveyData.xml");
    string sXML = xmlDoc.InnerXml;
    req.ContentLength = sXML.Length;
    System.IO.StreamWriter sw = new
System.IO.StreamWriter(req.GetRequestStream());
    sw.Write(sXML);
    sw.Close();

    res = (HttpWebResponse)req.GetResponse();

    XDocument xDoc = XDocument.Load(new
StreamReader(res.GetResponseStream()));

    System.Console.WriteLine(xDoc.Document);
}
}
catch (Exception ex)
{
    System.Console.WriteLine(ex.Message);
}

```

```

    }
  }
}

```

## JAVA CODE EXAMPLE

Save this code in a file named as InsigniaAPIClient.java then At command prompt run:

```
>javac InsigniaAPIClient.java
```

```
>java InsigniaAPIClient
```

```

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.io.OutputStreamWriter;
import java.net.URL;
import java.net.URLConnection;

public class InsigniaAPIClient {

    public static void main(String[] args) {

        StringBuffer buffer = new StringBuffer();

        java.net.URL url;
        String result;
        String line;
        String response = "";
        URLConnection connection;

        try {
            url = new URL( "https://<Web URL>/Services/IHAPI.svc/EnrollUser" );
            connection = url.openConnection();
            connection.setDoOutput( true );
            connection.setRequestProperty("User-Agent", "Mozilla/5.0 (Windows; U; Windows NT
6.0; en-US; rv:1.9.0.5) Gecko/2008120122 Firefox/3.0.5");
            connection.setRequestProperty("Content-Type", "application/xml");
            connection.setRequestProperty("Accept", "*/*");

            OutputStreamWriter wr = new OutputStreamWriter(connection.getOutputStream(),
"UTF-8");
            /*
            //This is how you need to pass the data to us for enrolling a user. Following XML
fields are minimum required fields.
            //Note: You can also send us first name, date of birth, address, email, etc.
            //Third party identifier & email is your unique id for the user that you will use to
enroll in our (Insignia Health) system.

            <?xml version="1.0" encoding="utf-8" ?>
            <request>
                <user>

```

```

        <clienttextid>283504140</clienttextid>
        <clientpasskey>P@ssword1</clientpasskey>
        <subgroupextid>321038815</subgroupextid >
        <thirdpartyidentifier>1234567890</thirdpartyidentifier>
        <primaryemail>1234567890@insigniahealth.com</primaryemail>
        <tcppaccepted>Yes</tcppaccepted>
    </user>
</request>
*/
buffer.append("<?xml version='1.0' encoding='utf-8' ?>");
buffer.append("<request>");
buffer.append("<user>");
buffer.append("<clienttextid>*****</clienttextid>");
buffer.append("<clientpasskey>*****</clientpasskey>");
buffer.append("<subgroupextid>*****</subgroupextid >");
buffer.append("<thirdpartyidentifier>111222333444</thirdpartyidentifier>");
buffer.append("<primaryemail>111222333444@insigniahealth.com</primaryemail>");
buffer.append("</user>");
buffer.append("</request>");

result = new String( buffer.toString().getBytes(), "UTF-8" );
System.out.println( "\n *Input XML sent to Insignia Health API * \n\n" + result +
"\n\n\n\n\n");
wr.write( result );
wr.flush();
wr.close();

BufferedReader rd = new BufferedReader(new
InputStreamReader(connection.getInputStream()));
while ((line = rd.readLine()) != null) {
response += line;
}
wr.close();
rd.close();

System.out.println( "\n *Output recieved from API * \n\n" + response +
"\n\n");

}
catch(Exception e){
System.out.println("Errors...");
System.out.println(e);
}

}

/*
+-----+
|Submit Survey for user you just created above.
<thirdpartyidentifier>111222333444</thirdpartyidentifier>
+-----+

```

```

*/

try {
    url = new URL( "https://<WebURL>/Services/IHAPI.svc/SubmitUserSurvey");
    connection = url.openConnection();
    connection.setDoOutput( true );
    connection.setRequestProperty("User-Agent", "Mozilla/5.0 (Windows; U; Windows NT
6.0; en-US; rv:1.9.0.5) Gecko/2008120122 Firefox/3.0.5");
    connection.setRequestProperty("Content-Type", "application/xml");
    connection.setRequestProperty("Accept", "*/*");

    OutputStreamWriter wr = new
    OutputStreamWriter(connection.getOutputStream(), "UTF-8");
    /*
    // Submit Survey For the USer you just registerd above:
    <?xml version="1.0" encoding="utf-8" ?>
    <Request>
    <user>
    <clientextid>*****</clientextid>
    <clientpasskey>*****</clientpasskey>
    <thirdpartyidentifier>111222333444</thirdpartyidentifier>
    </user>
    <Survey>
    <Language>en</Language>
    <SurveyName>PAM10</SurveyName>
    <Administration day="19" month="11" year="2012" />
    <SurveyResponse Age="" Gender="" SurveyDeliveryMode="System Name">
    <Answer ID="PA1">1</Answer>
    <Answer ID="PA2">2</Answer>
    <Answer ID="PA4">4</Answer>
    <Answer ID="PA5">0</Answer>
    <Answer ID="PA6">1</Answer>
    <Answer ID="PA7">2</Answer>
    <Answer ID="PA10">0</Answer>
    <Answer ID="PA11">1</Answer>
    <Answer ID="PA12">2</Answer>
    <Answer ID="PA13">3</Answer>
    </SurveyResponse>
    </Survey>
    </Request>
    */
    buffer.setLength(0); // Clean string buffer.

    buffer.append("<?xml version='1.0' encoding='utf-8' ?>");
    buffer.append("<request>");
    buffer.append("<user>");
    buffer.append("<clientextid>*****</clientextid>");
    buffer.append("<clientpasskey>*****</clientpasskey>");
    buffer.append("<thirdpartyidentifier>111222333444</thirdpartyidentifier>");
    buffer.append("</user>");

```

```

        buffer.append("<Survey>");
        buffer.append("<Language>en</Language>");
        buffer.append("<SurveyName>PAM10</SurveyName>");
        buffer.append("<Administration day='19' month='12' year='2013' />");
        buffer.append("<SurveyResponse Age='' Gender='' SurveyDeliveryMode='System
Name'>");
        buffer.append("<Answer ID='PA1'>1</Answer>");
        buffer.append("<Answer ID='PA2'>2</Answer>");
        buffer.append("<Answer ID='PA4'>4</Answer>");
        buffer.append("<Answer ID='PA5'>0</Answer>");
        buffer.append("<Answer ID='PA6'>1</Answer>");
        buffer.append("<Answer ID='PA7'>2</Answer>");
        buffer.append("<Answer ID='PA10'>0</Answer>");
        buffer.append("<Answer ID='PA11'>1</Answer>");
        buffer.append("<Answer ID='PA12'>2</Answer>");
        buffer.append("<Answer ID='PA13'>3</Answer>");
        buffer.append("</SurveyResponse>");
        buffer.append("</Survey>");
        buffer.append("</request>");

        result = new String( buffer.toString().getBytes(), "UTF-8" );
        System.out.println( "\n *Input XML sent to Insignia Health API * \n\n" +
result + "\n\n\n\n\n\n");
        wr.write( result );
        wr.flush();
        wr.close();
        BufferedReader rd = new BufferedReader(new
InputStreamReader(connection.getInputStream()));
        while ((line = rd.readLine()) != null) {
            response += line;
        }
        wr.close();
        rd.close();

        System.out.println( "\n *Output received from API * \n\n" + response +
"\n\n");
    }
    catch(Exception e){
        System.out.println("Errors...");
        System.out.println(e);
    }
}
}
}

```

**DOCUMENT HISTORY**

Date	Version	Description	Author
<b>May 31, 2013</b>	1.0	Initial version.	IH
<b>December 04, 2013</b>	1.1	Added examples for each type of API request.	IH
<b>February 26, 2014</b>	1.2	Added PAM response example.	IH
<b>April 25, 2014</b>	1.3	Added a new API call for getting PAM level and score for existing users.	IH
<b>May 16, 2014</b>	1.4	Added support information.	IH
<b>July 30, 2014</b>	1.5	Added request for deactivating users.	IH
<b>October 28, 2014</b>	1.6	Modified to follow IH process for setting up API clients.	IH
<b>February 20, 2015</b>	1.7	Modified IH process for setting up API clients.	IH
<b>April 28, 2015</b>	1.8	Added PAM Completion Date to responses. Added error messages for verifying survey administration date. Added Survey Delivery Mode attribute to survey submission request.	IH
<b>June 3, 2015</b>	1.9	Changed UAT credentials.	IH
<b>June 12, 2015</b>	1.10	Removed credentials to access Demo site.	IH
<b>February 4, 2016</b>	2.0	Restructured the document and added support for non-SNI API calls.	IH
<b>June 15, 2016</b>	2.0	Added a new tag for user's insurance type. Added URL to EUN.	IH
<b>June 16, 2016</b>	3.0	Added language codes for surveys. Added a new tag for user's language preference.	IH
<b>October 21, 2016</b>	3.1	Removed fields from UpdateUser that cannot be updated. They will be added back with the next release.	IH
<b>February 3, 2017</b>	Mansfield	Added a new attribute for updating Subgroup External ID to the Update User request.	IH
<b>March 6, 2017</b>	Amber	Masked client IDs and passkeys and updated some tag values within provided examples.	IH
<b>April 14, 2017</b>	Amber	Added test cases and survey statement texts.	IH
<b>October 4, 2017</b>	Emerald	Added new status codes and messages for responses for deactivated subgroup. Added new status codes and messages for DOB and gender validation.	IH