|  |
| --- |
|  |
| Design Specification for  eLearning Management System  Document Number: IIM-2016-DS0001 |
| The information contained herein is the property of Wizlearn Technologies Pte Ltd and may not be copied, used or disclosed in whole or in part to any third party except with written approval of Wizlearn Technologies Pte Ltd or, if it has been authorised under a contract. |

Table of Contents

[1 Introduction 4](#_Toc467158740)

[1.1 Purpose 4](#_Toc467158741)

[1.2 Distribution List 4](#_Toc467158742)

[2 Scope 5](#_Toc467158743)

[2.1 Definition, Acronyms and Abbreviations 5](#_Toc467158744)

[2.1.1 Acronyms 5](#_Toc467158745)

[2.1.2 Definitions 5](#_Toc467158746)

[2.1.3 Abbreviations 5](#_Toc467158747)

[3 Overview of Web-Services 5](#_Toc467158748)

[3.1 Workflow 5](#_Toc467158749)

[3.2 Assumptions and Dependencies 5](#_Toc467158750)

[4 Web-Services 6](#_Toc467158751)

[4.1 1. Create User Web-Services (For New Registration via IIM webpage) 6](#_Toc467158752)

[4.1.1 Create User Web-Services Response Status 7](#_Toc467158753)

[4.1.2 Execution Logic: 7](#_Toc467158754)

[4.1.3 Exception: 7](#_Toc467158755)

[4.1.4 Create User Request API End Point Format 7](#_Toc467158756)

[4.1.5 Status Messages 8](#_Toc467158757)

[4.2 2. Validate User (Login via IIM webpage) 9](#_Toc467158758)

[4.2.1 Validate User Fail Response Status 9](#_Toc467158759)

[4.2.2 Execution Logic: 9](#_Toc467158760)

[4.2.3 Exception: 10](#_Toc467158761)

[4.2.4 Validate User Request API End Point Format 10](#_Toc467158762)

[4.2.5 Status Messages 10](#_Toc467158763)

[4.3 3. Course Catalogue 11](#_Toc467158764)

[4.3.1 LMS to IIM 11](#_Toc467158765)

[4.3.2 Execution Logic: 12](#_Toc467158766)

[4.3.3 Exception: 12](#_Toc467158767)

[4.3.4 Course Catalogue Request API End Point Format 12](#_Toc467158768)

[4.3.5 Status Messages 13](#_Toc467158769)

[4.4 4. Assign Course 13](#_Toc467158770)

[4.4.1 Assign Course Response Status 14](#_Toc467158771)

[4.5 Execution Logic: 14](#_Toc467158772)

[4.5.1 Exception: 14](#_Toc467158773)

[4.5.2 Assign Course Request Api Endpoint Format 14](#_Toc467158774)

[4.5.3 Status Messages 15](#_Toc467158775)

[4.6 5. Learner Results 16](#_Toc467158776)

[4.6.1 Received Data Results Status 16](#_Toc467158777)

[4.6.2 Execution Logic: 16](#_Toc467158778)

[4.7 6. Sync Password 16](#_Toc467158779)

[4.7.1 Sync Password Response Status 16](#_Toc467158780)

[4.7.2 Execution Logic: 17](#_Toc467158781)

[4.7.3 Exception: 17](#_Toc467158782)

[4.7.4 Sync Password Request Api Endpoint Format 17](#_Toc467158783)

[4.7.5 Status Messages 17](#_Toc467158784)

[5 API Over All Status Messages 19](#_Toc467158785)

[5.1 Api Endpoint Encryption and Decryption Info 20](#_Toc467158786)

[5.2 Api Endpoint Sample Encryption and Decryption 20](#_Toc467158787)

[5.3 Sample API SOAP 1.1 21](#_Toc467158788)

[5.4 HTTP GET 22](#_Toc467158789)

[5.5 HTTP POST 22](#_Toc467158790)

Version History

|  |  |  |  |
| --- | --- | --- | --- |
| Version Number | Date of Issue | Author(s) | Brief Description of Change |
| 0.1 | 6 Oct 2016 | Tin Win | Initial release of version |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

The purpose of this design specification is to provide precise technical requirements definition of the IIM eLearning Management System web-services to IIM.

## Distribution List

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Prepared by** | |  |  |  |  |  | |  |  |  |
|  |  |  |  |  |  |  | |  |  |  |
| **Name** | | | **Company** | | **Project Role** | | **Signature** | | **Date** | |
| Tin Win | | | Wizlearn | | Developer | |  | | 6-Oct-2016 | |
|  | | |  | |  | |  | |  | |
|  | | |  | |  | |  | |  | |
|  |  |  |  |  |  |  | |  |  |  |
|  |  |  |  |  |  |  | |  |  |  |
| **Reviewed by** | |  |  |  |  |  | |  |  |  |
|  |  |  |  |  |  |  | |  |  |  |
| **Name** | | | **Company** | | **Project Role** | | **Signature** | | **Date** | |
| Soh Chze Keong | | | Wizlearn | | Project Manager | |  | |  | |
|  | | |  | |  | |  | |  | |
|  | | |  | |  | |  | |  | |
|  |  |  |  |  |  |  | |  |  |  |
|  |  |  |  |  |  |  | |  |  |  |
| **Approved by** | |  |  |  |  |  | |  |  |  |
|  |  |  |  |  |  |  | |  |  |  |
| **Name** | | | **Company** | | **Project Role** | | **Signature** | | **Date** | |
|  | | |  | |  | |  | |  | |
|  | | |  | |  | |  | |  | |
|  | | |  | |  | |  | |  | |
|  | | |  | |  | |  | |  | |

# Scope

## Definition, Acronyms and Abbreviations

### Acronyms

|  |  |
| --- | --- |
| **S/No** | **Acronyms** |
| 1. | LMS – Learning Management System |

### Definitions

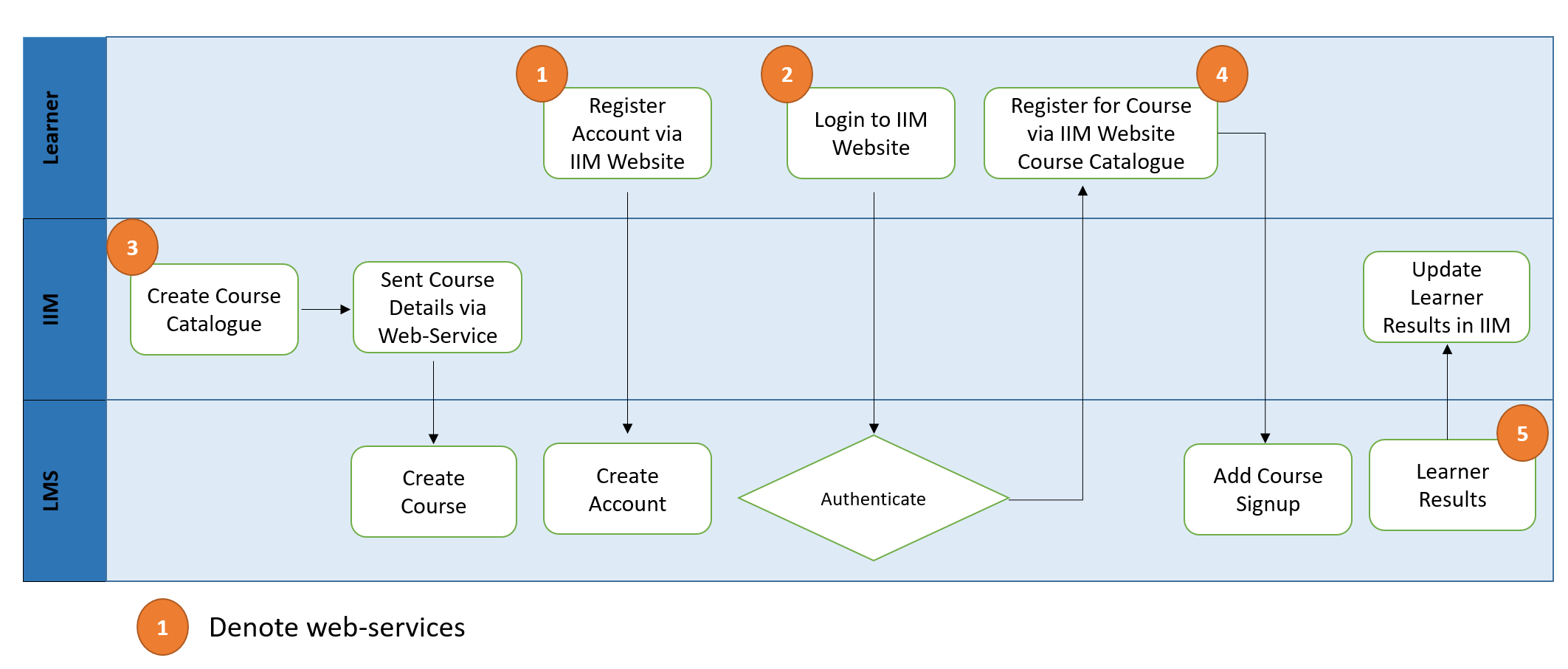
Not Applicable.

### Abbreviations

Not Applicable.

# Overview of Web-Services

## Workflow



## Assumptions and Dependencies

Conditional sign-off is included in this Functional Specification. This only applies for User Interface and Design. Screenshots used in this specs are just for illustrations only and may differ from the actual representation.

# Web-Services

IIM System

Wizlearn LMS

1. Create User

3. Course Catalogue

4. Assign Course

2. Validate User

5. Course Results

5. Sync Password

## 1. Create User Web-Services (For New Registration via IIM webpage)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S/N** | **Fields** | **Flow** | **Data Type** | **Required** | **Remarks** |
| 1 | User\_ID | IIM to LMS | String (150) | Y |  |
| 2 | User\_Name | IIM to LMS | String (150) | Y |  |
| 3 | Group\_ID | IIM to LMS | String (255) | Y | 1 level group if not found -> Return Exception.  \* No space and minimum number characters is 3 for Group Id |
| 4 | Group\_Name | IIM to LMS | String (255) | Y |  |
| 5 | Email | IIM to LMS | String (255) | N |  |
| 6 | Request\_Date | IIM to LMS | Date Time  Format:(ddMMyyyyHHmm) | Y | Current Date |

Note:

1. Default LMS to trigger email for new account creation, do we need to CC the email to admin? **NOPE**

[Wiz, 17112016] To confirm we will still trigger email to user and these email will not be cc to admin.

1. API to be provided by LMS

### Create User Web-Services Response Status

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S/N** | **Fields** | **Flow** | **Data Type** | **Remarks** |
| 1 | Response Status | LMS to IIM | String (4) | Success or Fail  Message Status |

### Execution Logic:

1. Check Requested date must be current date and date format must correct.
2. Check if group exist in LMS, if group exist, will continue else throw error status.
3. Checks if IIM user ID (User ID) exists in LMS system. If user does not exist, create new user account
4. If user already exists, then update user’s email and user’s name.
5. Assign user to the group
6. if user already created and assigned to group already then send email to user and Cc to admin (if applicable).
7. Finally pass success message status to IIM

**Note**: Either user email or admin email should not be empty to send email.

### Exception:

Information pass from IIM

1. User ID and Group ID cannot contain space.
2. User Name and User ID cannot be less than 3 and cannot be more than 150.
3. User Group id and Group name length cannot be less than 3 and cannot be more than 150.
4. User Group should already be existing in LMS before user register. (Note: Group path will define by LMS).
5. If user email is empty, user will not receive email.

### Create User Request API End Point Format

|  |  |  |
| --- | --- | --- |
| **No.** | **Name** | **Value** |
| 1 | Request Api Endpoint Format | http://{Domain}/{App\_Name}/webservice/api/IMM/IIMWebService.asmx/IIM\_LMS\_Register\_User?Key={7a5bc }&param={encryptedstring} |
| 2 | Parameter Without Encryption | userid=DemoUser1&username=Demo User [1&groupid=IIM&groupname=IIM&email=noreply@wizlearn.com&requestdate=061020160907](mailto:1&groupid=IIM&groupname=IIM&email=noreply@wizlearn.com&requestdate=061020160907) |
| 3 | Parameter (AES256 Encryption) | MPFdRajFWybYWCbl2paAdy+XOkJ0yikhDZK3OGd9cZvspLdUFDYu8HcvtzPVQPIp+BzdksrR6zz9FPuRQ05Fe9vDQBJccOlctREBgZSOGas+DLD/rOoUHzKGRlKGpxDpJzZTVNWq2G2NDP3gEhHUIF/3P8y4+NOzDBRa9wSD16k= |
| 4 | Parameter with Url Encode | MPFdRajFWybYWCbl2paAdy%2bXOkJ0yikhDZK3OGd9cZvspLdUFDYu8HcvtzPVQPIp%2bBzdksrR6zz9FPuRQ05Fe9vDQBJccOlctREBgZSOGas%2bDLD%2frOoUHzKGRlKGpxDpJzZTVNWq2G2NDP3gEhHUIF%2f3P8y4%2bNOzDBRa9wSD16k%3d |
| 5 | API End Point (Url Encode) | http://{Domain}/{App\_Name}/Webservice/Api/IIM/IIMWebService.asmx/IIM\_LMS\_Register\_User?Key=T7D+IbdKOBfF3zd+LT+cYGDFhQ4YcHU2&param=MPFdRajFWybYWCbl2paAdy%2bXOkJ0yikhDZK3OGd9cZvspLdUFDYu8HcvtzPVQPIp%2bBzdksrR6zz9FPuRQ05Fe9vDQBJccOlctREBgZSOGas%2bDLD%2frOoUHzKGRlKGpxDpJzZTVNWq2G2NDP3gEhHUIF%2f3P8y4%2bNOzDBRa9wSD16k%3d |

### Status Messages

The tables below list the possible messages that may be encountered during web service request.

|  |  |  |
| --- | --- | --- |
| **No**. | **Message Code** | **Description** |
| 1 | 1000 | An invalid parameter, parameters cannot decrypt. |
| 2 | 1001 | An invalid parameters length, parameters length is incorrect what expected for current Api. |
| 3 | 1002 | An invalid parameter request date, request date parameter is not equal to current date. |
| 4 | 1003 | An invalid group id parameter, group id cannot contain space |
| 5 | 1004 | An invalid group id parameter, group id cannot be less than 3 characters. |
| 6 | 1005 | An invalid group name parameter, group name cannot be less than 3 characters. |
| 7 | 1007 | An invalid user id parameter, User Id cannot contain space. |
| 8 | 1008 | An invalid user id parameter, User Id length cannot be less than 3. |
| 9 | 1009 | An invalid user id parameter, User Id length cannot be more than 150. |
| 10 | 1010 | An invalid username parameter, cannot be less than 3 characters. |
| 11 | 1011 | An invalid username parameter, cannot be more than 150. |
| 12 | 1025 | An invalid AES Key, AES key is not same as LMS AES Key to decrypt. |
| 13 | 1026 | An invalid group Id parameter, group id cannot be more than 255 characters. |
| 14 | 1027 | An invalid group name parameter, group name cannot be more than 255 characters. |
| 15 | 1028 | An invalid group parameter, multiple group found in system  (Note: Lms will handle this part.) |
| 16 | 2029 | An invalid group id, group Id not correct unique identifier format.  (Note: Lms will handle this part.) |
| 17 | 1111 | Exception error detail. |
| 18 | 2000 | Request is successful. |

## 2. Validate User (Login via IIM webpage)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S/N** | **Fields** | **Flow** | **Data Type** | **Required** | **Remarks** |
| 1 | User ID | IIM to LMS | String (150) | Y | Encrypted |
| 2 | Password | IIM to LMS | String (150) | Y | Encrypted |
| 3 | CourseId | IIM to LMS | Integer | Y | Encrypted |
| 4 | Request\_Date | IIM to LMS | Date Time  Format:(ddMMyyyyHHmm) | Y | Current Date |

Note:

1. IIM and LMS will store own password and user id, how do we synchronize these information? Might new another web-service to cater for IIM password reset and disable LMS password reset.
2. Course Id cannot be empty.
3. API to be provided by LMS.

### Validate User Fail Response Status

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S/N** | **Fields** | **Flow** | **Data Type** | **Remarks** |
| 1 | Response Status | LMS to IIM | String (4) | Fail  Message Status  (if pass, redirect to LMS=> My Task) |

### Execution Logic:

1. Check requested date is equal to current date (server), if requested date is current date then procced to another step.
2. If requested date is not equal to current date, return error message status.
3. Checks if user ID (User ID) exists in LMS system. If user does not exist, return error message status.
4. If user Id exists, then validate authentication with user’s email and user’s password in LMS.
5. If user authenticated, then login to LMS and redirect to LMS My Task Page.

**[IIM, 17112016] Can we pass the redirect page to LMS so that we can specify the URL we want to?**

**[WIZ, 17112016] If you provide URL = www.yahoo.com.sg , we will just direct to www.yahoo.com.sg???**

**What we understand is that IIM will have a list of all courses that the users will click and upon clicking, it will call this API, LMS will validate that he/she is valid user to the course and LMS will navigate to the course page.**

1. If user not authenticated, will return error message status

### Exception:

Information pass from IIM

1. Requested date must be current date and date format must be correct.
2. User ID and Password cannot be empty.
3. User ID must exist in system.
4. User credential must be correct.

### Validate User Request API End Point Format

|  |  |  |
| --- | --- | --- |
| **No.** | **Name** | **Value** |
| 1 | Request Api Endpoint Format | http://{Domain}/{App\_Name}/SSO/IIM/SSO.aspx? Key={7a5bc }&param={encryptedstring} |
| 2 | Parameter Without Encryption | userid=DemoUser1&password=DemoUserPwd1&courseid=34&requestdate=061020161209 |
| 3 | Parameter (AES256 Encryption) | MPFdRajFWybYWCbl2paAd9peUuszruQiyPO425A2oIk2L5TJ5HO/9UhHPLNtPF12+HGXApg6MJgq/ghv2L5GIw== |
| 4 | Parameter (Url Encode) | MPFdRajFWybYWCbl2paAd9peUuszruQiyPO425A2oIk2L5TJ5HO%2f9UhHPLNtPF12%2bHGXApg6MJgq%2fghv2L5GIw%3d%3d |
| 5 | API End Point (Url Encode) | http://{domain}.com/{App\_Name}/SSO/IIM/SSO.aspx?Key=T7D+IbdKOBfF3zd+LT+cYGDFhQ4YcHU2&param=MPFdRajFWybYWCbl2paAd9peUuszruQiyPO425A2oIk2L5TJ5HO%2f9UhHPLNtPF12%2bHGXApg6MJgq%2fghv2L5GIw%3d%3d |

### Status Messages

The tables below list the possible messages that may be encountered during web service request.

|  |  |  |
| --- | --- | --- |
| **No**. | **Message Code** | **Description** |
| 1 | 1000 | An invalid parameter, parameters cannot decrypt. |
| 2 | 1001 | An invalid parameters length, parameters length is incorrect what expected for current Api. |
| 3 | 1002 | An invalid parameter request date, request date parameter is not equal to current date. |
| 4 | 1017 | An incorrect credential parameter, User Account does not exist. |
| 5 | 1018 | An incorrect credential parameter, please check back User ID or Password. |
| 6 | 1019 | An incorrect credential parameter, Password cannot be empty. |
| 8 | 1013 | An invalid course id parameter, course id does not exist. |
| 9 | 1031 | An invalid course id, course id must be integer. |
| 10 | 1012 | An invalid course id parameter, course id cannot be 0. |
| 11 | 1033 | An invalid course id, course id cannot be empty. |
| 12 | 1034 | An invalid user to access course, user does not have permission to access this course. |
| 13 | 1024 | An incorrect user id, user id cannot be empty. |
| 14 | 1111 | Exception error detail. |

## 3. Course Catalogue

IIM to LMS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S/N** | **Fields** | **Flow** | **Data Type** | **Required** | **Remarks** |
| 1 | IIMCourseID | IIM to LMS | Integer | Y | Pass IIM Course ID to LMS |
| 2 | Subject Name | IIM to LMS | String (150) | Y | Get Subject Title from LMS |
| 3 | Course\_ID | IIM to LMS  LMS to IIM | Integer | Y | LMS Course\_ID is auto incremented and cannot be passed over from IIM  Pass Course ID Value as 0 to create new course.  To edit course, course id value required. |
| 4 | Course\_Title | IIM to LMS | String (255) | Y |  |
| 5 | Course\_Desc | IIM to LMS | String (500) | Y |  |
| 6 | Request\_Date | IIM to LMS | Date Time  Format:(ddMMyyyyHHmm) | Y | Current Date |

Note:

1. If Course\_ID is found in LMS, update Course\_Title and Course\_Desc
2. If Course\_ID is not found in LMS, create new course
3. API will be provided by LMS
4. Subject name can get from Lms course setup page.
5. IIM need to send iim course id to LMS

**[IIM, 17112016] Course Catalogue supposed to be a service that return all courses that User ID has been enrolled, not one specific course**

**[WIZ, 171102016] The purpose of this API is to pass the course information created in IIM over to LMS and we will pass back the course ID as course ID is auto generated in our system. When user enroll for a course, this will be cover by Assign Course API which will pass LMS course ID that the user has registered, e.g. when user enroll course A on 11/11/2016 via IIM, IIM will call Assign Course API and we will save this information in LMS and user could login to attempt the course within on the course start and end date. When the user enroll for another course B on 15/11/2016, we will capture this information when Assign Course API is triggered by IIM. Upon completion of the course A, LMS will call IIM API to push over the course A results attempted by the user.**

**If you need to have all the course enrolled by a user we will need to add another web-service which is not really necessary as this inform is already available in your system when user enroll and at our LMS when Assign Course API is called.**

### LMS to IIM

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S/N** | **Fields** | **Flow** | **Data Type** | **Required** | **Remarks** |
| **1** | IIMCourseID | LMS to IIM | Integer |  | Return back iim course id |
| 1 | Course\_ID | LMS to IIM | String (255) |  | LMS Course\_ID is auto incremented. |
| 2 | Course\_Title | LMS to IIM | String (255) |  | iim course title |
| 3 | Course\_Desc | LMS to IIM | String (500) |  | iim course description |
| 4 | Request\_Date | IIM to LMS | Date Time  Format:(ddMMyyyyHHmm) |  | Current Date |
| 5 | Status | IIM to LMS | String (50) | Y | Api endpoint response status |

Note:

1. API will be provided by LMS

### Execution Logic:

1. Check if Curse ID is 0, then continue next step to create new course.
2. If Course ID is not 0, then update existing Course Title and Course Description based on Course ID.
3. Check if Course Title length is less than 10 or more than 255, return error status message.
4. After new Course already created or Existing Course already updated in LMS, then LMS will return the Success Status Message including Course ID, Course Title, Course Description and Request Date.
5. Check if subject name is empty or not exist in lms, will return error status code. If subject name exist in lms, will continue to create course.

### Exception:

Information pass from IIM

1. Requested date must be current date and date format must be correct.
2. Course Title cannot be empty.
3. Subject name cannot be empty.

### Course Catalogue Request API End Point Format

|  |  |  |
| --- | --- | --- |
| **No.** | **Name** | **Value** |
| 1 | Request Api Endpoint Format | http://{Domain}/{App\_Name}/webservice/api/IMM/IIMWebService.asmx/IIM\_LMS\_Register\_CourseCatalog?Key={7a5bc }&param={encryptedstring} |
| 2 | Parameter Without Encryption | iimcourseid=23&subjectname=General&courseid=20&coursetitle=Create a new course for Api test&coursedescription=Create a description for course&requestdate=061020161106 |
| 3 | Parameter (AES256 Encryption) | PAhgYvFDkcHog5H8wf42NLXbRHEiefsoBJ28xDwyMwke/eQElmVAl7NRUpuXrPVIjltKNYj5aXJNbbFDVv/7DznOoQ/kAW7nws/TcOkC3wWTTQ85doMxGR0Zypbf280z/iwD3S3NMD8t+QtWfhTVePiname4AGzm/EgRQctsS/RCh1RDHJ/eBrl/xMbx2DNHKOD6wKIaV7Hd8Jyrtt3enoPGfkheKn8KxqyRW5jwsns= |
| 4 | Parameter (Url Encode) | PAhgYvFDkcHog5H8wf42NLXbRHEiefsoBJ28xDwyMwke%2feQElmVAl7NRUpuXrPVIjltKNYj5aXJNbbFDVv%2f7DznOoQ%2fkAW7nws%2fTcOkC3wWTTQ85doMxGR0Zypbf280z%2fiwD3S3NMD8t%2bQtWfhTVePiname4AGzm%2fEgRQctsS%2fRCh1RDHJ%2feBrl%2fxMbx2DNHKOD6wKIaV7Hd8Jyrtt3enoPGfkheKn8KxqyRW5jwsns%3d |
| 5 | API End Point (Url Encode) | http://{Domain}/{App\_Name}/webservice/Api/IIM/IIMWebService.asmx/IIM\_LMS\_Register\_CourseCatalog?Key=T7D+IbdKOBfF3zd+LT+cYGDFhQ4YcHU2&param=PAhgYvFDkcHog5H8wf42NLXbRHEiefsoBJ28xDwyMwke%2feQElmVAl7NRUpuXrPVIjltKNYj5aXJNbbFDVv%2f7DznOoQ%2fkAW7nws%2fTcOkC3wWTTQ85doMxGR0Zypbf280z%2fiwD3S3NMD8t%2bQtWfhTVePiname4AGzm%2fEgRQctsS%2fRCh1RDHJ%2feBrl%2fxMbx2DNHKOD6wKIaV7Hd8Jyrtt3enoPGfkheKn8KxqyRW5jwsns%3d |

### Status Messages

The tables below list the possible messages that may be encountered during web service request.

|  |  |  |
| --- | --- | --- |
| **No**. | **Message Code** | **Description** |
| 1 | 1000 | An invalid parameter, parameters cannot decrypt. |
| 2 | 1001 | An invalid parameters length, parameters length is incorrect what expected for current Api. |
| 3 | 1002 | An invalid parameter request date, request date parameter is not equal to current date. |
| 4 | 1013 | An invalid course id parameter, course id does not exist. |
| 5 | 1015 | An invalid course title parameter, cannot be less than 10 characters. |
| 6 | 1016 | An invalid course title parameter, cannot be more than 255 characters. |
| 7 | 1025 | An invalid AES Key, AES key is not same as LMS AES Key to decrypt. |
| 8 | 1030 | An invalid subject title, subject does not exist in system. |
| 9 | 1031 | An invalid course id, course id must be integer. |
| 10 | 1032 | An invalid iim course id, iim course id cannot be empty. |
| 11 | 1111 | Exception error detail. |
| 12 | 2000 | Request is successful. |

## 4. Assign Course

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S/N** | **Fields** | **Flow** | **Data Type** | **Required** | **Remarks** |
| 1 | User\_ID | IIM to LMS | String (150) | Y |  |
| 2 | Course\_ID | IIM to LMS | Integer | Y |  |
| 3 | Course\_Start\_Date | IIM to LMS | Date | Y | Format: DD/MM/YYYY  Example: 26/12/2016 |
| 4 | Course\_End\_Date | IIM to LMS | Date | Y | Format: DD/MM/YYYY  Example: 26/12/2016 |
| 5 | Request\_Date | IIM to LMS | Date Time  Format:(ddMMyyyyHHmm) | Y | Current Date |

Note:

1. Learner will attempt registered course via LMS link found in IIM. Upon click, LMS>My Task page will be displayed.
2. API will be provided by LMS

### Assign Course Response Status

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S/N** | **Fields** | **Flow** | **Data Type** | **Remarks** |
| 1 | Response Status | LMS to IIM | String (4) | Success or Fail  Message Status |

## Execution Logic:

1. Check if User ID is less than 3 or more than 150, then return error status message.
2. Check if User ID exists in LMS, if not exist, then return error status message.
3. Check if Course ID is less than 3 or more than 100, then return error status message.
4. Check if Course ID exists in LMS, if not exist, then return error status message.
5. Check if Course End Date cannot be less than current date.
6. If User ID, Course ID and Course End Date is valid, then assign user to Course.

### Exception:

Information pass from IIM

1. Requested date must be current date and date format must be correct.
2. Course ID, User ID cannot be empty.
3. Course ID and User ID must exist in LMS
4. Course End Date must equal or greater than current date.

### Assign Course Request Api Endpoint Format

|  |  |  |
| --- | --- | --- |
| **No.** | **Name** | **Value** |
| 1 | Request Api Endpoint Format | http://{Domain}/{App\_Name}/webservice/api/IMM/IIMWebService.asmx/IIM\_LMS\_Assign\_Course?Key={7a5bc }&param={encryptedstring} |
| 2 | Parameter Without Encryption | userid=DemoUser1&courseid=20&coursestartdate=06/10/2016&courseenddate=06/10/2016&requestdate=061020161145 |
| 3 | Parameter (AES256 Encryption) | MPFdRajFWybYWCbl2paAdzEZCLO5sCtRo5H1+U1gVkZ/FhYH/y4QzBDuzbW4ZLuCSEVhgBgFX81cqWnY4380PUU3ZE6lsGpArPIrUEOphWXV+qq6dTJlFQLEU++8DFJucZR+uRUsnSeeBIcOTtre3Q== |
| 4 | Parameter (Url Encode) | MPFdRajFWybYWCbl2paAdzEZCLO5sCtRo5H1%2bU1gVkZ%2fFhYH%2fy4QzBDuzbW4ZLuCSEVhgBgFX81cqWnY4380PUU3ZE6lsGpArPIrUEOphWXV%2bqq6dTJlFQLEU%2b%2b8DFJucZR%2buRUsnSeeBIcOTtre3Q%3d%3d |
| 5 | API End Point (Url Encode) | http://{Domain}/{App\_Name}/Webservice/Api/IIM/IIMWebService.asmx/IIM\_LMS\_Assign\_Course?Key=T7D+IbdKOBfF3zd+LT+cYGDFhQ4YcHU2&param=MPFdRajFWybYWCbl2paAdzEZCLO5sCtRo5H1%2bU1gVkZ%2fFhYH%2fy4QzBDuzbW4ZLuCSEVhgBgFX81cqWnY4380PUU3ZE6lsGpArPIrUEOphWXV%2bqq6dTJlFQLEU%2b%2b8DFJucZR%2buRUsnSeeBIcOTtre3Q%3d%3d |

### Status Messages

The tables below list the possible messages that may be encountered during web service request.

|  |  |  |
| --- | --- | --- |
| **No**. | **Message Code** | **Description** |
| 1 | 1000 | An invalid parameter, parameters cannot decrypt. |
| 2 | 1001 | An invalid parameters length, parameters length is incorrect what expected for current Api. |
| 3 | 1002 | An invalid parameter request date, request date parameter is not equal to current date. |
| 13 | 1012 | An invalid course id parameter, course id cannot be 0. |
| 14 | 1013 | An invalid course id parameter, course id does not exist. |
| 15 | 1014 | An invalid course end date, course end date must equal or later than today date. |
| 16 | 1020 | An incorrect course start date, course start date cannot be empty. |
| 17 | 1021 | An incorrect course end date, course end date cannot be empty. |
| 18 | 1022 | An incorrect course start date format, course start date incorrect format. |
| 19 | 1023 | An incorrect course end date format, course end date incorrect format. |
| 20 | 1111 | Exception error detail. |
| 21 | 2000 | Request is successful. |

## 5. Learner Results

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S/N** | **Fields** | **Flow** | **Data Type** | **Required** | **Remarks** |
| 1 | User\_ID | LMS to IIM | String (150) | Y |  |
| 2 | Course\_ID | LMS to IIM | Integer | Y |  |
| 3 | Complete\_Date | LMS to IIM | Date | Y | Format: DD/MM/YYYY  Example: 26/12/2016 |
| 4 | ScormScore | LMS to IIM | String(100) | Y | Pass lms user scorm score to  IIM |
| 4 | Request\_Date | IIM to LMS | Date Time  Format:(ddMMyyyyHHmm) | Y | Current Date |

### Received Data Results Status

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S/N** | **Fields** | **Flow** | **Data Type** | **Remarks** |
| 1 | Response Status | IIM to LMS | String (4) | Success or Fail  Message Status |

Note:

1. This web-services will be trigger whenever learner complete a course.
2. API will be provided by IIM
3. \*\*This implementation will only apply to one scorm package with one result.
   * If scorm contains multiple scorm scores, system will return (-)
   * If course contains multiple scorms, system will return first scorm score
   * If course does not have scorm, system will return (-)
4. \*\* scorm package must contain cmi key of ‘cmi.core.score.raw’, lms will catch this value as scorm score.

### Execution Logic:

1. Requested date must be current date and date format must be correct.
2. LMS will call IIM API whenever user complete the course.
3. LMS will pass back User ID, Course ID, Completion Date.
4. LMS will receive API Request Response Status from IIM.

## 6. Sync Password

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S/N** | **Fields** | **Flow** | **Data Type** | **Required** | **Remarks** |
| 1 | User\_ID | IIM to LMS | String (100) | Y |  |
| 2 | Password | IIM to LMS | String (255) | Y |  |
| 4 | Request\_Date | IIM to LMS | Date Time  Format:(ddMMyyyyHHmm) | Y | Current Date |

### Sync Password Response Status

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S/N** | **Fields** | **Flow** | **Data Type** | **Remarks** |
| 1 | Response Status | LMS to IIM | String (4) | Success or Fail  Message Status |

Note:

1. This web-services will always updated User Password if user exists in lms
2. API will be provided by LMS
3. Only student and teacher roles are allowed to update password.

### Execution Logic:

1. Requested date must be current date and date format must be correct.
2. IIM will call LMS API when user need to update password in lms system.
3. Check user exists in lms by user id, if user exists in lms, then will update new password in lms else if user not exists, then return user not exist message status
4. IIM will receive response message status which is pass or fail in updating user password in LMS

### Exception:

Information pass from IIM

1. Requested date must be current date and date format must be correct.
2. User ID must be existing in lms.

### Sync Password Request Api Endpoint Format

|  |  |  |
| --- | --- | --- |
| 1 | Request Api Endpoint Format | http://{Domain}/{App\_Name}/webservice/api/IMM/IIMWebService.asmx/IIM\_LMS\_SyncPassword?Key={7a5bc }&param={encryptedstring} |
| 2 | Parameter Without Encryption | userid=User1&password=iim$123@&requestdate=161120161240 |
| 3 | Parameter (AES256 Encryption) | POHT7FnFCrUwoNizM+UZ0hrzOBf3OVRWhmU4bv9E/NQ84uIU4D8uQbQzj3TBXhMhwAmxkL7Do3Ko281fK0UaxQ== |
| 4 | Parameter (Url Encode) | POHT7FnFCrUwoNizM%2bUZ0hrzOBf3OVRWhmU4bv9E%2fNQ84uIU4D8uQbQzj3TBXhMhwAmxkL7Do3Ko281fK0UaxQ%3d%3d |
| 5 | API End Point (Url Encode) | http://{Domain}/{App\_Name}/Webservice/api/IIM/IIMWebService.asmx/IIM\_LMS\_SyncPassword?Key=T7D+IbdKOBfF3zd+LT+cYGDFhQ4YcHU2&param=POHT7FnFCrUwoNizM%2bUZ0hrzOBf3OVRWhmU4bv9E%2fNQ84uIU4D8uQbQzj3TBXhMhwAmxkL7Do3Ko281fK0UaxQ%3d%3d |

### Status Messages

The tables below list the possible messages that may be encountered during web service request.

|  |  |  |
| --- | --- | --- |
| **No**. | **Message Code** | **Description** |
| 1 | 1000 | An invalid parameter, parameters cannot decrypt. |
| 2 | 1001 | An invalid parameters length, parameters length is incorrect what expected for current Api. |
| 3 | 1002 | An invalid parameter request date, request date parameter is not equal to current date. |
| 4 | 1017 | An incorrect credential parameter, user account does not exist. |
| 5 | 1019 | An incorrect credential parameter, password cannot be empty. |
| 6 | 1111 | Exception error detail. |
| 7 | 2000 | Request is successful. |

# API Over All Status Messages

The tables below list the possible messages that may be encountered during web service request.

|  |  |  |
| --- | --- | --- |
| **No**. | **Message Code** | **Description** |
| 1 | 1000 | An invalid parameter, parameters cannot decrypt. |
| 2 | 1001 | An invalid parameters length, parameters length is incorrect what expected for current Api. |
| 3 | 1002 | An invalid parameter request date, request date parameter is not equal to current date. |
| 4 | 1003 | An invalid group id parameter, group id cannot contain space |
| 5 | 1004 | An invalid group id parameter, group id cannot be less than 3 characters. |
| 6 | 1005 | An invalid group name parameter, group name cannot be less than 3 characters. |
| 7 | 1006 | An invalid group parameter, group does not exist. |
| 8 | 1007 | An invalid user id parameter, User Id cannot contain space. |
| 9 | 1008 | An invalid user id parameter, User Id length cannot be less than 3. |
| 10 | 1009 | An invalid user id parameter, User Id length cannot be more than 150. |
| 11 | 1010 | An invalid username parameter, cannot be less than 3 characters. |
| 12 | 1011 | An invalid username parameter, cannot be more than 150. |
| 13 | 1012 | An invalid course id parameter, course id cannot be 0. |
| 14 | 1013 | An invalid course id parameter, course id does not exist. |
| 15 | 1014 | An invalid course end date, course end date must be later that today date. |
| 16 | 1015 | An invalid course title parameter, cannot be less than 10 characters. |
| 17 | 1016 | An invalid course title parameter, cannot be more than 255 characters. |
| 18 | 1017 | An incorrect credential parameter, user account does not exist. |
| 19 | 1018 | An incorrect credential parameter, please check back user name or password. |
| 20 | 1019 | An incorrect credential parameter, password cannot be empty. |
| 21 | 1020 | An incorrect course start date, course start date cannot be empty. |
| 22 | 1021 | An incorrect course end date, course end date cannot be empty. |
| 23 | 1022 | An incorrect course start date format, course start date incorrect format. |
| 24 | 1023 | An incorrect course end date format, course end date incorrect format. |
| 25 | 1024 | An incorrect user id, user id cannot be empty. |
| 26 | 1025 | An invalid AES Key, AES key is not same as LMS AES Key to decrypt. |
| 27 | 1026 | An invalid group Id parameter, group id cannot be more than 255 characters. |
| 28 | 1027 | An invalid group name parameter, group name cannot be more than 255 characters. |
| 29 | 1028 | An invalid group parameter, multiple group found in system |
| 30 | 2029 | An invalid group id, group Id not correct unique identifier format. |
| 31 | 1030 | An invalid subject title, subject does not exist in system. |
| 32 | 1031 | An invalid course id, course id must be integer. |
| 33 | 1032 | An invalid iim course id, iim course id cannot be empty. |
| 34 | 1033 | An invalid course id, course id cannot be empty. |
| 35 | 1034 | An invalid user to access course, user does not have permission to access this course. |
| 36 | 1035 | An invalid parameter key, parameter can not be empty. |
| 37 | 1036 | An invalid request parameter, request parameter session time out. |
| 38 | 1037 | Exception occur, fail to insert user records into user course mapping table. |
| 39 | 1038 | An invalid request user role, Invalid User Role To Update Password. |
| 40 | 1111 | Exception error detail. |
| 41 | 2000 | Request is successful. |

## Api Endpoint Encryption and Decryption Info

|  |  |  |
| --- | --- | --- |
| **No**. | **Name** | **Value** |
| 1 | **Crypto Key** | **T7D+IbdKOBfF3zd+LT+cYGDFhQ4YcHU2** |
| 2 | **IV** | **ABC123DEF456HIJ7** |
| 3 | **Encryption method** | **encrypt (AES256)** |
| 4 | **Decryption method** | **decrypt (AES256)** |
| 5 | **Url Encode method** | **UrlEncode (AES256 Encrypted Text)**  Sample  **data = UrlEncode(AESEncrypt256(encryptdata, KEY\_256, IV)** |
| 6 | **Url Decode method** | **AES Decrypt (UrlDecode(AES256 Encrypted Text) )**  Sample  **data = AESDecrypt256(UrlDecode(decryptdata), KEY\_256, IV)** |

## Api Endpoint Sample Encryption and Decryption

|  |  |  |
| --- | --- | --- |
| **No** | **Name** | **Description** |
| **1** | **Endpoint without encryption** | http://{Domain}/{App\_Name}/webservice/api/IMM/IIMWebService.asmx/IIM\_LMS\_Register\_User?Key={7a5bc }&param={encryptedstring} |
| **2** | **Parameter (AES256 Encryption)** | MPFdRajFWybYWCbl2paAdy+XOkJ0yikhDZK3OGd9cZvspLdUFDYu8HcvtzPVQPIp+BzdksrR6zz9FPuRQ05Fe9vDQBJccOlctREBgZSOGas+DLD/rOoUHzKGRlKGpxDpJzZTVNWq2G2NDP3gEhHUIF/3P8y4+NOzDBRa9wSD16k= |
| 3 | **Encrypted endpoint (Url encode)** | http://{Domain}/{App\_Name}/Webservice/Api/IIM/IIMWebService.asmx/IIM\_LMS\_Register\_User?Key=T7D+IbdKOBfF3zd+LT+cYGDFhQ4YcHU2&param=MPFdRajFWybYWCbl2paAdy%2bXOkJ0yikhDZK3OGd9cZvspLdUFDYu8HcvtzPVQPIp%2bBzdksrR6zz9FPuRQ05Fe9vDQBJccOlctREBgZSOGas%2bDLD%2frOoUHzKGRlKGpxDpJzZTVNWq2G2NDP3gEhHUIF%2f3P8y4%2bNOzDBRa9wSD16k%3d |

## Sample API SOAP 1.1

The following is a sample SOAP 1.1 request and response. The placeholders shown need to be replaced with actual values.

POST /asknlearn/Webservice/Api/IIM/IIMWebService.asmx HTTP/1.1

Host: local.asknlearn

Content-Type: text/xml; charset=utf-8

Content-Length: length

SOAPAction: "http://tempuri.org/IIM\_LMS\_Register\_User"

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

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<IIM\_LMS\_Register\_User xmlns="http://tempuri.org/">

<Key>string</Key>

<param>string</param>

</IIM\_LMS\_Register\_User>

</soap:Body>

</soap:Envelope>

HTTP/1.1 200 OK

Content-Type: text/xml; charset=utf-8

Content-Length: length

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

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<IIM\_LMS\_Register\_UserResponse xmlns="http://tempuri.org/">

<IIM\_LMS\_Register\_UserResult>string</IIM\_LMS\_Register\_UserResult>

</IIM\_LMS\_Register\_UserResponse>

</soap:Body>

</soap:Envelope>

## HTTP GET

The following is a sample HTTP GET request and response. The placeholders shown need to be replaced with actual values.

**GET** /asknlearn/Webservice/Api/IIM/IIMWebService.asmx/IIM\_LMS\_Register\_CourseCatalog?Key=string&param=string HTTP/1.1

Host: local.asknlearn

HTTP/1.1 200 OK

Content-Type: text/xml; charset=utf-8

Content-Length: length

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

<string xmlns="http://tempuri.org/">string</string>

## HTTP POST

The following is a sample HTTP POST request and response. The placeholders shown need to be replaced with actual values.

**POST** /asknlearn/Webservice/Api/IIM/IIMWebService.asmx/IIM\_LMS\_Register\_CourseCatalog HTTP/1.1

Host: local.asknlearn

Content-Type: application/x-www-form-urlencoded

Content-Length: length

Key=string&param=string

HTTP/1.1 200 OK

Content-Type: text/xml; charset=utf-8

Content-Length: length

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

<string xmlns="http://tempuri.org/">string</string>