LearningZen

API Technical Specifications

LEARNINGZEN.COM

February 27, 2026

Contents

Acknowledgements

Document Development Team

LearningZen Support Team

Copyright Information

Trademarks

Disclaimer

Every effort has been made to ensure the accuracy of the information included in this training guide. However, this training guide is provided without any warranty whatsoever, either expressed or implied, including but not limited to the contents of this guide, its marketability, or fitness for any particular purpose. Neither the authors nor anyone else involved with the development, production or delivery of this material shall be liable for any reason.

Contact Information

LearningZen Gold Systems, Inc.
1240 E 2100 S Ste 102
Salt Lake City, UT 84106

(877) 850-1214
support@learningzen.com

Overview

An API (Application Programming Interface) is a mechanism for two software systems to interact. The LearningZen API allows your system to access the data stored inside LearningZen. It provides mechanisms to manage your portal as well as ways to get course and student data for your portal.

Abbreviations

- LZ = LearningZen
- TP = Third-Party (user of the LZ API)

Authentication

LearningZen supports two methods of authentication:

- Secret Token: A secret token that is passed with each API call that identifies the caller. You can choose an API token or LearningZen can assign one to you.
- Single-Use, Time-Based Token: An initial handshake sequence produces a single-use token that can be used for only one subsequent API method call. The token must be used within one hour of being verified. The handshake works as follows:
    - Make a GET to `https://[YOURPORTAL].learningzen.com/api/wake?token=[TOKEN]` where the token is a previously unused token generated by the TP system.
    - LZ will verify the token via a GET to the TP API: `http://[TP-API]/verify?token=[TOKEN]`
    - If LZ receives verification from the TP that the token is valid, and if the token has not been previously used in an LZ method call, LZ will respond with a success to the initial /wake method call.
    - Detailed information about the handshake can be found in the API Methods section below. To use this authentication method, the TP must implement the /verify method and provide LZ with the base URL for their API.

Testing Environment

LearningZen has a test environment that, once configured for your portal, is available for integration testing. Please substitute lzlogin.com for learningzen.com in all API calls described in this document.

Calling a Method

LZ supports multiple ways to call API methods. API users can make method calls either by making a GET and passing URL parameters or by POSTing XML.

Via GET

Make a GET to:

```
https://[YOURPORTAL].learningzen.com/api/[API METHOD]?token=abcdef123456&[METHOD-SPECIFIC PARAMETERS]
```

The response:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
       <success>1</success>
       [ANY API METHOD-SPECIFIC ELEMENTS]
</response>
```

By POSTing XML

Make a POST to:

```
https://[YOURPORTAL].learningzen.com/api/[API METHOD]
```

The POST should contain:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
       <token>abcdef123456</token>
       <sourceIP>127.0.0.1</sourceIP>
       <portalHost>thirdpartysandbox</portalHost>
       [ANY API METHOD-SPECIFIC ELEMENTS]
</request>
```

The response:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
       <success>1</success>
       [ANY API METHOD-SPECIFIC ELEMENTS]
</response>
```

Data Format

Data Requirements

Each request initiated by LearningZen will contain common elements.

Name	Type	Optional	Notes
token	string	N	The security/session token passed for each request, this can be a single-use, time-based token or a secret token, depending on which method of authentication is used.
sourceIP	IP address	Y	The IP address of the originating user
portalHost	string	Y (if using portal-specific URL)	The LearningZen portal hostname

Responses from the LearningZen API will also contain common elements.

Name	Type	Optional	Notes
success	bit	N	"1" if the request was successful; "0" if it failed.
Message	string	Y	Additional information, usually when success is "0". Child of "messages" element. More than one message possible.

Data Format

When calling an API method, the request can be POSTed as XML or the parameters can be passed as URL parameters. Responses are always XML. Following is an example of the common elements sent as XML when a TP initiates a request.

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
       <token>U38Neqk0lqz4Cx</token>
       <sourceIP>127.0.0.1</sourceIP>
       <portalHost>sandbox</portalHost>
</request>
```

The response from the LZ API with only the common elements would be formatted as follows:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
       <success>1</success>
<messages>
              <message>Success</message>
              <message>Additional message</message>
       </messages>
</response>
```

API Methods

Method names are case-insensitive. When calling via GET, URL parameters are case-insensitive.

wake

Only used with the Single-Use, Time-Based Token Authentication method. Registers a single-use, time-based token to be used in a subsequent LZ API call. The given token will be verified via a GET to the TP API: **https://[TP-API]/verify?token=[TOKEN]**. If LZ receives verification from the TP that the token is valid, and if the token has not been previously used in an LZ method call, LZ will respond with a success. The response from the /verify call should look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>

<response>
        <success>1</success>
        <validToken>1</validToken>
        <token>[TOKEN]</token>
</response>
```

URL example: https://[YOURPORTAL].learningzen.com/api/wake

Restrictions: Token must not be null or greater than 256 characters. Token must be alphanumeric, and contain no special characters other than dash (-) and underscore (_).

Request example:

Via URL Parameters:

```
/wake?token=abcdef123456
```

Via POST:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
        <token>abcdef123456</token>
        <sourceIP>127.0.0.1</sourceIP>
        <portalHost>thirdpartysandbox</portalHost>
</request>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
        <success>1</success>
        <validToken>1</validToken>
        <token>abcdef123456</token>
</response>
```

Name	**Type**	**Notes**
validToken	bit	1 (yes) or 0 (no)
token	string	The token specified in the request

isRegistered

Check to see if a user is registered within LearningZen.

URL example: https://[YOURPORTAL].learningzen.com/api/isRegistered

Request example:

Via URL Parameters:

```
/isRegistered?token=abcdef123456&accountID= test123456
```

Via POST:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
        <token>abcdef123456</token>
        <sourceIP>127.0.0.1</sourceIP>
        <portalHost>thirdpartysandbox</portalHost>
        <accountID> test123456</accountID>
</request>
```

Name	Type	Notes
accountID	string	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student. **NOTE**: the AccountID value should be a unique and persistent identifier value such as DBID, GUID, Employee number etc. You should never use an email address or any other value that may change in the future.

Response examples:

Example 1 - AccountID does not exist:

```
<response>
    <success>1</success>
    <messages>No user with specified AccountID: test123456</messages>
    <accountID></accountID>
</response>
```

Example 2 - AccountID exists and the user account is already a member of the portal:

```
<response>
    <success>1</success>
    <accountID>test123456</accountID>
</response>
```

Example 3 - AccountID exists but the user is NOT a member of the portal (removed user)

```
<response>
    <success>1</success>
    <messages>User exists with the specified AccountID: test123456, but removed from
portal.</messages>
    <accountID></accountID>
</response>
```

Notes:

A success value of 1 indicates that the call completed with no errors.

A success value of 0 indicates that an unexpected error has occurred.

If the student indicated by the supplied accountID is already a member of the portal, the accountID will be populated. If the student indicated by the supplied accountID does NOT exist or has been removed from the portal, the accountID value will be blank.

registerUser

Registers a new user at LearningZen in the given portal. Also used to re-import a removed user.

URL example: https://[YOURPORTAL].learningzen.com/api/registerUser

Restrictions:

Token & AccountId must not be null or greater than 256 characters.

Token must be alphanumeric, and contain no special characters other than dash (-) and underscore (_).

userGroups & managerGroups must not contain any commas "," in the group names. For example a group named "Managers, Authors, & Admins" will be split into 3 groups   "Managers", "Authors", & "Admins"

If the user previously existed in the LearningZen portal, The supplied Email address & AccountID values must match the values for the user at the time of removal.

Errors may occur if the user belongs to another portal on Learningzen.

Request example:

Via URL Parameters:

```
/registerUser?token=abcdef123456&userGroups=Group+One,Group+Two&managerGroups=Group+Three&isPortalAdmin=0&isAuthor=0&isManager=0&firstName=John&lastName=Doe&emailAddress=john@doe.com&timeZoneName=Eastern+Standard+Time&accountID=678
```

Via POST:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
       <token>abcdef123456</token>
       <sourceIP>127.0.0.1</sourceIP>
       <portalHost>thirdpartysandbox</portalHost>
       <userGroups>Group One,Group Two</userGroups>
       <managerGroups>Group Three</managerGroups>
       <isPortalAdmin>0</isPortalAdmin>
       <isAuthor>0</isAuthor>
       <isManager>0</isManager>
       <firstName>John</firstName>
       <lastName>Doe</lastName>
       <emailAddress>john@doe.com</emailAddress>
       <timeZoneName>Eastern Standard Time</timeZoneName>
       <accountID> test123456</accountID>
       <newPassword>password</newPassword>
</request>
```

Name	Type	Notes
userGroups	string list	Comma separated list of groups to which the user belongs. LearningZen will automatically update the user's membership in the portal groups specified, removing the user from all groups not included in the list. userGroups & managerGroups can be ignored via a portal configuration.
managerGroups	string list	Comma separated list of groups for which the user is a manager. Functionality is otherwise the same as for the userGroups element. Only applicable if isManager below is "1". userGroups & managerGroups can be ignored via a portal configuration.
isPortalAdmin	bit	If "1", LearningZen will update the user's access rights to be a portal administrator. If "0", LearningZen will remove any administrative rights for the user. isPortalAdmin, isAuthor, isManager can be ignored via a portal configuration.
isAuthor	bit	If "1", LearningZen will add the user to the list of portal authors. If the portal is already at the maximum number of portal authors, this request will be ignored. If "0", LearningZen will remove the user from the list of portal authors. isPortalAdmin, isAuthor, isManager can be ignored via a portal configuration.
isManager	bit	If "1", LearningZen will add the user to the list of portal managers. If "0", LearningZen will remove the user from the list of portal managers. isPortalAdmin, isAuthor, isManager can be ignored via a portal configuration.
firstName	string	User's first name.
lastName	string	User's last name.
emailAddress	string	User's email address. Used in LearningZen as the account login.
timeZoneName	string	Possible values
accountID	string	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student.
newPassword	string	Student password for logon. If left blank, the system will provide a random password.

Response example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
	<success>1</success>
	<accountID> test123456</accountID>
</response>
```

Name	Type	Notes
accountID	string	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student.

updateUser

Updates a user at LearningZen in the given portal.

URL example: https://[YOURPORTAL].learningzen.com/api/registerUser

Restrictions:

Token & AccountId must not be null or greater than 256 characters.

Token must be alphanumeric, and contain no special characters other than dash (-) and underscore (_).

userGroups & managerGroups must not contain any commas "," in the group names. For example a group named "Managers, Authors, & Admins" will be split into 3 groups "Managers", "Authors", & "Admins"

Request example:

Via URL Parameters:

```
/updateUser?token=abcdef123456&userGroups=Group+One,Group+Two&managerGroups=Group+Three&isPortalAdmin=0&isAuthor=0&isManager=0&firstName=John&lastName=Doe&emailAddress=john@doe.com&timeZoneName=Eastern+Standard+Time&accountID=678&newPassword=password
```

Via POST:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
        <token>abcdef123456</token>
        <sourceIP>127.0.0.1</sourceIP>
        <portalHost>thirdpartysandbox</portalHost>
        <userGroups>Group One,Group Two</userGroups>
        <managerGroups>Group Three</managerGroups>
        <isPortalAdmin>0</isPortalAdmin>
        <isAuthor>0</isAuthor>
        <isManager>0</isManager>
        <firstName>John</firstName>
        <lastName>Doe</lastName>
        <emailAddress>john@doe.com</emailAddress>
        <timeZoneName>Eastern Standard Time</timeZoneName>
        <accountID> test123456</accountID>
<newPassword>password</newPassword>
<oldPassword>oldpassword</oldPassword>
</request>
```

Name	Type	Notes
userGroups	string list	Comma separated list of groups to which the user belongs. LearningZen will automatically update the user's membership in the portal groups specified, removing the user from all groups not included in the list. userGroups & managerGroups can be ignored via a portal configuration.
managerGroups	string list	Comma separated list of groups for which the user is a manager. Functionality is otherwise the same as for the userGroups element. Only applicable if isManager below is "1". userGroups & managerGroups can be ignored via a portal configuration.

isPortalAdmin	bit	If "1", LearningZen will update the user's access rights to be a portal administrator. If "0", LearningZen will remove any administrative rights for the user. isPortalAdmin, isAuthor, isManager can be ignored via a portal configuration.
isAuthor	bit	If "1", LearningZen will add the user to the list of portal authors. If the portal is already at the maximum number of portal authors, this request will be ignored. If "0", LearningZen will remove the user from the list of portal authors. isPortalAdmin, isAuthor, isManager can be ignored via a portal configuration.
isManager	bit	If "1", LearningZen will add the user to the list of portal managers. If "0", LearningZen will remove the user from the list of portal managers. isPortalAdmin, isAuthor, isManager can be ignored via a portal configuration.
firstName	string	User's first name.
lastName	string	User's last name.
emailAddress	string	User's email address. Used in LearningZen as the account login.
timeZoneName	string	Possible values
accountID	string	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student.
newPassword	String	Password to set for user logon. If either this or the old password is left blank, the user's password will not update.
oldPassword	String	This is the user's old password. If they are updating their current password, they will need to provide the old Learning Zen password.

Response example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
        <success>1</success>
        <accountID> test123456</accountID>
</response>
```

Name	Type	Notes
accountID	string	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student.

removeUser

The user will be removed from the portal. As a part of this task the user will have any active course assignments withdrawn, be removed from all groups, and remove the user from all portal data and reporting. Note: the user's record will still be linked to the AccountID specified in the /registerUser method

URL example: `https://[YOURPORTAL].learningzen.com/api/removeUser`

Restrictions: Token and AccountId must not be null or greater than 256 characters, must be alphanumeric, and contain no special characters other than dash (-) and underscore (_).

Request example:

Via URL Parameters:

```
/removeUser?token=abcdef123456&accountID=54321
```

Via POST:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
        <token>abcdef123456</token>
        <sourceIP>127.0.0.1</sourceIP>
        <portalHost>thirdpartysandbox</portalHost>
        <accountID> test123456</accountID>
</request>
```

Name	Type	Notes
accountID	string	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student.

Response example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
        <success>1</success>
</response>
```

registerCourse

Assigns a course to a student

URL example: https://[YOURPORTAL].learningzen.com/api/registerCourse

Restrictions: Token and AccountId must not be null or greater than 256 characters, must be alphanumeric, and contain no special characters other than dash (-) and underscore (_).

Request example:

Via URL Parameters:

```
/registerCourse?token=[TOKEN]&accountId= test123456&emailAddress=john@doe.com&courseId=4321
```

Via POST:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
        <token>abcdef123456</token>
        <sourceIP>127.0.0.1</sourceIP>
        <portalHost>thirdpartysandbox</portalHost>
        <accountID> test123456</accountID>
        <courseID>4321</courseID>
</request>
```

Name	Type	Notes
accountId	string	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student. In this method, the accountId has been encrypted by the TP.
courseId	int	LearningZen internal course identifier
emailAddress	string	User's LearningZen account email address. In the event of a non TP identification token, pass in the user's LearningZen registration email address.

Response example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
	<success>1</success>
	<accountID>test123456</accountID>
	<courseID>4321</courseID>
</response>
```

Name	Type	Notes
accountID	string	Either student ID used by the TP to identify the student, or for non SSO users, the LearningZen email address. This value may be used in future API method calls to identify the student.
courseID	Int	LearningZen internal course identifier.

Notes:

A success value of 1 indicates that the call completed with no errors.

A success value of 0 indicates that an unexpected error has occurred. If there is an error that occurs, the response will contain a message indicating the error.

courseCompletions

Returns information about completed courses. Passing an accountID will return all completed courses for that student. Passing a courseID will show all completed courses for all students who completed the given course. Passing both courseID and accountID will show completions of one course by one student. Passing lastDate and lastTime will filter out any course completions that occurred before the give date/time. All parameters are optional. If accountID is specified, the <accountID> element will not appear in the response.

URL example: https://[YOURPORTAL].learningzen.com/api/courseCompletions

Restrictions: Token and AccountId must not be null or greater than 256 characters, must be alphanumeric, and contain no special characters other than dash (-) and underscore (_).

Request example:

Via URL Parameters:

```
/courseCompletions?token=abcdef123456&accountID= test123456
```

Via POST:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
```

```
        <token>abcdef123456</token>
        <sourceIP>127.0.0.1</sourceIP>
        <portalHost>thirdpartysandbox</portalHost>
        <accountID> test123456</accountID>
</request>
```

Name	Type	Notes
accountID	String	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student.
courseID	int	LearningZen internal course identifier
lastDate	date	Return course completions since this date. Format yyyymmdd.
lastTime	time	Return course completions since this time. Format HHmmssSSS, where SSS is optional. Time must be in Coordinated Universal Time (UTC).

Response example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
        <success>1</success>
        <courses>
                <course>
                        <courseID>80</courseID>
                        <title>Food Handling Safety</title>
                        <completionStatus>Passed</completionStatus>
                        <completionDate>4/8/2010 5:19:49 PM</completionDate>
                        <scorePercent>100</scorePercent>
                </course>
                <course>
                        <courseId>473</courseId>
                        <title>New Hire Training</title>
                        <completionStatus>Failed</completionStatus>
                        <completionDate>9/8/2010 8:14:23 AM</completionDate>
                        <scorePercent>66.6666666666667</scorePercent>
                </course>
        </courses>
</response>
```

Name	Type	Notes
accountID	string	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student. This element won't be returned if accountID was specified in the request.
courseID	int	LearningZen internal course identifier
title	string	Course title
completionStatus	string	Passed or Failed

completionDate	datetime	Date/time of course completion
scorePercent	double	Percentage of questions answered correctly. "n/a" if there was no exam for the course.

courseNonCompletions

Returns information about courses which have been assigned but not yet completed. Passing an accountID will return all incomplete course assignments for that student. Passing a courseID will return all incomplete assignments for that course for all students. Passing both courseID and accountID will show if a single course is incomplete for one student. Passing the assignedStartDateTimeUTC will return incomplete assignments that were assigned after the indicated date/time. Passing the assignedEndDateTimeUTC will return incomplete assignments that were assigned before the indicated date/time.

URL example: `https://[YOURPORTAL].learningzen.com/api/courseNonCompletions`

Restrictions: Token must not be null or greater than 256 characters, must be alphanumeric, and contain no special characters other than dash (-) and underscore (_). accountID, courseID, assignedStartDateTimeUTC, and assignedEndDateTimeUTC are all optional.

Request example:

Via URL Parameters:

```
/courseNonCompletions?token=abcdef123456&accountID=test123456
```

Via POST:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
       <token>abcdef123456</token>
       <sourceIP>127.0.0.1</sourceIP>
       <portalHost>thirdpartysandbox</portalHost>
       <accountID>1234567890</accountID>
       <courseID>5678</courseID>
       <assignedStartDateTimeUTC>2025-01-01 00:00:01</assignedStartDateTimeUTC>
       <assignedEndDateTimeUTC>2025-12-31 23:59:59</assignedEndDateTimeUTC>
</request>
```

Name	**Type**	**Notes**
accountID	String	Student ID used by the TP to identify the student.
courseID	int	LearningZen internal course identifier.
assignedStartDateTimeUTC	datetime	Returns incomplete assignments that were assigned after this date/time.
assignedEndDateTimeUTC	datetime	Returns incomplete assignments that were assigned before this date/time.

Response example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
    <success>1</success>
    <courses>
```

```
        <course>
            <accountID>1234567890</accountID>
            <emailAddress>blue@devgs.net</emailAddress>
            <courseID>5678</courseID>
            <courseTitle>New Hire Training Course</courseTitle>
            <assignedDateTimeUTC>2025-12-19 21:29:21</assignedDateTimeUTC>
        </course>
    </courses>
</response>
```

Name	**Type**	**Notes**
accountID	string	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student.
emailAddress	String	Email address of the assigned student.
courseID	int	LearningZen internal course identifier.
courseTitle	string	Course title.
AssignedDateTimeUTC	datetime	Date/time in UTC that the course was assigned.

courseInfo

Returns information about all the courses in the portal

URL example: https://[YOURPORTAL].learningzen.com/api/courseInfo

Restrictions: Token must not be null or greater than 256 characters. Token must be alphanumeric, and contain no special characters other than dash (-) and underscore (_).

Request example:

Via URL Parameters:

```
/courseInfo?token=abcdef123456
```

Via POST:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
        <token>abcdef123456</token>
        <sourceIP>127.0.0.1</sourceIP>
        <portalHost>thirdpartysandbox</portalHost>
</request>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
        <success>1</success>
        <courses>
                <course>
                        <courseID>80</courseID>
                        <courseName>Food Handling Safety</courseName>
```

```
                      <description>This course cites the United States Department of
Agriculture's standards on Food Safety and Inspection Services. The material in this course
will cover the basics of food handling safety including; properly cleaning, avoiding cross-
contamination, cooking temperatures, and refrigeration.   The six video series that is
presented throughout the course was created by the Idaho Central Health Department. In total,
the videos will not take more than 30 minutes of viewing time.</description>
                      <ceuCredits>2.75</ceuCredits>
                      <cost>10.00</cost>
                      <timeReqd>Less than one hour</timeReqd>
                      <status>Active</status>
                      <category>Public Health</category>
               </course>
        </courses>
</response>
```

Name	**Type**	**Notes**
courseID	int	LearningZen internal course identifier
courseName	string	Course name
description	string	Non-HTML description of course
ceuCredits	double	Number of credits
cost	double	Cost of the course in US dollars
timeReqd	string	Possible values: Less than 30 minutes, Less than one hour, Four hours or less, Less than eight hours, More than eight hours
status	string	Possible values: Incomplete, Under Review, Active, Archived, Deleted, Taken Offline by Author
category	string	Course Category

takeCourse

Automatically log a registered user into the specified course

*This method should only be used if you need to restrict students in learningzen to taking courses only

*To simply land a student in a course it is recommended to use SSO to https://[YOURPORTAL].learningzen.com/Study/ViewCourse.aspx?CourseID=[CourseID]

URL example: https://[YOURPORTAL].learningzen.com/api/takeCourse

Via URL Parameters:

```
/takeCourse?accountId= test123456&courseId=4321&return=/portal/courses
```

Name	**Type**	**Notes**
accountId	string	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student. In this method, the accountId has been encrypted by the TP.
courseId	int	LearningZen internal course identifier

return	string	Partial URL to be appended to a separate, configurable value provided by the portal. Users are returned to this combined URL when exiting the course.

After LZ receives the API request, it will attempt to verify the encrypted accountId through an API call to the TP; ://[TP-API]/verifyAccountGuid?token=[accountId]. The TP will decrypt the accountId, determine its authenticity, and respond to LZ with the decrypted accountId as in the following examples:

```
<?xml version="1.0" encoding="UTF-8"?>
   <response>
        <success>1</success>
        <accountId> test123456</accountId>
        <token>4B320DBC16968CA8</token>
   </response>
```

```
<?xml version="1.0" encoding="UTF-8"?>
   <response>
        <success>0</success>
        <message>The provided accountId could not be verified.</message>
        <token> A3320DBC16968CA9</token>
   </response>
```

Name	Type	Notes
success	bit	"1" if the request was successful and the token is valid; "0" if it failed for any reason.
message	string	Additional information, usually when success is "0"
accountId	int	The decoded TRS user ID
token	string	The token (GUID) parameter from the request

Notes:

If the response returns successfully, the user will be logged in presented with the specified course. In the event that a problem occurred, the user will be redirected to the specified return URL, with any errors represented by a message parameter, as in the following example:

http://[TP-API]/userReturn?message=User%20Authentication%20Failed

registerPath

Assigns a student to a Learning Path

URL example: https://[YOURPORTAL].learningzen.com/api/registerPath

Restrictions: Token and AccountId must not be null or greater than 256 characters, must be alphanumeric, and contain no special characters other than dash (-) and underscore (_).

Request example:

Via URL Parameters:

```
/registerPath?token=[TOKEN]&accountId= test123456&emailAddress=john@doe.com&pathId=4321
```

Via POST:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
        <token>abcdef123456</token>
        <sourceIP>127.0.0.1</sourceIP>
        <portalHost>thirdpartysandbox</portalHost>
        <accountID> test123456</accountID>
        <pathID>4321</pathID>
</request>
```

Name	Type	Notes
accountId	string	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student. In this method, the accountId has been encrypted by the TP.
pathId	int	LearningZen internal learning path identifier
emailAddress	string	User's LearningZen account email address. In the event of a non TP identification token, pass in the user's LearningZen registration email address.

Response example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
        <success>1</success>
        <accountID> test123456</accountID>
        <courseID>4321</courseID>
</response>
```

Name	Type	Notes
accountID	string	Either student ID used by the TP to identify the student, or for non SSO users, the LearningZen email address. This value may be used in future API method calls to identify the student.
pathId	Int	LearningZen internal learning path identifier.

Notes:

A success value of 1 indicates that the call completed with no errors.

A success value of 0 indicates that an unexpected error has occurred. If there is an error that occurs, the response will contain a message indicating the error.

pathCompletions

Returns information about completed learning path assignments. Passing an email address will return all completed courses for that student. Passing a learningPathId will show all completed path assignments for all students who completed the given learning path. Passing both courseID and email will show completions of one learning path by one student. Passing lastDateTimeUTC will filter out any path completions that occurred before the given date/time. All parameters are optional, but at least one parameter must be supplied

URL example: https://[YOURPORTAL].learningzen.com/api/pathCompletions

Restrictions:

Token and AccountId must not be null or greater than 256 characters, must be alphanumeric, and contain no special characters other than dash (-) and underscore (_).

Token must not be null or greater than 256 characters, must be alphanumeric, and contain no special characters other than dash (-) and underscore (_).

One or more of the following parameters must be present: emailAddress, pathTitle, lastDateTimeUTC.

pathTitle must not be null or greater than 256 characters, must be alphanumeric, and contain no special characters other than dash (-) and underscore (_).

lastDateTimeUTC must be a valid date in the format YYYY-MM-dd HH:mm:ss, and should represent the desired date & time as UTC+0 hours

Request example:

Via URL Parameters:

```
/pathCompletions?token=abcdef123456&emailAddress=student%40yourdomain.com

/pathCompletions?token=abcdef123456&pathTitle=A%20Test%20Track  {A Test Path}

/pathCompletions?token=abcdef123456&lastDateTimeUTC=2000-01-01%2000:00:00 {2000-01-01
00:00:00}
```

Via POST:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
       <token>abcdef123456</token>
       <sourceIP>127.0.0.1</sourceIP>
       <portalHost>thirdpartysandbox</portalHost>
       <accountID> test123456</accountID>
</request>
```

Name	Type	Notes
EmailAddress	String	The email address the student uses to login to LearningZen
PathTitle	int	The name of the learning path as it appears in learningzen
CompletionDateUTC	date	Only completion data occurring after this date & time will be returned (Date Format: YYYY-MM-dd HH:mm:ss)

Response example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
        <success>1</success>
        <paths>
                <path>
                        <emailAddress>testuser@domail.com</emailAddress>
                        <pathTitle>Food Handling Safety</pathTitle>
                        <lastDateTimeUTC>4/8/2010 5:19:49 PM</lastDateTimeUTC>
                </path>
                <path>
                        <emailAddress>testuser@domail.com</emailAddress>
                        <pathTitle>New Hire Training</pathTitle>
                        <lastDateTimeUTC>2018-07-04 16:56:25</lastDateTimeUTC>
                </path>
        </paths>
</response>
```

pathNonCompletions

Returns information about learning paths which have been assigned but not yet completed. Passing an accountID will return all incomplete path assignments for that student. Passing a pathID will return all incomplete assignments for that path for all students. Passing both pathID and accountID will show if a single path is incomplete for one student. Passing the assignedStartDateTimeUTC will return incomplete assignments that were assigned after the indicated date/time. Passing the assignedEndDateTimeUTC will return incomplete assignments that were assigned before the indicated date/time.

URL example: https://[YOURPORTAL].learningzen.com/api/pathNonCompletions

Restrictions: Token must not be null or greater than 256 characters, must be alphanumeric, and contain no special characters other than dash (-) and underscore (_). accountID, pathID, assignedStartDateTimeUTC, and assignedEndDateTimeUTC are all optional.

Request example:

Via URL Parameters:

```
/pathNonCompletions?token=abcdef123456&accountID=test123456
```

Via POST:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
        <token>00000000-0000-0000-0000-000000000077</token>
        <sourceIP>127.0.0.1</sourceIP>
        <portalHost>thirdpartysandbox</portalHost>
        <accountID>1234567890</accountID>
        <pathID>1234</pathID>
        <assignedStartDateTimeUTC>2025-01-01 00:00:01</assignedStartDateTimeUTC>
        <assignedEndDateTimeUTC>2025-12-31 23:59:59</assignedEndDateTimeUTC>
</request>
```

Name	**Type**	**Notes**
accountID	String	Student ID used by the TP to identify the student.
pathID	int	LearningZen internal path identifier.

assignedStartDateTimeUTC	datetime	Returns incomplete assignments that were assigned after this date/time.
assignedEndDateTimeUTC	datetime	Returns incomplete assignments that were assigned before this date/time.

Response example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
    <success>1</success>
    <paths>
        <path>
            <accountID>1234567890</accountID>
            <emailAddress>blue@devgs.net</emailAddress>
            <pathID>1234</pathID>
            <pathTitle>New Hire Training Path</pathTitle>
            <assignedDateTimeUTC>2025-12-19 21:29:21</assignedDateTimeUTC>
        </path>
    </paths>
</response>
```

Name	Type	Notes
accountID	string	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student.
emailAddress	String	Email address of the assigned student.
pathID	int	LearningZen internal path identifier.
pathTitle	string	Path title.
AssignedDateTimeUTC	datetime	Date/time in UTC that the path was assigned.

pathInfo

Returns information about all the learning paths in the portal

Name	Type	Notes
EmailAddress	String	Student ID used by the TP to identify the student. This value may be used in future API method calls to identify the student.
PathTitle	int	LearningZen Learning Path name
CompletionDateUTC	date	Only completion data occurring after this date & time will be returned (Date Format: YYYY-MM-dd HH:mm:ss)

URL example: https://[YOURPORTAL].learningzen.com/api/pathInfo

Restrictions: Token must not be null or greater than 256 characters. Token must be alphanumeric, and contain no special characters other than dash (-) and underscore (_).

Request example:

Via URL Parameters:

```
/pathInfo?token=abcdef123456
```

Via POST:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request>
        <token>abcdef123456</token>
        <sourceIP>127.0.0.1</sourceIP>
        <portalHost>thirdpartysandbox</portalHost>
</request>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
        <success>1</success>
        <paths>
                <path>
                        <learningPathID>80</learningPathID>
                        <pathTitle>Food Handling Safety</pathTitle>
                        <orderedCourses>1</orderedCourses>
                        <status>Published</status>
                        <CertificateOption>Course Completion</category>
                </path>
        </paths>
</response>
```

Name	Type	Notes
learningPathID	int	LearningZen internal learning path identifier
pathTitle	string	Learning Path name
OrderedCourses	bit	"1" or "0"
status	string	Possible values: Published, Offline, Deleted
certificateOption	string	Possible values: Course Completion,Path Completion, Both Course & Path Completion, No Certificates, Unknown