ED2GO PARTNER API 4.1 IMPLEMENTATION GUIDE

May 1, 2020

A resource for understanding how to manage students, courses, and registrations using the ed2go API.

ed2go Partner API

Version 4.1 – May 1, 2020

Introduction

This document is intended for existing or prospective ed2go partners and their associated application developers. It details the features and functionalities supported within the ed2go Partner API. The API structures its data using three core objects:

- **Students** can be anyone who is interested in learning through your organization.
- **Courses** are ed2go products that can be offered to your students.
- **Registrations** are the intersection of your students and ed2go courses.

Capabilities of the API

- Provide ed2go course information in detail.
- Create and manage ed2go student account records.
- Submit student registrations in ed2go courses.
- Authenticate students into their ed2go account or course material.
- Provides student activity, progress, and completion data for ed2go registrations.
- Manage your ed2go course offerings and pricing.

Potential uses

- Synchronize ed2go student, catalog, and registration data with your registration systems.
- Display and sell ed2go courses on your own website or platform.
- Create reports that allow users to review student activity and progress.

What's New in 4.1?

Version 4.1 of the ed2go Partner API builds upon the existing infrastructure. None of the new functionality added is required and upgrading to the new version should require little to no effort.

Tagging

Student and Registration tagging is now supported as of the 4.1 version of the ed2go Partner API in order to provide additional information and context about the record, or as a way of relating records to each other. Tagging in the API functions similarly to the hash tagging convention seen around the web.

Single Sign On Return URL

API integrations can now optionally specify a ReturnURL in the SingleSignOn methods found in both the StudentManager and Registration services. This allows the ed2go system to direct students back to the ReturnURL specified when the user logs out of the ed2go system instead of being directed to an ed2go login page.

Getting started

Prerequisites

1. Create an institutional account with ed2go (also known as an ed2go site).
2. Ensure you have a valid sandbox API key and can connect to the Partner API web services.
3. Become familiarized with the request and response format of SOAP (Simple Object Access Protocol). For more information, see: https://msdn.microsoft.com/en-us/library/ms995800.aspx

ed2go Partner API

Version 4.1 – May 1, 2020

4. Fully review this document and ed2go Partner API web service instructions found here:
    https://x-admin.ed2go.com/static/oac/api/ed2go-partner-api4-1.pdf

Connecting to the sandbox

You may utilize our sandbox environment to develop and test your implementation. Data in the sandbox environment may be periodically refreshed without notice. Sandbox web services found below:

*   Student Manager Service (SM)
    https://x-api.ed2go.com/partner/4.1/StudentManagerService.asmx
*   Course Information Service (CI)
    https://x-api.ed2go.com/partner/4.1/CourseInformationService.asmx
*   Registration Service (R)
    https://x-api.ed2go.com/partner/4.1/RegistrationService.asmx
*   Catalog Manager Service (CM)
    https://x-api.ed2go.com/partner/4.1/CatalogManagerService.asmx

To troubleshoot your implementation, you may access the sandbox Online Administration Center to review API logs, students, courses, and registrations at the link below:

*   Online Administration Center
    https://x-admin.ed2go.com

The API key

The API key is a 36 character GUID (Global Unique Identifier) and a critical component to each and every interaction with our API. Once ed2go has activated API access for your institution, an API key will be provided. You may also locate your API key within the affected institution's ed2go Online Administration Center. The API key is an extremely sensitive piece of data and we recommended that you store it in a secure manner.

Implementation principles

Please keep the following principles in mind when considering implementation options:

1. Due to the sensitivity of the data we require that all transactions utilize TSL/SSL cryptographic protocols.
2. Your API implementation should be able to handle the addition of new fields and reordering of existing fields by ed2go with ease, simply ignoring these values without error.
3. Not all fields appear in every scenario and it is generally safe to treat a null field, empty set, or absence of field in the same way.

Going live

You should make sure your implementation works well in the sandbox environment before deciding to move into the live ed2go production environment. Once you feel comfortable with the sandbox implementation, you may request a production API key in order to launch.

ed2go Partner API

Version 4.1 – May 1, 2020

- Student Manager Service (SM)
  https://api.ed2go.com/partner/4.1/StudentManagerService.asmx
- Course Information Service (CI)
  https://api.ed2go.com/partner/4.1/CourseInformationService.asmx
- Registration Service (R)
  https://api.ed2go.com/partner/4.1/RegistrationService.asmx
- Catalog Manager Service (CM)
  https://api.ed2go.com/partner/4.1/CatalogManagerService.asmx
- Online Administration Center
  https://admin.ed2go.com/

Web services & methods

In order to ensure a minimally viable integration, there are methods that we require be utilized. The required methods are denoted along with the method name and description for each web service.

Student Manager Service

Allows implementations to create, retrieve, and update student data. It also allows implementations to retrieve single sign on (SSO) information pertaining to the specific student account.

Methods

Name	Description
AddStudent *Required*	Adds a new student record to the ed2go system.
GetStudents *Required*	Gets students based on filter criteria.
UpdateStudent *Required*	Updates student info. Excluded elements are left as is. To remove data from an optional field, element must be included with an empty value.
GetSingleSignOn *Optional**	Gets single sign on URL for the student. You must implement to at least one of the GetSingleSignOn methods (RegistrationService recommended).

Course Information Service

Allows implementations to retrieve ed2go course and category information/details.

Methods

Name	Description

ed2go Partner API

GetCourses Required	Gets courses based on filter criteria.
GetCategories *Optional*	Gets category information by CategoryID.
GetCategoryList Optional	Gets a list of all categories.

Registration Service

Allows implementations to create and cancel registrations. It also allows implementations to retrieve registration details, activity, progress, and single sign on (SSO) information pertaining to the specific registration.

Methods

Name	Description
AddRegistration *Required*	Adds a new registration.
GetRegistrations *Required*	Gets registrations based on filter criteria.
UpdateRegistration Required	Updates values associated with an existing registration.
GetSingleSignOn *Recommended*	Gets single sign on URL for a registration. You must implement to at least one of the GetSingleSignOn methods (RegistrationService recommended).
CancelRegistration *Optional*	Cancels an existing registration.
GetRegistrationActivityReport *Optional*	Gets student activity report for the registration.
GetRegistrationProgressReport *Optional*	Gets student progress report for the registration.

ed2go Partner API

CatalogManagerService

Allows implementations to retrieve and modify catalog offerings and price data associated with the corresponding ed2go hosted site. Implementing to this service should only be needed if you plan to have students enroll at an ed2go hosted site.
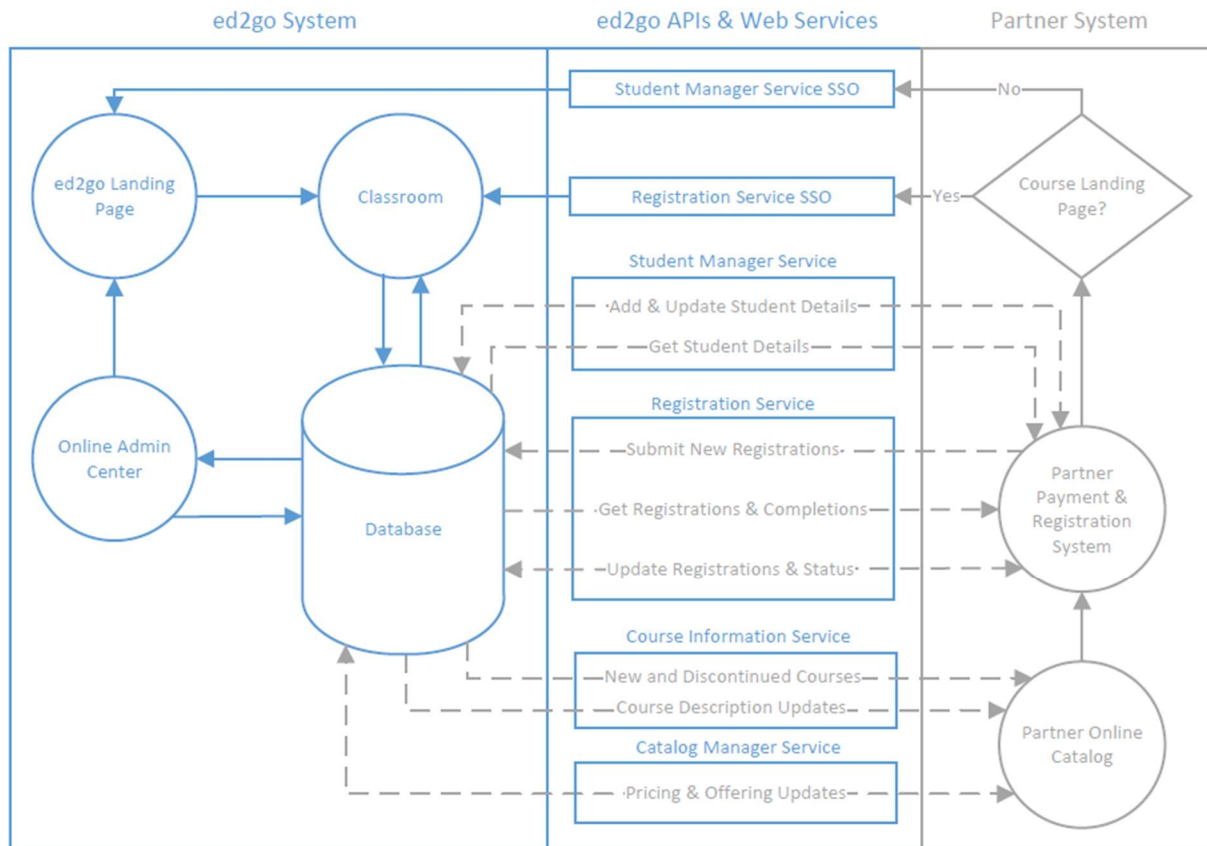
Methods

Name	Description
GetCourseStandardPrices *Required*	Gets standard pricing for provided courses.
GetOfferedCoursePrices Required	Gets course prices based on filter criteria.
GetUnofferedCourses *Required*	Gets unoffered courses based on filter criteria.
RemoveOfferedCourses *Required*	Disables course offerings based on filter criteria.
SetCoursePrice *Recommended*	Sets course prices in the ed2go system based on filter criteria.
GetCoursesBySectionNumber *Optional*	Gets courses with provided section number.
GetOfferedCourses *Optional*	Gets offered courses based on filter criteria.
GetSettings *Optional*	Gets general catalog settings for the site.
SetCourseSectionNumber *Optional*	Sets course section number based on filter criteria.

ed2go Partner API

Version 4.1 – May 1, 2020

Interactions Diagram

Resources, fields, and types

General

Generic fields that display in all web services.

Field	Description
APIKey guid	Value required for all requests. Acts as the password for your implementation and is validated against the key associated with your ed2go site.
Result object	Object output in each API response. Outlines the results of the request.
Success boolean	Child node of the Result object. Indicates whether the request was successful.

ed2go Partner API

Errors object	Child node of request object. Parent object of any errors encountered with the request.
Error string	Coded error responses. Successful responses may still contain errors.

Students

Fields and types that are specific to students.

Field	Description
StudentKey guid	ed2go's unique identifier for the student. Each student identifier is distinct from any other student identifier in the ed2go system.
CustomStudentKey string	Your unique identifier for the student. The ed2go system enforces that this identifier is unique amongst your institution.
Email string	The student's email. This value will also act as the student's username in the ed2go system.
FirstName string	The student's first name.
LastName string	The student's last name.
Address string	Object containing strings that denote the student's address. If the course enrolled utilizes textbooks or other materials, this is the address we will ship them to.
Address1 string	Child node of the Address object. The student's house number and street name.
Address2 string	Child node of the Address object. The student's supplementary address information. Examples include suite number, apartment number, etc. Optional field.
City string	Child node of the Address object. The student's city.

StateProvince string	Child node of the Address object. The student's state or province. Supports abbreviations or full state/province name.
PostalCode string	The student's zip/postal code.
CountryCode string	The student's country. Supports abbreviations or full country name.
Tags object	Array of values that can be submitted or retrieved by your integration. When retrieving records by their tags, the following logical operators are supported (only one can be included): **AND:** Returns the students that have all of the tags included in the request. **OR:** Returns students that have any of the tags included in the request. **NOT:** Returns students that don't have any of the tags included in the request.
Tag string	Child node of the Tags object. Indicates an individual tag value attribute of the affected student. Tag values may only contain alphanumeric characters, and cannot contain blank or empty characters. Tag values must be 50 characters or less. Tag values are case sensitive for submission, but case insensitive for retrieval. For retrieval, partial matches are not supported. Tags submitted will overwrite all existing tags rather than append to existing, allowing for tag removal.
Custom object	Group of additional student data fields that may be collected on behalf of the content provider.
Field key-value pair	Child node of the Custom object. Denotes additional student information that may have been collected from the student.
Destination string	Specifies where the student should arrive upon successfully being authenticated via the SSO. Must be one of the following: **Default:** Directs the user to the default location, in this case it is the same as "Home" **Home:** Directs the student to their ed2go course landing page known as My Classroom. **Profile:** Directs the student to the ed2go My Account or Student Center page.
ReturnURL string	Specifies where the student shall return upon logging out of the ed2go system. This should be a secure (HTTPS) URL. Special characters are expected to be URL encoded. If left blank, the system will return the user to an ed2go login page.

ed2go Partner API

SingleSignOn object	Contains the information necessary to build out the registration Single Sign On (SSO) request.
ExpirationDate datetime	Attribute of the SingleSignOn object. The date and time that the URL will no longer be valid for authentication.
Form object	Child node of the SingleSignOn object. Contains the form elements to produce the SingleSignOn request.
Action string	Child node of the Form object. Represents the URL that the single sign on request should be posted to.
Method string	Child node of the Form object. Represents the method that should be used for the form. At this time, it will always be a POST method.
Data object	Child node of the Form object. Contains the variables necessary for the SSO form submission.
Input object	Child node of the Data object. Has three attributes: Name, Value, and Type. These are the form variables to be passed along with the SSO form submission
AutoPost string	Child node of the SingleSignOn object. Provides the complete html source to allow for SSO post.

Courses

Fields and types that are specific to courses. Many fields get excluded from responses by default and must be included by using the *IncludeFields* object. Course fields may appear in any of the following services: Catalog Manager, Course Information, and Registration.

Field	Description
CourseCode string	ed2go's unique identifier for the course. Also used to represent course categorizations and relationships.
Title string	The title for the affected course.
Summary string	A brief HTML-based text description.

Overview string	An HTML-based overview of the course.
Details string	HTML-based listing of details regarding the course.
Outline string	HTML-based listing of the structure of content delivery in the course.
Objective string	HTML-based listing of the purpose or goals of the course and what the student can expect upon completion.
FAQs string	HTML-based listing of frequently asked questions for the course.
Requirements string	HTML-based listing of requirements for the course. Usually specific to software/hardware requirements.
Prerequisites string	HTML-based listing of prerequisites for the course. Assumed previous knowledge or skillsets recommended for student success in the course.
Instructor string	HTML-based listing of instructor information for the course.
Duration string	ISO 8601 compliant duration. Represents the amount of time the student has access to the course.
Hours decimal	The amount of time the student should spend in real time to complete the course. This value is printed on the certificate of completion provided to the student upon completion.
Image string	The URL reference for the course image.
Index string	The URL for the course details page.
Demo string	The URL to the affected course's demo. Demos are not available for all courses.

ReleaseDate datetime	The date that the course became available for partner offering and student enrollment.
DiscontinueDate datetime	The date that the course was discontinued by ed2go. The course is no longer available for offering or enrollment after this date.
StartDates object	Contains the Start Dates available presently for the course. Will only appear for courses that follow the session model. If applicable, up to four start dates will be included.
StartDate datetime	Child node of the StartDates object. One of the dates available for enrollment for the session based course.
RelatedCourses object	Group of CourseCodes that are related to the affected course.
StudentReviews object	Group of student reviews for the affected course.
StudentReview string	HTML-based listing of a single student review for the affected course.
Categories object	Grouping of course categories.
Category object	Child node of the Categories object.
CategoryID integer	Attribute of the Category object. ed2go's unique identifier for the affected category. The course belongs to each category associated with the CategoryID within the affected course object.
Name string	Child node of the Category object. HTML-based listing of the affected category name.
Description string	Child node of the Category object. HTML-based listing of the affected category description.
WholesalePrice decimal	The price ed2go charges for the affected course.

SuggestedRetailPrice string	The price ed2go suggests for resale of the affected course.
RetailPrice string	The price that the course is currently configured to sell at for the affected partner. This price only impacts sales made at the ed2go hosted site. Registration prices can be set at the time the registration is submitted to the API.

Registrations

Fields and types that are specific to registrations.

Field	**Description**
RegistrationKey guid	ed2go's unique identifier for the registration. This value will be provided upon successful creation of a registration.
LastModifiedDate datetime	The last date & time that a change was made to the registration.
RegistrationDate datetime	The date & time that the registration was created.
StudentKey guid	ed2go's unique identifier for the student of the affected registration.
CustomStudentKey string	Your unique identifier for the student of the affected registration.
CourseCode string	ed2go's unique identifier for the course of the affected registration.
Status string	The current status of the registration. Possibilities include: Pending, Unprocessed, Upcoming, Active, Incomplete, Complete, Suspended, Canceled, or Error. Definitions of these status can be found below.
SectionNumber string	The section number associated with the course at the time the registration was created.
RetailPrice decimal	The price the student was charged for the course in the affected registration.

Currency string	The currency for the affected price.
AccessStartDate datetime	The date that the student can begin accessing the course content for the registration.
Tags object	Array of values that can be submitted or retrieved by your integration. When retrieving records by their tags, the following logical operators are supported (only one can be included): **AND:** Returns the registrations that have all of the tags included in the request. **OR:** Returns registrations that have any of the tags included in the request. **NOT:** Returns registrations that don't have any of the tags included in the request.
Tag string	Child node of the Tags object. Indicates an individual tag value attribute of the affected registration. Tag values may only contain alphanumeric characters, and cannot contain blank or empty characters. Tag values must be 50 characters or less. Tag values are case sensitive for submission, but case insensitive for retrieval. For retrieval, partial matches are not supported. Tags submitted will overwrite all existing tags rather than append to existing, allowing for tag removal.
AccessEndDate datetime	The last day the student can access their course content for the registration.
ActivityReport object	Grouping of the activities a student performed within the affected registration's course content.
Activity array	Child node of ActivityReport. Grouping of a specific activity's details. Has a type attribute of either "Online" or "Offline". This indicates whether the time spent was done within the online content or done via offline study materials.
ActivityDate datetime	Child node of Activity. The date that the specified activity took place.
TimeSpent string	Child node of Activity. **ISO 8601** compliant duration. Represents the amount of time the student spent for the associated activity and type.
Progress object	Grouping of the progress a student has achieved within the affected registration's course content.

PercentProgress decimal	Child node of Progress. Represents the how far the student has progressed through the course as a percentage.
LastAccessDate datetime	Child node of Progress. Represents the date the student last accessed their online course content.
CoursePassed boolean	Child node of Progress. Represents whether the student has passed the course within the affected registration.
PercentOverallScore decimal	Child node of Progress. Represents the overall score the student achieved as a percentage.
CompletionDate datetime	Child node of Progress.
TotalTimeSpent string	Child node of Progress. **ISO 8601** compliant duration. Represents the total amount of time the student has spent in the course regardless of Activity type.
Custom object	Group of additional registration fields that may be collected of the student by the content provider. Can belong to various registration objects.
Field key-value pair	Child node of the Custom object. Denotes additional student information that may have been collected from the student.
Destination string	Specifies where the student should arrive upon successfully being authenticated via the SSO. Must be one of the following: **Default:** Directs the user to the default location, in this case is the same as "Home". **Home:** Directs the user to the landing page of the specific course. For most ed2go courses this is the list of lessons available in the course. Not all courses support this level of SSO, in these cases the user is directed to "StudentHome". **Orientation:** Directs the user to the orientation content for the specific course. Not all courses have orientation content, in these cases user is directed to "Home". **Resume:** Directs the user to the last place that the student was in the affected course. Not all courses support this level of SSO, in these cases the user is directed to "Home". **StudentHome:** Directs the student to the ed2go My Account or Student Center page. Is identical to the "Home" option in StudentManagerService.
SingleSignOn object	Contains the information necessary to build out the registration Single Sign On (SSO) request.

ExpirationDate datetime	Attribute of the SingleSignOn object. The date and time that the URL will no longer be valid for authentication.
Form object	Child node of the SingleSignOn object. Contains the form elements to produce the SingleSignOn request.
Action string	Child node of the Form object. Represents the URL that the single sign on request should be posted to.
Method string	Child node of the Form object. Represents the method that should be used for the form. At this time, it will always be a POST method.
Data object	Child node of the Form object. Contains the variables necessary for the SSO form submission.
Input object	Child node of the Data object. Has three attributes: Name, Value, and Type. These are the form variables to be passed along with the SSO form submission
AutoPost string	Child node of the SingleSignOn object. Provides the complete html source to allow for SSO post.