

edudip next API

Version 2019-03-27

Einsatz

Die edudip next API kann dazu genutzt werden, die Funktionen von edudip next programmatisch zu steuern. So können unter anderem angelegte Webinare ausgelesen und verändert werden. Dies ermöglicht es, edudip next Webinare in die eigene Webseite oder Applikation zu integrieren.

Genereller Aufbau

Bei der edudip next API handelt sich um eine REST API (<https://de.wikipedia.org/wiki/RepresentationalStateTransfer>). Als Format für den Datenaustausch wird JSON verwendet.

Jede Funktion der API wird durch einen so genannten Endpunkt abgebildet. Jeder Endpunkt besteht aus einer URL, dem entsprechenden HTTP-Verb (GET, POST, PUT, DELETE) und optional einer Liste benötigter Parameter.

In diesem Dokument geben wir nur den Pfad des Endpunktes an. Jedem Endpunkt ist somit immer `https://api.edudip-next.com` voranzustellen.

Sollte ein Endpunkt eine Liste von Parametern benötigen, so sind diese Parameter als `multipart/form-data` (<https://www.w3.org/TR/html5/sec-forms.html#multipart-form-data>) oder `application/x-www-form-urlencoded` (<https://www.w3.org/TR/html5/sec-forms.html#urlencoded-form-data>) in der HTTP Anfrage zu kodieren.

Jede API Anfrage sollte zudem den HTTP Header `Accept` mit dem Wert `application/json` beinhalten.

Beispiel für eine Implementierung einer POST Anfrage mittels PHP und cURL:

```
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, 'https://api.edudip-next.com/api/webinars');
$httpHeaders = [
    'Accept: application/json',
    'Authorization: Bearer Ihr API-Token',
];
curl_setopt($ch, CURLOPT_HTTPHEADER, $httpHeaders);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, 'parameter1=value1&parameter2=value2');
$response = curl_exec($ch);
curl_close($ch);
var_dump($response);
```

Für alle API Anfragen, die erfolgreich verarbeitet wurden konnten, wird der HTTP-Status-Code `200 OK` zurückgeliefert. Sollte eine API Anfrage fehlschlagen, so wird ein passender HTTP-Status-Code zurückgegeben.

Sollten Ihnen Funktionen in der API zur Integration in Ihre Applikation fehlen, treten Sie gerne mit uns in Kontakt, wir erörtern dann gerne mit Ihnen, ob diese Funktionalität in der API ergänzt werden kann.

Referenzimplementierung

Unter <https://github.com/edudip/next-api-client> kann eine Beispiel-Implementierung der edudip next API heruntergeladen werden. Diese Implementierung basiert auf PHP und cURL und kann mittels composer leicht in Ihr Projekt eingebunden werden.

Authentifizierung

Authentifizierte Requests mittels API Token

Alle in diesem Dokument beschriebenen API Anfragen müssen authentifiziert werden. Zur Authentifizierung von Anfragen werden so genannte API Tokens verwendet. Ein API-Token ist ein ASCII String von einer Länge von 60 Zeichen. Jeder API Token ist einem Benutzer in edudip next zugeordnet. Jede Aktion die über die API durchgeführt wird somit über den API Token einem Teammitglied zugeordnet und besitzt somit die gleichen Zugriffsrechte wie dieses Teammitglied.

Der API-Token muss in jeder API Anfrage als HTTP Header mit dem Namen "Authorization" mitgeschickt werden:

```
Authorization: Bearer [API-TOKEN]
```

Ersetzen Sie hierbei [API-TOKEN] durch Ihren generierten API-Token.

Api-Tokens dürfen nicht an Dritte weitergegeben werden und sollten sicher aufbewahrt werden. Um volle

Sicherheit gewährleisten zu können, müssen alle API Anfragen über HTTPS stattfinden.

Erstellen von API Tokens

Um einen API Token zu generieren, loggen Sie sich bitte mit Ihrer E-Mail Adresse und Passwort in edudip next ein. Klicken Sie nun auf den Menüpunkt "Einstellungen" auf der linken Seite. Scrollen Sie nun so weit nach unten bis zu dem Punkt "API-Tokens". Klicken Sie nun auf die Schaltfläche "Weiteren API-Token generieren". Das System fordert Sie nun auf, anzugeben welchem Teammitglied dieser API-Token zugeordnet werden soll. Wählen Sie das gewünschte Teammitglied aus und klicken danach auf "API-Token" generieren.

Hinweis: Sollte der Punkt "API-Token" nicht erscheinen oder Sie keine Möglichkeit haben einen API-Token zu generieren, so wurde die API für Ihren Account nicht freigeschaltet. Wenden Sie sich in diesem Fall bitte an sales@edudip.com.

Webinare

Alle Webinare auflisten

Endpunkt: GET /api/webinars

Liefert eine Liste aller vorhandenen Webinare zurück.

Rückgabe:

```
{
  "success": Boolean,
  "total": UInt,
  "webinars": [
    // Liste mit Webinar Objekten
  ]
}
```

Die Property "success" wird im Erfolgsfall auf "true" gesetzt. Die Property "total" gibt die Gesamtzahl an vorhandenen Webinaren an. Die Property "webinars" beinhaltet ein Array mit Webinar-Objekten. Ein Webinar Objekt wird als JSON Objekt enkodiert und besitzt folgende Properties:

Property	Datentyp	Beschreibung
id	Uint	Primärer, eindeutiger Identifier des Webinars
users_id	Uint	Benutzer Id des Teammitglieds, dass dieses Webinar angelegt hat
title	String	Titel bzw. Name des Webinars
access	Enum/String ("all" oder "invitation")	Beschreibt, wer das Webinar buchen darf. "all" bedeutet, dass jeder das Webinar buchen darf. "invitation" bedeutet, dass der Teilnehmer sich nur mit einer Einladung zu dem Webinar anmelden darf.
registration_type	Enum/String	Beschreibt, ob sich Teilnehmer für einen Termin des Webinars oder immer für alle Termine des Webinars registrieren (Einzel- oder Serientermin)
dates	Array	Eine Liste mit Terminen des Webinars
next_date	Object	Beinhaltet den nächsten stattfindenden Termin des Webinars. Ist <code>null</code> wenn kein nächster Termin existiert.
max_participants	Uint	Anzahl der maximalen Teilnehmer am Webinar.
moderators	Array	Eine Liste mit (Co-)Moderatoren des Webinars. Der Ersteller des Webinars ist immer als Hauptmoderator eingetragen.
participants_count	Uint	Anzahl der bereits angemeldeten Teilnehmer zu diesem Webinar
created_at	String	Zeitpunkt der Erstellung des Webinars in der Form <code>YYYY-MM-DD HH:ii:ss</code>
updated_at	String	Zeitpunkt der letzten Änderung des Webinars in der Form <code>YYYY-MM-DD HH:ii:ss</code>

Neues Webinar erstellen

Endpunkt: POST /api/webinars

Dieser API Endpunkt kann dazu genutzt werden, neue Webinare anzulegen. Folgende Parameter müssen bei Benutzung des Endpunktes als HTTP POST Feld übergeben werden:

Parameter	Datentyp	Benötigt	Beschreibung
title	String	✓	Der Titel/Name des Webinars. Muss mindestens 5 und maximal 190 Zeichen umfassen
max_participants	Uint	✓	Maximale Anzahl an Teilnehmern des Webinars. Muss mindestens auf 1 gesetzt werden. Der Höchstwert hängt von Ihrem gebuchten edudip next Abo ab.
recording	Uint	✓	Soll ein Videomitschnitt des Webinars aufgezeichnet werden? 1 = Das Webinar wird aufgezeichnet; 0 = Das Webinar nicht aufzeichnen
registration_type	String	✓	Kann die Werte "series" oder "date" annehmen. "series" = Terminreihe: Teilnehmer registrieren sich für alle Termine gleichzeitig.; "date" = Alternativtermine: Teilnehmer melden sich für jeden Termin einzeln an.
access	String	✓	Kann die Werte "all" oder "invitation" annehmen. "all" = Jeder darf sich anmelden, "invitation" = Nur eingeladene Teilnehmer dürfen sich anmelden
dates	String	✓	JSON enkodiertes Array mit einzelnen Datums-Objekten, an denen das Webinar stattfinden soll. Jedes Datum-Objekt muss zwei Properties besitzen: "date" mit dem Datums-String in der Form YYYY-MM-DD HH:MM:SS, an dem der Termin stattfinden soll, sowie die Property "duration", die in Minuten angibt, wie lange der Termin dauert. Beispiel: <pre>[{"date": "2018-01-20 12:00:00", "duration": 20}]</pre>
users_id	Uint	×	Legt fest, welches Teammitglied der Eigentümer (Hauptmoderator) des Webinars sein soll. Dieses Teammitglied braucht eine Moderatoren-Lizenz. Wenn der Parameter nicht übergeben wird, dann wird der Benutzer, zu dem der API-Token gehört, als Eigentümer eingetragen

Bitte beachten Sie, dass der Ersteller des Webinars (Hauptmoderator) im Webinar anwesend sein muss, damit dieses gestartet werden kann bzw. automatisch startet.

Ein einzelnes Webinar auslesen

Endpunkt: GET /api/webinars/[Webinar-Id]

Dieser Endpunkt kann dafür genutzt werden, die Werte eines einzelnen Webinars gezielt auszulesen. Es werden hierbei mehr Werte des Webinars im Vergleich zu dem Endpunkt `GET /api/webinars` zurückgegeben.

Die Rückgabe liefert im Erfolgsfall folgende JSON Rückgabe zurück:

```
{
  "success": true,
  "webinar": {
    // Webinar-Objekt
  },
  "stats": {
    "views_total": Uint,
    "registrations_total": Uint
  }
}
```

Die Property `webinar` enthält hierbei ein Objekt, dass alle Werte des angefragten Webinars enthält. Zusätzlich enthält die Property `stats` die Werte `views_total` mit der Anzahl an Landingpage Views, die zu diesem Webinar gehört. Zusätzlich gibt der Wert `registrations_total` die Anzahl an vorhandenen Registrierungen zu diesem Webinar an.

Das zurückgelieferte Webinar-Objekt ist weitestgehend identisch mit dem Webinar-Objekt, dass über den Endpunkt `GET /api/webinars` ausgeliefert wird, es werden jedoch noch folgende Werte zusätzlich ausgeliefert:

Property	Datentyp	Beschreibung
registration_type_editable	Boolean	Steht dieser Wert auf <code>true</code> , kann die Anmeldeart (Alternativtermine/Terminreihe, Property <code>registration_type</code>) nicht mehr geändert werden.
recording	Uint	1 = Das Webinar wird aufgezeichnet; 0 = Das Webinar wird nicht aufgezeichnet
slug	String	Der SEO Slug, der für die Webinar Landingpage URL benutzt wird
participants	Array	Eine Liste mit angemeldeten Teilnehmern zu diesem Webinar. Eine Beschreibung eines einzelnen Teilnehmer-Objektes befindet sich unterhalb dieser Tabelle
dialin_enabled	Uint	Dieses Feature ist zur Zeit ohne Funktion 0 = Telefoneinwahl nicht verfügbar; 1 = Telefoneinwahl für dieses Webinar ist verfügbar
recordings	Array	Eine Liste von Aufzeichnungen, die zu diesem Webinar angefertigt wurden.

Ein Teilnehmer-Objekt hat folgende Properties:

Property	Datentyp	Beschreibung
firstname	String	Vorname des Teilnehmers
lastname	String	Nachname des Teilnehmers
auth_key	String	Authentifizierungs Schlüssel. Wird benötigt, um den Webinar-Raum zu betreten und sich zu Webinaren abzumelden
email	String	E-Mail Adresse des Teilnehmers
created_at	String	Zeitpunkt der Registrierung
updated_at	String	Zeitpunkt der letzten Änderung des Datensatzes

Webinar löschen

Endpunkt: DELETE /api/webinars/[Webinar-Id]

Bestehendes Webinar modifizieren

Endpunkt: PUT /api/webinars/[Webinar-Id]

Ändert die Werte/Einstellungen eines bestehenden Webinars ab. Folgende Parameter können beim Aufruf des Endpunktes übergeben werden (analog zu HTTP POST-Anfragen). Dabei müssen nicht immer alle Parameter übergeben werden, es kann auch nur eine beliebige Teilmenge übergeben werden:

Parameter
title
max_participants
recording
registration_type
access

Die Bedeutung und den Wertebereich der einzelnen Parameter entnehmen Sie der Beschreibung der Endpunkt `GET /api/webinars` und `GET /api/webinars/[Webinar-Id]`.

Als Rückgabe im Erfolgsfall wird ein JSON Objekt ausgegeben, mit der Property `success` mit dem Wert `true`, sowie der Property `webinar` welche das modifizierte und aktualisierte Webinar-Objekt beinhaltet.

Neuen Termin zu einem Webinar hinzufügen

Endpunkt: POST /api/webinars/[Webinar-Id]/add-date

Property	Datentyp	Beschreibung
date	String	Datum und Uhrzeit, an dem der Termin stattfinden soll. Format: <code>YYYY-MM-DD HH:MM:SS</code> (z.B. 2019-12-01 12:30:00)
duration	Uint	Dauer des Termins in Minuten.

Rückgabe im Erfolgsfall:

```
{
  "success": true,
  "date": [Webinar-Date Objekt]
}
```

Bestehenden Termin eines Webinars löschen

Endpunkt: DELETE /api/webinars/[Webinar-Id]/dates/[Webinar-Date-Id]

Bitte beachten, dass der letzte Termin eines Webinars nicht gelöscht werden kann, d.h. zu jedem Webinar muss mindestens ein Termin existieren. Desweiteren werden, wenn ein Termin gelöscht wird, auch alle zugehörigen Daten gelöscht. Darunter fallen unter anderem alle Aufzeichnungen und Anmeldedaten von Teilnehmern zu diesem Termin. Wir empfehlen Ihnen daher vor dem Löschen eines Termins alle Aufzeichnungen und relevanten Daten zu sichern.

Einen Teilnehmer an einem Webinar anmelden

Endpunkt: POST /api/webinars/[Webinar-Id]/register-participant

Registriert einen Teilnehmer zum angegebenen Webinar. Folgende POST Parameter können beim Aufruf des Endpunktes angegeben werden:

Parameter	Datentyp	Benötigt	Beschreibung
email	String	✓	E-Mail Adresse des Teilnehmers
firstname	String	✓	Vorname des Teilnehmers
lastname	String	✓	Nachname des Teilnehmers
webinar_date	String	×	Datum im Format <code>YYYY-MM-DD HH:MM:SS</code> , sofern sich nur für einen Termin angemeldet werden soll (wenn Property <code>registration_type</code> des Webinars auf "date" (Einzeltermin) steht).

Rückgabe im Fehlerfall:


```

{
  "success": false,
  "error": {
    "message": String, Fehlermeldung
    "type": Fehler-Code bzw. eindeutiger Identifier der Fehlerklasse (siehe unten)
    "fields": [
      // Optional, nur wenn "type" Wert "form_validation" hat. Dann Array mit Feldnamen, die nicht oder fehlerhaft ausgefüllt wurden
    ]
  }
}

```

Rückgabe im Erfolgsfall:

```

{
  "success": true,
  "participant": {
    "auth_key": String,
    "firstname": String,
    "lastname": String,
    "updated_at": String der Form YYYY-MM-DD HH:ii:ss,
    "created_at": String der Form YYYY-MM-DD HH:ii:ss
  },
  "registeredDates": [
    {
      "date": String der Form YYYY-MM-DD HH:ii:ss,
      "key": String,
      "room_link": String
    }
  ]
}

```

Das Feld `auth_key` enthält den persönlichen Authorisierungs-Schlüssel, mit dem sich ein Teilnehmer im Raum identifizieren kann. Dieser Schlüssel ist bereits in den Webinar-Raum-Link in der Property `room_link` eingesetzt. Der Teilnehmer gelingt über die Property `room_link` direkt in den Webinar Raum zu dem entsprechenden Termin. Jeder Webinar-Termin besitzt einen eigenen Link. Der Link ist personalisiert und nur für diesen Teilnehmer gedacht.

Fehlercodes:

Code	Bedeutung
form_validation	Ein oder mehrere Registrierungs-Felder wurden nicht oder fehlerhaft ausgefüllt
date_missing	Als Registrierungs Typ wurde Einzeltermin ausgewählt, es wurde jedoch kein Termin angegeben, bei dem der Teilnehmer angemeldet werden soll
date not bookable	Der angegebene Webinartermin (Post Parameter "webinar_date") ist nicht mehr buchbar oder liegt in der Vergangenheit
participant_exists	Ein Teilnehmer mit der angegebenen E-Mail Adresse wurde bereits registriert

Einen Teilnehmer zu einem Webinar abmelden

Endpunkt POST /api/webinars/[Webinar-Id]/cancelRegistration

Parameter	Datentyp	Beschreibung
email	String	E-Mail Adresses des Teilnehmers
auth_key	String	Der Authorisierungs Schlüssel des Teilnehmers (siehe "Einen Teilnehmer an einem Webinar anmelden").

Einen Teilnehmer löschen

Endpunkt: DELETE /api/participants/[E-Mail-Adresse des Teilnehmers]

Löscht alle Teilnehmerdaten aus edudip next. Kann zum Beispiel dazu verwendet werden, um Anfragen nach Art. 17 DSGVO auf Recht auf Löschung, nachzukommen. Der Unterschied zu dem Endpunkt

POST /api/webinars/[Webinar-Id]/cancelRegistration besteht darin, dass nach einer Abmeldung die Daten des Teilnehmers im System gespeichert bleiben, um diese weiterverarbeiten zu können.

Moderator zu einem Webinar hinzufügen

Endpunkt: POST /api/webinars/[Webinar-Id]/moderators/add

Fügt einem bestehenden Webinar einen neuen (Co-)Moderator hinzu. Es können bis zu drei (Co-)Moderatoren hinzugefügt werden (zusätzlich zum Eigentümer des Webinars). Folgende Parameter müssen diesem Endpunkt übergeben werden.

Parameter	Datentyp	Beschreibung
email	String	E-Mail Adresses des neuen Moderators
firstname	String	Vorname des Moderators (wird im Webinarraum angezeigt)
lastname	String	Nachname des Moderators (wird im Webinarraum angezeigt)

Bitte beachten, dass der Haupteigentümer des Webinars im Webinarraum anwesend sein muss, um das Webinar starten zu können.

Bestehenden Moderator von einem Webinar entfernen

Endpunkt: DELETE /api/webinars/[Webinar-Id]/moderators/[Moderator-Email]

Entfernt einen (Co-)Moderatoren wieder von einem Webinar.

Parameter	Datentyp	Beschreibung
email	String	E-Mail Adresses des Teilnehmers
auth_key	String	Der Authorisierungs Schlüssel des Teilnehmers (siehe "Einen Teilnehmer an einem Webinar anmelden").

Anwesenheitsdauer der anwesenden Teilnehmer eines Webinar-Termins auslesen

Endpunkt: GET /api/webinars/[Webinar-Id]/[Webinar-Termin-Id]/participant-attendance

Die Rückgabe der Anwesenheitsdauer wird in der Form eines JSON-Objektes ausgeliefert, wobei die E-Mail Adresse des Teilnehmers als Property und die Anwesenheitsdauer in Minuten als Wert gesetzt werden.

Die Webinar-Termin-Id des Termins, für welchen die Anwesenheitsdauer ausgelesen werden soll, kann über den API Endpunkt `Ein einzelnes Webinar auslesen` ausgelesen werden.

Beispiel einer Rückgabe im Erfolgsfall

```
{
  "success": true,
  "attendance": {
    "max.mustermann@example.com": 12,
    "john.doe@example.com": 31,
    ...
  }
}
```

Aufzeichnungen

Alle Aufzeichnungen auflisten

Endpunkt GET /api/recordings

Folgende GET-Parameter können dem API Endpunkt zwecks Paginierung der Ergebnisse übergeben werden:

Parameter	Datentyp	Benötigt	Beschreibung
offset	UInt	×	Index des ersten Elementes, dass ausgegeben werden soll (0 entspricht erstem Element der Liste)
limit	UInt	×	Anzahl der ausgegebenen Elemente

Bitte beachten, dass die Erstellung der Aufzeichnungen Zeit in Anspruch nimmt und Aufzeichnungen nicht unmittelbar nach dem Webinar zur Verfügung stehen. Es dauert in der Regel etwa die Hälfte der Zeit, die das Webinar gedauert hat. Aus diesem Grund kann es sein, dass Sie in Ihrer Anwendung - je nach Anwendungsfall - einen Polling-Mechanismus zur implementieren müssen, der auf das Vorhandensein von Aufzeichnungen zu einem Webinar prüft. Wir empfehlen hierbei einen Zeitintervall von 10 Minuten.

Download URL einer einzelnen Aufzeichnung ausgeben

Endpoint GET /api/recordings/[Recording-Id]/download-url

Gibt eine URL zurück, mit der die Aufzeichnung als MP4 Datei heruntergeladen werden kann. Die Recording-Id kann über den Endpunkt GET /api/records ausgelesen werden. Bitte beachten Sie, dass die Download-URLs nur zeitlich begrenzt gültig sind. In der Regel sind die URLs ca. 2 Stunden gültig.