

ITEM Course Manager API

ITEM Course Manager stiller systemets informationer om kurser og forløb til rådighed for andre systemer igennem et REST API. Samme API gør det muligt at oprette kursustilmeldinger og leads fra andre systemer, typisk fra et website.

Begreber

ITEM Course Manager opererer bl.a. med følgende begreber:

Serie. En helt overordnet kursuslinje, f.eks. "Firmakurser" eller "Uddannelser". Svarer til en mappe i et filsystem.

Kategori. Det fælles emne for en samling kurser – det kan f.eks. være "Design" eller "Organisering, ledelse og samarbejde". Svarer til en mappe i et filsystem.

Kursus. Overordnet kursustype, f.eks. "Ledelse i praksis" eller "Virksomhedsøkonomi". Har en kort beskrivelse, pris ekskl. moms, oplysning om udbyder og leverandør m.m.

Kursusforløb. Det samme som et hold, dvs. en faktisk instans af et kursus - med start- og slutdato, fremmødedato(er), lokation, underviser(e), antal pladser m.m. Hvert forløb har en unik forkortelse, som indeholder information om kursustype, id, startdato, lokation og sprog, f.eks. LEDELSE-2800-270817-AAR-DA (kursusforløb i Ledelse, 27. august i Aarhus – på dansk).

Kursusdato. En dato, hvor der foregår "noget" på et kursus – typisk en form for fremmøde. Anvendes på kursusforløb, der varer flere dage. Hver kursusdato indeholder information om dato, mødetidspunkt (f.eks. 9:00-16:00) samt et eller flere tags, der f.eks. fortæller om der er eksamen, virtuelt fremmøde, kick off e.l.

Hierarkisk forholder ovenstående begreber sig sådan til hinanden:

Firmakurser (serie)

- └ Organisering, ledelse og samarbejde (kategori)
 - └ Ledelse i praksis (kursus)
 - └ LEDELSE-2800-270817-AAR-DA (kursusforløb)
 - └ 27-08-17 (kursusdato)
 - └ 28-08-17 (kursusdato)
 - └ LEDELSE-2801-210917-KBH-DA (kursusforløb)
 - └ 21-09-17 (kursusdato)
 - └ 22-09-17 (kursusdato)
 - └ Organisation og kommunikation (kursus)
 - └ ORGKOM-2802-220717-ODE-DA (kursusforløb)
 - └ 22-07-17 (kursusdato)
 - └ 23-07-17 (kursusdato)
 - └ 29-07-17 (kursusdato)

osv.

Tilmelding. En tilmelding til et bestemt kursusforløb. Indeholder information om kursisten, evt. virksomhed, status (ny, godkendt, annulleret m.fl.) og meget andet.

Lead. En interessetilkendegivelse fra en person. Gælder typisk et bestemt kursus, evt. på en bestemt lokation. Indeholder information om personen og evt. kursus/lokation.

Dataformat

Alle data i API'et er i JSON format. Her er et eksempel på, hvordan data fra et kursus kan se ud:

```
"course": {
  "id": 3,
  "name": "Ledelse i praksis",
  "abbreviation": "LEDELSE",
  "description": "Faget er et af de obligatoriske fag på akademiuddannelsen i
    ledelse.I faget arbejder du med dig selv som leder. Du får indblik i
    forskellige lederroller og bliver bevidst om din egen ledelsesstil.
    Du får også viden om motivation og om, hvordan du bedst kan motivere
    dine medarbejdere.",
  "category": {
    "id": 1,
    "name": "Organisering, ledelse og samarbejde",
    "series": {
      "id": 1,
      "name": "Firmakurser",
    }
  },
  "price": 12900.0,
  "active": true,
  "type": 0
}
```

Metoder

Alle API-kald kræver et brugernavn og en adgangskode, som skal overføres i en basic authentication header i hvert request. Adgangen til API'et kan endvidere være begrænset til bestemte IP-adresser.

Den absolutte URL til et API-kald sammensættes på følgende måde:

`https://[domæne][metode]`

Hvor [domæne] er det domæne, hvor ITEM Course Manager er sat i drift, og [metode] er en af metoderne i nedenstående liste.

Bemærk, at nogle metoder kræver parametre, typisk ID'er på entiteter. Disse er noteret i krøllede parenteser og med information om datatypen, f.eks. {courseId:int} – hvilket betyder et kursus-ID af typen integer. Et spørgsmålstegn efter datatypen betyder, at parameteren er valgfri, f.eks. {daysAhead:int?}.

API'et **skal** kaldes via en sikker forbindelse (https) og fra serverside – det understøtter ikke AJAX CORS.

Metoderne er af typen GET, hvis ikke andet er oplyst. Ved POST eller DELETE metoder returneres et statusobjekt med oplysninger om status, evt. fejlkoder samt returværdi(er).

Serier og kategorier

/api/series

Returnerer alle serier og tilhørende kategorier.

/api/categories

Returnerer alle kategorier.

/api/series/{seriesId:int}/courses

Returnerer alle kurser i serien med det angivne *seriesId*.

/api/categories/{categoryId:int}/courses

Returnerer alle kurser i kategorien med det angivne *categoryId*.

/api/series/{seriesId:int}/instances/{daysAhead:int?}

Returnerer alle fremtidige kursusforløb i serien med det angivne *seriesId*. Hvis *daysAhead* er angivet, vises der kun kursusforløb med start inden for det angivne antal dage – og ellers for de kommende 60 dage.

/api/categories/{categoryId:int}/instances/{daysAhead:int?}

Returnerer alle fremtidige kursusforløb i kategorien med det angivne *categoryId*. Hvis *daysAhead* er angivet, vises der kun kursusforløb med start inden for det angivne antal dage – og ellers for de kommende 60 dage.

Lokationer

/api/mainlocations

Returnerer alle hovedlokationer (typisk byer), f.eks. København, Odense, Aarhus.

/api/locations

Returnerer alle lokationer (kursusadresser), f.eks. navn, adresse, kortkoordinater mv. på kursusfaciliteterne i København, Odense og Aarhus.

/api/mainlocations/{mainLocationId:int}/instances/{daysAhead:int?}

Returnerer alle fremtidige kursusforløb på hovedlokationen med det angivne *mainLocationId*. Hvis *daysAhead* er angivet, vises der kun kursusforløb med start inden for det angivne antal dage – og ellers for de kommende 60 dage.

/api/locations/{locationId:int}/instances/{daysAhead:int?}

Returnerer alle fremtidige kursusforløb på lokationen med det angivne *locationId*. Hvis *daysAhead* er angivet, vises der kun kursusforløb med start inden for det angivne antal dage – og ellers for de kommende 60 dage.

Kursusformer

/api/coursetypes

Returnerer alle kursusformer: 0=Fysisk fremmøde, 1=Online selvstudium, 2=Online med instruktør, 3=Fysisk fremmøde + Online selvstudium, 4=Fysisk fremmøde + Online med instruktør.

Kurser

/api/courses

Returnerer alle kurser.

/api/courses/{courseId:int}/instances/{daysAhead:int?}

Returnerer alle fremtidige kursusforløb tilhørende kurset med det angivne *courseId*. Hvis *daysAhead* er angivet, vises der kun kursusforløb med start inden for det angivne antal dage – og ellers for de kommende 60 dage.

[POST] /api/courses/import

Opretter kurser, kursusforløb og kursusdatoer ud fra en JSON-model. Alle entiteter SKAL have en fremmed-nøgle, dvs. et unikt ID fra det system, de kommer fra. Allerede oprettede entiteter springes helt over og opdateres altså IKKE.

Krævet felt

model: Se bilag A: Importmodel

Returnerer komplet statusinformation i JSON-format.

TILFØJET NOVEMBER 2017

[DELETE] /api/courses/{courseId:int}

Sletter kurset med det angivne *courseId*. Kurset må IKKE have nogen fremtidige kursusforløb, ellers kan det ikke slettes.

Kursusforløb

/api/instances/{daysAhead:int?}

Returnerer alle fremtidige kursusforløb. Hvis *daysAhead* er angivet, vises der kun kursusforløb med start inden for det angivne antal dage – og ellers for de kommende 60 dage.

TILFØJET NOVEMBER 2017

/api/instances/{instanceIdList:string}/enrollments

Returnerer alle tilmeldinger på hvert kursusforløb, der er angivet i *instanceIdList*, som er en kommasepareret liste med ID'er (af typen int) på kursusforløb.

TILFØJET NOVEMBER 2017

[DELETE] /api/instances/{instanceId:int}

Sletter kursusforløbet med det angivne *courseInstanceId*. Kursusforløbet må IKKE have nogen tilmeldinger, ellers kan det ikke slettes.

Pladser og reservationer

`/api/instances/{instanceIdList:string}/seats`

Returnerer antal ledige pladser for hvert kursusforløb, der er angivet i *instanceIdList*, som er en kommasepareret liste med ID'er (af typen int) på kursusforløb.

`[POST] /api/instances/{instanceIdList:string}/reserve`

Reserverer en plads på hvert kursusforløb, der er angivet i *instanceIdList*, som er en kommasepareret liste med ID'er (af typen int) på kursusforløb. Returnerer et reservationsnummer (GUID) for hvert kursusforløb til brug ved tilmelding. Gælder typisk 30 minutter (kan konfigureres).

`[DELETE] /api/reservations/{reservationIdList:string}`

Annulerer hver reservation, der er angivet i *reservationIdList*, som er en kommasepareret liste med reservationsnumre (af typen GUID).

`[POST] /api/reservations/{reservationIdList:string}/renew`

Forlænger hver reservation, der er angivet i *reservationIdList*, som er en kommasepareret liste med reservationsnumre (af typen GUID). Forlængelsen medfører, at gyldighedsperioden starter forfra.

Tilmeldinger

ÆNDRET MAJ 2019

`[POST] /api/instances/{instanceId:int}/enroll/form`

Opretter en tilmelding på kursusforløbet med det angivne *instanceId*. Returnerer et objekt med status – samt unikt ID på tilmeldingen (parameters.enrollmentId) og unikt ID på kursisten (parameters.id), hvis tilmeldingen er gennemført med succes.

Krævede felter

firstNames: Fornavn(e)

lastName: Efternavn

email: E-mail (anvendes også som brugernavn)

Valgfrie felter

address: Adresse

postalCode: Postnr.

city: By

phone: Telefon

dateOfBirth: Fødselsdato i formatet dd-mm-yy, f.eks. 01-07-79

reservationId: Reservationsnummer (GUID)

languageCode: Ønsket sprog (f.eks. "da-DK" eller "en-GB" – bruges ved sprogvælger på kursusforløb)

address2: Evt. sted (del af adresse)

TILFØJET NOVEMBER 2017

enrollmentType: Heltal, der specifikt angiver tilmeldingstypen (1=Ledig, 2=Virksomhed, 3=Privat). Detekteres normalt automatisk ud fra de øvrige data på tilmeldingen, men kan altså også sættes specifikt.

enrollmentDate: Dato og tid i en-US format (yyyy-MM-dd HH:mm:ss). Sæt kun, hvis tilmeldingstidspunktet IKKE skal være dags dato og det aktuelle tidspunkt.

status: Evt. status på tilmeldingen (heltal - se statuskoder i bilag B: Status på tilmelding). Sæt kun, hvis status IKKE skal være 1: Ny.

invoiced: Evt. flag (bool, dvs. true/false), der fortæller, om tilmeldingen er faktureret.

emailTemplateId: ID på den skabelon, der skal bruges til at sende e-mail i forbindelse med tilmelding. ID slås op i Course Manager under Indstillinger. Sæt ID = 0 for IKKE at sende e-mail ved tilmelding.

TILFØJET JANUAR 2018

overwriteExisting: Flag (bool, dvs. true/false), der fortæller, om en evt. eksisterende tilmelding må overskrives. Hvis flaget er sat til true og der findes en ikke allerede godkendt tilmelding på kursistens e-mail adresse i forvejen, vil den eksisterende tilmelding blive slettet (*soft deleted*) og en ny tilmelding vil blive oprettet med to noter: En, der fortæller om kursistens stamdata før og efter tilmelding og en, der fortæller, at den aktuelle tilmelding har overskrevet en tidligere tilmelding.

Krævede felter ved tilmelding gennem virksomhed

companyVAT: Virksomhedens CVR-nr.

companyName: Virksomhedens navn

companyContact: Navn på kontaktperson i virksomheden

companyContactPhone: Kontaktpersonens telefonnr.

Valgfrie felter ved tilmelding gennem virksomhed

companyAddress: Virksomhedens adresse

companyAddress2: Evt. sted (del af virksomhedsadressen)

companyPostalCode: Virksomhedens postnr.

companyCity: By, hvor virksomheden ligger

companyAccountNumber: Virksomhedens kontonr. i ERP e.l.

companyContactEmail: E-mail på kontaktperson i virksomheden

TILFØJET NOVEMBER 2017

companyVATCountryCode: Landekode på virksomhedens CVR-nr. Standard: DK

TILFØJET NOVEMBER 2017

/api/enrollments/{enrollmentIdList:string}

Returnerer alle tilmeldinger, der er angivet i *enrollmentIdList*, som er en kommasepareret liste med ID'er (af typen int) på tilmeldinger.

TILFØJET NOVEMBER 2017

/api/students/{studentId:int}/enrollments

Returnerer alle tilmeldinger for den kursist, der er angivet i *studentId*.

Leads

[POST] /api/leads

Opretter et nyt lead. Et lead kan udtrykke interesse for et specifikt kursus (oplyst i et kursus-ID), en specifik kategori (oplyst i et kategori-ID) eller for et valgfrit emne (oplyst som ren tekst).

Krævede felter

leadName: Navn

leadEmail: E-mail

leadInterestType: Heltal, der angiver leadtypen (1=Kursus, 2=Kategori, 3=Valgfrit emne)

leadCourseId: ID på kursus
ELLER leadCategoryId: ID på kategori
ELLER leadCustomInterest: Tekst

Valgfrie felter

leadAddress: Adresse
leadPostalCode: Postnr.
leadCity: By
leadPhone: Telefon
leadLocation: Navn på ønsket kursussted
leadCompany: Navn på virksomhed
leadComment: Evt. kommentar

Brugere

TILFØJET NOVEMBER 2017

`/api/users/{userId:int}`

Returnerer den bruger, der er angivet i *userId*.

Automatisering

`api/auto/{autoTaskAlias}(/?executenow=false)`

Afvikler automatiseringsopgaven med det alias, der er specificeret i *autoTaskAlias*. En automatiseringsopgave er en systemopgave, som man ønsker at afvikle automatisk, f.eks. en gang i døgnet, ved at kalde den udefra fra et såkaldt *cronjob*. Det vil typisk dreje sig om udsendelse af e-mails og SMS'er.

Når metoden kaldes, oprettes et nyt *job*, og systemet tjekker, om der er brugere, der er omfattet af de regler, der er konfigureret for opgaven. Det kan f.eks. være "find alle godkendte kursister på kurser, der starter om tre dage". Hvis der er omfattede brugere, oprettes der en handling for hver bruger – nemlig den, som er valgt på opgaven. Det kan f.eks. være "send e-mail med skabelonen Velkomstmil".

Hvis den valgfri parameter *?executenow=false* anvendes, vil systemet oprette jobbet og de tilhørende handlinger, men uden at eksekvere dem. Det vil sige, at systemet nu "ved", at det skal udsende e-mail X til bruger Y, men det har ikke gjort det endnu. Parameteren bruges typisk i forbindelse med test.

Metoden returnerer information om det oprettede job, herunder et unikt ID. Se også næste metode.

`api/auto/{jobId:int}/execute`

Hvis en automatiseringsopgave er afviklet med parameteren *?executenow=false*, er jobbet og de tilhørende handlinger kun oprettet, og ikke eksekveret. Det kan efterfølgende eksekveres ved at kalde denne metode, hvor *jobId* er det unikke ID, som blev returneret ved oprettelsen (se forrige metode).

Webhooks

TILFØJET NOVEMBER 2017

`/api/webhooks/broadcast`

Kalder alle konfigurerede webhooks på baggrund af de hændelser, der er logget, siden de sidst blev kaldt. Course Manager samler normalt alle systemhændelser i f.eks. 5 eller 10 minutter ad gangen og udløser derefter alle relevante webhooks på baggrund af de samlede hændelser.

Brug denne metode, hvis du vil udløse alle relevante webhooks lige nu – uden ventetid. Det kan f.eks. være til testformål eller for at sikre, at kursusdata på et website bliver opdateret med det samme.

Anbefalinger ved integration i et website

Opret egen struktur på websitet

Det er sjældent, at den måde, man organiserer tingene på i et internt system, også er den bedste måde at organisere dem på i markedsføringen. Derfor vil det som regel være en dårlig idé at oprette sider med f.eks. kursusserier og kategorier dynamisk ud fra data fra API'et. ITEM Course Manager indeholder heller ikke anden information om hverken serier eller kategorier end navn og ID, da de kun anvendes til at skabe struktur i administrationssystemet.

Derfor: Skal der være oversigtssider på websitet, som f.eks. beskriver en kursusserie eller en kategori, så opret siden manuelt i det CMS e.l. der anvendes og brug de værktøjer, CMS'et stiller til rådighed, til at oprette tekst, billeder, video osv. Brug kun API'et til f.eks. at hente data til en kursuskalender e.l.

Se eksempel på en sådan kursuskalender nederst på forsiden af <https://nyledige.dk/>

Opret en side til hvert kursus

Tilsvarende er det optimalt at oprette en side til hvert kursus i CMS'et. Der vil typisk være behov for en længere beskrivelse af kursets indhold, deltagerforudsætninger, forventet udbytte, testimonials fra tidligere deltagere osv. – alt sammen information, som ikke hører hjemme i kursusadministrationen, men i høj grad hører hjemme i markedsføringen, altså på websitet.

Brug API'et til at hente alle kommende kursusforløb ind på kursussiden. Brug CMS'et til at præsentere de hentede data om lokationer, datoer osv. på en overskuelig måde. Skab selv det tilmeldingsflow, som er ideelt for netop dette website - og brug kun API'et til at tjekke ledige pladser, reservere plads – og til at gennemføre tilmeldingen, når den er bekræftet (og måske betalt) på websitet.

Se eksempel på en kursusside på <https://nyledige.dk/kurser/projektledelse-og-forretningsudvikling/prince2/>

Anvend caching af data – og forny via et webhook

Alle kursusdata bør så vidt muligt caches, når de hentes gennem API'et. I ITEM Course Manager kan man oprette et såkaldt webhook, dvs. at man specificerer en ekstern URL, som systemet skal kalde, hver gang data ændres. Brug det til at invalidere cachen – og anvend gerne en form for "delta", så der ikke reageres på mange, gentagne opdateringer inden for kort tid.

Begrænset antal pladser? Hent pladsinfo og reservér en plads

Hvis der er et begrænset antal pladser på kurserne, er det muligt at hente information om ledige pladser. Denne information skal IKKE caches, men derimod altid så vidt muligt være i realtid. Hent derfor gerne informationen lige før et kursusforløb præsenteres – og igen lige før en tilmelding påbegyndes.

Det er muligt at reservere en plads, når tilmeldingen påbegyndes. Reservationen gælder typisk i 30 minutter og sikrer, at pladsen er til rådighed, når tilmeldingen gennemføres. Hvis der er brug for længere tid, kan reservationen fornyes løbende, f.eks. for hvert trin, slutbrugeren skal igennem i forbindelse med tilmelding.

Dataformatet for pladsinformation ser sådan ud:


```
"seats": [{
  "courseInstanceId": 2849,
  "total": 12,
  "reserved": 1,
  "taken": 5,
  "available": 6
}]
```

Dvs. at på kursusforløbet med det ID, der står i *courseInstanceId*, er der 12 pladser i alt, hvoraf 1 er reserveret og 5 er optagne – og dermed er der 6 ledige pladser tilbage. NB: Hvis egenskaben *total* er *null* er der ingen pladsbegrænsning på forløbet, og så skal pladsinformationen helt ignoreres.

Priser

Prisen på et kursus findes i egenskaben *course.price* på hvert forløb. Den er altid oplyst i DKK ekskl. moms.

Offentlige noter

En kursusadministrator kan oprette en offentlig note på et kursusforløb. Det er en note, der bør vises på websitet i forbindelse med forløbet. Det vil typisk være vigtig information om, at et bestemt forløb afviger fra de andre på en eller anden måde. Hvis der er en offentlig note på et forløb, vil den være tilgængelig i egenskaben *publicNote*.

Leads

Et lead kan i ITEM Course Manager have udtrykt interesse for tre forskellige ting:

1. Et specifikt kursus, f.eks. 'PRINCE 2'. I så fald skal kursets ID være registreret på lead'et.
2. En hel kursuskategori, f.eks. 'Projektledelse'. I så fald skal kategoriens ID være registreret på lead'et.
3. Et emne, beskrevet i ren tekst, f.eks. 'Kurser i SCRUM eller lignende'.

Der kan endvidere tilknyttes en kommentar med ønsker til kursussted, f.eks. 'København'.

Interesse i kursus

Registrering af denne form for interesse kan f.eks. bruges til at implementere venteliste-funktionalitet: Hvis der ikke er plads på et bestemt forløb, kan personen i stedet oprettes som et lead, hvorpå kursets ID registreres. Så er det nemt for administratoren at finde vedkommende frem igen og tage kontakt, når der oprettes nye forløb på kurset.

EKSEMPEL PÅ TEKNISK IMPLEMENTERING

[POST] /api/leads

```
data: {
  leadName: 'personnavn',
  leadEmail: 'e-mail',
  leadInterestType: 1, // = interesse i et specifikt kursus
  leadCourseId: 1234, // et kursus-ID,
  leadAddress: 'adresse',
  leadPostalCode: 'postnummer',
```

```
    leadCity: 'by'  
    leadPhone: 'telefon'  
    leadLocation: 'ønsket kursussted'  
    leadCompany: 'navn på virksomhed',  
    leadComment: 'kommentar'  
}
```

Interesse i kategori

Registrering af denne form for interesse kan f.eks. anvendes som call-to-action for besøgende, som ikke er interesserede i et specifikt kursus, men dog ved, at deres interesse ligger inden for en kendt kategori, som f.eks. Projektledelse.

EKSEMPEL PÅ TEKNISK IMPLEMENTERING

[POST] /api/leads

```
data: {  
  leadName: 'personnavn',  
  leadEmail: 'e-mail',  
  leadInterestType: 2, // = interesse i en kategori  
  leadCategoryId: 3456, // et kategori-ID,  
  leadAddress: 'adresse',  
  leadPostalCode: 'postnummer',  
  leadCity: 'by'  
  leadPhone: 'telefon'  
  leadLocation: 'ønsket kursussted'  
  leadCompany: 'navn på virksomhed',  
  leadComment: 'kommentar'  
}
```

Interesse i emne

Registrering af denne form for interesse kan f.eks. anvendes som call-to-action for besøgende, som er interesserede i et emne, der aktuelt ikke udbydes kurser i.

EKSEMPEL PÅ TEKNISK IMPLEMENTERING

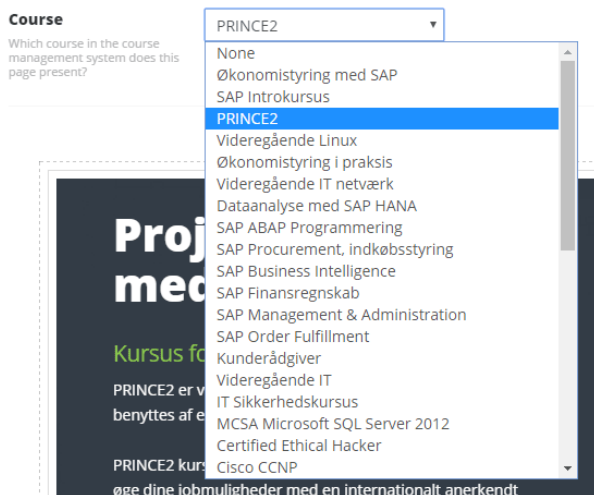
[POST] /api/leads

```
data: {  
  leadName: 'personnavn',  
  leadEmail: 'e-mail',  
  leadInterestType: 3, // = interesse i et emne  
  leadCustomInterest: 'en beskrivelse af emnet i ren tekst',  
  leadAddress: 'adresse',  
  leadPostalCode: 'postnummer',  
  leadCity: 'by'  
  leadPhone: 'telefon'  
  leadLocation: 'ønsket kursussted'  
  leadCompany: 'navn på virksomhed',
```

```
    leadComment: 'kommentar'  
}
```

Byg evt. dine egne "vælgere" i CMS'et

Hvis CMS'et gør det muligt, kan du anvende data fra API'et til at bygge "vælgere", som webredaktøren kan anvende til at vælge, hvilke data, der skal trækkes ind på forskellige sider. Lav f.eks. en "kursusvælger", hvor redaktøren kan vælge, hvilket kursus en side skal hente data fra.



Eksempel på en kursusvælger, der trækker på data fra API'et - her i Umbraco CMS

Sprogvalg på kurser

I ITEM Course Manager er det muligt at oprette kursusforløb, hvor deltagerne selv kan vælge, hvilket sprog, det skal foregå på. En del af de data, der findes på et kursusforløb, ser sådan ud:

```
"isMultiLanguage": true,
"languages": [
  {
    "id": 1,
    "name": "Dansk",
    "languageCode": "da-DK"
  },
  {
    "id": 2,
    "name": "Engelsk",
    "languageCode": "en-GB"
  }
]
```

Hvis egenskaben *isMultiLanguage* er *true*, betyder det, at der kan vælges mellem flere sprog på forløbet. De sprog, der kan vælges imellem, fremgår af arrayet *languages* – og du kan bruge informationerne deri til at bygge en sprogvælger, som kursister kan anvende ved tilmelding. Det er værdien i egenskaben *languageCode*, der i så fald skal sendes med i feltet af samme navn i forbindelse med tilmelding (se *enroll* metoden ovenfor).

Hvis *isMultiLanguage* er *false*, kan du helt se bort fra sprogvvalg.

Bilag A: Importmodel

Den JSON-model, der skal anvendes i forbindelse med metoden `/api/courses/import` har følgende format:

```
{
  'mode': 'create', // pt. eneste understøttede
  'disableWebhooks': false, // true = fyr ikke webhooks på importerede emner
  'courses': [{
    'foreignKey': '120015', // fremmednøgle fra andet system
    'categoryId': 1, // id på kategori fra /api/categories
    'name': 'Dataanalyse med SAP HANA',
    'abbreviation': 'SAP-HANA',
    'description': 'SAP HANA er softwaren, der giver virksomheder
      mulighed for at...',
    'active': true, // false = skjult og uden tilmeldingsmuligheder
    'typeId': 0, // id på kursusform fra /api/coursetypes
    'assessment': 0, // 0 = ingen beståelse, 1 = administrator/instruktør
      må vurdere beståelse, 2 = administrator må vurdere beståelse
    'price': 11499.95, // en-US decimal (pris i DKK)
    'defaultCourseInstanceSeats': 25, // heltal (standard antal pladser)
    'courseInstances': [{
      'foreignKey': '567898', // fremmednøgle fra andet system
      'startDate': '2018-08-06', // en-US dato
      'endDate': '2018-08-07', // en-US dato
      'locationId': 1, // id på lokation fra /api/locations
      'cancelled': false, // true = forløbet er aflyst
      'enableEnrollment': true, // åbent for tilmelding
      'seats': 20, // pladser, overstyrer defaultCourseInstanceSeats
      'hideOnWebsite': false, // (udgår, anvend hiddenInCatalogue)
      'hiddenInCatalogue': false, // true = vis ikke i dataudtræk
      'dates': [{
        'foreignKey': '956578', // fremmednøgle fra andet system
        'date': '2018-08-06', // en-US dato
        'time': '9:00-16:00' // tidsrum som streng
      }],
    },
    {
      'foreignKey': '956579',
      'date': '2018-08-07',
      'time': '9:00-15:00'
    }
  ]
}]
}
```

Bilag B: Status på tilmeldinger

En tilmelding (*enrollment*) kan have en af følgende statuskoder:

- 1 Ny
- 2 Afventer
- 3 Godkendt
- 4 Afvist
- 5 Annulleret
- 6 Flyttet til andet forløb
- 7 Midlertidigt godkendt

Ny: Er lige kommet ind i systemet, og ingen har gennemset tilmeldingen endnu.

Afventer: Er gennemset af en administrator, men den kan ikke umiddelbart godkendes.

Godkendt: Tilmeldingen er godkendt og kursisten er klar til kursusstart

Midlertidigt godkendt: Tilmeldingen er principielt godkendt, og vi lader kursisten starte på kurset - men der mangler stadig et eller andet, f.eks. dokumentation, betaling, kontrakt e.l., man skal være opmærksom på.

Afvist: Tilmeldingen er helt afvist, f.eks. pga. manglende betaling, manglende godkendelse e.l.

Annulleret: Ordren er annulleret og kursisten er afmeldt kurset igen.

Flyttet til andet forløb: Kursisten tilmeldte sig oprindeligt det forløb, vi kigger på, men er efterfølgende blevet flyttet til et andet - til denne status hører altid en note om, hvilket forløb kursisten er overført til.