XML Documentation

An automatic integration of your courses to our website, through XML

What's required from you is that you create a file that generates an XML structure so that fields from your database can be matched with fields in our database. The file should be publicly available on your website so our application can automatically retrieve the information at any time. When setting up your XML, contact us and we will provide you with your Institute ID. [institute-id] must be replaced with the actual numeric id the institute has in our system. On the following pages you will find a specification with technical details on how to set up an XML-file. Below here is some information on how it works when the XML-solution is set up and your courses are published on our website.

1. How it works

When the XML-solution is created according to our specifications, all of your courses can be published and updated on our website. Nightly, our web server calls your web server to acquire a list - as well as any information about the courses you may want to publish on our website.

Update a course, already existing in the XML-file

When you are updating (e.g. prices, dates or text) a course that already is included in your XML => the course will automatically be updated on our website

Remove a course, existing in the XML-file

When you remove a course from your XML => the course will automatically also be deleted on our website

Add a new course to the XML-file

When you add a new course in your XML => the course will be visible after we've manually activated and categorized it for publishing. *

*this applies only if you don't use our category node (learn more below under 3.9): we are manually going through your XML twice per week to ensure that newly added courses are activated and categorized. (If you've added new courses that are not visible on our website, the reason might be that we haven't activated them yet.) If you know, that you've added a big amount of course and want to speed up the process getting them active on our website, please send an e-mail to us and we will provide you with the service of speeding up the process.

If you have any questions, please contact us and we will provide you with needed information.

2.1 XML document outline

In order to be successfully validated and imported, the XML document must strictly follow the XML standard. It is important to note that any other representation of these data won't be considered valid and the document will be rejected by the import engine.

The XML file will be validated according to the schema file published at this address: http://static.educations.com/xml/schemas/xml-import/2.0/xml-import.xsd.

2.2 Use of HTML in fields

If you are using HTML in a field, the HTML code needs to be handled to avoid problems with the XML structure. Content that use HTML should be inserted inside a CDATA block to make sure the HTML code is not validated as XML.

Example:

```
<Description>
<![CDATA[<p><b>Lorem ipsum</b> dolor sit amet, consectetuer adipiscing elit.</p>
<p>Suspendisse molestie <b>odio</b> nec nunc. Duis id est.
<br />Cras risus diam, placerat non, facilisis at, lacinia sed, neque.</p>]]>
</Description>
```

2.3 Explanation to text in [ ]

[num] = numerical value

[text]* = text of unlimited length

[text]n = text with n number of signs

[bool] = booleanskt value ("True" or "False")

3.1 Data formats

Decimal values

Decimal values use the dot "." as decimal separator (e.g. "123456.12")

Date values

Date values follow the ISO 8601 format YYYY-MM-DD where:

- [YYYY] refers to a four-digit year (e.g. 2012)
- [MM] refers to a zero-padded two-digit month of the year between 01 and 12
- [DD] refers to a zero padded two-digit day of the month between 01 and 31

E.g. "2012-08-01"

Time values

Time values follow the ISO 8601 format hh:mm:ss where:

- [hh] refers to a zero-padded hour between 00 and 24
- [mm] refers to a zero padded minute between 00 and 59
- [ss] refers to a zero padded minute between 00 and 59

E.g. “09:00:00”

Combined date and time values follow the ISO 8601 format by concatenating a complete date expression, the letter T as a delimiter, and a valid time expression.

E.g. “2012-06-25T11:55:00”

3.2 Basic structure for a single Institute

Basic structure for a single Institute will look like the snippet (example) below. All the elements presented in this snippet are required. [institute-id] must be replaced with the actual numeric id the institute has in our system. When you are setting up your xml, contact us and we will provide you with your Institute ID.

Example

```
<?xml version="1.0" encoding="utf-8"?>
<informationUpdateBatch xmlns=http://educations.com/XmlImport
        xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
        version="2.0">
        <institute id="[institute-id]">
                <educations>
                        <education />
                        ...
                </educations>
                <locations>
                        <location />
                        ...
                </locations>
        </institute>
</informationUpdateBatch>
```

3.3 Institute nodes

The institute node is composed by two nodes:

1 <locations> that will contain all the institute’s locations

2 <educations> that will contain all the institute’s educations

3.4 Location nodes

A location node allows you to fully describe the locations where the educations will be held. An institute can contain several location nodes; in this case, it is important that the unique identifier of the location is never repeated in the collection.

Example

```
<location uniqueIdentifier="[text] 128" name="[text] 128"
        isFoodProvided="{true|false}" hasAccomodation="{true|false}">
        <!-- Required field -->
        <place>[text] 128</place>

        <!-- Optional fields -->
        <coordinates latitude="[decimal]" longitude="[decimal]" />
        <visitingAddress street="[text] 256" city="[text] 128" co="[text] 128"
                country="[text] 128" zip="[text] 64" />
        <mailAddress street="[text] 256" city="[text] 128" co="[text] 128"
                country="[text] 128" zip="[text] 64" />
        <description>[text]*</description>
        <residentialInformation>[text]*</residentialInformation>
        <contacts telephone="[text] 64" fax="[text] 64"/>
</location>
```

3.5 Education nodes

As for locations, you can define several education nodes and each of them must have a uniqueIdentifier attribute whose content is unique for the institute.

In an education node, you can describe an education by providing its name, the ID of its education type*, and the address of the specific web page of the education on your own web site.

The different education types for **educations.com** are as follows:

62 Bachelor's Degree

63 Master's Degree

65 Doctorate / PhD

64 Diploma Program

1 Language course

71 Study abroad program

133 Summer / Short course

303 Foundation program

195 Associate's Degree

196 Graduate certificate

You can also specify the education's duration (both providing a descriptive text and a computer readable format) and a default price valid for all its events (by specifying the price, the currency, the vat percentage and whether the vat is already included in the given price).

You can also specify up to 15 keywords that will be bound by the search service to the education. Each keyword can be maximum 64 characters long.

Example

```
<education uniqueIdentifier="[text] 256"
        educationTypeID="[numeric]" name="[text] 128">
        <duration text="[text] 64">
                <specific unit="{hours|days|weeks|months|years|semesters}"
                value="[decimal]" />
        </duration>
        <defaultPrice price="[decimal]" vatIncluded="{true|false}"
        vat="[decimal]" currency="{SEK|EUR|USD|GBP|NOK|DKK|CAD}" />
        <link>[url 512]</link>
        <keywords>
                <keyword>[text 64]</keyword>
                ...
        </keywords>
        <events />
        <categories />
        <informationRequestSettings />
        <contentFields />
</education>
```

3.6 Content field nodes

A content field node represents a text property and its content. Each node can either be of type "default" or "custom"

Content field "Default"

If its type is set to "default", you must choose a value for the property "name" among these:

- **description** – a general course description /presentation
- **qualification** – the course target group, also any pre-requisites/ requirements **degree** – certification that participants may receive
- **continuing** – continuing studies of this course (i.e. advanced level etc.)
- **detailedCost** – detailed information about pricing and what's included in the price
- **technicalPrerequisites** – technical requirements (i.e. computer, operating system)

By using these default fields, you're making sure that course information that is found on our site is in the location that users are used to.

Content field “Custom”

If the chosen type is “custom”, the content of the property “name” is free,

but you can specify whether the property can either contain html text.

Example

```
<field xsi:type="custom" name="[text] 128" isHtml="{true|false}">[text]*</field>
<field xsi:type="default" name="{description|qualification|degree|continuing|detai
ledCost|technicalPrerequisites}">[text]*</field>
```

3.7 Event nodes

As an education can define several events, an education node can contain several event nodes. Each of these nodes must be of one of these types:

- LocationEvent
- AreaEvent
- DistanceEvent

These types have both specific and common information. Beside the mandatory unique identifier, for each event the customer can specify:

- the **pace** of that event (a decimal value between 0.0 and 100.0 representing a percentage,
- the **language** of the event
- the **event type** associated to the specific event (this field is required; it is always **7**)
- the **price** of that specific event (the same data as for education’s default price is required)

  information about the begin of the event (see below)

LocationEvent: In addition to these properties, a location event must specify the location where the education is held for that specific event. In order for the document to be valid, the specified location identifier must be already present in the location section explained above.

AreaEvent: An area event isn’t bound to a specific location as the education is generally bound to the specified area (e.g. an area event bound to Germany means that the education is generally available in Germany, not being bound to any specific location).

Example

```
<event xsi:type="LocationEvent" locationUID="[text] 128" eventTypeID="7"*
                    uniqueIdentifier="[text] 128" language="{language}">
        <pace>100</pace>
        <price price="[decimal]" vat="[decimal]" vatIncluded="{true|false}"
currency="{SEK|EUR|USD|GBP|NOK|DKK|CAD}" />
        <start />
```

```
</event>
<event xsi:type="DistanceEvent" eventTypeID="7"
                uniqueIdentifier="[text] 128" language="{language}">
</event>
<event xsi:type="AreaEvent" place="[text] 128" eventTypeID="7"
                uniqueIdentifier="[text] 128" language="{language}">
</event>
```

**eventTypeID is always 7*

Language

A language can be any of the following: Afrikaans, Arabic, Balinese, Basque, Buginese, Bulgarian, Chinese, Croatian, Czech, Danish, Dutch, English, Estonian, Finnish, French, Frisian, German, Greek, Hebrew, Hindi, Hungarian, Indonesian, Italian, Japanese, Javanese, Korean, Latin, Latvian, Norwegian, Persian, Polish, Portuguese, Romanian, Russian, Sami, Sanskrit, Serbian, Slovakian, Slovenian, Somali, Spanish, Sundanese, Swedish, Tamil, Thai, Tibetan or Urdu.

3.8 Start nodes

Each event can specify information about when the education is actually held.

A start node can be of the following types:

Fixed: Must specify the exact day the education is starting. Optionally the start time, the end date and the end time can be specified.

Text: there is no specific data available, just a descriptive text.

Example

```
<start xsi:type="Fixed" startDate="[date]" endDate="[date]"
        startTime="[time]" endTime="[time]" />
<start xsi:type="Text" description="[text] *"/>
```

3.9 Category nodes

If a customer wants to set categories/subjects for his courses, he can define those in the category node of the XML. Our system then matches these category nodes with the existing category structure on our website. For each education you can set up to three categories/subjects using the category node.

Example

```
<categories>
        <category name="[text] 64"/>
        ...
</categories>
```

3.10 informationRequestSettings node

For each education the customer can define up to 5 information request receiver addresses by using the informationRequestSettings node.

Example

```
<informationRequestSettings>
        <emailReceivers>
                <receiver email ="[text] 64”/>
                …
        </emailReceivers>
</informationRequestSettings>
```

If you have any questions when building the XML, please contact your account manager or Ingmar Bertram at ingmar.bertram@kursfinder.de and we will provide you with any needed information.