



Content Submission Guidelines

EPUB2/3 and PDF

[Introduction](#)

[Key Features](#)

[Content Submission Procedure](#)

[Metadata](#)

[EPUB file](#)

[PDF file](#)

[Cover file](#)

Introduction

The PUBLizard Reader also fully supports legacy EPUB2 books, Adobe PDFs and Comic book archives.

Key Features

- HTML5 elements such as videos, audios or any other elements are fully supported
- Full scripting support
- Full CSS3 support
- SVG and any multimedia files
- MathML and TeX supported through [MathJax](#)
- Media Overlays
- Global language support
- Embedded fonts
- Fixed Layout or Reflowable
- Canonical Fragment Identifiers

- Thumbnails
- Full Text and Metadata Searching with [Lucene](#)
- [IMS QTI™](#) support - for online test assessments

All of this is working within a standard browser, making it easy to integrate our Reader in other sites or third party apps, while fully retaining all its functionalities, such as note taking, sharing, highlighting, bookmarking, full text & metadata searching, etc. The EPUB3 books are displayed without any alteration and can be uploaded without constraints.

Content Submission Procedure

The book data is a set of files that must be uploaded to an FTP account. The set of files must include an XML metadata file, a cover image file and an EPUB file or a PDF file.

Although this is not mandatory, to facilitate the detection of data corruption during the file transfer it is recommended to compress the various files to a single ZIP file.

The various filenames should be in the form 'UID'.ext (e.g. 123456.pdf, 123456.epub, 123456.xml where 123456 is the UID).

XML metadata file

The XML metadata file is mandatory. It defines essential metadata about the book and contains references to the content files.

Element name	Mandatory	Type	Description
UID	Yes	Alphanumerical string	The UID is the unique identifier for the book/document in the reader. It can be the ISBN or any other arbitrary identifier, must start with 2 digits and should only contain alphabetical characters or digits.
Title	Yes	String	The book/document title
EANs/EAN	No	String	Additional identifiers (e.g. other ISBNs)
Publisher	No	String	The publisher name
Imprint	No	String	The imprint name
Description	No	String	Free text description
Code	No	String	Classification code - the classification code depends on the partner. If multiple codes are present, the next form should be used.
Codes/Code	No	String	Classification code (if multiple are present)
Authors/Author	No	String	Author name(s)

PublicationYear	No	String	Publication date
Contents/Data	Yes	String	File name for the EPUB or the PDF file
Contents/Cover	Yes	String	File name for the cover file
Rights/Copy	No	String	yes or no (default: yes). Controls whether book content can be copied to the clipboard.
Rights/Print	No	String	yes or no (default: yes). Controls whether book pages can be printed.
Rights/PrintLimit	No	Numeric value	Percentage that controls how much of the book can be printed. Default: 120% (to account for reprints).

Sample:

```
<?xml version="1.0" encoding="utf-8"?>
<Book>
  <UID>9780415221016</UID>
  <Title>Cognitive Science</Title>
  <EANs>
    <EAN>1134598823</EAN>
    <EAN>9781134598823</EAN>
  </EANs>
  <Publisher>Taylor & Francis</Publisher>
  <Imprint>Routledge</Imprint>
  <Description />
  <Codes>
    <Code>science.social_science</Code>
    <Code>science.biology</Code>
  </Codes>
  <Authors>
    <Author>Kolak, Daniel</Author>
    <Author>Waskan, Jonathan</Author>
  </Authors>
  <PublicationYear>2006</PublicationYear>
  <Contents>
    <Data>9780415221016.epub</Data>
    <Cover>9780415221016.jpg</Cover>
  </Contents>
  <Rights>
    <Copy>yes</Copy>
    <Print>yes</Print>
    <PrintLimit>60</PrintLimit>
  </Rights>
</Book>
```

Note: any other metadata can be added and processed if necessary.

EPUB file

The submitted EPUB must be a valid EPUB2 or EPUB3 file; it must pass a validation check by a utility such as [EPUBCHECK](#). If the EPUB3 navigation document is not provided, the EPUB2 NCX file must be present. If both are provided the navigation document will be used.

PDF file

The PDF file should conform to the PDF/A-1b specification (in particular, it must embed all fonts in use). It should not be password-protected and the Content Copying or Extraction restriction should be allowed. It should be bookmarked and the bookmark order must match the page order.

Cover image file

The cover image file must be a jpeg file. The width should at least be of 200 pixels, ideally more than 760 pixels.

Appendix 1: Metadata XML Schema

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema targetNamespace="http://tempuri.org/XMLSchema.xsd"
  elementFormDefault="qualified"
  xmlns="http://tempuri.org/XMLSchema.xsd"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Book">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="UID" type="xs:string">
        </xs:element>
        <xs:element name="Title" type="xs:string"></xs:element>
        <xs:element name="EANs" minOccurs="0">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="EAN" minOccurs="0" maxOccurs="unbounded" type="xs:string"></xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="Publisher" type="xs:string" minOccurs="0"></xs:element>
        <xs:element name="Imprint" type="xs:string" minOccurs="0"></xs:element>
        <xs:element name="Description" type="xs:string" minOccurs="0"></xs:element>
        <xs:element name="Code" type="xs:string" minOccurs="0"></xs:element>
        <xs:element name="Codes" minOccurs="0">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Code" minOccurs="0" maxOccurs="unbounded" type="xs:string"></xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="Authors" minOccurs="0">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Author" minOccurs="0" maxOccurs="unbounded" type="xs:string"></xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="PublicationYear" type="xs:string" minOccurs="0"></xs:element>
        <xs:element name="Contents">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Data" type="xs:string"></xs:element>
              <xs:element name="Cover" type="xs:string"></xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
</xs:sequence>  
</xs:complexType>  
</xs:element>  
</xs:schema>
```