# User Guide for Generate OPF Software

Vision Statement:

### Software to generate the “content.opf” file for all types of ePub files. Software will perform below task to generate well formatted OPF file.

### Reads all the files within the “OEBPS” folder and write the “Unique ID”, “file Path”, “File Type” in the Manifest section of OPF File.

### Ids will be “unique-id” (Ex: id-1,id-2..) and for the html file if file name doesn’t contains “3 digits” value in the file name (Ex: html file name is page1.html), then we will change the single or double digit numbers as triple digit numbers in the id (Ex: page001 for page1.html file).

### Add the updated html file ids in the <spine> section of the opf file with the below order.

### If cover.html/frontcover.html file found in the OEBPS folder, then it will be added as first entry of the Spine section. If these files are not found, then we are adding the spine section as ascending order of page1 to pageEnd.

### If backcover.html/back\_cover.html file found in the OEBPS folder, then it will be added as last entry of the Spine section.

### If you are running for one title, you can use the text field in the UI of the software to update Title Name, Author Name, ISBN, copyright Information,Publisher Name. If you are running for more titles in batch process, you can create a text file with above information by comma separator. (Ex: “OPF Book”,”Dinesh and Subramani”,”9999999999999”,”@2016 copyright rights by Newzel Publishing Pvt Ltd”,Newzel Publishing Pvt Ltd”)

### We will update the above information in the meta data section of OPF file.

### If audio files are found within the OEBPS folder, then script will update the duration of each mp3 files in the <meta-duration> tag of meta data section.

### If any of the files present in the OEBPS folder which are not declared in the OPF file will be written as error report.

### If user wants Guide section, they can update which pages they want in the guide section, we will add the guide section in the OPF based on the user input in the Software UI.

Input Checklists:

* Each Folder name should be “Title Name” inside the Title Folder there should be “OEBPS” Folder
* All the ePub files should be placed inside the “OEBPS” folder.
* File name of the Front Cover Image should be “cover.jpg”. If “cover.jpg” file not found in the OEBPS folder, tool will write the errorLog.
* File name of the Front Cover HTML file should be “cover.html” or “frontcover.html”. If file with either of these name was not found in the OEBPS folder, tool will write the errorLog, if both files are present also throw the error over the script.
* File name of the Back Cover HTML file should be “backcover.html”. Same as above, script will write the error Report.
* File name of the interior Files should be “page001.html,page002.html and so on….”. Same as above, script will write the error Report.
* File name of the Smil File and appropriate HTML files should have same file name.
* User didn’t provide any metadata text file while executing the script, script will keep all the meta information as empty.
* Script will perform the input analysis task first and provide the alert to the user, if you wish to continue the execution, then only script will create the OPF file.

Arguments and its default Values:

Totally there are 8 Arguments/information user can provide in the execution.

Mandatory Field:

Input -> Input path is the mandatory input to execute the software

Below fields are optional:

ePub Version -> Version of the ePub, such as “2.0”, “3.0”. User didn’t give any value in it, software will update version as “3.0”.

ePub Title Name -> Title of the book. User didn’t enter the title name, software will update the folder name as title.

ePub ISBN -> ISBN Number of the book. If User didn’t enter the ISBN number, software will update “9999999999999”.

ePub Language -> Language of the book. If User didn’t enter the Language of the book, software will update “en-US” as default.

ePub Creator -> Creator of the Book. If User left this field blank, software won’t update <dc:creator> tag in metadata.

ePub Publisher -> Publisher Name of the Book. If User left this field blank, software won’t update <dc:publisher> tag in metadata.

ePub Copyrights Information -> Copyright Information of the Book. If User left this field blank, software won’t update <dc:rights> tag in metadata.

Audio Book -> True/False Radio Button. Default is true.

ePub Orientation -> portrait/landscape/auto. Default value is “auto”.

GuideRequired -> Guide Required for this book, True/False. Default is False.

Software Architecture:

Output Samples:

Features:

* Well Formed Opf File will be generated.
* Software have a feature to import the metadata csv file of all your titles.
* User can view the CSV in the Software Interface and they can modify the data if needed.
* Software will check the title name (folder Name) and update the appropriate title meta information in the OPF File.
* If User don’t have metadata CSV, they can fill the data simply in the User Interface of Software.
* **Batch Process** Feasibility, required information can be filled in the User Interface of software for all the titles.
* Software will update all the file information of the ePub and update in the manifest with unique-ID.
* Software will update appropriate media-type in the manifest and also update the media-overlay attribute for the html files which are embedded with audio.
* Software also order the Spine tag and also Guide Tag, based on the USER requirement.
* Software also update the media:class property in the meta data as ‘**-epub-media-overlay-active’**.
* Software also update the audio duration of the each page wise mp3 and also consolidated duration of all mp3 files into the **media:duration** tag in the meta data.
* Software also provide the errorLog, if software fails to get the required data from the input, such as “**OEBPS Folder, Page File name, Cover image file**”.