Personalization in the Interactive EPUB 3 Reading Experience: Accessibility Issues for Screen Reader Users

Barbara Leporini
ISTI-CNR
via G. Moruzzi, 1
56124 PISA (Italy)
Tel. +39 050 6212034
barbara.leporini@isti.cnr.it

Clara Meattini
ISTI-CNR
via G. Moruzzi, 1
56124 PISA (Italy)
Tel. +39 050 6212034
meattini.clara@gmail.com

ABSTRACT
In this paper, we describe the study conducted to investigate accessibility using EPUB 3 with particular focus on interaction via screen reader. A multimedia and interactive EPUB 3 prototype was designed for the purpose. In particular, personalization features based on user preferences were designed to customize the reading experience and enrich the interactive experience. Despite the fact that the EPUB format is based on HTML5, and numerous guidelines for web-based technology can be applied to overcome accessibility barriers, several issues still exist with the current standard EPUB 3 when accessing via screen reader. This study contributes to digital publishing for assistive technology and reading application development by promoting accessibility in EPUB interaction. Thus, some considerations and suggestions in that direction end the paper.

CCS Concepts
• **Human-centered computing~Accessibility** • *Human-centered computing~Human computer interaction (HCI)* • Applied computing~Hypertext / hypermedia creation

Keywords
Accessible publishing; EPUB 3; e-book; screen reader.

1. INTRODUCTION
Electronic books (eBooks) are becoming more widely available and widely used. By engaging directly with the individual reader, interactive e-books offer the chance to widen knowledge, improve learning as well as enhance the entertainment value of the book [5], [10] and [22]. E-books represent great opportunities especially for those who have difficulty in reading printed versions, such as visually-impaired people.

With the recent release of reading applications, authors are able to develop highly interactive e-books. Interactive documents can offer the advantage of turning the passive learning experience that is reading the textbook into a more interactive engagement that may reap the benefits of active learning [10].

W4A '19, May 13–15, 2019, San Francisco, NY, USA

ACM ISBN 978-1-4503-6716-5/19/05…$15.00
https://doi.org/10.1145/3315002.3317564

In this work we investigate accessibility issues when using EPUB 3, with particular attention to the interaction with the contents. To this end, we designed and developed an interactive eBook with multimedia content and interaction elements to enhance the reading experience. We refer to an 'interactive eBook' in terms of an eBook whose contents change based on how the user engages with it. To this end, typical user interface elements like checkboxes, buttons and links have been used to enable the personalization of the reading experience. We especially focused on those functions and tasks able to alter the contents according to the user's preferences and actions.

The interactive eBook was developed in EPUB 3 format. EPUB stands for Electronic PUBlication and is a format primarily used for digital books. The EPUB format was created by the International Digital Publishing Forum (IDPF) [13]. The EPUB 3 was chosen primarily because it is based on html, css and JavaScripts. This means that existing accessibility guidelines and techniques – such as WCAG 2.1 [25], WAI Aria (Accessible Rich Internet Applications) [24], and those proposed in [17] and [23] - can be applied to the user interface design during the development process in order to obtain an appropriate interaction with the assistive technology.

The aim of our study is to investigate whether reading applications as well as assistive technology are sufficiently mature to effectively support users of EPUB 3 in particular for interactive activities. We specifically focus on visually-impaired people who use screen reading software as assistive technology to access information services and applications. To this end, some inspection tests were conducted based on the EPUB 3 prototype. The aim was to evaluate accessibility support, especially with regard to interactive activities via screen reader. The evaluation results can assist in understanding the main accessibility issues which still exist, despite numerous guidelines and criteria available in the literature, and the EPUB standard being proposed some years ago.

This work offers a contribution to the digital publishing field for developers of assistive technology, e-readers and eBook reading devices with regard to accessibility support for effective EPUB 3 reading. The research questions we formulated can be summarized as: (1) Are the reading applications suitable for supporting an interactive EPUB 3? (2) Is the screen reader assistive technology mature enough to support the interactive EPUB 3? For the first question, we plan to investigate if popular reading applications are able to perform properly common

customization functions and interactive activities, such as (a) selection of User Interface (UI) control elements (e.g. checkboxes), (b) accomplishment of specific scripts like those used for changing color and fonts directly in the eBook, and (c) visualization of dynamic contents updated in accordance with the user's preferences. The second research question is aimed at evaluating how the screen reader assistive technology works with the (a) interaction UI components (e.g. checkboxes and buttons), (b) "live content" which is dynamically added to the eBook pages according to the user's actions, and (c) multimedia audio contents activated by the user.

The paper is organized as follows: section 2 introduces a background to the field, and section 3 the methodology used to conduct the study. Next the EPUB 3 prototype design and development are described in sections 4 and 5 respectively. Sections 6 and 7 present and discuss the results obtained from the tests conducted to evaluate how reading applications and screen readers work with EPUB interactive features. Some considerations and suggestions end the paper.

2. RELATED WORK

Numerous studies and researchers have been investigating accessibility in digital publishing and web-based technology for several years. A variety of methods and tools have been proposed to make a document or eBook accessible to everyone, including visually-impaired people [6], [8], [9] and [15]. The Daisy consortium, for instance, promotes, develops and maintains open international DAISY (Digital Accessible Information System) Standards for documents. This format was designed to provide eBooks accessible in both audio and text versions. In 2011, EPUB3 (DAISY 4 distribution format) which uses HTML 5, CSS and JavaScripts was approved as Final Recommended Specification by IDPF (International Digital Publishing Forum) [13]. Consequently, the publishers' standard format and accessibility format were integrated to achieve accessibility in the mainstream eBook industry [14]. Nevertheless, accessibility issues are still encountered when reading via assistive technologies, such as a screen reader, on both desktop and mobile platforms [3] and [4]. Several studies have focused on accessibility and usability for reading activities. The study in [12] discusses text-customization needed by visually-impaired users in order to read PDF documents. Instead, the work in [21] suggests an approach to personalize EPUB visualization so as to adapt the rendering for sighted, visually-impaired and blind people. The results of the studies have encouraged better text customization functionalities in reading tools. Other works propose specific reading tools or audio books for blind people, such as those described in [16] or [18]. Our approach is aimed at exploring new opportunities to use the widely-used EPUB format with everyday reading tools.

Interactive eBooks have been explored in various studies in the literature. [19] proposes an interactive eBook for students. It incorporates a number of active components such as video, code editing and execution, and code visualization as a way to enhance the typical static electronic book format. In [11] there is a description of an interactive HTML5-based digital book, which introduces a first proposal for interactive activities aimed at involving students in more activities. While these studies introduce the usefulness of interactivity in an eBook, they do not consider visually-impaired people and interactive features included directly in the EPUB format. Moreover, they do not consider how the proposed eBook works with everyday reading tools. Our study is aimed at investigating the accessibility of an interactive eBook with widely-used reading applications, while personalizing the content through interactive features included in the EPUB 3 format.

3. METHOD

In this work we designed an interactive EPUB 3 prototype, because to the best of our knowledge, there is no interactive EPUB including HTML form elements (i.e. with checkboxes, buttons, and other rendering customization, etc.) available on the market. During our study, we searched for some existing interactive eBooks on the Web, we also contacted various publishers, but we were able to find or obtain only some EPUB 3 e-books which are instead EPUB 2. The interactive components we found were just related to some animations through gifs or other graphical effects.

In order to design an interactive EPUB 3, we took an existing PDF document which was used as source content. So as to make our eBook prototype more interactive, features and functionalities for personalizing the reading experience were added.

To develop the eBook in the EPUB 3 format, we used (1) the Sigil software to manipulate the document[1], (2) the HTML5, CSS and JavaScripts to add accessibility features and interaction functionalities, and (3) WAI-ARIA techniques [24] to address some accessibility features. Sigil is a free multi-platform EPUB eBook Editor with EPUB 2 support, and with limited EPUB 3 support. In fact, the plug-in EPUB3-itizer[2] was installed on Sigil to handle EPUB 3 rather than EPUB 2. However, many features related to the interaction properties had to be added by working directly on the code. Some WAI-Aria roles and properties were also added to the code by hand. WAI-ARIA is a specification written by W3C, which defines a set of additional HTML attributes that can be applied to elements so as to improve accessibility wherever it is lacking. WAI-ARIA is a technology that can help by adding in further semantics that browsers and assistive technologies can recognize and use to let users know what is going on [7].

In order to evaluate the developed prototype, we selected some tools and reading applications (see section 5 for more details). Since the final prototype was found to offer limited accessibility via screen reader, we decided not to consider the inspection evaluation, and therefore not go ahead with user testing.

4. THE PROTOTYPE

Our goal was to develop an interactive accessible eBook with multimedia and personalized contents: interactive table of contents, images, sounds, hyperlinks, and contents which adapted to the user's preferences and actions. Thus, in designing our prototype, we included some features and functions to include in the eBook in order to make it dynamic and interactive. To this end, we decided to develop the prototype in the EPUB 3 format, which can support interaction properties as it is based on html, css and JavaScripts.

[1] Sigil editor, https://sigil-ebook.com/

[2] EPUB3itizer, https://www.mobileread.com/forums/showthread.php?p=2973066

4.1 EBook Description

We decided to make use of an existing document from the Siena municipality website. The document entitled "A spasso tra le ricette senesi" – 'Strolling with dishes from Siena- provides visitors with information about typical local dishes (including recipes) and characteristic walks around Siena, designed to burn off the calories consumed from a portion of the dish. The digital book is composed of 16 chapters: 8 chapters related to traditional Sienese dishes, each subdivided into sections on Ingredients, Preparation, Origins, Nutritional properties, Nutritional values, how to burn off calories by doing aerobic exercise. In addition 8 chapters describe recommended itineraries in the historic center of Siena. The document did not contain any multimedia content and was completely static. The reader could just move sequentially through the document.

Our idea was to design an enriched eBook by linking the dishes with the related walk, and add some multimedia and personalization features to adapt the reading and contents to the user's preferences. In short, the user should be able to read about (1) all the dishes and their recipes; (2) all the walking itineraries available in the town; (3) a specific walk together with the associated dish; (4) a< specific dish and the recipe together with the associated walk; (5) some details and personalized information based on the user's individual actions and preferences.

The eBook should be accessible to everyone, including people who use assistive technology. Consequently accessibility criteria and guidelines were considered when designing and developing the prototype.

4.2 Procedure

In designing the eBook, the main content was extracted from the PDF document. Next, other elements were added to create the eBook.

The procedure followed for designing and developing the interactive EPUB 3 prototype can be summarized as:

Step 1: Design of main content
Step 2: Design of interactive features
Step 3: E-Book development and application of accessibility requirements
Step 4: Code evaluation
Step 5: Prototype evaluation via apps and AT

The interactive EPUB 3 was created starting from a new project in Sigil (release 0.9.9). The content, multimedia components as well as the interactive features were added to the EPUB project. Figure 1 shows an example of a dish contained in the EPUB prototype.

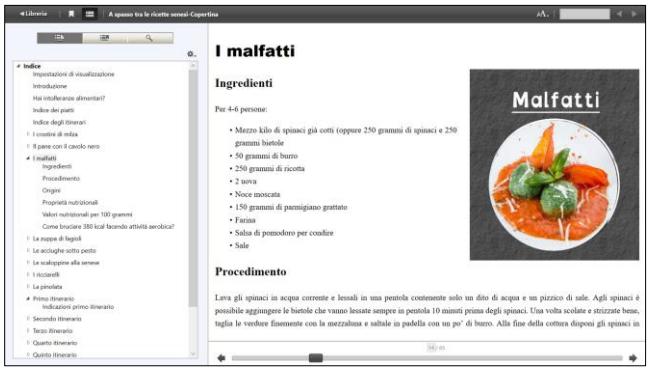


Figure 1 – EPUB prototype example: main book structure and a recipe

The main components considered in the eBook content preparation were:

- **Textual contents**: The main content was extracted from the source PDF and copied by a simple cut-and-paste into the Sigil project. We considered 'cut-and-paste' since it is a very common task used when working on a document, especially for contents coming from different sources.
- **Images**: Some better-quality images with very similar subjects to those available in the source document were added in order to improve the graphical user experience. Images were retrieved from the internet, and then processed for copyright reasons as well as to improve the quality of rendering. Every image was provided with an alternative description for screen reader users.
- **Audio**: 3D sounds were prepared and recorded in order to enhance reader experience, especially for visually-impaired users. The sounds, in mp3 format, are related to specific places and situations mentioned in the eBook. Spoken words and environment sounds were used for the audio tracks.

4.3 Interactive Features

To make the eBook interactive and dynamic, some features and functions for personalizing and interacting with the contents were designed and developed. In proposing the main interactive features, we considered how the contents could be enriched and personalized according to the user's preferences and actions. The main interaction properties we aimed to address can be summarized as:

a) **Easy navigation**: The eBook contains a wide variety of information (e.g. recipes for dishes and itineraries for the city) which is interconnected. However, the reader is forced to read the items individually. Switching from one item to another could require some effort by the user. Instead if prototype reading and switching between items, e.g. a dish and the related calorie-burning walk can be accessed via a linking connection, interaction with the contents is greatly improved. This is the reason why some navigation features were added to the eBook.

b) **Additional information and details**: Terms or contents can be enriched with detailed descriptions or extra details. Adding links to this information can improve the user experience.

c) **Personalization**: Customizing the content according to the user's preferences and choices can make reading more engaging and informative. For example, if the recipe for a dish contains a particular allergen, receiving an alert to indicate its inclusion in the ingredients will be very helpful.

With these interactive goals in mind, we added some specific interactive features to the EPUB prototype. The main interactive features developed are summarized in Table 1.

Table 1 - Interactive features designed in the EPUB 3 prototype

Feature	Design	Development
Interactive index	List of items	Link for each item
Audio descriptions	Mp3 3D audio files	Link on one or more words
Further details in	Text contents shown	

text format	on demand	dynamic block and link on one or more words
Visual rendering preferences	Set the background to yellow	Checkbox to change color
Visual rendering preferences	Set a dyslexic font	Checkbox to select the font
Personalized messages	Content block as an alert	Checkbox selection and block
Map instructions	Graphical map and textual instructions	blocks
Detailed table of contents	List of sections	Link for each section title

In the following, the development of the interactive EPUB 3 prototype is described in detail.

Most features were developed as standard HTML elements, such as links, buttons, checkboxes, item lists and headings. In particular, links, checkboxes, buttons and dynamic blocks controlled via the event onclick and onload were used to develop the interactive features. JavaScripts were added for handling the onclick and onload events, and for sharing variables between HTML pages, which is useful for storing user preferences. The scripts were edited via a simple text editor (NotePad ++[3]) and loaded into the folder /Misc in Sigil.

Accessibility guidelines, such as those proposed by W3C [25] were applied to the EPUB pages. To deal with some more specific accessibility issues, WAI-Aria roles, properties and states were added to the tags.

4.3.1 Interactive Index/Summary

To facilitate navigating between recipes and walks, in the first section of the EPUB a list of the titles for the dishes and itineraries was designed as an interactive index. This was developed by making each item (i.e. each title) a hyperlink to the matching content in the eBook. The list is a type of summary that allows the reader to get an overview of the dishes and walks available within the document. Each link has a clear text that corresponds to the title for the dish or walk. Figure 4 shows the page containing interactive index.

4.3.2 Additional Descriptions and Details

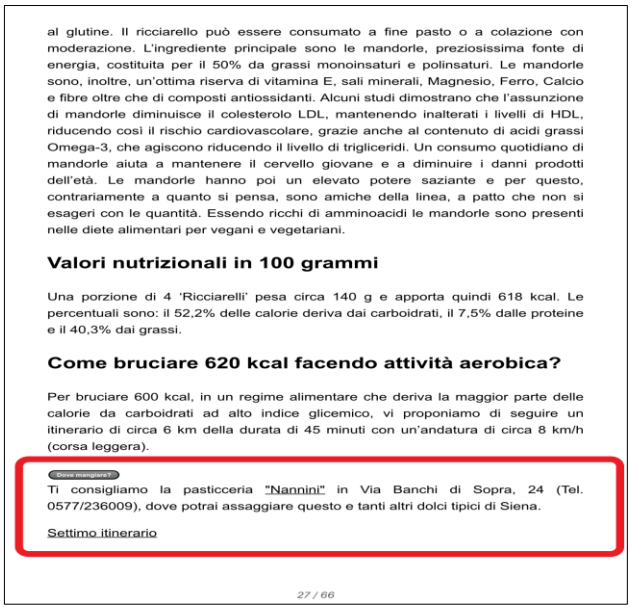
Extra details and curiosities were added to enrich the eBook and make the reading more engaging. Both textual and audio descriptions were designed for this purpose. For example, for each dish audio contents were added to provide information on its origin and typical use. This description was added in an aural format, in order to make the reading multimedia and therefore more enjoyable, particularly for the visually-impaired. In fact, given that eBook content is usually read via a voice synthesizer, an item read aloud by a human voice is more natural and pleasant to hear. Furthermore, this type of description helps listeners differentiate between the main content and additional information.

For some details and particular contents, textual description may be more appropriate. In this case a specific content-info block may be suitable for providing content to the user.

These features were implemented by:

- Mp3 3D audio files. An audio description was recorded in a 3D audio file. This file is associated to the content and a link was added to the text via one or more linked words.
- dynamic block. A new block containing the additional descriptions or details provided in a textual format is shown when clicking on a button or link. The block is not visible on the page and it appears only when clicking on the link or button.

Figure 2 shows the additional descriptions associated to a biscuit ("Ricciarelli"). When the button "Where to eat?" to get additional information is clicked, the dynamic content-info block containing suggestions for pastry shops which sell them is shown (the red circle in Figure 2).


al glutine. Il ricciarello può essere consumato a fine pasto o a colazione con moderazione. L'ingrediente principale sono le mandorle, preziosissima fonte di energia, costituita per il 50% da grassi monoinsaturi e polinsaturi. Le mandorle sono, inoltre, un'ottima riserva di vitamina E, sali minerali, Magnesio, Ferro, Calcio e fibre oltre che di composti antiossidanti. Alcuni studi dimostrano che l'assunzione di mandorle diminuisce il colesterolo LDL, mantenendo inalterati i livelli di HDL, riducendo così il rischio cardiovascolare, grazie anche al contenuto di acidi grassi Omega-3, che agiscono riducendo il livello di trigliceridi. Un consumo quotidiano di mandorle aiuta a mantenere il cervello giovane e a diminuire i danni prodotti dell'età. Le mandorle hanno poi un elevato potere saziante e per questo, contrariamente a quanto si pensa, sono amiche della linea, a patto che non si esageri con le quantità. Essendo ricchi di amminoacidi le mandorle sono presenti nelle diete alimentari per vegani e vegetariani.

Valori nutrizionali in 100 grammi

Una porzione di 4 'Ricciarelli' pesa circa 140 g e apporta quindi 618 kcal. Le percentuali sono: il 52,2% delle calorie deriva dai carboidrati, il 7,5% dalle proteine e il 40,3% dai grassi.

Come bruciare 620 kcal facendo attività aerobica?

Per bruciare 600 kcal, in un regime alimentare che deriva la maggior parte delle calorie da carboidrati ad alto indice glicemico, vi proponiamo di seguire un itinerario di circa 6 km della durata di 45 minuti con un'andatura di circa 8 km/h (corsa leggera).

Dove mangiare?

Ti consigliamo la pasticceria "Nannini" in Via Banchi di Sopra, 24 (Tel. 0577/236009), dove potrai assaggiare questo e tanti altri dolci tipici di Siena.

Settimo itinerario

27 / 66

Figure 2 - Information on the suggested pastry shop (the button "Where to eat?" is clicked)

4.3.3 Visual Rendering Preferences

EBook readers usually offer functionalities to change font and background color so as to set a more appropriate visual rendering when reading. These functionalities are not always available in apps nor particularly practical to use. In order to facilitate user customization - especially for those who are less experienced - these two features were included directly within the EPUB. They are located at the beginning of the eBook in a specific section named "settings" and allow the reader to express their choice and so customize the visual reading experience. This is just an example, a greater number of options could be developed for more varied personalization. Figure 3 shows the yellow background setting.

These two customization features were developed in the EPUB prototype as follows:

[3] NotePad++ https://notepad-plus-plus.org

- Background change checkbox. A checkbox labelled “Set to background yellow color” was added at the beginning of the EPUB. This is just to introduce one customization example for the background, but any other types of feature, or additional options within the feature can be designed.
- Font change checkbox. A selection to set a different font was added to the setting section. In this case the “open dyslexia font” was chosen for the personalization. This customization could be further extended with many other options. In this case the aim was to design a very simple feature for quickly selecting a pre-configured choice. To this end, a checkbox to be selected and unselected was used for the development. The label assigned to the checkbox was “Set a font for easy reading”. In this case only the font changing function was associated to the event onclick for the checkbox. However, a “richer” script could be assigned to the checkbox selection event.

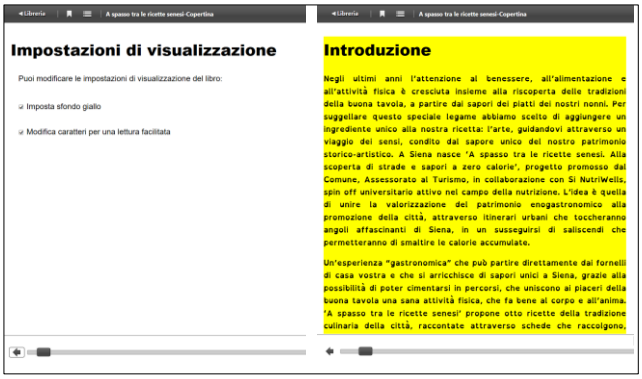


Figure 3 – Setting preferences: the background yellow color is selected

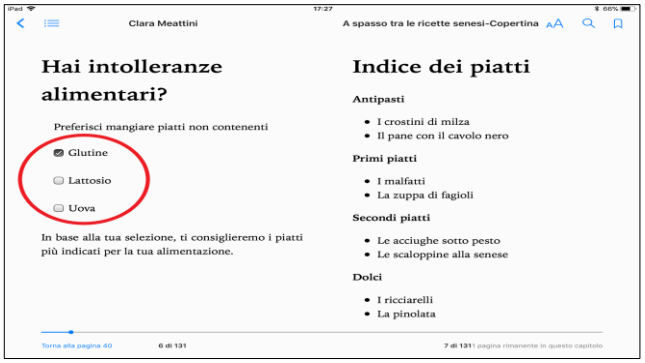


Figure 4 - Checkboxes for food intolerance selection by the user

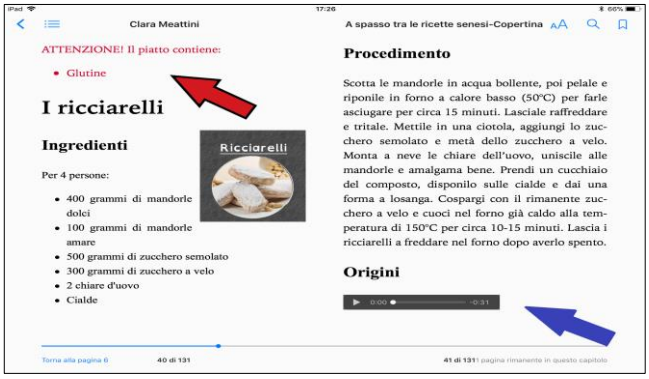


Figure 5 - Alert message for the inclusion of “gluten” in the “Ricciarelli” biscuit

4.3.4 Personalized messages

The eBook contains 8 recipes for traditional Sienese dishes. A user could have an intolerance to certain foods. In our prototype, we created a personalization case with a user preference setting to alert the user about the presence of the allergen in a recipe.

At the beginning of the book the reader is invited to select the undesired ingredients related to their intolerance. According to the ingredients selected, a specific alert is shown when reading a recipe related to a dish.

This customization functionality was designed by:

- Checkboxes. One checkbox for each ingredient is made available relating to the food intolernace. The label of each checkbox is the name of the food.
- block. A dynamic block is shown at the beginning of a recipe. It appears as an alert message.

In the prototype, three checkboxes were developed as food intolerance preferences: gluten, lactose and eggs. According to the selected preferences a short message to alert the user to the inclusion of the allergen in the recipe is displayed. Figure 4 shows an example of food intolerance selection, and Figure 5 illustrates the alert message for the “Ricciarelli” biscuit.

4.3.5 Map instructions

The original book contains a number of descriptions of suggested walking itineraries around the town, in the interactive eBook, a map is displayed at the end of the walk description on demand. To make this map accessible also to screen reader users, textual instructions were provided as well. The reader can view both map and instructions at the end of the description by clicking on the button “Show the map”.

This feature was implemented by:

- List of instructions. The textual instructions were designed as an unnumbered list. Each item is a textual instruction.
- Map. The route associated to the itinerary is shown on a map placed at the end of the instructions list.
- Button and dynamic block. A button labelled “Show the map” was developed at the end of the itinerary description. By activating it, the dynamic block is added to the EPUB.

Figure 6 shows the map and textual instructions related to one itinerary when the button “Show the map” is activated.

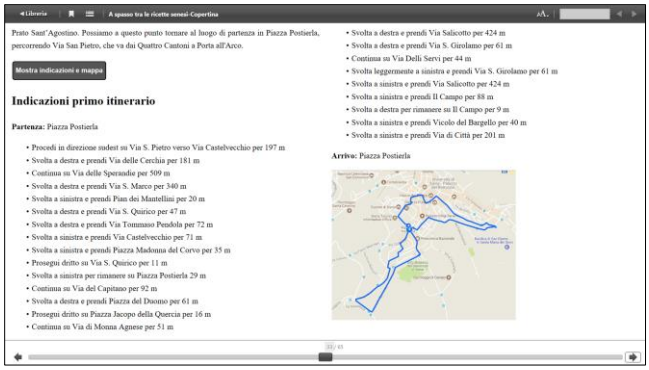


Figure 6 - Example of map and textual instructions related to the first itinerary

4.3.6 Table of contents

Typically, an eBook reader provides a function to display a table of contents, provided that the titles are properly structured and coded. In our prototype, a very structured summary of the contents was designed. This is aimed at allowing the reader to more easily obtain an overview of the contents, including the subsections. Thus, the table of contents is very detailed with the titles of sections and subsections. For example, each recipe is composed of different parts: ingredients, procedure, origins, nutritional properties, nutritional values, and how to burn off calories through aerobic activity. Each of them is a section title so that the contents are very structured into sections in HTML: heading 1 <h1> the name of the dish, and heading 2 <h2> for all the subsection titles. This type of structure is very useful in an HTML document when interacting via screen reader in a browser: by using a specific command (e.g. Insert+F6 for Jaws) the user is able to get a very detailed table of contents, go through the headings via arrow keys, and select one of them via the Enter key. A similar situation occurs in an EPUB document, provided the reading application makes available a function to show "contents". For example, in Figure 1 the table of contents is shown in a hierarchical structure in the Adobe Digital Edition application.

4.4 Code use cases

The development of the interactive properties in the EPUB is mostly based on dynamic contents updated by button and checkbox selection. Dynamic content updates (especially via JavaScripts or when events occur) can lead to problems for the screen reader. To solve accessibility issues via screen reader, WAI-Aria techniques were applied to the tags.

The personalized messages related to food intolerance based on the checkbox selection (see 4.3.4) or the information on the restaurants available for each dish displayed when the button "Where to eat?" (see 4.3.2) are two examples based on dynamic content updates. In these cases, the Aria-live, Aria-Atomic and aria-relevant properties were used in order to make the dynamic content perceivable via screen reader when it becomes visible as a result of button pressure. When the alert message (e.g. "Pay attention, this dish contains…") or the restaurant information content (e.g. "You can eat in the restaurant…") are displayed, the screen reader automatically announces them.

Figure 7 shows a portion of the code used to develop the button, dynamic information block and customization checkboxes.

Button:

```
<input type="button" value="Where to eat?" onclick=
        "document.getElementById('ristorante').style.display = 'block';"
        tabindex="0" id="b1"/>
```

Dynamic block:

```
<div   role="contentinfo" aria-live="polite" tabindex="-1"
       onload="cambio()" id="ristorante"
       aria-relevant="additions text" aria-atomic="true">
```

Checkbox:

```
<input type="checkbox" aria-checked="false" tabindex="0"
        onclick="mostra()" id="Glutine" name="intolleranze"
        aria-labelledby="Glutine" value="Glutine"/>
<label for="Glutine">Glutine</label>
```

Figure 7 - WAI-Aria usage for buttons, dynamic contents and checkboxes

4.5 Code Evaluation

In the EPUB development life cycle, the two tools EPUBCheck[4] and ACE (Accessibility Checker for EPUB)[5] were used to check code accuracy and accessibility compliance. This was performed especially for the interactive features added by hand in the code. At the end of the EPUB 3 development, the validation tools indicated the correctness of the code and accessibility guidelines compliance.

5. EVALUATION

5.1 Methodology

As seen in paragraph 4.5, the EPUB prototype passed the accessibility automatic validation tests in terms of code and guidelines compliance. Next, we evaluated if the EPUB 3 prototype was effectively accessible via screen reader as well as via reading apps. In fact, accessibility compliance does not ensure that it is fully usable via screen reader [20]. Interactive features were particularly considered in this assessment. In this section, we present the results obtained from the evaluation conducted with some eBook reading applications as well as screen reading assistive technology on desktop and mobile platforms. Thus, some eBook reading apps were selected for reading and navigating through the EPUB 3 contents. For each app, the test was firstly conducted with no assistive technology running, in order to verify that the eBook worked appropriately especially with regard to the interactive features. Afterwards, the same app was again used with screen reader interaction to test accessibility support for the interactive actions with the contents.

An inspection evaluation was conducted to verify some accessibility and interaction aspects while using the EPUB prototype. The tests were conducted by three sighted experts and three blind skilled users with experience in reading eBooks and using screen reading software. They were selected by the authors as experts in eBook interaction as well as in desktop and mobile screen reader usage. The blind people were users of IOS Voice Over and Jaws for Windows. In this work we tested only IOS and Windows environments because (1) the involved blind users were IOS and Windows users, and (2) we observed several issues when opening the EPUB 3 prototype with ADE on Android platform. In fact, the ADE version we tested on three devices (two

[4] EPUBCheck https://github.com/w3c/EPUBcheck

[5] ACE https://github.com/daisy/ace

smartphones and 1 tablet) did not support the interactive EPUB. A standard EPUB was appropriately handled by the app, but an interactive version was not. The ADE presents several issues in opening the eBook. Consequently, in this study we only considered applications for Windows and IOS operating systems. Two PCs equipped with Windows 10 and Windows 7 were used for desktop tests. An iPhone X was used as mobile IOS device.

5.2 Procedure

This first pilot inspection test was aimed at verifying that the designed EPUB 3 was properly handled by (1) the applications and (2) the screen reading software. To evaluate the interaction properties of our prototype with or without screen reader, we considered whether the following aspects were achieved:

1. eBook contents could be read properly;
2. interactive components (e.g. links, buttons and checkboxes) could be clearly identified in terms of a correct semantic of the interface elements;
3. interactive components could be properly activated via keyboard and gestures, and the related action could be perceived;
4. multimedia content could be perceived;
5. table of contents could be accessed.

In verifying these aspects, the tester was asked to carry out a number of activities related to four areas: (1) accessing the table of contents; (2) selecting checkboxes and buttons; (3) listening to the additional multimedia descriptions and (4) reading the alert messages in a recipe based on the preferences expressed by selecting one or more allergen in the "food intolerance" section. To this end, in addition to a free reading of the EPUB, some specific tasks like "Go to recipe x", "Select the food intolerance for gluten", and "Listen to the audio description for dish x" were assigned to the testers. They were asked to note down any problems which occurred and any points to consider.

5.3 Application tests

Different reading applications were selected for our purposes: Adobe Digital Edition (ADE) for Windows and IOS platforms, Voice Dream and iBooks for IOS-based devices. These apps were chosen because the majority of visually-impaired people use IOS-Devices via VoiceOver and the blind people involved in this evaluation were IOS users.

5.3.1 Adobe Digital Edition App

Most major publishers use Adobe Digital Editions (ADE) to proof-read their books [1]. On the web site you can read: "the support for EPUB 3 standard allows you a richer reading experience, including: rendering of audio and video content; interactive quizzes, and huge improvements in support for assistive technologies".

While reading the designed EPUB 3 prototype via ADE, the following issues were observed:

- ADE for PC (release 4.5) – without screen reader support – It did not work at all on various PCs with a Windows 10 operating system whereas, it worked correctly with ADE on Windows 7. When running, ADE worked properly with all content and interactive features. The interactive elements could be selected only via mouse click, since activation via the keyboard did not work. Some keys performed actions differently from those expected (e.g. spacebar).
- ADE for PC (release 4.5) – with screen reader support - Several accessibility issues arose on some PCs with Windows 10 and Windows 7:
  (1) the eBook cover made the application crash. To solve this issue, some tests were performed with images in different formats: jpg, gif, bmp and svg;
  (2) interactive elements were only detected as a textual content, i.e. the screen reader did not detect the typology of the elements (e.g. checkbox or button), only the label was read as plain text;
  (3) the interactive elements could not be activated via keyboard (e.g. via spacebar or Enter key). The keys performed different actions from those expected (e.g. spacebar moved to next page rather than selecting the element);
  (4) some sentences or paragraphs at the end of the page were eliminated by the screen reader.
- ADE on IOS devices – without screen reader support – worked properly with the EPUB 3 prototype. All designed interactive features could be correctly used on the touch-screen device.
- ADE on IOS devices – with screen reader VoiceOver support – It worked only in part:
  (1) the contents as well as the interactive elements were correctly detected by VoiceOver. Buttons and links could be appropriately pressed, but the checkboxes did not work: when selecting them via a double tap, the selection status did not change; as a result, none of the actions associated to checkboxes were performed;
  (2) it was not very easy to read the contents paragraph by paragraph, or sentence by sentence. The screen reader considered many contents, paragraphs and sentences, as a whole block. Consequently, the gestures did not work very well as a way to read individual sentences or some blocks;
  (3) the ADE app had many accessibility problems when accessed via Voiceover: e.g. the ADE interface buttons were not detected, and opening eBooks needed to be performed using tasks made available by the operating system ("Copy to ADE" action) rather than those offered by the ADE itself;
  (4) the table of contents was not accessible via VoiceOver, especially because the "Content" button was not detectable via VoiceOver and gestures on the touch-screen.

5.3.2 Books App

Apple iBooks is an e-book reading and store application pre-loaded onto IOS and macOS operating systems and devices. IBooks was renamed to Apple Books alongside the release of iOS 12 and macOS Mojave in September 2018. Books is able to "read the contents of any page (to the user)" using VoiceOver.

We tested our EPUB prototype on both versions, iBooks and Books, with and without screen reader support. However, the issues observed during the tests are the same for both iBooks and Books. The following behaviors were observed while interacting with the EPUB prototype:

- Books for IOS devices – without screen reader support - was able to read all the content, including the interactive components (audio, textual descriptions, buttons, links, and so on). The interaction properties included in the EPUB 3 prototype were fully supported by the application.

- Books for IOS devices – with screen reader VoiceOver support - was able to interpret all the main textual content, but was not able to detect and carry out the interactive components, i.e.:
  (1) the checkboxes and buttons were detected as simple plain text. All the links were detected as simple text too. Consequently the user was not informed about the semantic of the elements, i.e. about the opportunity to activate those components to get additional multimedia content or for navigating quickly through the content;
  (2) the table of contents navigation worked appropriately by relying on the specific button "Content" made available by the application to render the table of contents;
  (3) however, by exploring the content on the touch-screen via finger [2], the label of the checkboxes or buttons could be read by VoiceOver. They could be selected via a double tap. By doing this, the actions associated to the checkboxes and buttons were activated even if the screen reader was not able to announce the semantic and states of the interface components. However, in order to proceed in this way, the user needed to figure out the semantic of the interactive components by readying the plain text and guessing which might be the potential labels for interactive components.

5.3.3 Voice Dream App

Voice Dream is a reading app designed for people who have dyslexia, visual impairment or other learning styles[6]. It reads articles, documents and books out loud. It turns text into speech via voice synthesizers available in numerous languages. With advanced text-to-speech and a highly configurable visual layout, it can be tailored to suit every reading style and level. The app supports may document formats, including DRM-free EPUB eBooks.

We tested the Voice Dream app for IOS devices, loaded on both iPhone and iPad with an IOS 12 operating system. Our EPUB 3 prototype was loaded appropriately into the app library. The textual content was read by the selected voice synthesizer, but the interactive components and elements were detected as only plain text. The interactive components could not be activated neither via screen reader nor without the screen reader support.

5.4 Accessibility Issues and Proposed solutions

As emerged from the conducted assessment, the accessibility of the eBook was not optimal, even if the document passed automatic validations. The checks made through the applications and the screen readers were aimed at verifying especially the implementation of the interactive aspects of the eBook interface.

Most accessibility issues were due to the screen reader interpretation and can be summarized as:

- checkboxes and buttons labels are read as plain text. No information about the typology and selection status is provided to the user.
- links were not detected by the screen reader. Accordingly, the related content could not be activated.

Based on the fact that an EPUB document is a set of html files, we tested the pages via a browser in order to verify the content accessibility via screen reader. The interaction proved to be fully accessible, including the operability with the interactive components and multimedia content. This allowed us to verify that the pages of the book were designed in an accessible way, thanks to the applied guidelines and WAI-Aria techniques.

In order to overcome the accessibility issues observed in the EPUB prototype, we tested some possible technical solutions applied to the code. We especially focused on the interactive checkboxes and buttons. The following solutions were applied by using WAI-Aria:

I. *Role "Checkbox"*. The checkbox was included in a block with a role "checkbox". Nothing happened, the screen reader continued to not detect the typology.
II. *Label addition*. A clearer label associated with the checkbox was next added. The "labelledby" attribute was used to the purpose. The screen reader still did not correctly interpret the checkbox.
III. *Hidden label*. In order to provide some information about the typology of the element, a possible strategy tested was a non-visible label regarding the semantic of the element, something like "This is a checkbox". The hidden label was handled via CSS properties. Such an approach is not a "adequate solution" to suggest; we just tested it to improve the user interface comprehension via screen reader. A standard approach should work with assistive technology.

The final technical solution we tested is coded via the HTML tags and ARIA attributes as reported in Figure 8. Using this approach, the reading on both ADE and Books improved, as the screen readers was able to announce "This is a checkbox". On ADE for PC the complication still persisted, because although the user could perceive the presence of a checkbox, the selection via keyboard still existed.

```
<div role="checkbox" aria-labelledby="Glutine">
    <span class="hidden">This is a checkbox</span>
    <input type="checkbox" aria-checked="false"
           onclick="mostra()" id="Glutine"
           name="intolleranze" value="Glutine" tabindex="0"/>
  <label for="Glutine">Glutine</label>
</div>
```

Figure 8 - Example of HTML code of a proposed solution

6. DISCUSSION AND SUGGESTIONS

The evaluation tests revealed that several accessibility issues still exist when interacting via screen reader with an EPUB digital book. On the one hand, the main core of contents is properly detected and read by all the reading applications when interacting via screen reader. For instance, tables or headings are detected by the AT. We tested these elements also both via existing EPUBs containing these components, and via an eBook we developed in a previous work [4]. On the other hand, some difficulties were encountered in both ADE and iBooks via gestures to navigate paragraph by paragraph, or sentence by sentence.

The majority of problems are related to the detection of the interactive elements and in operating successfully with them. Indeed, by interacting with an interactive element via double tap on a mobile device, the feature is performed even if the screen reader was not able to detect the semantic of the element itself. This means that it would be possible for an experienced user to attempt to interact with some potential buttons or checkboxes, provided they were able to figure out what their labels were while reading the textual contents. For example, if the user read something like "Get additional information", they could suppose

[6] Voice Dream http://www.voicedream.com/

it is a button or checkbox, i.e. an interactive element. Nonetheless, it is clear that interactive features cannot be considered as fully accessible via screen reader on a touch-screen. Furthermore, accessibility by Adobe Digital Edition is currently very limited on a PC. The main core and table of contents can be accessed, but the interactive tasks cannot be used by a blind user. Table 2 gives an overview of the interactive features supported by the apps: + indicates that the feature is supported; * indicates that the feature is not supported or is partially supported.

Most popular applications are able to correctly operate with the EPUB interaction, including multimedia and interactive contents. Adobe Digital Edition showed some interaction problems via keyboard and in working well with some versions of the Windows operating system. Most issues are related to the interaction via screen reader on both desktop and mobile platform.

In short, we can conclude that accessibility in the EPUB 3 interaction is not yet fully supported by the screen reader on desktop and mobile devices. Reading applications, instead, are able to handle multimedia and interactive features in the EPUB 3. So, with regard to the two research questions expressed for this study, we can answer the first question affirmatively, and negatively to the second. In fact, the reading applications handle the customization functions and interactive activities appropriately: the font and color can be changed directly in the eBook and UI control elements and dynamic contents work adequately with the popular reading applications tested. On the other hand screen reader assistive technology still presents several accessibility issues while interacting with the UI components (e.g. links, buttons, checkboxes) and live contents added dynamically to the EPUB User Interface. WAI-Aria technique is not yet fully supported by the assistive technology in the reading applications. In short, we can answer the second question that the screen reader is able to work partially with the multimedia and interactive tasks. The accessibility level achieved by the screen reader depends on the compatibility between the screen reader and the reading application used to interact with the EPUB 3. For instance, links work adequately with VoiceOver in the mobile ADE (M-ADE), but not with Books for the mobile platform (M-Books).

Table 2 - Summary of the tested features: + indicates a positive response, * indicates that the feature is not supported; SR stands for "screen reader" and M- for "mobile"

App	TOC	Audio	Links	Buttons	Checkboxes	Dynamic content
ADE	+	+	+	+	+	+
ADE-SR	+	*	*	*	*	*
M-ADE	+	+	+	+	+	+
M-ADE-SR	*	+	+	+	*	*
M-Books	+	+	+	+	+	+
M-Books-SR	+	*	*	*	*	*
Voice Dream	+	*	*	*	*	*

Some suggestions may be provided for assistive technology and reading applications developers:

a) *App and assistive technology operability*. Applications should fully work with assistive technology in operating with eBook interaction. The applications should provide appropriate semantics on the components of the eBook as a common User Interface. In spite of numerous guidelines and principles in the field, issues related to the operability between assistive technology and applications still exist.

b) *Accessibility compliance*. Application User Interfaces should be fully and effectively compliant with accessibility guidelines. Buttons and any other UI controls need to have clear and accessible labels made available to the assistive technology. EBook contents including multimedia and interactive elements must be designed in an accessible way. This is still a first step to ensure that assistive technology is able to correctly interpret the interfaces and contents, and consequently support people with disabilities.

c) *Advanced assistive technology gestures and commands*. Assistive technology like screen readers should work well with the main functions made available in the applications as well as with eBook contents, via gestures or keyboard depending on the platform. This means that various levels of granularity in gestures and commands should be available when interacting with the content, such as paragraph by paragraph, word by word, and so on. This would allow the reader to navigate and perceive very well all the content.

d) *Semantics in the objects*. Assistive technology should be able to fully interpret and handle the UI information provided by the API (Application Program Interfaces). This should be achieved provided that app developers include appropriate semantic information to the objects, like those provided by WAI-Aria in web applications.

7. CONCLUSIONS

In this work we presented an interactive EPUB 3 digital book designed to evaluate accessibility support by reading applications and screen reading AT. Since the standard EPUB 3 is based on HTML5, CSS and JavaScripts, WAI-WCAG Guidelines and WAI-Aria techniques have been considered to obtain accessible pages. Furthermore, JavaScripts, User Interface interaction controls and multimedia contents have been used to design a more engaging and interactive eBook. As a result, some interactive features have been proposed and designed to enrich static contents. In particular, personalization functions have been designed to customize the reading experience through user preferences. Visual rendering settings, personalized messages and multimedia descriptions are examples of the interactive features designed for our purposes.

In this study we intended to evaluate if (1) the reading applications as well as (2) the screen reader assistive technology are mature enough to fully support interactive and multimedia digital books. Although the developed EPUB 3 proved to be accessible via the validation tools, the interaction was revealed to be not completely accessible via screen reader. Most accessibility issues are specifically related to screen reading technology when working with reading applications. Firstly, screen readers still do not interpret well some interaction components (e.g. buttons and checkboxes) as well as WAI-Aria properties and states. Secondly, the applications still do not operate properly with assistive technologies. Usually HTML+Java Script code renderer engines are nowadays able to well interpret the elements via AT in a browser. When these are used by eBook readers however, the communication between the AT and these engines is not yet fully supported. Reading applications should exploit standard HTML rendering engines, considering that Screen reader AT are able to interpret appropriately HTML elements for the web pages. As a result, the steps necessary to allow the AT to work properly with

an interactive EPUB 3 should be minimal for developers in order to integrate them in the module for EPUB reading.

In conclusion, reading applications are sufficiently mature to support interactive EPUB 3; whereas, screen reader assistive technology is not yet suitable to interact with multimedia and interactive EPUB 3 contents. Although the study cannot be generalized, the results herein observed may give a further contribution in the eBook accessibility field. The study needs to be extended, in order to evaluate other elements and other EPUB 3 eBooks. Future work includes user testing with sighted people to evaluate the engagement offered by the EPUB 3 prototype and further tests on other platforms and applications.

8. REFERENCES

[1] Adobe. Adobe Digital Edition (ADE), https://www.adobe.com/it/solutions/ebook/digital-editions.html

[2] Apple support. Vision Accessibility, https://www.apple.com/accessibility/iphone/vision/

[3] Bartalesi, V., & Leporini, B. (2013). Investigating an accessible and usable EPUB book via VoiceOver: a case study. In Human Factors in Computing and Informatics (pp. 272-283). Springer, Berlin, Heidelberg.

[4] Bartalesi, V., & Leporini, B. (2015, August). An Enriched EPUB eBook for Screen Reader Users. In International Conference on Universal Access in Human-Computer Interaction (pp. 375-386). Springer, Cham.

[5] Biňas, M., Štancel, P., Novak, M., & Michalko, M. (2012, November). Interactive eBook as a supporting tool for education process. In Emerging eLearning Technologies & Applications (ICETA), 2012 IEEE 10th International Conference on (pp. 39-44). IEEE.

[6] Brady, E., Zhong, Y., & Bigham, J. P. (2015, May). Creating accessible PDFs for conference proceedings. In Proceedings of the 12th Web for All Conference (p. 34). ACM.

[7] Brown, A., & Harper, S. (2013, May). Dynamic injection of WAI-ARIA into web content. In Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility (p. 14). ACM.

[8] Calabrò, A., Contini, E., & Leporini, B. (2009, November). Book4All: A tool to make an e-book more accessible to students with vision/visual-impairments. In Symposium of the Austrian HCI and Usability Engineering Group (pp. 236-248). Springer, Berlin, Heidelberg.

[9] Darvishy, A., Hutter, H. P., Horvath, A., & Dorigo, M. (2010, July). A flexible software architecture concept for the creation of accessible PDF documents. In International Conference on Computers for Handicapped Persons (pp. 47-52). Springer, Berlin, Heidelberg.

[10] Fenwick Jr, J. B., Kurtz, B. L., Meznar, P., Phillips, R., & Weidner, A. (2013, March). Developing a highly interactive ebook for CS instruction. In Proceeding of the 44th ACM technical symposium on Computer science education (pp. 135-140). ACM.

[11] Gavrilis, D., Angelis, S., & Tsoulos, I. (2013, March). Building interactive books using EPUB and HTML5. In International Conference on Ambient Media and Systems (pp. 31-40). Springer, Cham.

[12] Henry, S. L. (2012, July). Developing text customisation functionality requirements of PDF reader and other user agents. In International Conference on Computers for Handicapped Persons (pp. 602-609). Springer, Berlin, Heidelberg.

[13] IDPF (2014). EPUB 3 specification, Recommended Specification 26 June 2014, available at http://www.idpf.org/EPUB/301/spec/EPUB-overview.html

[14] Kasdorf, B. (2018). Why accessibility is hard and how to make it easier: Lessons from publishers. Learned Publishing, 31(1), 11-18.

[15] Kim, H. Y., Park, J. H., & Lim, S. B. (2017). Automatic and Semi-automatic 2-tier Check System for EPUB Accessibility.

[16] Kulkarni, A., & Bhurchandi, K. (2015, February). Low cost e-book reading device for blind people. In Computing Communication Control and Automation (ICCUBEA), 2015 International Conference on (pp. 516-520). IEEE.

[17] Leporini, B., & Paternò, F. (2008). Applying web usability criteria for vision-impaired users: does it really improve task performance?. Intl. Journal of Human–Computer Interaction, 24(1), 17-47.

[18] Lundh, A. H., & Johnson, G. M. (2015). The use of digital talking books by people with print disabilities: a literature review. Library Hi Tech, 33(1), 54-64.

[19] Miller, B. N., & Ranum, D. L. (2012, July). Beyond PDF and ePub: toward an interactive textbook. In Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education (pp. 150-155). ACM.

[20] Power, C., Freire, A., Petrie, H., & Swallow, D. (2012, May). Guidelines are only half of the story: accessibility problems encountered by blind users on the web. In Proceedings of the SIGCHI conference on human factors in computing systems (pp. 433-442). ACM.

[21] Schwarz, T., Rajgopal, S., & Stiefelhagen, R. (2018, July). Accessible EPUB: Making EPUB 3 Documents Universal Accessible. In International Conference on Computers Helping People with Special Needs (pp. 85-92). Springer, Cham.

[22] Sirkiä, T., & Sorva, J. (2015, July). How Do Students Use Program Visualizations within an Interactive Ebook?. In Proceedings of the eleventh annual International Conference on International Computing Education Research (pp. 179-188). ACM.

[23] Theofanos, M. F., & Redish, J. G. (2003). Bridging the gap: between accessibility and usability. interactions, 10(6), 36-51.

[24] W3C (2017), Accessible Rich Internet Applications (WAI-ARIA) 1.1, W3C Recommendation 14 December 2017. Available at http://www.w3.org/TR/wai-aria/

[25] W3C (2018). Web Content Accessibility Guidelines (WCAG) 2.1, available at http://www.w3.org/TR/WCAG21/