ACIT4920 Group Work

Group members

* Duy Le
* Alexis Grigoris
* Alhousseiny

For the accessibility evaluation of a system or prototype, we rely on pre-established standards and guidelines. For ICT products, we rely on WGAC guidelines. The Guidelines evolve to accommodate the current needs of accessibility, trying to catch up with the evolution of technologies. Therefore, we are going to rely on the latest version, WGAG 2.1, published in 2018. WCAG is based on four main principles: the content must be perceivable, operable, understandable, and robust, and its 78 success criteria (*Introduction to Understanding WCAG 2.0 | Understanding WCAG 2.0*, n.d.).

To provide an organized structure, the guidelines are classified into three conformance levels: A, AA, and AAA. (*Understanding Conformance | Understanding WCAG 2.0*, n.d.).

Level A is the lowest.

The main goal of this project is to evaluate our chosen website (fig 3) to highlight the potential accessibility issues and develop a prototype to correct them to the most significant extent possible.

For this work, we used the google chrome extensions Lighthouse to evaluate the webpage. From the lighthouse, the home page, and the website's contact page.

These two pages offer us the opportunity to review work on most of the critical accessibility issues on a webpage. It will cover forms, images, layout, and other problems we encounter daily as users.

The home page scored 79 out of 100 on accessibility, and the contact page got 72 out of 100 on accessibility. On both pages, it pointed out certain items as critical such as:

* Buttons do not have an accessible name: In the absence of an accessible name, screen readers pronounce buttons as "button", preventing users who depend on screen readers from using them.
* There is no name associated with the links: The navigation experience for screen reader users is enhanced by link text that is recognizable, distinctive, and focusable (including alternate text for images when used as links).
* Colors in the background and foreground do not have sufficient contrast: Many users find it difficult or impossible to read text with low contrast.

Additional accessibility issues detected on the home page:

* Heading elements are not in a sequentially descending order: When headings are arranged correctly and do not skip levels, the semantic structure of the website can be understood and navigated more easily with the aid of assistive technologies.
* Image elements do not have “alt" attributes: Short, descriptive alternate text should be the goal for informative items. If the “alt” attribute is left empty, decorative components can be disregarded.

An additional accessibility issue was detected on the contact page:

* Form elements lack corresponding labels: Labels ensure that assistive technology like screen readers properly announce form controls.

From this first evaluation using an automated tool, the website presents serious issues perceivable and operable to a collaborative group of users. Some image elements are missing alternative text, and most of the buttons are missing important descriptions. Therefore, the website fails to comply with the guideline 1.1 (text alternative) by providing text alternatives to make the content more perceivable. The website does not qualify for WGAG 2.1 level A by failing those success criteria. Returning to the first principle of WGAG 2.1, ensuring perceivable content, better work can be done to offer a better contrast ratio (guideline 1.4.3 contrast minimum).

Furthermore, on this page, the automated tool missed out that the guidelines 2.4.6 (level AA) and 2.10 (level AAA) were not met because we could not correctly navigate through the webpage through the titles of different sections.

Overall, this page does not meet the Level A accessibility conformance of WCAG 2.1, and the same goes for the contact page. The form element used to collect the user’s information does not have associated labels that can be understood by screen readers, violating the principle 1.1.

The most significant advantage of automated evaluation tools is that they point out the part of the code with which it has found an issue. As a developer, it helps you work better to fix those issues. It also shows a section for the features, element, or sections that has passed the audit.

Having a look at that section helps you be sure nothing has gone under the rock. It lists a list of Items that need to be manually checked, and last but not least, it makes a section called non-applicable, and there you can find all the features and items that the evaluation tool could not match.

However, those tools have their limits. For example, the logical structure of the content, the layout, its responsiveness to screen sizes, the navigation (it should be natural and intuitive for all groups of users), and the grouping of the elements to cite those.

Also, automated evaluation tools cannot always understand if a developer used the <span> or <p> to present a title by just increasing the text size, which will lead to challenging navigation for the blind user because the screen reader will not consider it as a title.

Therefore, a valuable review of the accessibility features by an individual with the right expertise is necessary to interpret the result of the evaluating tool and cross-examine them. The expert will also investigate the relevance of the alternative text and text description for elements like images, buttons, and links.

On the chosen webpages (home page and contact page), we run our accessibility investigation by using a screen reader on the page, trying to navigate using the tab, reviewing the relevance of the alternative text, and analyzing the webpage content.

* We could notice a lousy choice of HTML attributes which leads to the impossibility of navigating between different headlines and sections. For instance, only some of the headers were coded using the heading tags, which will cause a hard time for screen readers making it impossible to navigate (guideline 2.4.6 headings and labels and 2.4.10 section heading) by a group of content.
* In addition, some features, like the use of a floater icon overlayed on the webpage, violate the second principle of WCAG2.1(operable). That icon is supposed to be a quick way to call the university. Unfortunately, there is no other way to reach it without using the mouse. It is totally inaccessible to a screen reader. The grouping could be more effective, which also makes navigation harder.

* The use of images containing relevant text without offering an alternative text makes that information impossible to access for all groups of users relying on screen readers.

The absence of captions for the video content on the home page excludes the information in that video to all deaf and hard of hearing users and users with temporary disabilities, such as being in a noisy environment.

It is mandatory to provide an alternative way to access any content on the website for all users as the guideline 1.1(Text alternative) of WCAG 2.1 require us.

* On the footer, the screen readers could not recognize handy icons like phone and email. Useful links like Facebook and Instagram are embedded in icons with no description (guideline 2.4.4 Link purpose (in context) and 2.4.9 Link purpose (link only)).
* On the form of the contact page, in addition to what has been outlined by the automated accessibility evaluation tool, the form fails the meet the following WCAG 2.1 requirements. First, the input errors are not automatically detected. The errors are not outlined and described to the users when they fail to submit the form due to lousy input (guideline 3.3.1 Error identification). Then the guideline 3.3.3 to suggestion should be proposed to the user for detected no errors.

It is essential to mention that our testing method did not fully cover some aspects, such as dyslexia, persona, and cognitive disability.

1. **Prototype**

Based on the requirements, standards, and guidelines we mentioned in the previous section, combined with the accessibility problems we pointed out on the website. We recreate an accessible prototype containing two pages of the FPT University website: its home page and contact page.

For the home page, we select the program of study section to redesign because we believe this section will affect users the most. On the one hand, the primary goal of the users who access this page is to get more information about the school. If the information is unavailable or inaccessible, that website would be meaningless. On the other hand, the school that owns the website would want the students who are related to them to be provided with valuable information about the school. Moreover, this is also an educational website about a school that teaches technology. Failing to achieve such an important task as ensuring the availability and accessibility of data seems not perceivable to convey to students who want to join the university.

Firstly, we chose to redesign the program of study, which is one of the crucial sections of the website. Giving information about the programs which are available at the school. The first problem we identified and fixed is the hover effects on each block of the program of study section. This is unnecessary. Instead of showing the programs with their specializations directly, the website developers hide those specializations and only display them when the users hover over the study program. Tested on mobile devices, this action would cause many problems in terms of accessibility, users need to hold the selected element for a short amount of time for the content to be displayed, but this also triggers a built-in action in mobile when the users touch and hold an element on the screen, a pop up with multiple actions will be displayed which blocks the users from seeing the content.

Additionally, navigating a website's content with the keyboard does not work for hovering animation. Users can navigate the content, but it will not be shown on the screen. The hovering animation would make sense and understandable if the university has too many programs and cannot show them all on the screen. However, only two or three items are on the list. Animation like this causes unintended accessibility issues or misleading information. Hence, we display all of the programs in those sectors without requiring the users to hover over them and provide the information more clearly.

A picture containing text, screenshot, orange

Description automatically generated

*Figure 5 - Prototype of Home Page*

Also, in this section, we have multiple images. Even though there are pictures that demonstrate students’ lives at the school would be, there is not much information contained, mainly visually appealing, so we marked it as a decorative image by having an empty value in the “alt” attribute.

There are two parts in the contact page, which are the school’s contact information, and a form users need to send to get support from the school. We help users who are using keyboards to navigate or assistive technologies like screen readers to be able to reach the university’s information section. Icons from the original websites are created by using icon fonts, which are failed to convey the information to the users. Instead of that, we use icons from images with alternative text provided to ensure accessibility.

Graphical user interface, application

Description automatically generated

*Figure 6 - Prototype of Contact Page*

We identified several issues with the contact form and decided to recreate it, adding more accessibility capabilities to it. As this is a place where people reach when they need more information, they are needed. Those persons are interested in the education system and want to reach the university to get in touch. What good will it do to the university if potential students do not want to join the school anymore because they are not able to contact the school?

When implementing the form control for the contact page, we try to use more native HTML controls as many as possible over the custom one. Because the standard form controls have more built-in accessibility supports and are more robust across browsers and devices. In case we really need to use a custom control, we reuse HTML elements that can perform some parts of the functions that the designed form controls are intended to ensure robustness. Furthermore, we also use some WAI-ARIA attributes to improve accessibility.

All the fields in the contact form are associated with matching form labels and use the correct input type, such as “tel” for the phone number is “email” type for the email field. Having the correct input type gives more accessibility and convenience for the users. For example, when users fill out the form on mobile devices, the phone number field will open the phone number keyboard, and the virtual keypad will have the @ when entering the email field. It is the responsibility of website creators to give visitors the tools they need to accomplish the interactions website creators ask them to. When the form field is given focus, sighted users can use check the text adjacent to the form fields to decide what information they need to enter, whereas people with visually impaired use screen readers to convert the text to audio or braille displayed. Making sure our forms include proper labels and instructions to ensure that our redesigned website is more accessible ("Why form labels and instructions are important for Digital Accessibility," 2022).

There are two usages of the “placeholder”, it can be used as a suggestion or an example. In our case, it would be better to use a placeholder as an example. It is more concise and gives more details, as the website's audience can see what the input should look like, helping them understand the input we are expecting. Regardless of the inaccessibility of the placeholder, like low contrast, causing a problem for people with low vision. We can increase the color contrast, but the users might mistake it for the value. The reason why we still use placeholder attributes for our fields is mainly that it provides helpful information for the form control, playing a role as an information supporter for the form field.

We also have the “autocomplete” attribute in the form field. Users can rapidly fill in fields with data from earlier submissions. For regular website users, this is an excellent time-saving feature, especially for persons with motor limitations who will find this helpful.

The original website has two versions. One is for people who live in Vietnam or know the Vietnamese language, and the other version is for international students. We create a new field which is country selection. If the users enable the location permission, it will automatically select the country with the phone country code number, or else the users can choose by themself. Eliminating the burden for users to type the + sign and remember the country code when they provide their phone number.

For many years, many techniques have been used to separate human users of websites from robots. Although new methods have emerged, the conventional CAPTCHA method of asking users to spot masked text in an image is still widely used. Users are required to complete a job that is thought to be relatively simple for humans but challenging for robots in all interactive ways. Unfortunately, many persons with impairments are intrinsically excluded by the interactive task's very nature, which results in a denial of service to these users. Additionally, according to research, many standard CAPTCHA techniques are no longer incredibly safe or effective, which makes it more challenging to provide services that are secure against robotic infiltration and accessible to people with disabilities. Every time an interactive CAPTCHA technique is used, it will raise many challenges to the accessibility of the web form for people with various sensory and cognitive impairments. Humans have a wide range of intellectual abilities and limitations ("Inaccessibility of CAPTCHA," 2021). After researching and comparing multiple CAPTCHA techniques. To provide equality for everyone, a non-interactive approach like [reCAPTCHA v3](https://developers.google.com/recaptcha/docs/v3) is the best choice here. Which is running in the background, and we, as developers, will handle the logic to verify whether the current users are real people.

I use a red \* after the label to indicate that field is required. This is one of the most often used techniques for alerting users that a form control is necessary. Moreover, we use the attribute “aria-hidden” on that \* element to prevent the screen reader from saying the word “star” or “asterisk”. Additionally, we use the “aria-required” attribute for the required form. When encountering the form, it will make the screen reader announce the word “require”. So visually impaired people can know it. The website will not let the users submit the form if they do not fill out the required fields. An indicator will appear and the field that has not been filled will gain focus. Besides the required fields, other fields like Subject and Content are optional, and they do not need a specific sign to indicate. The users also get a notification.

We also redesigned the footer to make it more accessible for people who are navigating with a keyboard and assistive technologies like a screen reader.

Text

Description automatically generated

*Figure 7 - Prototype Footer*

Today, mobile phones are essential tools in our daily life. They have already surpassed desktop counterparts due to their ability to access services and content online. On the street, it is simple to observe that practically everyone is holding a smartphone. One-on-one communication devices are now fully functional computers thanks to these powerful smartphones, which have long outlived their original purpose. Ten years ago, people would use their desktop computers to check their e-mail, play games, and edit documents. They now do the same things on a mobile phone as well, probably more frequently (Guerreiro et al., 2019). Noticing the importance of the mobile web, we also make sure our application is accessible on mobile devices as well.