Bachelor Project

Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Computer Science

Identification of Human illnesses and disabilities and adaptation of user interface

Maria Potapova

Supervisor: Ing. Jiří Šebek
January 2021

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Potapova** Jméno: **Maria** Osobní číslo: **466326**

Fakulta/ústav: **Fakulta elektrotechnická**

Zadávající katedra/ústav: **Katedra počítačů**

Studijní program: **Softwarové inženýrství a technologie**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Zjištění lidských vad a nemocí a následná adaptace uživatelského rozhraní

Název bakalářské práce anglicky:

Identification of Human illnesses and disabilities and adaptation of user interface

Pokyny pro vypracování:

Implementace uživatelského rozhraní pro širokou veřejnost nemusí být vždy optimální. Pro uživatele s vadami nebo nemoci může být použití tohoto UI problematické. Cílem této práce je prozkoumat lidské vady a nemoci, které ovlivňují interakci s uživatelským rozhraním a navrhnout adaptivní metody nebo techniky, které interakci zlepší.

Cíle:
1) Vytvořit rešerši na nemoci nebo vady lidí, které ovlivňují interakci s uživatelským rozhraním.
2) Vytvořit rešerši, jak tyto nemoci nebo vady identifikovat, čím se projevují.
3) Navrhnout adaptivní metody / techniky pro zlepšení uživatelského rozhraní pro uživatele s těmito vadami.
4) Otestovat navržené techniky na demo aplikaci a na reálných uživatelích.

Seznam doporučené literatury:

1) Šebek, J. - Richta, K.: Usage of Aspect-Oriented programming in Adaptive Application Structure. New Trends in Databases and Information Systems: ADBIS 2016 Short Papers and Workshops, BigDap, DCSA, DC, Prague, Czech Republic, August 28-31, 2016, Proceedings
2) Debbie Stone, Caroline Jarrett, Mark Woodroffe, Shailey Minocha: User Interface Design and Evaluation

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Jiří Šebek, kabinet výuky informatiky FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **14.02.2020** Termín odevzdání bakalářské práce: **05.01.2021**

Platnost zadání bakalářské práce: **30.09.2021**

Ing. Jiří Šebek podpis vedoucí(ho) práce	podpis vedoucí(ho) ústavu/katedry	prof. Mgr. Petr Páta, Ph.D. podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Studentka bere na vědomí, že je povinna vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

.

Datum převzetí zadání

Podpis studentky

Acknowledgements

I would like to thank my supervisor Ing. Jiří Šebek for his patience during my work on this thesis.

Declaration

I declare that the presented work was developed independently and that I have listed all sources of information used within it in accordance with the methodical instructions for observing the ethical principles in the preparation of university theses.

In Prague, January 5

Abstract

This bachelor thesis focuses on the disability and impairment identification and adaptation of the user interface. To simplify the interaction between individuals with disabilities and UI, the adaptation techniques and methods were defined. Based on the knowledge obtained from the research, the library that can identify the impairments and adapt the UI was developed. The demo application developed with the help of the library was tested on real users. . . .

Keywords:

Supervisor: Ing. Jiří Šebek

Abstrakt

Tato bakalářská práce se zaměřuje na identifikaci a následně adaptaci uživatelského rozhraní pro lidé s vady a postižení. Pro zjednodušení interakce mezi uživatele s vady a postižení a uživatelským rozhraním byly navrhnuty adaptivní techniky a metody. Na základě rešerše byla implementována knihovna, která dokáže identifikovat druh postižení a adaptovat uživatelské rozhraní. Pomocí knihovny byla demo aplikace a otestována na reálných uživatelích. . . .

Klíčová slova:

Překlad názvu: Zjištění lidských vad a nemocí a následná adaptace uživatelského rozhraní

Contents

Figures

Tables

Chapter 1

Introduction

1.1 Motivation

Nowadays, no one can imagine their daily life without mobile devices, personal computers, or other modern technologies. Checking out the news or the weather in a selected location, chatting with both family and friends, transferring money, monitoring daily activities, and doing other things – there are many reasons to use technologies that sometimes do not exceed the average size of the palm. According to Global System for Mobile Communications (GSMA) real-time intelligence data, there are exactly 5.13 billion people in the world who own mobile devices, and this number represents over 60% of the world's population. [1] Despite the tremendous opportunities the modern world has to offer, some people are not able to use them to their full potential.

Individuals with impairments have a restriction in the ability that makes it difficult to perform certain activities or interact with people and the outside world to some extent. The accessibility issue becomes more pertinent due to the increasing amount of systems that require the user's interaction. Any system, whether it is a website, mobile application, or an operating system, should be accessible – it is required by law. [2]

Different types of impairments may require different approaches in user interface design. An option for satisfying the needs of every type of impairment is to design an adaptive user interface. A system that would be able to change the UI depending on the input data collected from the environment, device's sensors, the user's actions, and the user themself can make a significant contribution regarding the interaction between the user with impairments and the system.

1.2 Objective

There are several objectives of this bachelor thesis. From the research standpoint, the objective is to review the areas of disabilities and learn how they affect people's lives and the interaction with the UI. The research also requires to become more familiar with the AUI and learn which components of UI should be adapted depending on the areas of disabilities.

Based on the knowledge obtained from the research, the techniques of impairment identification and subsequent adaptation of the UI should be developed and applied to the demonstrative application. Based on the work undertaken on the developed techniques and tested demonstrated application, certain conclusions will be drawn.

Chapter 2
Research

2.1 User Interface

User Interface (UI) is a bridge that connects one with any system. Debbie Stone, Caroline Jarrett, Mark Woodroffe, Shailey Minocha define the good UI as an “interface that is easy to use and easy to understand, that meets the needs of the intended users, and that supports users in the tasks they wish to undertake”. [3] Even though the user interface has an easy-to-understand objective, it is an essential part of the system that will strongly influence the user experience. The UI has two components: *i*nput and *o*utput.

The input component defines how the user communicates with the system. Let us consider a personal computer. In this case, the most common input components are a mouse and a keyboard. It also can be a voice, or gestures tracked by sensors – nowadays, the input devices list is diverse. The output component defines how the system brings information to the user. This is why the UI is a subset of human-computer interaction. [4] *H*uman-computer interaction (HCI) studies how the system and its users should work together to satisfy the user’s needs efficiently.

While designing a system, many aspects must be taken into consideration, such as the user’s expectations and requirements – this is what user-centered design (UCD) is about. The UCD prioritizes the user’s needs and tries to achieve a quality user experience. Also, the UCD has to make sure that the product they are creating meets the users’ needs. [5]

2.1.1 Adaptive User Interface

Adaptive user interface (AUI) is the UI that can change appearance and or interaction behavior to satisfy the requirements of the individual. The system with AUI collects data from various sources to learn the individual’s interests and then creates a user model. Based on the data collected for the user model from the environment, the system starts the adaptation process to perform an adaptation effect (Figure 2.1).

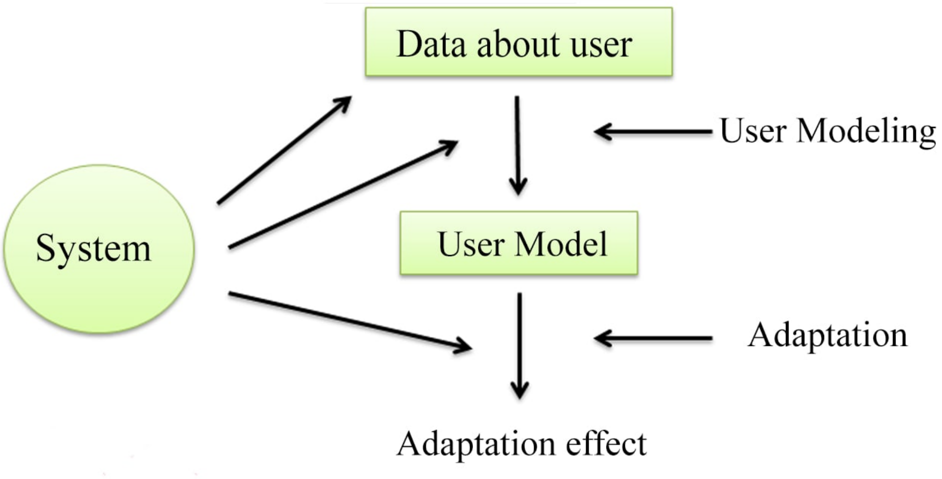


Figure 2.1: Structure of the system with AUI (taken from [11])

According to Peter Brusilovsky and Mark T. Maybury, there are three types of adaptation effects: [11]

- *E*diting the properties of the UI elements. For example, the UI can change the light mode to dark mode depending on the time of day. In this case, the layout would change from white color to dark color.
- *P*ersonalizing the content. Knowing what the user prefers the most, the system can adapt to satisfy their needs.
- *N*avigation support. The system can adapt the navigation process by streamlining the links.

2.2 Disabilities

Laura Kalbag, the author of “Accessibility for Everyone,” claims that there are five areas of disability experience problems that affect how people interact with the UI. [6] This section gives information about the following areas of disability.

2.2.1 Visual Impairments

Based on the data provided by the World Health Organization, at least 2.2 billion people are vision-impaired or blind. [12] Eye diseases such as myopia, hyperopia, and astigmatism, which significantly affect vision, can affect people of any age.

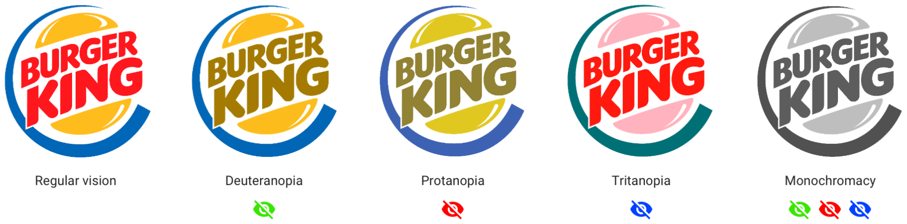


Figure 2.2: Comparison of the perception of the Burger King logo by individuals with different types of color blindness. [7] Simulated with Coblis – Color Blindness Simulator [9]

Globally, approximately 8% of males and 0.5% of females suffer from color blindness. [13] To clarify, if an individual is color blind, it does not always signify that they are not able to see the colors at all. A human eye identifies colors with the help of millions of receptors called cones that respond to the light waves with different lengths reflected from objects. There are three types of cones:

- S-cones (blue cones) pick up the short-wavelength lights;
- M-cones (green cones) pick up the middle-wavelength lights;
- L-cones (red cones) pick up the long-wavelength lights.

These three types of cones allow us to recognize a wide range of shades and colors. If the proper function of the cones is disrupted, it leads to color blindness.

Individuals with *d*euteranopia color blindness cannot see green colors and shades due to a lack of green cones. Individuals with *p*rotanopia color blindness are missing red cones, and that hampers their ability to see red and orange colors and shades.

Individuals with *t*ritanopia color blindness are missing blue cones. The rarest type of color blindness is *m*onochromacy; it affects only 0.00003% of the population. [13] Individuals with monochromacy color blindness are lacking all three cones and only see values without color. Figures 2.2 and 2.3 demonstrate the perception of individuals with different types of color blindness described above.

Eyesight loss and partial eyesight loss (Low Vision) is another type of visual impairment that affects the interaction with the UI. WHO defines several categories of low vision based on specific levels of visual acuity and field of vision. The field of vision is the area an individual can see when their eyes are fixed in one position.

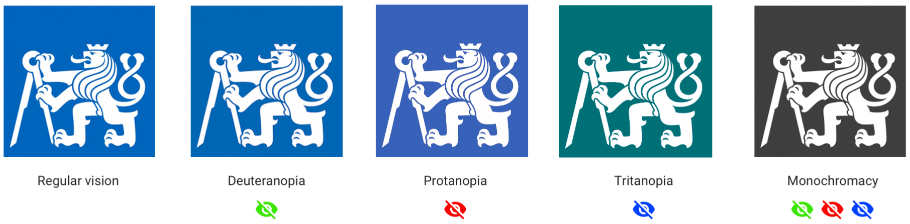


Figure 2.3: Comparison of the perception of the Czech Technical University in Prague logo by individuals with different types of color blindness. [8] Simulated with Coblis – Color Blindness Simulator [9]

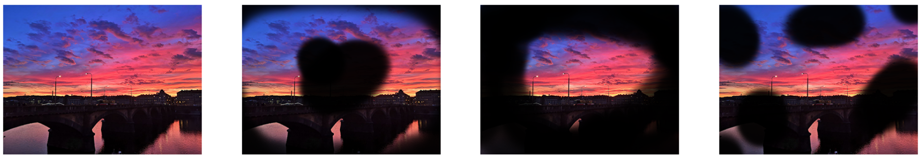


Figure 2.4: Simulated comparison of the perception by individuals with different types of partial eyesight loss: age-related macular degeneration, glaucoma, and diabetic retinopathy (based on [6])

Age-related macular degeneration is the leading cause of blindness in adults. With this type of eyesight loss, the center of the field of vision seems blurry. The damage to the optic nerve defines Glaucoma, and has the opposite effect of macular degeneration mentioned earlier: instead of the center, the edges are obscured. Individuals with diabetes might have diabetic retinopathy. It causes dark spots in the field of vision, distorting what the eyes see. [6] Figure 2.4 depicts the simulated comparison of the perception by individuals with different types of partial eyesight loss.

2.2.2 Auditory Impairments

Several factors may affect auditory impairment. Some people have congenital hearing loss – they are deaf since birth. Others might lose hearing due to accidents or illness. An essential cause of hearing loss is age-related since human hearing reduces with age. Figure 2.5 illustrates a direct correlation between hearing loss and age. Unfortunately, the chart provided by the National Institute of Deafness and Other Communication Disorders is significantly outdated and does not provide relevant data.

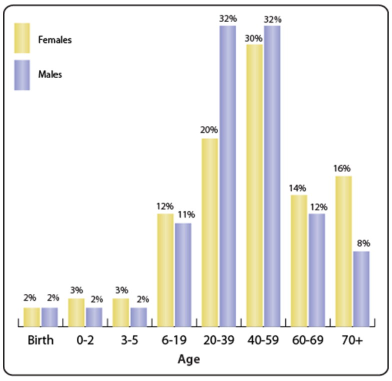


Figure 2.5: Chart illustrating the age at which hearing loss begins. Created by the NIDCD. [10]

The Hearing Loss Association in America distinguishes three types of hearing loss. [14] Conductive (mechanical) hearing loss happens when sounds can not get through the outer and middle ear to the inner ear. It is usually caused by a blockage or abnormality in the ear.

Sensorineural hearing loss (SNHL) is caused by damage to the nerves in the ear or problems with hair cells inside the hearing organ called the cochlea. Those with this type of hearing loss typically struggle to understand speech. Mixed hearing loss is a combination of both conductive and sensorineural hearing loss.

2.2.3 Motor Impairments

Motor impairment is a loss or a reduction in movement or muscle control, usually in one or more limbs. Neural-tube defects may cause weakness and even paralysis. Cerebral palsy affects muscle control and movement. Brain or spine injuries affect neurological, motor, sensory, and cognitive problems. Individuals with motor impairments may struggle to hold things or even use them. One of the common examples of motor impairment is tremor – a rhythmically repeating movement in one or more parts of the body. [15]

2.2.4 Cognitive Impairments

American Psychiatric Association defines learning disorders as a set of complications that relate to having difficulty learning and developing. [16] Having a learning disability is not associated with an individual's intelligence.
It means that they require alternative teaching methods to approach the result.

- *D*yslexia (impairment in reading) is a term for learning difficulties characterized by problems with accurate or fluent word recognition, poor decoding, and poor spelling abilities. [16] Dyslexia affects both oral and written communication throughout an individual's life. These individuals usually have trouble identifying letters or words that might result in inaccurate and slow reading. Moreover, dyslexia can also cause difficulty with spelling since an individual can accidentally add or forget some letters. Having reading issues means that there may be difficulty in understanding written text.

- *D*ysgraphia (impairment in writing) is a term for difficulties in written expression, such as impaired writing ability and trouble with spelling and grammar. [16] Usually, individuals with dysgraphia have problems with composing basic sentences and grammar and struggle with thinking and writing at the same time.
- *D*yscalculia (impairment in mathematics) is a term for difficulties in math processing. [16] These individuals have a poor understanding of numbers. The most common problem is the lack of number sense, an intuitive understanding of how the numbers work and how to manipulate them. Individuals with dyscalculia also struggle with reading charts and graphs.

Moreover, the adult literacy problem is worth mentioning. As an example, according to the Program for the International Assessment of Adult Competencies (PIAAC) in the United States, approximately 43.1 million adults have low English literacy level or cannot read at all. [17]

2.2.5 Vestibular disorders

Vestibular or balance disorders are caused by damage to the vestibular system. The vestibular system is a small set of organs placed in the inner ear, which works with the visual system to process the information from the outside world and send it to the brain. The brain sends the received information to the muscles so that the individual can move and stay balanced. Issues in the vestibular system cause balance disorders.

Being off-balance, feeling that you might fall on the ground, thinking that everything you see is spinning – or, in short, having vertigo, – having dizziness, blurred or bounced vision, hearing ringing in your ears, struggling to concentrate on other things while you try to stay balanced, being sensitive to bright lights – these are some symptoms telling that an individual might have a vestibular disorder. Vestibular disorders can be caused by head trauma, ear infection, or particular medications. [18]

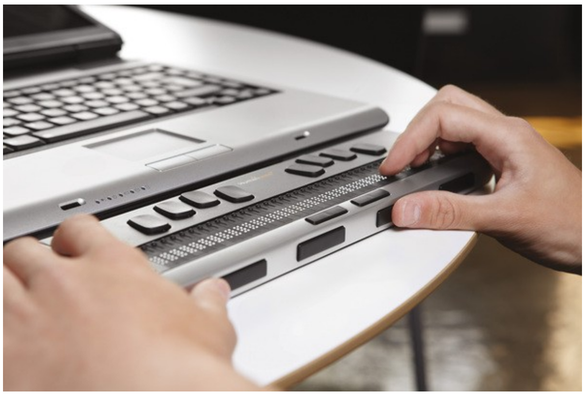


Figure 2.6: Refreshable Braille Display [19]

2.3 Accessibility Technologies

After reviewing the main areas of disabilities, those that may affect people since birth as well as throughout life, a problem worth considering is digital accessibility. It is crucial to provide the use of applications and technologies such as mobile devices and personal computers for everyone without regard to disorders, disabilities, and impairments. Even so, the way of approaching more accessible technology is complicated due to the variety of conditions that must be satisfied.

This subsection focuses on the solutions that were applied and implemented to satisfy the needs of disabled people and facilitate the use of technologies from both hardware and software perspectives. Both assistive technology and adaptive technology are tools used by people with disabilities. There is a slight difference between assistive and adaptive technologies: assistive technology tools are usually customized for the disabled considering their limitations, while adaptive technology tools are specifically designed for people with disabilities. [25]

Assistive Technology tools have a large variety of services and needs that can be satisfied, ranging from ambulation devices such as walkers and wheelchairs that are used by individuals with motor impairment, to learning materials that can help gain knowledge more efficiently for those with learning disabilities. [26] [27]

Figure 2.6 shows a model of refreshable braille display – a hardware tool for blind individuals. The machine, connected to a computer, laptop, or mobile device, converts the text on the device's screen to braille alphabet and vice versa[28]. The disadvantage of these screens is the price. The price for a braille display ranges from 3,500$ to 15,000$. [29]

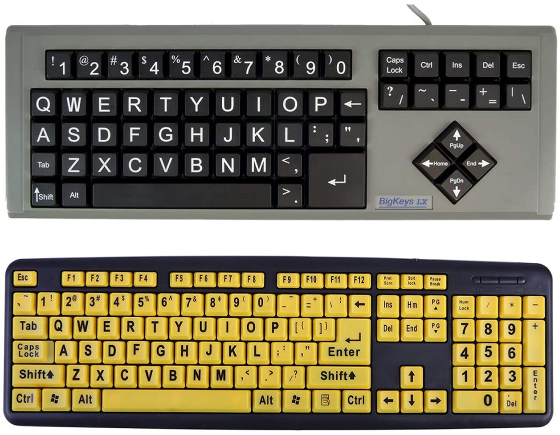


Figure 2.7: Keyboards for people with motor and visual impairments. Borrowed from [20] and [21]

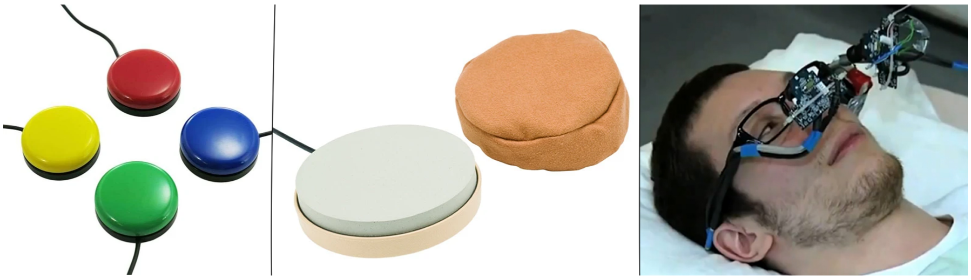


Figure 2.8: Tools for individuals with motor impairments: orby switches, pillow switches, eye tracking device.
Borrowed from [22] [23] [24]

Figure 2.7 shows two keyboards designed for visually-impaired and motion-limited individuals. The upper keyboard is mostly suited for people with motor impairments: the keys are big to prevent inaccuracy in typing because of tremors in the hands. The lower keyboard has a vast contrast ratio between the color of the keys and the color of letters and other signs, facilitating the search for the right key.

Another tool for individuals with motor impairments is an adaptive switch. When triggered, it sends commands to the connected device. Typically, the switches have one or more big movable buttons helping the individual to navigate between the elements of the device. The surface of the switch can be solid, for when it is to be used with hands, as well as soft for using it with a cheek or the head. If an individual cannot use their hands to manipulate the devices, there is also the option of using other types of assistive technology tools such as head and eye trackers [30]. Some of the mentioned tools are shown in 2.8.

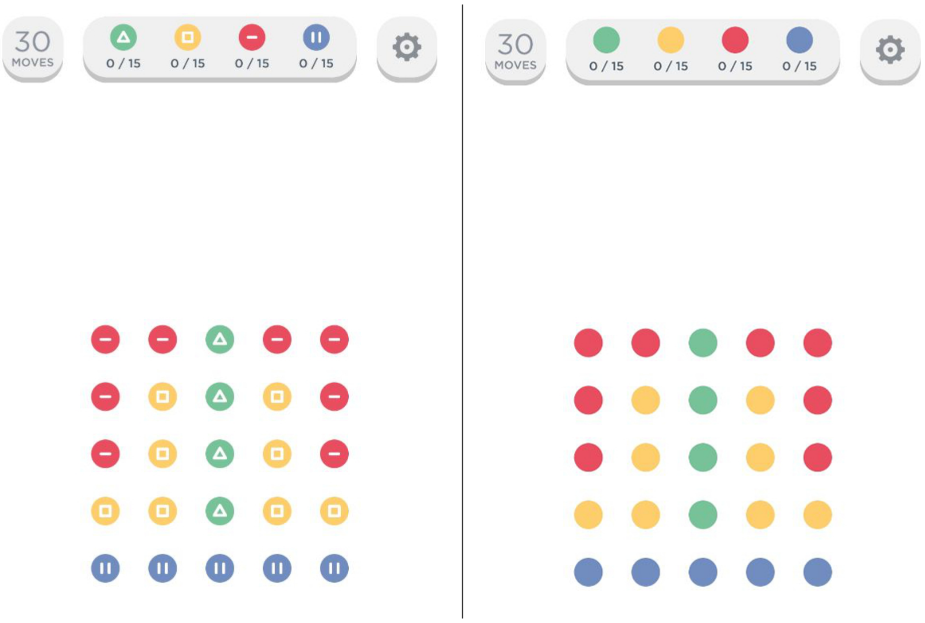


Figure 2.9: "TwoDots" [32] with and without the colorblind mode.

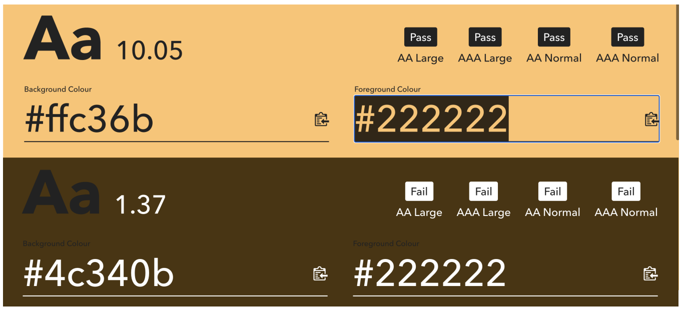
Screen readers are useful for visual or motor-impaired people, as well as people with dyslexia. Some screen readers are available in accessibility settings of the device like Narrator from Microsoft, VoiceOver from Apple, TalkBack from Android. Several screen readers are paid: the Job Access With Speech (JAWS) single-user home license costs 1000$ and can be unaffordably expensive for some. [31]

2.4 Accessible UI Guidance

This section describes how the impairments can affect the interaction with UI and how it can be adapted for the needs of individuals with impairments.

2.4.1 UI for individuals with visual impairments

It was mentioned in section 2.2.1 that some individuals are not able to see certain colors and shades. Thus, colorblind individuals cannot rely on color only. UI designers should pay attention to color choice to avert confusion from the colorblind users' side. Green and red, green and dark orange, green and blue, blue and purple, light green and yellow, blue and gray should not be used together.



Aa 10.05

Pass AA Large | Pass AAA Large | Pass AA Normal | Pass AAA Normal

Background Colour #ffc36b

Foreground Colour #222222

Aa 1.37

Fail AA Large | Fail AAA Large | Fail AA Normal | Fail AAA Normal

Background Colour #4c340b

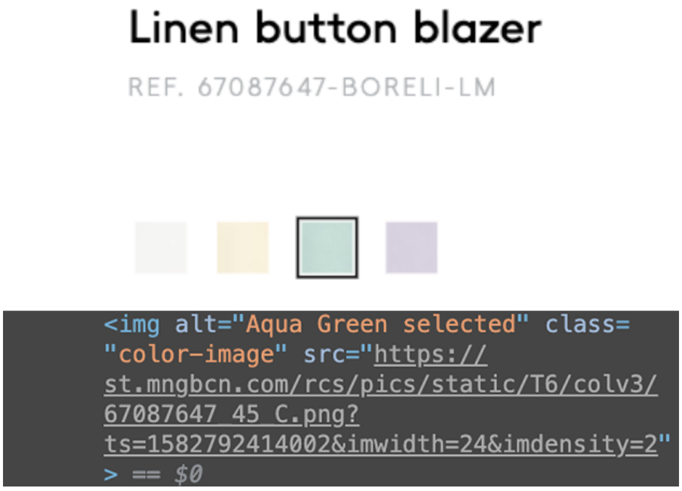
Foreground Colour #222222

Figure 2.10: Contrast ratio generated in Color Contrast Checker [34]

If the colors cannot be changed for certain reasons, a solution still exists. Adding shape textures on the elements helps people who are colorblind navigate between elements, even if their colors look similar. An excellent example of this technique is a mobile game called "TwoDots" where you have to connect the dots with the same color. [32] Figure 2.9 illustrates how the game offers a colorblind mode that adds shapes to the dots.

It is also necessary to think about the contrast ratio between background and foreground for visually-impaired users. The ratio is defined by the precise contrast, ranging from 0 to 21: 0 for the lowest value and 21 for highest value. [33] There are many color contrast checking websites, so the designer can always confirm the suggested color combination for an acceptable contrast ratio (see Figure 2.10).

In addition to the proper selection of colors, all images should have an alternative text. Individuals with visual impairment rely on screen readers to browse websites and use applications, making alternative text essential for people with visual impairment. An alternative text must accurately describe a picture. As an example, e-shops display items with many variations of color, and the color blocks located near the item display are used to represent it. A screen reader would not be able to tell the user which color is selected and shown if the blocks do not provide an alternative description. Figure 2.11 illustrates how the alternative description is used for picking the color for a product.



Linen button blazer

REF. 67087647-BORELI-LM

```
<img alt="Aqua Green selected" class=
"color-image" src="https://
st.mngbcn.com/rcs/pics/static/T6/colv3/
67087647_45_C.png?
ts=1582792414002&imwidth=24&imdensity=2"
> == $0
```

Figure 2.11: An example of usage of an alternative description for visually-impaired users. [40]

2.4.2 UI for individuals with motor impairments

For motor-impaired users, it is necessary to think about the size of each interactive element. Individuals with motor impairments might have unstable movements that could cause inaccuracy in using the system. Bigger buttons and links reduce the likelihood that an individual with a limited range of motion would miss the element they would like to interact with.

Interactive elements in accessible websites or applications' should be streamlined. We already know that some users with motor impairment can use an adaptive switch as an input UI component. Streamlined UI elements will ensure that individuals with motor impairment, who use switches to navigate through the website, would not struggle to navigate.

2.4.3 UI for individuals with cognitive Impairments

The UI that cares about users with cognitive impairments and neurodiversity should focus on several principles. The font choice for these individuals is important. The letters of the font should be recognizable and distinctive, with The best fonts are Helvetica, Verdana, Arial, Comic Sans, Garamond, Consolas. [35] [36]

For this type of impairment, the textual content should be easy-to-understand. The user might be tangled in complex sentence structure if It is not recommended to write long sentences. Using plain language benefits everyone, especially people with a low level of literacy. For better comprehension, important information or buttons should be accompanied by visual information, such as icons and images. [35] [36]

2.4.4 UI for individuals with auditory Impairments

From the moment when the UI became dynamic with the help of animated elements, it also started affecting users with vestibular disorders. Val Head defined types of animations that might cause the symptoms of vestibular disorders: animated elements that move too far on the screen, fast parallax, scrolling, and sliding animations. [38] Once Apple launched iOS7 with a very dynamic UI, it was criticized for causing dizziness, headaches, and other symptoms attributed to vestibular disorders. [39] In that case, it is important to think about the number of animated elements on one page.

Chapter 3

Related Work

This chapter describes the approached methods of detecting different disabilities using modern technologies. These methods can be useful as an inspiration for developing own disease identification techniques, which is one of the main objectives of this thesis.

Tremor detection is quite relevant to the researchers. Iván García-Magariño, Carlos Medrano, Inmaculada Plaza, and Bárbara Oliván designed the mobile application called HTrembApp (Hand Trembling detector App) for detecting hand tremors based on a user's everyday activities. The tremor is detected with the help of the built-in smartphone accelerometer. [41] V. Parra, G. Figueras, M. Huerta, IEEE Member, A. Marzinotto, R. Gonzalez, R. Alvizu designed a mobile application for Parkinson tremor detection that also uses the data collected from an accelerometer. The collected data from this application can be generated into a report file. The user can save a report in the smartphone's storage or send it via email. Unlike HTrembApp, this application does not support the analysis during everyday activities. The smartphone with the installed application is secured with a flexible strap to ensure the accuracy of the data. [42]

Some works are related to neurological disorder identification. Scientists from Bournemouth University in Poole, UK also developed an application called SmartAbility, that can detect "physical abilities" (such as cerebral palsy, autism, and Noonan syndrome) using built-in smartphone sensors, to recommend suitable assistive technologies. [43] Scientists from NIH (National Institutes of Health) studied neurological disability identification via smartphone. They worked on an application that performs a neurological examination by analyzing the data collected from screen touching. [44] Mikian Musser wrote an article about autism spectrum disorder detection in child's age using computer vision. [45]

Deaf and hard of hearing individuals depend on vibration signals to be alerted. Hamed Ketabdar and Tim Polzehl presented an application that detects the changes of the ambient acoustic environment and notifies a hearing-impaired user with visual and tactical alerts. [46]

Judith Harkins, Paula E. Tucker, Norman Williams, Jeff Sauro studied special patterned vibrations for emergency alerting via mobile devices. [47] They asked for the opinion of mobile users who depend on vibration for their signaling needs, where the users had to rate the effectiveness of suggested vibration patterns and the vibration length.

Eye disease detection is popular among scientists as well. Sadaf Malik, Nadia Kanwal, Mamoona Naveed Asghar, Mohammad Ali A. Sadiq, Irfan Karamat, and Martin Fleury presented a framework for the standardized recording of patient symptoms and clinical observations. [48] With the help of machine learning, scientists from Khulna University studied early blindness detection based on retinal images to prevent eyesight loss among patients with diabetic retinopathy. [49]

Vestibular disorders are complicated to identify. Vijay Renga mentioned the usage of the phone camera for analyzing vestibular dysfunction only as a tool for recording the eye movements for further analysis. [50]

For this thesis, the approaches of the disease and impairment identification described in this chapter are used as an inspiration for identification methods and techniques. Since the identification of vestibular disorders is currently complicated and does not have enough methods to inspire with, this thesis will focus on the visual, auditory, hearing, and cognitive impairments.

Chapter 4

Analysis and design

One of the main objectives of this thesis is to create a demo application that demonstrates the techniques which can improve the interaction between the user with impairments and the UI. To accomplish this objective, the library that can determine the user's probable impairments and adapt the UI will be developed. It is more advantageous to create an independent library. In this case, it can be used for other applications. The techniques defined in this chapter are inspired by the works described in chapter 3. Also, the information about diseases, accessibility technologies, and accessible UI guidance obtained from research help to define the technologies properly.

The main purpose of this library can be divided into two parts: impairment identification and UI adaptation. Figure 4.1 illustrates the use case of the library, whose parts will be described in this chapter in more detail. Figure 4.2 illustrates the package diagram of the library.

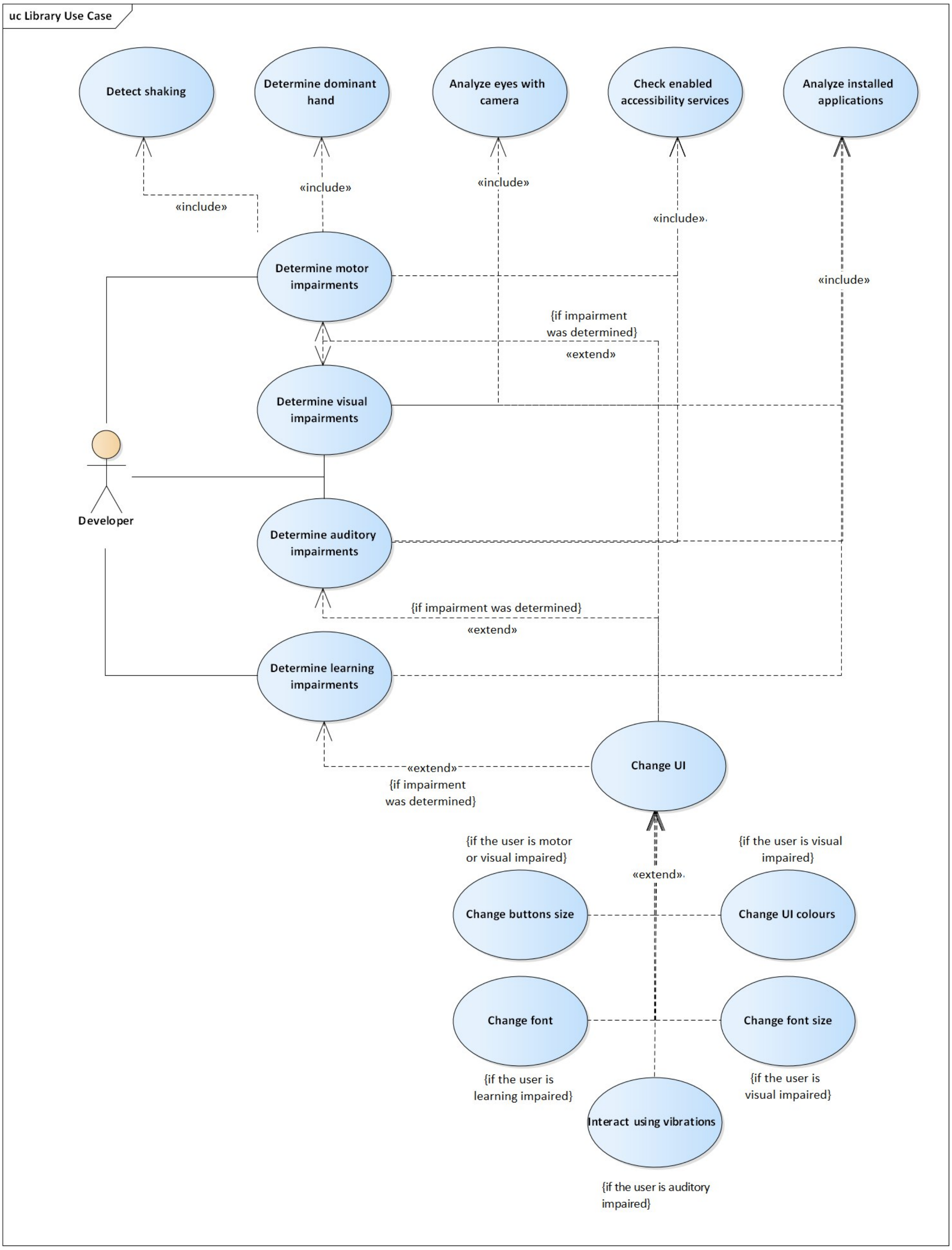

uc Library Use Case
Detect shaking
Determine dominant hand
Analyze eyes with camera
Check enabled accessibility services
Analyze installed applications
«include»
«include»
«include»
«include»
«include»
Determine motor impairments
{if impairment was determined}
«extend»
Determine visual impairments
Developer
Determine auditory impairments
{if impairment was determined}
«extend»
Determine learning impairments
«extend»
{if impairment was determined}
Change UI
{if the user is motor or visual impaired}
{if the user is visual impaired}
«extend»
Change buttons size
Change UI colours
Change font
Change font size
{if the user is learning impaired}
{if the user is visual impaired}
Interact using vibrations
{if the user is auditory impaired}

Figure 4.1: Library's use case diagram

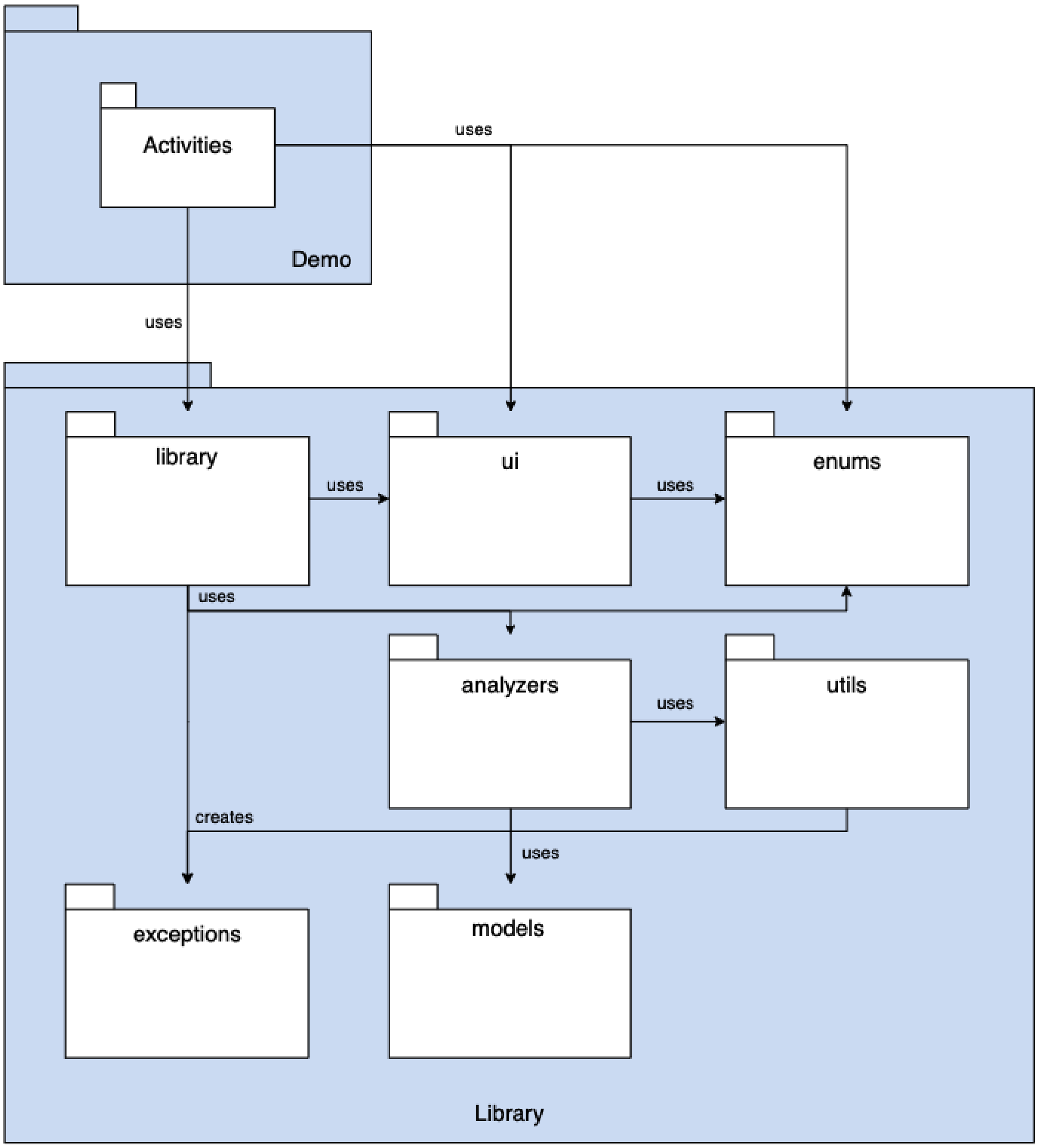


Figure 4.2: Library's package diagram

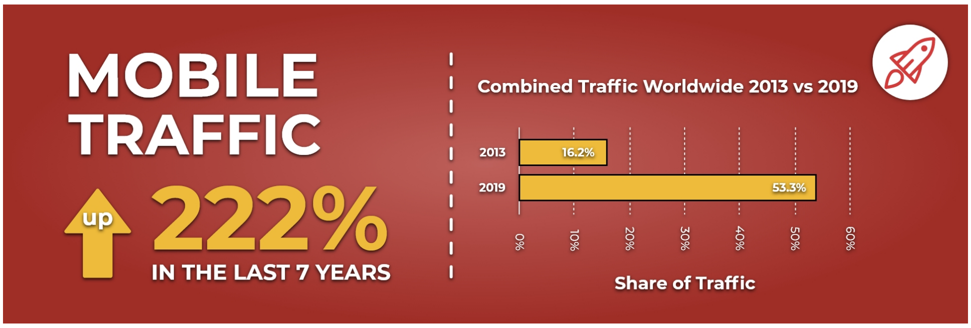


Figure 4.3: Mobile traffic usage comparison over the period 2013-2019 [51]

4.1 Platform choice

Before starting defining the techniques, the platform for the library should be chosen. As it was stated in research, UI must be adapted for individuals with disabilities on each platform. However, this thesis is aimed at developing techniques that can be used primarily for mobile application development. This chapter explains why exactly the mobile platform was chosen.

The reason why mobile devices are the main target of this thesis is that usage of mobile traffic increases year by year, only getting higher. Figure 4.3 illustrates the difference in traffic usage in 2013 compared to 2019. For the first time, the usage rose over 50 percent in 2018. [51]

Two main operating systems are usually run for mobile devices: either Android powered by Google or iOS engineered by Apple. According to data from “GlobalStsats”, collected in November 2020, Android together with iOS share 99.37% of the worldwide mobile operating system market. [52]

Even if iOS is one of the main operating systems, Android devices rank first in the worldwide market (it conquered 71% of the global market in November 2020). Second, unlike iOS, Android is an open-source OS, which allows using more third-party tools for development.

Mobile applications are divided into three different categories: native, hybrid, and web. Native mobile applications are designed for a specific platform. The majority of applications installed on devices are native, written in the official language for developing on selected OS. For Android, these languages are Kotlin and Java, while iOS applications are written in Swift or Objective-C. A huge benefit of designing a native application is that the developers can freely use available capabilities and features, such as push notifications, motion sensors, cameras, swipe gestures, and much more. However, in this way, the main disadvantage of native applications comes from its definition: a native application can be developed for one platform only. Thus, the application for Android, which is native, can not be run on devices running on iOS, and vice versa.

A mobile application that is compatible with several operating systems and platforms (or cross-platform for short) is called hybrid. Usually, hybrid applications are written with the help of frameworks like Xamarin or React Native. Developing a hybrid application might be lucrative when the project should be done fast, or a small budget is given. Still, unlike the native applications, the hybrid one lacks functionality. Thus, hybrid applications cannot support native-only functions and features of mobile devices.

Websites accessed by a mobile browser are called web applications. Even if these websites are called web applications, that is not exactly true. Because of its ability to adapt well, it looks like a mobile application. However, web applications are deprived of multiple mobile features. On top of that, this type of mobile application does not support offline mode.

Being an open-source platform, Android was chosen as a platform for the thesis demo application. Since the application will run on Android devices, the library becomes platform dependent.

4.2 Identification

Once the platform has been chosen, the techniques applied to the library can be defined. Taking into account studies from scientists regarding disease identification (chapter 3), built-in sensors, Android OS assistive technologies and services, the following identification techniques are suggested in this section.

4.2.1 Enabled accessibility services

Each Android-compatible mobile device provides accessibility settings, where the user can adjust the settings in audio, visual, and interaction control categories to suit their needs. Google has two assistive technologies: TalkBack and Switch Access, both are pre-installed on devices. TalkBack is a screen reader that comes on Android. This technology has a gesture-based user interface, it is controlled by swiping and various gestures. Switch Access is a technology that helps individuals with limited dexterity or limited range of motion navigate through the device and interact with the elements of the UI with the help of volume keys, switches, or special keyboards. Also, Android has a captioning service for deaf or hard of hearing users.

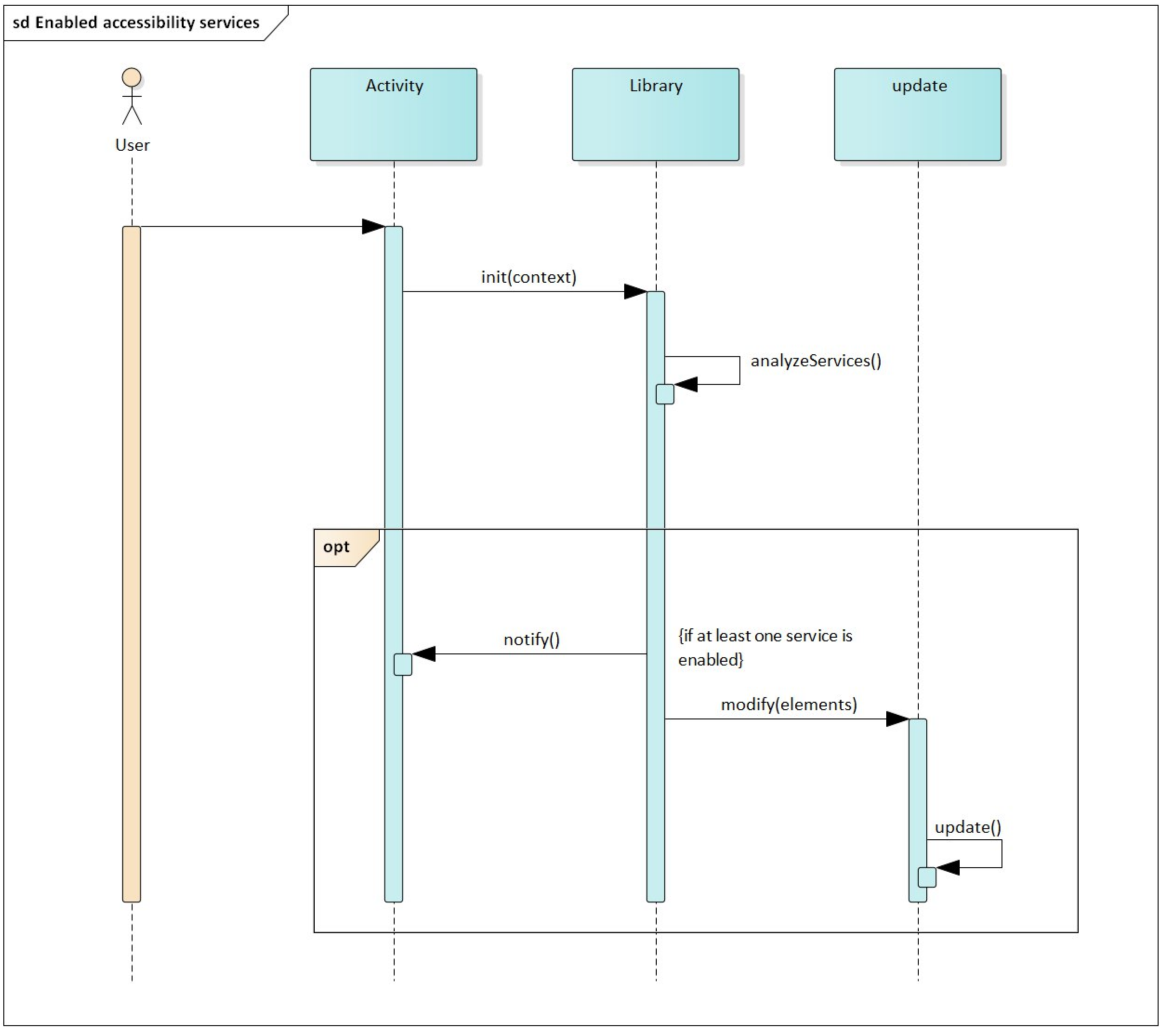


Figure 4.4: Sequence diagram of enabled accessibility service analysis

Unfortunately, it is impossible to learn if every option listed in the accessibility menu is turned on – Android does not share the data about all options or this information cannot be accessed. As an example, Android's source code has settings for color blindness, but it is secured and requires privileged access [53]. However, the developer can identify if captioning service, TalkBack, or Switch Access are turned on. By getting this data the developer can adapt the UI for users with these assistive technologies and services enabled. Figure 4.4 illustrates a sequence diagram of this technique.

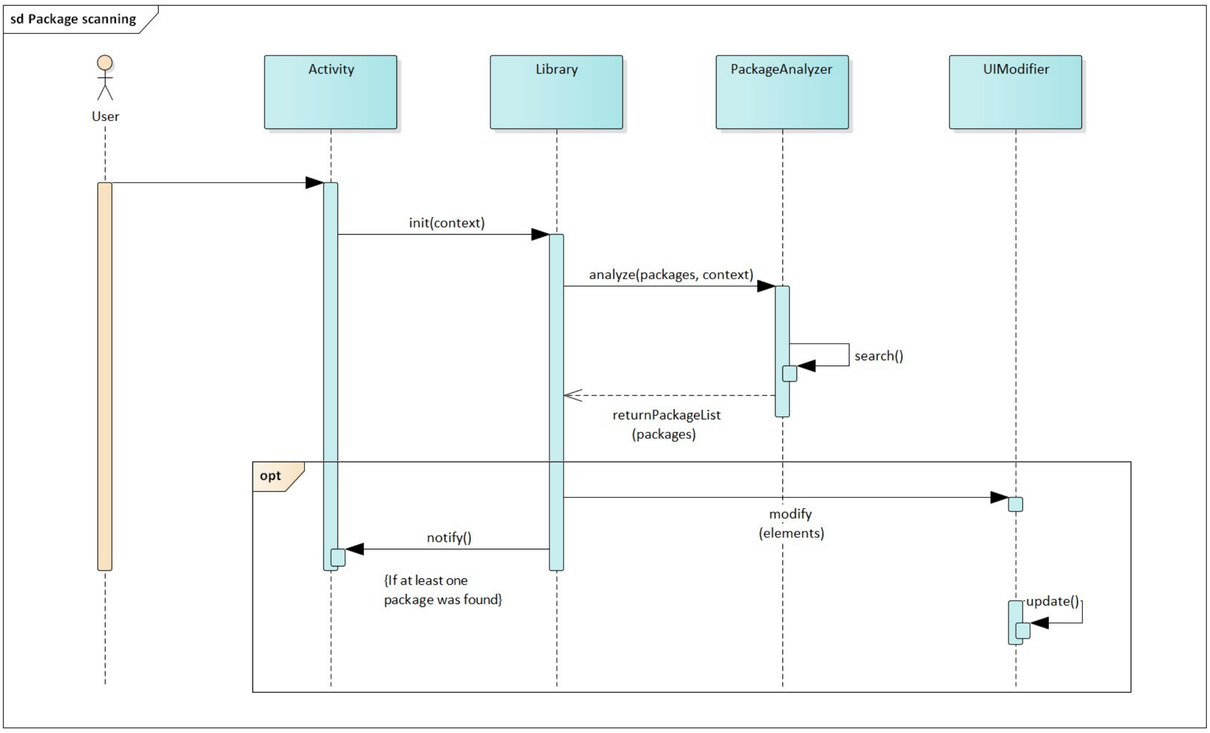


Figure 4.5: Sequence diagram of installed package scanning technique

4.2.2 Package scanning

Assistive technologies or applications are not confined to Switch Access and TalkBack, or captioning. Thus, the idea of package scanning lays in an attempt to guess the user's impairment type (in case if it exists) by packages installed on the user's device to adapt the UI based on this impairment. The library iterates through all installed packages from the user's device, trying to discover at least one package developed for the disabled. Despite the lack of accuracy prediction, this technique provides one of the fastest ways to guess the user's type of disability and can be applied for all types of disabilities. Figure 4.5 illustrates the sequence diagram of this technique.

4.2.3 Tremor detection

With the help of a built-in accelerometer available on every mobile device, it is possible to predict if the user has a hand tremor. As we know from 3, tremor detection is being studied actively. Nevertheless, this technique requires deeper research of the hand movements performed by the user that suffers from tremor. Because of that, the identification logic for this library has been simplified and the library reacts to continuous shaking instead. Figure 4.6 illustrates the sequence diagram of the tremor detection technique.

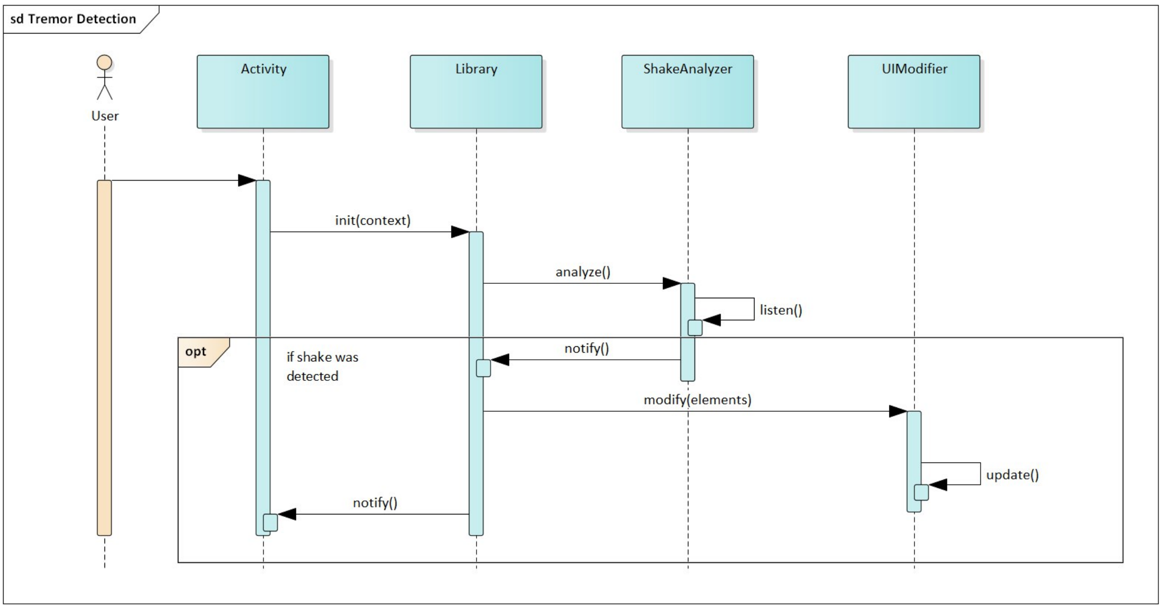


Figure 4.6: Sequence diagram of hand tremor detection technique

4.2.4 Dominant hand detection

A dominant hand is a hand that has been used most frequently for everyday activities. For several users, it is not comfortable to reach the other side of the device's screen to touch the interactive element, especially if the screen resolution is enormously large. The idea of the dominant hand identification technique lies in UI adaptation in favor of the dominant hand: if the user's left hand is a dominant one, the buttons will be located on the left side of the screen, and vice versa. The idea of identification by touches is inspired by the research provided by scientists from Bournemouth University in Poole, and described in chapter 3. [43] This technique might help people who have only one arm, and also can benefit people without disabilities. Figure 4.7 illustrates the sequence diagram of this technique.

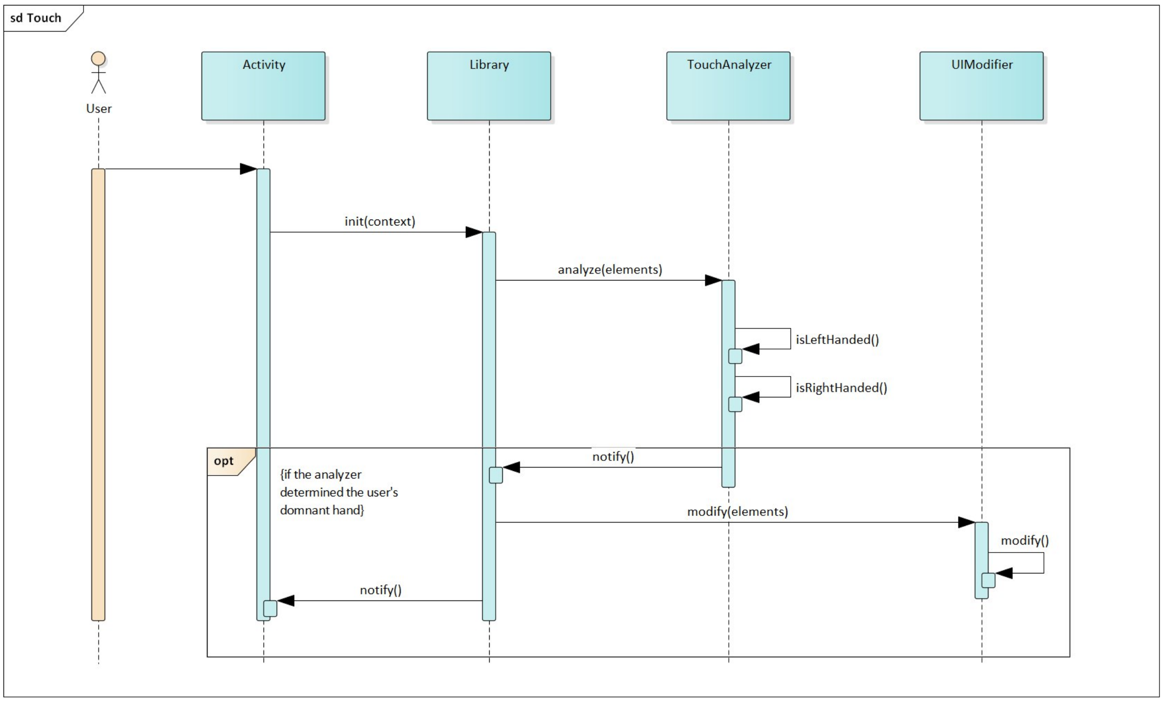


Figure 4.7: Sequence diagram of hand tremor detection technique

4.2.5 Eye infection detection

Since the human's eyes can not function properly if they are infected, eye infections can affect the interaction with the UI as well. Several research papers mentioned in chapter 3 use the computer vision as the approach to detect an impairment or a disease. With the use of a camera, the library would be able to detect if the eye area has redness and inflammation.

To apply the eye infection analysis technique, the library has to launch a camera and find a frame containing a face with face detection technologies. After the library detects a face, it has to check if the eyes are open. Because for this technique the eyes are the point of interest, only the frame with opened eyes must be analyzed. Figure 4.8 depicts a sequence diagram for this technique.

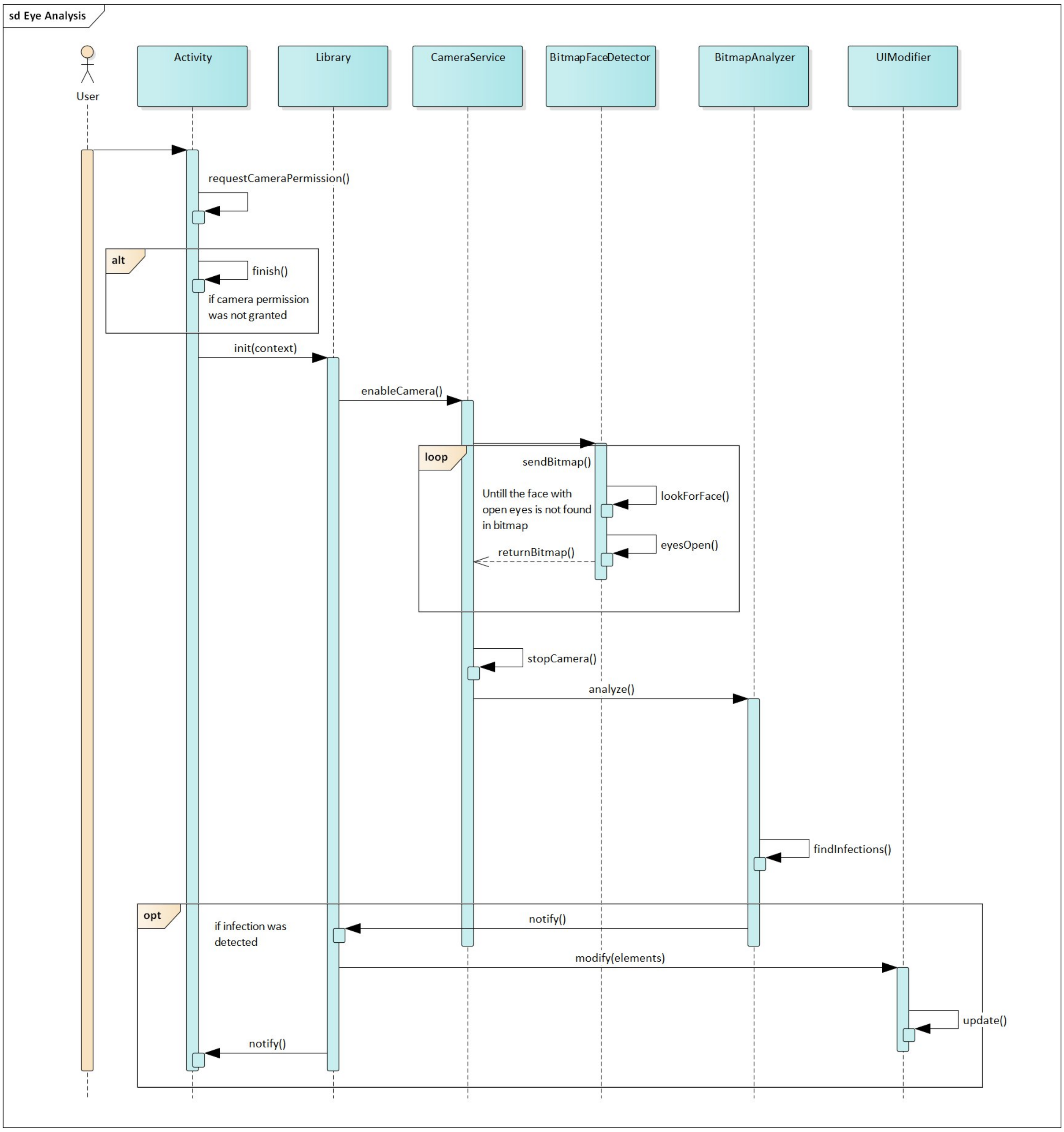


Figure 4.8: Sequence diagram for eye infection detection

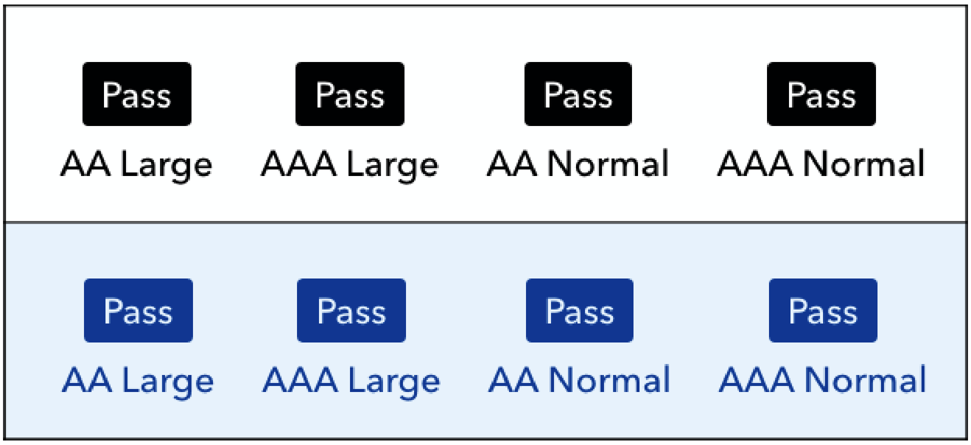


Figure 4.9: Contrast ratio results from Colour Contrast Checker [34]

4.3 UI Adaptation

As previously stated, if the system supports the AUI, UI elements can change their appearance as the result of an adaptation effect. This chapter describes how the library adapts the UI if the library identifies an impairment with the techniques analyzed in the previous section.

4.3.1 Color choice

The library suggests picking between two color pallets for UI adaptation: blue shades and gray shades because these shades are friendly for individuals with each type of color blindness (see figure 2.3). Color Contrast Checker confirmed that the combination of these shades passes all levels of contrast ratio. [34] The color change is applied for individuals with visual impairments, because color blindness is considered as a visual impairment. Figure 4.9 demonstrates the color combinations and the passed color contrast tests.

4.3.2 Textual content adaptation

The library changes the textual content when it identifies that the user has either visual impairment or cognitive impairment. For individuals with cognitive impairment, the font changes to Comic Sans as a recommended font. The color and the size of the textual content changes for visually-impaired individuals.

4.3.3 Button adaptation

The library changes the buttons for individuals with visual, cognitive, and motor impairments. For users with visual impairment, the buttons change the colour and the size. For users with cognitive impairment, the button's label font changes to Comic Sans. For users with motor impairment, the buttons change its size. Also, the positions of buttons can change if the user's dominant hand is detected and if the button has enough space on the screen to move left or right.

4.3.4 Alerts

Unlike the individuals with other impairments, the buttons or textual content do not need to be adapted for auditory impaired users. Instead, the interaction between the application and the user with this type of impairment should be supported by visual or vibration alerts. This approach was inspired by the research about the alerts for hard of hearing or deaf people, described in chapter 3. When the library detects, that the user is deaf or hard of hearing, the application vibrates to notify a user. If the sound needs to be described, the vibration will be accompanied by the textual description of the alert or the sound.

4.3.5 Personal adaptation

The developer has an option to receive the information from the library about the identified impairments and adapt the UI themself. This solution helps when the developer wants to implement a complicated logic for the user with a certain type of impairment.

4.4 Technologies

This section defines the technologies and the reasons why exactly these technologies are being used to develop the library.

4.4.1 OpenCV

OpenCV is a free and open-source distributed library written in C++ programming language, widely used for image processing, machine learning, and real-time operations. It offers a broad range of purposes, such as object tracking and detection, text detection, gesture detection, and much more. According to information, provided on the OpenCV official website, the library can be used for Windows, Linux, Mac OS, iOS, and Android operating systems. [54]

4.4.2 MLKit

MLKit is an SDK developed by Google that offers machine learning features. Video and image analysis APIs (or shortly Vision APIs) provide barcode scanning, image labeling, and other useful options. MLKit is clear and easy-to-use, making it a perfect SDK for this project. [55] For this library, OpenCV and MLKit are used for eye infection analysis technique.

4.4.3 CameraX

CameraX is a camera-support library from Android Jackpack – the suite of Android libraries, designed to reduce boilerplate code. [56] It is backward compatible with 90% of devices running on Android. The parts of Camera2 API, CameraX's predecessor, were abstracted away, making CameraX easier to use. A new camera-support library gives three predefined use cases:

- Preview: to display the camera preview on the device's screen;
- Image analysis: to perform an analysis of the frames received via camera;
- Capture: to simply capture a video or a picture.

For this project, CameraX is used to capture the stream of frames acquired from a device's camera for further frame processing with MLKit and OpenCV inside of the image analyzer use case.

4.4.4 accessibilitools

Accessibilititools (usually with a lowercase "a" letter) is a library of utility classes developed by team Novoda. [57] The purpose of this library is to easily check if TalkBack and Switch Access assistive technologies and captioning services are enabled when the application is used. For this project's library, accessibilitools is used for enabled accessibility tools technique, because in such a case, there would be no need to write own util classes for an enabled service check.

4.4.5 Seismic

Seismic is a framework developed by team Square [58]. It is intended for shake detection with a built-in accelerometer. The framework is platform-dependent and can be implemented only for Android projects. For this project's library, seismic is being used for tremor detection technique.

Chapter 5

Implementation

This chapter presents the implementation of the library that can be used for application development. As it was mentioned earlier (Section 4.1), the library is platform-dependent and can run only on Android devices. Android Studio from JetBrains was chosen as a development environment. The library is written in Java 8 since this version supports lambdas, Stream API, and other features that make the implementation easier. It is currently impossible to use newer versions of Java for Android. The library's code is covered with Javadoc commentaries and log messages.

5.1 Requirements

Every Android application project has an Android application manifest file that describes essential information for Android build tools and other Android services. To use the library, the developer must declare in the manifest file to ask for the camera and vibration permissions and ask the user to grant permissions to use the camera and vibrating at the first launch. Otherwise, the library will not work. The library also uses an accelerometer for tremor detection. Listing 5.1 illustrates what the developer must declare in their Android manifest file.

Listing 5.1: Adding permissions and features to AndroidManifest file

```
<uses-permission android:name="android.permission.CAMERA" />
<uses-permission android:name="android.permission.VIBRATE" />
<uses-feature
android:name="android.hardware.sensor.accelerometer"
android:required="true" />
```

5.2 Initialization

Once all permissions were granted, the developer can initialize the library from the activity. Since the CPU of the device might be overloaded because of the frame processing, the library has two options of initialization: with and without camera analysis (see Listings 5.2 and 5.3). To initialize the library, the developer must send an application context and the collection of the elements. If the developer wants to include camera analysis, they need to add an Android lifecycle object and add the texture view to the elements collection. The developer can choose if they want the library to modify the UI elements and pick a color palette between blue and monochrome shades (these color palettes were defined in subsection 4.3.1).

Listing 5.2: Initializing the library with camera frame processing

```
AccessibilityUI accessibilityUI = new AccessibilityUI(
    this, elements, this, true, ColourThemeEnum.BLUE);
```

Listing 5.3: Initializing the library without camera frame processing

```
AccessibilityUI accessibilityUI = new AccessibilityUI(
this, elements, true, ColourThemeEnum.MONOCHROME);
```

The library has a listener as well. When the impairment is detected, the developer can manipulate with received context to adapt the UI. Listing 5.4 shows an example of the library's usage.

Listing 5.4: Example of UI adapting with library's listener

```
    @Override
    public void onHearingImpairmentDetected() {
        linearLayout.setVisibility(View.GONE);
    }
```

5.3 Analyzers

This section describes how the identification techniques were implemented.

5.3.1 Installed packages

The library gets the list of installed applications from PackageManager that retrieves the data about application packages currently installed on the device. Then the list of installed packages is compared with the list of packages designed for individuals with impairments stored in a JSON file (Listing 5.5).

Listing 5.5: Code sample from the PackageAnalyzer class's device

```
List<PackageData> packageDataList = getApplicationData(context);

List<PackageData> installedPackageData = packageDataList.stream()
.filter(
packageData -> applicationInfoList.stream()
.anyMatch(a -> packageData.getPackageName()
.equals(a.packageName)))
.collect(Collectors.toList());

return installedPackageData.stream()
.map(PackageData::getApplicationType)
.distinct().collect(Collectors.toList());
```

5.3.2 Eye infection

It was already mentioned, that the activity must have a texture view element to work with the camera, which is used for eye analysis. Before starting the analysis, the library checks if the texture view was found in the collection of the elements. Once the required element is found, the library starts the camera, otherwise, the library throws an exception.

The library launches the camera in CameraService class supported by CameraX. Before the camera launches, the preview and image analysis use cases must be initialized. Then, CameraX starts receiving data from the camera (Listing 5.6).

Listing 5.6: CameraX initializing

```
CameraX.unbindAll();
Preview preview = setPreview();
ImageAnalysis imageAnalysis = setImageAnalysis();

CameraX.bindToLifecycle(lifecycleOwner, preview, imageAnalysis);
```

The ImageAnalysis receives the frames from the image view as a bitmap object. Each frame is sent to an ImageAnalysis' analyzer function, which is implemented by this library's code. During the analysis process, the library passes a bitmap to BitmapDataHandler class. The usage of BitmapDataHandler is needed to identify a bitmap with a face with eyes open. The identification process is supported by MLKit's FaceDetector.

Once the bitmap with the face and opened eyes was found, the library stops the camera and passes the bitmap to the BitmapAnalyzer class for further analysis. BitmapAnalyzer finds the eye landmarks and creates separated bitmaps with left and right eyes from the original bitmap with the user's face. The color analysis is performed by OpenCV library. It converts a bitmap to a matrix and starts looking for colors in the range stored in the JSON object (see Listing 5.7). BitmapAnalyzer notifies the library if the eyes are red or infected.

Listing 5.7: Analyzing eye colors with OpenCV

```
try {
    Core.inRange(
    hsvMat,
    ScalarDataParser.parseStringDataToScalar(colorHSVData.getHsvLowValue()),
    ScalarDataParser.parseStringDataToScalar(colorHSVData.getHsvHighValue()),
    hsvMat
    );
} catch (WrongScalarParseDataException e) {
    Log.e(TAG, e.getMessage());
}
```

5.3.3 Enabled services

It was mentioned in analysis that Android allows to check only three services that are suitable for visual, auditory and motor impairment detection. Listing 5.8 shows how the library checks how the accessibility service identification is implemented.

Listing 5.8: Method analyzeServices()

```
private void analyzeServices() {
    AccessibilityServices services = AccessibilityServices.newInstance(context);
    if (services.isSpokenFeedbackEnabled()) {
        Log.i(TAG, "Found␣spoken␣feedback␣enabled");
        accessibilityUIListener.onVisualImpairmentDetected();
        dataEnums.add(DataEnum.VISUAL_IMPAIRMENT);
    } else if (services.isClosedCaptioningEnabled()) {
        Log.i(TAG, "Found␣closed␣cationing␣enabled");
        accessibilityUIListener.onHearingImpairmentDetected();
        dataEnums.add(DataEnum.AUDITORY_IMPAIRMENT);
    } else if (services.isSwitchAccessEnabled()) {
        Log.i(TAG, "Found␣switch␣access␣enabled");
        accessibilityUIListener.onMotorImpairmentDetected();
        dataEnums.add(DataEnum.MOTOR_IMPAIRMENT);
    }
    modify();
}
```

5.3.4 Shake detection

The library starts listening to the accelerometer once the library is initialized. The shake detection is handled in ShakeAnalyzer class. Once ShakeAnalyzer detects the shaking movements, the analyzer notifies the library (Listing 5.9).

Listing 5.9: Code sample from the ShakeAnalyzer class

```
@Override
public void hearShake() {
    Log.i(TAG, "Detected␣shaking");
    shakeListener.onShakeDetected();
}
```

5.3.5 Touch analysis

The library starts listening to the button touches in TouchAnalyzer class. Once the button is pressed by the user, the library obtains the touch coordinates. After five touches, the TouchAnalyzer starts comparing the coordinates with the middle point of the button. The TouchAnalyzer notifies the library if the user pressed the same side of the button at least four times out of five. Listing 5.10 illustrates the methods of dominant hand detection by using the coordinates data and the pressed button's middle point.

Listing 5.10: Code sample from the TouchAnalyzer class

```
boolean isLeftHanded(List<Coordinates> coordinates, int midpoint) {
    return coordinates.stream()
    .filter(coord −> coord.getX() < midpoint)
    .count() > 3;
}

boolean isRightHanded(List<Coordinates> coordinates, int midpoint) {
    return coordinates.stream()
    .filter(coord −> coord.getX() > midpoint)
    .count() > 3;
}
```

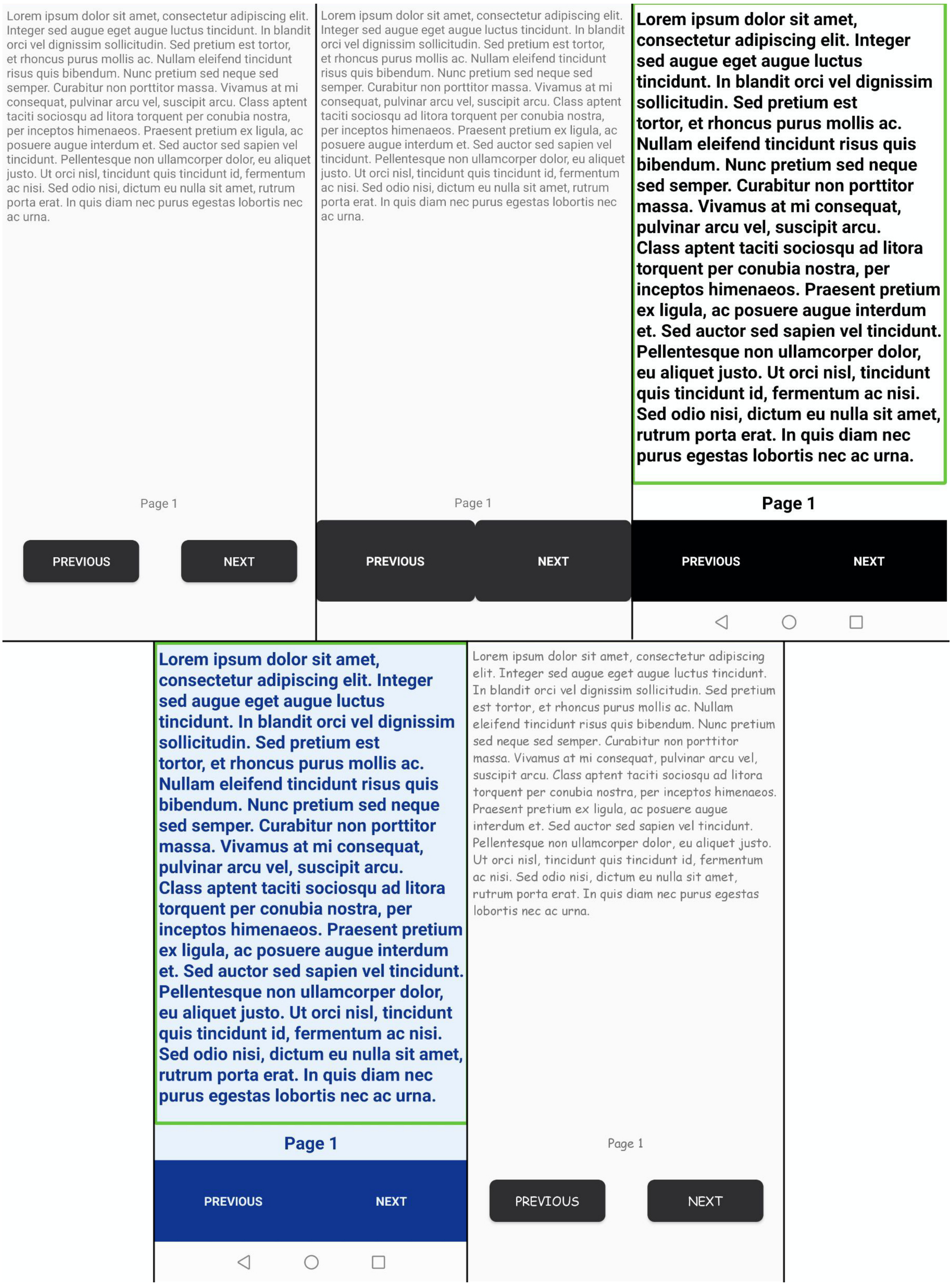
Chapter 6

Library usage examples

This chapter provides examples of the library usage for a demonstrative application and how the UI can adapt based on the identified impairment.

6.1 Modifying textual content

Some applications require reading a lot of textual information from the device's display. Messengers, news applications, electronic reader applications – many users cannot imagine their daily life without these applications. In some applications, the size of the textual content can be modified manually. However, it would be more comfortable if the application was able to understand the need to increase the size, the color, or the font of the text. If the library learns that the user has an impairment, it would change the application's UI regarding the area of impairment. Figure 6.1 illustrates the comparison of how the library can adapt the UI programmatically depending on the detected impairment as follows: without adaptation, for users with motor impairment, for visually-impaired users (black and white colors), for visually-impaired users (blue shades), and for users with cognitive impairment.



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer sed augue eget augue luctus tincidunt. In blandit orci vel dignissim sollicitudin. Sed pretium est tortor, et rhoncus purus mollis ac. Nullam eleifend tincidunt risus quis bibendum. Nunc pretium sed neque sed semper. Curabitur non porttitor massa. Vivamus at mi consequat, pulvinar arcu vel, suscipit arcu. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Praesent pretium ex ligula, ac posuere augue interdum et. Sed auctor sed sapien vel tincidunt. Pellentesque non ullamcorper dolor, eu aliquet justo. Ut orci nisl, tincidunt quis tincidunt id, fermentum ac nisi. Sed odio nisi, dictum eu nulla sit amet, rutrum porta erat. In quis diam nec purus egestas lobortis nec ac urna.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer sed augue eget augue luctus tincidunt. In blandit orci vel dignissim sollicitudin. Sed pretium est tortor, et rhoncus purus mollis ac. Nullam eleifend tincidunt risus quis bibendum. Nunc pretium sed neque sed semper. Curabitur non porttitor massa. Vivamus at mi consequat, pulvinar arcu vel, suscipit arcu. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Praesent pretium ex ligula, ac posuere augue interdum et. Sed auctor sed sapien vel tincidunt. Pellentesque non ullamcorper dolor, eu aliquet justo. Ut orci nisl, tincidunt quis tincidunt id, fermentum ac nisi. Sed odio nisi, dictum eu nulla sit amet, rutrum porta erat. In quis diam nec purus egestas lobortis nec ac urna.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer sed augue eget augue luctus tincidunt. In blandit orci vel dignissim sollicitudin. Sed pretium est tortor, et rhoncus purus mollis ac. Nullam eleifend tincidunt risus quis bibendum. Nunc pretium sed neque sed semper. Curabitur non porttitor massa. Vivamus at mi consequat, pulvinar arcu vel, suscipit arcu. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Praesent pretium ex ligula, ac posuere augue interdum et. Sed auctor sed sapien vel tincidunt. Pellentesque non ullamcorper dolor, eu aliquet justo. Ut orci nisl, tincidunt quis tincidunt id, fermentum ac nisi. Sed odio nisi, dictum eu nulla sit amet, rutrum porta erat. In quis diam nec purus egestas lobortis nec ac urna.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer sed augue eget augue luctus tincidunt. In blandit orci vel dignissim sollicitudin. Sed pretium est tortor, et rhoncus purus mollis ac. Nullam eleifend tincidunt risus quis bibendum. Nunc pretium sed neque sed semper. Curabitur non porttitor massa. Vivamus at mi consequat, pulvinar arcu vel, suscipit arcu. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Praesent pretium ex ligula, ac posuere augue interdum et. Sed auctor sed sapien vel tincidunt. Pellentesque non ullamcorper dolor, eu aliquet justo. Ut orci nisl, tincidunt quis tincidunt id, fermentum ac nisi. Sed odio nisi, dictum eu nulla sit amet, rutrum porta erat. In quis diam nec purus egestas lobortis nec ac urna.

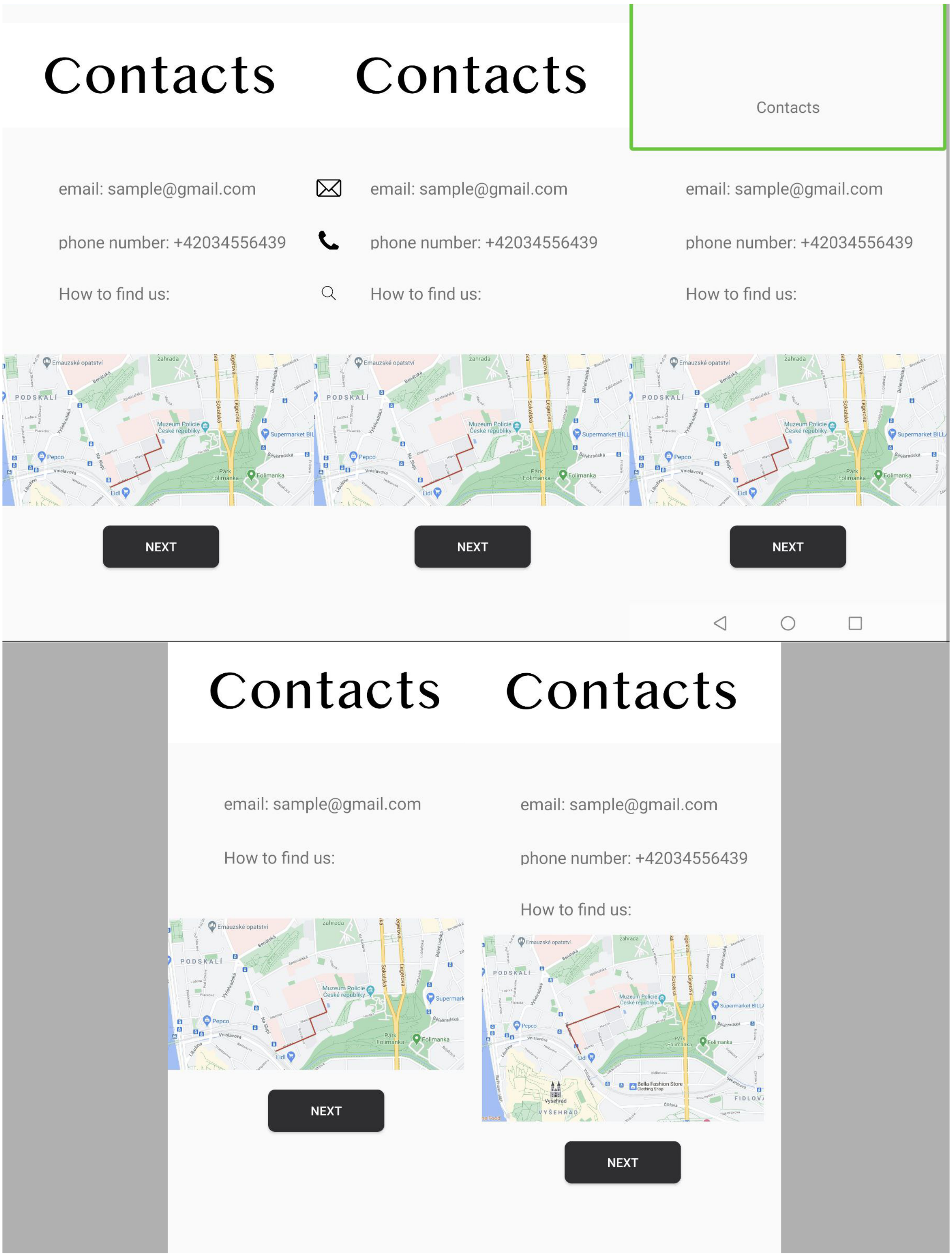
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer sed augue eget augue luctus tincidunt. In blandit orci vel dignissim sollicitudin. Sed pretium est tortor, et rhoncus purus mollis ac. Nullam eleifend tincidunt risus quis bibendum. Nunc pretium sed neque sed semper. Curabitur non porttitor massa. Vivamus at mi consequat, pulvinar arcu vel, suscipit arcu. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Praesent pretium ex ligula, ac posuere augue interdum et. Sed auctor sed sapien vel tincidunt. Pellentesque non ullamcorper dolor, eu aliquet justo. Ut orci nisl, tincidunt quis tincidunt id, fermentum ac nisi. Sed odio nisi, dictum eu nulla sit amet, rutrum porta erat. In quis diam nec purus egestas lobortis nec ac urna.

Figure 6.1: Comparison of the UI adapted by the library

6.2 Content adaptation

Individuals with impairments might struggle to try to understand the content originally displayed on the screen. The alternative version of important information can be implemented as a more profitable approach. Changing the textual or graphic content can benefit users with each type of impairment. To demonstrate the content adaptation based on the type of impairment, the sample of the “Contact us” window was implemented. It contains a picture with the text, e-mail address, phone number, and a map screenshot with a directory.

Sometimes, the textual content such as advertisements or sales announcements into the picture because it can be accompanied by the bright graphic elements that attract more attention. However, it does not benefit the users with visual impairment, because the screen readers are not able to read the text from the image. Individuals with hearing impairment more likely would communicate via e-mail. The destination can be reached by the different routes, and for wheelchair users, the application can show the route that is more comfortable. Some individuals with cognitive impairments may require icons next to the textual content for a better understanding of the context. Figure 6.2 illustrates how the developer can adapt the UI with the help of the library's listeners as follows: without adaptation, for users with cognitive impairment, for visually-impaired users, for users with auditory impairment, and for users with motor impairment.



Contacts

email: sample@gmail.com

phone number: +42034556439

How to find us:

Contacts

email: sample@gmail.com

phone number: +42034556439

How to find us:

Contacts

email: sample@gmail.com

phone number: +42034556439

How to find us:

Contacts

email: sample@gmail.com

How to find us:

Contacts

email: sample@gmail.com

phone number: +42034556439

How to find us:

Figure 6.2: Comparison of the UI adaptation based on the impairment identified by the library

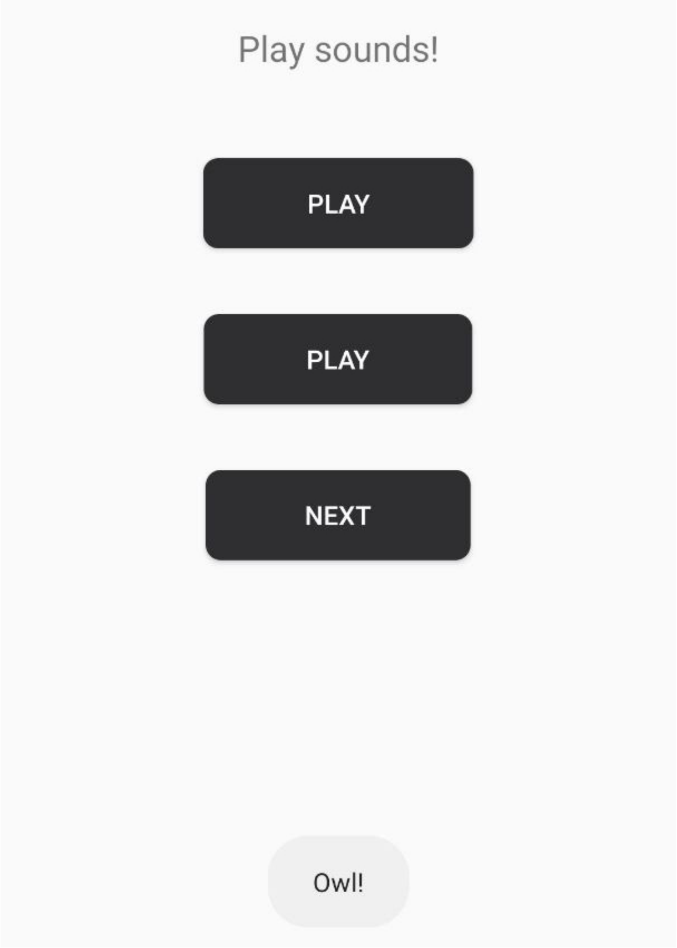


Figure 6.3: Textual description of the sound for users with auditory impairment

6.3 Alerts for deaf or hard of hearing users

Individuals with auditory impairments can experience difficulty in noticing the audio-based notifications or sounds. The library can send a vibration alert and describe the sound if it determines that the user suffers from hearing loss or deaf. Figure 6.3 illustrates how the user with auditory impairment can receive a textual description of the sound.

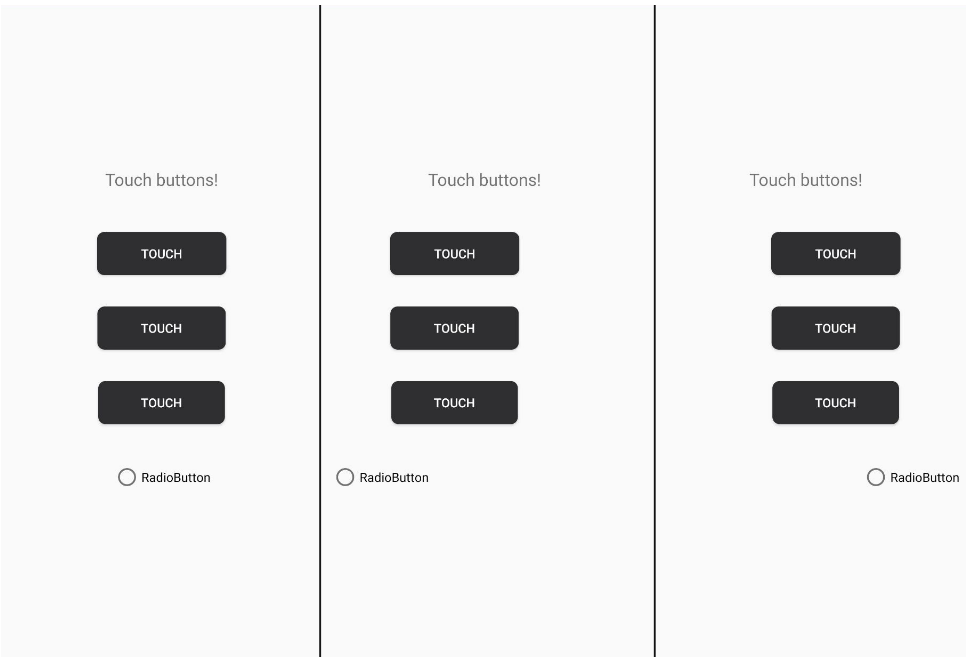


Figure 6.4: UI adaptation based on the dominant hand detection technique

6.4 Changing the position of the UI elements

The last example that appears in the demonstrative application is changing the positioning of the elements, based on the dominant hand technique. The closer the interactive elements are to the user's dominant hand, the easier is the device to use. Figure 6.4 depicts how the positioning of the elements can change if the library detects the dominant hand.

Chapter 7

Testing

One of the main objectives of this thesis is to test the identification and adaptation techniques among real users. Unfortunately, because of the thesis topic's narrow specialization and COVID-19 pandemic, it was complicated to find the advanced mobile users with each area of impairments mentioned in this work. Thus, five advanced mobile users with low vision were picked for the user testing. Due to the pandemic, the testing could not be performed in person. Instead, it was held via messaging platform that suggests a mobile screen sharing feature.

7.1 Specification

The object of testing is a demonstrative application described in chapter 6. The testing subject is an advanced mobile user with low vision who owns a device run on Android. The minimum required version of Android has to be Android 7.0 (Android Nougat). The participants were suggested to go through the demonstrative application with static UI and then they had to go through the application that was modified with the help of the library. The participants had to go through the application a few times to see how it changes with:

1. Enabled TalkBack.
2. Enabled captioning.
3. Enabled Switch Access.
4. Installed application called Otsimo [60].
5. Shaking the mobile device.

The camera was run in the background and the participants did not know about its purpose.

7.2 Testing method and questionnaire

The usability of suggested techniques was measured by the System Usability Scale (SUS)[59]. The questionnaire consisted of 10 questions where the responses ranged from "Strongly agree" to "Completely disagree". The participants had to reply to the following questions:

1. I think the mobile applications I use on an everyday basis are not implemented well for individuals with disabilities.
2. Sometimes I have a hard time reading the textual content of some applications.
3. I find the idea of hiding the part of the system for individuals with impairments discriminative.
4. I was confused by how the elements suddenly changed in front of me.
5. I would prefer to change the size of the elements myself rather than by the application.
6. I think the important textual information should not be stored as an image.
7. I think the changes suggested in the demo application cannot benefit individuals with impairments.
8. The methods of UI adaptation for individuals with impairments can benefit individuals without impairments as well.
9. I find the idea of user interface adaptation based on the type of impairment promising.
10. I would like to have interactive elements on the side of my dominant hand.

7.3 Data interpretation

The answers were scored from 1 to 5, where 1 is "Totally disagree" and 5 is "Totally agree" for questions with a positive sentiment. For questions with a negative sentiment, 1 is "Totally agree" and 5 is "Totally disagree". The sum of the total score from every participant has to be multiplied by the value of 2.5 to convert the results into a 100 point grading system. Table 7.1 shows the question evaluation. The SUS score that is above 68 is considered as "above average". Thus, the SUS score must be above a 68 to call the system successful.

Answer	Points for 1, 2, 6, 8, 9, 10	Points for 3, 4, 5, 7
Totally agree	5	1
Agree	4	2
I don't know	3	3
Disagree	2	4
Totally disagree	1	5

Table 7.1: Questions evaluation.

7.4 Testing results

Table 7.2 illustrates how the participants replied to the questions. The final score of the application is 89. It means that the idea of UI adaptation for disabled users and suggested methods is successful. However, the participants did not like that the UI elements suddenly change. Moreover, several participants are not sure if the suggested changes can benefit individuals with impairments. As for the camera, the UI adapted only twice to the third participant.

Participant	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Sum * 2
P1	3	4	4	2	2	5	2	5	5	3	87.5
P2	4	2	5	3	1	5	3	5	3	3	85
P3	4	4	2	2	4	4	4	5	5	4	95
P4	1	3	4	5	1	4	3	3	4	5	82.5
P5	4	4	4	3	3	5	4	5	3	3	95

Table 7.2: Testing results.

Chapter 8

Installation

The demonstrative application can be launched on devices run on Android OS only. In addition to that, the minimal required Android version is Android 7.0 (Android Nougat). The vibration and camera permissions must be granted.

To launch the project

- Unzip an archive file with the project
- Open the project in Android Studio.
- Build it with gradle

Chapter 9
Conclusion

The aim of this bachelor thesis was to suggest the adaptation techniques and methods that can simplify the interaction between individuals with impairments and disabilities, and UI. In the research, the areas of impairments were reviewed. During the research, the tools that are used by disabled and individuals with impairments were observed. Also, the ways of improving the UI for different areas of disabilities and impairments were specified.

Based on the information obtained from the research, the identification and UI adaptation techniques and methods were suggested. The suggested techniques and methods were used to develop the library for mobile devices run on Android OS. The library is able to identify the type of impairment based on the data received from the user and their mobile device, and adapt the UI elements based on the impairment.

The demonstrative application was developed with the help of the library. The usability of the techniques was tested on five users with low vision and measured by SUS. The testing participants approved the techniques with the final SUS score was 89. All bachelor thesis objectives were completed.

Chapter 10

Future work

The idea for future work is to focus on one area of disability and develop adaptation techniques that would be more accurate. Disease and impairment identification is, indeed, a broad and promising topic, that can be divided into many categories. However, working on the UI adaptation for disabled requires the accuracy. Another approach is to pick only one identification technique and use it for advanced research. The eye infection can be analyzed by blood vessels. The tremor detection technique can be improved in collaboration with medical institutions.

The major plan is to collaborate with the support center for people with special needs for collecting user data and performing frequent user testing in person, which is really important in this field. As an example, ČVUT has a support center for students called ELSA. [61] Working together with disabled would be more profitable, because they can give useful advice for improving the adaptation process.

Bibliography

[1] GSMA. *The Mobile Economy 2020.* Available online: https://www.gsma.com/mobileeconomy/wp-content/uploads/2020/03/GSMA_MobileEconomy2020_Global.pdf

[2] GSA Government-wide IT Accessibility Program. *IT Accessibility Laws and Policies.* Available online: https://www.section508.gov/manage/laws-and-policies

[3] Debbie Stone Caroline Jarrett Mark Woodroffe Shailey Minocha. *User Interface Design and Evaluation.* 1st Edition. 2005. ISBN: 9780120884360.

[4] Wilbert O. Galitz. *The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques.* 3rd Edition. 2007. ISBN: 978-0-470-14622-4.

[5] Travis Lowdermilk. *User-Centered Design: A Developer's Guide To Building User-Friendly Applications.* 1st Edition. O'Reilly Media. April 23, 2013. ISBN-10: 1449359809.

[6] Laura Kalbag. *Accessibility for Everyone.* A Book Apart. 2017. ISBN: 9781937557621.

[7] Burger King. Burger King logo. Available online: https://www.burgerking.cz/

[8] Czech Technical University in Prague. Czech Technical University in Prague logo. Available online: https://www.fel.cvut.cz/en/

[9] Colblindor. Coblis — Color Blindness Simulator. Available online: https://www.color-blindness.com/coblis-color-blindness-simulator/

[10] National Institute on Deafness and Other Communication Disorders (NIDCD). *Age at Which Hearing Loss Begins.* November 30, 2012. Available online: https://www.nidcd.nih.gov/health/statistics/age-which-hearing-loss-begins

[11] Brusilovsky, Peter & Maybury, Mark. (2003). *From Adaptive Hypermedia to the Adaptive Web. Communications of the ACM.* 45. 10.1145/506218.506239.

[12] World Health Organization. *WHO launches first World report on vision.* Available online: https://www.who.int/news-room/detail/08-10-2019-who-launches-first-world-report-on-vision

[13] Colour Blind Awareness. Available online: http://www.colourblindawareness.org

[14] Hearing Loss Association of America. *Types of hearing loss along with the causes and treatments.* Available online: https://www.hearingloss.org/hearing-help/hearing-loss-basics/types-causes-and-treatment

[15] Neura. *Motor Impairment Blog.* Available online: https://motorimpairment.neura.edu.au

[16] American Psychiatric Association. *Diagnostic And Statistical Manual Of Mental Disorders.* Fifth Edition. 2013. ISBN: 9780890425541.

[17] Data Point. *Adult Literacy in the United States.* July 2019. Available online: https://nces.ed.gov/pubs2019/2019179.pdf

[18] Vestibular Disorders Association. Available online: vestibular.org

[19] Canadialog. *Brailliant B 80 (NEW generation) Braille Display* Available online: https://www.canadialog.com/produit/brailliant-b-80-new-generation-braille-display

[20] Amazon. *AbleNet BigKeys LX Large Print Computer Keyboard USB Wired (BKLXBQ)* Available online: https://www.amazon.com/AbleNet-BigKeys-Computer-Keyboard-BKLXBQ/dp/B003SGEZZY

[21] Amazon. *HDE Large Print Keyboard Wired Large Letter Key Keyboard USB Keyboards for Visually Impaired (Yellow Keys)* Available online: https://www.amazon.com/HDE-Computer-Keyboard-Oversized-Individuals/dp/B071L3YXGX

[22] Keyboard Specialists. *X-keys Orby Switches* Available online: https://www.keyboardspecialists.co.uk/products/xk-orby-switch-r

[23] Tecla. *Ablenet Pillow Switch* Available online: https://gettecla.com/products/ablenet-pillow-switch

[24] Medgadget. *Ultra Cheap 3D Eye Tracking for Quadriplegics and Other Seriously Disabled Patients.* July 2012 Available online: https://www.medgadget.com

[25] Dr. Robert D. Atkinson and Daniel D. Castro. *Accessibility for People with Disabilities. Digital Quality of Life: Understanding the Personal and Social Benefits of the Information Technology Revolution.* ITIF. Available online: https://itif.org/files/DQOL-7.pdf

[26] Medline Plus. *Mobility aids.* 2016. Available online: https://medlineplus.gov/mobilityaids.html

[27] Center on Technology and Disability. *Assistive technology 101.* 2018. Available online: https://www.ctdinstitute.org/sites/default/files/file_attachments/CTD-AT101-V4.pdf

[28] Perkins. *School for the blind eLearning. An overview of Braille Devices.* Available online: https://www.perkinselearning.org/technology/blog/overview-braille-devices

[29] American Foundation for the Blind. *Refreshable Braille Displays.* Available online: https://www.afb.org/node/16207/refreshable-braille-displays

[30] AbilityNet. *Keyboard and mouse alternatives and adaptations.* Available online: https://abilitynet.org.uk/factsheets/keyboard-and-mouse-alternatives-and-adaptations

[31] JAWS. *Job Access With Speech.* Available online: https://www.freedomscientific.com/products/software/jaws/

[32] PlayDots. *TwoDots.* Available on Google Play: https://play.google.com/store/apps/details?id=com.weplaydots.twodotsandroid

[33] W3C. *Web Content Accessibility Guidelines (WCAG) 2.2. W3C Working Draft.* 11 August 2020. Available online: https://www.w3.org/TR/2020/WD-WCAG22-20200811/

[34] Color Contrast Checker. Available online: https://colourcontrast.cc

[35] Northern Ireland Assembly. *Guidance for Assembly staff on involving people with autism in Committee meetings and events.* Available online: http://www.niassembly.gov.uk/visit-and-learning/

[36] Evan Brown. Full Fabric. *How to design visual learning resources for neurodiverse students.* Available online: https://blog.fullfabric.com/

[37] Lewis Wake. University of St. Andrews. *Designing for users who are deaf or hard of hearing.* Available online: https://digitalcommunications.wp.st-andrews.ac.uk/2020/02/12/designing-for-users-who-are-deaf-or-hard-of-hearing/

[38] Val Head. *Designing Safer Web Animation For Motion Sensitivity.* Available online: https://alistapart.com/article/designing-safer-web-animation-for-motion-sensitivity/

[39] Craig Grannell. The Guardian. *Why iOS 7 is making some users sick.* 27.09.2013. Available online: https://www.theguardian.com/technology/2013/sep/27/ios-7-motion-sickness-nausea

[40] Mango. *Linen button blazer.* Available online: https://shop.mango.com/cz-en/women/jackets-blazers/linen-button-blazer_67087647.html?c=45

[41] García-Magariño, I., Medrano, C., Plaza, I. et al. *A smartphone-based system for detecting hand tremors in unconstrained environments.* Pers Ubiquit Comput 20, 959–971 (2016). `https://doi.org/10.1007/s00779-016-0956-2`

[42] Parra, V & Enrique, George & Huerta, Mónica & Member, Ieee & Marzinotto, Alejandro & González, Ricardo & Alvizu, Rodolfo. (2013). *A Smartphone Application for Parkinson Tremor Detection.* 10.13140/2.1.3418.2405.

[43] Paul Whittington, Huseyin Dogan, Keith Phalp & Nan Jiang (2020) *Detecting physical abilities through smartphone sensors: an assistive technology application, Disability and Rehabilitation: Assistive Technology,* DOI: 10.1080/17483107.2020.1839135

[44] Boukhvalova AK, Kowalczyk E, Harris T, Kosa P, Wichman A, Sandford MA, Memon A, Bielekova B. *Identifying and Quantifying Neurological Disability via Smartphone.* Front Neurol. 2018 Sep 4;9:740. DOI: 10.3389/fneur.2018.00740. PMID: 30233487; PMCID: PMC6131483.

[45] Mikian Musse. *Detecting Autism Spectrum Disorder in Children With Computer Vision.* Available online: https://towardsdatascience.com/detecting-autism-spectrum-disorder-in-children

[46] Ketabdar, Hamed & Polzehl, Tim. (2009). *Tactile and visual alerts for deaf people by mobile phones.* 253-254. 10.1145/1639642.1639701.

[47] Judith Harkins, Paula E. Tucker, Norman Williams, Jeff Sauro, *Vibration Signaling in Mobile Devices for Emergency Alerting: A Study With Deaf Evaluators,* The Journal of Deaf Studies and Deaf Education, Volume 15, Issue 4, Fall 2010, Pages 438–445. Available online: `https://doi.org/10.1093/deafed/enq018`

[48] Malik, Sadaf & Kanwal, Nadia & Asghar, Mamoona & Ali, Mohammad & Karamat, Irfan & Fleury, Martin. (2019). *Data Driven Approach for Eye Disease Classification with Machine Learning.* Applied Sciences. 9. 10.3390/app9142789.

[49] Sikder, Niloy & Chowdhury, Md & Arif, Abu & Nahid, Abdullah. (2019). *Early Blindness Detection Based on Retinal Images Using Ensemble Learning.* Available online: `https://www.researchgate.net/publication/336829911_Early_Blindness_Detection_Based_on_Retinal_Images_Using_Ensemble_Learning`

[50] Vijay Renga. (2019). *Clinical Evaluation of Patients with Vestibular Dysfunction.* Article ID 3931548. DOI: 10.1155/2019/3931548.

[51] Broadband Search. *Mobile vs. desktop internet usage.* Latest 2020 data. Available online: `https://www.broadbandsearch.net/blog/mobile-desktop-internet-usage-statistics`

[52] GlobalStats. *Mobile Operating System Market Share Worldwide.* Available online: https://gs.statcounter.com/os-market-share/mobile/worldwide https://www.broadbandsearch.net/blog/mobile-desktop-internet-usage-statistics

[53] Google Git. *SimulateColorSpacePreferenceController.java class.* Available online: https://android.googlesource.com/

[54] OpenCV official website. Available online: https://opencv.org/

[55] MLKit official website. Available online: https://developers.google.com/ml-kit

[56] Android developers. *CameraX.* Available online: https://developer.android.com/training/camerax

[57] GitHub. accessibilitools by Novoda. Available online: https://github.com/novoda/accessibilitools

[58] GitHub. seismic by square https://github.com/square/seismic

[59] Usability.gov. *System Usability Scale (SUS)* Available online: https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html

[60] Otsimo. *TwoDots.* Available on Google Play: https://play.google.com/store/apps/details?id=com.otsimo.app

[61] ELSA Centre *TwoDots.* Available online: https://www.elsa.cvut.cz/index.php?menu=english