Vistula University

The Faculty of Computer Engineering, Graphic Design and Architecture

Program of study Computer Science

Anatolii Shcherbak

Student number 63570

DESIGN AND IMPLEMENTATION OF AN ACCESSIBILITY-FOCUSED INTERACTIVE GAMING SYSTEM FOR SIMULATING DISABILITIE

The engineering thesis

written under the supervision of

mgr inż. Stanislav Mishchenko

Warsaw, 2026

Contents

Student Statement

I, Anatolii Shcherbak, declare that this thesis is my own work and has not been submitted elsewhere for academic credit.

Synopsis

This thesis outlines the development of an interactive gaming system designed to be accessible to players with disabilities while incorporating features to simulate disabilities for educational purposes. The system aims to promote inclusiveness and empathy, targeting gamers, and people interested in related topics. It leverages modern game development technologies and adheres to accessibility guidelines.

Keywords

Disability, Game design, challenges, empathy, market, Game development, Unity

1 Introduction

In this Thesis will be analyzed current existing games focused on disabilities, their audience market, technologies used there, how they could be used in other projects and significance of those projects. As well as Accessibility-focused features that are most used in games such as subtitles, sound navigation, color inversion etc. The project proposes a video game designed to simulate various disabilities, including color vision deficiency, as a means of delivering a novel and educational player experience. Through an interactive narrative, players will engage with stories inspired by real individuals with disabilities, gaining insight into their daily challenges. Gameplay incorporates mini games in which the simulated disabilities function as constraints or "debuffs," providing challenges calibrated to encourage both immersion and sustained engagement. This design aims to foster empathy and understanding while maintaining the motivational aspects of traditional game mechanics.

The thesis focuses on developing an interactive game incorporating features to simulate disabilities, fostering empathy and understanding. The aim of the project is to show that games made with a focus on disabilities can also be interesting for a person who is healthy. The system aims to be inclusive and educational, adhering to guidelines such as those from the Game Accessibility Guidelines1. Potential users include people with disabilities, educators, and just usual gamers. The system can be used in education and entertainment.

According to the World Health Organization (WHO**), 1.3 billion people**, or about **16%** of the **world's population**, have a **"significant disability**". This includes a broad range of impairments - physical, sensory, intellectual, or psychological - that, in combination with environmental or personal factors, reduce a person's ability to fully participate in social, economic, and daily activities. 2

Why It Matters for Game Development:

Ethical / Social & Business Reasons:

- **Inclusion and equal opportunity**. In the review Accessibility in Video Games: authors assert that many games and the industry treat accessibility as a low priority - which from an ethical and social justice standpoint contributes to exclusion of disabled players from a cultural and social activity widely enjoyed by others.3 Designing with accessibility in mind supports the principle that digital entertainment should be inclusive and available to as many people as reasonably possible.

- **Diversity of needs and fairness**. Disabilities are diverse - ranging from mild visual impairments, color vision deficiencies, to motor or cognitive limitations. Accessibility features (e.g., adjustable contrast, scalable UI, colorblind modes, remappable controls, alternative input modalities) help ensure that games are enjoyable for people with different abilities, reducing barriers to participation.

- **Social responsibility and reputation**. Game developers and publishers who prioritize accessibility show social responsibility, signal respect for diversity, and contribute to reducing exclusion. This can build goodwill, align with human rights norms, and reflect positively on their brand in a global, varied community of players.

Business / market / financial reasons:

- **Large potential audience**. Given the high global prevalence of disabilities and impairments, accessible design expands the potential market. Many players who might be excluded by default game designs could become customers if games are made accessible.

  On mainstream media side, the article The Independent titled "Video game makers aren't doing enough to cater for **gamers with disabilities**, study finds" reports on a poll of disabled gamers where **81% reported struggle** due to inaccessible features (poor controls, unreadable text, fast gameplay, flashing lights); **39%** said they **were forced to** stop playing or **abandon a game**. 4

Photo 1 Survey Study of gaming with disabilities

Source:<Seppo.io >, date 01.01.2026

- **High spending power & market growth**. Inclusive design can tap into a demographic that may be neglected by standard game design. As demographic trends (aging populations, chronic conditions) and global health burdens increase, the proportion of potential players with disabilities may grow - making accessibility a long-term commercial advantage.

- **Regulatory and reputational risk mitigation**. As awareness and legal frameworks around disability rights grow globally, companies that ignore accessibility risk public criticism, regulatory scrutiny, or exclusion from markets where disability rights are enforced. Proactively building inclusive games mitigates these risks.

- **Innovation and competitive differentiation**. Accessible design often leads to innovations (e.g., UI flexibility, multiple control schemes, adaptive audio, assistive

options) that can improve the experience for all players - not only those with disabilities. That can become a competitive strength and a feature distinguishing developers/ publishers committed to inclusion.

1.1 Potential Users and Target Audience

- Gamers who love challenges
- People who are curious about disabilities

1.2 How to Use the Implemented Project

The game is structured into five chapters, each presenting the narrative of a character with a distinct disability, employing a conventional visual novel storytelling framework. A key innovation is the dynamic adaptation of the game environment and interface to reflect the specific disability portrayed in each chapter. For instance, in the chapter depicting a character with color vision deficiency, the game's color scheme is adjusted to grayscale for the player, accompanied by chapter specific mini games designed to simulate the challenges experienced by the character. This approach facilitates emotional engagement by immersing players in authentic experiences derived from real life accounts. All system modifications are automatically applied in accordance with the selected chapter, ensuring seamless integration between narrative and gameplay mechanics.

1.3 Possible Areas of Usage.

- Entertainment: Inclusive gaming for diverse players.
- Education: Simulating the disability to understand how feels everyday life of person with different kind of disabilities

Approximate date of sending to the Promoter:..... (obligatory field)

(field for the Promoter)

Date of receipt by the Promoter	Mark

2 Analysis of Selected Design Solutions

This chapter reviews existing games and systems related to accessibility and disability simulation, providing a foundation for the proposed project.

To investigate further we should answer first about what disability is and what Accessible game is.

Definition of Disability

Disability is defined as "a mismatch between the needs of the individual and the service, product or environment offered." This means that anyone can experience a disability, and that it can be a short term or situational condition. Envision what challenges gamers with these conditions might have when playing your game, and think about how your game can be better designed for them. 5

Definition of What Makes a Game Accessible

For a game to be considered accessible, it must account for various forms of interaction and perception. Interfaces should be clear and adaptable, while user experiences must balance challenge with accessibility. 6

Existing solutions:

Visual Accessibility Solutions:

- Subtitles and Captions: Textual representation of spoken dialogue and key audio cues to assist players with hearing impairments.
- High Contrast Interfaces and Scalable Text: Improves visibility for users with low vision by ensuring sufficient contrast between UI elements and allowing font sizes to be increased.
- Colorblind Modes: Adjusts colors or applies filters to accommodate different types of color vision deficiency, ensuring important visual information remains distinguishable.
- Visual Audio Indicators: Displays on screen prompts or icons for critical sounds, helping users who cannot rely on auditory cues.

Motor Accessibility Solutions:

- Customizable Controls and Adaptive Inputs: Allows remapping of keys and compatibility with adaptive controllers for players with limited mobility.
- Simplified Action Schemes: Features such as one-handed play or hold to confirm actions reduce accidental inputs and allow easier interaction.

Audio Accessibility Solutions:

- **Independent Volume Controls:** Separate adjustment of music, effects, and dialogue ensures players can tailor audio levels according to their needs.

- **Audio to Visual Redundancy:** Critical audio cues are supplemented with visual indicators to aid players with hearing impairments.
- **Text to Speech or Narration:** Converts menus or in game text into audio output, providing additional accessibility for visually impaired users.

Cognitive and UX Accessibility Solutions:

- **Consistent and Clear UI Layouts:** Reduces complexity and helps users with cognitive or learning impairments navigate interfaces effectively.
- **Tutorial and Assistance Modes:** Provides step by step guidance to familiarize players with mechanics at a comfortable pace.
- **Motion Reduction Options:** Eliminates or reduces camera shake, rapid flashes, and other visual effects that could cause sensory overload.

Engine Level Accessibility Tools

Modern game engines provide tools to assist developers in implementing accessibility:

- **Unity:** While Unity does not include a dedicated accessibility toolkit, its UI and input systems allow developers to create scalable interfaces, remappable controls, and visual/audio feedback systems. Developers often combine built in features with custom scripts or third party plugins. (Unity Support).7
- **Unreal Engine:** Unreal supports screen readers for UI elements and provides APIs for integrating accessibility features such as subtitles, input remapping, and visual indicators. These tools allow developers to implement inclusive features that improve usability for a variety of impairments. (Unreal Documentation). 8

2.1 Review of Existing Accessibility-Focused Games

Accessibility in video games has developed over many decades, with some titles pioneering features that later became industry standards. This section highlights selected games.

Zork: Grand Inquisitor - Early Subtitle Implementation

One of the earliest widely recognized video games to include subtitles for all spoken dialogue was Zork: Grand Inquisitor (1997). As video games transitioned from purely text-based narratives to fully voiced audio experiences, hearing-impaired players increasingly faced barriers to accessing important plot information. The introduction of subtitles in this game addressed these barriers by offering a text representation of spoken dialogue, providing better accessibility to the narrative. This implementation is an important early step in the development of accessibility practices and laid the foundation for subtitles to become a standard feature in narrative-driven video games.

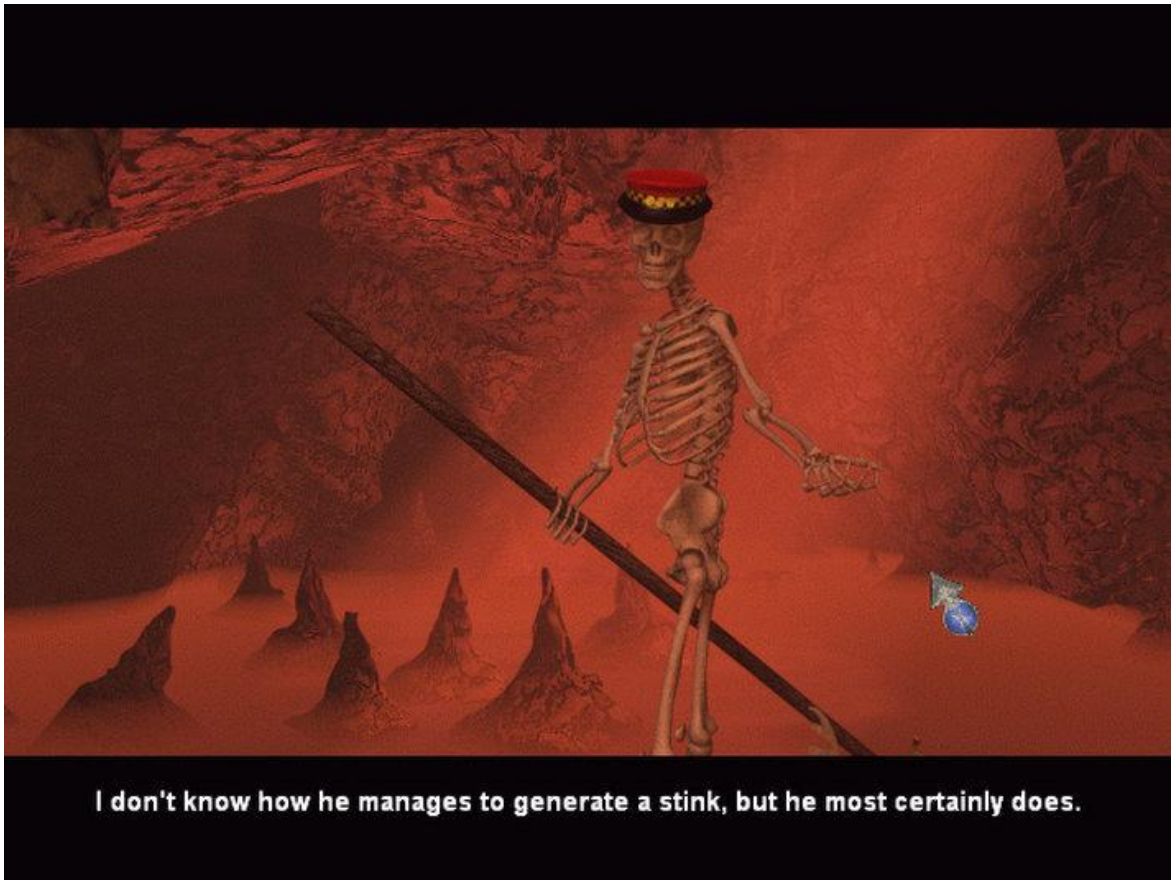


Photo 2 Example Game (Zork: Grand Inquisitor)

Source:<Steam>, date 01.01.2026

Audio Accessibility Solutions Example: Gears 5

Gears 5 serves as a prominent example of audio accessibility in a mainstream game. Beyond enhanced subtitle support, the game integrates audio-based gameplay cues that communicate spatial and navigational information. A key example is the *navigational ping* feature, where a beeping sound increases in volume and frequency as the player approaches significant objectives, helping players locate goals without relying solely on visual or textual indicators. Additionally, varied audio feedback for combat and interactive events supports comprehension of game states for players using assistive audio strategies or alternative sound processing. These design elements illustrate how complex game systems can incorporate auditory accessibility to benefit players with diverse sensory needs

Photo 3 Example Game (Gears 5)

Source:<gearsofwar>, date 01.01.2026

NeuroRacer (2013) - Cognitive Function Game

According to a 2013 study published in the journal Nature by neuroscientists at the University of California, San Francisco**Error! Reference source not found.**: NeuroRacer is a scientifically developed cognitive training game that represents an important milestone in the use of video games for cognitive accessibility and neuropsychological research. Developed by neuroscientists at the University of California, San Francisco, the game was designed primarily to study and improve cognitive control, attention, and multitasking abilities, particularly in older adults. Unlike traditional entertainment-focused games, NeuroRacer was created as a research-driven interactive system, positioning it as an early example of games intentionally designed around cognitive accessibility principles.

The gameplay of NeuroRacer combines two simultaneous tasks: a driving task that requires the player to keep a vehicle on a winding road, and a secondary visual signal task that demands quick responses to specific on-screen cues. This dual-task structure directly targets executive cognitive functions such as divided attention, task switching, and working memory. Importantly, the game dynamically adapts its difficulty based on the player's performance, ensuring that cognitive demands remain challenging without becoming overwhelming. This adaptive difficulty system is a key accessibility feature, as it allows players with varying cognitive abilities to engage meaningfully with the game.

From a cognitive accessibility and user experience (UX) perspective, NeuroRacer demonstrates how games can be designed to accommodate cognitive limitations rather than penalize them. The interface is intentionally minimalistic, avoiding unnecessary visual complexity or distracting elements that could increase cognitive load. Clear visual cues, simple controls, and consistent feedback mechanisms help players focus on the core

cognitive tasks without confusion. These design choices align with accessibility principles aimed at supporting users with reduced attention span, slower reaction times, or age-related cognitive decline.

Although NeuroRacer does not address motor accessibility to the same extent as modern accessible games, its simplified control scheme reduces physical demands and prioritizes cognitive interaction over precise motor skills. This makes it accessible to a broader range of users, including those with limited gaming experience or mild motor impairments. The game's primary contribution lies in demonstrating that accessibility can extend beyond physical input and sensory accommodations to include cognitive inclusivity as a core design goal.

In the context of accessibility-focused game development, NeuroRacer is historically significant because it helped establish games as valid tools for cognitive training and assessment. Its success influenced subsequent research and commercial titles that incorporate adaptive difficulty, cognitive load management, and UX simplification as accessibility strategies. As such, NeuroRacer serves as an important early example of how game design can support cognitive accessibility through scientifically informed mechanics and user-centered design.

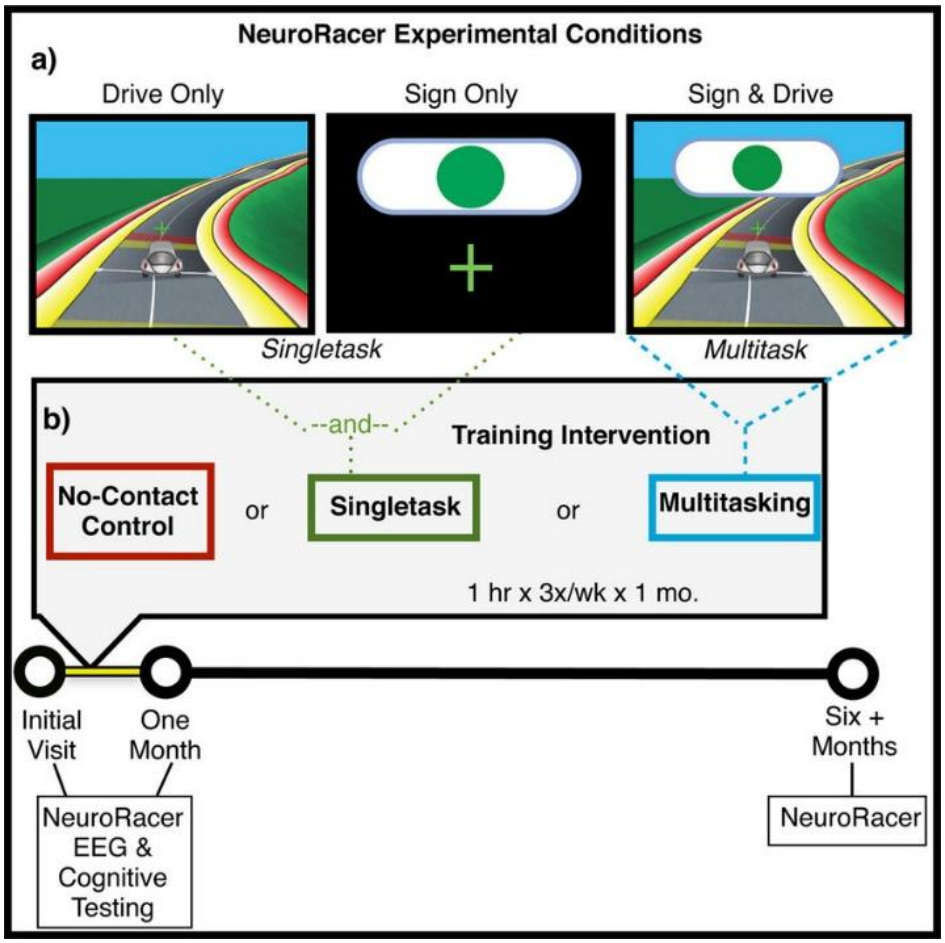


Photo 4 Example NeuroRacer (2013)

Source:<researchgate>, date 01.01.2026

2.2 Analysis of Disability Simulation Games

Disability simulation games are a subgenre of video games designed specifically to provide users with experiences that simulate the challenges faced by individuals with various disabilities. These games serve as both a form of education and empathy-building, offering players the opportunity to experience the world through the perspective of someone with a specific disability. By immersing players in the challenges associated with disabilities, such games aim to foster understanding, raise awareness, and ultimately promote inclusivity within the gaming community. This chapter analyzes the design principles, objectives, and impact of disability simulation games.

Design Objectives of Disability Simulation Games

Disability simulation games are intentionally designed to provide players with a **first-person perspective** on disability experiences, either through simulation of sensory impairments, cognitive challenges, or physical limitations. The goal is not merely to challenge the player but to **sensitize** them to the difficulties faced by individuals with disabilities in navigating the world. Unlike traditional accessibility tools, which focus on providing players with disabilities the means to play conventional games, disability simulation games aim to engage a broader audience in understanding and experiencing disability firsthand.

The **immersive nature** of disability simulation games relies heavily on environmental design, sensory cues, and user interface (UI) adjustments that reflect the player's limited ability. This unique design approach aims to evoke empathy by highlighting the complexities of day-to-day life that are often overlooked in mainstream game environments.

Empathy and Awareness through Gameplay

One of the primary objectives of disability simulation games is to foster **empathy** and **awareness**. By confronting players with challenges that are not their own, these games encourage the player to experience frustration, limitations, and moments of triumph, which can lead to a more nuanced understanding of disability.

Case Study 1: *The Vale: Shadow of the Crown* (2020)

A noteworthy example of a disability simulation game is *The Vale: Shadow of the Crown* (Falling Squirrel, 2020), a narrative-driven action game designed specifically for visually impaired players. This game removes the reliance on visual stimuli, placing players in an immersive auditory environment where the primary method of interaction is through sound. The game is entirely **audio-based**, with **3D spatial audio** used to guide players through the environment and narrative.

Photo 5 Example *The Vale: Shadow of the Crown* (2020)

Source:<Steam>, date 01.01.2026

Core Design Features

In *The Vale: Shadow of the Crown*, players control a blind protagonist who embarks on a quest within a fantasy world. The gameplay primarily focuses on **narrative exploration** and **combat** through **audio cues**. Using spatialized audio, players navigate through soundscapes, interacting with the environment by listening for sounds such as footsteps, the rustle of leaves, or distant voices. Combat, often a central element in many action games, is translated into a series of audible cues and vibrations, requiring players to rely on **auditory feedback** rather than visual indicators.

The decision to design the game in this manner addresses several significant barriers faced by visually impaired players in traditional games, where reliance on visual cues often limits participation. By focusing on auditory input, *The Vale* offers a compelling example of how **inclusive game design** can both entertain and educate.

Photo 6 Example *The Vale: Shadow of the Crown* (2020)

Source:<Steam>, date 01.01.2026

Impact and Reception

Upon its release, *The Vale: Shadow of the Crown* received positive feedback for its **innovative approach** to game design. The game's focus on sound design and immersive audio technologies has made it one of the few mainstream examples of a game specifically designed for players with visual disabilities. This approach has not only provided a unique and engaging experience for visually impaired players but has also fostered **greater awareness** of the challenges blind and low-vision gamers face when interacting with traditional gaming environments (Leighton, 2020).10

Critics have highlighted the way in which the game challenges traditional perceptions of gameplay, where visual experiences are considered essential. By embracing audio as the primary mode of interaction, *The Vale* demonstrates that there are alternative ways to create an engaging narrative and game mechanics that can resonate with players who face sensory limitations (Davis, 2020).11

Case Study 2: *BlindSide* (2012)

Another prominent example of a disability simulation game is *BlindSide* (2012), a **text-based adventure game** designed for blind players. Like *The Vale: Shadow of the Crown*, *BlindSide* uses **audio cues** and **narrative-based exploration** as the central gameplay mechanic. However, unlike the fantasy setting of *The Vale*, *BlindSide* focuses on a dystopian environment in which the player navigates a world in complete darkness.

Photo 7 Example *BlindSide* (2012)

Source:<blindsidegame.com>,date01.01.2026

Core Design Features

The game is unique in that it incorporates **speech synthesis** and audio signals to simulate the experience of blindness. Players control a protagonist who has lost their sight and must navigate the world by interpreting auditory signals. The game challenges players to rely entirely on their hearing and memory, simulating the process of **mental mapping** that visually impaired individuals might employ in real life.

One of the standout features of *BlindSide* is the **real-time speech synthesis** that describes objects and environments in a vivid auditory landscape. This allows players to create a mental map of their surroundings without visual reference, simulating the experience of navigating through the world without sight (Bryce, 2007).12

Reception and Educational Value

BlindSide has been praised for its ability to offer both **entertainment and educational value** by promoting understanding of the experience of blindness. Its success lies in its simplicity and ability to communicate the importance of auditory cues in navigating the world. It also serves as an example of how **gameplay mechanics** can be tailored to provide insights into the lived experiences of people with disabilities (Snyder, 2008).1313

Challenges in Disability Simulation Games

While disability simulation games can be effective in promoting empathy, they also face certain challenges in their design and reception. One challenge lies in the **accuracy** of the experience. As disability simulation games attempt to convey complex sensory experiences, they risk oversimplifying or misrepresenting the difficulties faced by people with disabilities. For example, although *The Vale: Shadow of the Crown* offers an engaging auditory experience, it cannot fully replicate the breadth of challenges experienced by visually impaired people in real-life situations, such as the nuances of mobility or social interactions (Leighton, 2020).10

Additionally, **game accessibility** and **sustainability** remain concerns. While these games are designed to increase empathy, they may not always provide the level of accessibility required for **long-term engagement**. As these games target a specific group, the broader gaming community may have limited exposure to their **educational potential**.

Conclusion

Disability simulation games, such as *The Vale: Shadow of the Crown* and *BlindSide*, represent important contributions to the discourse on **game design and accessibility**. These games offer players a unique opportunity to experience the world from the perspective of individuals with disabilities, thereby promoting **awareness**, **empathy**, and a deeper understanding of the challenges faced by these communities. While such games are valuable tools for education, they must continue to evolve to ensure that they reflect the **complex realities** of living with a disability. Moving forward, there is a need for continued innovation in creating games that are not only entertaining but also serve as platforms for greater inclusivity and social impact.

2.3 Comparison of Current Technologies and Methodologies

As the demand for accessible video games increases, developers are looking toward both **game engines** and **accessibility frameworks** to create inclusive experiences. **Game engines** like **Unity** and **Unreal Engine** provide the foundation for game development, while specialized **accessibility frameworks** and guidelines offer structures to implement features that cater to players with disabilities. This chapter compares the current technologies and methodologies for accessibility, focusing on the capabilities of **Unity** and **Unreal Engine**, and examines the role of accessibility guidelines, particularly the **Game Accessibility Guidelines (GAG)**.

Unity Engine

Unity is known for its flexibility, ease of use, and large developer community. It is popular for 2D, mobile, and 3D games, and offers accessibility support primarily through plugins and some built-in features.

Visual Accessibility (Blind/Low Vision)

- Mobile screen reader support via iOS/Android VoiceOver and TalkBack.
- Dynamic text scaling and bold fonts through Accessibility Settings API.
- Colorblind filters can be applied using post-processing shaders.
- High contrast and reduced transparency available via Apple Accessibility Plugin.

Auditory Accessibility (Deaf/Hard of Hearing)

- Closed captioning with official prefab and tutorials.
- Audio visualization via custom indicators.
- Per-channel volume control with Audio Mixer.
- Haptic feedback supported on gamepads and mobile devices.

Motor Accessibility (Physical/Mobility)

- Input remapping at runtime using RebindActionUI.
- Supports multiple devices and adaptive controllers.
- Switch control integration via Apple Accessibility Plugin.
- Keyboard and gamepad UI navigation through UGUI Selectable system.
- Extensible to eye tracking, voice input, and other alternative methods.

Cognitive Accessibility (Neurodiversity, Learning)

- Localization with multi-language support.
- Assist modes adjusting gameplay pace.
- Simplified UI and tutorial systems.
- Motion sensitivity settings to reduce blur, camera shake, and other effects.

Accessible functions Data from (https://accessforge.io/accessibility-game-engines)14

Unreal Engine

Unreal Engine excels in high-fidelity graphics and AAA game development. While it has a steeper learning curve, it provides extensive built-in accessibility features.

Visual Accessibility

- UMG screen reader support and Slate Screen Reader plugin for full UI narration.
- Built-in color vision deficiency correction (e.g., protanopia, deuteranopia).
- High contrast UI achievable through custom UMG styles.
- Text-to-speech plugins for additional narration and audio feedback.

Auditory Accessibility

- Built-in subtitle system and enhanced captioning (UE 5.6+).
- Visualized sound effects using UMG-based directional indicators.
- Haptic feedback including adaptive triggers.
- Voice chat transcription available via plugins.

Motor Accessibility

- Enhanced Input System with runtime key remapping.
- Alternate control schemes and Co-Pilot dual-controller support.
- Full vibration and adaptive trigger integration.
- Supports alternative inputs through plugins like eye tracking or voice commands.

Cognitive Accessibility

- Time dilation and global slowdown via built-in nodes.
- Persistent objective and quest tracking through UMG.
- Simplified, toggleable UI layouts.
- Interactive tutorial systems.
- IRIS plugin for photosensitivity and flash detection.

Accessible functions Data from (https://accessforge.io/accessibility-game-engines) 14

Comparison of Key Accessibility Features in Unity and Unreal Engine

Feature	Unity	Unreal Engine
Screen Reader Support	Native support for Windows, macOS, Android, and iOS via UnityEngine.Accessibility.	Built-in **Screen Reader Plugin** supports NVDA/JAWS on Windows and VoiceOver on iOS
Text-to-Speech (TTS)	Requires third-party plugins (e.g, Read	**Native TTS Plugin** allows developers to narrate UI

Feature	Unity	Unreal Engine
	Speaker) for real-time menu.	text and element states directly.
UI Semantic Markup	UI Toolkit uses semantic properties (Hints, Descriptions) to label elements for assistive tech.	**UMG Accessibility Settings** in the Details panel allow overriding accessible defaults for widgets.
Color Vision Deficiency	Managed through custom shaders or camera post-processing volumes.	**Built-in CVD Settings** allow real-time simulation and adjustment of color space.
Subtitles / Captions	Manual implementation via UI or through localization-integrated text systems.	**Audio Subtitle Plugin** allows globally enabling/disabling subtitles and supports multiple entries per asset.
Input & Navigation	Supports focus rings and keyboard navigation by default in UI Toolkit.	Supports keyboard/controller navigation via UMG and "Common UI" for cross-platform input
Localization	Advanced text shaping for RTL (Arabic/Hebrew) and Unicode built into UI Toolkit.	Native localization dashboard for text formatting and pluralization support.

Table 1 (Comparison of Key Accessibility Features in Unity and Unreal Engine)

Accessibility Frameworks: Game Accessibility Guidelines (GAG)

The **Game Accessibility Guidelines (GAG)** provide a comprehensive set of recommendations for developers aiming to create inclusive and accessible gaming experiences. GAG covers a wide range of accessibility concerns, including **visual**, **auditory**, **motor**, and **cognitive impairments**, and provides actionable steps for integrating these features into game development processes.

Key Areas of the Game Accessibility Guidelines

1. **Visual Accessibility**: GAG recommends features like **high-contrast modes**,

colorblind support, **subtitle customization**, and **text-to-speech options**. These features are crucial for ensuring that games are playable for those with visual impairments. Both **Unity** and **Unreal Engine** provide tools that align with these guidelines, but Unreal's **Slate UI framework** provides more comprehensive built-in support (Game Accessibility Guidelines, 2021).1

2. **Auditory Accessibility**: Guidelines suggest including **closed captions**, **sound descriptions**, and **auditory cues** to assist players who are deaf or hard of hearing. **Unreal Engine** offers spatialized audio and other auditory enhancements to ensure these features can be implemented effectively. Unity, while flexible, requires third-party solutions to achieve the same level of auditory accessibility (Seddon et al., 2021).

3. **Motor Accessibility**: The GAG outlines how games should allow for **customizable controls** and support for **adaptive controllers**. Unreal's **Blueprint Visual Scripting** system simplifies the implementation of rebindable controls and other motor accessibility features, while Unity developers must rely on external assets or manual coding to achieve similar functionality.

4. **Cognitive Accessibility**: GAG also emphasizes the importance of **simplified UI**, **clear instructions**, and **adjustable difficulty levels** for players with cognitive disabilities. Both engines allow for the implementation of these features, though **Unreal Engine's Blueprint system** makes it easier for developers to integrate them without extensive programming knowledge.

Official website of GAG(GameAccessibilityGuidelines)**(**gameaccessibilityguidelines)14

Conclusion

Both **Unity** and **Unreal Engine** offer robust support for creating accessible games, with each engine providing different strengths in terms of built-in features and customization options. **Unreal Engine** stands out for its comprehensive suite of accessibility tools, including **spatialized audio**, **customizable UI**, and **visual scripting** through Blueprints. It also aligns closely with **Game Accessibility Guidelines (GAG)**, making it an attractive choice for developers focused on accessibility.

On the other hand, **Unity** provides flexibility and ease of use, with a rich selection of third-party plugins available to enhance accessibility. However, it requires more effort to implement advanced accessibility features compared to Unreal, especially for visual and auditory feedback systems.

The **Game Accessibility Guidelines (GAG)** serve as a critical framework for both engines, offering actionable steps to ensure inclusivity. By following these guidelines, developers can create games that are accessible to a wider audience, enhancing the gaming experience for players with disabilities.

Approximate date of sending to the Promoter:..... (obligatory field)

(field for the Promoter)

Date of receipt by the Promoter	Mark

3 Technologies Used

This chapter outlines the technologies employed in developing the gaming system, justifying their selection.

3.1 General Characteristics of Basic Technologies

- Game engines: Unity.
- Programming languages: C# (Unity).
- Accessibility APIs and adaptive hardware support.

3.2 Characteristics of Main Technology for Accessibility

Unity as a Platform

- Unity supports accessibility through scripts, UI adaptation, and custom input handling.
- It's versatile for creating simulations that modify visual, auditory, and motor experiences.
- Community support is strong with tutorials, forums, and accessibility assets.

Unity Accessibility Toolkit (UAT):

- Provides features like screen reader support, color adjustments, font scaling.
- Supports keyboard navigation and focus indicators.

Adaptive Controllers:

- Input devices like Xbox Adaptive Controller allow users with limited mobility to interact with games.
- Unity can map custom inputs to these controllers.

Simulation Features Developed in My Project:

Each design of (levels) or parts of the games are written in with some sort of disability which represents difficulty for game as well as a tools written for those purposes.

- Level 1: Simulates colorblindness (grayscale/contrast adjustment for whole level).
- Level 2: Simulate deafness (Sound adjusted for player in a distant noise).
- Level 3: Simulate partial blindness (Character blind in one eye, right part of game screen is closed for player to see)
- Level 4: Simulate motor limitations (Character does not have hand so respectfully not all keys/mouses works for Character).
- Level 5: Simulate blindness (Character blind, screen completely black for the rest of the game)

Unity offers a robust environment for implementing accessibility features. While standard tools like the Unity Accessibility Toolkit provide built-in solutions such as screen readers, color adjustments, and input remapping, this project focused on creating custom simulations. For example, the first level modifies color palettes to grayscale, simulating common forms of color vision deficiency. Other levels implement adaptive input response to mimic motor disabilities, and selective audio reduction to simulate auditory impairments. These features were developed in Unity using scripts, shaders, and UI adjustments.

3.3 Justification for Choice of Technology

Community Support

- Unity has an extensive developer community with forums, tutorials, and open-source projects.
- This support is critical when implementing accessibility features.

Accessibility Features

- Unity allows:
  - Custom shaders for visual simulations.
  - Flexible UI modifications.
  - Integration with adaptive hardware.

Suitability for Disability Simulation

- You could implement custom simulations for educational purposes.
- No pre-existing tools fully matched your need for level-by-level disability simulation.
- The decision to use Unity enabled complete control over the user experience, such as reversing colors, modifying controls, or changing audio cues for learning purposes.

Unity was chosen as the primary technology due to its flexibility, extensive documentation, and strong community support. Unlike existing accessibility solutions, which often focus on general usability, Unity allowed the creation of customized disability simulations tailored for educational purposes. Each level of the project presents a different disability scenario, including visual, motor, and auditory impairments, allowing users to experience challenges first-hand. The ability to fully control input handling, UI feedback, and visual rendering made Unity the most suitable platform for this approach.

Approximate date of sending to the Promoter:..... (obligatory field)

(field for the Promoter)

Date of receipt by the Promoter	Mark

4 Project Description

This chapter details the design process, including assumptions and technical specifications.

4.1 Design Document

1. Game Concept:

The game is designed as a visual novel in which each chapter presents a self-contained story focused on a different character living with a disability. The core concept is to combine narrative storytelling with interactive mechanics to both educate the player and create emotional empathy. Each chapter explores the personal experiences, challenges, and feelings of the character, allowing the player to see the world from their perspective. Beyond narrative elements, every chapter includes unique minigames that simulate aspects of the featured disability through gameplay mechanics. For example, in a chapter centered on a deaf character, sound may be distorted, distant, or completely absent, directly affecting player interaction. By integrating storytelling and gameplay, the project aims to raise awareness of disabilities while demonstrating how game mechanics can be used to communicate subjective human experiences.

2. Game Objectives

- Raise awareness of accessibility challenges faced by people with different disabilities by presenting their experiences through interactive storytelling.
- Encourage empathy and understanding by allowing players to experience the world from the perspective of characters with disabilities.
- Explore innovative game design approaches in which disabilities are represented not only narratively but also through gameplay mechanics.
- Use interactive challenges and minigames to simulate sensory and cognitive limitations, transforming them into meaningful gameplay experiences.
- Demonstrate how games can function as educational tools while remaining engaging and emotionally impactful for players.

3. Key Features

- Multiple chapters or levels, each representing a different disability and presented through a unique character and narrative scenario.
- Visual novel–style storytelling that emphasizes character development, emotional depth, and player choice.
- Disability-specific gameplay mechanics and minigames that simulate sensory or perceptual limitations, such as reduced or absent sound, altered visuals, or delayed input.

- Integration of narrative and gameplay, where mechanics directly support the story and emotional experience of each character.
- Educational focus combined with interactive design, ensuring that learning objectives are embedded naturally within the gameplay experience.

4. Accessibility Features:

- **Visual Simulation:** Grayscale or color-reversed palettes for colorblindness.

  Half, or nothing showed on screen to simulate blindness.
- **Auditory Simulation:** Reduced or delayed sound cues for auditory impairments.

5. User Interface (UI) Design

The user interface is designed to be minimalistic and intuitive to support narrative immersion and accessibility. UI elements such as dialogue boxes, menus, and interaction prompts are kept visually clear and unobtrusive, allowing players to focus on the story and gameplay. The interface adapts to each chapter's theme and mechanics, subtly changing to reflect the featured disability while maintaining overall usability. Accessibility considerations include readable typography, clear visual hierarchy, and consistent navigation across all chapters. The UI design aims to balance functional clarity with emotional storytelling, ensuring that interface elements enhance rather than distract from the player's experience.

Photo 8 Project Game (Bondless Colors) **date 11.01.2026**

5. Disability Simulation Modes

Level	Simulated Disability	Technical Implementation	Purpose
1	Visual (Colorblindness)	Grayscale shader, color inversion script	Demonstrates challenges with color perception
2	Auditory Impairment	Muted audio tracks, delayed sound cues	Demonstrates reduced access to auditory information.
3	Visual (partial blindness)	Only Half of the screen is visible	Demonstrates limitations of the visual field.
4	Motor limitations	Some keyboard key or mouse clicks are limited	Demonstrates restricted motor control.
5	Visual/Auditory (Blindness/ partial deafness)	Screen is black, sounds are distant, distorted)	Demonstrates navigation with severely limited sensory input.

Table 2 (Description of levels and disabilities presented in them)

6. Technical Requirements

- **Hardware:** PC, monitor, mouse(recommended)/or touchpad, keyboard, headphones(recommended) or Speaker.
- **Software:** .Exe Build of the game
- **Internet** is required to download the build.

  To run the Unity-based computer game, no additional software is required beyond a compatible operating system. To open and edit the game project, Unity Hub and the Unity Editor are required, along with a code editor such as Visual Studio.

7. Flow and Gameplay

The game begins at Level 1, which presents the story of a colorblind character, Neil. In this level, the game automatically applies grayscale and inverted color settings throughout the entire interface, including the main menu (Look Photo 8). to simulate the character's visual experience. Players are required to complete each level before progressing to the next; subsequent story-levels remain locked until the previous one is finished.

Upon completing a level, the game automatically adjusts settings to simulate the next character's disability. For example, progression from Level 1 to Level 2 involves transitioning from Neil's colorblindness (grayscale, inverted colors) to a deaf character, Soundiel, where colors are presented normally, but audio is distorted or absent. This design allows players to gradually experience abilities and perceptions that were inaccessible in earlier levels, such as distinguishing colors that were previously obscured.

Completing all levels enables the player to gain a comprehensive understanding of each character's life experience. Each new perspective contributes to a fuller, more complete narrative, combining **educational value** with **emotional engagement**, as the game is based on real-life stories and prototypes of individuals with disabilities.

Photo 9 Project Game (Bondless Colors) **date 11.01.2026**

4.2 Detailed Design of Accessibility Features and Simulations

The game incorporates accessibility features and sensory simulations in each level to allow players to experience the challenges faced by individuals with various disabilities. Each level is designed around a specific disability, with technical implementations that replicate the sensory or motor limitations associated with that condition. The goal is to provide both educational and emotional insights into the lived experiences of people with disabilities, while integrating these simulations seamlessly into gameplay.

Level 1: Visual (Colorblindness)

In the first level, the game simulates colorblindness using a combination of a grayscale shader and a color inversion script. These visual adjustments apply to all game elements, including the main menu and in-game environment. This simulation allows players to experience the difficulties associated with distinguishing colors and recognizing visual cues that rely on color perception, demonstrating the challenges faced by individuals with color vision deficiencies. As additional challenge minigame to find right color is provided to player through sound - Each sound represents color and player must find colors accordingly to sounds. This minigame based on real person experience **Neil Harbisson,** a colorblind man who uses physical instrument which tells colors by its frequency to Neil with different sounds.15

Photo 10 Project Game (Bondless Colors) **date 11.01.2026**

Level 2: Auditory Impairment

The second level focuses on auditory impairments. To simulate reduced hearing, the game employs muted audio tracks and delayed sound cues, which affect both environmental sounds and character interactions. This design demonstrates the difficulties of perceiving and interpreting auditory information, emphasizing how sound limitations impact gameplay and daily experiences.

Level 3: Visual (Partial Blindness)

In the third level, partial blindness is simulated by restricting the visible portion of the screen to only half of the normal display area. This mechanic highlights the limitations of the visual field, requiring players to navigate and interact with the game environment while experiencing constrained vision. This approach demonstrates the impact of restricted sight on spatial awareness and situational perception.

Level 4: Motor Limitations

Level four addresses motor limitations by limiting the player's ability to use certain keyboard keys or mouse clicks. This simulation replicates reduced motor control, slowing interaction speed and increasing the difficulty of performing precise in-game actions. It allows players to understand the challenges of engaging with standard input devices when motor abilities are impaired.

Level 5: Visual/Auditory (Blindness / Partial Deafness)

The fifth level combines visual and auditory impairments to simulate blindness and partial deafness. The screen is completely black, and audio is either distant, distorted, or absent. This design demonstrates the difficulty of navigating an environment with severely limited sensory input, emphasizing the reliance on remaining senses and alternative cues to complete tasks.

Conclusion:

Through these targeted simulations, the game provides players with progressive experiences of sensory and motor limitations, enabling a deeper understanding of the challenges faced by people with disabilities. Each level not only conveys the functional impact of a specific disability but also encourages empathy and reflection, making the gameplay both educational and emotionally engaging.

4.3 Project User Manual

Project User Manual

(For Diploma Submission - Accessibility Simulation Game)

1. Introduction

Purpose:

- Explain what the game is and its objective.
- Clarify the target audience: healthy/able-bodied users.
- Highlight the educational goal: understanding the perspective of people with disabilities.

This game is an educational simulation designed to allow healthy users to experience the challenges faced by people with various disabilities. Each level of the game simulates a different impairment, such as visual, motor, or auditory limitations, automatically adjusting gameplay to reflect these experiences. The primary purpose of this project is to increase awareness and foster empathy toward accessibility challenges faced by individuals with disabilities by presenting these challenges through a realistic and experience-based perspective. The project aims to demonstrate how limitations in accessibility affect interaction, navigation,

and decision-making within digital environments.

In addition, the project seeks to illustrate that accessibility features should not be viewed solely as compensatory or auxiliary elements, but as integral components of thoughtful and engaging game design. By embedding accessibility mechanics directly into gameplay, the project explores how inclusive design can enhance player engagement, immersion, and narrative depth while remaining accessible to a wider audience.

2. System Requirements

- **Operating System:** Windows 10 or higher
- **Processor**: Intel Core i3 (2nd generation) or equivalent
  or AMD Athlon II / FX-series or better
- **Memory:** 4 GB RAM
- **Graphics:** Integrated graphics (Intel HD 4000 or equivalent)
  DirectX 10 compatible GPU
- **Storage:** 2 GB free space
- **Input Devices:** Keyboard and mouse

3. Installation and Launch

Guide:

- Step-by-step guide on how to install and run the game.
- Since the game auto-adjusts, no settings configuration instructions are required.

Instructions:

1. Download the game folder to your computer.
2. Open the executable file BoundlessColors.exe.
3. The game will automatically apply Current level accessibility options, (Default Level 1 - monochromacy)picture8*.
4. Click ‘Start’ Button to level selection menu. Click on character image or name on the image to start chosen characters story (By default unlocked only 1 Character Neil, player must finish whole story ‘Level’ to open next characters Story. *After finishing story next characters accessibility configuration would automatically apply for game*) picture9*.

The game has started, no further action is required.

4. Game Overview

Purpose: A brief summary of how the game is structured and what the player will experience.

The game consists of five levels, each simulating a different disability. Players navigate through interactive visual novel story and minigames designed to demonstrate the challenges faced by people with impairments. All accessibility simulations are applied automatically, and players experience the limitations firsthand without having to adjust any settings.

5. Level Descriptions

Level 1: The first level tells the story of 'Neil' an artist suffering from achromatopsia. The player is invited to see the world in black and white and to touch on the experiences and problems of people born with such disabilities. A person with this disability not only cannot recognize colors, but is also sensitive to light, making it difficult for them to look at bright colors. Despite this, there are famous artists in history who created works in color, even though they themselves could never see the "colors" of their paintings. The character is based on the artist Neil Harbisson15, who attached a physical device to his head that reads the frequency of colors and transmits beeps to Neil to tell him what color is in front of him.

Level 2: The second level tells the story of 'Soundiel' a deaf singer. In today's world, we are surrounded by the sounds of cars, animals, people, electronic devices; sound is practically inescapable... Music has become a salvation for many. How painful it must be for those who have devoted their lives to music to lose their hearing. However, there are musicians who have created global hits, even though they themselves cannot hear because they have lost their hearing. For the love of music, they have resorted to various tricks, such as memorizing the rhythm of a song by watching other singers' lips. The story is based on the author's personal memories of a person with deafness and on the story of Mandy Harvey16.

Level 3: The third level tells the story of 'Elias' a police officer who lost his eye and replaced it with a synthetic cyber eye. Although this may sound like a plot from a book about the future, this story was inspired by a real-life story from our present day. Rob Spence17 is a man who has a camera instead of an eye. Unfortunately, our technology has not yet advanced to the point where a "cyber" eye looks natural; it looks quite terrifying, like many other devices that exist to improve the lives of people who have suffered trauma. Although these devices improve lives, they are not yet capable of fully replacing a human organ, their price is very high, and their appearance can be frightening...

Level 4: The fourth level tells the story of 'Jessa' a female blacksmith. Our world is full of injustice and inequality. Despite all our laws on equality, women still face poor treatment and unequal working conditions, especially when it comes to physical labor, due to stereotypes. And for Jesse, it's even harder because she lost her arm, which complicates not only her work but also her everyday life... Many of the things we buy, such as flat-pack furniture, are designed in such a way that we must hold a part with one hand and screw it in with the other. When creating such instructions and designs, we don't even think about people who only have one hand, for example. This story is based on the author's memories of war veterans who lost

limbs and the story of Jessica Smith18.

Level 5: The fifth level is the culmination of all the previous stories from the perspective of a blind, deaf old woman. We don't need eyes to see the suffering of others, or ears to hear their cries. Even though everyday life is difficult for 'Zia Notte', she treats the previous characters as her children and watches over them. Sometimes bright desires can create miracles, and even the blind can see. The character is inspired by many people, as well as the story of Daniel Kish19, who taught himself to make waves with his tongue like bats to use echo location.

6. Controls

- **Interaction:** Spacebar / Left mouse button
- **Exit:** Esc key
- Minigame keyboards buttons which are assigned for minigames are shown during those minigames.
- **Additional** quick buttons for muting/unmuting game, quick save etc., could be seen as well as configured on the option menu.

Note: Input and controls are affected by the simulation automatically. Players will notice differences in responsiveness depending on the level being played.

8. Learning Outcomes

- Gain awareness of the challenges faced by people with visual, motor, or auditory impairments.
- Understand the importance of accessibility design in games and digital media.
- Develop empathy and a user-focused perspective for inclusive design.

9. Troubleshooting

- Ensure your system meets the minimum requirements.
- Verify that all game files are in the same folder.
- Restart the game if any input or audio issues occur.
- *Deleting any files in saving folders would lead to corruption of save file.

Approximate date of sending to the Promoter:..... (obligatory field)

(field for the Promoter)

Date of receipt by the Promoter	Mark

5 Implementation of the Diploma Thesis Project

This chapter describes the development process and integration of components.

5.1 Development Process

The development of the accessibility simulation game followed an iterative approach inspired by Agile methodologies. Each feature of the game was developed in small increments, allowing continuous testing and refinement. Version control was maintained using Git, which enabled tracking changes, managing different versions of the project, and ensuring code stability.

The development process included:

1. **Planning:** Defining game objectives, levels, and disability simulations, researching materials about people with disabilities to create a template for stories and characters
2. **Implementation:** Developing core mechanics in Unity, creating shaders, scripts, and UI components for accessibility features.
3. **Testing:** After each level or feature was implemented, functionality was tested to ensure it correctly simulated the intended disability.
4. **Iteration:** Adjustments and improvements were made based on testing results.

Git repositories were used to manage source code, allowing rollback to previous versions if necessary and ensuring that all changes were documented systematically.

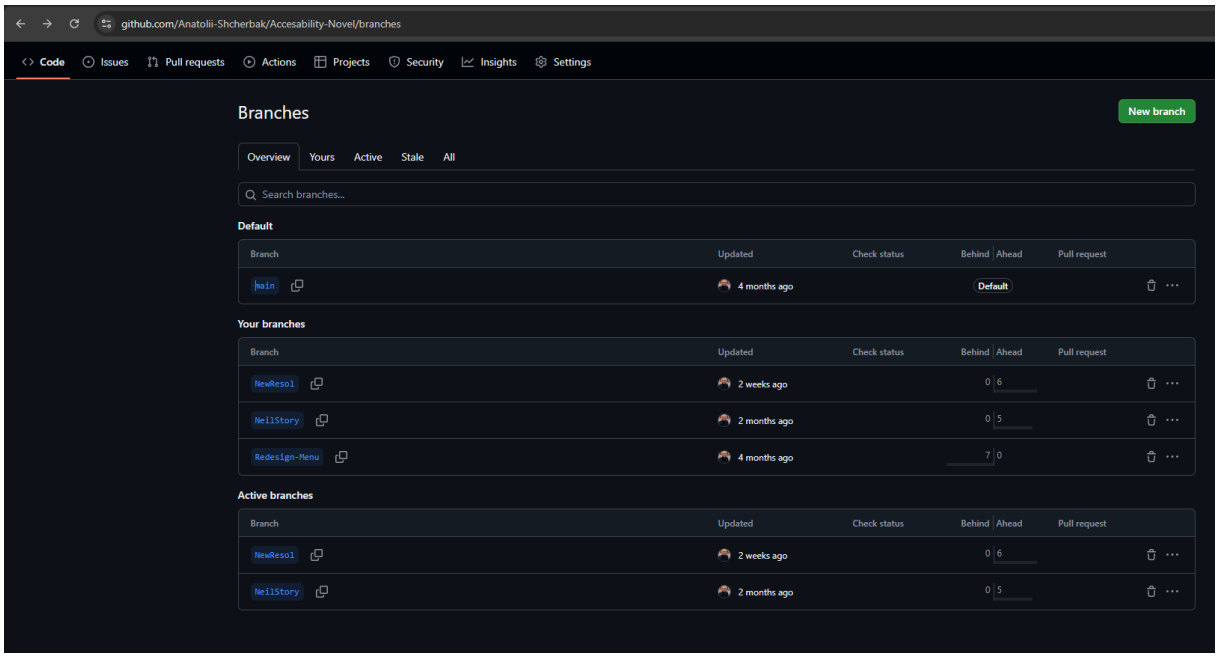


Photo 11 Git Control Project Game (Bondless Colors) **date 14.01.2026**

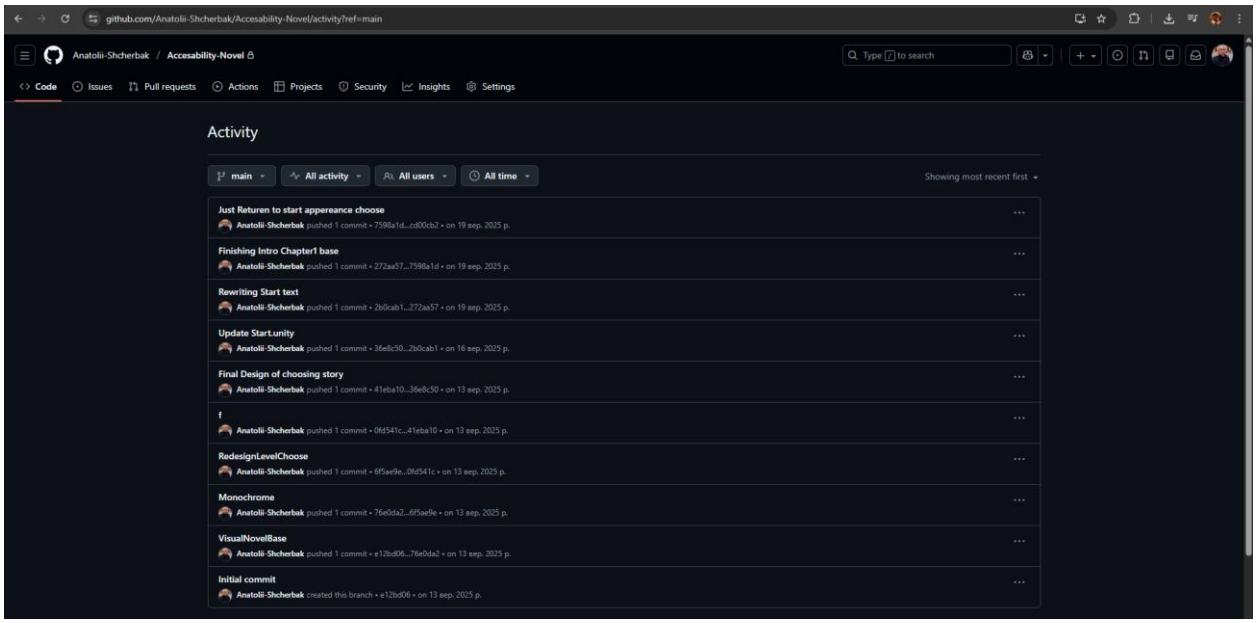


Photo 12 Git Control Project Game (Bondless Colors) **date 14.01.2026**

5.2 Coding and Integration

The game is a visual novel, So, first, a complete foundation for a visual novel was developed, namely a system for parsing text from a TXT file to display lines and names on the screen, as well as commands such as change: scenes, character images, locations, and music… In other words, the plot is written in a text file rather than in the code, game menu. Implemented save and load system, achievement system, and some other standard elements for visual novels that are still in development. Game was implemented entirely in Unity, with scripts and shaders used to create disability simulation modes. Each level was designed as a one scene, with using shaders and scripts specifically for level from settings script with specific features integrated to simulate a different disabilities.

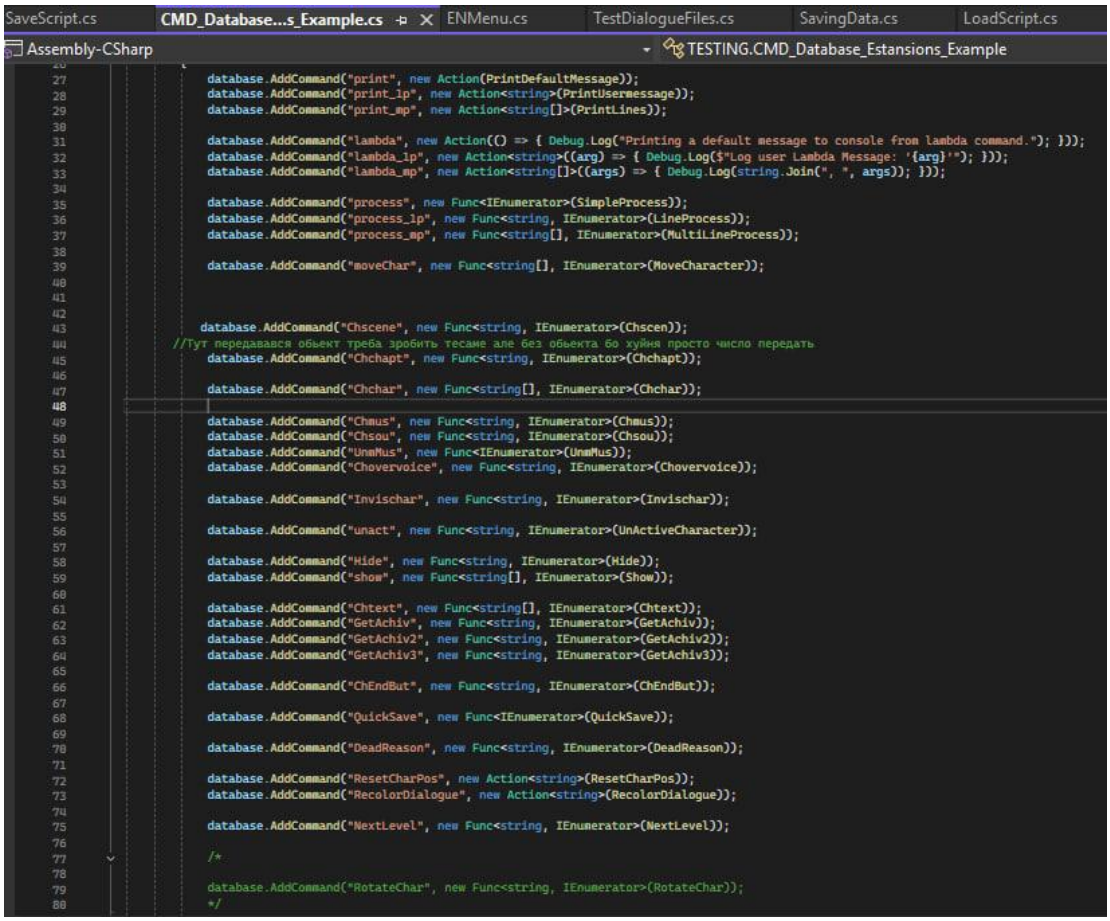


Photo 13 (Project Game) Implemented Command database date 20.01.2026

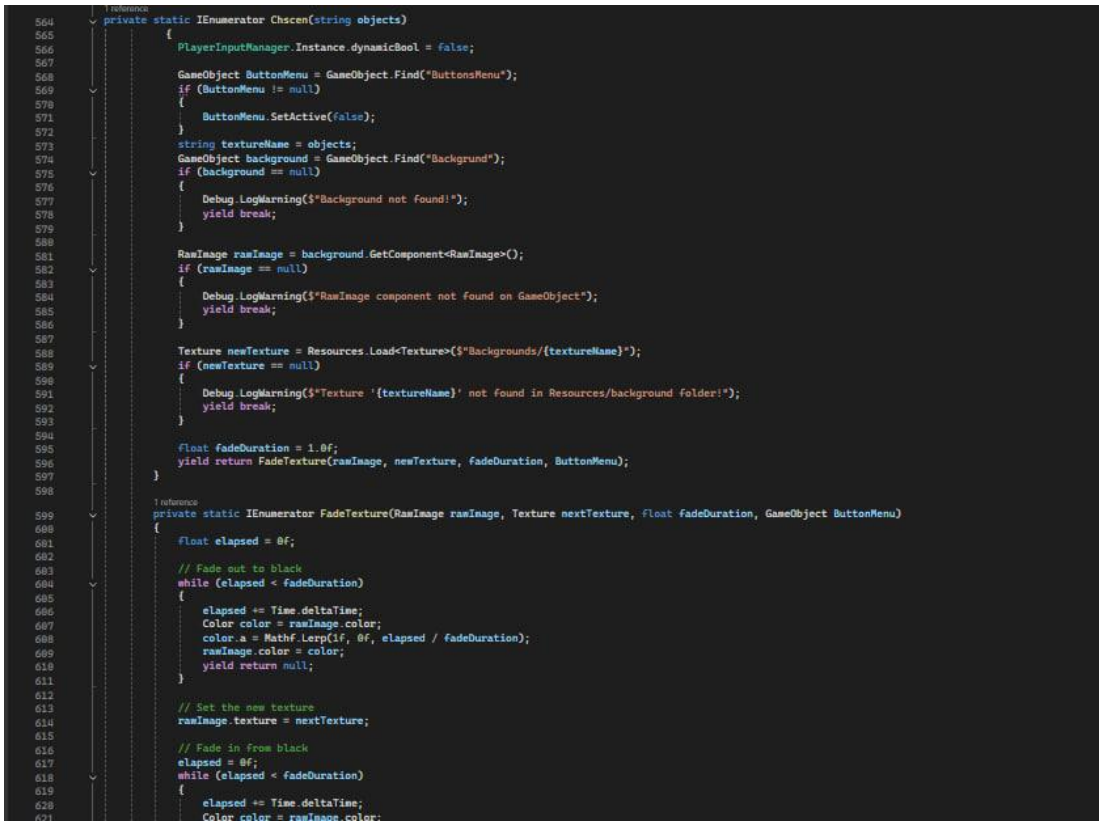


Photo 14 (Project Game) Implemented Command database date 20.01.2026

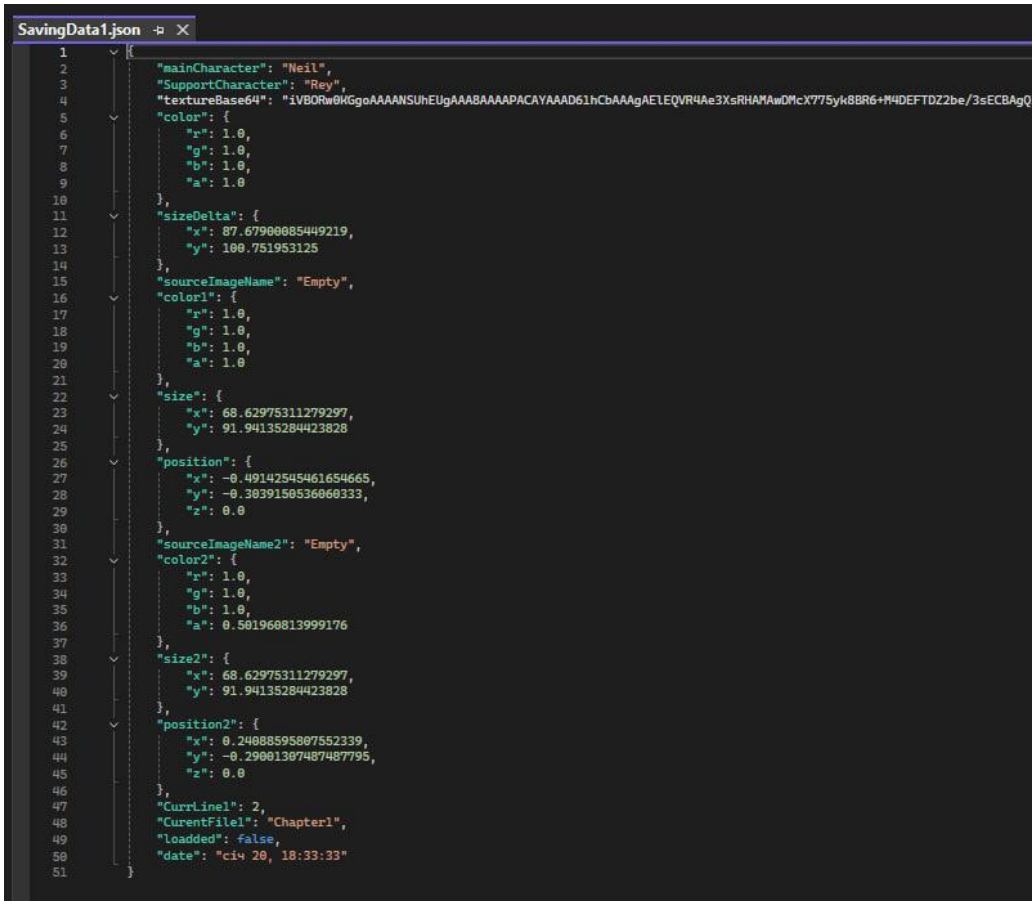


Photo 15 (Project Game) Saved Game data **date 20.01.2026**

On the **Photo 16** above shown json file which is created upon saving game. This file includes: Screenshot of current game as well as current date (to be shown in save/loading menu), Position and image of Characters in scene, Image of 'Scene', Name of txt file which is currently used, Current line in this file, status 'Loaded' for handling possible load issues Example: Saved Line 3 is command for changing scene, since commands are implemented through coroutines to be implemented at the same time, status loaded is marked true upon loading, and changes decreases lines until it wont be text or first line of the script, then changes loaded status to false* This was implemented to avoid any unintended interactions and skips of commands upon loading, as well as game crashes

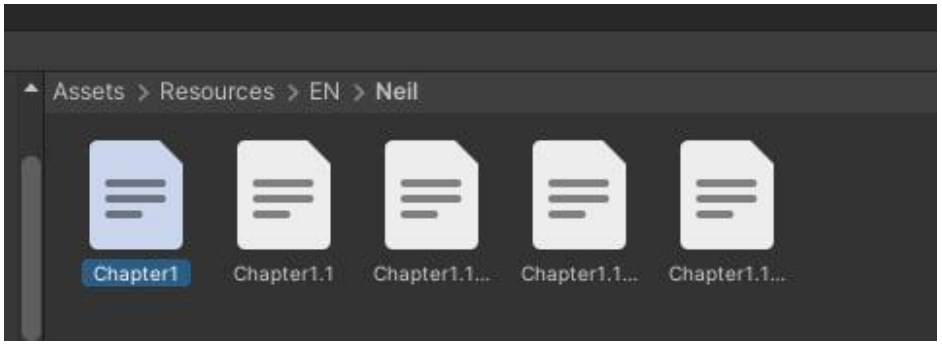


Photo 16 (Project Game) Chapters **date 20.01.2026**

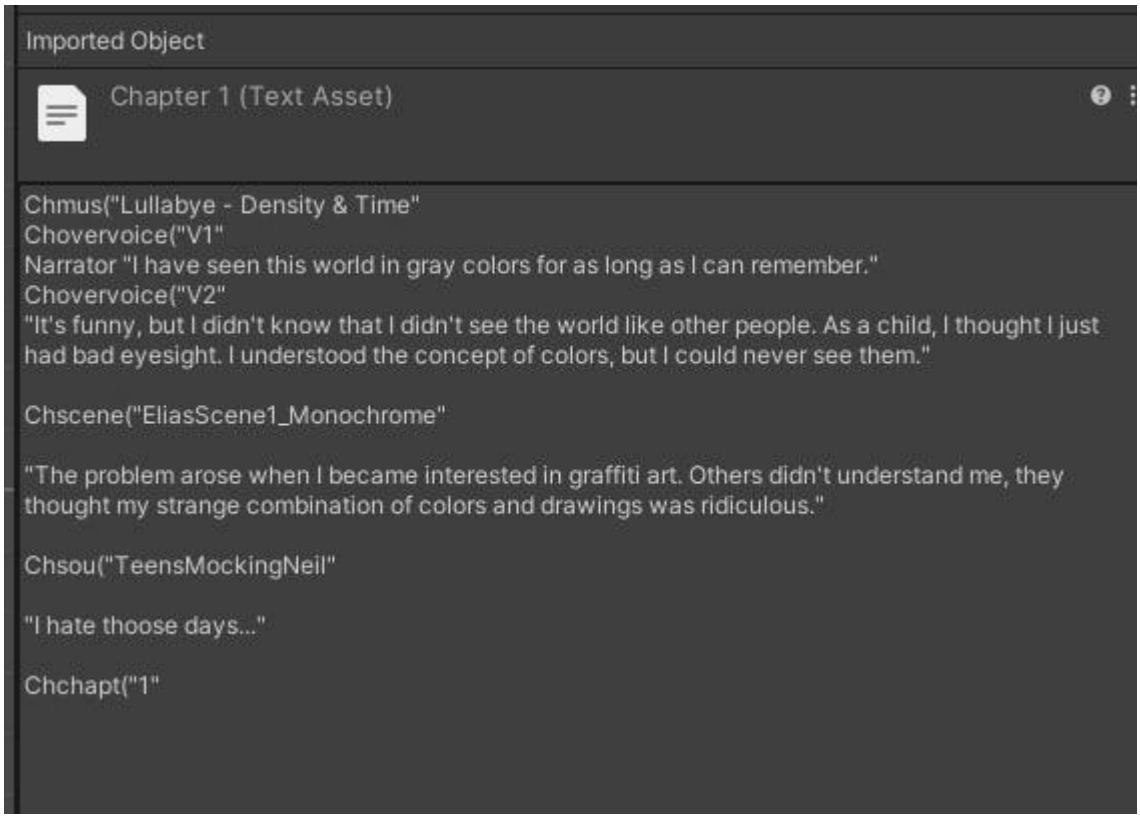
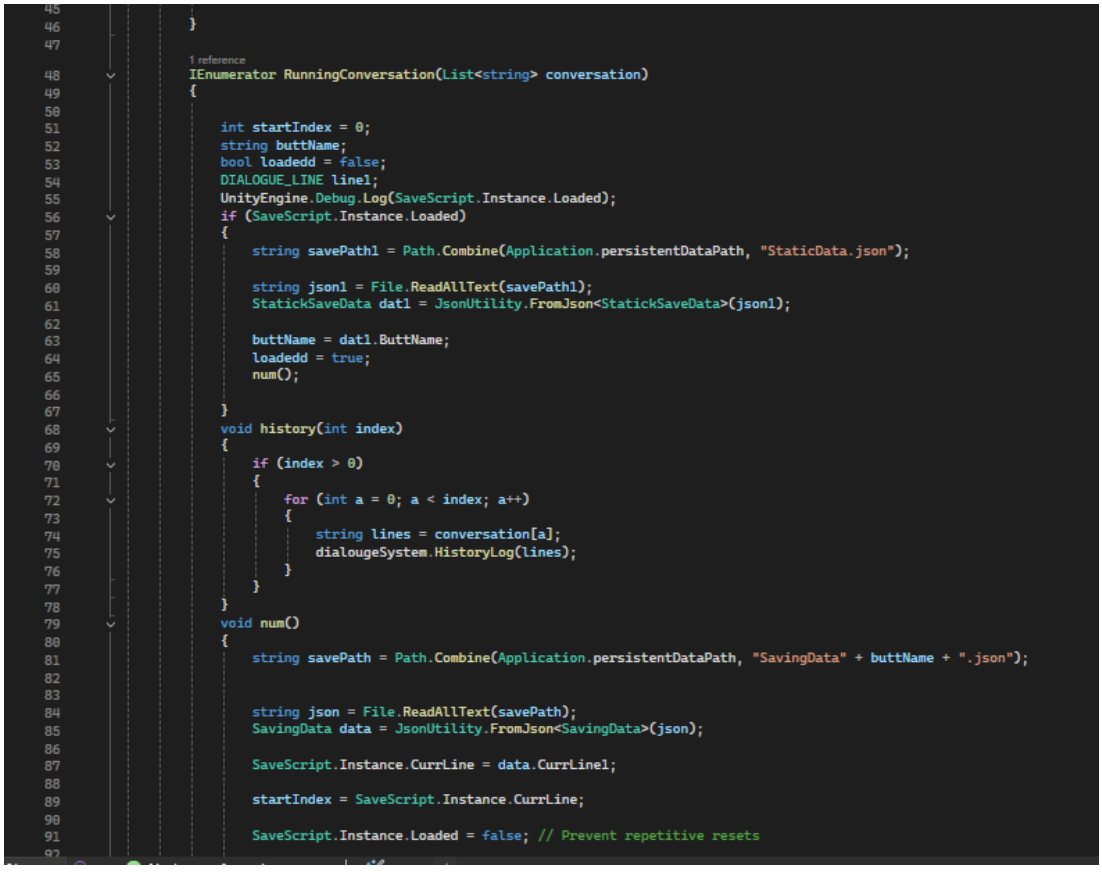


Photo 17 (Project Game) Chapters date 20.01.2026



```
}

1 reference
IEnumerator RunningConversation(List<string> conversation)
{

    int startIndex = 0;
    string buttName;
    bool loadedd = false;
    DIALOGUE_LINE line1;
    UnityEngine.Debug.Log(SaveScript.Instance.Loaded);
    if (SaveScript.Instance.Loaded)
    {
        string savePath1 = Path.Combine(Application.persistentDataPath, "StaticData.json");

        string json1 = File.ReadAllText(savePath1);
        StatickSaveData dat1 = JsonUtility.FromJson<StatickSaveData>(json1);

        buttName = dat1.ButtName;
        loadedd = true;
        num();

    }
    void history(int index)
    {
        if (index > 0)
        {
            for (int a = 0; a < index; a++)
            {
                string lines = conversation[a];
                dialougeSystem.HistoryLog(lines);
            }
        }
    }
    void num()
    {
        string savePath = Path.Combine(Application.persistentDataPath, "SavingData" + buttName + ".json");


        string json = File.ReadAllText(savePath);
        SavingData data = JsonUtility.FromJson<SavingData>(json);

        SaveScript.Instance.CurrLine = data.CurrLine1;

        startIndex = SaveScript.Instance.CurrLine;

        SaveScript.Instance.Loaded = false; // Prevent repetitive resets
```

Photo 18 (Project Game) Conversation Lines upon starting to read txt file date 20.01.2026

Levels Code descriptions:

- **Level 1 - Visual Impairment:** A grayscale shader and color inversion script were applied to all visual elements to simulate colorblindness. This required integrating custom materials and adjusting the UI for readability.

  After competing Level 1, player view is changed for next Pearson, which gives possibility for player to view game colors after starting level 2, which is applied both for the game itself and Game menu, below is comparing pictures of game view for level 2(Upper picture) and level 1 (Picture below). Look at **Photo 20** and **Photo 21***

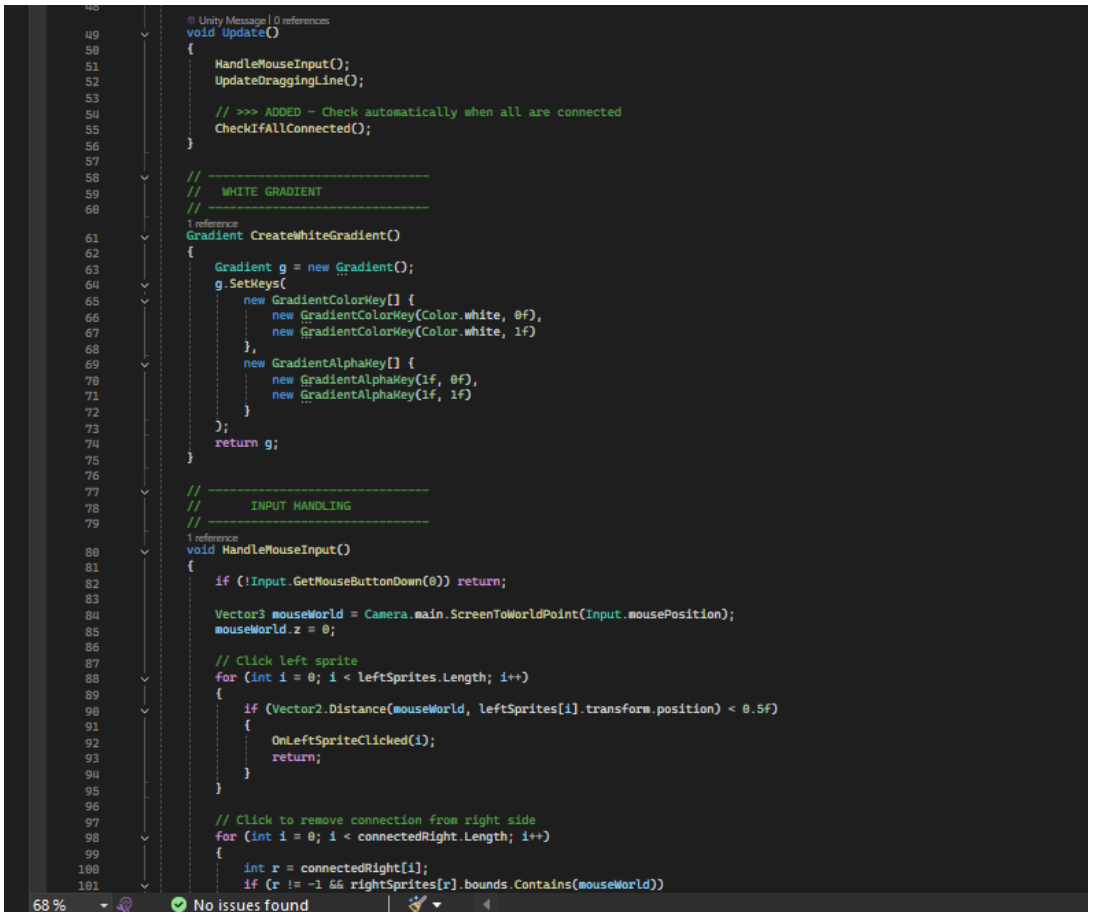
  For this level, a minigame was developed where the player must click on the 'spray' and, according to the frequency of the sounds heard and the number of connections, connect it to the correct color. To do this, you need to click on the spray while holding down the mouse button and, without releasing it, draw a line to the color. At the end of the minigame, a window appears showing the number of correct connections and an offer to finish the minigame or start over. If the minigame is completed without connecting 6 out of 6 correctly, the player gets the first bad ending. If the game is completed with the correct options, the player can continue and open the second level of the next story. Look at **Photo 22** for correct mapping to beat minigame, look for code implementation at Photo **23** and visual representation at **Photo 24***

Photo 19 (Project Game) Comparison of level 2 and level 1 views date 20.01.2026

Photo 20 (Project Game) Comparison of level 2 and level 1 views date 20.01.2026

```
// >>> ADDED - Correct mapping
private int[] correctMapping = new int[] { 5, 0, 3, 4, 2, 1 };
```

Photo 21 (Project Game) Correct Mapping for 'Spray' minigame date 20.01.2026



```
Unity Message | 0 references
void Update()
{
    HandleMouseInput();
    UpdateDraggingLine();

    // >>> ADDED — Check automatically when all are connected
    CheckIfAllConnected();
}

// ---------------------------------
//   WHITE GRADIENT
// ---------------------------------
1 reference
Gradient CreateWhiteGradient()
{
    Gradient g = new Gradient();
    g.SetKeys(
        new GradientColorKey[] {
            new GradientColorKey(Color.white, 0f),
            new GradientColorKey(Color.white, 1f)
        },
        new GradientAlphaKey[] {
            new GradientAlphaKey(1f, 0f),
            new GradientAlphaKey(1f, 1f)
        }
    );
    return g;
}

// ---------------------------------
//       INPUT HANDLING
// ---------------------------------
1 reference
void HandleMouseInput()
{
    if (!Input.GetMouseButtonDown(0)) return;

    Vector3 mouseWorld = Camera.main.ScreenToWorldPoint(Input.mousePosition);
    mouseWorld.z = 0;

    // Click left sprite
    for (int i = 0; i < leftSprites.Length; i++)
    {
        if (Vector2.Distance(mouseWorld, leftSprites[i].transform.position) < 0.5f)
        {
            OnLeftSpriteClicked(i);
            return;
        }
    }

    // Click to remove connection from right side
    for (int i = 0; i < connectedRight.Length; i++)
    {
        int r = connectedRight[i];
        if (r != -1 && rightSprites[r].bounds.Contains(mouseWorld))
```

68 % No issues found

Photo 22 (Project Game) Implementation of ‘Spray’ minigame date 20.01.2026

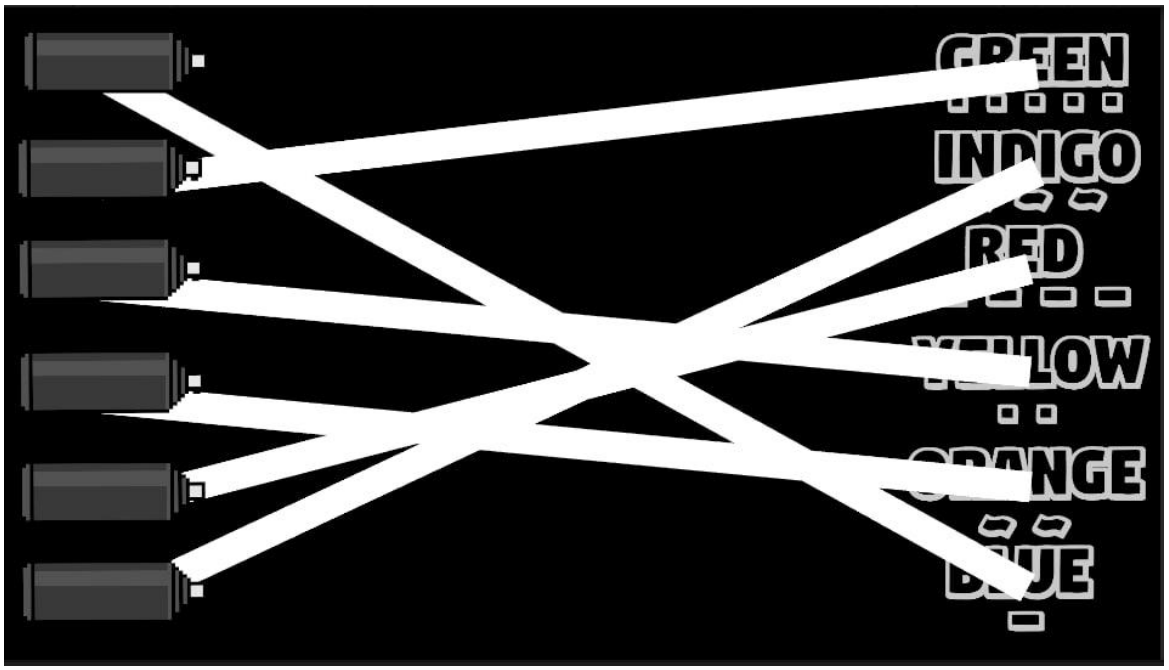


Photo 23 (Project Game) ‘Spray’ minigame date 20.01.2026

- **Level 2 - Auditory Impairment:** For this level, all colors were converted back from grayscale to RGB, but the music tracks and the song sung by the heroine were distorted to simulate deafness, allowing the player to hear approximately how deaf people hear.

Table of Sound effects

Effect Name	Recommended Setting / Value	Action in Audacity
High-Pass Filter	**Frequency:** 400 Hz **Rolloff:** 24 dB per octave	Removes the "muddiness" and bass to make it sound thin.
Low-Pass Filter	**Frequency:** 3500 Hz - 4000 Hz **Rolloff:** 12 dB per octave	Muffles the "sparkle" and high-end clarity.
Distortion	**Type:** Soft Clipping **Drive:** 15% - 25%	Adds that warm, slightly "broken" analog grit.
Vibrato	**Frequency:** 0.5 Hz - 1.5 Hz **Depth:** 5% - 10%	Creates the "pitch wobble" of a damaged cassette tape.
Amplify	**New Peak Amplitude:** -3.0 dB	Use this after the filters to prevent the audio from clipping.
Compressor	**Threshold:** -20 dB **Ratio:** 3:1	"Squashes" the sound together to make it feel more compact.

Table 3 (Changes made to Audio Track in Audacity to simulate Deafness)

- **Levels 3 - 5** are currently under development; the code for these levels has not yet been implemented, and will not be implemented for the demo version for this thesis

Summary:

All scripts, shaders, and assets were integrated within Unity's scene hierarchy. Components such as player controller, environment objects, and interactive elements were linked to the scripts controlling accessibility features. Integration testing ensured that all features worked seamlessly: visual simulations did not interfere with controls, input modifications did not break navigation, and audio adjustments were correctly synchronized with gameplay events.

Version control through Git ensured that each change was documented, enabling systematic debugging and rollback when needed. This combination of Unity development and Git integration facilitated a stable, maintainable, and modular game architecture.

Approximate date of sending to the Promoter:..... (obligatory field)

(field for the Promoter)

Date of receipt by the Promoter	Mark

6 Summary

The design documentation and demo version of the project were presented to a selected group of gamers for evaluation. Overall feedback was positive. The majority of participants supported the concept of representing human disabilities as a meaningful and engaging gameplay challenge. In addition, the narrative approach of the project was well received, particularly its intention to raise awareness and encourage greater interest in understanding the experiences of people with disabilities.

However, a significant conceptual contradiction was identified during the evaluation process. While the game is designed to simulate various forms of disability and is intended to promote awareness and empathy, it currently lacks fundamental accessibility features that would allow people with disabilities to play it themselves. This creates a notable irony, as a project focused on representing disability does not yet implement inclusive or assistive design principles. For instance, the absence of features such as automatic narration in sound-oriented mini-games limits accessibility and highlights an important area for future improvement.

In conclusion, the project demonstrates strong potential both as an entertainment product and as an educational tool. Future development should focus on improving accessibility features in order to ensure that the game aligns more closely with its inclusive goals and can be experienced by a wider range of users.

6.1 Status of the Implemented Project

Current state: functional prototype.

Fully working Visual novel system game .txt file text converted and shown as text on the as well as character speaker name, and image. Commands database to change Images, characters, backgrounds, positions, levels in the .txt file.

Fully working Save/load system with date of saved game and screenshot.

Additional Menus such as: Achievements, Save/load menu, Level choose menu, Options.

Additional buttons: Skip.

Additional mechanic: Base for future language options, created language folders: EN, UA, all chapter files are saved in EN folder. Written separate script and check in important scripts like load for checking which language is used and accordingly to read chapters from folders, default is EN. Not working since UA folder is empty, left for be done after finishing whole game

Level 1 Chapter menu, Main minigame, Main achievements, and color inversion are implemented.

Lacks: Full written story, additional minigame, achievements

Level 2. Chapter menu, Color inversion, corrupted lead song of heroine simulating deafness are implemented

Lacks: Full written story, Main minigame, achievements

Levels 3 - 5. Design document with characters story, and disabilities mechanics that should be applied are created, no Code implementation

Additional buttons in progress: Auto, Config, shortcut buttons, Languages option.

6.2 Necessary Works to Complete the Project

Since the project was created to show that disability itself can be used as an interesting challenge for mechanics in game design, the game's plot text implementation is not important for this diploma. Ideally, it would be best to create a disability for each level, but since this affects every other level, and game as whole, it takes a lot of time, the presented working demo version and the description of the design document detailing which disability should be shown in the next levels are sufficient for this work.

6.3 Possible Improvements and Extensions

Writing full stories for all the characters, implementing levels 3-5, finishing shortcuts buttons, implementing Config and About menus, working auto button.

6.4 Difficulties Encountered

A significant amount of preliminary work was conducted to analyze existing tools aimed at improving accessibility and enhancing the gaming experience for people with disabilities. During this analysis, it was identified that three-dimensional input devices used for position tracking, effects, and similar interactions are often complex and difficult to operate for the target audience. Therefore, the decision was made to develop a two-dimensional project as a more accessible alternative.

One of the main challenges of the project was the development of the game design itself. This required an in-depth study of user feedback, including comments, reviews, and interviews with people with disabilities, in order to better understand their experiences and needs. Based on this research, the goal was to design engaging gameplay challenges that both reflect real accessibility limitations and draw attention to the broader issues faced by this group.

Since the primary characters - players in the game are individuals with disabilities, the project introduces novel gameplay mechanics. As a result, there were no existing scripts, frameworks, or libraries that could be directly applied. This necessitated the development of custom solutions. Frequent modifications to the game logic and mechanics led to code conflicts and increased development time, particularly given that the project was implemented by a single developer.

Approximate date of sending to the Promoter:..... (obligatory field)

(field for the Promoter)

Date of receipt by the Promoter	Mark

7 Bibliography

1. Game Accessibility Guidelines https://gameaccessibilityguidelines.com/
2. World Health Organization
3. ResearchGate https://www.researchgate.net/publication/360457520_Accessibility_in_Video_Games_A_Review
4. The Independent https://www.independent.co.uk/tech/video-game-makers-disabilities-mental-health-b2423844.html?
5. Microsoft https://learn.microsoft.com/en-us/windows/uwp/gaming/accessibility-for-games?
6. Kokkugames https://kokkugames.com/making-games-accessible-lessons-in-ui-ux/
7. Unity Unity Support
8. Unreal Unreal Documentation
9. Anguera, J. A. et al. *Video game training enhances cognitive control in older adults*. Nature, 2013.
10. Leighton, T. (2020). *The Vale: Shadow of the Crown: A New Era of Audio-Based Gaming*. Journal of Game Studies.
11. Davis, M. (2020). *Empathy through Sound: The Impact of Audio-Only Games*. Game Design Review
12. Bryce, J. (2007). *BlindSide: The Evolution of Audio-Only Gaming*. Journal of Interactive Media.
13. Snyder, D. (2008). *Audio Games: A New Paradigm in Inclusive Design*. Technology and Disability.
14. Accessforge https://accessforge.io/accessibility-game-engines
15. Neil Harbisson - The man who hears colour https://www.bbc.com/news/magazine-16681630
16. Mandy Harvey - Turns tragedy into a Triumph https://tanyadalton.com/podcasts/episode-102-how-mandy-harvey-turns-tragedy-into-triumph/
17. Rob Spence - man with a camera inside his eye https://www.bbc.com/future/article/20170721-the-man-with-a-camera-inside-his-eye
18. Jessica Smith - Daily life as new mum with one hand https://www.mamamia.com.au/jessica-smith-interview
19. Daniel Kish - Human echolocation https://youthtimemag.com/the-eyelessman-taught-himself-to-see/

List of Pictures

List of Tables

Attachments

1. GitHub link for this Thesis: https://github.com/Anatolii-Shcherbak/DyplomWork_AnatoliiShcherbak_Id63570
2. Link for web version of the game: WebVersion Note* please use full screen mode!

Approximate date of sending to the Promoter:..... (obligatory field)

(field for the Promoter)

Date of receipt by the Promoter	Mark