BACHELOR‘S THESIS ASSIGNMENT

I. Personal and study details

Student's name:	**Jánsky Dušan**	Personal ID number:	**503208**
Faculty / Institute:	**Faculty of Electrical Engineering**		
Department / Institute:	**Department of Computer Graphics and Interaction**		
Study program:	**Open Informatics**		
Specialisation:	**Computer Games and Graphics**		

II. Bachelor’s thesis details

Bachelor’s thesis title in English:

Educational computer game

Bachelor’s thesis title in Czech:

Edukativní počítačová hra

Guidelines:

The goal of this thesis is to create a demo for a project focused on educating children aged 10 to 15 about internet safety through a narrative game. Collaborate with the lead game designer to design game mechanics and subsequently implement these mechanics for the game, including their integration into the game's narrative context. Additionally, communicate with the team in designing the user interfaces and then proceed with their implementation. Based on scenarios from the lead game designer, develop a game with this story featuring various paths of progression. Document the results in these chapters:

1. Problem Analysis
2. Description of Use Cases and Target Audience
3. Collaboration in Design and Implementation of User Interface
4. Technology Analysis for Implementation
5. Design and Implementation of Basic Game Mechanics for the Game
6. Design and Implementation of Game Mechanics According to the Provided Scenario
7. Game Testing with a target user group using qualitative testing method.

Bibliography / sources:

https://unity.com/
https://learn.microsoft.com/en-us/dotnet/csharp/
https://www.inklestudios.com/ink/
Buttfield-Addison Paris: Unity Game Development Cookbook: Essentials for Every Game, Oreilly Media, 2019, ISBN 9781491999158

Name and workplace of bachelor’s thesis supervisor:

Ing. Martin Klíma, Ph.D. Department of Computer Graphics and Interaction

Name and workplace of second bachelor’s thesis supervisor or consultant:

Date of bachelor’s thesis assignment: **19.02.2024** Deadline for bachelor thesis submission: **24.05.2024**

Assignment valid until: **15.02.2026**

Ing. Martin Klíma, Ph.D. Supervisor’s signature	Head of department’s signature	prof. Mgr. Petr Páta, Ph.D. Dean’s signature

III. Assignment receipt

The student acknowledges that the bachelor’s thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the bachelor’s thesis, the author must state the names of consultants and include a list of references.

.

Date of assignment receipt

Student’s signature

Bachelor Project

Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Computer Graphics and Interaction

Educational computer game

Dušan Jánsky

Supervisor: Ing. Martin Klíma, Ph.D.
May 2024

Acknowledgements

I would like to thank Ing. Martin Klíma, Ph.D. for his guidance and express my gratitude to the entire team whose contributions were essential to the final product.

Declaration

I declare that this work is all my own work and I have cited all sources I have used in the bibliography.

Prague, May 20, 2024

Prohlašuji, že jsem předloženou práci vypracoval samostatně, a že jsem uvedl veškerou použitou literaturu.

V Praze, 20. května 2024

Abstract

The main aim of this thesis is to develop game mechanics for an educational game project targeted at children aged 10-15, focusing on internet safety, awareness of scams, and cyberbullying. The game is fundamentally narrative-driven, enriched with diverse story branches, and falls within the adventure genre. The players engage as amateur detectives solving cases. The intended outcome is a playable and tested demo that highlights a gamified scenario that aligns with the educational goals of the project.

Keywords: educational games, internet safety, cyberbullying, Unity, Ink

Supervisor: Ing. Martin Klíma, Ph.D.
Human-Computer Interaction Group,
Resslova 307/9,
12000 Praha 2

Abstrakt

Hlavním cílem této bakalářské práce je vyvinout herní mechaniky pro vzdělávací herní projekt zaměřený na děti ve věku 10-15 let, se zaměřením na bezpečnost na internetu, povědomí o podvodech a kyberšikanu. Hra je zásadně řízena narativem, obohacena o rozmanité dějové větve a spadá do žánru adventura. Hráči se zapojují jako amatérští detektivové, kteří řeší případy. Zamýšleným výsledkem je hratelná a otestovaná ukázka, která zdůrazňuje herně zpracovaný scénář, který je v souladu s vzdělávacími cíli projektu.

Klíčová slova: vzdělávací hry, bezpečnost na internetu, kyberšikana, Unity, Ink

Překlad názvu: Vzdělávací počítačová hra

Contents

Figures

Tables

Chapter 1

Introduction

The aim of this thesis is the development of an educational computer game in the adventure genre. The game is designed for children aged 10 to 15 years. The main objective of the game is to educate children about internet safety in an entertaining way.

1.1 Cyberbullying and Cyber Security

The number of children who have access to the Internet from a young age is increasing. Children spend increasingly more time online.[13]

Cyberbullying impacts numerous young people every day. It involves the use of digital technologies, such as smartphones and the internet, to intimidate or harass others. There are various forms of cyberbullying. Different forms[1]:

- Sending harmful or threatening messages via emails or to someone's mobile phone.
- Spreading false information about someone online or through text messages.
- Posting malicious or intimidating content on social media platforms or websites.
- Illegally accessing someone's account to post harmful content.
- Impersonating someone online to cause harm.
- Distributing unflattering images of someone via digital means.
- Engaging in "sexting" by sharing sexually suggestive content about people without their consent.

Many studies have linked cyberbullying with depression. The researches have also found correlations between self-harm and suicidal tendencies as consequences of cyberbullying[32].

Cybersecurity of children is a topic that goes hand in hand with cyberbullying. Potential consequences of the bad awareness[20]:

- **Identity theft**: Illegal acquisition and use of someone's private information.
- **Cyberstalking**: The use of the internet or other electronic means to harass or stalk an individual.
- **Spam receiving**: Unwanted receipt of large volumes of unsolicited emails, often for advertising.
- **Computer virus**: Malicious software designed to spread to other computers and interfere with computer operation.
- **Financial scam**: Deceptive practice resulting in financial gain for the scammer at the expense of the victim.

1.2 Aim of the Thesis

Research indicates that the safety of children on the internet is a significant issue, which has already garnered attention from numerous initiatives and projects aimed at educating this age group.

To educate the target group on cybersecurity, an adventure computer game is used as a medium.

A key element of this computer game is its story, where players typically investigate various events through conversations with characters and collecting clues. Through the narrative, players learn about the security mistakes the victims made.

Playing this game should enhance the players' caution in real life by informing them about the threats present in online space and how to avoid them.

The objective of this thesis is to implement and test such a game. Collaboration with other team members is essential to transform the expertise of specialists in various fields into a concise game. The game, resulting from this thesis, could serve as a foundational building block for a larger project aiming to develop a full-scale game.

1.3 Team Structure

The development of the game required a collaborative effort of experts across various fields. The team was composed of individuals fulfilling the following roles:

1. **Martin Vaňo**
   - **Domain:** game design, technical design, script writing.
   - **Role:** leader game and technical design, gamification of online threats, creation of scenarios and dialogues.

2. **Dušan Jánsky (author of the thesis)**
   - **Domain:** game development.
   - **Role:** development of the game, selection of character animations, sound design, UI, testing, and game design support.
3. **Nela Vlasáková**
   - **Domain:** graphic designer,
   - **Role:** design and creation of in-game characters (including rigging), and 2D illustrations.
4. **Jakub Urban**
   - **Domain:** art direction, graphic design.
   - **Role:** direction of the overall graphic design, creation of map assets, and design of the maps.
5. **Others**
   - **Domains:** cybersecurity, clinical psychology, business.
   - **Roles:** Supplier of different cybersecurity and cyberbullying scenarios that resonate the most with the educational purpose of the game. Team coordination ensures that the game has a direction aligned with the company's business plan.

1.3.1 Acknowledgement

It is worth noting that the author was not responsible for the game's graphic design. This individual's expertise in graphic design significantly contributed to the visual aesthetics of the game and its overall artistic direction.

The screenwriting and dialogue aspects of the game, referenced in this thesis, were crafted by a dedicated team member.

The author's focus was on aspects of game mechanics, programming, and overarching narrative architecture. The discussions of screenwriting and dialogue in the thesis relate to their synergy with these technical and story elements.

1.3.2 Development Process

In Figure 1.1, you can see a diagram depicting how the outputs of team members were sequentially interconnected. It also shows the order in which different parts of the development were arranged. The purpose of this chart is not to detail every aspect of development.

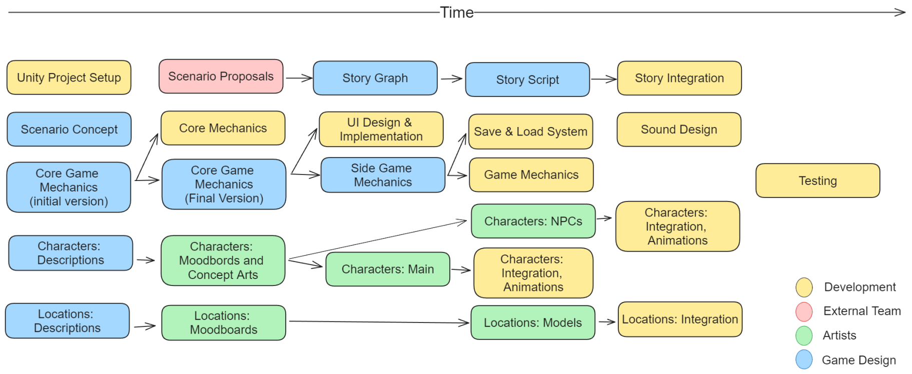


Figure 1.1: Development Diagram

Chapter 2
Analysis

The analysis chapter outlines the theoretical background of the project and describes the various requirements derived from these considerations, which significantly influence the design and specific implementation. Additionally, it explores various design options for individual mechanisms.

A critical factor is ensuring that the educational format is accessible and appealing to the target audience through the chosen media. The computer game (as a media) perfectly meets the specifications as it is widely consumed by a large portion of the target audience.[21]

The choice of themes the game highlights was managed by a team segment focused on cybersecurity, clinical psychology, and business. For the detective case discussed in this work, the issue is the security of public WiFi networks. The following practices that could lead to the compromise of online accounts are also mentioned:

- Sharing passwords with others is a bad practice that can lead to the misuse of access credentials.
- The use of antivirus software and its up-to-date maintenance.
- The importance of a strong password is highlighted, and what constitutes a strong password is explained.

In the future, the game's narrative could be expanded to include many more topics.

2.1 Target Audience

The target audience is children aged 10 to 15 years. The game is designed as an educational tool for schools.

A suitable specific medium for communication is the use of teen-friendly language and environment.

According to source[8], the educational game design recommendations are:

1. **True gamification:** The game should not just be an educational system disguised as a game but it should be a complete well-designed game that is enjoyable for the audience.

2. **Include the ability to make informed decisions:** Players should feel that their choices matter. Allow players to have some degree of freedom.

3. **Difficulty:** The game should feel challenging yet solvable.

4. **An opportunity to learn through feedback:** Players often appreciate failing in the game if they receive relevant feedback. In these situations, players feel that they failed, but they would be successful if they were in the same situation the second time. This aspect can be highly motivating for the player.

Specific recommendations for the target group are:

1. **In-game environment:** Creation of an environment that is appealing to the target group.

2. **Non-linear problems:** Should allow freedom in problem-solving. Opportunity for exploration and player choice.

Beyond these and other unmentioned specifics of game design that are somewhat more specific to the target group, general game design principles apply.

In addition to the mechanics mentioned earlier, the game designed for this target audience has the potential to educate children in areas beyond cybersecurity and cyberbullying. These areas include the recognition and management of emotions, social awareness, relationships, responsible decision-making, and several other disciplines.[8]

2.2 Dialogues

The story diagram was processed by the lead designer into dialogues (a plain document file).

2.3 Functional Requirements

Functional requirements were sorted by the MoSCoW Method based on priority. Most of the requirements were derived from the Game Design Document (GDD) and communication with the team.

2.3.1 Must Have

- **FR01 Controllable character:** Players must be able to move the character around the scene with defined controls.

- **FR02 Camera position and movement:** Camera position must reflect the active player position.

- **FR03 Multiple playable characters:** Players always play only one character at a time, but they can switch to a different one during gameplay.
- **FR04 Dialogue system - integration:** The game must contain dialogues written by the scriptwriter.
- **FR05 Story structure design:** Story must be converted from a document plain text format to a structured format in a way that structured format is readable for the technical environment and humans.
- **FR06 Story integration - branching:** The game must be able to make decisions based on the current game state (aligned with the scenario provided by the scriptwriter).
- **FR07 Story integration - consequences:** The game must be able to make consequences based on choices of the player (aligned with the scenario provided by the scriptwriter).
- **FR08 Dialogue system - story syncing:** Communication must be able to display appropriate dialogues and choices from the dialogue system that correspond to the current state of the story.
- **FR09 Dialogue system - group chat:** Main characters must be able to communicate with each other without an outer actor (typically an NPC). This must be part of the Dialogue System with all of its capabilities.
- **FR10 Group chat - triggers:** Group chat must be able to trigger by the characters entering a certain zone in the scene.
- **FR11 Interactable object - door:** Players must be able to travel between scenes by interacting with an interactive object (door).
- **FR12 Main menu:** Game must contain a main menu. Players must be able to play and exit the game from the main menu.
- **FR13 Pause menu:** Game must contain pause menu. This pause menu must allow players to resume the game and exit.

2.3.2 Should Have

- **FR14 Dialogue system UX - speaker distinction:** The dialogue system should visualy distinct different speakers.
- **FR15 Camera zoom:** Camera zoom should be appropriate so players can easily navigate and orient themselves in the scene.
- **FR16 Following system:** Unselected characters should follow the character that is currently controlled by the player. The following mechanic should feel as natural as possible.

- **FR17 Dialogue system - interactable objects:** It should be possible to enter the dialogue system by interaction with an object.
- **FR18 Characters animations:** NPCs and main characters should have idle animation.
- **FR19 Dialogue system - conditional choices:** Some story choices should be possible only if some story choices in the past were made.
- **FR20 Extra communication from an NPC to the player:** An NPC should have a way to communicate with the player (not the dialogue system). This is only one-way communication (an NPC to the player).
- **FR21 Save system:** Players should have the ability to save the game progress.
- **FR22 Load system:** Players should be able to load the story progression of the game (that was saved).
- **FR23 Interactable items - distinctness:** Interactable items should be distinct from other objects in the scene. The level of distinctness should be based on the scenario use-case, e.g.: a door should be more distinct than a hidden clue in the level.
- **FR24 Spawn:** Main characters should append at a location in the scene that is defined by the level designer.
- **FR25 Pause menu - time:** Game time should stop when a player is in the pause menu. Time should continue when the player chooses to resume the game.
- **FR26 Sound - character actions:** Character should have sounds connected to actions.
- **FR27 Sound - UI actions:** UI should have reactive sounds.

2.3.3 Could Have

- **FR28 Camera zoom - reactivity:** Camera zoom could react to various situations.
- **FR29 Door - conditions:** The player's ability to switch location could be based on the state of the story. Sometimes do not allow players to change the location, because they did not progress enough in the current location.
- **FR30 NPCs reactive animations**: NPCs could have extra animations to feel more reactive. NPCs can react to emotions (story) or player interactions.
- **FR31 Dialogue system - clickable text clues**: By clicking on the text in the dialogue system player could be able to collect clues.

- **FR32 Dialogue system UX - Fluent Progression:** The dialogue system could have techniques to catch players' attention, like text appending animation, sound effects, and others.
- **FR33 Interactable objects - scene environment:** The scenes provided by the artist could contain custom objects that make the environment more reactive. These elements could make players curious, so they want to explore.
- **FR34 Main menu - credits:** The main menu could have credits where there is info about the authors.
- **FR35 Main menu - settings:** The main menu could have settings (sound, resolution, and others).
- **FR36 Main character - emotions:** Main characters could have emotions based on the player's gameplay. For example, boredom when nothing is happening or excitement in the opposite case. These emotions could be visualized in various ways including animations, sound effects, character's actions, and so on.
- **FR37 Reactive environment:** The environment in the game could be reactive (contain reactive objects), to enhance immersion and need for exploration.
- **FR38 Dialogue system - emotions:** During dialogues characters could express their emotions to enhance the game experience.
- **FR39 City map (UI):** Players choose where they are heading (on interaction with the map).
- **FR40 Interactable objects - signpost:** Connected to **FR39**. After interaction with a Signpost player choose where to go.
- **FR41 Sound - scene-based background music:** The game could have background music or sounds based on the location.

2.3.4 Won't Have

1. **FR42 Ethical compass**: Complex value-based mechanism that tracks ethical decisions in the background and dynamically implicates the outcome.
2. **FR43 Social connections:** A communication can make connections between fractions (e.g.: police or street kids).
3. **FR44 Time management:** Characters can be assigned to a task. From that point, they are unavailable until the task is finished (time period).
4. **FR45 Report system:** A web app that allows easy validation state of the game for each student.

5. **FR46 Cut scenes:** Short films or scripted camera and character movement, 2D illustrations, animations and others.

6. **FR47 Dialogue system - visualizations:** Sequences of images or short videos during dialogues (technical topics).

2.4 Non-functional Requirements

Some of the non-function requirements are aimed at the future development of the project. The outcome of this thesis should be designed concerning these future intentions.

- **NFR1 Isometric 2.5D environment**

- **NFR2 Platform - Windows:** The game needs to be playable on Windows 10.

- **NFR3 Platform (future) - Android platform:** Technology should allow seamless development for the Android platform on top. However, it is not a requirement to develop the solution as a part of this thesis.

- **NFR4 Controls - mouse concentric:** Aligned with **NFR3**. Mouse concentric controls are typical for the adventure genre and they are also easy to convert to touchscreen devices.

- **NFR5 Responsiveness - frame-rate 30 FPS on entry-Level hardware:** See the table 2.1).

Hardware	**Minimal or target specification**
System	64-bit operating system
Processor	Intel Core i7-7500U
Graphic Card	Intel HD Graphics 620
Storage requirements	10 GB
RAM	8 GB

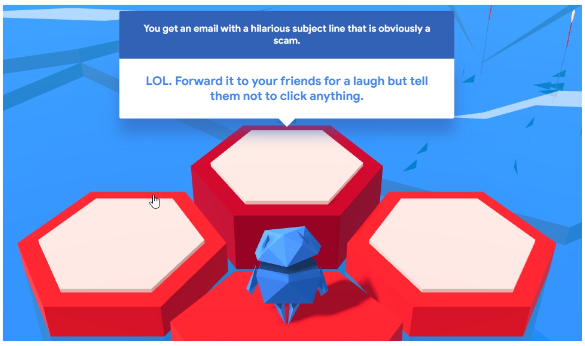
Table 2.1: Minimal hardware specifications based on Steam Hardware Survey[23]

2.5 State of the art

In this section, there are listed video games that focus on cybersecurity education (see the table 2.2).

Name	Genre	Target Audience	Educational Elements	Gameplay, Notes
Digital Compass[19]	Adventure.	Age 6 to 8.	Moral choices: piracy, social dilemma, etc.	Dense cutscenes with choices.
Cyber Legends[5]	Action adventure.	Children	Cyber-security topics.	Action top-down: dialogues and combat.
Interland: Kind Kingdom[14]	2d platformer.	Children.	Manners on social media.	Platforming and simple mini-game.
Interland: Reality River[14]	Adventure.	Children.	Fake news, misinformation.	Choice based with text explanations.
Interland: Mindful Mountain[14]	Logic, puzzle.	Children.	Sharing of personal information.	Laser puzzles.
Interland: Tower of Treasure[14]	Tower defense.	Children.	Strong passwords.	Choice based.
Trust No One[10]	Adventure.	Adults.	No, only topic.	First person, mouse-focused.
Nova Cybersecurity Lab[11]	Logic.	Older children.	Company vulnerabilities.	Harder complexity and difficulty.
BrainPop: The Meaning of Beep[2]	Minigame	Age 8 and older	Internet safety.	Missing words filling.

Table 2.2: Comparison of Similar Games


You get an email with a hilarious subject line that is obviously a scam.

LOL. Forward it to your friends for a laugh but tell them not to click anything.

Figure 2.1: Interland[14]: in-game screenshot

Other projects and details:

1. **Interland**: Interland is an interactive, web-based game developed by Google as part of its "Be Internet Awesome" campaign. The game is designed to teach children about internet safety and digital citizenship through engaging gameplay. The game can be played in the internet

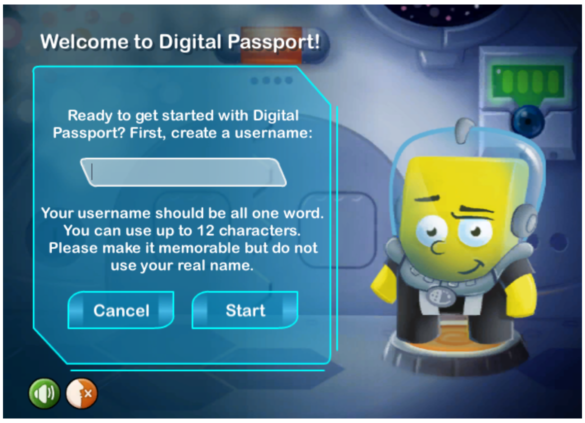
Figure 2.2: Cyber Legends[5]: in-game screenshot

browser. It consists of 4 mini-games that are listed in table 2.2 individually.[14]

2. **Cyber Legends**: On the top of the game mentioned in the table 2.2, Cyber Legends also do other activities besides the top-down adventure game. Cyber Legends have a blog about various topics and they also do short animations for kids.[5]

3. **Digital Passport by Common Sense Media**: This platform offers several games teaching the importance of making good online decisions. Games like Share Jumper and Evolve provide scenarios about information sharing and understanding cyberbullying.[19]

   For example, a mini-game called Password Protect will guide players on how to make strong passwords. Children start by typing their password suggestion and the game gives them advice that make the password stronger until the desired strength of the password is reached (see the figure 2.3)



Welcome to Digital Passport!

Ready to get started with Digital Passport? First, create a username:

Your username should be all one word. You can use up to 12 characters. Please make it memorable but do not use your real name.

Figure 2.3: Digital Passport: in-game screenshot (a minigame called Password Protect)

To summarize the table 2.2 in more detail, the main differences are:

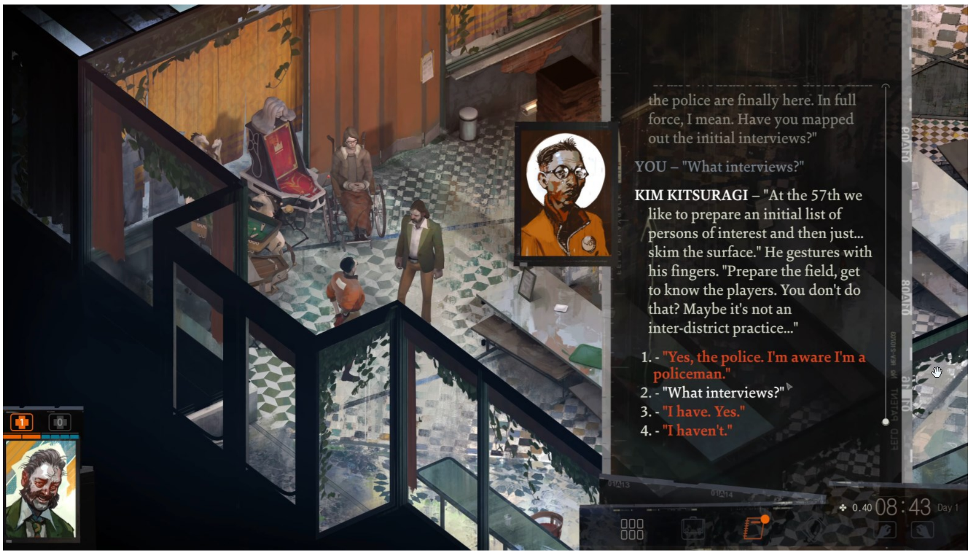
1. **Environment:** The game implemented in this thesis is set in the near future real world. The characters are in a similar age group as the target audience. This fact may impact the immersive experience of the players and provide greater transitions of the mentioned scenarios.

   The story is concentrated on the target group. Problems that characters face in the games can happen in real life and children can feel it from both game scenes and narrative.

2. **Genre:** Games mentioned 2.2 do not implement the same genre as the game from this thesis.

To summarize the games, they have done a great job with an education of children in a wide range of cybersecurity areas. None of the projects matches the narrative-driven genre directly. In the further section, there are games with similar genres, designs, and core game mechanics.

1. **Disco Elysium:** Known for its deep narrative and character development, this game offers a unique blend of detective work and role-playing elements. Unique aspect of the game is that it features a highly detailed dialogue system that significantly influences the progression of the story. This game was the main inspiration for the lead game designer. Disco Elysium has many similar game mechanics.[6]



the police are finally here. In full force, I mean. Have you mapped out the initial interviews?"

YOU – "What interviews?"

KIM KITSURAGI – "At the 57th we like to prepare an initial list of persons of interest and then just... skim the surface." He gestures with his fingers. "Prepare the field, get to know the players. You don't do that? Maybe it's not an inter-district practice..."

1. - "Yes, the police. I'm aware I'm a policeman."
2. - "What interviews?"
3. - "I have. Yes."
4. - "I haven't."

Figure 2.4: Disco Elysium: in-game screenshot

2. **Detroit: Become Human:** This game has a branching narrative structure and multiple endings based on player choices (see the figure 2.6). The game allows players to explore various ethical dilemmas and moral choices. [7]

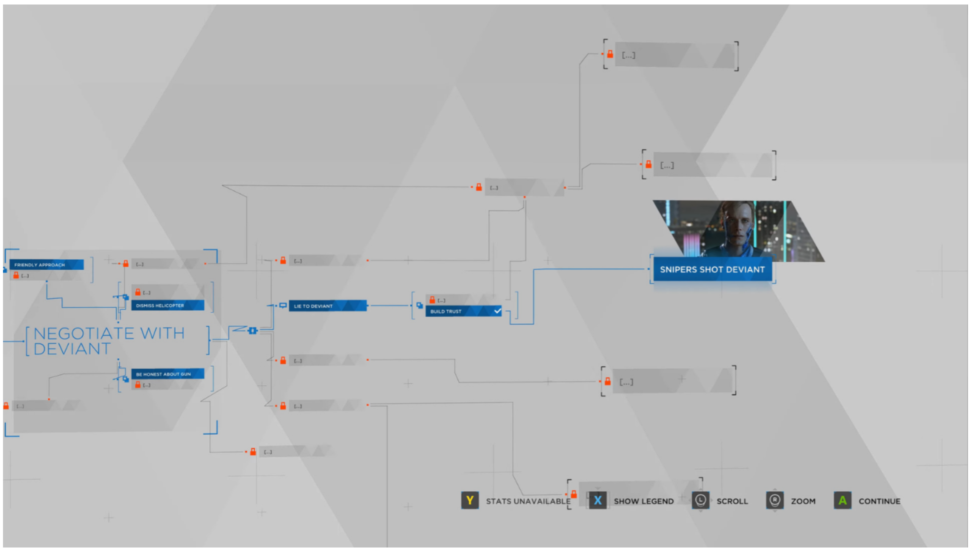
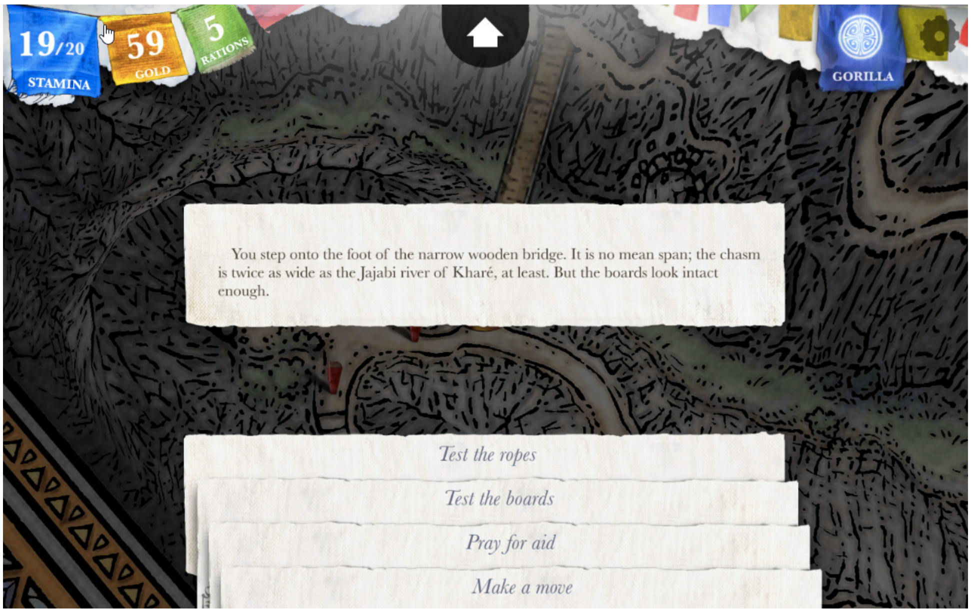


Figure 2.5: Detroit: Become Human: story branches

3. **Life is Strange** - A unique mechanic of the game is that it incorporates a time-travel mechanic, allowing players to rewind and alter past actions. Life is Strange is known for its emotional storytelling and the impact of player decisions on the narrative. Time rewind game mechanics is a unique way that allows to explore different story paths by players.[18]

4. **Sorcery!** - Unique role-playing game with great freedom. The game world is designed in a way that possible scenarios are emerging at runtime. Authors are open about how the game is designed: game design, technical design, utilized technology etc.[22]

Listed games have inspiring mechanics that can translate to our project.



You step onto the foot of the narrow wooden bridge. It is no mean span; the chasm is twice as wide as the Jajabi river of Kharé, at least. But the boards look intact enough.

Figure 2.6: Sorcery! Part 4: in-game screenshot

2.6 Choice of Game Engine

The choice of the game engine is important for the project. Valid factors to consider are team experience, project scale, and costs. Project requirements on engine environment:

- **Project Size**: The game has a small to medium size. Throughout the creation of this work, only one developer (the author of the thesis) worked on the project.
- **Team Experience**: The team has limited experience with game engines. The author of this thesis had some experience with Unity.
- **Features**: Regarding the list of requirements, all popular engines in the overview table 2.3 provide sufficient tools.
- **Story**: From the perspective of story development, it can be an advantage to have a combination of a story creation tool and an engine that has seamless integration.
- **Target Platform Support:** defined as *NFR2* and *NFR3*.
- **Pricing:** All engines are free when no revenue is generated (that is the case during thesis development). However, pricing needs to be considered in the future.

In table 2.3, you can find a brief comparison of the most frequently used game engines that were considered for development [4].

To develop a custom game engine from scratch is time-consuming and expensive.

Engine	Multiple platforms	Learning Curve	Industry standard	Royalty fee (per year)
Unity[28]	Yes, straightforward.	Fast, strong community.	Yes	Free or 1-10% (paid per seat)
Unreal Engine[9]	Yes, straightforward.	Steep, powerful features.	Yes	Free or 5%
Cry Engine[12]	Yes, some limitations.	Moderate, niche community.	No	Free or 5%
Godot[17]	Yes, very accessible.	Moderate, growing community.	No	Free

Table 2.3: Comparison of Popular Game Engines

2.6.1 Pricing of Game Engines in Detail

See the tables 2.4, 2.6, 2.7 and 2.5 for an individual engine pricing details.

In the table 2.3 I estimated the approximate year fee for the Unity engine. The pricing system for the Unity engine is more complex than for other engines.

Tier	Pricing (per year)
Under 1M USD revenue	Free
Epic Store revenue	No royalty fee
Above 1M USD revenue	Royalty fee 5%

Table 2.4: Unreal Engine Licensing and Pricing

Tier	Pricing (per year)
Universal	Free

Table 2.5: Godot Engine Licensing

Tier	Pricing (per year)
First 5K USD/game/year	Free
Otherwise	Royalty fee 5%

Table 2.6: CryEngine Licensing and Pricing

Tier	Pricing (per year)
Under 200K USD/year	Free
From 200K USD to 1M USD	2,040 USD per seat
More than 1M USD revenue	4,950 USD per seat

Table 2.7: Unity Engine Licensing and Pricing

2.7 User Interface

The options for the user interface are related to the choice of the game engine, which is described in the design chapter. Unity was chosen as the game engine.

Available systems for creating the user interface:

1. **UI Toolkit:** The most recent and recommended UI system for most use-cases. Supports (are heavily inspired) by standard web formats such as HTML, XAML, and CSS. For many developers, this may be the standard way. It is still missing some features (e.g.: World-space (3D) rendering).

2. **The Unity UI (uGUI) package:** Older GameObject-based UI system still heavily used. Uses components and the Game view to arrange, position, and style the user interface. Has a complete set of features. Can be used to customize or build editor UI.

3. **Immediate Mode Graphical User Interface (IMGUI):** Dedicated for building custom editor UI.

When choosing a UI creation system in this case, it is important to ensure that the system is capable of implementing all the necessary features. If it conforms to industry standards, that is a significant advantage, as skills are transferable among many projects, thus making it easier to find workers to maintain and expand the project in the future.

2.8 Tools for Narrative-driven Scenarios

Since our game is narrative-driven, the choice of the right tool that allows scenarios to branch and merge is crucial.

1. **Ink**: Ink is an open-source scripting language. The advantage is that it has dedicated integration into Unity. Many games were created with Ink including Sorcery! and Heaven's Vault[16].

2. **Twine**: Twine is an open-source tool for telling interactive, nonlinear stories. It is graph-based (see image 2.7). Integration into game engines is not straightforward[29].

3. **Custom tool from scratch**: This way the tool can be completely custom to cover all needs of the project. On the other hand, to build a tool comparable to Ink or Twine including a user-friendly interface for the writer can cost a lot of development time.

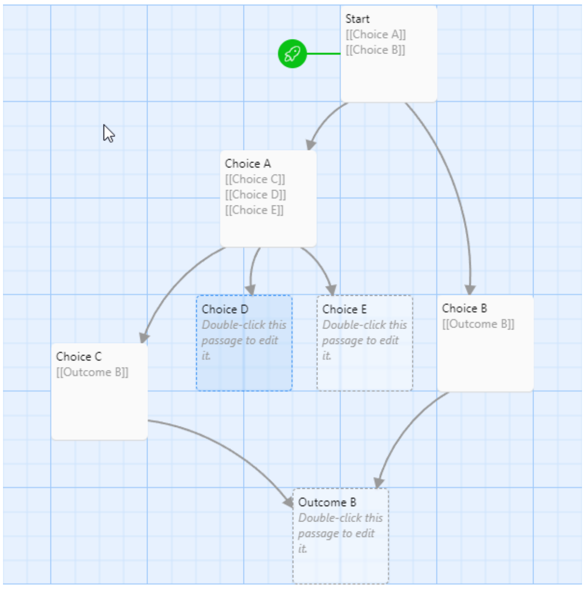


Figure 2.7: Twine: structure showcase

2.8.1 Game Progression

We need to set different conditions based on the state of the story. For instance, what conversation should the player have with NPC A? During the story, the player might have various conversations with this NPC based on the state of the story. The system needs to identify which conversation to choose at the current state of the story. The selection of the conversation could depend on various conditions.

To set the correct conversation, we must record the state of the story. The question is, what is the most effective way to record this state? Some alternatives are listed below:

1. **The story playthrough record:** We track the player's entire journey through the story. The advantage is that the tools we use typically remember this by default. The downside is that we may need to check many conditions to determine where we are in the story.

2. **The linked list with milestones:** We can simplify the story representation to a list of story milestones. We can always change the state upon reaching a significant milestone in the story. This system is simple yet very effective in capturing the state of the story.

3. **Multiple linked lists with milestones:** The last option is the most complex. We can use multiple linked lists with milestones. This achieves a more concrete representation of the world and removes linearity from the story. The disadvantage of this approach is that it can be too complicated.

The most complex and sophisticated system (option number 3) was presented at the Game Developer Conference (GDC) by the creators of Ink, which was used, for example, in the game Sorcery![22], and other games. This system allows for a detailed determination of the story's state and concludes it. Moreover, such a system is sustainable in the long term, even for very expansive worlds. Authors believe that this system is one of the keys to their ability to create game worlds that feel dynamic and responsible.[15]

The approach was chosen by the story provided by the lead game designer.

Chapter 3

Design

3.1 Story

The story is structured into detective cases (part of this thesis is one detective case only). Players move between locations. Players progress by collecting clues (dialogues with NPCs, interactions with objects).

Players have the opportunity to choose. Players can face moral dilemmas and logical choices.

An important aspect of this narrative-driven design is that the story can be branched. The player's choices can directly impact the story in different ways.

Cases that need to be resolved are structured into phases. The last phase of each case is the conclusion and outcome of the case.

In the future extension of the game: As the game progresses, more cases are unlocked.

3.1.1 Scenario

This is a shortened, paraphrased version of the full scenario. The full scenario (Czech language) is in the appendix F.

Schoolmates enjoy spending time after school, visiting a café in the city center. They decide to work on a school project there, using the internet and thus connecting to the public Wi-Fi. During work, they take short breaks to browse the internet, check social networks, log into the school system, and shop for cosmetics online. The afternoon passes uneventfully, but in the evening, one discovers she cannot access her social media because her password doesn't work. They find all their accounts inaccessible. The next day, the school principal confronts them with vulgar messages sent from their accounts, threatening expulsion. They are distraught and clueless.

The main characters, upon investigating, learn the girls were working at the café and nowhere else. They discover the café's public Wi-Fi is unprotected and likely monitored. By connecting and using a network monitoring tool, they confirm the network traffic could be observed, and encrypted messages are unreadable, but not the unencrypted ones. They deduce that the attacker, who was also present at the café, intercepted the girls' logins for certain unencrypted sites. The attacker has stolen and changed their passwords.

Returning to the café to identify the attacker, they notice a security camera. Ethically obtaining the footage from the staff, they identified a suspect, a classmate who had shown interest in the girls. Confronting him leads to a confession and rectification of his actions, including disclosing the changed passwords and admitting his deeds to the principal.

The scenario was converted to diagram B (by the lead game designer) and additional requirements were derived.

3.2 Unity Engine

Unity Engine is the engine of choice. Unity has all the features to make the game. Unity has seamless multiplatform support, which is great for the future (in the future development of the game it is planned to include Android devices).

Unity is a popular engine for indie game developers and small studios. When developers use Unity Engine, they can get things done quickly. At the same time, Unity allows developers to customize specific things. The advantage of unity is community. The community of Unity Engine is strong and old, so it can be easy to resolve incoming issues and find valuable resources on various topics.

In the future pricing will depend on the expected revenue. The project will likely reach the 200k USD revenue per year borderline. If the revenue per year will be below 200k, there is no royalty fee.

Some of the possible future scenarios for different revenue, team size, and estimated price paid per year are introduced in the list below.

- **5 Seats and revenue in range 200k - 1M USD per year** (Unity Pro): 10,200 /yr, comparable with 1 - 5% royalty fee (range is based on revenue).
- **10 Seats and revenue in range 200k - 1M USD per year** (Unity Pro): 20,400 USD /yr, comparable with 2 - 10% royalty fee (range is based on revenue).
- **5 Seats and revenue above 1M USD per year** (Unity Enterprise): 24,750 USD /yr, comparable with 2,5% royalty fee and less (based on revenue).
- **10 Seats and revenue above 1M USD per year**(Unity Enterprise): 49,500 USD /yr, comparable with 5% royalty fee and less (based on revenue).

It is worth noting that Unity designed a new pricing system. This new pricing system can increase fees significantly.

"The Runtime Fee will only apply to games created with or upgraded to Unity 6, the next Long Term Support (LTS) version of Unity to be released in 2024, and any future associated betas, Tech Streams, or LTS releases."[26]

The expected pricing is from 2.5% to 5%, where the worst scenario is 10 seats in combination with 200k USD revenue. This leads to the equivalent of 10% of the revenue royalty fee. The optimal approach is to keep the team small while revenue is small (< 1M USD) since the pricing is seat-based. This way, the royalty fee won't be higher than the usual pricing (around 5% of the revenue).

3.2.1 The Unity UI

The table 3.1 contains some of the features needed for our project.[27] Some of the listed features are not required at the moment, however they align with the long-term requirements of the project (Integration with Animation Clips and Timeline: for possible cutscenes that would involve UI components).

Feature	**UI Toolkit**	**Unity UI**
Adaptive layout	Yes	Yes
Sprites support	Yes	Yes
Screen-space (2D) rendering	Yes	Yes
World-space (3D) rendering	No	Yes
Custom materials and shaders	No	Yes
Integration with Animation Clips and Timeline	No	Yes

Table 3.1: Unity Engine Licensing and Pricing

We chose Unity UI because the UI Toolkit lacks a few features even though the UI Toolkit lacks the recommended standard.

3.3 Ink

The Ink was the chosen tool for narrative-driven scenarios because it seamlessly integrates with the Unity engine.[16]

Ink is text-based (compared to the graph-based structure of Twine, see figure [29]), as it may seem like a disadvantage at first, but it is an advantage from a certain scale for the writer.[15] The lead game designer also preferred the text structure. Ink is also an industry standard: many games were created with Ink.

3.3.1 Ink Key Features

Ink is organized into knots. Knots are similar to nodes in a graph. Pieces of scenario logic are decoupled into knots. Ink and Unity code can invoke a jump between these knots.

Ink has many useful features for writers and also for developers. This combination creates a powerful arsenal. The most important features and strong points are listed below.[16]

1. **Choices**: Key mechanics for branching and merging the story.
2. **Variables and Lists**: Programming like data structures. The variable in combination with list logic can act as an enumeration of the story stage.
3. **Conditional Choices**: Extension to pure choice functionality is that certain choices are available under specified conditions. Typically when other choices were or were not visited. Also, it can reflect the global state of the story.
4. **Knots and stitches**: Act like nodes of a story graph. A stitch is a subsection of a knot. We can invoke jumps to knots from the Unity C# logic.
5. **External Functions**: Easily invoke a function in the Unity C# codebase.
6. **Scripting nature**: Editing of Ink files is represented as text - not as graph opposite to Twine (see the figure 2.7).

See an example of conversation (with choices, knots): code and corresponding graph (see the figure 3.1).

```
=== start
+   Choice A -> outcome
+   Choice B
+ +     Choice B1 -> start
+ +     Choice B2 -> start

=== outcome
//  COMMENT: the outcome is defined here
    END
```

Listing 3.1: Ink choices example

3.3.2 Game State Representation

The game represents its main state in Ink files. We use ink variables, lists, and other elements of Ink features.

The story has different branches (see the appendix B for details), however, it can be simplified to a linked list structure with the usage of milestones because the overall structure is linear. From the game design perspective, it makes sense, because player's granular decisions are tracked by ink. It is useful to have a high overview of the state for maintainability.

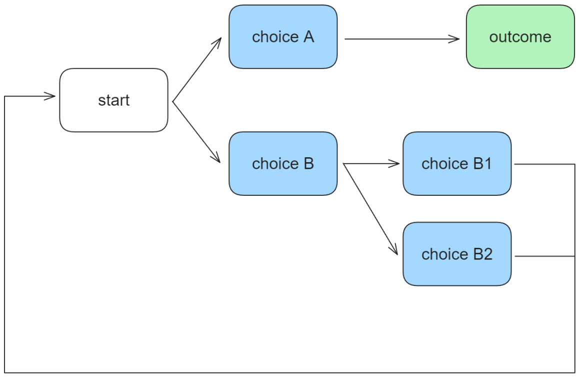


Figure 3.1: Flow example: an NPC conversation with choices, where the player seeks the outcome.

There was a question about what part of the game state we wanted to represent in Unity compared to Ink. The outcome is that the representation of the state that is needed to restore the state of the story is located in Ink scripts. A temporary game state that is unique to the session (no need to keep it) is in Unity.

3.3.3 Passive State

Since the game is not turn-based the *passiveState* knot plays a crucial role in non-linear narratives, acting as a neutral ground where gameplay continues outside of active conversations. It ensures the game world remains dynamic and interactive, even when direct dialogue or action sequences are not occurring.

Conversation is jumping between knots (each node typically an NPC) and other sections with dialogues. When the conversation ends, we can go back to this passive state knot.

In the referenced graph (see figure 3.2) there is a flow of the story. The narrative begins with the exit from a passive state, indicated by the "Conversation Enter" arrow. The conversation then progresses to one of several knots. From this point, the conversation can develop in various directions based on player choices and their consequences, as shown by the "Random Conversation Flow" arrows. Upon the conclusion of the conversation, it transitions back into the passive state, denoted by the "Conversation Exit" arrow.

3.3.4 Game Progression

Based on the size of the project the second approach from the proposed approaches has been chosen: Ink has a feature called *list*. It can be compared to enum in a programming language.

See figure 3.3: milestones are highlighted (blue color) in the scenario chart (left part of the image). These milestones were extracted to a linked list, which can be easily enumerated.

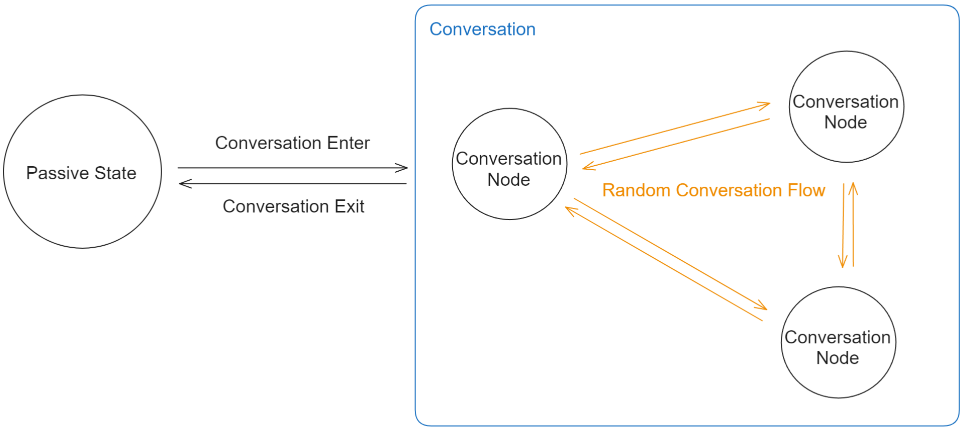


Figure 3.2: Passive state transition

Possible states can be defined, it is shown in the listing 3.2.

```
LIST possibleStates = Start, Milestone1, Milestone2, Milestone3, Solve
```

Listing 3.2: Ink list example

Now we can define the ink variable for the current state and assign it one of the *list* values (see the listing 3.3).

```
VAR caseState = Start
```

Listing 3.3: Ink variable initialization

Later we can modify the variable as the story milestones are reached (see the listing 3.4).

```
~ caseState = Milestone1
```

Listing 3.4: Ink variable modification

When conversation flows, we can check the state variable. Based on the current story progression we can choose the correct flow with the if statement (see the listing 3.5).

```
{
    - caseState == Milestone1:
        -> storyFlow1
    - caseState == Milestone2:
        -> storyFlow2
    - else:
        -> storyFlow3
}
```

Listing 3.5: Ink if statement

This is essentially how the main state of the game is tracked. Minor story conditions are tracked with ink history and boolean variables (to abstract more complicated flow checking).

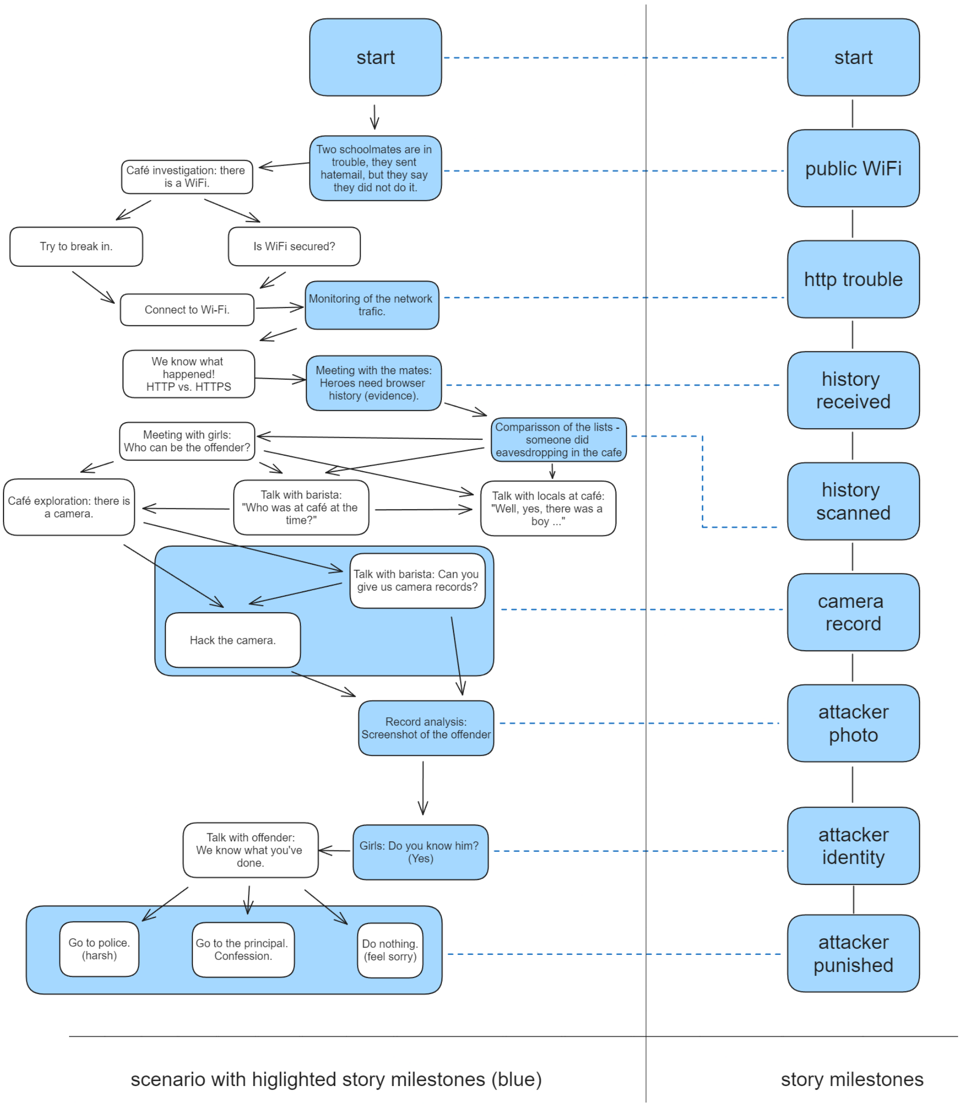

start
Two schoolmates are in trouble, they sent hatemail, but they say they did not do it.
Café investigation: there is a WiFi.
Try to break in.
Is WiFi secured?
Connect to Wi-Fi.
Monitoring of the network trafic.
We know what happened! HTTP vs. HTTPS
Meeting with the mates: Heroes need browser history (evidence).
Comparisson of the lists - someone did eavesdropping in the cafe
Meeting with girls: Who can be the offender?
Café exploration: there is a camera.
Talk with barista: "Who was at café at the time?"
Talk with locals at café: "Well, yes, there was a boy ..."
Talk with barista: Can you give us camera records?
Hack the camera.
Record analysis: Screenshot of the offender
Girls: Do you know him? (Yes)
Talk with offender: We know what you've done.
Go to police. (harsh)
Go to the principal. Confession.
Do nothing. (feel sorry)
start
public WiFi
http trouble
history received
history scanned
camera record
attacker photo
attacker identity
attacker punished
scenario with higlighted story milestones (blue)
story milestones

Figure 3.3: Scenario chart (left part) conversion to linked list (right part)

Ink history contains information about how many times the player visited the knot during the gameplay. This feature is very useful when controlling the flow of the story.

3.4 Main Characters

Main characters:

- **Jen:** introvert, coder, connected to ethical hackers, socially reserved but outspoken on IT security.
- **Kim:** influencer, artist, and DJ. More talkative than Jen.

3.4.1 Navigation of Characters

For navigation of character in the game Unity navigation system is used. The advantages are that this system is built-in and has all the features. Players control one character at a time. When players click the map, a new target is set and the character moves towards this target along an optimal path.

3.4.2 Character Switching

There are 2 main characters in the game. Players always control one character at a time. If players want to control the second character, they need to click on the unselected character. The previously selected character is deselected and the second character is selected (the one player clicked on). The camera position is smoothly adjusted according to character selection.

Selection is indicated with the blue circle under character, see the figure 3.4.

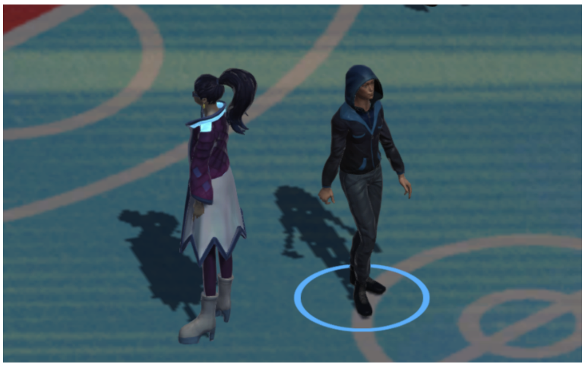


Figure 3.4: Character selection is indicated with blue circle

3.4.3 Camera

In the isometric environment of the game, the camera is set up with a fixed angle that provides a bird's-eye view of the game scene. This angle is 45 degrees to horizontal and vertical axes, ensuring a clear view of the gameplay area.

Camera follows (utilizing linear interpolation (LERP), see figure 3.5) the selected character's position.

A viewport size is set up in a way that it is easy for players to see all the details present in a scene and allow them to easy navigate in the scene at the same time.

3.4.4 Responsive Zoom

Viewport size (zoom) is adjusted based on situations:

- **Dialogues**: zoom-in during dialogues. During dialogues players it is natural to focus attention on the characters (not the overview of the scene).

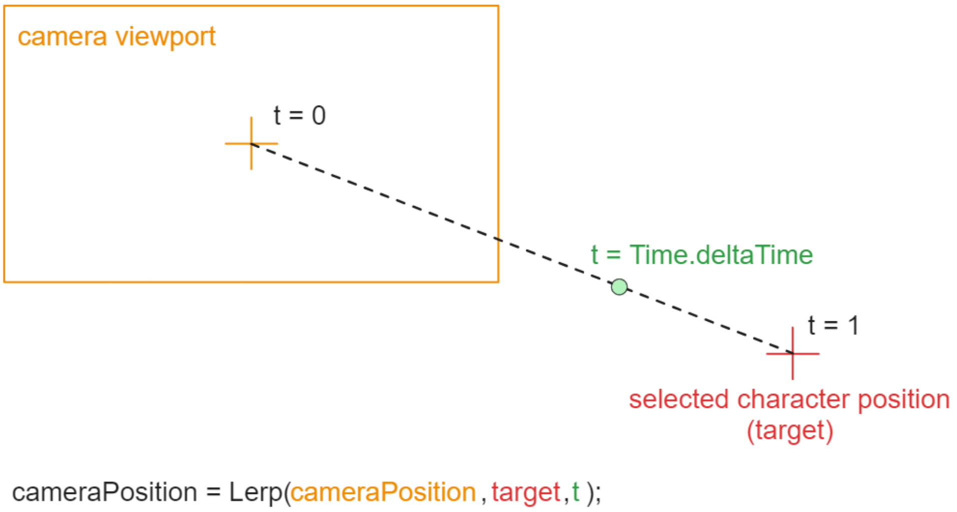

camera viewport
t = 0
t = Time.deltaTime
t = 1
selected character position
(target)
cameraPosition = Lerp(cameraPosition , target , t);

Figure 3.5: Camera LERP

- **Long distance traveling**: zoom-out when the player is moving in the scene for easier navigation.

3.4.5 Selected Character Following

An unselected characters follow the character that is currently controlled by the player. Following is triggered when distance between the selected character and the unselected character is bigger than threshold.

A random location from the circle around the player is selected. The unselected character is navigated to this location.

3.4.6 Animations: Idle and Walk

2 animations are active the most: idle animation and walk animation (see animation controller at figure 3.6). These animations are changed based on the character's state.

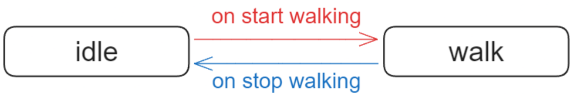


Figure 3.6: Animations diagram: idle and walk

3.4.7 Emotions

Emotions are triggered by state and time. When one emotion is reached, timers are restarted. There are 2 emotions:

- **Boredom:** When characters are not walking or talking for some time they will do unique animation, for example stretching of the neck. Both characters have multiple animations, on trigger one of them is selected randomly.

- **Aspiration:** If the main character is not selected as an active player for a long time. This can trigger action, bubble with the text that indicates that the bored character also wants to be in charge.

See the diagram of Jen's boredom emotion in the figure 3.7. One of the arrows is randomly selected when boredom is reached.

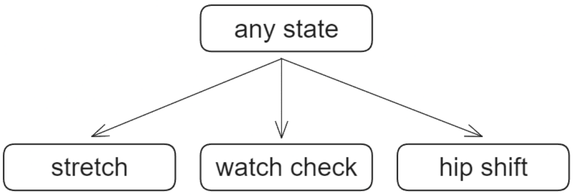


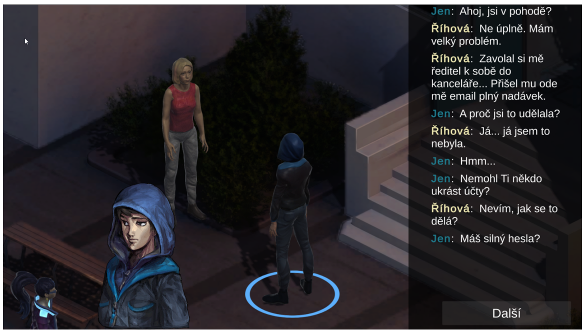
Figure 3.7: Animations diagram: Jen's boredom diagram

The main point of the emotion system is that characters feel more alive and reactive to player actions.

In the future system, extra actions and emotions can be added.

3.5 Dialogue System

The dialogue system provides an interface between the story and players. See the dialogue system window in figure 3.8.



Jen: Ahoj, jsi v pohodě?

Říhová: Ne úplně. Mám velký problém.

Říhová: Zavolal si mě ředitel k sobě do kanceláře... Přišel mu ode mě email plný nadávek.

Jen: A proč jsi to udělala?

Říhová: Já... já jsem to nebyla.

Jen: Hmm...

Jen: Nemohl Ti někdo ukrást účty?

Říhová: Nevím, jak se to dělá?

Jen: Máš silný hesla?

Figure 3.8: Dialogue: in-game screenshot

The dialogue system is connected directly to the ink representation of the story. It displays and processes corresponding dialogues, choices, etc. Notable features:

- **Character previews:** Based on the current speaker a preview image is set in the bottom down corner.

- **Text styling:** Each speaker in the dialogue has a different color for better distinction.

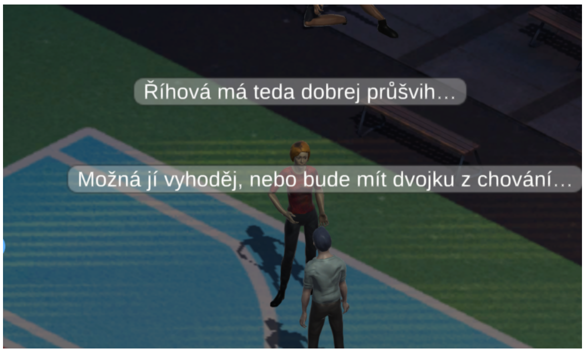
- **Emotions:** When a character in the dialogue has some emotion to be expressed it can be indicated in the ink file by tags. Tag invokes sound effects for emotion when dialogue is displayed. Multiple emotions can be triggered at the same time (e.g.: a woman laugh and a man laugh lead to two people laughing at the same time). Emotions have multiple sounds, so when the same emotion is triggered multiple times there is a variety of triggered sound effects.
- **Typing:** displaying the text is animated and the sound effect of typing is playing.
- **Skip button:** When text is appending player can hit the skip button.
- **Next button:** After players read the dialogue they have to manually trigger the next dialogue with a button.

The dialogue system can be entered by interaction with NPC, interactable objects, or stepping into the zone.

3.6 NPCs

An NPC (non-playable character) can be interactable. NPCs can emit comics-like bubbles above them. Players can also have a dialogue with NPCs through a dialogue system.

- **Comics bubble**: Bubble is typically triggered when players are nearby to catch the player's attention with brief sentences (see example at the figure 3.9). It is possible to emit bubbles on click, or mouse hover. They can be emitted in time intervals.



Říhová má teda dobrej průšvih...

Možná jí vyhoděj, nebo bude mít dvojku z chování...

Figure 3.9: Bubbles: in-game screenshot

- **Dialogue Overlay**: Interaction will trigger a dialogue between the current player's character and NPC. The dialogue system is described in the section above.

Each NPC in the game has a corresponding scriptable object. With this design NPCs are created comfortably from the editor. NPCs have assigned unique idle animations. Animation start time can be offset.

3.7 In-game Environment

An isometric environment leads to more constraints but to a unique design.

Specific to the isometric environment is that it allows designers to focus on one perspective. This simplifies navigation and interaction, put constrains on the camera's view angles, affecting storytelling and visual dynamics.

A key component of gameplay is mouse interaction. Interaction can give players information or predefined options on how to interact with an object. To point out that objects is interactable, there is an outline on mouse hover around objects that are interactable.

3.7.1 Door

Players can use the door to reach desired destinations based on story and choices. On interaction with the door, the current location is changed. Players can interact with doors by mouse.

The door has also attached hover animated effect with light and text hints to indicate the interactivity of the element.

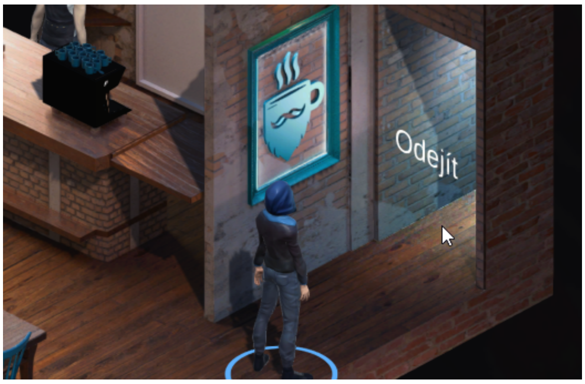


Figure 3.10: Door in-game screenshot

The door has attached a scriptable object. This object specifies what target destination is based on current state of the story. Sometimes it does not allow player to exit the scene (because player did not reached the story milestone yet). This is done by checking the current game progression ink variable.

3.7.2 Spawn

Spawn is a place where players append in the scene. Spawn is prefab that is manually placed to the scene when setting up the map. The range of the spawn zone has a predefined value, but can be customized in the editor. Main characters are randomly spawned within the set range from the spawn origin.

Random spawn position has an advantage over fixed position because it can be easily extended to spawn more characters.

Spawn up to 4 characters was a desired use case at some point during the development, but it was excluded. The ability to spawn multiple characters at random positions remained.

3.7.3 Group Chat

By triggering the group chat, dialogue is entered. The group chat game object can be triggered by the mouse or zone (character enters the zone). Trigger variants can be set from an editor.

Group chat is a prefab (easy placement, easily customized).

3.7.4 Story Interactable Objects

By interacting with these objects, players enter dialogue. These objects have typically additional unique scripts for indication of interactivity.

3.7.5 Responsive Interactable Objects

Some objects in the game are interactive, and designed to encourage player exploration. These objects are equipped with custom scripts and highlighted to indicate their interactivity through sound effects and outlines on hover, among other cues.

Examples of interactive objects include:

- **Arcade machines:** animation, random sound effect when clicked;
- **Lights:** toggle on and off with a click, including sound;
- **Particle dust effect at a laboratory window:** not interactive, but adds to the scene's liveliness.

3.8 Save & Load system

A requirement is to capture a story state. This state is captured by ink. Ink file creates JSON file at runtime.[24] This file can be used to restore the state of the story.[25]

One thing that is not tracked by the ink (by default) and is required to store. That is the player's location. The solution is to create a new variable in the ink file. This variable holds the name of the current location. When the player enters a location this variable is updated. When restoring the state of the game from a file we use the state of this variable to select the correct scene.

3.9 Animations Frameworks

Our project uses the Lean Tween library for animations. The Main Advantage of the Lean Tween library is that it significantly simplifies the animation process. It possesses all the features the project needs.

Using this library can also lead to better readability of the code base (short similar functions).

3.9.1 HUD

Character avatars at the bottom of the screen. Avatars can be clicked to select the character.

3.10 Menus UI

This section briefly describes game menus. There are 2 main menus. See how they look in the appendix D.

- **Main menu**: Players can play or load the game from the main menu. It's also possible to display information about the authors and quit the program.
- **Pause menu**: The pause menu players can be entered directly from the gameplay by pressing the escape key. Players can resume, save the progression, or go to the main menu.

Chapter 4

Implementation

- Unity Engine version: 2022.3.16f1
- Unity Ink plugin version: 1.1.8

4.1 Development Setup

Git was the required versioning solution by the contractor. It's industry standard. Graphic assets take up a lot of memory in the project, which is typical for a game development project. That is why Git Large File Storage (LFS) was added to the project.

Git LFS uses a remote server to store the files. Main advantages:

- **The same commands as Git:** Environment stays the same. Developers do not have to learn a new technology.
- **Space efficiency:** Git LFS does not make unnecessary snapshots of large files when configured correctly. This fact results in downloading and uploading the optimal amount of data.
- **Versioning:** Even large files can be versioned effectively with Git LFS.

The Git repository contains the Unity project, a git submodule[3] with an evaluator server, a README.md file, and Git configuration files. The README file has info about project structure and instructions for team members outside development who upload assets. See the structure 4.1.

```
WebOfLies/
+-- WebOfLiesUnityProject/
    +-- Assets/
        +-- Resources/
+-- GameRecordEvaluatorServer/
+-- .gitattributes
+-- .gitignore
+-- .gitmodules
+-- README.md
```

Listing 4.1: Project structure

Configuration files overview:

- **.gitattributes file**: Used for Git LFS to handle large files and binary data efficiently.
- **.gitignore file**: Specifies intentionally untracked files that Git should ignore.
- **.gitmodules file**: Manages the git submodules.

Excluded folders from the structure are not typically managed by the users directly.

The assets folder contains everything set up manually in the Unity project. Some of the examples are animations, scenes, scripts, and so on.

An important folder in the Assets folder is the Resources folder4.2. The resources folder contains mainly files that may need to be loaded at runtime4.3.

```
Resources/
+—— Prefabs/
+—— Sprites/
+—— Audio/
+—— ...
```

Listing 4.2: Resources folder structure

```
// Load Asset (e.g.: prefab of playable character)
var loadedObj = Resources.Load<GameObject>(resourePath);

// Create asset instance in the scene
GameObject instance = Instantiate(loadedObj);
```

Listing 4.3: Resource loading example

4.2 Locations Setup

The game environment is structured into locations. Many objects and NPCs placed in locations have been set up based on the story progression.

The lead designer defined the requirements for each map. The artist team then created a textured model of the locations. Part of the development was the integration of models into the Unity project and scene setup.

To set each location these main steps were processed for each environment:

1. **Import of the models.**
2. **Colliders:** Set up colliders for various objects in the scene. Balancing detailed collider geometry for natural interaction and simplified geometry for optimization is crucial.
3. **Mark static objects:** Static objects that players can collide with should be marked to be recognized by the navigation map.

4. **Walkable surfaces:** Identify and mark walkable surfaces.
5. **Bake the navigation map:** Compile a list of marked objects and walkable surfaces to bake into the navigation map. This ensures characters can navigate the scene, understanding where they can walk and what obstacles they must avoid.
6. **Door setup:** Configure the door objects within the scene.
7. **Lights:** Set up lighting to enhance the environment.
8. **NPCs:** Position NPCs according to their roles in the story.
9. **Interactable objects:** Place interactive objects as required by the narrative.
10. **Make scene alive:** Add small interactive elements with custom scripts to enhance the liveliness of the scene.

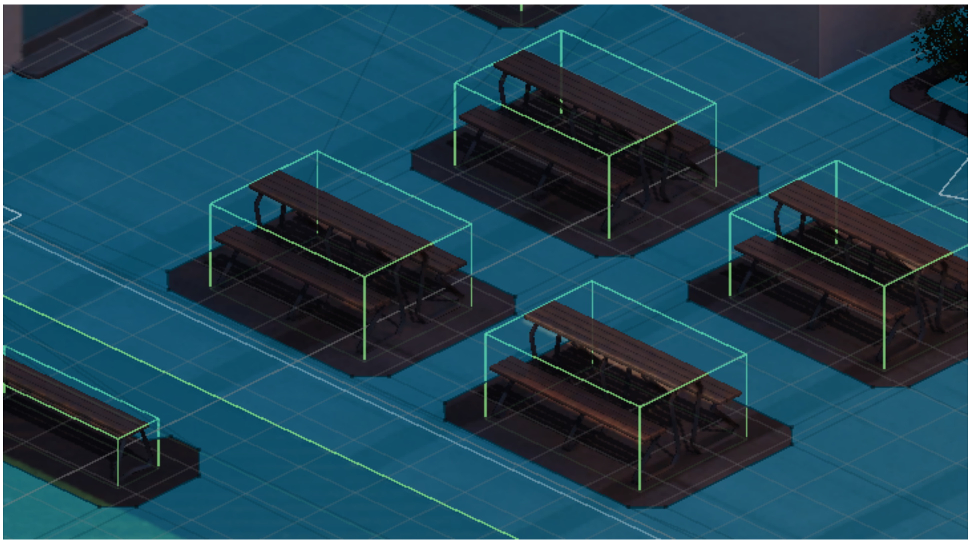


Figure 4.1: Map integration process

See the figure 4.1. You can see efficiently selected block colliders. Blue surface is a walkable surface. Benches are marked as static objects, so in the blue map, there are holes around the cube, which indicates surfaces that are not walkable.

4.3 UI Architecture

We designed an interface IView and a abstract base class ViewBase to unify common patterns and behavior in the project's UIs.

The ViewBase extends MonoBehaviour and can be used to initialize, manage, and control UI elements.

4.4 Save & Load System

Save files need to have a persistent location. Unity has a handy function for ensuring persistent data location.[30]

```
public static void Save(Json dataAsJson, string fname)
{
    string saveFilePath = System.IO.Path.Combine(
        Application.persistentDataPath,
        $"{fname}{saveFileExtension}"
    );
    System.IO.File.WriteAllText(saveFilePath, dataAsJson);
}
```

Listing 4.4: Save method

There is one thing that is not potentially in the mentioned save file. That is the current location where the player was at the save time. That is why we defined a list of locations in the ink file. We also defined the current location variable. This variable is modified when location is entered[31].

When loading the game from JSON file is used to load the appropriate scene.

4.5 Spawn

To summarize the implementation of the spawn mechanic, spawn is *Game Object* that have a custom script attached. It is a *prefab* so it can be easily used in multiple scenes.

The function of this script is that it loads up main character prefabs and *Instantiate*s them in the scene within the range of the spawn (circle).

To get random points from the circle this approach is used:

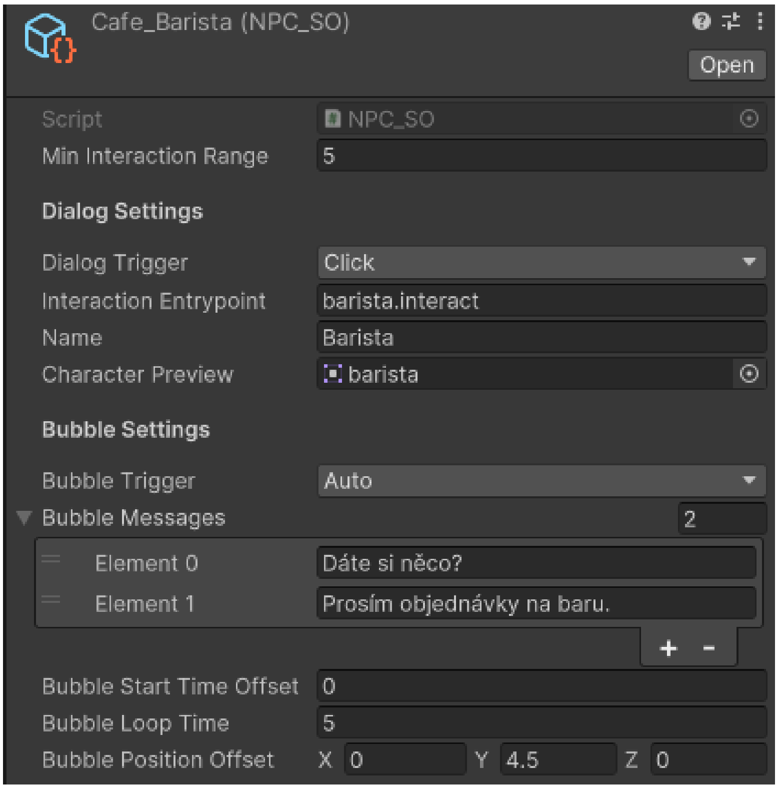
1. Get a random radius between 0 and max range set up.

2. Get a random point from the circle that has the radius generated.

4.6 NPC

There is a prefab for NPCs. This prefab has a parameter in the editor for of type NPC scriptable object. Based on the scriptable object assigned the properties of the NPC are set up. See the example of an NPC scriptable object at figure 4.2.

NPCs are defined by scriptable objects in the Resources folder 4.5.



Cafe_Barista (NPC_SO)

Script: NPC_SO

Min Interaction Range: 5

Dialog Settings

Dialog Trigger: Click

Interaction Entrypoint: barista.interact

Name: Barista

Character Preview: barista

Bubble Settings

Bubble Trigger: Auto

Bubble Messages: 2

Element 0: Dáte si něco?

Element 1: Prosím objednávky na baru.

Bubble Start Time Offset: 0

Bubble Loop Time: 5

Bubble Position Offset: X 0 Y 4.5 Z 0

Figure 4.2: NPC Scriptable Object Example

```
Resources/
+—— ScriptableObjects/
    +—— ...
    +—— NPCs/
        +—— Barista.asset
        +—— CafeCustomer1.asset
        +—— CafeCustomer2.asset
        +—— School_RandomClassmate1.asset
        +—— Schoo2_RandomClassmate1.asset
        +—— Rihova.asset
        +—— HateMailSender.asset
        +—— ...
```

Listing 4.5: NPCs folder in Resources

NPC prefab has NPCController script attached. This script handles mainly NPC's setup, reactivity, and animations.

When NPCs scriptable objects are created, we can set up their reactivity by setup up appropriate triggers that match the scenario. For example, Říhová NPC has a trigger of type Clicked (when the player uses the left mouse button when the mouse is over the NPC's collider) for dialogue and emits a bubble on hover (when the mouse position is over NPC's collider) in the game. All of these properties can be set comfortably from the Unity editor (no coding skills required). Possible trigger types:

- **None**: Action does not have a trigger (e.g.: the player can't have a dialogue with an NPC, so the dialogue trigger is None).
- **Click**: When the player clicks on the collider of the object with this trigger, a connected action is executed (e.g.: bubble emission, dialogue trigger).
- **Hover**: When the player hovers over the object's collider with this trigger, a connected action is executed.
- **Auto**: Can be set for bubble emission. Bubbles are emitted over time with custom loop time and start offset.

Chapter 5

Testing

A qualitative testing method was used to test the game. Testing had 3 stages. See the appendix C for detailed structure.

1. **Face-to-face Entry Interview:** Each tester went through a short interview. Testers were asked about their experiences with cyberbullying, cybersecurity, and computer games. Most of the questions were open-ended.

2. **Playtesting:** The gameplay of the game takes approximately 45 minutes. Testers played the entire game, structured into 8 segments. Each segment began with an initial task for the tester. During sessions, we observed the tester's actions and took detailed notes. This process was vital for identifying gameplay issues, improving user experience, and gathering insights about narrative flow, difficulty balance, and interface intuitiveness.

3. **Post-Play Interview:** After playtesting, testers went through a series of questions regarding their gameplay experience and the cybersecurity message of the game. Most of the questions were open-ended.

5.1 Hardware Requirements

The game was running on Thinkpad X1 Carbon 5th Generation (an office laptop released in 2017). The frame rate was not below 30 FPS during all sessions. The average value on this hardware setup was 45 FPS.

5.2 Testers

There were 6 testers. Testers are aged from 19 to 24. Each of them has different experience and knowledge of cybersecurity. All of the testers at least occasionally play video games. More detailed specifications of testers are attached in the appendix E.

5.3 Playtesting Highlights

All testers completed the game without serious struggles.

The playtesting tasks fluently followed one another. See the tasks in the form attached (appendix C).

It is not clear, which NPCs are interactable in the school yard (task 2). This is not a major issue, because players are not deeply involved with the gameplay yet. It is their first interaction with the game.

In task 6, some of the testers were not sure how to progress. To progress players need to search the scene at that point and find the security camera. What clues should players collect is not explicitly said and clue objects are small. It is also the first time in the game where the player has to find something a bit hidden (in other sections of the game there are interactions with NPCs and large objects that the player does not have to seek hard).

Players can achieve a goal with different actions and choices, which was the case with the testers. There were no identical playthroughs.

Some players were not paying as much attention to dialogue at the end of the game. It was clear that for some, the game had too much narrative.

5.4 Post Interview

The post interview questions are in the appendix C.

A few testers pointed out that the camera position and zoom are not great, because it is sometimes difficult to move to distant places. This can lead to unintentional interactions with the object.

A minority of the testers did not interpret the outcome message correctly. They were confused with a connection between http(s) and public WiFi. The message was clear for most of the testers.

Two testers pointed out that in the story, victim gives browser history to their classmates. They do not consider this act as a secure practice even if the victim is in trouble because this data could be misused.

Most of the testers in the past made the same mistake as the victim in the story. Most of the testers said that they will be more aware (or they will behave differently because they gained knowledge about the threat) in the future in similar situations.

5.5 Results

A high-level summary:

1. All testers completed the game with none or minimal issues.
2. All testers explored the in-game environment (without receiving specific instructions) and enjoyed its interactive elements.

3. The game's strong points included a captivating atmosphere, an interactive environment, humorous dialogues, and a sense of player choice.

4. There is a need for more diverse gameplay elements.

Areas for improvements and issues during the testing:

- **(Task 4) An issue in the story logic:** Unexpected game progression when interacting with an NPC in a specific situation. It was corrected by adjusting the dialogue conversion logic and adding default conversations to all interactable NPCs.

- **(Task 2) A distinctness of non-interactive NPCs**: Players were trying to interact with NPCs that are not interactable. NPCs could emit a bubble (on interaction) to clarify the game knows about the player's effort to interact. However, from the game's response, it would be clear that deep interaction with these NPCs is not possible.

- **(Task 6) More hints for players required:** Some testers were unaware of the need to seek a security camera (needs to be found to progress the story), which can be addressed by including hints in dialogues or visual clues. In particular, it could be possible to extend the emotion system: the second character could give hints on aspiration emotion to the players.

- **(Task 7) The text-centric nature of the gameplay:** At the end of the game a few testers were rushing through dialogues. A potential solution is to include some additional gameplay elements such as clickable clues in dialogues and minigames. The addition of dubbing could also add immersion to the dialogues, especially for the target group.

- **Balance of the responsiveness of the zoom:** Responsiveness of the zoom was not natural for a few testers. This was corrected by adjusting the source code to balance the responsiveness. However, if the issue remains the issue could be resolved by allowing players to control zoom manually.

- **Message not clear:** The message of the story was not clear for 2 testers (the connection between HTTP(S) and public WiFi technologies). This could be fixed by an alternation of dialogues and the addition of visualizations or cutscenes.

Among the spotted issues, there was only one great issue in terms of core functionality. This issue was corrected. Most of the other issues suggest that it is required to put more time into details. The remaining issues suggest future directions, improvement, and new core features.

Chapter 6
Conclusion

The thesis presents a comprehensive approach to the design and development of an educational computer game aimed at raising awareness among children aged 10-15 about the dangers of the internet, including cyberbullying and online scams. The game, designed in the adventure genre, successfully integrates narrative-driven gameplay with educational content, allowing players to engage in detective-like activities to solve cases related to internet safety.

The game development involved a multidisciplinary team, indicating a collaborative effort that combined expertise in game design, cybersecurity, and education. The implementation of the game through the Unity engine with an emphasis on interactive and engaging storytelling is a significant strength, as it ensures that players are not just passive recipients of information but active participants in learning.

From the testing phase, it appears that the game was effective in engaging users and providing educational content in an interactive manner. However, feedback suggests a need for clearer instructions and possibly more diverse gameplay elements to maintain player interest throughout.

Future development could focus on expanding the game's narrative to cover a broader range of cybersecurity topics and incorporating more dynamic interactive elements to enhance learning outcomes. Additionally, considering the diverse learning paces and styles of children, adaptive difficulty levels might be implemented to cater to a wider range of skills and to keep the game challenging yet accessible.

Overall, the thesis demonstrates the potential of using narrative-driven gameplay to educate children about internet safety effectively, laying a strong foundation for further development and research in this area.

Bibliography

[1] Admin. *Cyber Bullying Statistics - Bullying statistics*. en-US. July 2015. URL: `http://www.bullyingstatistics.org/content/cyber-bullying-statistics.html` (visited on 05/01/2024).

[2] BrainPOP. *BrainPop: The Meaning of Beep*. 2024. URL: `https://www.brainpop.com/games/themeaningofbeep/` (visited on 05/22/2024).

[3] Scott Chacon and Ben Straub. *Pro Git*. `https://git-scm.com/book/en/v2/Tools-Submodules`. Apress, 2024.

[4] Eleftheria Christopoulou and Stelios Xinogalos. "Overview and comparative analysis of game engines for desktop and mobile devices". In: (2017).

[5] CyberLegends. *Cyber Legends*. 2024. URL: `https://www.cyberlegends.com/game` (visited on 05/22/2024).

[6] *Disco Elysium*. URL: `https://discoelysium.com/` (visited on 05/22/2024).

[7] Quantic Dream. *Detroit: Become Human*. 2024. URL: `https://www.quanticdream.com/en/detroit-become-human` (visited on 05/22/2024).

[8] Kelli Dunlap and Susan Rivers. "Beyond Empathy: Games to Foster Teen's Social and Emotional Skills". In: (Jan. 2019).

[9] Inc. Epic Games. *Unreal Engine*. 2024. URL: `https://www.unrealengine.com/` (visited on 05/22/2024).

[10] Josh Feinsilber. *Trust No One*. 2020. URL: `https://blog.gimkit.com/blog/trust-no-one` (visited on 05/22/2024).

[11] WGBH Educational Foundation. *Nova Cybersecurity Lab*. 2024. URL: `https://www.pbs.org/wgbh/nova/labs/lab/cyber/` (visited on 05/22/2024).

[12] Crytek GmbH. *Cry Engine*. 2024. URL: `https://www.cryengine.com/` (visited on 09/30/2010).

[13] Donell Holloway, Lelia Green, and Sonia Livingstone. *Zero to Eight. Young Children and Their Internet Use*. Jan. 2013.

[14] Google Inc. *Interland*. URL: `https://beinternetawesome.withgoogle.com/` (visited on 05/22/2024).

[15] Jon Ingold. *Narrative Sorcery: Coherent Storytelling in an Open World.* Youtube. 2017. URL: `https://youtu.be/HZft_U4Fc-U?si=vW5f98IKuSKqqJSv`.

[16] *ink - inkle's narrative scripting language.* URL: `https://www.inklestudios.com/ink/` (visited on 05/02/2024).

[17] Ariel Manzur Juan Linietsky and contributors. *Godot Engine.* 2024. URL: `https://godotengine.org/` (visited on 05/22/2024).

[18] Square Enix Ltd. *Life is Strange.* URL: `https://lifeisstrange.square-enix-games.com/` (visited on 05/22/2024).

[19] Common Sense Media. *Digital Compass.* URL: `https://www.digitalcompass.org/` (visited on 05/22/2024).

[20] Farzana Quayyum, Daniela S. Cruzes, and Letizia Jaccheri. "Cybersecurity awareness for children: A systematic literature review". In: *International Journal of Child-Computer Interaction* 30 (2021), p. 100343. ISSN: 2212-8689. DOI: `https://doi.org/10.1016/j.ijcci.2021.100343`. URL: `https://www.sciencedirect.com/science/article/pii/S2212868921000581`.

[21] N. A. A Rahman et al. "The Importance of Cybersecurity Education in School". In: *International Journal of Information and Education Technology* (2020). URL: `https://api.semanticscholar.org/CorpusID:218824588`.

[22] *Sorcery!* URL: `https://www.inklestudios.com/sorcery/` (visited on 05/22/2024).

[23] *Steam Hardware Survey.* URL: `https://store.steampowered.com/hwsurvey` (visited on 05/01/2024).

[24] Inkle Studios. *ink JSON Runtime Format.* 2024. URL: `https://github.com/inkle/ink/blob/master/Documentation/ink_JSON_runtime_format.md` (visited on 05/01/2024).

[25] Inkle Studios. *ink JSON Runtime Format.* 2024. URL: `https://github.com/inkle/ink/blob/master/Documentation/RunningYourInk.md#saving-and-loading` (visited on 05/01/2024).

[26] Unity Technologies. *Changes to Unity plans and pricing.* 2024. URL: `https://unity.com/pricing-updates` (visited on 05/22/2024).

[27] Unity Technologies. *Comparison of UI systems in Unity.* 2024. URL: `https://docs.unity3d.com/Manual/UI-system-compare.html` (visited on 05/22/2024).

[28] Unity Technologies. *Unity Engine.* 2024. URL: `https://unity.com/` (visited on 05/22/2024).

[29] *Twine - an open-source tool for telling interactive, nonlinear stories.* URL: `https://twinery.org/` (visited on 05/02/2024).

[30] *Unity Scripting API: Application.persistentDataPath Property.* Unity Technologies. 2024. URL: `https://docs.unity3d.com/ScriptReference/Application-persistentDataPath.html` (visited on 05/01/2024).

[31] *Unity Scripting API: SceneManager.sceneLoaded Event*. Unity Technologies. 2024. URL: `https://docs.unity3d.com/ScriptReference/SceneManagement.SceneManager-sceneLoaded.html` (visited on 05/01/2024).

[32] Chengyan Zhu et al. "Cyberbullying among adolescents and children: a comprehensive review of the global situation, risk factors, and preventive measures". In: *Frontiers in public health* 9 (2021), p. 634909.

Appendix A

Shortcuts

Shortcut	**Full Version**
CS	computer science
e.g.	for example
GDC	Game Developer Conference
GDD	Game Design Document
HUD	heads-up display
LERP	Linear interpolation
LFS	Large File Storage
NPC	non-player character
UI	user interface
UX	user experience

Table A.1: List of shortcuts

Appendix B

Scenario Diagram

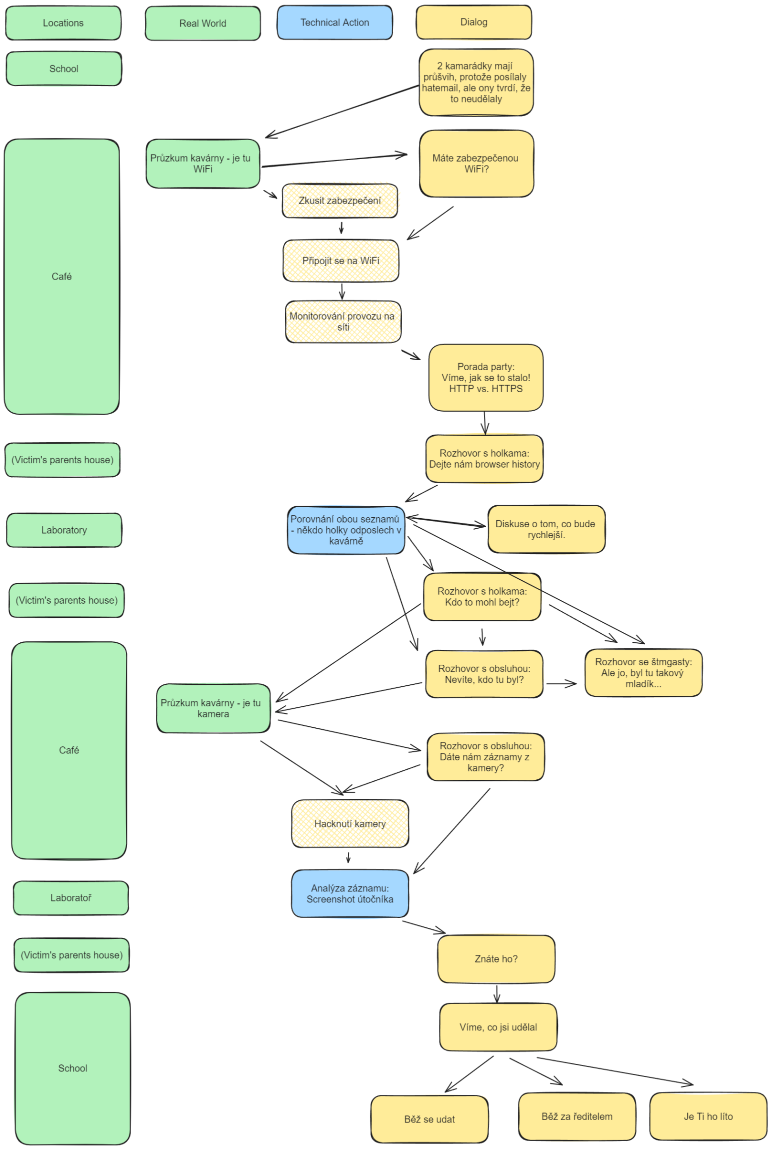

Locations
Real World
Technical Action
Dialog
School
2 kamarádky mají průšvih, protože posílaly hatemail, ale ony tvrdí, že to neudělaly
Café
Průzkum kavárny - je tu WiFi
Máte zabezpečenou WiFi?
Zkusit zabezpečení
Připojit se na WiFi
Monitorování provozu na síti
Porada party: Víme, jak se to stalo! HTTP vs. HTTPS
(Victim's parents house)
Rozhovor s holkama: Dejte nám browser history
Laboratory
Porovnání obou seznamů - někdo holky odposlech v kavárně
Diskuse o tom, co bude rychlejší.
(Victim's parents house)
Rozhovor s holkama: Kdo to mohl bejt?
Café
Průzkum kavárny - je tu kamera
Rozhovor s obsluhou: Nevíte, kdo tu byl?
Rozhovor se štmgasty: Ale jo, byl tu takový mladík...
Rozhovor s obsluhou: Dáte nám záznamy z kamery?
Hacknutí kamery
Laboratoř
Analýza záznamu: Screenshot útočníka
(Victim's parents house)
Znáte ho?
School
Víme, co jsi udělal
Běž se udat
Běž za ředitelem
Je Ti ho líto

Figure B.1: Scenario Diagram

Appendix C

Testing Process

At the beginning, there was an introduction, where testers were informed about the testing process. They received instructions for playtesting and got a short intro about the project, game story, and controls.

C.0.1 Entry Interview

The goal of the entry interview is to summarize the profile of the testers.

- **Age**
- **Gender**
- **Field of Work**
- **Knowledge of Cybersecurity:** How a tester subjectively evaluates his knowledge among the average population.
- **Experienced Cyberbullying:** If the tester personally experienced any form of cyberbullying.
- **Experienced Scam**: If the tester personally experienced any form of scam.

C.0.2 Playtesting

Full game completion is segmented into 8 tasks. These tasks are connected. Most of the time instruction was not needed. Players start in the main menu.

1. Play the game.
2. Look around the map and figure out what to do.
3. Continue to the cafe.
4. Figure out as many clues as possible about the public WiFi.
5. Continue to lab and progress with the case.
6. Investigate the cafe and collect as many clues as possible.

7. Analyse the camera record and identify the hacker.
8. Go find Vávra and confront him with the information.

C.0.3 Post-Play Interview

Post-play interviews had open-ended questions to capture essential feedback.

1. Rate controls.
2. Rate the intuitiveness of the game. Highlight the segments where you struggled and the opposite.
3. Describe the cybersecurity insights you gained.
4. Can you describe in your own words the problem Říhová encountered?
5. Can you explain in your own words why Říhová got into that problem?
6. Describe in your own words the consequences of careless actions.
7. Do you think you have ever acted carelessly in a situation similar to Říhová's?
8. If yes, would you have acted differently?
9. What was subjectively the strong point of the game?
10. What was subjectively the weak point of the game?

Appendix D

Game Menus

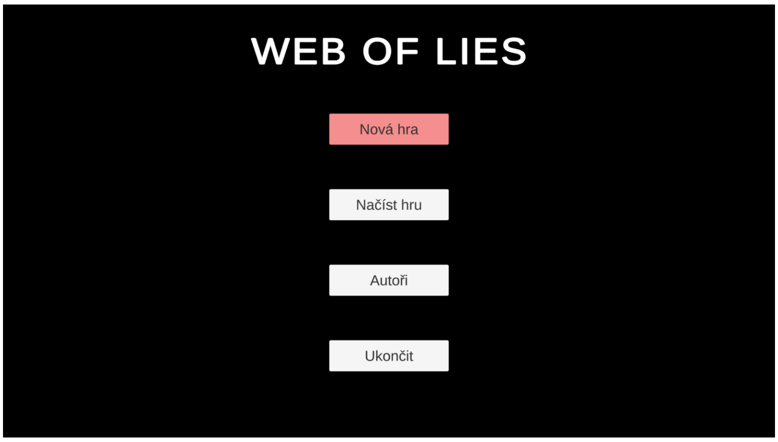


Figure D.1: Main menu

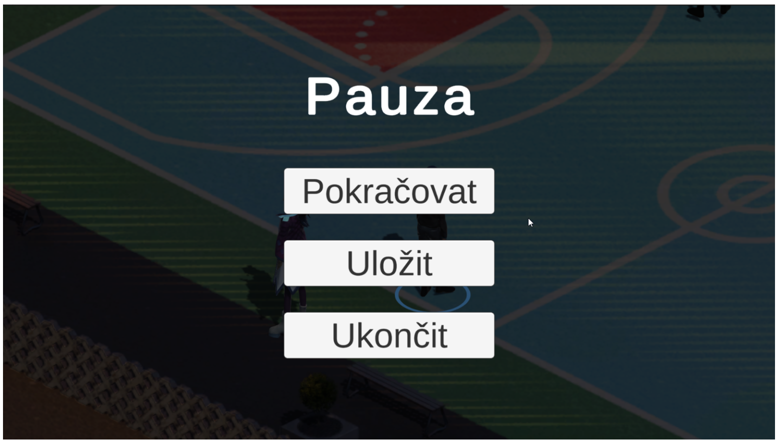


Figure D.2: Pause menu

Appendix E

Testers Table

Age	**Gender**	**Field**	**Cybersecurity Knowledge**	**Experienced Online Scam**	**Experienced Cyberbullying**	**Plays Video Games**
23	F	Student: Chemistry	Average	No	No	Occasionally
19	F	Student: CS	Above Average	No	Yes	Regularly
21	F	Student: Mathematics	Average	No	No	Regularly
23	M	Student: CS	Above Average	No	No	Often
24	M	Student: Mechanical Engineering	Bellow Average	Yes	No	Regularly
23	M	Car Mechanic	Average	No	No	Regularly

Table E.1: List of Testers

Appendix F

Story Reference

Dvě kamarádky spolu rády tráví čas po škole. Chodí do kavárny v centru města, kde je skvělá káva a wifi. Nedávno se rozhodly, že půjdou do kavárny, kde budou společně pracovat na školním projektu. Pro hledání infromací používají internet, jsou tedy připojené k veřejné wifi. Během práce na projektu si dávají krátké pauzy, kdy brouzdají internetem > kontrolují své sociální sítě, přihlašují se do školního systému, nakupují společně kosmetiku online. Odpoledne proběhne bez zvláštních událostí, kamarádky se rozejdou domů, ale večer se začne dít něco podivného. Jedna z nich se chce podívat na jednu ze sociálních sítí, ale její heslo nefunguje. Vyžádá si tedy na stránkách obnovu hesla, chce se podívat do emailu, ale ani tam její heslo nefunguje. Zavolá tedy kamarádce a zjistí, že ani její účty nefungují. Jsou z toho lehce nervózní, ale řeknou si, že se třeba někde stala chyba a ráno to jistě bude v pořádku. Ráno jdou do školy a hned první hodinu si obě zavolá ředitel do školy - z jejich školních účtů totiž dostal vulgární zprávy, kde mu dokonce vyhrožovaly. Chce vědět, co to má znamenat a hrozí jim vyhazovem. Dívky samořejmě nic netuší, pláčou a jsou zoufalé. Ředitel je pustí na vyučování s tím, že z jejich jednání vyvodí důsledky, a že se brzy znovu uvidí. Dívky v mezičase kontaktují naše hlavní hrdiny.

Naši hrdinové si promluví s dívkami, zjistí, co se stalo. Kromě zbytečných informací o tom, o čem si povídaly zjistíme, že společně pracovaly v kavárně a ten den nikde jinde nebyly. Jedeme tedy prozkoumat kavárnu a zjistíme, že zde mají veřejnou wifi. Zeptáme se personálu, zda je wifi chráněna nějakým heslem, případně jestli ji mají nějak jinak zabezpečenou. Zjistíme, že ne a to nám napoví, že pravděpodobně někdo monitoroval síť přímo z kavárny. Zkusíme se tedy připojit na kavárenskou wifi, a zapnout aplikaci monitor sítě. okamžitě začneme dostávat informace o tom, kdo co na kterém připojeném zařízení dělá, kam se kouká a podobně. Zprávy začínající https jsou nečitelné zašifrované, http zprávy ale číst můžeme. Víme tedy s velkou pravděpodobností, že v kavárně byl ve stejnou dobu i útočník, vzhledem k tomu, co nám dívky řekly o aktivitách během času v kavárně a i vzhledem k tomu, že ten den, nikde jinde kromě školy nebyly. Vrátíme se za dívkami a chceme, aby nám řekly, na jaké stránky přistupovaly > dají nám historii prohlížení. Na seznamu jsou stránky s protokolem https ale i http, mezi nimi školní systém, sociální síť i email. Víme tedy, že došlo k tomu, že útočník odposlechl síťový provoz a jelikož se http zprávy nešifrují, odchytil i přihlášení děvčat na tyto weby >získal jejich hesla. Ty následně změnil a vydával se za ně > psal do školy. Abychom měli naprostou jistotu, projedeme zařízení dívek antivirovým programem - který ale nenajde žádný podezřelý soubor - teorie o tom, že útok šel přes odposlech sítě je potvrzena.

Vrátíme se tedy do kavárny a zkoumáme, zda není možné nějak vypátrat, kdo se v kavárně nacházel ve stejný čas jako dívky. Zkusíme pohovořit s personálem, ale ti si pamatují jen to, že tam bylo ve stejnou dobu 5 dalších lidí. Nakonec si ale všimene, že v rohu kavárny je kamera. Půjdeme-li etickou cestou, požádáme personál o to, zda bychom si mohli prohlédnout záznam. Těm se nechce, ale po té, co jim řekneme příběh, nám to dovolí. Neetická cesta by byla oskenování wifi sítě, kde by nám vypadl seznam ip adres zařízení, která jsou zrovna připojená - mezi nimi i kamera. Jednoduše bychom pak jen mohli otevřít program CamCrack a na záznam bychom se dostali v úložišti kamery. Na záznamu vidíme podezřelého chlapce, který má černou kapuci, bohužel pro něj je mu ale v jednu chvíli vidět do obličeje. Uděláme si screen obrazovky a jdeme za děvčaty. Ty v něm poznají spolužáka z vedlejší třídy, který se o děvčata ucházel, ale neuspěl. Nastoupíme tedy na něj, pohrozíme mu policí a donutíme ho se přiznat a napravit vše, co udělal > říct děvčatům změněná hesla a jít se přiznat řediteli.