Visualization of Zero-Knowledge proofs that A ”Child” can Understand

Juekai Zhang

Master of Science

Computer Science

School of Informatics

University of Edinburgh

2023

Abstract

Zero-knowledge proofs (ZKPs) refer to the ability of a prover to convince a verifier that a statement is correct without providing any additional information. This study pioneers the use of interactive visualisations to explain zero-knowledge proofs, using a web game to achieve the visualisation goal of making zero-knowledge proofs accessible to non-experts and even children. The project uses Unity to build four ZKPs games of different difficulty, from simple to challenging. More than 90% of the participants who did not understand ZKPs before could choose the correct sentence to describe the concept after playing the game. According to the evaluation scale, the conclusion is that this project is both educational and entertaining, and can achieve the goal of allowing users to learn ZKPs while having a good playing experience. The game is available at `https://play.unity.com/mg/other/webgl-builds-353720`

Research Ethics Approval

This project obtained approval from the Informatics Research Ethics committee.
Ethics application number: 550871
Date when approval was obtained: 2023-07-19
The participants' information sheet and a consent form are included in the appendix.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Juekai Zhang*)

Acknowledgements

I would like to thank my supervisor, Dr Markulf Kohlweiss, who proposed this project idea and gave me a lot of freedom to develop the game.

I would like to thank the amazing ZKPs visualization works by Avi Wigderson, Amit Sahai, and Yang. Thanks to all volunteers who participated in the project survey.

I would also like to thank my parents and family. Thanks to my girlfriend, Fei, who always supports me.

Table of Contents

Chapter 1

Introduction

1.1 Project Overview

This project is available at `https://play.unity.com/mg/other/webgl-builds-353720`

Zero-knowledge proofs (ZKPs) are an intriguing concept in the field of cryptography. They enable the prover to demonstrate the validity of a statement to the verifier without revealing any additional information apart from the statement itself [8]. In an effort to enhance public understanding and accessibility, this project has been developed to create an online interactive experience that showcases the foundational principles of ZKPs. This interactive game aims to engage and educate users, providing an immersive and enjoyable learning journey through the fundamental cryptographic techniques that underpin zero-knowledge proofs. The project's objective is to elucidate the concepts behind ZKPs in a user-friendly manner, allowing a broader audience to grasp the significance and potential applications of this robust cryptographic tool. Through this interactive game, participants can acquire a practical understanding of how ZKPs operate and gain a greater appreciation for their role in bolstering privacy and security across diverse domains, including digital transactions, identity verification, and secure communication protocols.

The objective of this project is to design a game that is both user-friendly and visually captivating, delivering a clear depiction of ZKPs and their real-world applications. The game interface will prioritize simplicity and intuitive navigation, catering to users of all backgrounds and age groups. Interactive elements and step-by-step tutorials will guide players through the fundamental aspects of ZKPs. This game should also reflect

a certain degree of fun to better enhance the player experience.

1.2 Motivation

Zero-knowledge proofs (ZKPs) may appear abstract and counter-intuitive at first. However, let's consider a simple scenario to illustrate their concept [28]. Imagine you're at a bar and draw a card, without revealing its exact value. Now, you engage in a bet with another customer, aiming to prove that your card is black without disclosing the specific card.

To demonstrate and prove this, you have two choices. First, you could directly reveal your card, which would disclose all the information to the other party. Alternatively, you could opt to reveal the remaining red cards in the pile, implying that the card you hold must be black. By doing so, you provide convincing evidence without disclosing the exact card value.

In this scenario, you employ a zero-knowledge proof. While the other party cannot determine the specific content of your card, they can be convinced of its colour based on the evidence presented. This concept finds applications in cryptography and secure communication, where proving a statement without revealing unnecessary information is crucial for privacy protection.

By presenting this example, we can see that such simple examples can explain complex cryptographic concepts very well. So presenting an example or visualisation is very useful to educate the public about the concept and importance of cryptography, grasping the practical implications of zero-knowledge proofs in a relatable setting, and enhancing our understanding of their significance.

The public may be turned off by the complexities of cryptography, but they are still interested in how their information is protected. The visualisation of ZKPs enables the public to learn about cryptography in an easy-to-understand way. The public is made aware of the principles of the different implementations of ZKPs and the public is made more aware of information protection techniques.

1.3 Document Structure

This paper first introduces the history and applications of ZKPs in Chapter 2 to give the reader a general understanding of the background of ZKPs. This chapter also introduces work on ZKPs visualisation in the form of images, videos, web pages, games, and others. Research and investigation into the integration of cryptography education with games will also be discussed here.

Chapter 3 describes the pre-development process of the project. This includes game flow design and game prototyping.

Chapter 4 describes the complete development process of the project. This chapter describes in detail the tools and methodology used in the project, the platform, the game process and interface pictures, the challenges encountered and how to solve them.

Chapter 5 provides an analysis of the evaluation of the games developed for this project. The evaluation includes a general evaluation of the game design (self-evaluation form) and an evaluation by the human participants.

The last chapter gives a conclusion as to why this project is a good way to showcase ZKPs. The limitations of the project and the way forward will also be discussed.

Chapter 2

Background and Related Works

2.1 Cryptography and ZKPs

Cryptography [11] is the practice of enhancing the security of communication and information processing by converting it into an unintelligible or unforgettable form from unauthorized individuals. It encompasses diverse techniques and algorithms to ensure the confidentiality, integrity, authentication, and non-repudiation of data [14].

Zero-knowledge proofs (ZKPs) are the captivating cryptographic concept that enables one party to prove knowledge of certain information without revealing any details about that information [8]. In a ZKPs, the prover convinces the verifier that a statement is true without disclosing additional knowledge beyond the validity of the statement itself. ZKPs find applications in various domains, including authentication protocols and privacy-preserving computations.

A zero-knowledge proof of a statement must satisfy three properties:[7] **Completeness**: if the statement is true, an honest verifier will be convinced of this fact by an honest prover. **Soundness**: if the statement is false, no cheating verifier will be able to convince an honest verifier that it is true, except rare small probability. **Zero-knowledge**: if the statement is true, no verifier can learn anything other than the fact that the statement is true.

ZKPs provide a potent tool to address privacy concerns in numerous scenarios. For instance, in a ZKPs-based confidential transactions protocol, such as ZK-SNARKS [16], a user can prove their identity to a service provider without divulging their actual identity. The user can demonstrate knowledge of a secret key or password without

exposing the key itself. This facilitates authentication while safeguarding privacy. Moreover, traditional digital signatures involve revealing information about the signer's private key during the verification process. With ZK-SNARKs, a signer could prove the validity of a signature without revealing the private key.

With the development of Web3, ZKPs find applications in blockchain and cryptocurrency systems, where they can demonstrate the correctness of computations without revealing inputs or intermediate steps. This allows for secure and verifiable transactions without compromising sensitive information [20].

2.2 ZKPs Visualisation

Many visualizations have been attempted to explain ZKPs[28]. Hadas Zeilberger and Avi Wigderson provide such examples to demonstrate the concept. Hadas Zeilberger [27] shows how a prover can indirectly prove the existence of a Hamiltonian cycle in a graph without revealing any information to the verifier. Wigderson's example [2] involves coloring a map with three colors, where the verifier can only check neighboring colors each time without gaining any additional information. Yang [24] has deployed a website for the Interactive zero-knowledge 3-colorability demonstration, which visually demonstrates the existence of a solution for the three-colour diagram problem using ZKPs. The website allows the user to participate in the demonstration as a verifier, exposing one edge at a time and checking that the two colour blocks adjacent to the edge are the same colour. The verifier's confidence is also provided.

Some explanations aim to simplify zero-knowledge proofs for children, such as Quisquater et al.'s adaptation of The Ali Baba cave story, which illustrates the concept of not revealing the magic words while proving the ability to open the door [18]. In the cave story, Peggy knows a secret word to open a door, but she doesn't want to reveal it. Victor wants to confirm Peggy's knowledge without learning the word. They create a protocol where Peggy goes into the cave, chooses a path (A or B), and Victor waits outside. Victor then randomly shouts a path for Peggy to return through. If Peggy knows the word, she opens the door and returns as instructed. If Peggy doesn't know the word, her chance of correctly returning through the named path is 50%. By repeating this process, Peggy's consistent success in following Victor's requests becomes increasingly improbable if she doesn't know the word. This serves as evidence that Peggy indeed knows the secret word. The story highlights how Peggy can prove her knowledge

without disclosing the word by repeatedly demonstrating her ability to follow Victor's instructions.

Amit Sahai [1] hopes to popularize zero-knowledge proof to the public, from children, and students, to network engineers. He designed three approaches, that involve proving the presence of a puffin among penguins in a painting, revealing words written in a code box without disclosing the actual code itself, and three colorable map problems. These examples cater for different age groups, from children to university students, and range from easy to difficult, which helps in understanding the zero-knowledge proof.

2.3 Cryptography Games and Education

Cryptography games can be valuable tools for education[12], offering a fun and engaging way to teach important concepts in cryptography [26]. Cryptography games provide hands-on experiences that allow students to actively explore and interact with cryptographic concepts [9]. By participating in game-based activities, students can gain a deeper understanding of encryption, decryption, key management, digital signatures, and other fundamental principles of cryptography.

Cryptography games also promote practical problem-solving skills [26]. Games often involve puzzles, challenges, and strategic thinking. By solving these puzzles, students develop critical thinking and problem-solving skills, learning to analyze complex problems and devise effective cryptographic solutions [10]. Some games have the potential to generate interest and excitement about the field of cryptography [23]. By presenting cryptography in a playful and interactive manner, games can capture students' attention and motivate them to explore the subject further, potentially inspiring future interest in cybersecurity, computer science, or related fields.

Chapter 3

Game Design and Prototype

3.1 Game Design

This project first collects some existing interactive ZKPs visualization examples and organizes them into a table according to their different properties. As shown in Appendix A.1, The table records six common ZKPs visualization examples, respectively recording the game name, Commitments, prover/verifier actions, similar games and notes on these games. We found that some games were simpler to implement, such as the color blind game and the finding a puffin game. Some games are more complicated to implement, and may be also a bit difficult to understand, such as the candy game. These games all meet the basic conditions of ZKPs, such as the need for provers and verifiers, and a commitment. Some games look different, but they are variants, such as find puffin game and where is Waldo game. They have the same gameplay and commitment.

Considering the fun, ease of operation, and education of the game, four games were finally selected: Card game, Find puffin game, Color blind game, and 3-colorable map game. Find puffin game and Color blind game are easy to understand, so they would be a good start for beginners to learn about ZKPs. The card game is relatively challenging, and players need to choose their own strategies to prove statements to their opponents while achieving ZKPs. A 3-colorable map game is a classic example to explain ZKPs, which is used in many visualization projects.

3.1.1 Color Blind Game

The colorblind scenario is a commonly used example to explain zero-knowledge proofs (ZKPs) due to its relatability and intuitive nature. It allows individuals to understand the concept by relating to the challenge faced by a colorblind person in distinguishing between red and green. By using a simple protocol where one person proves the difference between the colors without revealing the specific colors to the other person, the scenario effectively demonstrates the need for privacy while proving knowledge. The visual representation and straightforward steps involved in the game make it accessible and help individuals grasp the fundamental principles of ZKPs, making it an effective tool for explaining the concept to a broader audience.

Imagine a scenario, where Alice (verifier) is color blind and unable to distinguish between red and green, Bob (prover) aims to prove to Alice that a red ball and a green ball are indeed different without disclosing the specific colors involved. This can be achieved through ZKPs protocol. Alice puts two balls behind her back each time, and chooses to exchange or not exchange the two balls. Bob can know whether Alice has exchanged balls by observing the colors of the two balls. Repeating this process increases Bob's confidence in proving that the two balls are different without revealing any information about the ball's color. Here's how they can proceed in detail:

1. Setup: There is a green ball and a red ball, but their actual colors are not revealed to Alice (because she is colorblind).
2. Commitment: Alice holds a ball in each hand, puts the balls behind her back, and chooses to swap them or not.
3. Challenge: Alice then shows Bob the ball and asks Bob if he swapped the ball
4. Response: Bob announces whether Alice has exchanged balls.
5. Verification: If Bob answers correctly, Alice's probability of believing Bob's statement increases.
6. Repeat: Steps 2-5 are repeated multiple times to further strengthen the proof and increase confidence.

In this case, each random guess by Bob has a 50% chance of being correct. Alice's confidence can be expressed by the following equation:

$$Confidence = 1 - (1 - 0.5)^n$$

where n is the number of repetitions.

The color blind game respect three properties of ZKPs:
Completeness: the honest verifier will be convinced of the fact that the two balls are different by the honest prover.
Soundness: if the statement is false, the verifier will find that the prover cannot every time tell whether the ball is swapped correctly, so that the verifier will not be convinced.
Zero-knowledge: if the statement is true, the color blind verifier can only learn the fact that the balls are different, but cannot learn why they are different (cannot get any color information).

3.1.2 Find Puffin Game

The game exemplifies zero-knowledge proofs (ZKPs) because it allows the prover to demonstrate knowledge of a specific fact (the presence of a puffin) without revealing any additional information (the exact location of the puffin). It showcases the concept of proving a statement while preserving privacy and confidentiality.

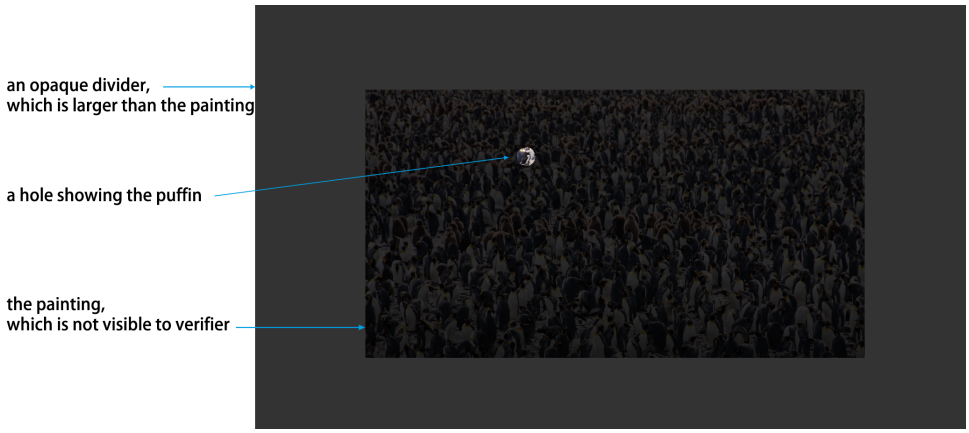


Figure 3.1: Verifier can see the puffin through the hole, but cannot know the exact location of the puffin.

The game involves a prover and a verifier. The prover claims that there is a puffin in a picture full of penguins but wants to keep the puffin's location secret. To prove this, the prover sets up a large divider, larger than the picture, as shown in Figure 3.1 and creates a hole in it. The hole's size matches the puffin in the picture. The verifier then looks through the hole and can see the puffin, verifying the prover's statement. However, since the partition is much larger than the picture, the verifier cannot determine the precise location of the puffin within the overall image. Here is how this game works in

detail:

1. Setup: The prover should find the puffin in the picture first. It might be challenging, so a hint is provided in case the player cannot find it.
2. Commitment: The prover should select the proper size of the opaque divider. Three different sizes are provided: smaller than the picture, same size as the picture, larger than the picture. The first two choices will leak the specific location of the puffin. Once the prover sets an opaque divider larger than the picture. They punched a hole in the bulkhead that matched the size of the puffin in the picture. This setup ensures that the verifier can only see the puffin through the hole and nothing else in the picture.
3. Challenge: The verifier decides to challenge the prover's claim and request evidence.
4. Response: The prover responds to the challenge by pointing the puffin to the hole in the divider.
4. Verification: The verifier observes the puffin in the picture through the hole in the divider. They can confirm that the prover's claims are true because they have visually verified the presence of the puffins. If the divider was much larger than the picture, the verifier could not determine the exact location of the puffin in the entire image. This verification process ensures that the prover can successfully prove the existence of the puffin without revealing its exact location.

The find puffin game respect three properties of ZKPs:
Completeness: the honest verifier will be convinced of the fact that there is a puffin in the picture by the honest prover.
Soundness: if the statement is false, the verifier will not see a puffin in the hole, so that the verifier will not be convinced.
Zero-knowledge: if the statement is true, the verifier can only learn the fact that there is a puffin in the picture, but cannot know the location of the puffin.

This game serves as a good example of ZKPs because it demonstrates the concept of proving knowledge without revealing specific details. The prover convinces the verifier of the presence of a puffin without disclosing its exact location. The opaque divider and the hole represent the means to provide evidence while preserving privacy. Furthermore, the game allows the player to think about the implementation of ZKPs, i.e. choose the different size of the opaque divider. It highlights the idea that the prover possesses knowledge about the puffin's existence in the picture without giving away unnecessary information. This aligns with the principles of ZKPs in cryptographic protocols.

3.1.3 Card Game

This game demonstrates the concept of zero-knowledge proofs (ZKPs) because the prover needs to convince the verifier of a specific property (the card being red) without revealing any unnecessary information (the actual card value). The prover aims to demonstrate knowledge of the card's color while keeping the value secret, highlighting the privacy-preserving aspect of ZKPs.

Before the game starts, the prover and verifier have a complete set of cards to check. The prover selects a card, assuming it is the King of Hearts, and wants to prove to the verifier that the card in their hand is red. They have two choices: either reveal the card in their hand or reveal all the remaining black cards. By revealing the hand card, the verifier knows the value but also the color. However, if the prover reveals the black cards, the verifier can only infer that the hand card is red without knowing the exact value. The prover strategically selects which cards to reveal, while the verifier questions and evaluates the provided information. The game process is designed as below:

1. Setup: The prover and verifier have a complete set of cards available for inspection before the game starts. The prover then chooses a card from the deck, assuming it is the Ace of Spades, and holds it in his hand.
2. Commitment: The prover strategically chooses which cards to reveal, aiming to convince the verifier that the card is black without revealing its value. For example, He can choose to show the cards in his hand, or other red cards.
3. Challenges:The verifier challenges the prover's claims and attempts to determine the truth of his statements.
4. Response: The prover clicks to select the cards to be displayed.
5. Verification: The verifier evaluates the information provided by the prover to determine whether it is sufficient to prove the statement that the card in the prover's hand is black. There are four cases:

- If the prover directly shows the cards in his hand, the verifier can verify and know the card information at the same time.
- If the prover does not show all red cards, the verifier cannot verify the color of the card in the prover's hand.
- If the prover shows all the red cards and some black cards, the verifier can verify

the color of the cards in the prover's hand, but the probability of the verifier's guessing the card is increased.

- If the prover only shows all the red cards, the verifier can verify the color of the card in the prover's hand, at this time the verifier has the lowest probability of guessing the card (ZKPs)

The probability formula for the verifier to guess the card is as follows (information leakage):

$$Probability = \frac{26}{26-n}$$

where n is the number of black cards(or red cards, based on the prover selection) revealed by the prover.

The card game respect three properties of ZKPs:
Completeness: the honest verifier will be convinced of the fact that the color of the card the honest prover selected .
Soundness: if the statement is false, the verifier will find the possible cheating situation, so that the verifier will not be convinced.
Zero-knowledge: if the statement is true, the verifier can only learn the the color of the card, but cannot know the exact value of the card.

This game serves as a good example of ZKPs because it illustrates how a prover can demonstrate a specific property (the color of the card) without revealing additional information (the card's value). It showcases the challenge of finding the best strategy for the prover to convince the verifier while preserving privacy. The game prompts users to explore the concept of ZKPs, understand the validity and limitations of the protocol, such as there may still be a small probability for the verifier to guess the card, and also encourages a deeper understanding of zero-knowledge proofs.

3.1.4 3-colorable Map Game

The 3-colorable map game respects ZKPs because it demonstrates how a prover can convince a verifier that they have a valid solution to a problem without revealing any information about the solution itself. In this game, the prover aims to prove that they can color a map with three colors in a way that no two adjacent regions have the same color, without disclosing the actual coloring scheme.The game process is designed as below:

1. Setup: The prover and verifier have a map with regions that need to be colored using three different colors. The map is initially uncolored.
2. Commitment: The prover secretly colors the map according to the rules (the adjacent blocks have different colors) and commits to the coloring without revealing it. After the prover completes the coloring, 6 identical coloring schemes can be generated (the color order is different)
3. Challenge: The verifier randomly selects pairs of adjacent regions on the map and asks the prover to provide the colors of those regions.
4. Response: The prover correctly provides the colors of the selected regions, demonstrating their ability to color the map in a valid way.
5. Verification: The verifier checks if the colors provided by the prover adhere to the rule that no two adjacent regions have the same color. If the prover consistently provides correct colorings, it provides strong evidence that they possess the knowledge of a valid 3-coloring solution without revealing the specific coloring scheme.

In this game, the confidence of the verifier is:

$$Confidence = 1 - (1 - \frac{1}{E})^n$$

where E is the total number of pair of adjacent blocks, n is the number of repetitions. $1 - \frac{1}{E}$ is the probability that in a single round, the prover successfully cheats. Subtracting this from 1 gives the probability of not catching the prover cheating in a single round. Raising this to the power of n gives the cumulative probability that the prover hasn't been caught cheating in any of the n rounds.

The 3-colorable map game respect three properties of ZKPs:
Completeness: the honest verifier will be convinced of the fact that the map can be colored with three colors.
Soundness: if the statement is false, the verifier will find the some adjacent fields have the same color, so that the verifier will not be convinced.
Zero-knowledge: if the statement is true, the verifier can only learn the map can be three colorable, but cannot know the exact color strategy of the map.

The 3-colorable map game is a good example of ZKPs because it showcases how a prover can convince a verifier about the existence of a valid solution without disclosing any sensitive information. The game allows the prover to prove their ability to color the map correctly while maintaining the privacy of the actual coloring scheme. Because of the random selection of six color combinations, the verifier cannot piece together

a complete map based on known information. It demonstrates the power of ZKPs in preserving confidentiality while providing a verifiable proof of knowledge. Additionally, the game is visually intuitive, making it accessible and easier for people to understand the concept of ZKPs and their practical applications.

3.2 Prototype Design

I first produced a prototype to verify the feasibility of the development. Some key functions are first implemented in the prototype. All prototype games have a prover (player, in the lower right corner) and a verifier (in the upper left corner). The prover will speak a statement, and the player needs to prove this statement to the verifier without revealing additional information. Each game has a restart button for restarting the game.

I developed the card game first, as shown in Figure 3.2a. The first card selected by the player is used as the picked card (in the upper left corner), and the player needs to prove its color to the opponent (computer). Players can click on a card to choose to reveal it. When the cursor moves over a card, the card moves up to indicate the selected card.

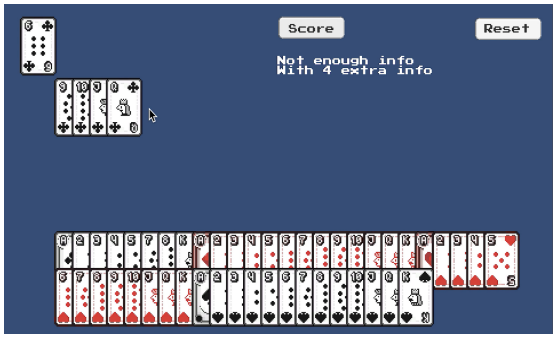
In the 3-colorable map prototype (Figure 3.2b), the player needs to drag the color object on the right to the large map in the middle first. Once the coloring is completed, 6 color combinations with the same strategy will be automatically generated, shown on the left. Every time a random small map on the left and a pair of random adjacent blocks are selected, then the verifier (computer) is allowed to reveal a pair of adjacent fields, and to check whether their colors are the same.

The color blind game is a relatively simple game (Figure 3.2c). Players need to prove to a colorblind opponent that the two balls are different. The lower left corner will display the previous showcase of the two balls. Players need to judge whether the ball is swapped according to the previous showcase and the current showcase, so as to convince the opponent.

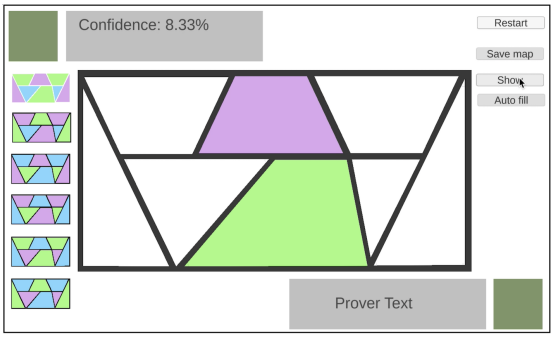
In the find puffin game, Figure 3.2d the player first needs to select the size of the opaque partition from the three buttons on the right, and then cover the picture to block the picture information. Holes showing puffins are automatically generated. Then the player clicks show to allow the opponent(computer) to verify the proof.

The development of the prototype verified the feasibility of the game. The flow above

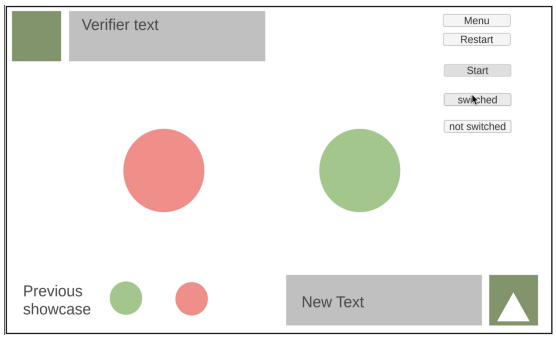
does not represent the final game flow. For example, in actual development, the 3-colorable map game has gone through multiple iterations, and the game play is different at each iteration, which will be introduced in detail in Chapter 4.



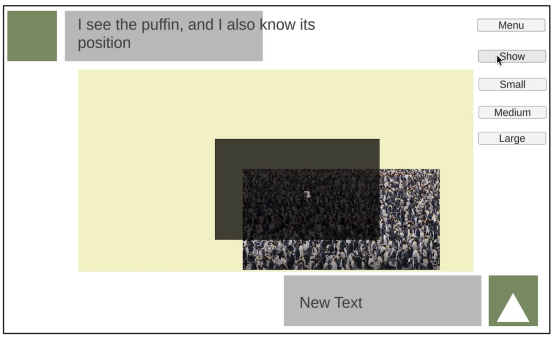
(a) Card game



(b) 3-colorable map



(c) Color blind game



(d) Find puffin game

Figure 3.2: The prototype interfaces of the four games

Chapter 4

Game Implementation

4.1 Art Style and Assets

This project uses pixel art as the game style for the following reasons. Pixel art games generally have lower system requirements than graphics-intensive games. This means that ZKPs games have the potential to run on a wider range of devices, thereby attracting a larger audience. Pixel art games generally require fewer graphical resources than more complex visual styles. This saves development time and resources, enabling more focus on the ZKPs mechanics and game design aspects of the project. In addition, the pixel style helps to unify the visual effects, making users more comfortable when switching between different games.

Figure 4.1 shows some of the game assets used in this project. Some game assets, such as pixel cards and avatars are derived from free non-commercial projects on the web. Asperite [19] is also used to create some simple pixel sprites like dialogue boxes and backgrounds.

4.2 Tools and Development Methodology

4.2.1 Game Engine: Unity

The game is made with Unity (Version 2021.3.22f1c1). Game scripts are written in C#. The game consists of a basic 2D game framework. In order to adapt to the pixel style and keep the game quality clear, I set the game size to 3480*2160. For pixel assets, I set them to 32 pixels per unit, and set the filter mode to point to ensure that the texture

Figure 4.1: Sprite sheet of the game objects. Pixel Cards [25], Pixel Portrait [13], Pixel Buttons [17]

will not appear blurry.

4.2.2 Agile development methodologies for one developer

The project uses Agile development methodologies, this is because although the general direction of the game will not change, some details and additional features may be modified during the development process. Additionally, agile development enables early identification and mitigation of risks. By breaking development down into smaller iterations, I can address potential issues or challenges early on. Frequent testing, continuous integration, and regular feedback loops help identify and resolve issues quickly, reducing overall project risk.

Since I am the only one who completes the development work, the agile development method has been adjusted to suit individual developers. For example, I don't need to communicate development details with colleagues, but every week I upload the development records to the online document to facilitate possible traceability. The development process takes a week as a cycle. I have regular weekly meetings with supervisors to analyze what has been done and set development goals for the next phase.

4.2.3 Testing

During game development, it is crucial to ensure the stability and reliability of core functionality, and unit testing is used to achieve this goal. For example, in the card game, unit tests are used to ensure that card positions do not appear in inappropriate positions. In a 3-colorable map game, unit tests are used to ensure that the map colors don't change unexpectedly as the game progresses. These tests are written to ensure that critical bugs are avoided during development that affects the overall game experience. As development progresses through various stages, comprehensive tests are conducted, running all the implemented tests to assess the game's performance and identify any potential risks. New tests are continually added to accommodate changes and additions, ensuring that the development process remains well-controlled and the quality of the game continuously improves.

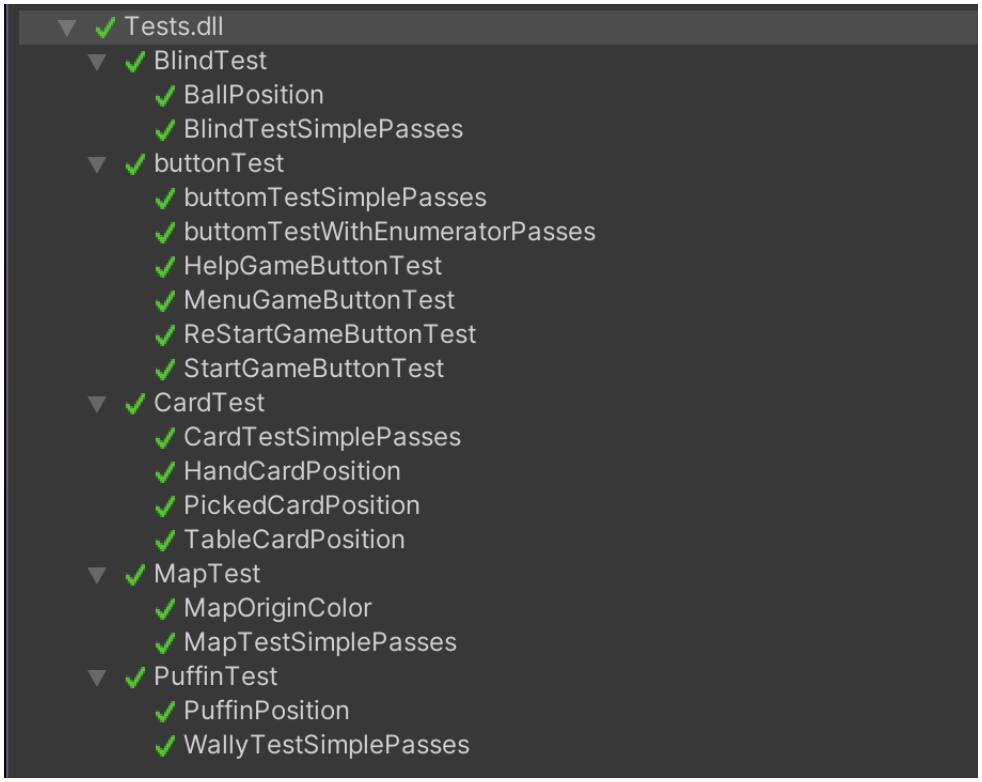
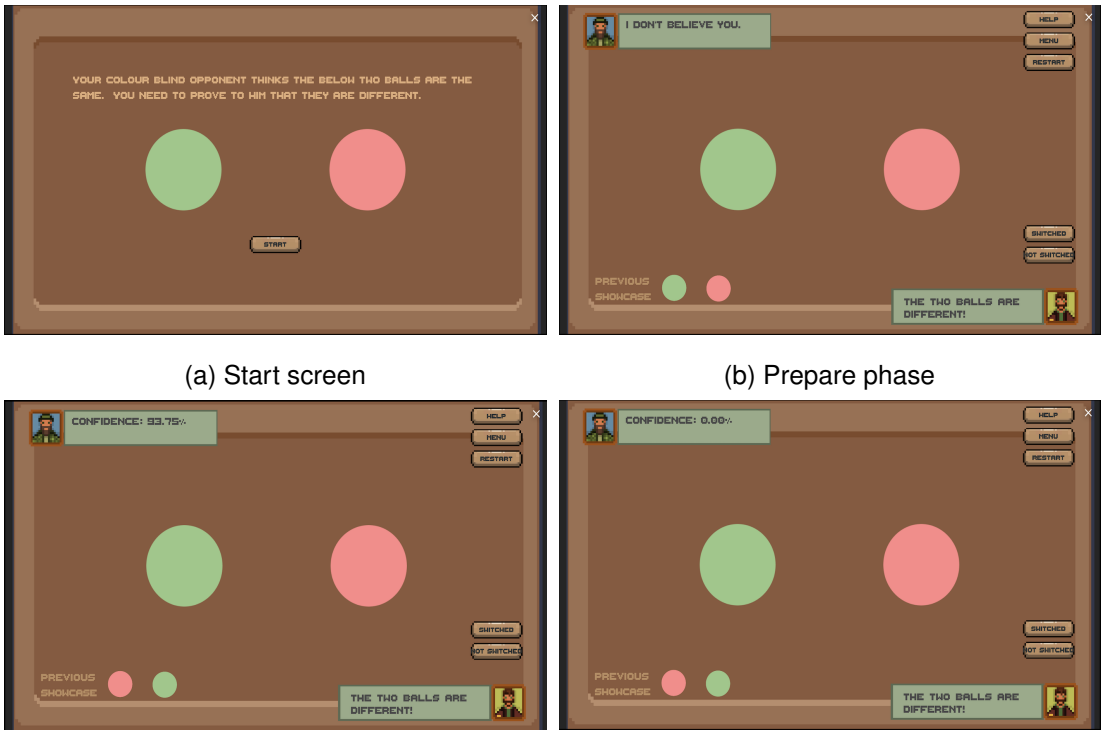


Figure 4.2: Unit Tests for the game development

4.3 Interface and Game Flow

4.3.1 Color Blind Game

After clicking start, the verifier (computer) will randomly swap the two balls (Figure 4.3b). As a memory aid, the previous showcase will be displayed in the lower left corner. Players can judge whether the verifier has swapped balls according to the showcase. The verifier's confidence will increase if the player consistently answers correctly (Figure 4.3c). Once the player answers incorrectly, the confidence will become 0% (Figure 4.3d).



(a) Start screen

(b) Prepare phase

(c) Keep answering correctly, confidence increases

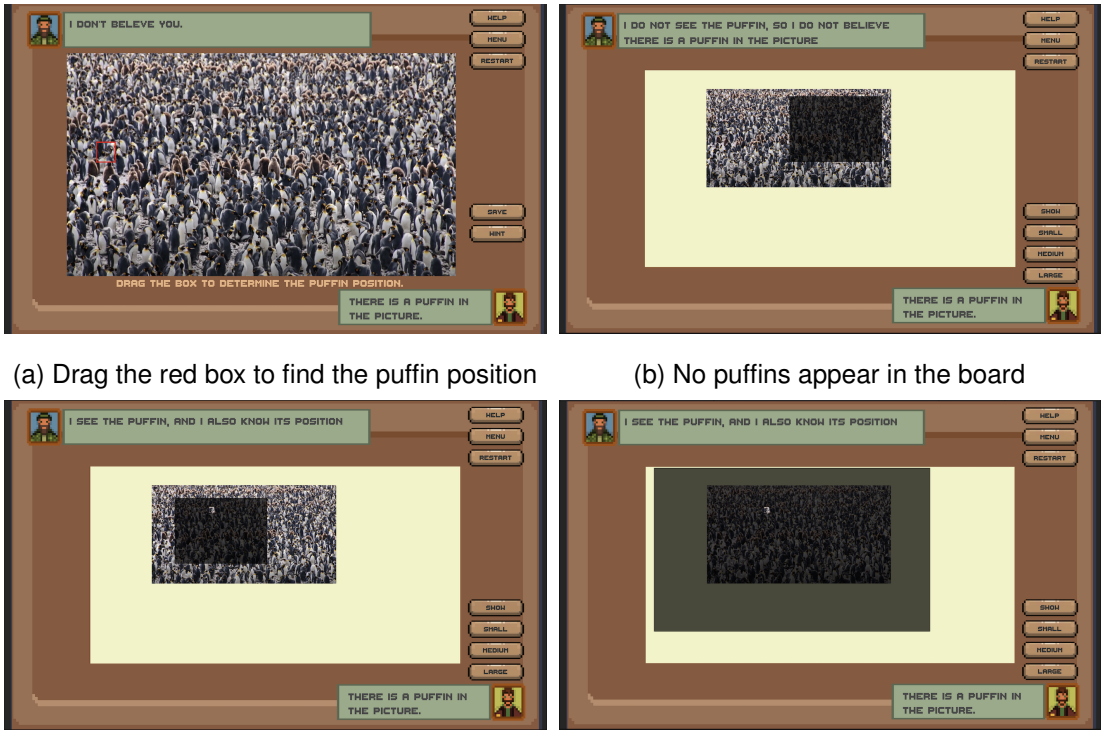
(d) Wrong answer, confidence drops to 0%

Figure 4.3: The interfaces and game flow of the color blind game

4.3.2 Find Puffin Game

After clicking the start button, the player first needs to find the location of the puffin (Figure 4.4a). This may be difficult, so players can also click the 'hint' button in the lower right corner to get location hints. Then the player needs to choose the appropriate board size to cover the picture but reveal the position of the puffin. Figure 4.4b shows

the situation where the puffin's position is not shown. Figure 4.4c shows that the board is too small, causing the verifier to know the position of the puffin. Figure 4.4d shows that the board is large enough to prove that there is a puffin while hiding the position of the puffin.



(a) Drag the red box to find the puffin position

(b) No puffins appear in the board

(c) The board is too small, so that the verifier know the position of the puffin

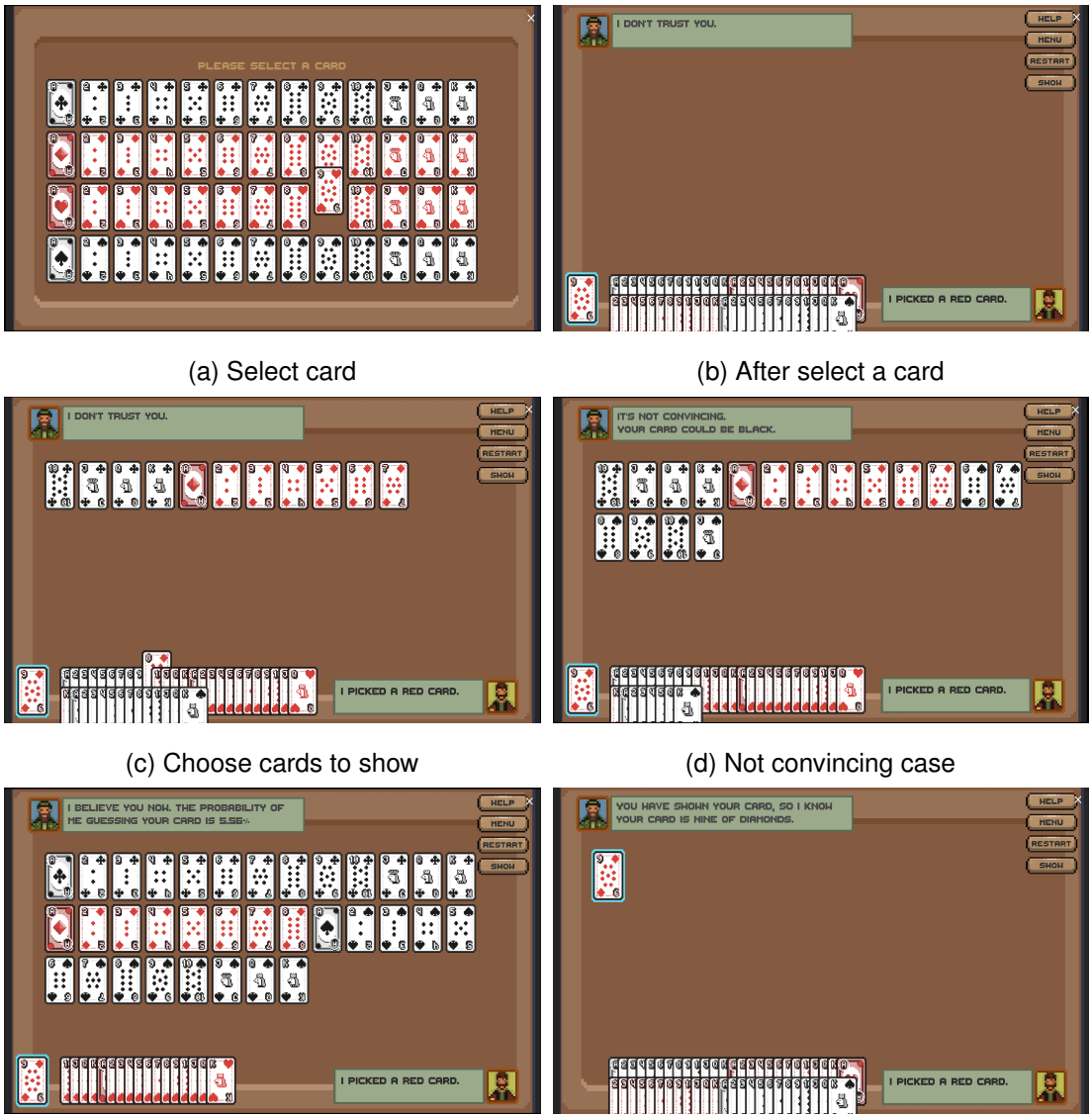
(d) The board is larger than the picture, so that the verifier does not know the position of the puffin

Figure 4.4: The interfaces and game flow of the color blind game

4.3.3 Card Game

First all 52 cards will be shown, as shown in Figure 4.5a, the player needs to choose a card. The selected card will be placed in the lower left corner, highlighted with a blue border, and the rest of the cards will be displayed below (Figure 4.5b). At this phase, the player can click to select any number of cards and place them on the table Figure 4.5c. The player can also click on a card on the table to retract it. When the player clicks the "SHOW" button, the opponent (computer) will inform the player of the result. Figure 4.5d shows the situation that did not convince the verifier. Figure 4.5e shows the situation that convinced the verifier, and at this time also shows the probability that the verifier guessed the player's hand. Figure 4.5f shows the situation where the player

directly reveals the cards in his hand.



(a) Select card

(b) After select a card

(c) Choose cards to show

(d) Not convincing case

(e) Convincing case

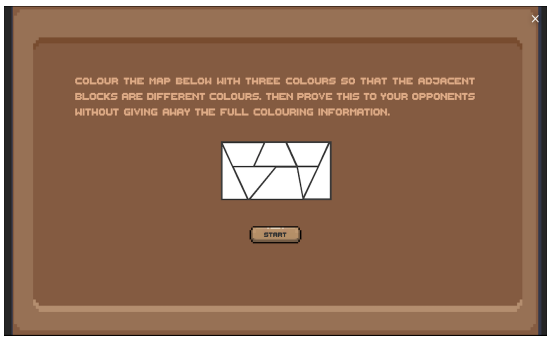
(f) Directly show the card

Figure 4.5: The interfaces of the card game, also presenting the game flow.

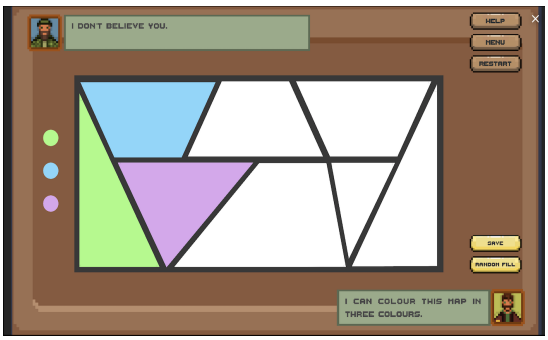
4.3.4 3-colorable Map Game

After clicking the start button, the player first needs to color the map (Figure 4.6a). There are three color objects on the left side of the screen that the player can drag to fill it in. Players can also choose ”random fill” in the lower right corner to randomly fill the color. When all the blocks are filled in, the player can click ”save” to save the color strategy. Player can next click ’permute’ button to generate maximum 6 small maps with the same strategy as the player’s coloring strategy (Figure 4.6c). The player then

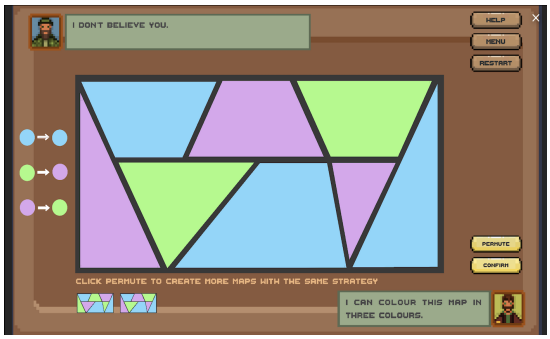
randomly chooses a minimap to display to the verifier. The verifier will also randomly chooses a pair of neighbour fields. The player can also click 'repeat' button to simplify the operation (keep the fields continuously revealed). The maximum reveal times is 50, and the confidence and information leakage of each time is displayed (Figure 4.6e). At each reveal, the highlighted color is the block that the verifier chooses to reveal, and the shaded color are other blocks, which are hidden from the verifier. If the verifier finds that two adjacent blocks have the same color, the confidence will drop to 0% (Figure 4.6f).



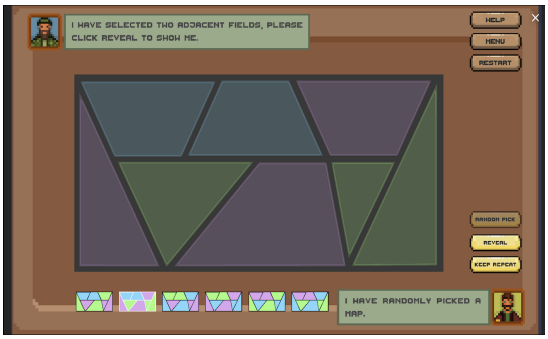
(a) Start screen



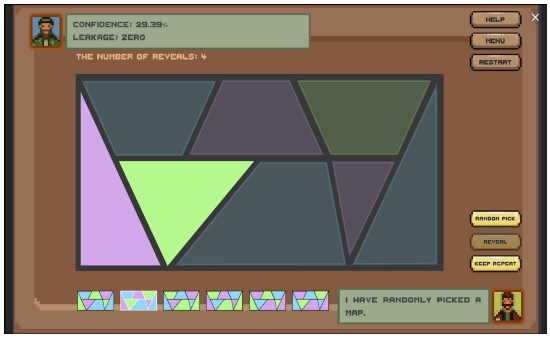
(b) Coloring the map



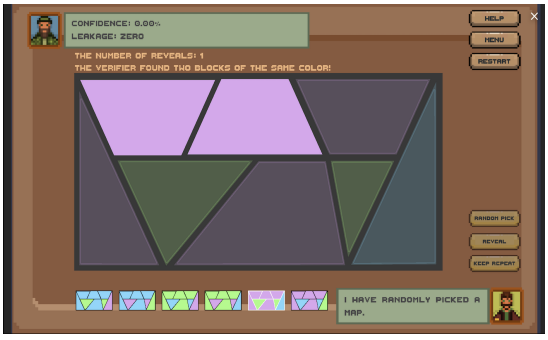
(c) Finish coloring, permute the color strategy



(d) Save map (generate 6 small maps)



(e) Convincing case



(f) Not convincing case

Figure 4.6: The interfaces and game flow of the 3-colorable map game

4.4 Challenges and Difficulties

4.4.1 Algorithms in 3-colorable Map

In the development process of 3-colorable map, how to save color and permute color combination is a key issue. First each block in the map is arranged in a list in a preset order. a private Color List variable is then created to store the colors. The order of Color List is same as Block List. Then assign each color a integer number, let's assume:

$$\text{green} \rightarrow 1$$
$$\text{blue} \rightarrow 2$$
$$\text{red} \rightarrow 3$$
So that a color list is like [1, 2, 3, 2, 1, 3, 2 ...]

The six combinations of the three colors are:

Origin	c_1	c_2	c_3	c_4	c_5	c_6
1	1	1	2	2	3	3
2	2	3	1	3	2	1
3	3	2	3	1	1	2

Where c_1 to c_6 represents the mapped number. For example, c_2 means
$1 \rightarrow 1$, $2 \rightarrow 3$, $3 \rightarrow 2$.

To create all permute lists based on the origin color list, each color can be mapped based on each combination. The steps are as follows:

for each combination c_i in [c_1, c_2, ... c_6]
    for each integer j in a color list, for example [1, 2, 3, 2, 1, 3, 2 ...]
        j = origin[j] $\rightarrow$ c_i[j]
    return the new color list

Through the above steps, six color combinations are generated, for example, if the player's combination is:[blue, red, green, blue], all excepted output lists would be: [blue, red, green, blue]; [red, blue, green, red]; [blue, green, red, blue]; [green, red, blue, green]; [green, blue, red, green]; [red, green, blue, red].

The algorithm ensures the accuracy and completeness of the results. The first map result is set as the player's own strategy, making the user interface easier to understand.

4.4.2 Collision for card games

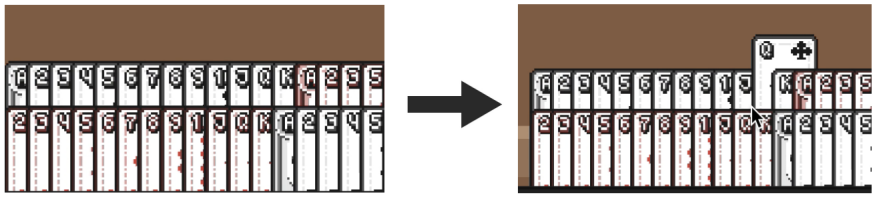


Figure 4.7: When the player moves the mouse over the card, the card will move up.

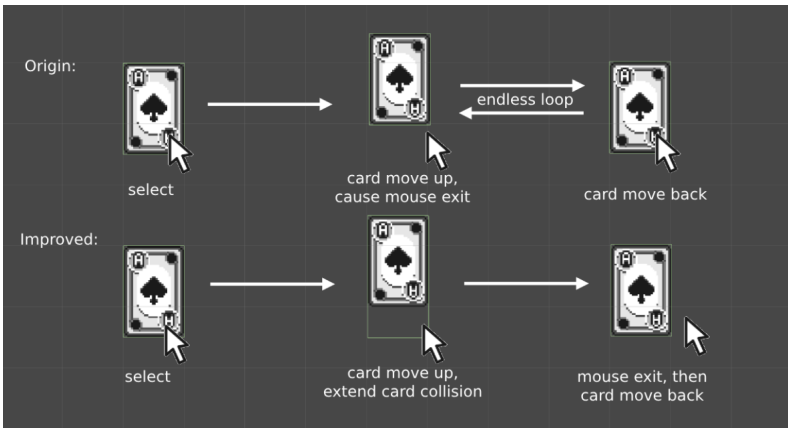


Figure 4.8: The green box indicates the collision of the card. When the player moves the mouse over the card, the card will move up to indicate the card being selected. As indicated by Origin, if the collision does not change, if the user puts the mouse on the bottom of the card, the card moves up first, making the collision out of the mouse range, and then making the card return to its original position, which causes this process to infinitely loop. After the improvement, the collision is extended, and the player must move the cursor outside the collision range to return the card to its original position.

During the development of the card game, in order to make the game clear, when the player moves the mouse over the card, the card will move up to indicate the card being selected, as shown in Figure 4.7. Many card games use this to improve the user experience. However, since the game object's collision moves with the game object, when the card moves up, the collision moves up with it. However, if the user puts the mouse on the bottom of the card, the card moves up first, making the collision out of the mouse range, and then making the card return to its original position, which causes this process to be infinitely looped, and the card continues to ”twitch”. This bug occurs in many card game implementations, such as Wingspan.[6].

To fix this, the collision is extended to the bottom of the card after the card moves up. After extending the collision, when the player cursor is at the bottom of the card, although the card is not within the cursor range, it is still within the collision range, preventing an infinite loop from being generated.

4.4.3 The iteration of 3-colorable map game

The 3-colorable map game has evolved through a series of iterations to culminate in its most challenging and interactive form.

In the initial version, the game lacked engagement as it simply involved the random selection of small maps and adjacent fields, leaving players with limited agency and interactivity. To address this, in version 2, leakage information was introduced, allowing players to observe varied results based on the number of small maps they chose, thereby gaining insights into the impact of different strategies.

In the third iteration, the leakage is changed to hard-coded. Leakage is related to the number of selected maps and the number of reveals. The more maps, the lower the leakage. The user selects the minimap to be displayed, and the verifier will automatically reveal it 50 times. Although this method simplifies the user's process, it reduces the operability of the game, making the player "watch" the game automatically running most of the time, and cannot show the interactivity well.

The fourth iteration divides the game into three phases. In the first phase, players color the map while considering adjacent areas. The second phase introduces an innovative twist, allowing players to arrange colors and generate minimaps that follow the same strategy, while also showing how colors shift, allowing players to understand the logic of color changes. The number of maps generated affects information leakage, encouraging players to think critically and strategically about their choices. This element of uncertainty adds to the appeal and keeps the game challenging. The final phase, Phase 3, emphasizes interactivity, with players randomly selecting maps and then showing them to validators. Each step is broken down to better show the game process, and attempts to closer to the real-life situation. The end result of these iterations is a three-colour map game that hopefully not only challenges players' logic and strategic acumen, but keeps them engaged through an interactive and dynamic experience.

4.5 File Structure and Version Control

I use folder hierarchy to organize the file of the project. The project mirrors the game's organizational structure by creating a clear and intuitive folder hierarchy. I use meaningful names for folders and subfolders, and group-related assets or scripts together. In the first layer of files, "Script", "Scene", "Asset", "Tests", and "Packages" folders are created. Within each folder, subfolders are created according to specific functions and different games. Also, the project establishes consistent naming conventions for files and folders. Clear and descriptive names can help quickly identify the purpose and content of each file and ease the search for specific assets or resources.

Since a Unity project is very large (more than 1G), it is not very suitable to use GitHub hosting for such a big project. Unity version control is used to host the game project. Similar to traditional git projects, it allows tracking changes to project source code, scripts and assets over time. This makes it easy to review and revert changes, compare different versions, and track the evolution of the codebase. Version control acts as a backup system, providing a centralized repository where all project files are stored. Unlike GitHub, the Unity version control system helps manage large binary files such as art assets, audio files, and scene data. They employ efficient algorithms to track changes to these files without storing redundant data, thereby optimizing storage space and reducing network transfer times.

4.6 Game Deployment

Unity Play Platform [21] is used in this project to host the game. Unity Play supports the use of webGL, allowing games to be played directly in a web browser without additional downloads or installations. This accessibility greatly broadens the potential player base, making it easier for players to enjoy games without hindrance. Unity Play's seamless integration with the Unity game engine further enhances its appeal. For this game, online games are easier to distribute to the public. It is also more conducive to iteration because players do not need to download and update the application, but all updates will be done online.

During the development process, I can update the game at any time to quickly respond to potential bugs. In addition, its good compatibility with Unity also reduces my server setup time and maintenance costs, and improves game development efficiency.

Furthermore, security and stability are also reasons to choose Unity Play. By hosting games on Unity Play, the risk of attack or hacking can be reduced. Compared with the self-built server host game, this saves setting up the defence system. Unity Play also handles load issues well, allowing a large number of players to play the game at the same time without overloading the server.

Chapter 5

Evaluation

5.1 User Interface Evaluation: Using Nielsen's 10 Usability Heuristics

Nielsen's 10 Usability Heuristics [15] is widely used in the self-evaluation of user interface design. This project uses Nielsen's 10 Usability to examine general principles of game design to enhance the player experience. Compared with System Usability Scale [4], Nielsen's 10 Usability Heuristics is more general and more in line with the evaluation of the game interface. The following is the interpretation of the results of Nielsen's 10 Usability Heuristics.

1. Visibility of system status

There is descriptive text in the game, which will be displayed at some stages to remind the player. For example, in a card game, the selection phase prompts the player to tap to select a card. In the Find Puffin game, the player is first prompted to find the location of the puffin. In addition, in all games there is a help button in the upper right corner of the game. When the player hovers over the button, a help interface will pop up to guide the user on available operations.

2. Match between system and the real world

The design of the prover and verifier dialogue boxes conforms to the player's realistic habits, which imitates the form of most chat applications, using avatars and chat bubbles.

3. User control and freedom

Players have a certain degree of freedom of operation when playing, but also some

limitations to make sure players reach their game goals. For example, in the Card game, the user can click on any card to operate, without limiting the range that the user can click. In the 3-colorable map, players can actually color the map without following the rules, and can color the map arbitrarily. In the process of free explanation, players can know the results related to their selection, which is conducive to a better understanding of the principle of zero-knowledge proof. In addition, the game has a 'restart' button that allows the user to restart the game under certain circumstances. Unwanted states can be reverted without a long process, improving the user gaming experience.

4. Consistency and standards

The art style of the game and the layout design is unified. In each game, there are common buttons such as 'menu', 'help', and 'restart' in the upper right corner, and specific buttons for specific games in the lower right corner. The button color, game font, and game background are unified, so that players can quickly learn another game after playing one game, reducing the user learning costs.

5. Error prevention

The game has passed the unit test, which can avoid most errors. For example, players are less likely to encounter problems with operations, including clicking cards, dragging objects, and clicking buttons. If an unexpected error occurs, players can click 'restart' to restart the game and start over.

6. Recognition rather than recall

In the Color Blind Game, the verifier will exchange the positions of the two balls, and the player needs to judge whether the verifier has been exchanged compared to the previous one. In order to reduce the player's memory load, a previous showcase is displayed in the lower-left corner of the game. Players don't need to rely on memory but can directly know the last showcase through prompts, which improves the player's gaming experience.

7. Flexibility and efficiency of use

In the 3-colorable map, there is a 'random fill' button in the lower right corner. After clicking this button, the map will be randomly filled with colors. Compared with dragging the color to fill in each time, this method is more flexible and efficient for advanced players.

8. Aesthetic and minimalist design

The overall design of the user interface is simple, and tries to use indicative pictures

instead of text. Text descriptions are used only when required. whenever possible information is hidden after the user make the need operation, making the game interface more concise and clear.

9. Help users recognize, diagnose, and recover from errors
Specific errors will prompt the user in the game interface. In the 3-colorable map, if the user does not fill the map color but clicks the 'save' button, the bottom of the map will prompt the player to fill all the blocks before proceeding.

10. Help and documentation
There is a help button in the upper right corner of the game. When the player hovers over the button, a help interface will pop up to guide the user on available operations. The help panel uses short sentences to introduce each function, reducing the trouble caused by cumbersome text to users.

Overall, the game's user interface and functionality follow Nielsen's 10 Usability Heuristics. This project prioritizes the player's experience, creating a game that is user-friendly, efficient and enjoyable. Good design encourages players to continue playing, and explains the concept of zero-knowledge proofs during the game better.

5.2 Evaluation with Human Participants

To verify the effectiveness of the game, some volunteers were recruited to evaluate the project. The questionnaire is attached in Appendix B. I distribute questionnaires and game links to participants via email. Participants were first asked to play the game, and feedback was collected through an anonymous online questionnaire. The questionnaire is divided into three parts. The first part first collects the background of the participants, that is, whether they have prior knowledge about ZKPs. The second part is the system usability scale (SUS) evaluation, which is used to test the usability of the game. The third part is the subjective evaluation of the games, such as whether the game is interesting, whether the game is easy to understand, etc. Here is the display and analysis of the results.

5.2.1 Understanding of ZKPs

As shown in Figure 5.1, it can be seen from the results that 91% of the participants did not understand the concept of zero-knowledge proof before, but after playing the

game, the proportion of correct answers to the concept of zero-knowledge proof is about 90%, which shows that this project is to a certain extent helpful to the public for understanding the basic concept of ZKPs.

Figure 5.1: Participants have no prior knowledge the ZKPs.

5.2.2 System Usability Scale Result

Table 5.1 shows the results of System Usability Scale. In the SUS scale, the higher the score for odd-numbered questions, the better, and the lower the score for even-numbered questions, the better. Participants were generally satisfied with the user interface, with average scores above 4 for both questions 3 and 5. A score of 1.5 for question 6 also indicates that the system does not have many inconsistent designs. The poor scores were for system usage. Questions 4, 7, and 10 did not score well, probably because the learning cost of the game is relatively high, and some functions and processes are not intuitive enough. Players may need multiple attempts to fully understand how to

operate. The total SUS score is 76.275. According to the scale (Figure 5.2), this game is between good and excellent.

SUS result	
SUS	**Average Score**
1. I think that I would like to use this system frequently.	4.17
2. I found the system unnecessarily complex.	2.00
3. I thought the system was easy to use.	4.25
4. I think that I would need the support of a technical person to be able to use this system.	2.33
5. I found the various functions in this system were well integrated.	4.58
6. I thought there was too much inconsistency in this system.	1.50
7. I would imagine that most people would learn to use this system very quickly.	3.75
8. I found the system very cumbersome to use.	2.08
9. I felt very confident using the system.	4.17
10. I needed to learn a lot of things before I could get going with this system.	2.5
SUS overall score	**76.275**

Table 5.1: System usability scale result. The formula for calculating the overall score is:
1: Convert strongly disagree to strongly agree to 1 to 5
2: X = Sum of the score for all odd-numbered questions – 5
3: Y = 25 – Sum of the score for all even-numbered questions
4: Overall SUS Score = (X + Y) x 2.5

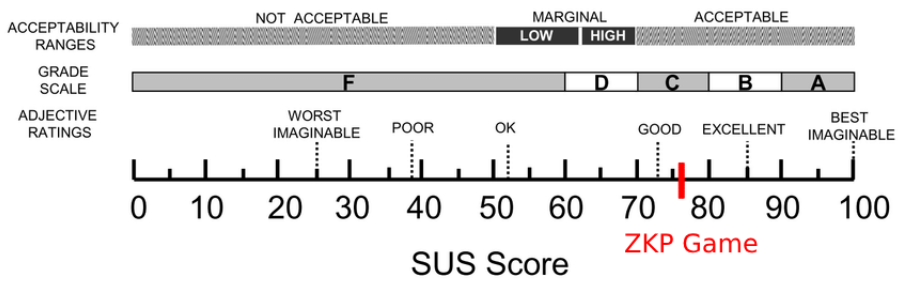


Figure 5.2: SUS scale, Modified from [3]

5.2.3 Participant feedback result

The scale in this section is the same as SUS, convert strongly disagree to strongly agree to 1 to 5. As shown in Table 5.2, the average scores of most questions are more than 4 points, indicating that the participants are generally satisfied with the game. Most of the participants think that the game interface is clear and the guidance is clear, which is reflected in the higher scores of questions 1 to 4.

For the four games, most users think that “Color blind game” and “Card Game” are the easiest to understand, and “3-colorable map” is the most difficult to understand, with scores below 4 points. This may be due to the more complicated operation of ’3-colorable map’, and the ZKPs logic behind it is not intuitive enough compared to other games. In addition, it scored 4 points for whether the game is fun, although as an educational game this project is a good balance between education and entertainment.

User game feedback result	
Question	**Average Score**
1. The game description is clear.	4.33
2. I can understand what each button does.	4.67
3. When playing the game, I know my goal.	4.58
4. When playing the game, I know what I’m doing.	4.50
5. I think the game is fun.	4.00
6. The game “color blind game” is easily understandable.	4.42
7. The game “find puffin game” is easily understandable.	4.17
8. The game “Card Game” is easily understandable.	4.42
9. The game “3-colorable map” is easily understandable.	3.92
10. The game made me understand zero-knowledge proof to a certain extent.	4.50

Table 5.2: User game feedback result

5.2.4 Evaluation Summary

In general, the participants’ evaluation of the game is relatively positive, which also shows to a certain extent that this game can bring users the basic concept of zero-knowledge proof.

However, questionnaire evaluation also has limitations. I didn’t participate in the

observation of the participants during the play, so I can't know which parts of the process are more difficult for the players. For example, some players said that the 3-colorable map game is difficult to understand, but I can't know which part it is, such as whether is it difficult to color the map or the game flow is not clear. In addition, in the questions about the concept of ZKPs, participants can directly search for answers on the Internet, so the evaluation results can only be used as a reference and cannot be used as conclusive evidence. In the future, it may be necessary to develop a more scientific and precise method to evaluate the game's effectiveness.

Chapter 6

Conclusion and Future Works

6.1 Achieved Result and Limitations

In conclusion, the ZKPs (Zero-Knowledge Proof) game presented in this project has successfully achieved its objectives of visualizing ZKPs concepts in an interactive and engaging manner. Through the use of Unity, a captivating game was designed, implemented, and deployed, ensuring that it not only provided entertainment but also respected Soundness and Zero-knowledge of ZKPs.

By incorporating the feedback and evaluation from human participants, the game has proven to be an effective and accessible tool for explaining ZKPs concepts in an easy-to-understand manner. The combination of gameplay elements and educational content has resulted in an immersive experience that allows players to grasp the intricacies of ZKPs without feeling overwhelmed or disinterested.

This project has paved the way for innovative approaches to teach complex cryptographic concepts, and the success of the ZKPs game encourages further exploration in the realm of interactive educational tools. As the world increasingly relies on privacy and security in the digital age, spreading awareness and understanding of concepts like ZKPs becomes essential, and this game serves as an exemplary model for achieving that goal.

However, the project has some obvious limitations. In the 3-colorable map game, this project uses a simple heuristic method to represent the degree of information leakage. Some other statistical methods or information leakage calculation methods may be used here in the future.

Overall, the ZKPs game has demonstrated the potential of gamification in making abstract ideas more accessible to a broader audience. It has not only accomplished its intended objectives but also holds promise as a stepping stone for future developments in the field of cybersecurity education and cryptography visualization.

6.2 Future Works

In future work, game iteration is an important task. Based on collected player feedback, clearer instructions need to be added to the game, as some players have stated that learning how to play the entire game is time-consuming. In addition, the fun of the game can also be improved. In the Color Blind Game, finding the difference between two pictures is used, which improves the fun of the game to a certain extent. Advanced versions of other games will be developed in the future to enhance the fun of the game and attract players better. For example, in a 3-colorable game, non-planar maps and examples of more than three colors will appear in the game as challenges. Add other pictures to the find puffin game. For each set different difficulty to suit different stages of players. Some features to improve user experience are also in the future development plan. For example, add a ”magnifying glass” function in the find puffin game to facilitate players to find the target.

Moreover, when calculating information leakage in 3-colorable map games, using statistical methods such as t-test or other information leakage algorithms will be more accurate than the current hard coding. In terms of new game features, maybe in the future, a multiplayer online mode can be developed instead of just a single-player online mode. Allow players to act as provers or verifiers, allow players to bluff, enhance the fun of the game, and increase the interaction and challenge between players.

Bibliography

[1] WIRED Amit Sahai. Computer scientist explains one concept in 5 levels of difficulty. https://www.youtube.com/watch?v=fOGdb1CTu5c, 2022.

[2] Numberphile Avi Wigderson. Zero knowledge proof (with avi wigderson) - numberphile. https://www.youtube.com/watch?v=5ovdoxnfFVc, 2021.

[3] Aaron Bangor, Philip Kortum, and James Miller. Determining what individual sus scores mean: Adding an adjective rating scale. *Journal of usability studies*, 4(3):114–123, 2009.

[4] Aaron Bangor, Philip T Kortum, and James T Miller. An empirical evaluation of the system usability scale. *Intl. Journal of Human–Computer Interaction*, 24(6):574–594, 2008.

[5] cossacklabs. Zero knowledge proof: Explain it like i'm 5. https://hackernoon.com/eli5-zero-knowledge-proof-78a276db9eff, 2020.

[6] Monster Couch. Wingspan. https://store.steampowered.com/app/1054490/Wingspan/.

[7] Uriel Fiege, Amos Fiat, and Adi Shamir. Zero knowledge proofs of identity. In *Proceedings of the nineteenth annual ACM symposium on Theory of computing*, pages 210–217, 1987.

[8] Shafi Goldwasser, Silvio Micali, and Chales Rackoff. The knowledge complexity of interactive proof-systems. In *Providing sound foundations for cryptography: On the work of shafi goldwasser and silvio micali*, pages 203–225. Goldwasser, Shafi and Micali, Silvio and Rackoff, Chales, 2019.

[9] Winston Anthony Hill Jr, Mesafint Fanuel, Xiaohong Yuan, Jinghua Zhang, and Sajad Sajad. A survey of serious games for cybersecurity education and training.

DigitalCommons@Kennesaw State University, 2020.

[10] Alice Jaffray, Conor Finn, and Jason RC Nurse. Sherlocked: A detective-themed serious game for cyber security education. In *Human Aspects of Information Security and Assurance: 15th IFIP WG 11.12 International Symposium, HAISA 2021, Virtual Event, July 7–9, 2021, Proceedings 15*, pages 35–45. Springer, 2021.

[11] Jonathan Katz and Yehuda Lindell. *Introduction to modern cryptography*. CRC press, 2020.

[12] Manzoor Ahmed Khan, Adel Merabet, Shamma Alkaabi, and Hesham El Sayed. Game-based learning platform to enhance cybersecurity education. *Education and Information Technologies*, pages 1–25, 2022.

[13] Loregret. Pixel-art portrait pack. `https://loregret.itch.io/pixel-art-portrait-pack`.

[14] Alfred J Menezes, Paul C Van Oorschot, and Scott A Vanstone. *Handbook of applied cryptography*. CRC press, 2018.

[15] Jakob Nielsen. Ten usability heuristics. 2005.

[16] Alexandre Miranda Pinto. An introduction to the use of zk-snarks in blockchains. In *Mathematical Research for Blockchain Economy: 1st International Conference MARBLE 2019, Santorini, Greece*, pages 233–249. Springer, 2020.

[17] Pokekas. Pixelart buttons. `https://pokekas.itch.io/pixelart-buttons`.

[18] Jean-Jacques Quisquater, Myriam Quisquater, Muriel Quisquater, Michaël Quisquater, Louis Guillou, Marie Annick Guillou, Gaïd Guillou, Anna Guillou, Gwenolé Guillou, and Soazig Guillou. How to explain zero-knowledge protocols to your children. In *Advances in Cryptology—CRYPTO'89 Proceedings*, pages 628–631. Springer, 2001.

[19] Igara Studio S.A. Aseprite, animated sprite editor & pixel art tool. `https://www.aseprite.org/`.

[20] Xiaoqiang Sun, F Richard Yu, Peng Zhang, Zhiwei Sun, Weixin Xie, and Xiang Peng. A survey on zero-knowledge proof in blockchain. *IEEE network*, 35(4):198–205, 2021.

[21] Unity Technologies. Unity play, play, create, inspire. https://play.unity.com/.

[22] Computational Thinking. Zero knowledge proofs. https://www.youtube.com/watch?v=5qzNe1hk0oY, 2022.

[23] Srikanth Vadla, Abhishek Parakh, Parvathi Chundi, and Mahadevan Surbamaniam. Quasim: A multi-dimensional quantum cryptography game for cyber security. In *Journal of The Colloquium for Information Systems Security Education*, volume 6, pages 19–19, 2019.

[24] Edward Z. Yang. Interactive zero knowledge 3-colorability demonstration. https://web.mit.edu/~ezyang/Public/graph/svg.html.

[25] Yewbi. Base pixel playing card set. https://unbent.itch.io/yewbi-playing-card-set-1.

[26] Younis A Younis and Mohammed Yahya Alghamdi. The use of computer games for teaching and learning cybersecurity in higher education institutions. *Journal of Engineering Research*, 9(3A), 2021.

[27] Hadas Zeilberger. A simple explanation of zero knowledge proofs. https://medium.com/web3studio/a-simple-explanation-of-zero-knowledge-proofs-ca574092e73b, 2019.

[28] Juekai Zhang. Informatics project proposal, visualization of zero-knowledge proofs that a ”child” can understand, 2023.

Appendix A

Different ZKPs visualization comparison table

Different ZKPs visualization comparison					
Name	**Commitments**	**Prover actions**	**Verifier actions**	**Game variant**	**Note**
Card game	The cards are complete and not cheating	Select which card(s) to reveal	Question or believe		
Combination lock game[1]	The Combination lock box	Unlock the box. Tell the secret message	Write the secret message, and then verify it		Might be no choices for player
Find puffin game[1]	maybe the painting not be changed	Choose different size of opaque divider	See if there is a puffin in the hole	Where is Waldo, Where is Wally	
Color blind people game [22]	maybe the balls, they are not changed	Point out if verifier switch balls or not	Switch the apples or not	The Ali Baba cave game.	require repeating several times
Color map game [2]	Envelops, so when verifier revealing, no color in envelops can be changed	Color the map with different color combinations.	Reveal two neighbor envelops, check if they are different color	Sudoku game, a prover uses colors to hide numbers. Each time a verifier reveals a row, a column, or a 3*3 section.	Require repeating several times
Candy game [5]	the locked box	Write “+” or “-”, put them in the locked box	Reveal one box, and show it to the prover.	Yao’s Millionaires’ problem	Prover and verifier can be switched

Table A.1: Different ZKPs visualization comparation

Appendix B

Participant Feedback Questionnaire

Zero-Knowledge proofs game

Game Link: https://play.unity.com/mg/other/webgl-builds-353720

This study pioneers the use of interactive visualisations to explain zero-knowledge proofs, using a web game to achieve the visualisation goal of making zero-knowledge proofs accessible to non-experts and even children. The study will collect your feedback on the game from concept understanding, user interface, user experience, fun level, etc.

* Required

Participant Information Sheet

This study was certified according to the Informatics Research Ethics Process, reference number 550871. Please take time to read the following information carefully. You should keep this page for your records.

Who are the researchers?
Juekai Zhang, Markulf Kohlweiss

What is the purpose of the study?
This study pioneers the use of interactive visualisations to explain zero-knowledge proofs, using a web game to achieve the visualisation goal of making zero-knowledge proofs accessible to non-experts and even children. The study will collect your feedback on the game from concept understanding, user interface, user experience, fun level, etc.

Why have I been asked to take part?
You have been invited to take part in the study because you volunteered to contribute feedback to the game.

Do I have to take part?
No – participation in this study is entirely up to you. You can withdraw from the study at any time, without giving a reason. Your rights will not be affected. If you wish to withdraw, contact the PI. We will stop using your data in any publications or presentations submitted after you have withdrawn consent. However, we will keep copies of your original consent, and of your withdrawal request.

What will happen if I decide to take part?
- You will be first asked to play the zero-knowledge proofs game which is deployed on the internet

ternet.

- You will be asked questions related to the zero-knowledge proofs game, including concept understanding, user interface, user experience, fun level.
- The whole process takes less than 30 minutes (including the playing game time)
- Your action will not be recorded. Your answers will be anonymized.

Are there any risks associated with taking part?
There are no significant risks associated with participation.

Are there any benefits associated with taking part?
No.

What will happen to the results of this study?
The results of this study may be summarised in published articles, reports and presentations. Quotes or key findings will be anonymized: We will remove any information that could, in our assessment, allow anyone to identify you. With your consent, information can also be used for future research. Your data may be archived for a maximum of 4 years.

1.

Data protection and confidentiality.
Your data will be processed in accordance with Data Protection Law. All information collected about you will be kept strictly confidential. Your data will be referred to by a unique participant number rather than by name. Your data will only be viewed by the researcher/research team: Juekai Zhang, Markulf Kohlweiss.
All electronic data will be stored on a password-protected encrypted computer, on the School of Informatics' secure file servers, or on the University's secure encrypted cloud storage services (DataShare, ownCloud, or Sharepoint) and all paper records will be stored in a locked filing cabinet in the PI's office. Your consent information will be kept separately from your responses in order to minimise risk.

What are my data protection rights?
The University of Edinburgh is a Data Controller for the information you provide. You have the right to access information held about you. Your right of access can be exercised in accordance Data Protection Law. You also have other rights including rights of correction, erasure and objection. For more details, including the right to lodge a complaint with the Information Commissioner's Office, please visit www.ico.org.uk. Questions, comments and requests about your personal data can also be sent to the University Data Protection Officer at dpo@ed.ac.uk.

For general information about how we use your data, go to: edin.ac/privacy-research

Who can I contact?
If you have any further questions about the study, please contact the lead researcher, Juekai Zhang, s2343556@ed.ac.uk.

If you wish to make a complaint about the study, please contact

inf-ethics@inf.ed.ac.uk. When you contact us, please provide the study title and detail the nature of your complaint.

Updated information.
If the research project changes in any way, an updated Participant Information Sheet will be made available on http://web.inf.ed.ac.uk/infweb/research/study-updates.

Consent
By proceeding with the study, I agree to all of the following statements:

statements:

· I have read and understood the above information.

· I understand that my participation is voluntary, and I can withdraw at any time.

· I consent to my anonymised data being used in academic publications and presentations.

I allow my data to be used in future ethically approved research. *

- ◯ I agree
- ◯ I disagree

Please first play the game through the following link

https://play.unity.com/mg/other/webgl-builds-353720

2. Do you know Zero-Knowledge proofs game before? *

- ◯ Yes
- ◯ No

3. After playing the games, which of the following sentences do you think best describes a zero-knowledge proof? *

- ◯ A zero-knowledge proof is a process by which one party shares all the details of their knowledge with another party to ensure transparency and trust.
- ◯ A zero-knowledge proof is a method used in cybersecurity to demonstrate that a system has no vulnerabilities or weaknesses.
- ◯ A zero-knowledge proof is an authentication mechanism that requires a user to provide no credentials or passwords.
- ◯ A zero-knowledge proof is a cryptographic technique that allows one party to prove to another party that a statement is true without revealing any information about how the statement is true.

System Usability Scale

In this section, you need to rate each item from 1-5.
1 is regarded as strongly disagree
2 is regarded as disagree
3 is regarded as neutral
4 is regarded as agree
5 is regarded as strongly agree

4. I think that I would like to use this system frequently.

1	2	3	4	5

5. I found the system unnecessarily complex.

1	2	3	4	5

6. I thought the system was easy to use.

1	2	3	4	5

7. I think that I would need the support of a technical person to be able to use this system.

1	2	3	4	5

8. I found the various functions in this system were well integrated.

1	2	3	4	5

9. I thought there was too much inconsistency in this system.

1	2	3	4	5

10. I would imagine that most people would learn to use this system very quickly.

1	2	3	4	5

11. I found the system very cumbersome to use.

1	2	3	4	5

12. I felt very confident using the system.

1	2	3	4	5

13. I needed to learn a lot of things before I could get going with this system.

1	2	3	4	5

Game feedback

In this section, you need to rate each item from 1-5.
1 is regarded as strongly disagree
2 is regarded as disagree
3 is regarded as neutral
4 is regarded as agree
5 is regarded as strongly agree

14. The game description is clear *

1	2	3	4	5

15. I can understand what each button does *

1	2	3	4	5

16. When playing the game, I know my goal *

1	2	3	4	5

17. When playing the game, I know what I'm doing *

1	2	3	4	5

18. I think the game is fun *

1	2	3	4	5

19. The game “color blind game” is easily understandable *

1	2	3	4	5

20. The game “find puffin game” is easily understandable *

1	2	3	4	5

21. The game “Card Game” is easily understandable *

1	2	3	4	5

22. The game “3-colorable map” is easily understandable *

1	2	3	4	5

23. The game made me understand zero-knowledge proof to a certain extent *

1	2	3	4	5

24. Please write any other comments here

Appendix C

Ethics information

C.1 Participants' information sheet

Participant Information Sheet

Project title:	Visualization of Zero-Knowledge proofs that A Child can Understand
Principal investigator:	Markulf Kohlweiss
Researcher collecting data:	Juekai Zhang s2343556
Funder (if applicable):	

This study was certified according to the Informatics Research Ethics Process, reference number 550871. Please take time to read the following information carefully. You should keep this page for your records.

Who are the researchers?

Juekai Zhang, Markulf Kohlweiss

What is the purpose of the study?

Zero-knowledge proofs refer to the ability of a prover to convince a verifier that a statement is correct without providing any additional information. This study pioneers the use of interactive visualisations to explain zero-knowledge proofs, using a web game to achieve the visualisation goal of making zero-knowledge proofs accessible to non-experts and even children. The study will collect your feedback on the game from concept understanding, user interface, user experience, fun level, etc.

Why have I been asked to take part?

You have been invited to take part in the study because you volunteered to contribute feedback to the game.

Do I have to take part?

No – participation in this study is entirely up to you. You can withdraw from the study at any time, without giving a reason. After this point, personal data will be deleted and anonymised data will be combined such that it is impossible to remove individual information from the analysis. Your rights will not be affected. If you wish to withdraw, contact the PI. We will keep copies of your original consent, and of your withdrawal request.

What will happen if I decide to take part?

- You will be first asked to play the zero-knowledge proofs game which is deployed on the internet.
- You will be asked questions related to the zero-knowledge proofs game, including concept understanding, user interface, user experience, fun level.
- The whole process takes less than 30 minutes (including the playing game time)
- Your action will not be recorded. Your answers will be anonymized.

Are there any risks associated with taking part?

There are no significant risks associated with participation.

Are there any benefits associated with taking part?

No

What will happen to the results of this study?

The results of this study may be summarised in published articles, reports and presentations. Quotes or key findings will be anonymized: We will remove any information that could, in our assessment, allow anyone to identify you. With your consent, information can also be used for future research. Your data may be archived for a maximum of 4 years. All potentially identifiable data will be deleted within this timeframe if it has not already been deleted as part of anonymization.

Data protection and confidentiality.

Your data will be processed in accordance with Data Protection Law. All information collected about you will be kept strictly confidential. Your data will be referred to by a unique participant number rather than by name. Your data will only be viewed by the researcher/research team- Juekai Zhang, Markulf Kohlweiss .

All electronic data will be stored on a password-protected encrypted computer, on the School of Informatics' secure file servers, or on the University's secure encrypted cloud storage services (DataShare, ownCloud, or Sharepoint) and all paper records will be stored in a locked filing cabinet in the PI's office. Your consent information will be kept separately from your responses in order to minimise risk.

What are my data protection rights?

The University of Edinburgh is a Data Controller for the information you provide. You have the right to access information held about you. Your right of access can be exercised in accordance Data Protection Law. You also have other rights including rights of correction, erasure and objection. For more details, including the right to lodge a complaint with the Information Commissioner's Office, please visit www.ico.org.uk. Questions, comments and requests about your personal data can also be sent to the University Data Protection Officer at dpo@ed.ac.uk.

Who can I contact?

If you have any further questions about the study, please contact the lead researcher, Juekai Zhang, s2343556@ed.ac.uk.

If you wish to make a complaint about the study, please contact inf-ethics@inf.ed.ac.uk. When you contact us, please provide the study title and detail the nature of your complaint.

Updated information.

If the research project changes in any way, an updated Participant Information Sheet will be made available on http://web.inf.ed.ac.uk/infweb/research/study-updates.

Alternative formats.

To request this document in an alternative format, such as large print or on coloured paper, please contact Juekai Zhang, s2343556@ed.ac.uk .

General information.

For general information about how we use your data, go to: edin.ac/privacy-research

C.2 Participants' consent form

Participant number:_____

Participant Consent Form

Project title:	Visualization of Zero-Knowledge proofs that A Child can Understand
Principal investigator (PI):	Markulf Kohlweiss
Researcher:	Juekai Zhang s2343556
PI contact details:	

By participating in the study you agree that:

- I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.
- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

1. I allow my data to be used in future ethically approved research.

Yes	**No**

2. I agree to take part in this study.

Yes	**No**

Name of person giving consent	Date dd/mm/yy	Signature

Name of person taking consent	Date dd/mm/yy	Signature

THE UNIVERSITY of EDINBURGH informatics

C.3 Online research participant information sheet (PIS) and consent form

Participant Information Sheet

Project title:	Visualization of Zero-Knowledge proofs that A Child can Understand
Principal investigator:	Markulf Kohlweiss
Researcher collecting data:	Juekai Zhang s2343556
Funder (if applicable):	

This study was certified according to the Informatics Research Ethics Process, reference number 550871. Please take time to read the following information carefully. You should keep this page for your records.

Who are the researchers?

Juekai Zhang, Markulf Kohlweiss

What is the purpose of the study?

Zero-knowledge proofs refer to the ability of a prover to convince a verifier that a statement is correct without providing any additional information. This study pioneers the use of interactive visualisations to explain zero-knowledge proofs, using a web game to achieve the visualisation goal of making zero-knowledge proofs accessible to non-experts and even children. The study will collect your feedback on the game from concept understanding, user interface, user experience, fun level, etc.

Why have I been asked to take part?

You have been invited to take part in the study because you volunteered to contribute feedback to the game.

Do I have to take part?

No – participation in this study is entirely up to you. You can withdraw from the study at any time, without giving a reason. Your rights will not be affected. If you wish to withdraw, contact the PI. We will stop using your data in any publications or presentations submitted after you have withdrawn consent. However, we will keep copies of your original consent, and of your withdrawal request.

What will happen if I decide to take part?

- You will be first asked to play the zero-knowledge proofs game which is deployed on the internet.
- You will be asked questions related to the zero-knowledge proofs game, including concept understanding, user interface, user experience, fun level.
- The whole process takes less than 30 minutes (including the playing game time)
- Your action will not be recorded. Your answers will be anonymized.

Are there any risks associated with taking part?

There are no significant risks associated with participation.

Are there any benefits associated with taking part?

No.

What will happen to the results of this study?

The results of this study may be summarised in published articles, reports and presentations. Quotes or key findings will be anonymized: We will remove any information that could, in our assessment, allow anyone to identify you. With your consent, information can also be used for future research. Your data may be archived for a maximum of 4 years.

Data protection and confidentiality.

Your data will be processed in accordance with Data Protection Law. All information collected about you will be kept strictly confidential. Your data will be referred to by a unique participant number rather than by name. Your data will only be viewed by the researcher/research team: Juekai Zhang, Markulf Kohlweiss.

All electronic data will be stored on a password-protected encrypted computer, on the School of Informatics' secure file servers, or on the University's secure encrypted cloud storage services (DataShare, ownCloud, or Sharepoint) and all paper records will be stored in a locked filing cabinet in the PI's office. Your consent information will be kept separately from your responses in order to minimise risk.

What are my data protection rights?

The University of Edinburgh is a Data Controller for the information you provide. You have the right to access information held about you. Your right of access can be exercised in accordance Data Protection Law. You also have other rights including rights of correction, erasure and objection. For more details, including the right to lodge a complaint with the Information Commissioner's Office, please visit www.ico.org.uk. Questions, comments and requests about your personal data can also be sent to the University Data Protection Officer at dpo@ed.ac.uk.
For general information about how we use your data, go to: edin.ac/privacy-research

Who can I contact?

If you have any further questions about the study, please contact the lead researcher, Juekai Zhang, s2343556@ed.ac.uk.
If you wish to make a complaint about the study, please contact inf-ethics@inf.ed.ac.uk. When you contact us, please provide the study title and detail the nature of your complaint.

Updated information.

If the research project changes in any way, an updated Participant Information Sheet will be made available on http://web.inf.ed.ac.uk/infweb/research/study-updates.

Consent

By proceeding with the study, I agree to all of the following statements:

- I have read and understood the above information.
- I understand that my participation is voluntary, and I can withdraw at any time.
- I consent to my anonymised data being used in academic publications and presentations.
- I allow my data to be used in future ethically approved research.