Cyber Security Awareness Using Serious Card Game

Final Year Project
Report

Submitted by

Avrati Motwani- A038

Soham Nanavati- A039

Mayank Parvatia- A077

Under the Guidance Of

Prof. Pintu Shah

In fulfillment for the award of the degree of

B.TECH.

INFORMATION TECHNOLOGY

At

Department of Information Technology
Mukesh Patel School of Technology Management & Engineering
NMIMS (Deemed –to-be University)
JVPD Scheme Bhaktivedanta Swami Marg,
Ville Parle (W), Mumbai-400 056.

March, 2021

CERTIFICATE

This is to certify that the project entitled “Cyber Security Awareness Using Serious Card Game” is the bonafide work carried out by Avrati Motwani(A038), Soham Nanavati(A039), Mayank Parvatia(A077) of B.Tech (IT), MPSTME, Mumbai, during the VIII Semester of the academic year 2020-2021, in partial fulfillment of the requirements for the award of the degree of Bachelors of Technology as per norms prescribed by NMIMS. The project work has been assessed and found to be satisfactory.

(Signature of Internal Mentor 1)	(Signature of External Examiner)
Name: Prof. Pintu shah	*Name:*
Designation: Asst. Professor	*Designation:*

HOD (IT)	Dean
(Dr. Ketan Shah)	(Dr. Alka Mahajan)

DECLARATION

We, Avrati Motwani, Soham Nanavati, Mayank Parvatia, roll numbers: A038, A039, A077 respectively, understand that plagiarism is defined as any one or the combination of the following:

1. Uncredited verbatim copying of individual sentences, paragraphs or illustrations (such as graphs, diagrams, etc.) from any source, published or unpublished, including the Internet.
2. Uncredited improper paraphrasing of pages or paragraphs (changing a few words or phrases, or rearranging the original sentence order)
3. Credited verbatim copying of a major portion of a paper (or thesis chapter) without clear delineation of who did or wrote what. (Source: IEEE, The Institute, Dec. 2004)

We have made sure that all the ideas, expressions, graphs, diagrams, etc., that are not a result of our work, are properly credited. Long phrases or sentences that had to be used verbatim from published literature have been clearly identified using quotation marks.

We affirm that no portion of our work can be considered as plagiarism and we take full responsibility if such a complaint occurs. We understand fully well that the guide of the seminar/project report may not be in a position to check for the possibility of such incidents of plagiarism in this body of work.

Signature:
Name: Avrati Motwani
Roll No.: A038
Date:

Signature:
Name: Soham Nanavati
Roll No.: A039
Date:

Signature:
Name: Mayank Parvatia
Roll No.: A077
Date:

ACKNOWLEDGEMENTS

Apart from the efforts of the project team, the success of this project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We would like to show our gratitude and greatest appreciation to Prof. Pintu Shah, we can’t say enough thanks for his tremendous support and help. We feel motivated and encouraged every time we attend a particular meeting. Without his guidance and encouragement this project would not have been materialized.

We would also like to thank our college Mukesh Patel School of Technology Management & Engineering, our HOD Dr. Ketan Shah and the faculty members of the Information Technology department for their support.

Table of Contents

List of Figures

List of Tables

Abstract

With an increasing use of technology, there has been a significant increase in the number of cyber-attacks, including financial fraud, identity theft and denial of service attacks just to name a few. Cybersecurity has become a fundamental problem and a challenge. The sophistication and frequency of cyber-attacks and the magnitude of impact they cause has become a major concern in recent times.

The most critical vulnerability that the hackers exploit is the lack of awareness amongst the users. In today’s world where possibly everything takes place online, not many people pay attention to the possible risks that exist online. The security factor is often sidelined. People are not equipped with sufficient knowledge on cybersecurity attacks, and the steps that they should take to ensure security from their end. This lack of knowledge is one of the main reasons behind the increasing cybersecurity attacks. Consequently, the need for enhanced awareness about cybersecurity among the general public has also increased.

Current training programs are not effective in attaining the intended learning objectives because of rote content and traditional delivery methods like presentation and Computer Based Test (CBT).In today’s digital age, where most of the people spend their time online, we need to come up with better mechanisms to stimulate the public’s interest in the field of cyber security. This is where serious games play an important role.

Serious games are basically games with purposes beyond pure entertainment and have gained a lot of popularity in recent times. These games have learning as an outcome too. It merges the fun element along with knowledge to make learning less monotonous, and more interactive. Learning in an interactive, fun game environment will stimulate the interest of the users and allow them to learn better and faster.

Therefore, in this report we are proposing a new Cyber Security Card Game. This game aims at raising awareness about cyber security, it’s attacks and defenses amongst people through an interactive card game. With the immense increase in the number of cyber security attacks, this game will equip the users with sufficient knowledge about the steps that they should take and follow so as to ensure security at their end and prevent themselves from these cyber-attacks.

This game consists of two levels. Level 1 deals with getting the users equipped with knowledge about various cyber security attacks and defenses. The Level 2 is an interactive multiplayer 1v1 attack-defense game where the users join a room and play the attack and defense cards against each other.

We also conducted a user study to test the response of the users. The game was proven to be quite effective and was found interesting by the users. The results of the survey have been mentioned in Chapter 7.

Chapter 1: Overview

Cybersecurity is concerned with applying security measures to ensure confidentiality, integrity and availability of digital data that is either stored, sent or received. Security measures can range from protection software to risk assessment or even awareness training, all of which aim to prevent security violations such as loss or theft of data, or damage to a system.

The education of the population represents a cornerstone of the system as citizens constitute both the basis of all actors involved in the development of cyber capabilities (cyber experts, specialized industries, military organizations); and the target of various cyber threats, which needs to be protected and defended (businesses, governments, critical infrastructures, general population). A more educated nation provides better prepared individuals for organizations, but also hinders the spread of cybercrimes.

1.1 Project specification

This project aims to educate the user on the measures one should take in the practical world to stay safe from various cyberthreats. Hence the game is divided into 2 levels. For the first level, the user plays a solitaire-like card game against the system so that he/she gets accustomed to the different cyberthreats and what defense measures are taken against them. The user will be tracked on the number of points. After reaching a benchmark number of points, the user gets access to the Level 2. Here, considering that the user is familiar with the concept of the game, plays an Attack-Defense game with another player in real time where they get assigned roles of an Attacker or a Defender on turn-by-turn basis. So essentially, this level is a test of knowledge for the user.

1.2 Literature Survey

1.2.1 Enhancing cyber security awareness with mobile games [1]

This research paper presents a methodology that employs gameplay methods over traditional approaches to make the awareness process more engaging. It proposes two games that aim towards achieving this goal.

a. Password Protector

It aims at training the users to create complex, and strong passwords which are memorable as well. It will be an important learning aspect for the user and they might be able to use this game training to implement in real life.

This game is designed with an aim to equip the players with password creation skills by training them on the aspects of what constitutes a strong password. The players practice strong password creation in a competitive context. The players are provided with a limited set of characters from which they are required to create the best possible strong, memorable passwords. The standard password strength principles of long passwords with a diverse set of characters are used as guidelines to define the strength of the password. The game is timed and the passwords created are rated via a password meter. Further, to test whether the password is memorable or not, the players are required to repeat the password.

b. Malware Guardian

This game aims at training the players with skills to understand and identify malicious software and creates awareness about the importance of using up-to-date malware protection systems and taking backups as a safety measure.

In the game, the players are required to defend their system by roleplaying as an anti-malware package that scans files as it approaches a computer device. The player blocks the files that are malicious before it reaches the computer device. It includes different types of malware such as viruses, trojans, spyware, ransomware that can infect the computer devices in different ways. Only clean and uninfected files must reach the computer device. The player tasks are also combined in parallel with the task of updating the malware software and taking regular backups of the device.

1.2.2 Game based learning approach to cyber security [2]

In this paper, the author proposes a popular and successful option for gamification which is competition based on catching the flag, also known as Capture The Flag (CTF). Participants can put into practice the knowledge acquired by studying, reading and viewing tutorials. CTF competitions are becoming of great interest in the context of cybersecurity, since a set of challenges are presented to the user, including several clues. A final score is achieved depending on the number of employed clues.

In the first dynamic, which is trivia-based, players should answer open questions. The second, more common, implies searching for a textual chain through an exercise that combines various areas such as cryptography, forensic analysis, reverse engineering, etc. The third type of challenge, more complete although difficult to orchestrate, is known as the attacker/defender game. In this modality, each group of players is assigned a software system, usually a virtual machine, that must defend against the other participants of the competition while attacking the machines of other competitors. In these scenarios, almost every type of attack is allowed, except for denial-of-service attacks.

The different types of challenges in this CTF proposal are: -

- *Quiz*: This challenge is a question with only one true answer. Answers can be anything, from a single word to a whole sentence, not being case sensitive.
- *Flag*: This type of challenge is composed of a problem that users must solve to obtain a particular string, the flag, that they can submit to get points.

These challenges are classified into the following categories: Cryptography, Digital Forensics, Reverse Engineering, Web Exploitation, Binary Exploitation and Quiz.

In this work, a set of CTF challenges have been defined and included in a CTF environment in order to study its suitability for the learning design of practical activities on distance. This environment has been integrated in a Cybersecurity subject. For this reason, the CTF experience has been linked with its contents, learning outcomes, objectives and topics of the subject.

1.2.3 Cyberspace Odyssey: A Competitive Team-Oriented Serious Game in Computer Networking [3]

Cyberspace Odyssey (CSO) is a serious game supporting networking education by engaging students in a race to successfully perform various cyber security tasks in order to collect clues and solve a puzzle in a virtual near-Earth 3d space. In this game, each team uses computer and network hacking skills as well as critical thinking to solve logical puzzles.

The gameplay involves a host institution that has received intelligence that a group of hackers/terrorists seek to hijack the space station and breach the host institution’s security. There is a culprit/spy working at one of the five space stations who is in charge of hacking. The other agents have to identify the spy and solve the mystery by collecting enough evidence for conviction. The clues for a simple logical puzzle that has been designed such that each clue contains the list of potential suspects and all clues are necessary to list down to a single suspect. If the team has obtained all the clues and their accusation is correct, then the team completes the game objectives.

The authors hosted several sessions where the students were asked to play the game. During the session, they observed that the students appeared to be motivated and focused on succeeding and even enjoyed playing the game. They also performed a quantitative analysis of the collected data that indicated that the game was effective as a teaching tool. Also, the question-by-question analysis showed an improvement on the application level questions than on the knowledge and comprehension level questions, indicating an increase in the practical knowledge.

1.2.4 Improving software security awareness using a serious game [4]

In this paper, the authors have proposed a serious game to improve the software security awareness amongst users.

In this game, a game map is used to provide context of the software system. The game map includes the physical layout of the system components, and the operators of the system functionalities. Based on the information provided in the game map, the players are asked to play the role of an organization security protection team. The job of the security protection teams to identify the potential risks, including cyber security threats, human vulnerabilities, and possible countermeasures.

They have based their game around a card game to minimize communication barriers between the team players as well as the competing teams. Usually, the teams are divided into a group of 3 to 4. Every team member represents at least one particular type of character roles such as Social Engineer Attacker, Network Attacker, etc. The player has to plan the attack, choose a target and plan the path that will be taken for reaching the target. The player has to save himself from the police while moving through the path. They also have to solve a puzzle in order to get access to the target room. Next, the player has to select one of the assets randomly which he/she thinks of as a possible spy. If the selection is correct, the card will give the player the direction towards the infected device. Then, the player has to move to the infected device room and suggest ways to compromise the infected device. After getting sufficient information such as vulnerabilities, etc., the attacker has to suggest the most feasible attack from the attack cards and then write a hypothetical scenario for the attack.

This game basically allows the players to learn while thinking from an attacker’s point of view.

1.2.5 Games currently in the market

Along with this, we also had a look at some games that already exist with the same motive. However, they suffer from some certain limitations. Our goal is to address these gaps in our game.

Game	Details	Gap Addressed	Target Audience
Elevation of Privilege	Card game to facilitate identification of attacks among users	Our game not only allows the players to identify the different threats, but also allows them to identify which preventive measures should be taken against the attack	Development groups only
Beckers and Pape serious game	Players learn about the attack and defense strategies related to human behaviors	Our game consists of social engineering attacks and allows players to learn about a larger variety of attacks	Security engineers and IT administrators
Persuaded	Allows players to learn the effectiveness of defense controls against common social engineering attacks. But it does not raise the awareness of actual attack vectors that the attackers can exploit	Our game not only allows the players to learn the different defense controls, but also raises awareness about the different attacks	Company Employees
[d0x3d!]	Designed to teach network security concepts to students. The players impersonate white-hat hackers who exploit network vulnerabilities.	Our game allows the player to learn a wider range of security threats, other than just network security threats	Students only
WebMe	It is a quiz-based game that only educates about the attacks.	Our game teaches the user about the defense measures that have to to be taken against the attacks as well.	Every internet user
Enter	Puzzle based game on how to attack a system and steal the data	Our game teaches about the different defense mechanisms as well.	Students/ naive internet users

Table I : Games currently in the market

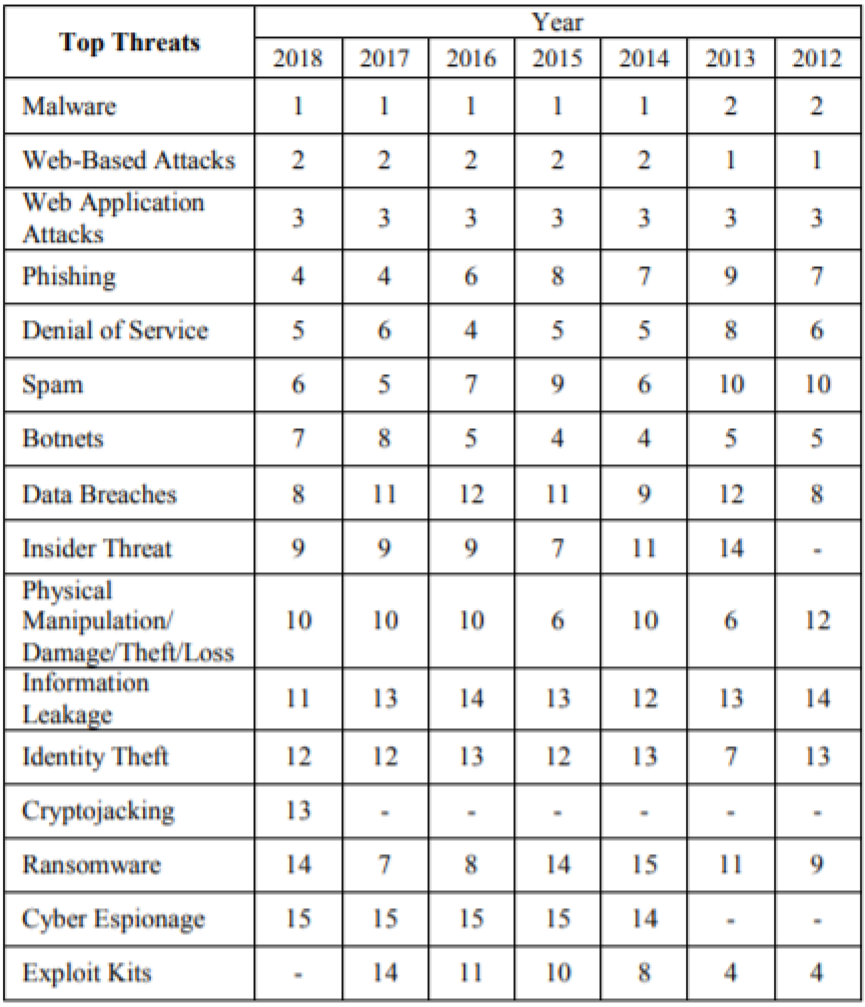
From the above table we can see that the reported games either educate people on a limited category of threats, or they allow the players to either think like attackers or to defend against attacks, but not both. Also, most of the games are quiz based and hampers collaborative learning. These gaps are addressed in our game, which covers the majority of the threats and allows the players to learn about them. It also allows the player to think from the mindset of an attacker as well a defender to gain deeper insights into the attack and defense strategies, and what goes on behind them.

What stands out in our game: -

- Serious game implementation to make learning fun.
- Players can relate the situations in game to real life.
- Covers major scenarios instead of focusing on one or a few such as Phishing and its types, Malwares, Web based attacks, Denial of Service, Ransomwares, Identity thefts, Physical manipulation, Password attacks and Man-in-the-middle attacks.
- Covers both attack and defense strategies and does not focus on either one of these.
- Players can think from the mindset of an attacker as well as a defender which helps the users to learn more.
- No prerequisite knowledge is required, the player will be able to learn everything through the game itself.
- Players can play 1v1 in a room by inviting using the room name.
- Players have a role of attacker as well as a defender point of view with a given asset for e.g., Computer, Mobile, Wi-Fi.

1.2.6 On the top threats to Cyber systems [5]

In this paper the author proposes the annual change in ranking of the top fifteen threats according to ETL from 2012 to 2018. The European Union Agency for Cybersecurity (ENISA) is the Union's agency dedicated to achieving a high common level of cybersecurity across Europe. ENISA contributes to EU cyber policy, enhances the trustworthiness of ICT products, services and processes with cybersecurity certification schemes, cooperates with Member States and EU bodies, and helps Europe prepare for the cyber challenges of tomorrow. An analysis of the top threats shows an increase in the incidences of attack and attack tactics as well as advancements in defense. Ransomware attacks were a dominant threat. There was a massive increase in phishing attacks. In the sequel we describe each of these cyber threats in order of their ranking in 2018



Top Threats	Year						
	2018	2017	2016	2015	2014	2013	2012
Malware	1	1	1	1	1	2	2
Web-Based Attacks	2	2	2	2	2	1	1
Web Application Attacks	3	3	3	3	3	3	3
Phishing	4	4	6	8	7	9	7
Denial of Service	5	6	4	5	5	8	6
Spam	6	5	7	9	6	10	10
Botnets	7	8	5	4	4	5	5
Data Breaches	8	11	12	11	9	12	8
Insider Threat	9	9	9	7	11	14	-
Physical Manipulation/ Damage/Theft/Loss	10	10	10	6	10	6	12
Information Leakage	11	13	14	13	12	13	14
Identity Theft	12	12	13	12	13	7	13
Cryptojacking	13	-	-	-	-	-	-
Ransomware	14	7	8	14	15	11	9
Cyber Espionage	15	15	15	15	14	-	-
Exploit Kits	-	14	11	10	8	4	4

Table II Top Cybersecurity Threats

I. Malware

Malware is software with a malicious intent to destroy a computer, server, or network. It causes harm or acts against the interests of the user. Common malware are viruses, worms, Trojan horses, spyware, and ransomware. Mitigation of malware includes malware detection on all in-bound and outbound channels and sufficient security policies for response.

II. Web-Based Attacks

Web-Based Attacks make use of web-enabled systems such as browsers, webpages, and content managers. They are the most common threat for financial attacks. As a widely used content manager, WordPress is particularly vulnerable. Drive-by download attacks involve malicious JavaScript and do not require action from the user. Malicious URLs use Blackhat Search Engine Optimization (SEO) to attract targets. Mitigation includes patching vulnerabilities and web traffic filtering.

III. Web Application

Attacks Web application attacks take advantage of Application Programming Interfaces (APIs) which are exposed and open. Government and financial institution apps are particularly popular targets. SQL injection can be used to retrieve passwords stored in databases. Mitigation includes policies for secure app development and for the authentication and validation of mechanisms.

IV. Phishing

Phishing uses social engineering to lure targets into revealing sensitive information. Spearfishing targets people within a specific organization. Often disguised as legitimate organizations, one million new phishing websites are created each month. In fact, in 2018 over 90% of malware infections and 72% of data breaches in organizations originated from Phishing Attacks.

V. Denial of Service Attacks

Denial of Service (DoS) attacks occur when machine or network resources are made unavailable to their intended users by disrupting service, usually by flooding the network with requests, often from botnets. Distributed Denial of Service (DDoS) attacks strike a target from many sources and are harder to stop. Pulse wave DDoS attacks come in short bursts on multiple targets and can last for days. In 2017, the Mirai Internet of Things (IoT) botnet was responsible for the largest DoS attack in history. Mitigation includes a reaction plan, Internet Service Providers (ISPs) with DoS protection, firewalls and access control lists.

VI. Physical Damage and Loss

Physical damage and loss of computers and equipment can result in data breaches. Encryption would solve the data breach problem, but only 43% of reporting organizations use encryption in 2018. Additional examples of physical threat include Automatic Teller Machine (ATM) drilling in which a drilled hole near the PIN pad allows for wired access and control of the machine. In addition to encryption, mitigation includes asset inventory.

VII. Information Leakage

Information Leakage is a breach caused not by a direct attack but from unsecured data. Mobile devices make such breaches easier. Information leakage is usually the result of human error, often an insider action or failure. However, information can also be leaked through coding errors, particularly on mobile devices. In 2018, information collected by the mobile fitness tracking and sharing app Strava has highlighted the locations of secret US military bases worldwide.

VIII. Identity Theft

Identity Theft is a cyber threat in which the attacker obtains information about a person or computer system for the purpose of impersonating the target. In the UK, identities were stolen at the rate of 500 per day in 2017. Top threats include skimmers on credit card devices, dumpster diving for hard copy personal information, phishing, hacking, and telephone impersonators. Mitigation includes protection of documents, strong privacy settings on social media, password protection on devices, and care when using public Wi-Fi.

IX. Ransomware

Ransomware is Malware that encrypts files or locks down a system until the target pays the actor to remove the restrictions. In the case of wipe ware, the encryption is never removed. In the latter, a computer is invaded in a way similar to ransomware, but instead of demanding a ransom, a malicious software is installed to start cryptocurrency mining without the computer owner's noticing. Mitigation is through limited access rights to data which potentially makes fewer data vulnerable to encryption, and an off-line backup to recover data as well as an up-to-date and patched software and operating system.

To give an overview, the scenarios in our game will be based on the cyber security threats that common people are most likely to be vulnerable to. The scenarios will equip users with knowledge about threats and their defenses such as Phishing and its types, Malwares, Web based attacks, Denial of Service, Ransomwares, Identity thefts, Physical manipulation, Password attacks and Man-in-the-middle attacks. Our game will provide the users with knowledge related to these attacks and will allow them to learn the steps to be taken to defend themselves against these attacks.

Over the years there has been a significant increase in the number of cyber security attacks and the only way to mitigate them is by creating awareness about it amongst the users. This would equip the users with the safety measures that they should take in order to ensure all security and defense at their end.

Chapter 2: Analysis and Design

This section will give a high-level overview into our game and will help us gain some insight into the game, it's features and it's working. The main purpose of this section is to describe the system's capabilities and functions that it will ultimately perform. This section also consists of a use case diagram- to help understand the users and how they interact with the system, an activity diagram- to help understand the workflow of the system, frontend design- to give an overview of user interface and the backend design- to give details about the backend schema.

2.1 Requirement Analysis

2.1.1 External Interface Requirements

User Interfaces:

- User interface of this program is the common Windows/Android interface, nothing additional is required.
- Buttons and functions include: Play a game, Level 1, Level 2, Settings, Quit, Back to Menu, Manual, Help, etc.

Hardware Interfaces:

- This game works on Windows pc, Android, mobile devices and tablets. No other hardware is required.

Software Interfaces:

- Our game will only need an Android version 4.0 or higher/ i3 processor or higher in order to perform.

2.1.2 Other Nonfunctional Requirements

Performance Requirements:

- The game should be available at all times at all hours.
- Starting the game should not take more than 10 seconds. There should be less wait time.
- Proper coordination should exist amongst players in the same game room.
- The response time of the game should be as low as possible.
- There should be no scrolling jerks when the user navigates through the app.

Safety Requirements:

In case of app-related critical incidents, the user should be aware of the availability of support personnel or online help.

Security Requirements:

- All data, especially the login credentials, etc., should be confidential whether it is at rest, in-transit or being processed.
- User authentication is carried out using the login/signup methodologies.
- Ensure that critical information such as user credentials are encrypted to prevent any data leakage.

Implement data portability - allow user to sign-in through different platforms like Google.

Other Requirements

Privacy and end user policies

The application will consist of a privacy policy which aims to protect the data of the consumer and an End User policy which states that the user must agree to the terms and conditions of the developers.

Database

The application will require a database for e.g., Firebase database to collect and keep the data of the players.

2.2 Feasibility Study

Gaming is one of the upcoming fields which has been used to create awareness about daily life threats, cyber threats etc. rather than educating in a more formal mode like texts, speech etc. Card games have been popular since many years and almost everyone is familiar with the concept, so using that concept the main goal is to create awareness about the cyber threats among the masses in a fun intuitive way using the serious card game. The threats like drive-by download, phishing, vishing, man in the middle, network attack, clickjacking, malware, theft, dos, ransomware, smishing etc. The technology of the platform Unity3D is freely available in the market for students so there is no money constraint because some of the assets are free for student accounts. This makes the project financially feasible. Also, the game is socially acceptable as the users enjoyed the game. Level 1 beta version of the game was released and circulated which was played by many users, a survey was conducted with some questions and based upon that ratings were given according to the gameplay.

2.3 Design Development

2.3.1 UML Use Case Diagram

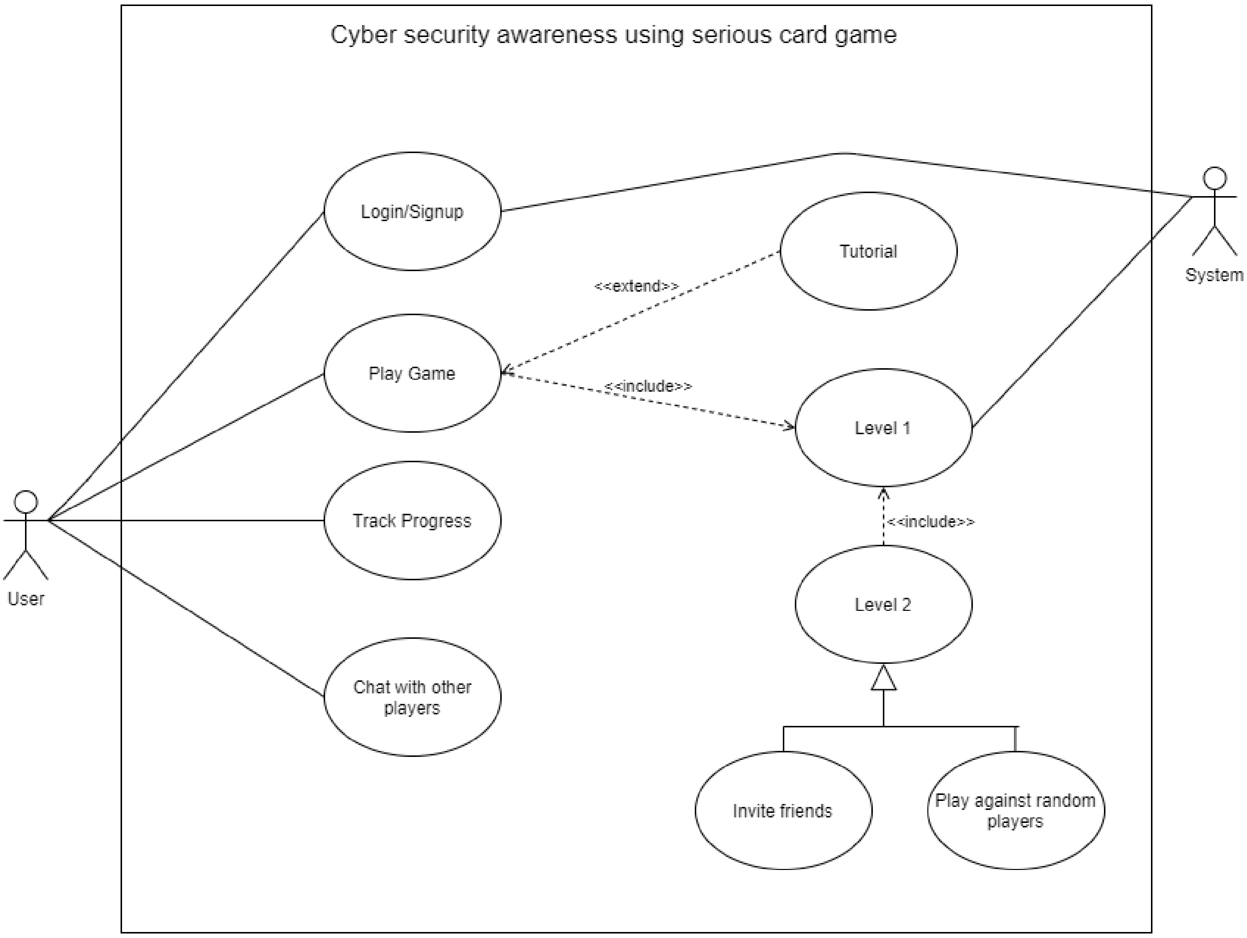


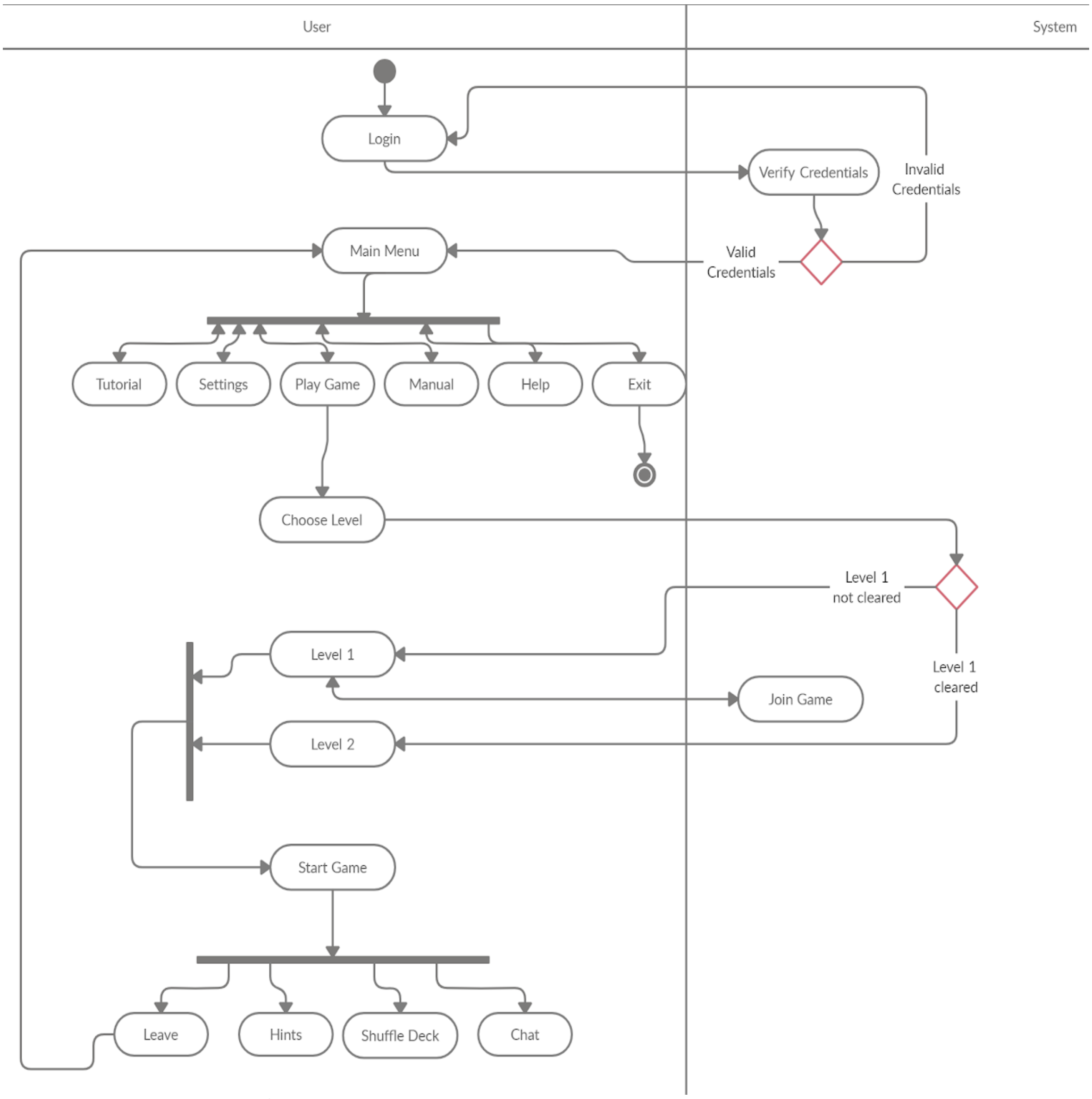
Figure 2. 1 UML Use Case Diagram

In the Use Case diagram, we have two actors: the user and the system. The user will first have to login/signup in which the system will play the role of validating/storing the credentials in the database.

The user will then be able to play a game. If the user is not accustomed to the game, he/she can also opt for a tutorial. He would then have to play Level 1 first. After a certain amount of progress has been achieved by the user in Level 1, Level 2 will then be unlocked and the user will be allowed to play Level 2 also. In Level 2, the user will be able to invite his/her friends to the room and will be able to play online against other random players too.

Other than this, the user will also be able to track his/her performance, and chat with other players.

2.3.2 Activity Diagram



i

gure 2. 2 UML Activity Diagram

This activity diagram focuses on the "Play Game" scenario and is therefore made around it only.

Once the activity starts, the user will be asked to login. The system will then validate the credentials. If the credentials are invalid, the user will be redirected back to the login page, otherwise, he would be redirected to the main menu.

The main menu will have options for:

- *Tutorial:* Wherein the user will be given a tutorial of the game and how it works.
- *Settings:* Wherein the user will be able to set his/her preferences.

- *Play Game:* Wherein the user will enter the game room to play the game.
- *Manual:* This would equip the user with basic cyber security terminologies that he can refer to in case he is stuck somewhere or is confused.
- *Help:* This would allow the user to contact us in case he needs some help regarding the app or the platform.
- *Exit:* The user will exit the app.

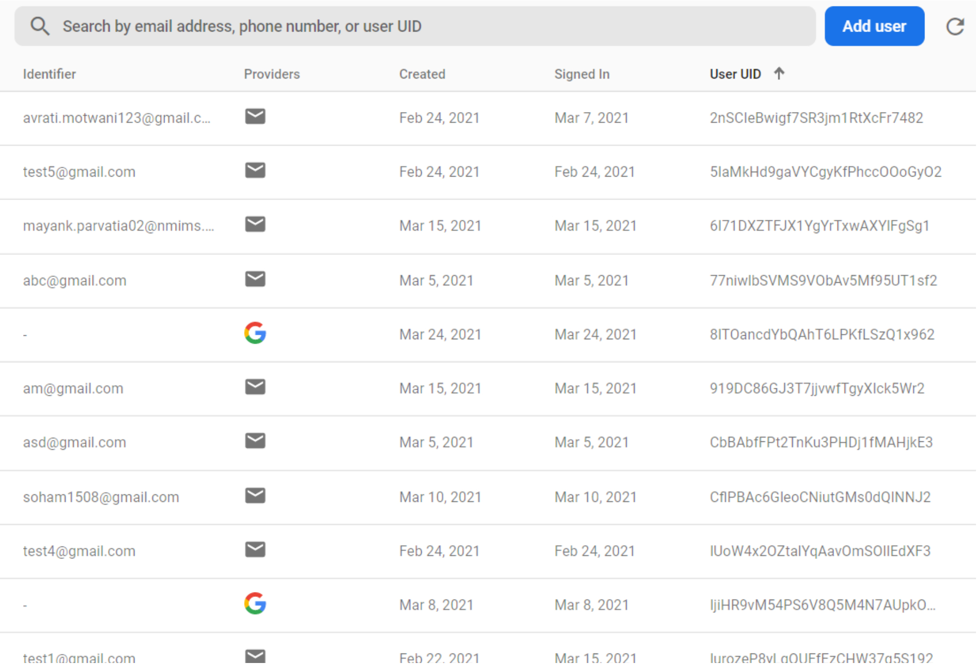
On clicking the “Play Game” option, the user will be able to choose the level he wants to play. If he or she has not fulfilled the requirements needed to clear Level 1, he will have the option to play Level 1 only. In Level 1, the system will join as a computer player and the user will play against the computer. If he has cleared Level 1, only then he will have the option to play Level 2.

After starting the game, the player will have the option to use his/her hints, chat with other players, shuffle the cards or leave the game.

If the user decides to leave the game, he/she will be redirected back to the Main Menu.

2.3.3 Database Design

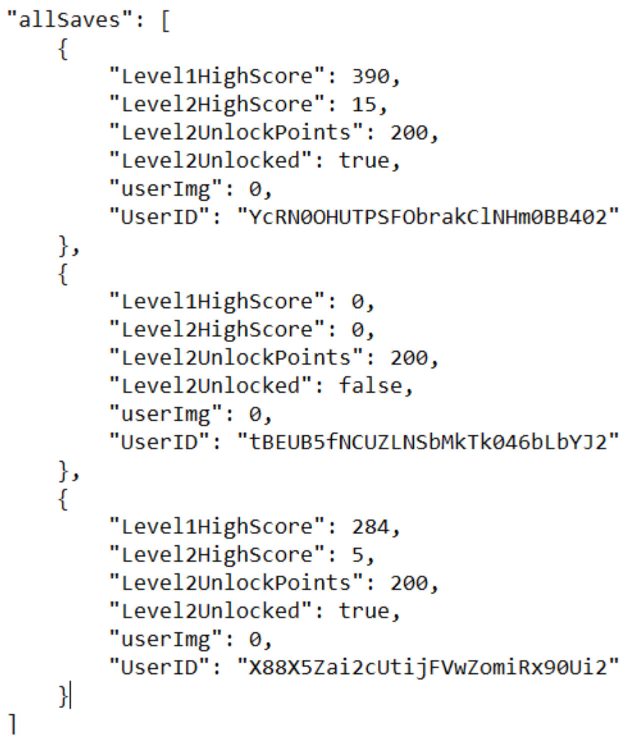
- For the database in this project, we have used Google’s firebase to store the user login details and authenticate their credentials.



Identifier	Providers	Created	Signed In	User UID ↑
avrati.motwani123@gmail.c...	✉	Feb 24, 2021	Mar 7, 2021	2nSCleBwigf7SR3jm1RtXcFr7482
test5@gmail.com	✉	Feb 24, 2021	Feb 24, 2021	5laMkHd9gaVYCgyKfPhcc0OoGyO2
mayank.parvatia02@nmims....	✉	Mar 15, 2021	Mar 15, 2021	6I71DXZTFJX1YgYrTxwAXYIFgSg1
abc@gmail.com	✉	Mar 5, 2021	Mar 5, 2021	77niwlbSVMS9VObAv5Mf95UT1sf2
-	G	Mar 24, 2021	Mar 24, 2021	8ITOancdYbQAhT6LPKfLSzQ1x962
am@gmail.com	✉	Mar 15, 2021	Mar 15, 2021	919DC86GJ3T7jjvwfTgyXlck5Wr2
asd@gmail.com	✉	Mar 5, 2021	Mar 5, 2021	CbBAbfFPt2TnKu3PHDj1fMAHjkE3
soham1508@gmail.com	✉	Mar 10, 2021	Mar 10, 2021	CflPBAc6GleoCNiutGMs0dQINNJ2
test4@gmail.com	✉	Feb 24, 2021	Feb 24, 2021	IUoW4x2OZtalYqAavOmSOIlEdXF3
-	G	Mar 8, 2021	Mar 8, 2021	IjiHR9vM54PS6V8Q5M4N7AUpkO...
test1@gmail.com	✉	Feb 22, 2021	Mar 15, 2021	lurozeP8vLgOUFfFzCHW37g5S192

Figure 2. 3 Google Firebase Database

- Other details such as scores, progress etc. is stored locally on the corresponding device in the form of a flat file.



```
"allSaves": [
    {
        "Level1HighScore": 390,
        "Level2HighScore": 15,
        "Level2UnlockPoints": 200,
        "Level2Unlocked": true,
        "userImg": 0,
        "UserID": "YcRN0OHUTPSFObrakClNHm0BB402"
    },
    {
        "Level1HighScore": 0,
        "Level2HighScore": 0,
        "Level2UnlockPoints": 200,
        "Level2Unlocked": false,
        "userImg": 0,
        "UserID": "tBEUB5fNCUZLNSbMkTk046bLbYJ2"
    },
    {
        "Level1HighScore": 284,
        "Level2HighScore": 5,
        "Level2UnlockPoints": 200,
        "Level2Unlocked": true,
        "userImg": 0,
        "UserID": "X88X5Zai2cUtijFVwZomiRx90Ui2"
    }
]
```

Figure 2. 4 Flat File Sample

2.4 Technology and Software Details

2.4.1 Software and Hardware

- Min. 2GB RAM and above.
- Storage: 500MB and above.
- Android 5.0 and above.
- Windows 8 and above.

2.4.2 Platform

- Unity3D
- Firebase

2.4.3 Languages

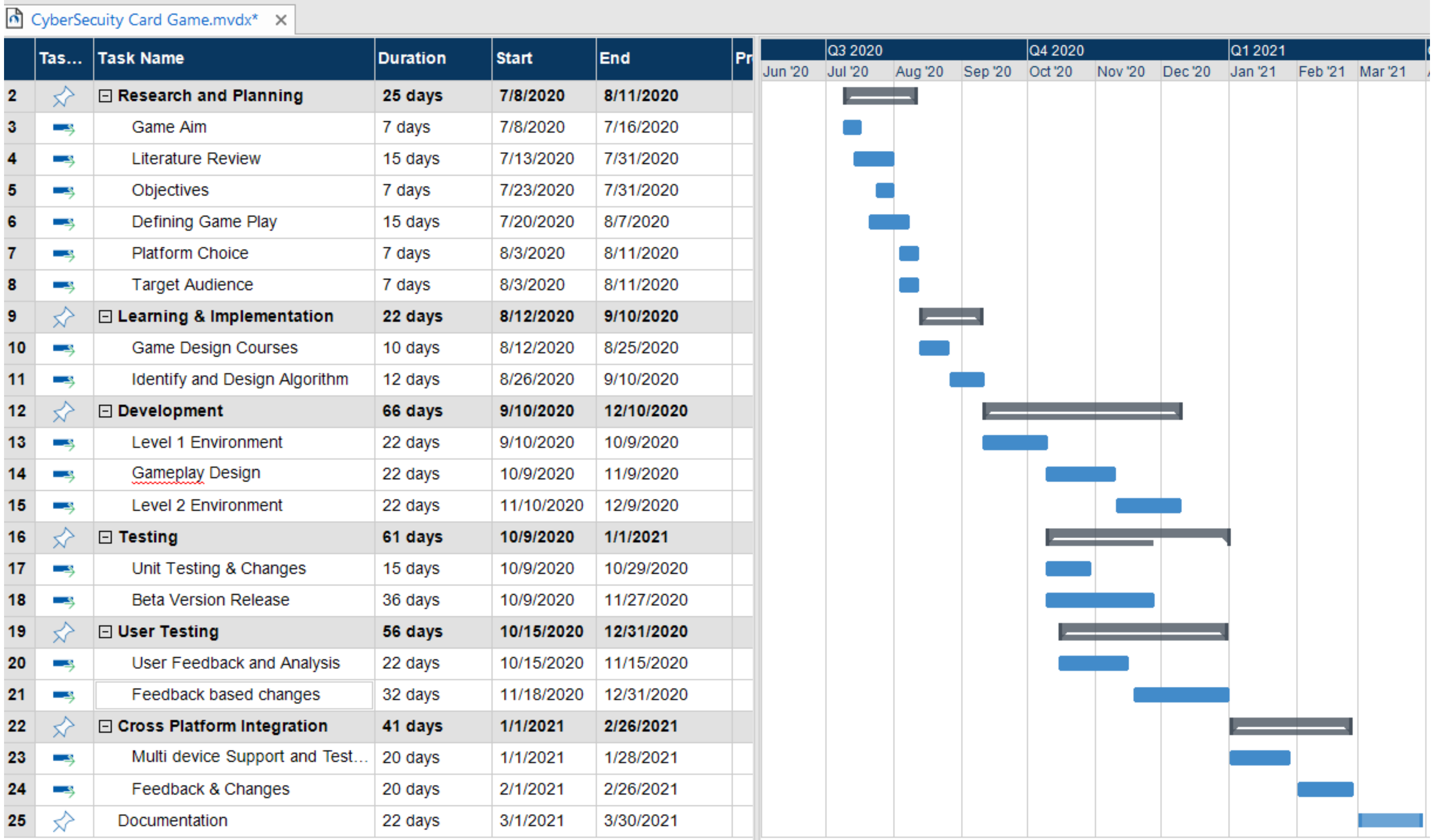
- C#
- NoSQL

2.4.4 Technical modules

- Google API: for login.
- Photon Unity Networking (PUN) asset for multiplayer
- Unity WebGL
    - Publish content as JS programs.
    - HTML5/JS, WebAssembly, WebGL rendering API, etc.
- Android Build Support Module
    - Android Environment Setup
    - To build and run for Android
    - Android SDK and NDK

2.5 Project Planning

2.5.1 Gantt Chart



	Tas…	Task Name	Duration	Start	End	Pr
2		**⊟ Research and Planning**	**25 days**	**7/8/2020**	**8/11/2020**	
3		Game Aim	7 days	7/8/2020	7/16/2020	
4		Literature Review	15 days	7/13/2020	7/31/2020	
5		Objectives	7 days	7/23/2020	7/31/2020	
6		Defining Game Play	15 days	7/20/2020	8/7/2020	
7		Platform Choice	7 days	8/3/2020	8/11/2020	
8		Target Audience	7 days	8/3/2020	8/11/2020	
9		**⊟ Learning & Implementation**	**22 days**	**8/12/2020**	**9/10/2020**	
10		Game Design Courses	10 days	8/12/2020	8/25/2020	
11		Identify and Design Algorithm	12 days	8/26/2020	9/10/2020	
12		**⊟ Development**	**66 days**	**9/10/2020**	**12/10/2020**	
13		Level 1 Environment	22 days	9/10/2020	10/9/2020	
14		Gameplay Design	22 days	10/9/2020	11/9/2020	
15		Level 2 Environment	22 days	11/10/2020	12/9/2020	
16		**⊟ Testing**	**61 days**	**10/9/2020**	**1/1/2021**	
17		Unit Testing & Changes	15 days	10/9/2020	10/29/2020	
18		Beta Version Release	36 days	10/9/2020	11/27/2020	
19		**⊟ User Testing**	**56 days**	**10/15/2020**	**12/31/2020**	
20		User Feedback and Analysis	22 days	10/15/2020	11/15/2020	
21		Feedback based changes	32 days	11/18/2020	12/31/2020	
22		**⊟ Cross Platform Integration**	**41 days**	**1/1/2021**	**2/26/2021**	
23		Multi device Support and Test…	20 days	1/1/2021	1/28/2021	
24		Feedback & Changes	20 days	2/1/2021	2/26/2021	
25		Documentation	22 days	3/1/2021	3/30/2021	

Figure 2. 5 Gantt Chart

2.5.2 Work Breakdown Structure (WBS)

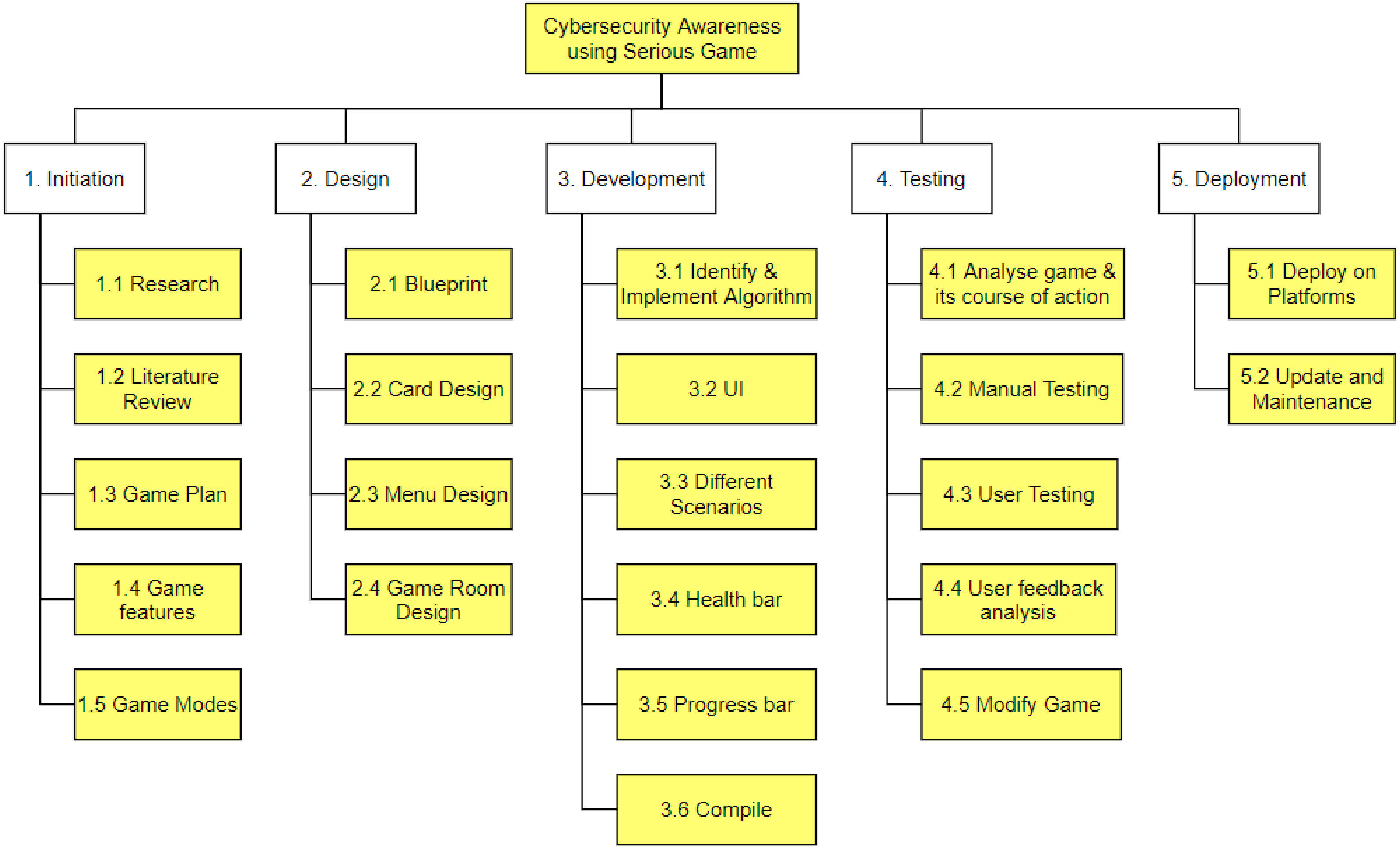


Figure 2. 6 Work Breakdown Structure

2.5.3 WBS Summary

WBS Code	WBS Name	WBS Description
1	Initiation	Defining requirements
1.1	Research	Study research papers
1.2	Literature Review	Comprehensive summary of previous research.
1.3	Game Plan	Deciding the foundation of the game
1.4	Game Features	Defining the features of the game
1.5	Game Modes	Defining variations in game play
2	Design	Transforming and implementing the requirements
2.1	Blueprint	Defining a high-level plan
2.2	Card Design	Designing the cards
2.3	Menu Design	Designing the main menu
2.4	Game Room Design	Designing the components inside the game room
3	Development	Programming and development of the game
3.1	Identify & Implement Algorithm	Defining and implementing the mechanics behind the game.
3.2	UI	Building the User Interface component
3.3	Different Scenarios	Development of Scenarios
3.4	Health bar	Development of the health bar for the players
3.5	Progress bar	Development of progress power to display the progress of players
3.6	Compile	Processing all the components together
4	Testing	Evaluation of software against requirements
4.1	Analyze game & its course of action	Determining whether the game works as planned
4.2	Manual Testing	Developers test the game among themselves
4.3	User Testing	Gain user feedback and insight into user experience
4.4	User Feedback Analysis	Analysis of the feedback given by the users
4.5	Modify Game	Make changes with respect to the feedback provided

5	Deployment	Process to make the game available to use
5.1	Deploy of Platforms	Make the game available on different platforms
5.2	Update & Maintenance	Provide updates and maintain the game’s performance

Table III : WBS Dictionary

Chapter 3: Project Description

This section includes in depth details about the project and its components. It will provide an insight into the various features of the game and what each component does. The game includes a number of functionalities that we will explore in this section.

3.1 Register/Sign up

This window is used to register the users on the platform. The registration window asks the users for a unique username, email, password, and confirm password. Implemented with firebase authentication, this section ensures that the user details are valid and in the correct format and stores the user details in the database, which the users can use to access the platform.

3.2 Login

This window allows the user to log into their accounts and approves the user by validating the details such as email and password from the database. Also implemented with firebase authentication, it ensures that the user details are correct, valid and in the required format.

3.2.1 Google Login

This feature allows the users to use their Google accounts to log into the game. The Google OAuth Login allows users to easily login to the platform with just the click of a button

3.2.2 Manual Login

Manual Login allows the users to log into the platform by manually entering their email and password that they had used at the time of registration. These details are then authenticated from the database and if they are correct, the user successfully logs into the application.

3.3 Main Menu

This window appears after the user successfully logs into the platform. It provides the users with plenty of options to explore the application. These options are described below.

3.3.1 User details

An icon with the user’s avatar along with the user’s username is present on the top left corner of the screen. On clicking that, the user can check their details such as their level 1 and level 2 high score and the points that the user needs to unlock level 2. It also allows the users to change their avatar in this window.

3.3.2 Play Level 1

This option redirects the user to the window where he/she can play level 1. It trains the users about the various cybersecurity threats and their defenses. More details about Level 1 will be discussed in the further sections.

3.3.3 Level 1 Tutorial

The user is provided with an option to view a tutorial for Level 1 to understand what the level is about. It would allow the users to get equipped with the gameplay which would eliminate any confusion that the user might have and allow them to play the game smoothly.

3.3.4 Play Level 2

This option is initially locked for the user as the user needs to score at least 200 points in Level 1 to unlock Level 2. This option redirects the user to the window where multiple players can create/join a room where they can play level 2.It allows the users to use the knowledge gained by them in Level 1 to use various attacks and defenses against each other. More details about Level 2 will be discussed in the further sections.

3.3.5 Level 2 Tutorial

This option is initially locked for the user as the user needs to score at least 200 points in Level 1 to unlock Level 2. After Level 2 is unlocked, the user is provided with an option to view the tutorial for Level 2 to understand what the level is about. It would allow the users to get equipped with the gameplay which would eliminate any confusion that the user might have and allow them to play the game smoothly.

3.3.6 Manual

This option redirects the user to the Manual, which is a scrollable document that gives the details into the various cybersecurity attacks and their defenses that we are covering in the game. This manual provides the users with detailed knowledge about the threats and defenses and would allow them to learn and gain knowledge on these topics.

3.3.7 Logout

This option is used when the user wishes to log out from their account. The user is then redirected to the login page where he/she can log into the application again.

3.3.8 Quit

A power off icon is present on the top right icon of the screen which allows the users to quit the application.

3.3.9 Settings

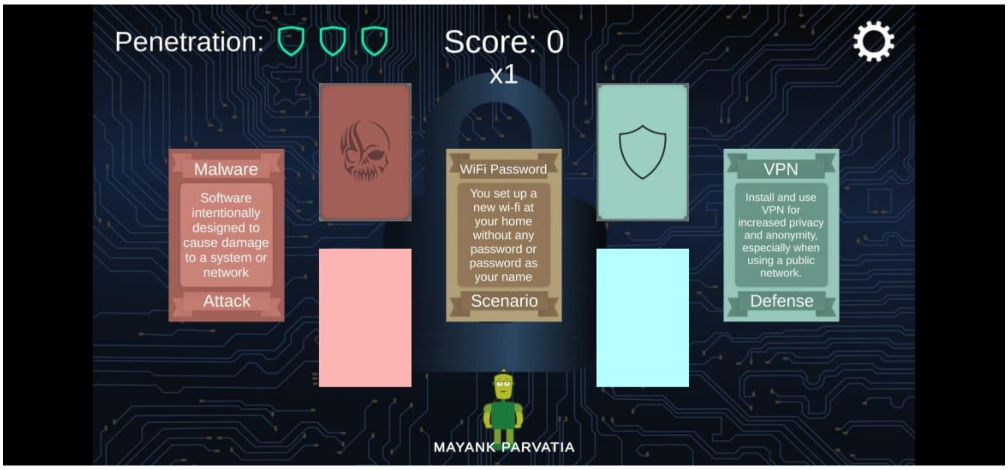
A settings icon is present on the top right icon of the screen which allows the users to change the volume settings of the application.

3.4 Level 1

Level 1 is activated for the user. The user needs to score at least 200 points to unlock Level 2. The main purpose of this level is to equip the users with knowledge about various cybersecurity attacks and their defenses. This level has a scenario card, an attack card and a defense card. The user needs to map the correct attack card and then defense card to any random scenario that appears on the screen in order to gain points. The various components in this level are: -

3.4.1 Sample Scenario Card

A scenario card is randomly displayed for the user which contains a particular context with respect to which the user needs to map the correct attack and defense card.



Malware

Software intentionally designed to cause damage to a system or network

Attack

WiFi Password

You set up a new wi-fi at your home without any password or password as your name

Scenario

VPN

Install and use VPN for increased privacy and anonymity, especially when using a public network.

Defense

Figure 3. 1 Sample Card

The red card with the skull icon indicates the Attack deck and the blue card with the shield icon indicates the Defense deck. The brown card in the center indicates the scenario card.

3.4.2 Attack Card

An attack card contains various attacks and their short descriptions. First, the user needs to choose a correct attack card with respect to the displayed scenario.

3.4.3 Attack Deck

An attack deck is given to the user which contains various attack cards. The user needs to find the correct attack card from the attack deck to map with the displayed scenario.

3.4.4 Defense Card

A defense card contains various defenses and their short descriptions. The user needs to choose the correct defense card, after he/she has chosen the correct attack card with respect to the displayed scenario.

3.4.5 Defense Deck

A defense deck is given to the user after the user chooses the correct attack card for the particular scenario. This deck consists of various defense cards from which the user needs to find the correct defense card to map with the respective scenario and attack cards.

3.4.6 Penetration

Penetration is essentially the number of attempts the user has left to choose the correct attack/defense card for a particular scenario. For each round, the user has three attempts. For every wrong attack/defense card chosen, the penetration value (i.e., the number of attempts) decreases by one. When one attempt or penetration value is left, the user is given an option to view the hint for the attack/defense card. After all the attempts have been consumed, the correct attack and defense card is displayed to the user and a new scenario is loaded.

3.4.7 Score

The score is essentially the number of points earned by the user while playing the level. Initially, the score is 0. For every correct attack/defense card, the user gains points (5*the score streak value) and for every incorrect attack/defense card, the user loses 2 points. The user needs to reach the score value of at least 200 to unlock level 2.

3.4.8 Scorestreak

The users are shown a scorestreak multiplier which determines the number of points a user will gain in case a correct attack/defense card is chosen. The scorestreak value is initially set to 1 and increases by one every time a correct attack/defense card is chosen. The scores gained by the users is equal to 5*the scorestreak value. If a wrong attack/defense card is chosen by the user, the scorestreak value resets to 1.

3.4.9 Hints

A question mark icon appears on the top right corner of the screen when the number of attempts left is equal to one. The user has the option to view the hint in order to determine the correct attack/defense card for the particular scenario.

3.4.10 Settings

A settings icon is displayed on the top right corner of the screen from which the user can change the volume settings according to his/her preference, view the tutorial to get an idea about the level gameplay, or go back to the main menu.

3.5 Level 2

Initially, Level 2 is locked for the user. This is because the user needs to learn and understand the basic cybersecurity attacks and defenses in order to play Level 2 and actually use those attacks and defenses against other players. Only after the user scores 200 points in Level 1, the Level 2 is unlocked. This level is a 1v1 multiplayer attack defense game wherein the users are provided with an asset, and they have to attack other players based on that asset and defend themselves based on the attack played by the other player. The various components in this level are: -

3.5.1 Create/Join a Room

This level involves a 1v1 multiplayer gameplay. This means, the users will have to either join a game created by some player, or create their own new game and invite someone to join the game. For this, the users will have to create or join a room. The user needs to enter a room name. If the room already exists with one player waiting, the user will join that game. If the room does not exist, the user can share the room name and will have to wait for another player to join the room.

3.5.2 Asset Card

The asset card consists of the asset related to which the users have to attack and defend against one another. Three assets have been defined in the game - Wi-Fi, PC and mobile phones. Each of these assets have their own attack and defense deck and the users will have to play the attack and defense cards related to the asset being displayed.

3.5.3 Attack Card

An attack card contains various attacks that can be played on the particular asset and their short descriptions. First, the user who has the role of attacking needs to play a particular attack. These roles of the users get switched after each round.

3.5.4 Attack Deck

The attack deck consists of all the attack cards related to the asset being displayed on the screen. The user who has the role of attacking can choose any of the cards from the deck to attack the other player.

3.5.5 Defense Card

A defense card contains various defenses related to the asset and their short descriptions. The user who has the role of defending needs to choose the correct defense card to defend themselves against the attack played by the other player.

3.5.6 Defense Deck

The defense deck is only activated after the attack has been played by the other user. The defense deck contains all the defense cards related to the asset being displayed. The user who has the role of defending needs to choose the correct defense from the deck depending on the attack played by the other user.

3.5.7 Score

The score is essentially the number of points earned by the user while playing the level. Each user has their own score. Initially, the score is 0. The score calculated for each player while playing the game are as follows:

- The attacker plays the attack card with respect to the asset. If the defender plays the correct defense card for that attack, the defender gains points (5*scorestreak value). No points are gained by the attacker.
- The attacker plays the attack card with respect to the asset. If the defender plays the wrong defense card for that attack, the attacker gains points (5*scorestreak value). No points are gained by the defender.

3.5.8 Scorestreak

Each user has a scorestreak multiplier which determines the number of points the user will gain. The points gained by the user is equal to 5*scorestreak value. The scorestreak value is initially set to 1 and increases as follows:

- The attacker plays the attack card with respect to the asset. If the defender plays the correct defense card for that attack, the defender's scorestreak increases by one.
- The attacker plays the attack card with respect to the asset. If the defender plays the wrong defense card for that attack, the attacker's scorestreak increases by one.

3.5.9 Health Bar

Each user has their own health bar. The game continues until a player's health bar reduces to zero. The other player subsequently wins. The health bar reduces as follows:

- The attacker plays the attack card with respect to the asset. If the defender plays the correct defense card for that attack, the attacker's health bar reduces.
- The attacker plays the attack card with respect to the asset. If the defender plays the wrong defense for that attack, the defender's health bar reduces.

3.5.10 Chat

A chat icon is displayed on the top right corner of the screen. The players can click on that icon to send messages and chat with each other.

3.5.11 Time

Each user is given a time of 30 seconds to play their respective attack/defense card. If the user isn't able to play his/her card within that time, he/she misses their turn and we move on to the next chance/round.

3.5.12 Scorestreak

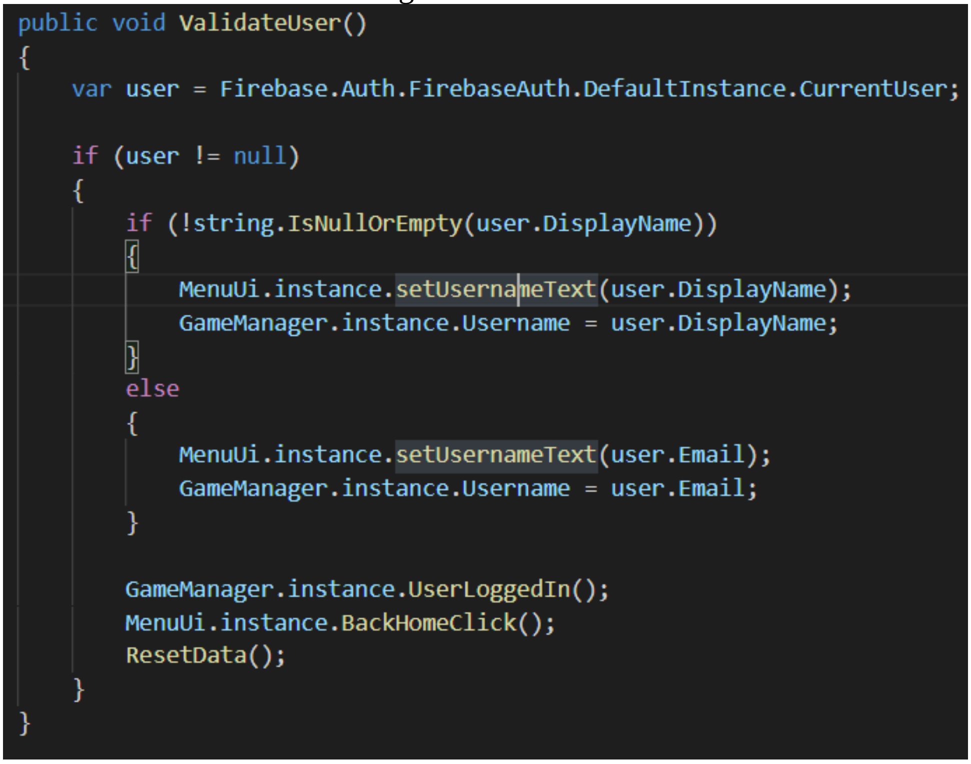
A settings icon is displayed on the top right corner of the screen from which the user can change the volume settings according to his/her preference, view the tutorial to get an idea about the level gameplay, or go back to the main menu.

Chapter 4: Project Implementation

4.1 Google Login

In the LoginPanel.cs script, we have used Google's Firebase to provide the ability for users to sign in with their Google accounts. Upon entering the credentials, data is updated on the firebase database as indicated by screenshots below.

The function ValidateUser () checks if the user is logged in via Firebase. If yes then it displays the name of the user that is registered with Firebase.



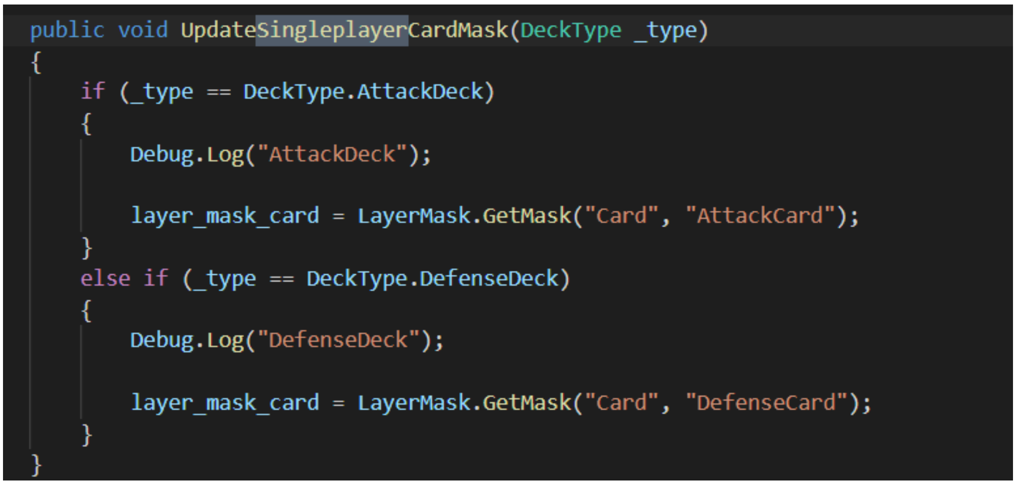
```
public void ValidateUser()
{
    var user = Firebase.Auth.FirebaseAuth.DefaultInstance.CurrentUser;

    if (user != null)
    {
        if (!string.IsNullOrEmpty(user.DisplayName))
        {
            MenuUi.instance.setUsernameText(user.DisplayName);
            GameManager.instance.Username = user.DisplayName;
        }
        else
        {
            MenuUi.instance.setUsernameText(user.Email);
            GameManager.instance.Username = user.Email;
        }

        GameManager.instance.UserLoggedIn();
        MenuUi.instance.BackHomeClick();
        ResetData();
    }
}
```

Figure 4.1 Validate User

4.2 Level 1:

4.2.1 Touch Handler to activate and deactivate card decks

In Level 1, we have two decks, Attack and Defense. For a particular scenario being displayed, we want the user to first choose the Attack Card for the particular scenario and then the Defense Card. To implement this, we have created a function named UpdateSingleplayerCardMask. In this function, while the DeckType activated is Attack, the user will only be able to control the Scenario Card (Card) and the Attack Card (AttackCard). Whereas, when the DeckType activated is Defense, the user will only be able to control the Scenario Card (Card) and the Defense Card (DefenseCard)



```
public void UpdateSingleplayerCardMask(DeckType _type)
{
    if (_type == DeckType.AttackDeck)
    {
        Debug.Log("AttackDeck");

        layer_mask_card = LayerMask.GetMask("Card", "AttackCard");
    }
    else if (_type == DeckType.DefenseDeck)
    {
        Debug.Log("DefenseDeck");

        layer_mask_card = LayerMask.GetMask("Card", "DefenseCard");
    }
}
```

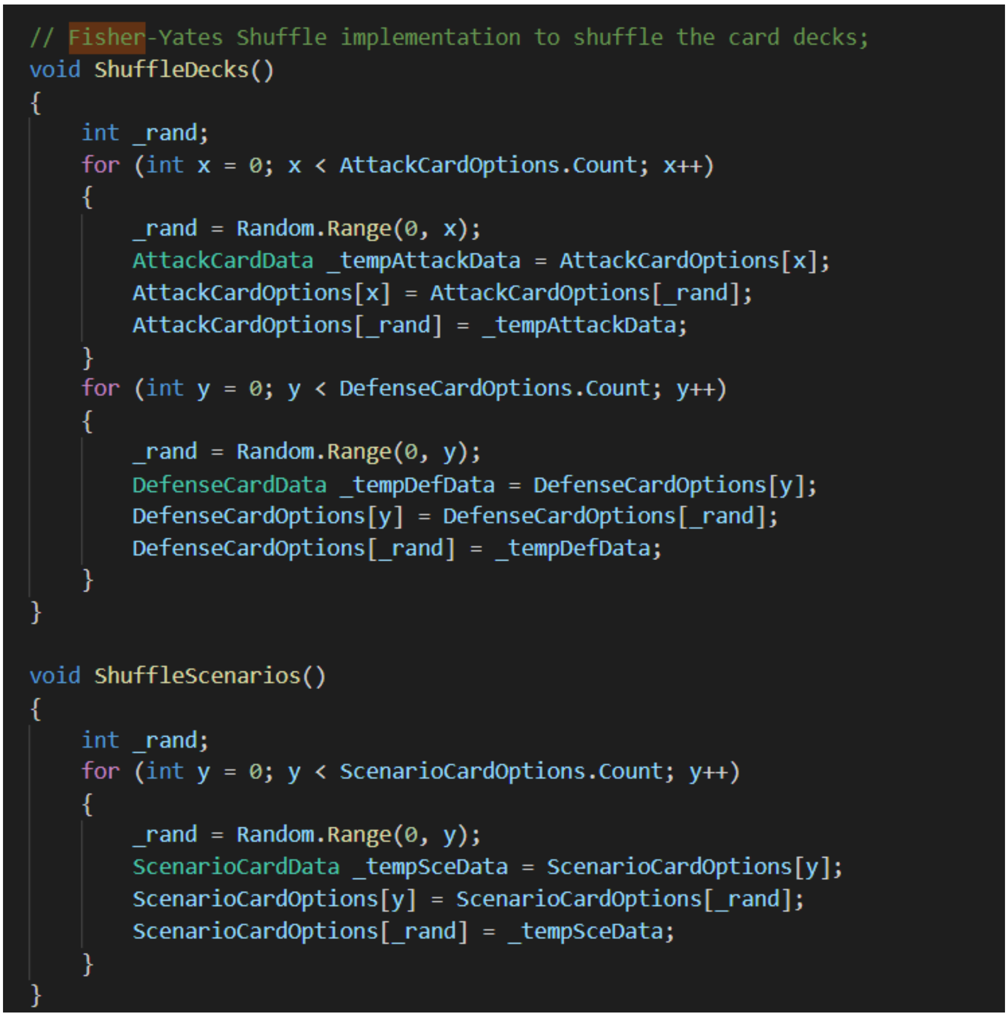
Figure 4.2 TouchHandler

4.2.2 Fisher-Yates shuffle implementation

In the script CardManager.cs, we have defined 2 functions, ShuffleDecks (), which shuffles the attack and defense cards, and ShuffleScenarios () which shuffles the scenario cards. For shuffling these decks, we have used, the Fisher Yates Shuffle Algorithm.

In this algorithm we have a for loop that continues until all cards have been shuffled. We have defined a variable _rand that generates a random number between 0 to x where x is the count of cards. We create a temporary variable (_tempAttackData) that stores the card data for the current card (AttackCardOptions[x]) whose position we will be shuffling. Then, in the position of the current card (AttackCardOptions[x]), we store the data of any random card from the array (AttackCardOptions[_rand]). Then, in place of the random card's position (AttackCardOptions[rand]), we store the data of the current card that we had copied in the temporary variable (_tempAttackData). This process continues for all cards in the deck.

Similarly, the Defense and Scenario cards are also shuffled.



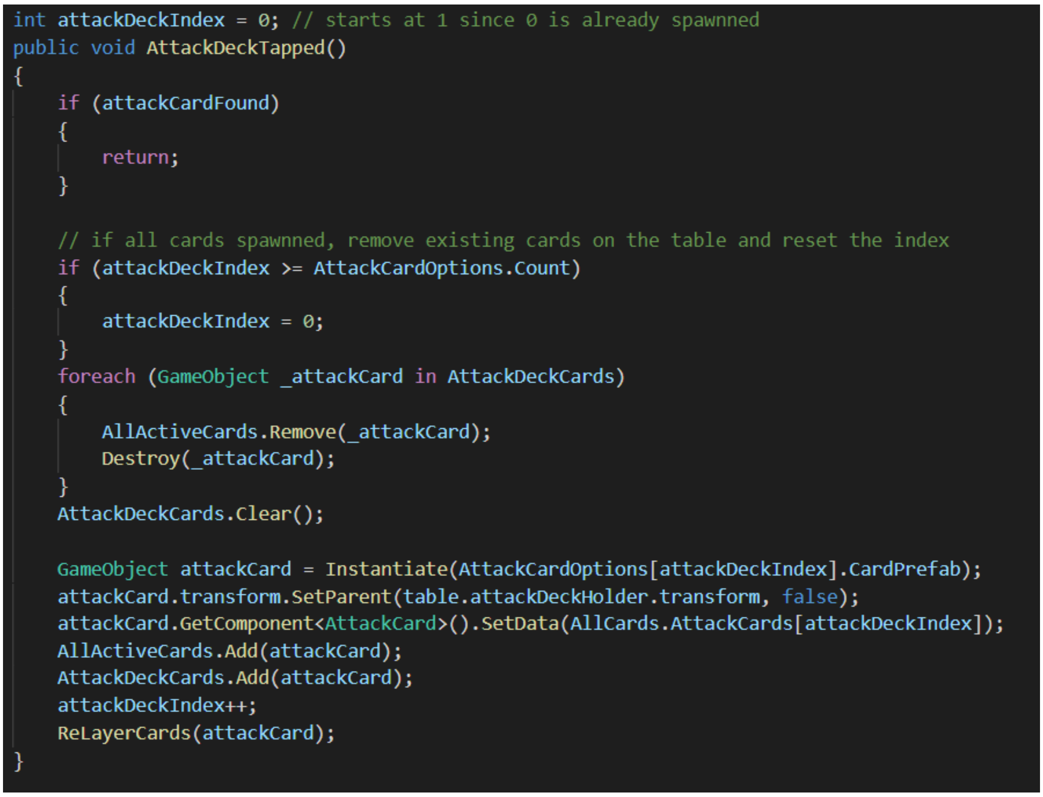
```
// Fisher-Yates Shuffle implementation to shuffle the card decks;
void ShuffleDecks()
{
    int _rand;
    for (int x = 0; x < AttackCardOptions.Count; x++)
    {
        _rand = Random.Range(0, x);
        AttackCardData _tempAttackData = AttackCardOptions[x];
        AttackCardOptions[x] = AttackCardOptions[_rand];
        AttackCardOptions[_rand] = _tempAttackData;
    }
    for (int y = 0; y < DefenseCardOptions.Count; y++)
    {
        _rand = Random.Range(0, y);
        DefenseCardData _tempDefData = DefenseCardOptions[y];
        DefenseCardOptions[y] = DefenseCardOptions[_rand];
        DefenseCardOptions[_rand] = _tempDefData;
    }
}

void ShuffleScenarios()
{
    int _rand;
    for (int y = 0; y < ScenarioCardOptions.Count; y++)
    {
        _rand = Random.Range(0, y);
        ScenarioCardData _tempSceData = ScenarioCardOptions[y];
        ScenarioCardOptions[y] = ScenarioCardOptions[_rand];
        ScenarioCardOptions[_rand] = _tempSceData;
    }
}
```

Figure 4.3 Fisher Yates

4.2.3 When Attack or Defense Deck is tapped

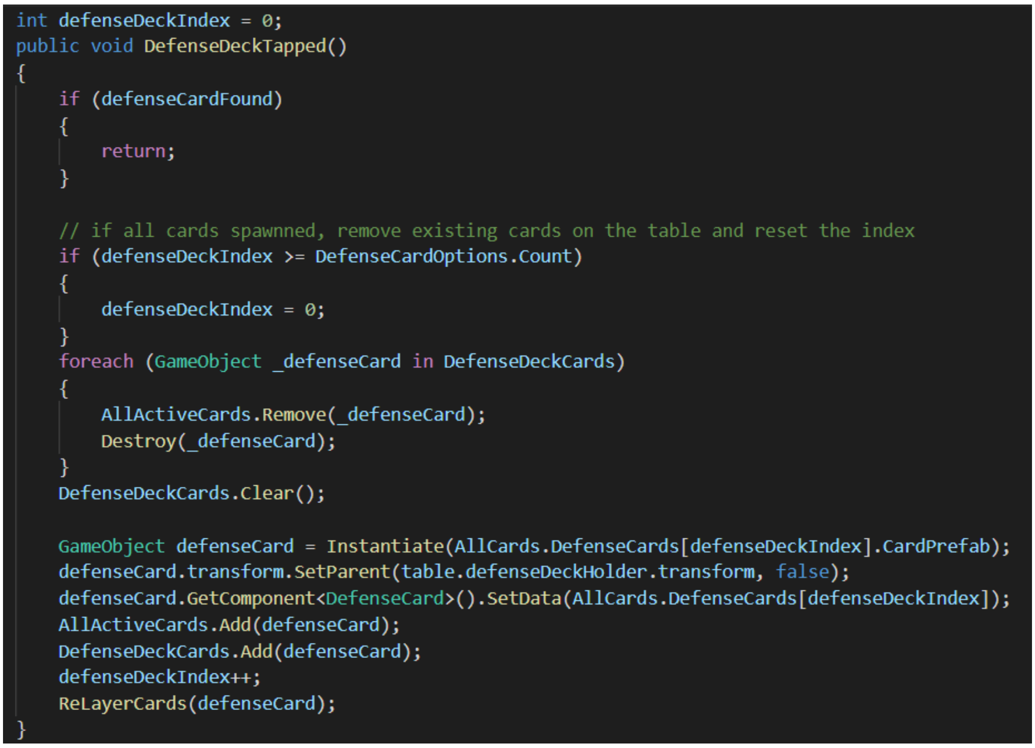
In the CardManager.cs script, we have defined two functions named AttackDeckTapped () and DefenseDeckTapped (). When the Attack or Defense deck is tapped, we start spawning the cards from the 0th index of the shuffled array. Every time the deck is tapped, for each Card Game object (_attackcard) in the deck (AttackDeckCards), we first destroy and remove the previous card from the Active Cards array and then spawn a new card and add it to the Active Cards array. Finally, we increment the index.

Similarly, the Defense Cards are also spawned.



```
int attackDeckIndex = 0; // starts at 1 since 0 is already spawnned
public void AttackDeckTapped()
{
    if (attackCardFound)
    {
        return;
    }

    // if all cards spawnned, remove existing cards on the table and reset the index
    if (attackDeckIndex >= AttackCardOptions.Count)
    {
        attackDeckIndex = 0;
    }
    foreach (GameObject _attackCard in AttackDeckCards)
    {
        AllActiveCards.Remove(_attackCard);
        Destroy(_attackCard);
    }
    AttackDeckCards.Clear();

    GameObject attackCard = Instantiate(AttackCardOptions[attackDeckIndex].CardPrefab);
    attackCard.transform.SetParent(table.attackDeckHolder.transform, false);
    attackCard.GetComponent<AttackCard>().SetData(AllCards.AttackCards[attackDeckIndex]);
    AllActiveCards.Add(attackCard);
    AttackDeckCards.Add(attackCard);
    attackDeckIndex++;
    ReLayerCards(attackCard);
}
```

Figure 4. 4 Attack Deck Tapped



```
int defenseDeckIndex = 0;
public void DefenseDeckTapped()
{
    if (defenseCardFound)
    {
        return;
    }

    // if all cards spawnned, remove existing cards on the table and reset the index
    if (defenseDeckIndex >= DefenseCardOptions.Count)
    {
        defenseDeckIndex = 0;
    }
    foreach (GameObject _defenseCard in DefenseDeckCards)
    {
        AllActiveCards.Remove(_defenseCard);
        Destroy(_defenseCard);
    }
    DefenseDeckCards.Clear();

    GameObject defenseCard = Instantiate(AllCards.DefenseCards[defenseDeckIndex].CardPrefab);
    defenseCard.transform.SetParent(table.defenseDeckHolder.transform, false);
    defenseCard.GetComponent<DefenseCard>().SetData(AllCards.DefenseCards[defenseDeckIndex]);
    AllActiveCards.Add(defenseCard);
    DefenseDeckCards.Add(defenseCard);
    defenseDeckIndex++;
    ReLayerCards(defenseCard);
}
```

Figure 4.5 Defense Deck Tapped

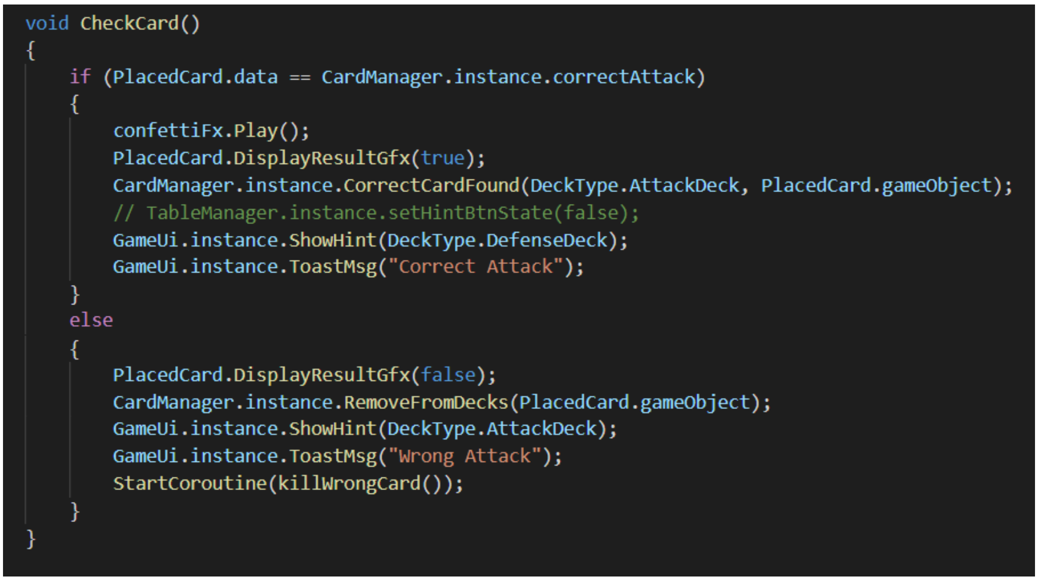
4.2.4 Check whether the cards are correct or wrong

In the scripts AttackPlacement.cs and DefensePlacement.cs, we have defined a function CheckCard() that checks whether the Attack Card or the Defense card are correct or not, respectively.

In this function, we check whether the data of the card placed by the user matches the data of the correct card. If so, the confetti animation gets activated and we call the CorrectCardFound() function in CardManager.cs and display the correct card toast message. Else, we display the animation with the wrong sign, remove the instance of the placed card, and display the wrong card toast message.

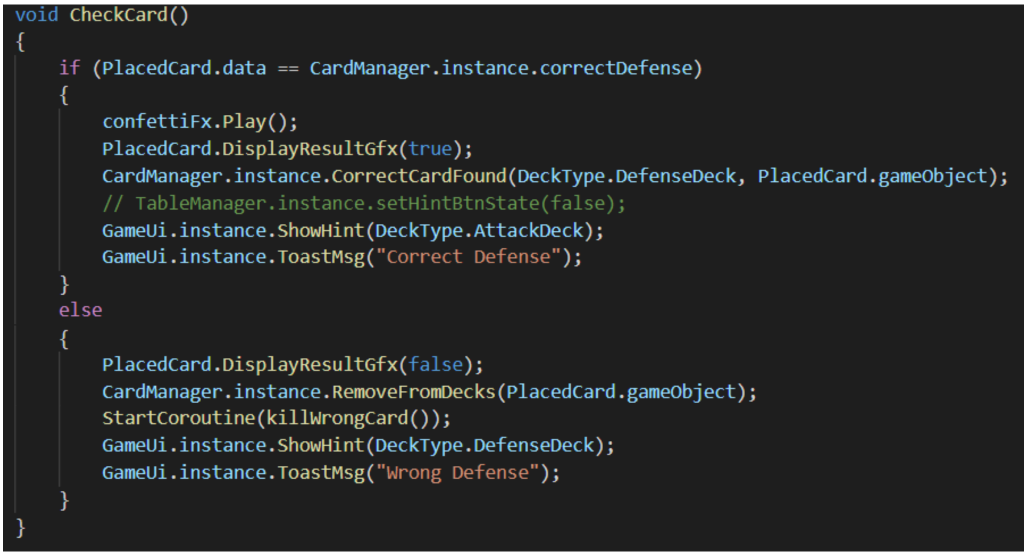
The CorrectCardFound() function is called when the user chooses a correct attack/defense card. In this function, if the deck type is Attack, we add the score (5*scorestreak value) and increase the value of the scorestreak. We then activate the Defense deck for the user.

Whereas, if the deck type is Defense, we just add the score (5*scorestreak value) and increase the value of the scorestreak. If both, correct attack and correct defense cards are found, we load the next scenario.



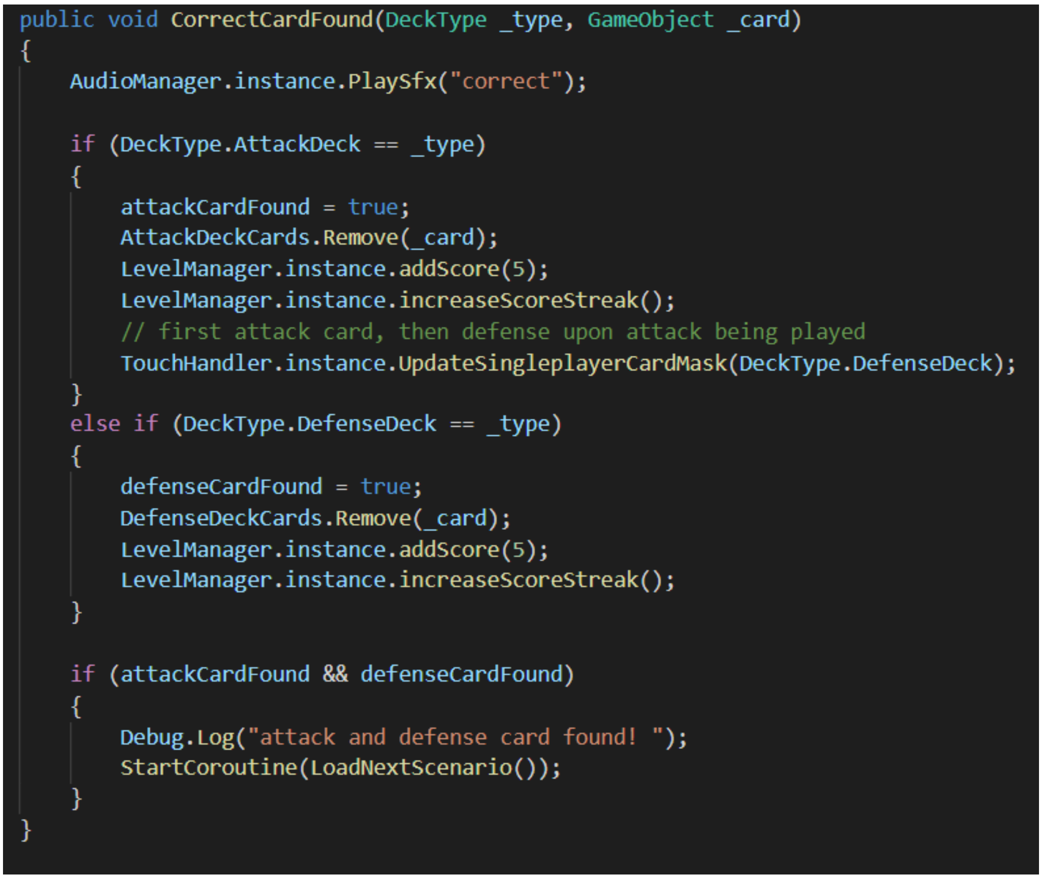
```
void CheckCard()
{
    if (PlacedCard.data == CardManager.instance.correctAttack)
    {
        confettiFx.Play();
        PlacedCard.DisplayResultGfx(true);
        CardManager.instance.CorrectCardFound(DeckType.AttackDeck, PlacedCard.gameObject);
        // TableManager.instance.setHintBtnState(false);
        GameUi.instance.ShowHint(DeckType.DefenseDeck);
        GameUi.instance.ToastMsg("Correct Attack");
    }
    else
    {
        PlacedCard.DisplayResultGfx(false);
        CardManager.instance.RemoveFromDecks(PlacedCard.gameObject);
        GameUi.instance.ShowHint(DeckType.AttackDeck);
        GameUi.instance.ToastMsg("Wrong Attack");
        StartCoroutine(killWrongCard());
    }
}
```

Figure 4.6 Check Attack Card



```
void CheckCard()
{
    if (PlacedCard.data == CardManager.instance.correctDefense)
    {
        confettiFx.Play();
        PlacedCard.DisplayResultGfx(true);
        CardManager.instance.CorrectCardFound(DeckType.DefenseDeck, PlacedCard.gameObject);
        // TableManager.instance.setHintBtnState(false);
        GameUi.instance.ShowHint(DeckType.AttackDeck);
        GameUi.instance.ToastMsg("Correct Defense");
    }
    else
    {
        PlacedCard.DisplayResultGfx(false);
        CardManager.instance.RemoveFromDecks(PlacedCard.gameObject);
        StartCoroutine(killWrongCard());
        GameUi.instance.ShowHint(DeckType.DefenseDeck);
        GameUi.instance.ToastMsg("Wrong Defense");
    }
}
```

Figure 4. 7 Check Defense Card



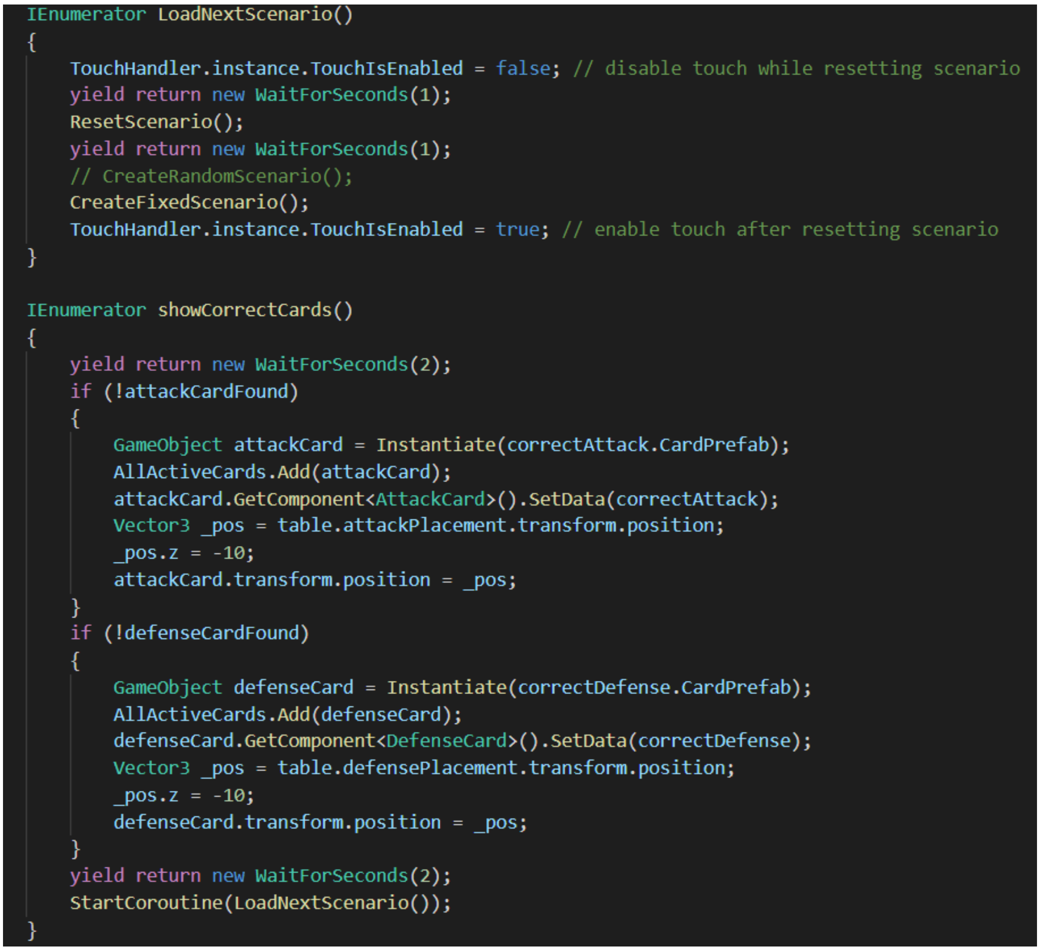
```
public void CorrectCardFound(DeckType _type, GameObject _card)
{
    AudioManager.instance.PlaySfx("correct");

    if (DeckType.AttackDeck == _type)
    {
        attackCardFound = true;
        AttackDeckCards.Remove(_card);
        LevelManager.instance.addScore(5);
        LevelManager.instance.increaseScoreStreak();
        // first attack card, then defense upon attack being played
        TouchHandler.instance.UpdateSingleplayerCardMask(DeckType.DefenseDeck);
    }
    else if (DeckType.DefenseDeck == _type)
    {
        defenseCardFound = true;
        DefenseDeckCards.Remove(_card);
        LevelManager.instance.addScore(5);
        LevelManager.instance.increaseScoreStreak();
    }

    if (attackCardFound && defenseCardFound)
    {
        Debug.Log("attack and defense card found! ");
        StartCoroutine(LoadNextScenario());
    }
}
```

Figure 4. 8 Correct Card Found

4.2.5 Show correct cards if all attempts are exhausted

In the CardManager.cs script, we have defined a function called showCorrectCards() that displays the correct cards to the user in case all the attempts have been exhausted.

It checks if no attack/defense cards were found. If so, then this function loads the correct card’s data and adds it to the correct position (i.1., where the cards were being placed). It then loads the next scenario by calling the LoadNextScenario() function.



```
IEnumerator LoadNextScenario()
{
    TouchHandler.instance.TouchIsEnabled = false; // disable touch while resetting scenario
    yield return new WaitForSeconds(1);
    ResetScenario();
    yield return new WaitForSeconds(1);
    // CreateRandomScenario();
    CreateFixedScenario();
    TouchHandler.instance.TouchIsEnabled = true; // enable touch after resetting scenario
}

IEnumerator showCorrectCards()
{
    yield return new WaitForSeconds(2);
    if (!attackCardFound)
    {
        GameObject attackCard = Instantiate(correctAttack.CardPrefab);
        AllActiveCards.Add(attackCard);
        attackCard.GetComponent<AttackCard>().SetData(correctAttack);
        Vector3 _pos = table.attackPlacement.transform.position;
        _pos.z = -10;
        attackCard.transform.position = _pos;
    }
    if (!defenseCardFound)
    {
        GameObject defenseCard = Instantiate(correctDefense.CardPrefab);
        AllActiveCards.Add(defenseCard);
        defenseCard.GetComponent<DefenseCard>().SetData(correctDefense);
        Vector3 _pos = table.defensePlacement.transform.position;
        _pos.z = -10;
        defenseCard.transform.position = _pos;
    }
    yield return new WaitForSeconds(2);
    StartCoroutine(LoadNextScenario());
}
```

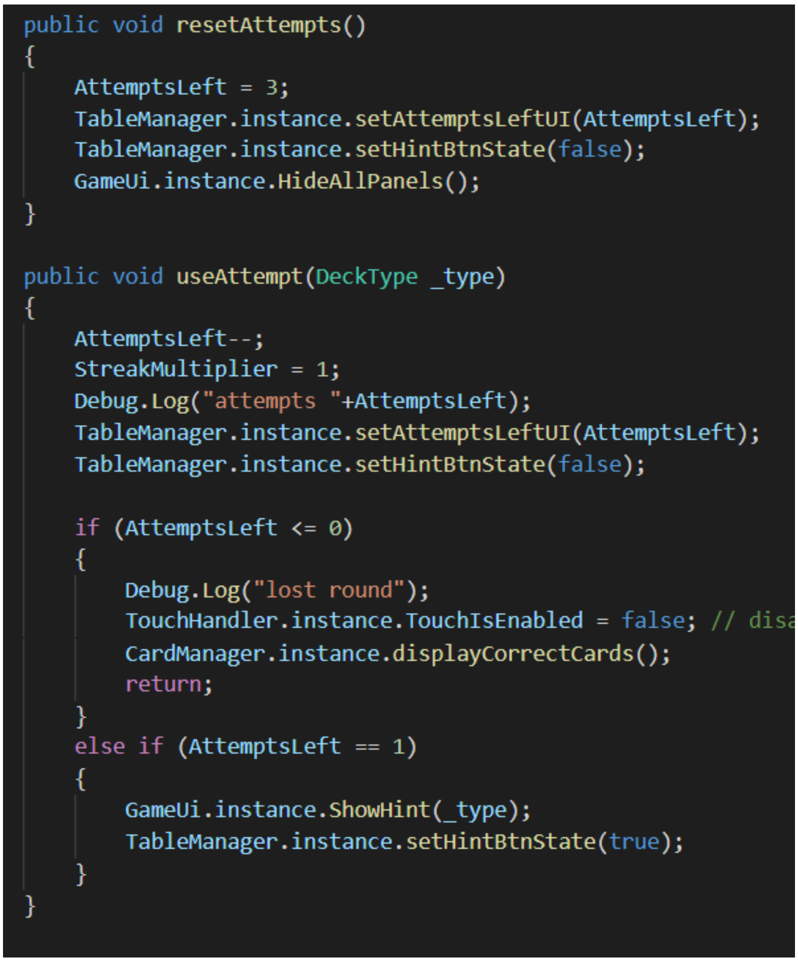
Figure 4.9 Show Correct Cards Function

4.2.6 Attempts

In the LevelManager.cs script, we have defined two functions named resetAttempts() and useAttempt().

The resetAttempts() function is called every time a new scenario card is loaded. The number of AttemptsLeft is reset to 3 and 3 icons of the attempts (shields) are displayed.

The useAttempt() function is called every time a user places any wrong attack/defense card. The number of AttemptsLeft decreases by one and the StreakMultiplier is reset to 1. If all attempts are exhausted by the user, we display the correct cards by calling the displayCorrectCards() function which in turn calls the showCorrectCards() function and the correct cards are displayed. If only 1 attempt is left, we enable the Hint icon which shows the hint for the attack/defense card.



```
public void resetAttempts()
{
    AttemptsLeft = 3;
    TableManager.instance.setAttemptsLeftUI(AttemptsLeft);
    TableManager.instance.setHintBtnState(false);
    GameUi.instance.HideAllPanels();
}

public void useAttempt(DeckType _type)
{
    AttemptsLeft--;
    StreakMultiplier = 1;
    Debug.Log("attempts "+AttemptsLeft);
    TableManager.instance.setAttemptsLeftUI(AttemptsLeft);
    TableManager.instance.setHintBtnState(false);

    if (AttemptsLeft <= 0)
    {
        Debug.Log("lost round");
        TouchHandler.instance.TouchIsEnabled = false; // disa
        CardManager.instance.displayCorrectCards();
        return;
    }
    else if (AttemptsLeft == 1)
    {
        GameUi.instance.ShowHint(_type);
        TableManager.instance.setHintBtnState(true);
    }
}
```

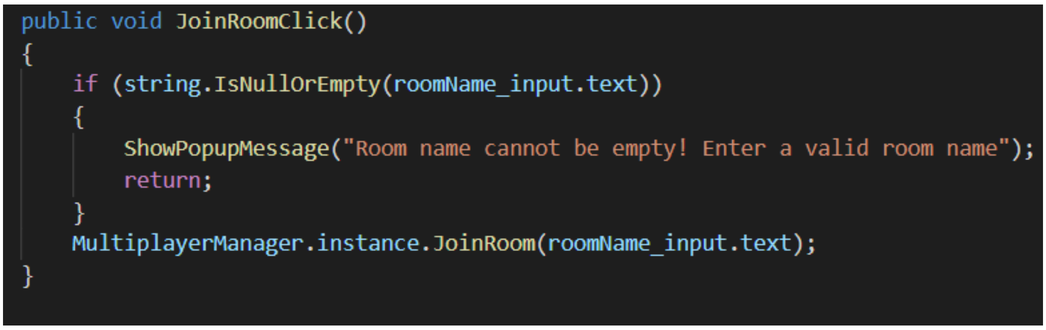
Figure 4.10 Attempts

4.3 Level 2:

4.3.1 Create/Join room

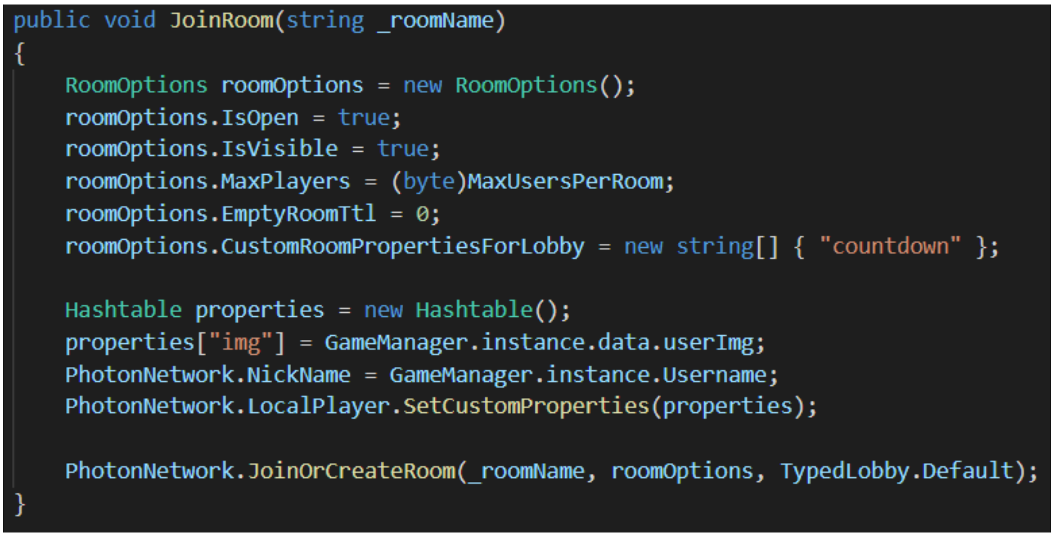
In the MultiplayerUI.cs script, we have defined a function called JoinRoomClick() which is called when the user clicks the Create/Join Room button in Level 2. This function checks whether the room name field is empty or not. If it is empty, it throws an error. If not, it calls the JoinRoom() function from the MultiplayerManager.cs and passes the room name into that function.

The JoinRoom() function in MultiplayerManager.cs script mainly contains predefined functions and methods of Photon Unity Network (PUN). It initializes the room settings and sets the maximum players per room. The PhotonNetwork.JoinOrCreateRoom() is a PUN function that checks whether the room has been created or not. If yes, it checks the number of players. If the number of players in the room is less than the maximum number of players, the user joins the room. Else, a new room is created by the user.



```
public void JoinRoomClick()
{
    if (string.IsNullOrEmpty(roomName_input.text))
    {
        ShowPopupMessage("Room name cannot be empty! Enter a valid room name");
        return;
    }
    MultiplayerManager.instance.JoinRoom(roomName_input.text);
}
```

Figure 4.11 JoinRoomClick Function

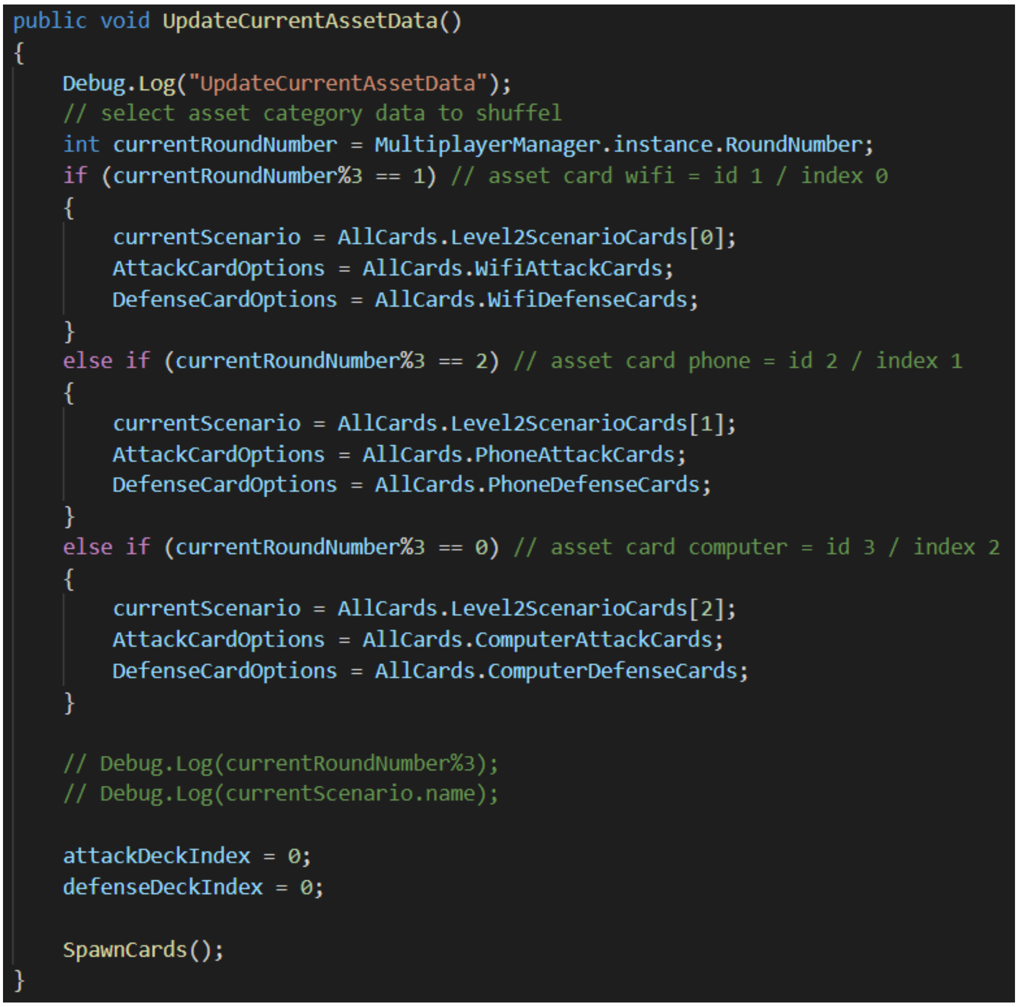


```
public void JoinRoom(string _roomName)
{
    RoomOptions roomOptions = new RoomOptions();
    roomOptions.IsOpen = true;
    roomOptions.IsVisible = true;
    roomOptions.MaxPlayers = (byte)MaxUsersPerRoom;
    roomOptions.EmptyRoomTtl = 0;
    roomOptions.CustomRoomPropertiesForLobby = new string[] { "countdown" };

    Hashtable properties = new Hashtable();
    properties["img"] = GameManager.instance.data.userImg;
    PhotonNetwork.NickName = GameManager.instance.Username;
    PhotonNetwork.LocalPlayer.SetCustomProperties(properties);

    PhotonNetwork.JoinOrCreateRoom(_roomName, roomOptions, TypedLobby.Default);
}
```

Figure 4.12 JoinRoom Function

4.3.2 Determining Asset Cards and loading respective decks

In the MultiplayerCardManager.cs script, we have a function named updateCurrentAssetData (). In this function, we update the asset cards based on the round number. if the round number % 3 == 1, we load the Wi-Fi asset card and its respective decks. If the round number % 3 == 2, we load the phone asset and its respective decks. If the round number % 3 == 0, we load the computer asset and it6s respective decks.



```
public void UpdateCurrentAssetData()
{
    Debug.Log("UpdateCurrentAssetData");
    // select asset category data to shuffel
    int currentRoundNumber = MultiplayerManager.instance.RoundNumber;
    if (currentRoundNumber%3 == 1) // asset card wifi = id 1 / index 0
    {
        currentScenario = AllCards.Level2ScenarioCards[0];
        AttackCardOptions = AllCards.WifiAttackCards;
        DefenseCardOptions = AllCards.WifiDefenseCards;
    }
    else if (currentRoundNumber%3 == 2) // asset card phone = id 2 / index 1
    {
        currentScenario = AllCards.Level2ScenarioCards[1];
        AttackCardOptions = AllCards.PhoneAttackCards;
        DefenseCardOptions = AllCards.PhoneDefenseCards;
    }
    else if (currentRoundNumber%3 == 0) // asset card computer = id 3 / index 2
    {
        currentScenario = AllCards.Level2ScenarioCards[2];
        AttackCardOptions = AllCards.ComputerAttackCards;
        DefenseCardOptions = AllCards.ComputerDefenseCards;
    }

    // Debug.Log(currentRoundNumber%3);
    // Debug.Log(currentScenario.name);

    attackDeckIndex = 0;
    defenseDeckIndex = 0;

    SpawnCards();
}
```

Figure 4.13 UpdateCurrentAssetData Function

4.3.3 Updating the turns

In the MultiplayerManager.cs script, we have a function known as TurnUpdate() which allows us to update the turns and change the Attack and Defense roles for the players.

We set the currentTurn value as that of the Attacker. We then check whether it is the MasterClient's turn or the other player's turn. We then check which deck each player can control.

For MasterClient's turn :-

- If the roundNumber%2 != 0, MasterClient will be able to move the AttackDeck, else they will be able to move the DefenseDeck.
- If the deck that the MasterClient can move is the same as that of currentTurn (AttackDeck), then the MasterClient's id will be active (MasterClient.ActorNumber = 1), else, the other player's turn (MasterClient.GetNext().ActorNumber = 2) will be active.

For the other player’s turn :-

- If the roundNumber%2 != 0, the user will be able to move the DefenseDeck, else they will be able to move the AttackDeck.
- If the deck that the player can move is the same as that of currentTurn (AttackDeck), then the player’s turn will be active (MasterClient.GetNext().ActorNumber = 2), else, theMasterClient’s turn (MasterClient.ActorNumber = 1) will be active.

The UI is then updated accordingly.

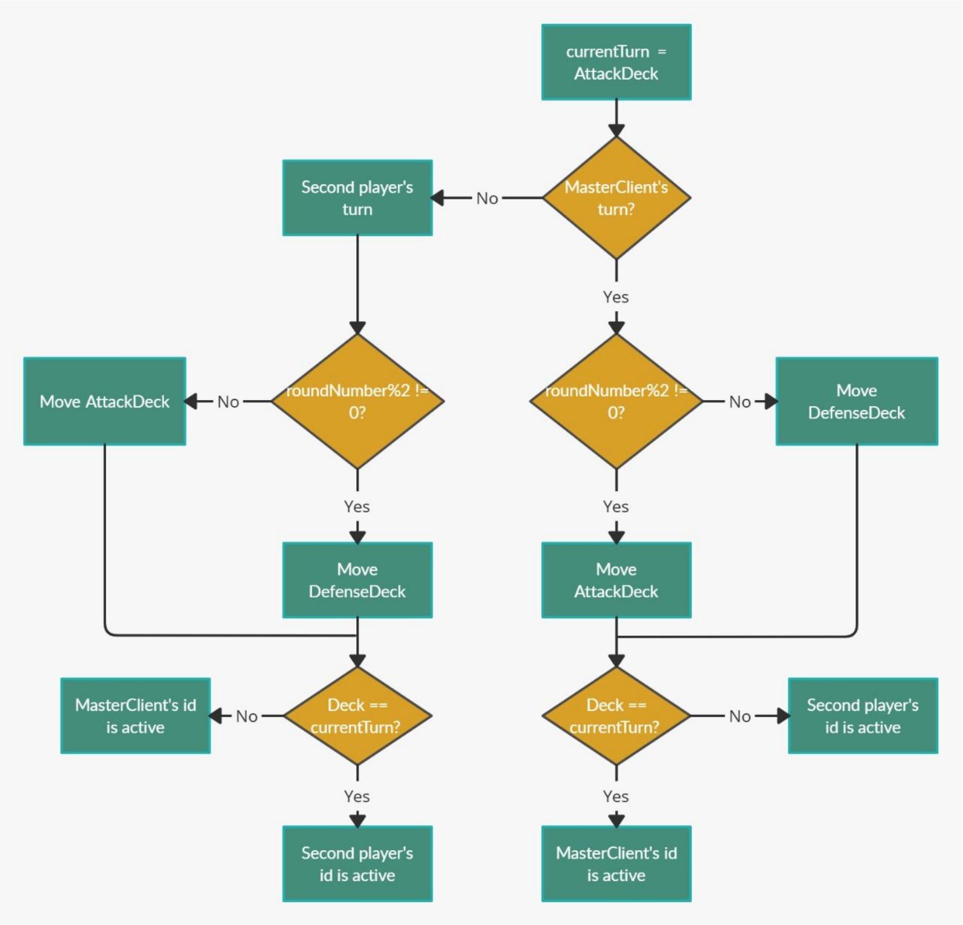
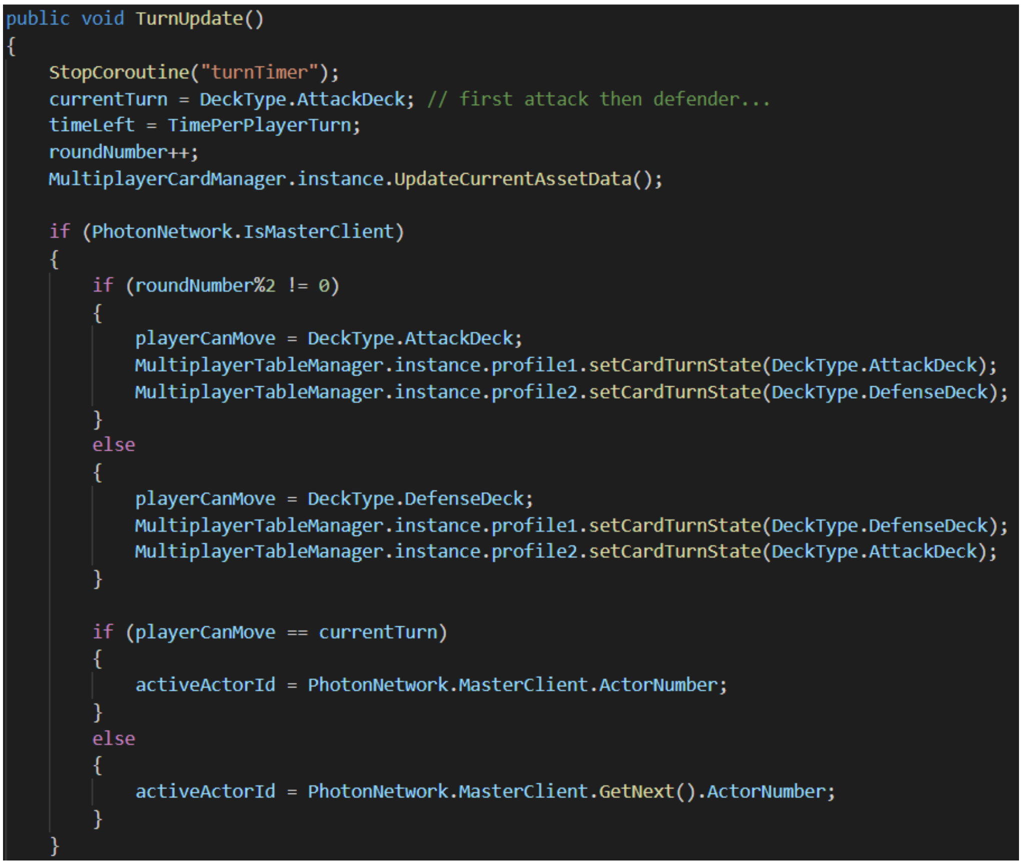
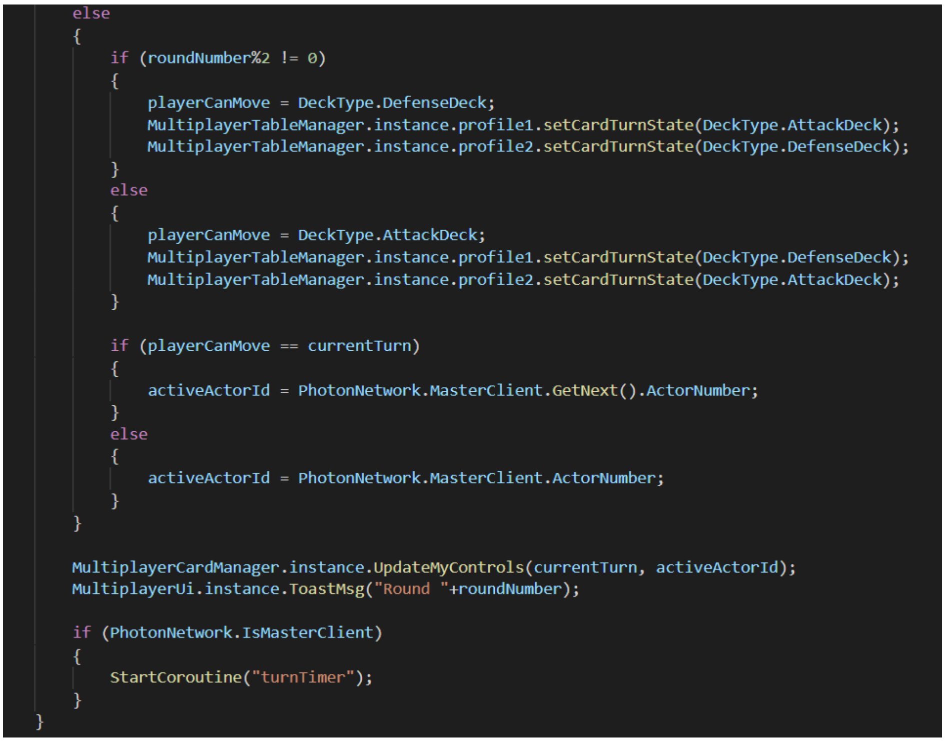


Figure 4.14 UpdateTurn Flowchart



```
public void TurnUpdate()
{
    StopCoroutine("turnTimer");
    currentTurn = DeckType.AttackDeck; // first attack then defender...
    timeLeft = TimePerPlayerTurn;
    roundNumber++;
    MultiplayerCardManager.instance.UpdateCurrentAssetData();

    if (PhotonNetwork.IsMasterClient)
    {
        if (roundNumber%2 != 0)
        {
            playerCanMove = DeckType.AttackDeck;
            MultiplayerTableManager.instance.profile1.setCardTurnState(DeckType.AttackDeck);
            MultiplayerTableManager.instance.profile2.setCardTurnState(DeckType.DefenseDeck);
        }
        else
        {
            playerCanMove = DeckType.DefenseDeck;
            MultiplayerTableManager.instance.profile1.setCardTurnState(DeckType.DefenseDeck);
            MultiplayerTableManager.instance.profile2.setCardTurnState(DeckType.AttackDeck);
        }

        if (playerCanMove == currentTurn)
        {
            activeActorId = PhotonNetwork.MasterClient.ActorNumber;
        }
        else
        {
            activeActorId = PhotonNetwork.MasterClient.GetNext().ActorNumber;
        }
    }
```

Figure 4.15 TurnUpdate Function (1)



```
    else
    {
        if (roundNumber%2 != 0)
        {
            playerCanMove = DeckType.DefenseDeck;
            MultiplayerTableManager.instance.profile1.setCardTurnState(DeckType.AttackDeck);
            MultiplayerTableManager.instance.profile2.setCardTurnState(DeckType.DefenseDeck);
        }
        else
        {
            playerCanMove = DeckType.AttackDeck;
            MultiplayerTableManager.instance.profile1.setCardTurnState(DeckType.DefenseDeck);
            MultiplayerTableManager.instance.profile2.setCardTurnState(DeckType.AttackDeck);
        }

        if (playerCanMove == currentTurn)
        {
            activeActorId = PhotonNetwork.MasterClient.GetNext().ActorNumber;
        }
        else
        {
            activeActorId = PhotonNetwork.MasterClient.ActorNumber;
        }
    }

    MultiplayerCardManager.instance.UpdateMyControls(currentTurn, activeActorId);
    MultiplayerUi.instance.ToastMsg("Round "+roundNumber);

    if (PhotonNetwork.IsMasterClient)
    {
        StartCoroutine("turnTimer");
    }
}
```

Figure 4.16 TurnUpdate Function (2)

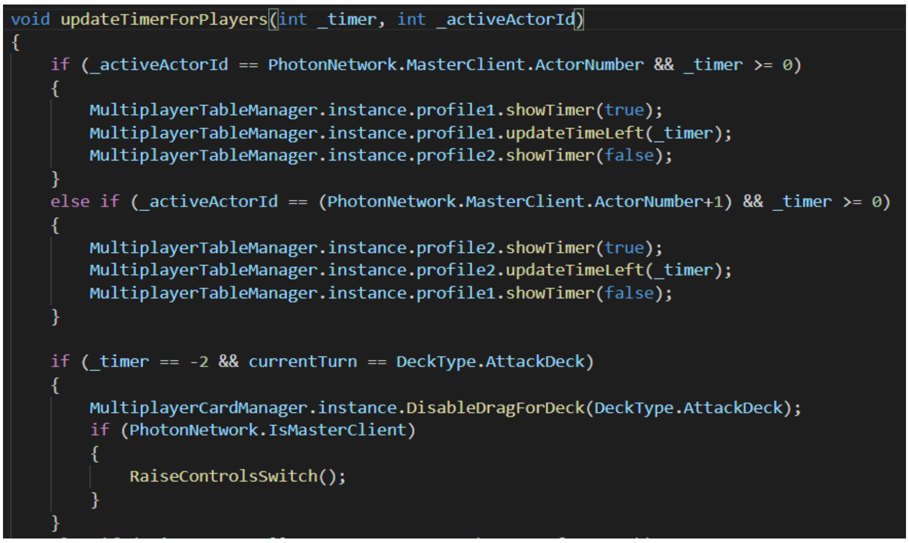
4.3.4 Updating Timer for Players

In the MultiplayerManager.cs, we have a function named updateTimerForPlayers() which updates the timer for each player. If currently it is MasterClient's turn, the timer is shown and updated for their profile (profile 1) and timer is not shown for the other player's profile (profile 2). Else, if it is the other player's turn, the timer is shown and updated for their profile (profile 2) and timer is hidden for MasterClient's profile (profile 1)

When the _timer value is equal to 0, if the current turn was that of the Attacker, the turn switches to the Defender. Else, the cards are reset and a new round begins.

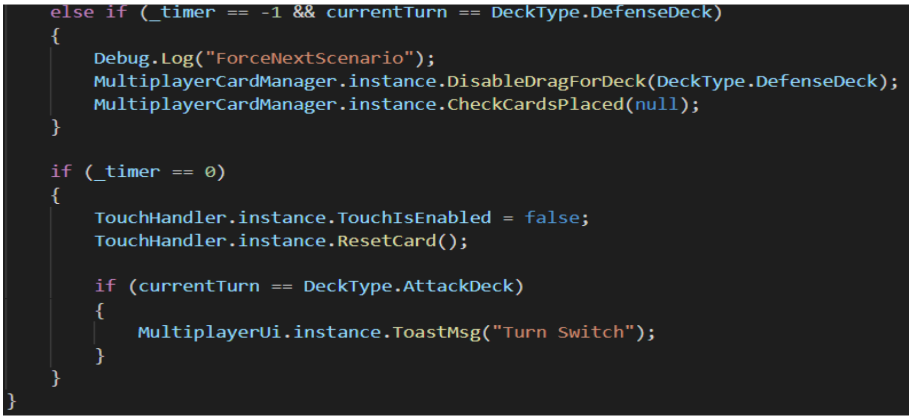
We need extra 1-2 seconds to change the UI settings. Therefore, 2 seconds after the timer runs out (_timer == -2), if the current turn was that of the Attacker, the controls for the attack deck are disabled and we turn the switch over to the defender.

1 second after the timer runs out (_timer ==-1), if the current turn was that of the Defender, we disable the controls for the defense deck and force the next round.



```
void updateTimerForPlayers(int _timer, int _activeActorId)
{
    if (_activeActorId == PhotonNetwork.MasterClient.ActorNumber && _timer >= 0)
    {
        MultiplayerTableManager.instance.profile1.showTimer(true);
        MultiplayerTableManager.instance.profile1.updateTimeLeft(_timer);
        MultiplayerTableManager.instance.profile2.showTimer(false);
    }
    else if (_activeActorId == (PhotonNetwork.MasterClient.ActorNumber+1) && _timer >= 0)
    {
        MultiplayerTableManager.instance.profile2.showTimer(true);
        MultiplayerTableManager.instance.profile2.updateTimeLeft(_timer);
        MultiplayerTableManager.instance.profile1.showTimer(false);
    }

    if (_timer == -2 && currentTurn == DeckType.AttackDeck)
    {
        MultiplayerCardManager.instance.DisableDragForDeck(DeckType.AttackDeck);
        if (PhotonNetwork.IsMasterClient)
        {
            RaiseControlsSwitch();
        }
    }
```

Figure 4.17 updateTimerForPlayers (1)



```
    else if (_timer == -1 && currentTurn == DeckType.DefenseDeck)
    {
        Debug.Log("ForceNextScenario");
        MultiplayerCardManager.instance.DisableDragForDeck(DeckType.DefenseDeck);
        MultiplayerCardManager.instance.CheckCardsPlaced(null);
    }

    if (_timer == 0)
    {
        TouchHandler.instance.TouchIsEnabled = false;
        TouchHandler.instance.ResetCard();

        if (currentTurn == DeckType.AttackDeck)
        {
            MultiplayerUi.instance.ToastMsg("Turn Switch");
        }
    }
}
```

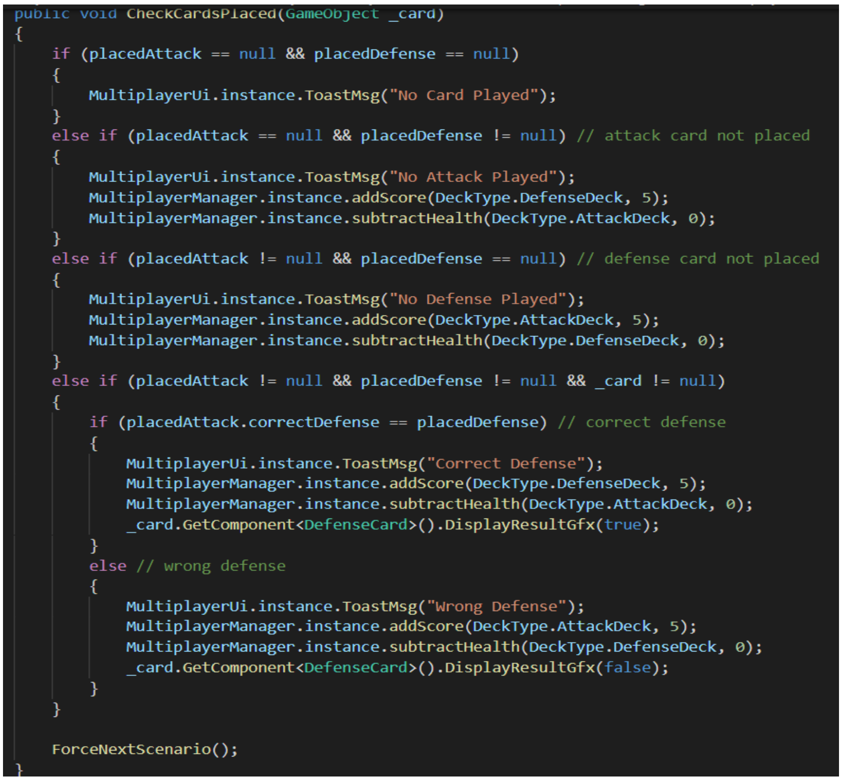
Figure 4. 18 updateTimerForPlayers (2)

4.3.5 Checking placed cards and allocating scores.

In the MultiplayerCardManager.cs script, we have a function called CheckCardsPlaced() that is called after both players are done with their turns. It checks the cards that have been placed by the user and allocates the scores accordingly.

- If no attack or defense cards have been placed, no points are allocated to either players and their scorestreaks are reset to 1.
- If no attack card has been placed but defense card has been placed, the Defender gains points equal to 5*scorestreak value and his scorestreak value increases by one, and the Attacker loses health and his scorestreak resets to 1.
- If attack card has been placed but no defense card has been placed, the Attacker gains points equal to 5*scorestreak value and his scorestreak value increases by one and the Defender loses health and his scorestreak value resets to 1.
- If both, attack and defense cards have been placed: -
    - If the Defender chooses the correct Defense Card for the Attack Card, the Defender gains points equal to 5*scorestreak value and his scorestreak value increases by 1. The Attacker loses health and his scorestreak value is reset to 1.
    - If the Defender chooses the wrong Defense Card for the Attack Card, the Attacker gains points equal to 5*scorestreak value and his scorestreak value increases by 1. The Defender loses health and his scorestreak value is reset to 1.

We then force the next round to begin.

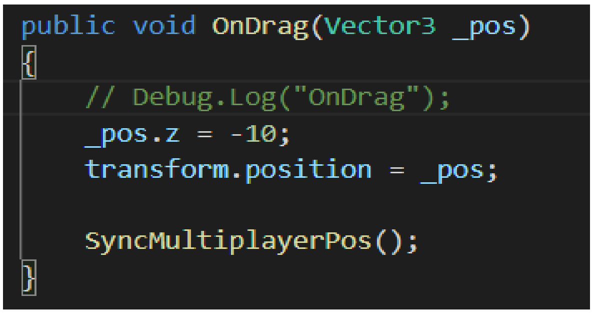


```
public void CheckCardsPlaced(GameObject _card)
{
    if (placedAttack == null && placedDefense == null)
    {
        MultiplayerUi.instance.ToastMsg("No Card Played");
    }
    else if (placedAttack == null && placedDefense != null) // attack card not placed
    {
        MultiplayerUi.instance.ToastMsg("No Attack Played");
        MultiplayerManager.instance.addScore(DeckType.DefenseDeck, 5);
        MultiplayerManager.instance.subtractHealth(DeckType.AttackDeck, 0);
    }
    else if (placedAttack != null && placedDefense == null) // defense card not placed
    {
        MultiplayerUi.instance.ToastMsg("No Defense Played");
        MultiplayerManager.instance.addScore(DeckType.AttackDeck, 5);
        MultiplayerManager.instance.subtractHealth(DeckType.DefenseDeck, 0);
    }
    else if (placedAttack != null && placedDefense != null && _card != null)
    {
        if (placedAttack.correctDefense == placedDefense) // correct defense
        {
            MultiplayerUi.instance.ToastMsg("Correct Defense");
            MultiplayerManager.instance.addScore(DeckType.DefenseDeck, 5);
            MultiplayerManager.instance.subtractHealth(DeckType.AttackDeck, 0);
            _card.GetComponent<DefenseCard>().DisplayResultGfx(true);
        }
        else // wrong defense
        {
            MultiplayerUi.instance.ToastMsg("Wrong Defense");
            MultiplayerManager.instance.addScore(DeckType.AttackDeck, 5);
            MultiplayerManager.instance.subtractHealth(DeckType.DefenseDeck, 0);
            _card.GetComponent<DefenseCard>().DisplayResultGfx(false);
        }
    }

    ForceNextScenario();
}
```

Figure 4.19 CheckCardsPlaced

4.3.6 Syncing of data across network

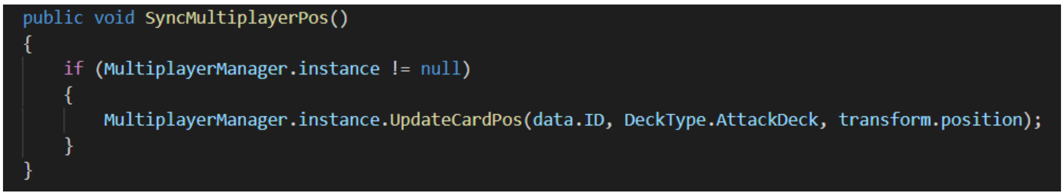
In the AttackCard.cs script and the DefenseCard.cs script, we have an OnDrag() function which is triggered each time a player drags and moves the card. The new position of the cards need to be synced across the other players too. To do this, this function calls the SyncMultiplayerPos() function which is present in both AttackCard.cs and DefenseCard.cs scripts.



```
public void OnDrag(Vector3 _pos)
{
    // Debug.Log("OnDrag");
    _pos.z = -10;
    transform.position = _pos;

    SyncMultiplayerPos();
}
```

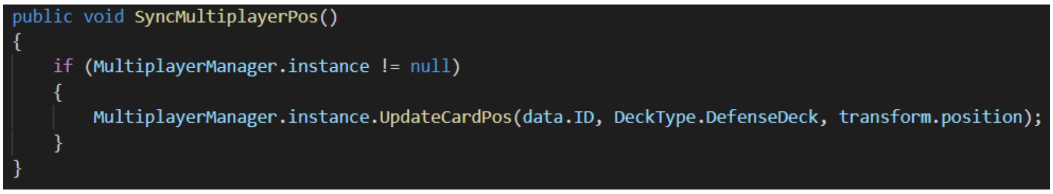
Figure 4.20 OnDrag()

SyncMultiplayerPos() first checks whether MultiplayerManager.instance != null, which implies that we are on Level 2. It then triggers the event UpDateCardPos() in MultiplayerManager.cs and passes the data to be synced such as the card ID, deck type and card position to sync the cards across the network.



```
public void SyncMultiplayerPos()
{
    if (MultiplayerManager.instance != null)
    {
        MultiplayerManager.instance.UpdateCardPos(data.ID, DeckType.AttackDeck, transform.position);
    }
}
```

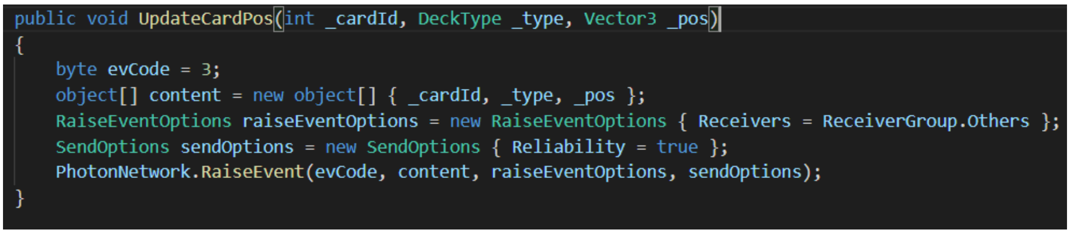
Figure 4.21 SyncMultiplayerPos () (Attack Deck)



```
public void SyncMultiplayerPos()
{
    if (MultiplayerManager.instance != null)
    {
        MultiplayerManager.instance.UpdateCardPos(data.ID, DeckType.DefenseDeck, transform.position);
    }
}
```

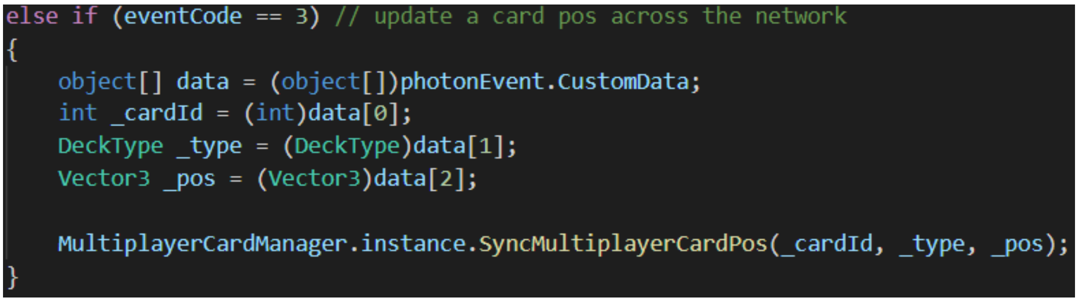
Figure 4.22 SyncMultiplayerPos() (Defense Deck)

When the UpdateCardPos() event is raised, this event collects all the data of the cards to be synced from all players and passes it to the SyncMultiplierCardPos() function in MultiplayerCardManager.cs.



```
public void UpdateCardPos(int _cardId, DeckType _type, Vector3 _pos)
{
    byte evCode = 3;
    object[] content = new object[] { _cardId, _type, _pos };
    RaiseEventOptions raiseEventOptions = new RaiseEventOptions { Receivers = ReceiverGroup.Others };
    SendOptions sendOptions = new SendOptions { Reliability = true };
    PhotonNetwork.RaiseEvent(evCode, content, raiseEventOptions, sendOptions);
}
```

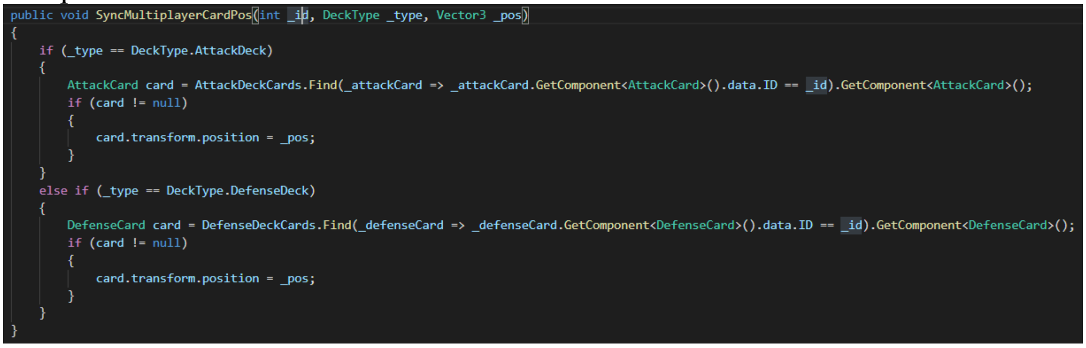
Figure 4.23 UpdateCardPos () (1)



```
else if (eventCode == 3) // update a card pos across the network
{
    object[] data = (object[])photonEvent.CustomData;
    int _cardId = (int)data[0];
    DeckType _type = (DeckType)data[1];
    Vector3 _pos = (Vector3)data[2];

    MultiplayerCardManager.instance.SyncMultiplayerCardPos(_cardId, _type, _pos);
}
```

Figure 4.24 UpdateCardPos () (2)

In the SyncMultiplierCardPos() function, it first checks what type of deck does the card whose data is being synced belongs to. It then finds the card that matches with the received data, i.e. with the same ID, deck. If the card is found, it then updates the position on both the users' screens.

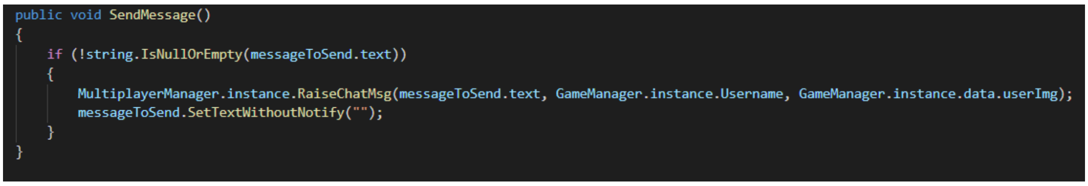


```
public void SyncMultiplayerCardPos(int _id, DeckType _type, Vector3 _pos)
{
    if (_type == DeckType.AttackDeck)
    {
        AttackCard card = AttackDeckCards.Find(_attackCard => _attackCard.GetComponent<AttackCard>().data.ID == _id).GetComponent<AttackCard>();
        if (card != null)
        {
            card.transform.position = _pos;
        }
    }
    else if (_type == DeckType.DefenseDeck)
    {
        DefenseCard card = DefenseDeckCards.Find(_defenseCard => _defenseCard.GetComponent<DefenseCard>().data.ID == _id).GetComponent<DefenseCard>();
        if (card != null)
        {
            card.transform.position = _pos;
        }
    }
}
```

Figure 4.25 SyncMultiplierCardPos ()

4.3.7 In Game-Chat

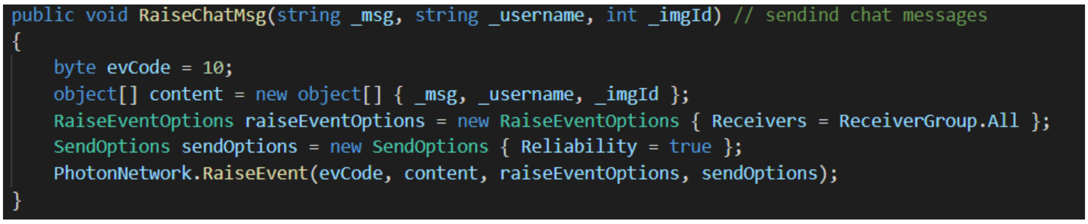
In the ChatPanel.cs script, we have a SendMessage() function. If the text field containing the message to be sent is not null, then it raises an event in the MultiplayerManager called as RaiseChatMsg() and passes the details such as the chat text, the username of the player sending the message and the avatar of the player sending the message.



```
public void SendMessage()
{
    if (!string.IsNullOrEmpty(messageToSend.text))
    {
        MultiplayerManager.instance.RaiseChatMsg(messageToSend.text, GameManager.instance.Username, GameManager.instance.data.userImg);
        messageToSend.SetTextWithoutNotify("");
    }
}
```

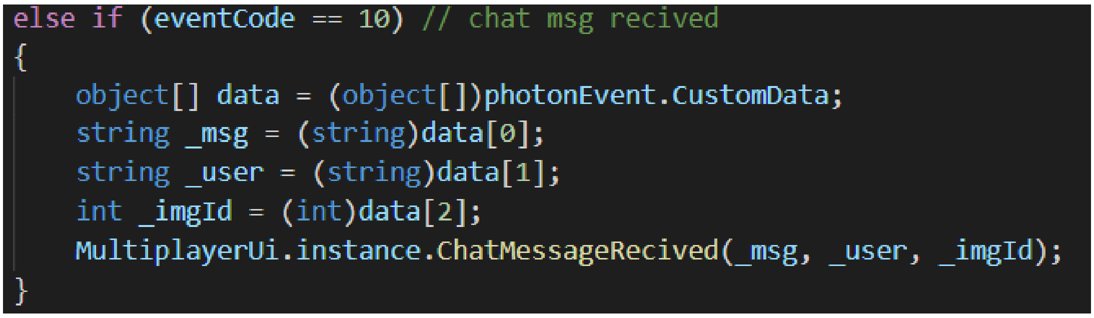
Figure 4.26 SendMessage ()

Every time the RaiseChatMsg() event is raised, it collects all the data about the chat message and the user details and calls the ChatMessageReceived() function in MultiplayerUI.cs and passes all the data into this function.



```
public void RaiseChatMsg(string _msg, string _username, int _imgId) // sendind chat messages
{
    byte evCode = 10;
    object[] content = new object[] { _msg, _username, _imgId };
    RaiseEventOptions raiseEventOptions = new RaiseEventOptions { Receivers = ReceiverGroup.All };
    SendOptions sendOptions = new SendOptions { Reliability = true };
    PhotonNetwork.RaiseEvent(evCode, content, raiseEventOptions, sendOptions);
}
```

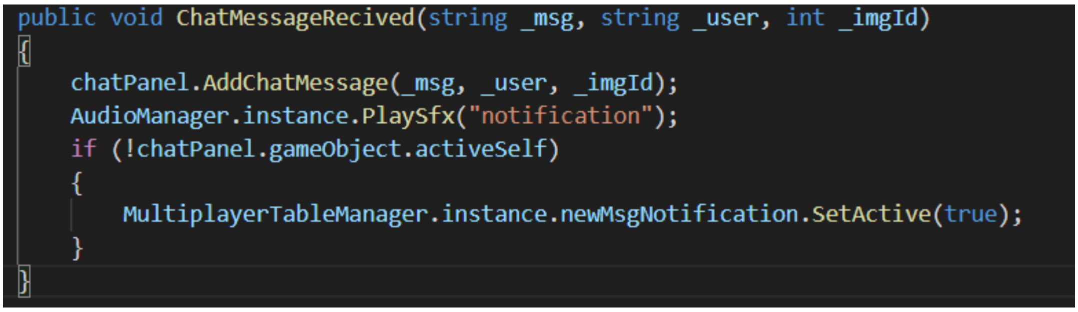
Figure 4.27 RaiseChatMsg()



```
else if (eventCode == 10) // chat msg recived
{
    object[] data = (object[])photonEvent.CustomData;
    string _msg = (string)data[0];
    string _user = (string)data[1];
    int _imgId = (int)data[2];
    MultiplayerUi.instance.ChatMessageRecived(_msg, _user, _imgId);
}
```

Figure 4.28 Event Raised

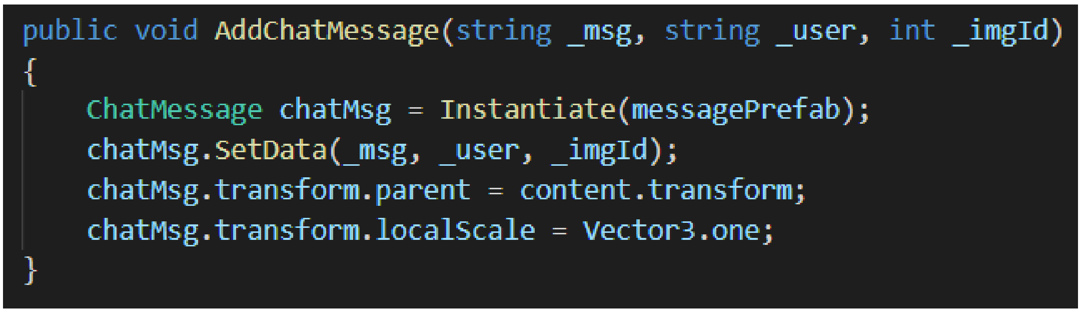
The Chat Message Received () function calls the AddChatMessage() function in chatPanel.cs and passes the chat and user data into the function. It also makes the UI changes and displays a notification exclamation mark onto the chat icon of the player who has received the message.



```
public void ChatMessageRecived(string _msg, string _user, int _imgId)
{
    chatPanel.AddChatMessage(_msg, _user, _imgId);
    AudioManager.instance.PlaySfx("notification");
    if (!chatPanel.gameObject.activeSelf)
    {
        MultiplayerTableManager.instance.newMsgNotification.SetActive(true);
    }
}
```

Figure 4.29 ChatMessageReceived ()

The AddChatMessage() function then receives this data and adds this chat message along with the player username and avatar to the chat panel. This data is synced across the network.



```
public void AddChatMessage(string _msg, string _user, int _imgId)
{
    ChatMessage chatMsg = Instantiate(messagePrefab);
    chatMsg.SetData(_msg, _user, _imgId);
    chatMsg.transform.parent = content.transform;
    chatMsg.transform.localScale = Vector3.one;
}
```

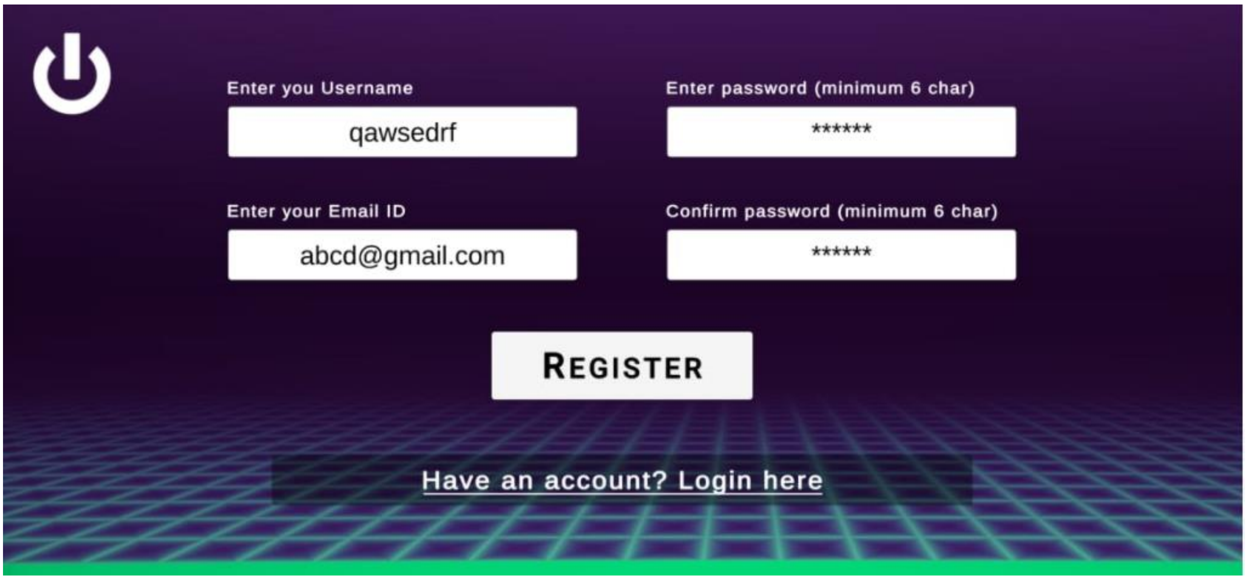
Figure 4.30 AddChatMessage ()

Chapter 5: Screenshots

This section includes the screenshots of the game to get an insight into what the game looks like. The screenshots are given in the order of the game flow to give users a clear idea of the flow of the game.

5.1 Register Page

The register page appears when the user clicks on “Don’t have an account? Register here” on the login page. The user needs to enter a unique username and email id, and create their password in order to register.



Enter you Username
qawsedrf

Enter password (minimum 6 char)
******

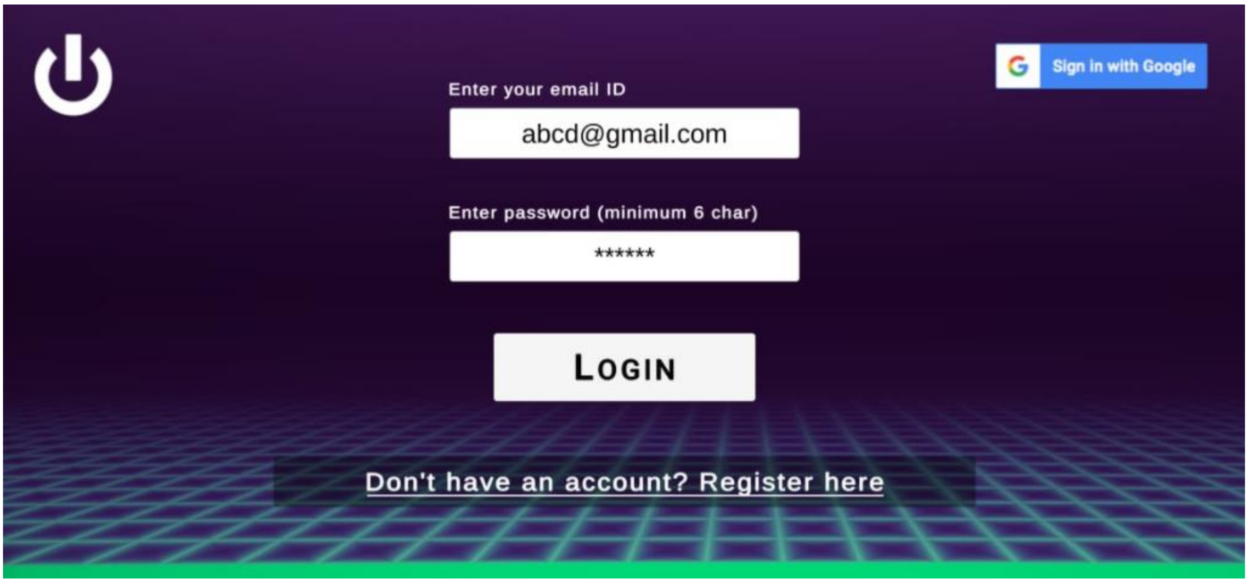
Enter your Email ID
abcd@gmail.com

Confirm password (minimum 6 char)
******

Figure 5.1 Register Page

5.2 Login Page

The login page appears when the user opens the application. The user needs to enter his email address and password with which he has registered into the application in order to sign in to the application.



Enter your email ID
abcd@gmail.com

Enter password (minimum 6 char)
******

Figure 5.2 Login Page

5.3 Login via Google Account

On the login page, we have an option to Sign in with Google. On clicking this option, the user is given an option to choose which Google account he/she wishes to sign in with.

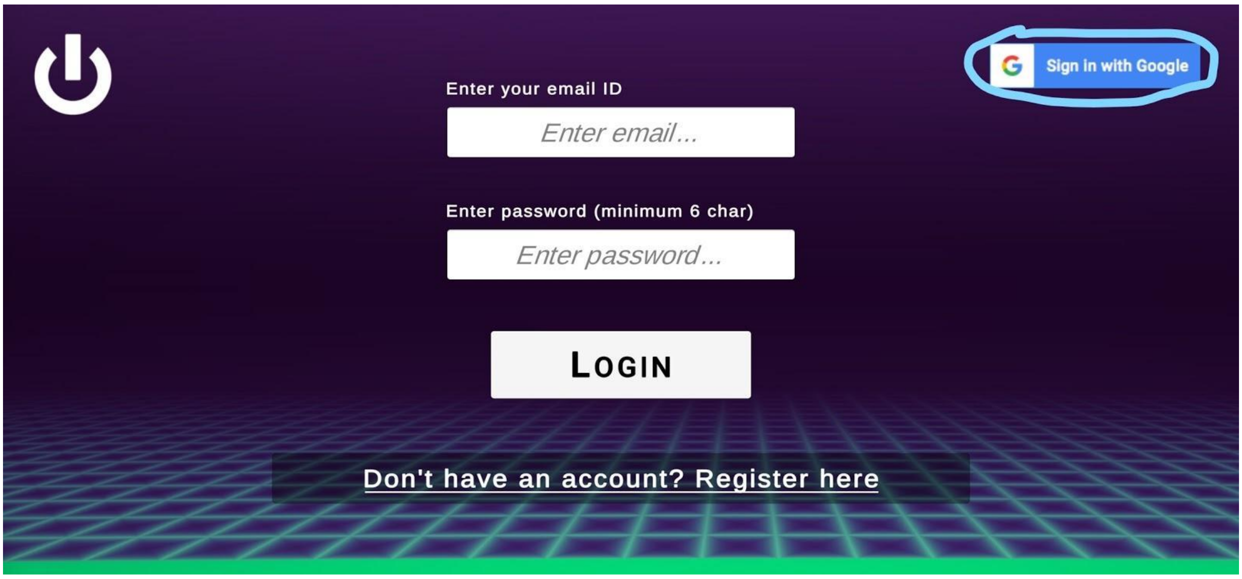
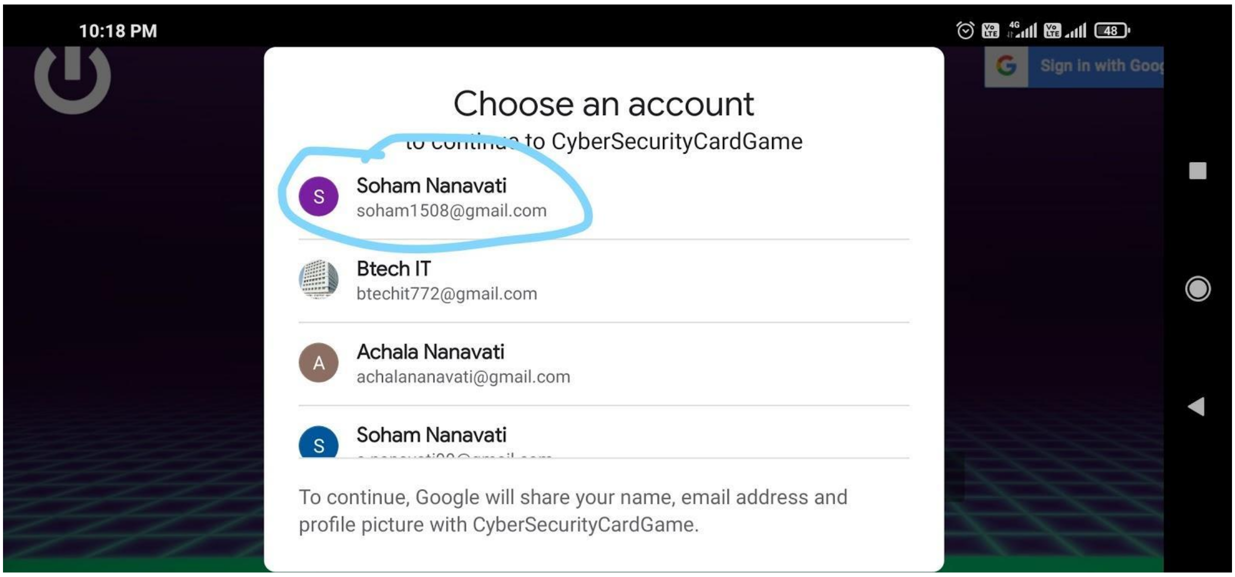


Figure 5. 3 Login via Google Account (1)



Choose an account

to continue to CyberSecurityCardGame

Soham Nanavati
soham1508@gmail.com

Btech IT
btechit772@gmail.com

Achala Nanavati
achalananavati@gmail.com

Soham Nanavati

To continue, Google will share your name, email address and profile picture with CyberSecurityCardGame.

Figure 5. 4 Login via Google Account (2)

5.4 Main Menu

The main menu appears after the user successfully logs in to the application. The Level 1 and Level 2 Tutorial options are initially locked for the user since the user needs to score at least 200 points in Level 1 to unlock Level 2.

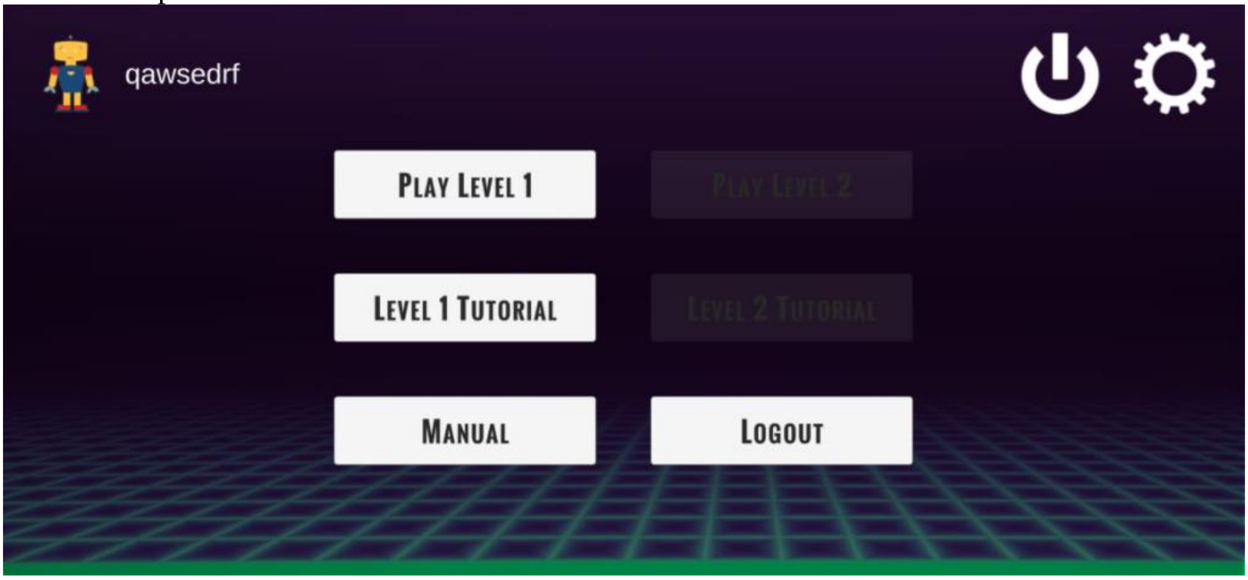
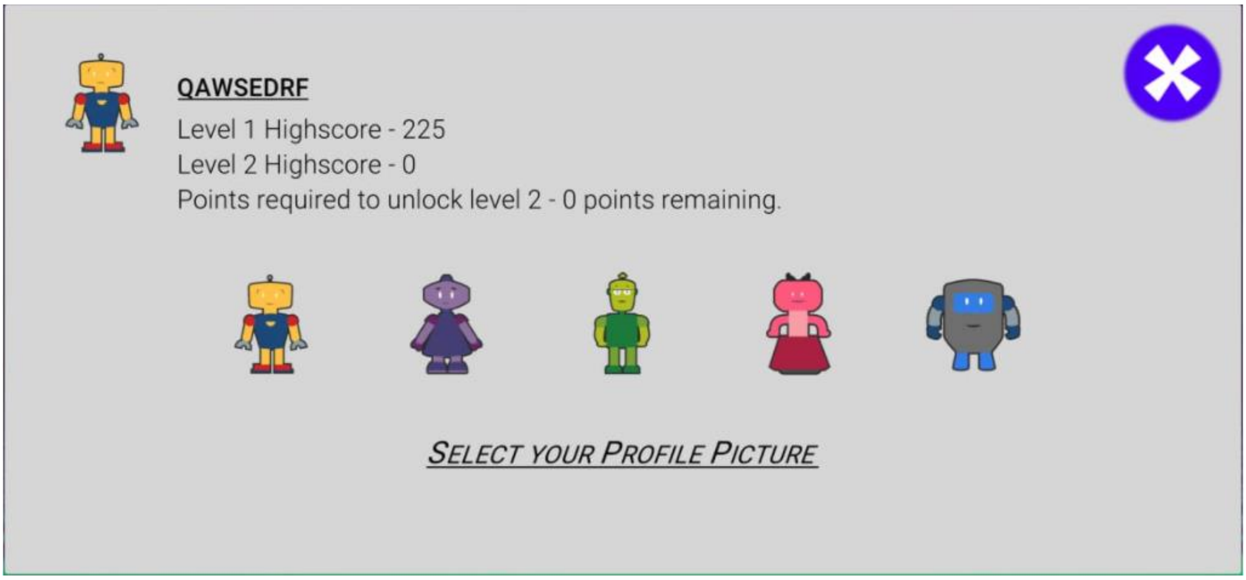


Figure 5. 5 Main Menu

5.5 User Profile

On the main menu, the user can click on his avatar on the top left corner to view his details and progress and change his avatar.



QAWSEDRF

Level 1 Highscore - 225

Level 2 Highscore - 0

Points required to unlock level 2 - 0 points remaining.

Figure 5.6 User Profile

5.6 Level 1

The user can play Level 1 by clicking on “Play Level 1” button on the Main Menu. It redirects the user to Level 1 where the user is given a random scenario, and he needs to pair the correct attack and defense cards with the displayed scenario.

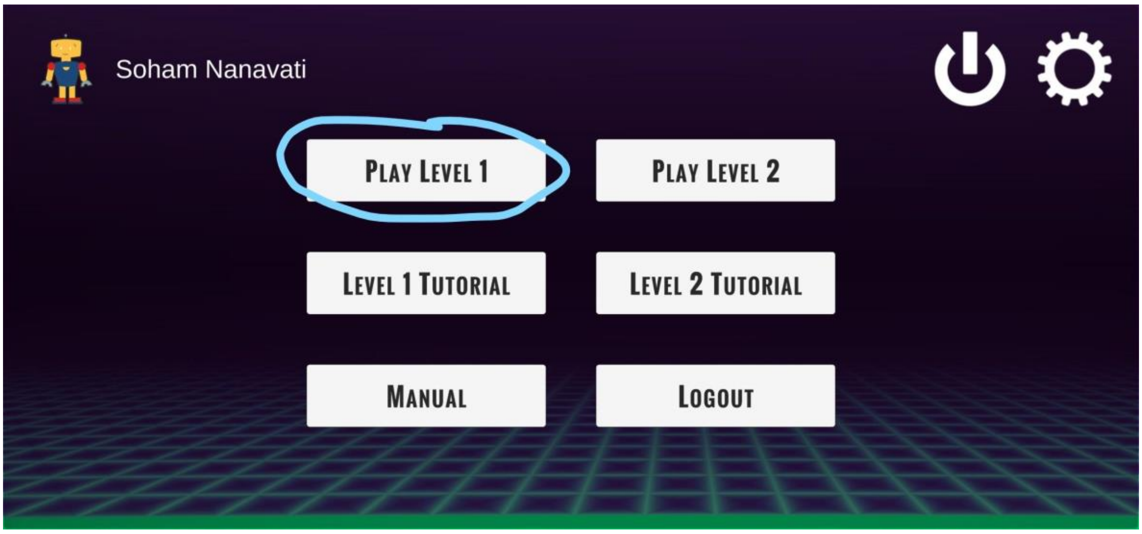
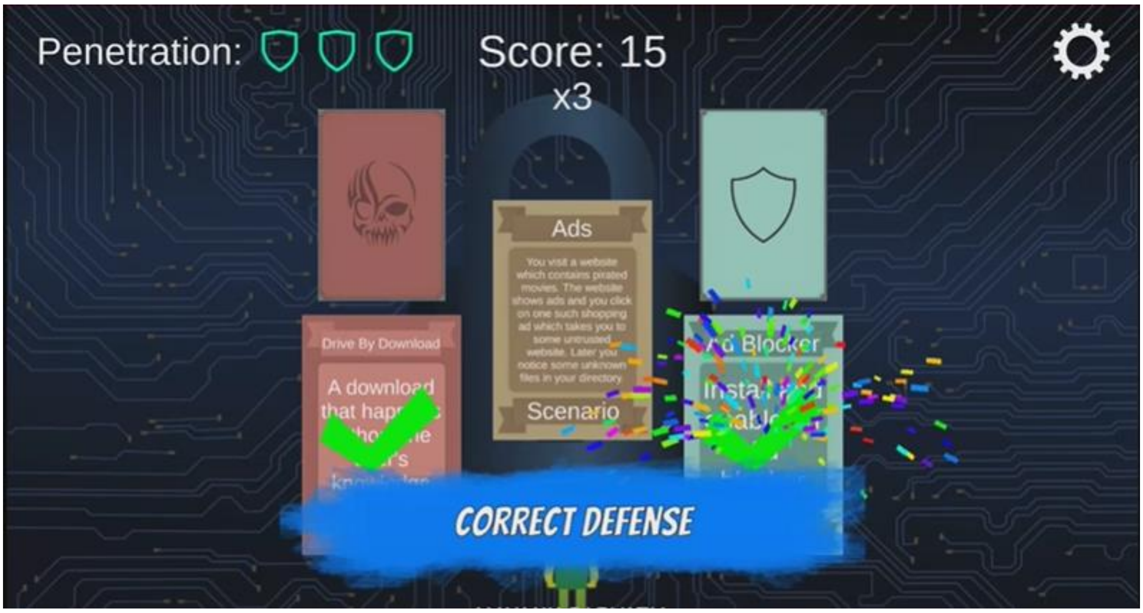
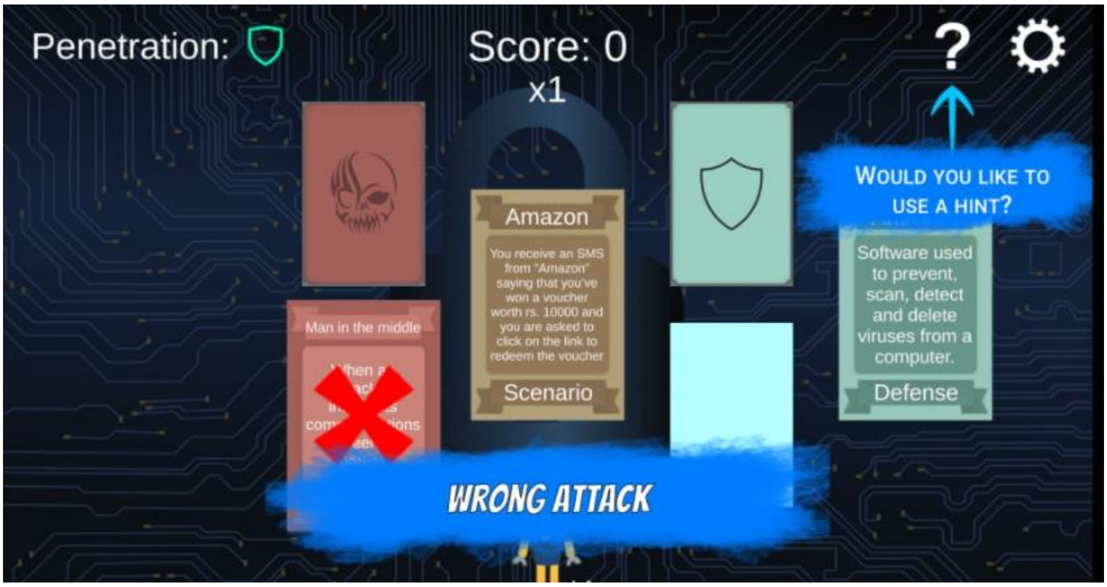


Figure 5.7 Play Level 1



You visit a website which contains pirated movies. The website shows ads and you click on one such shopping ad which takes you to some untrusted website. Later you notice some unknown files in your directory

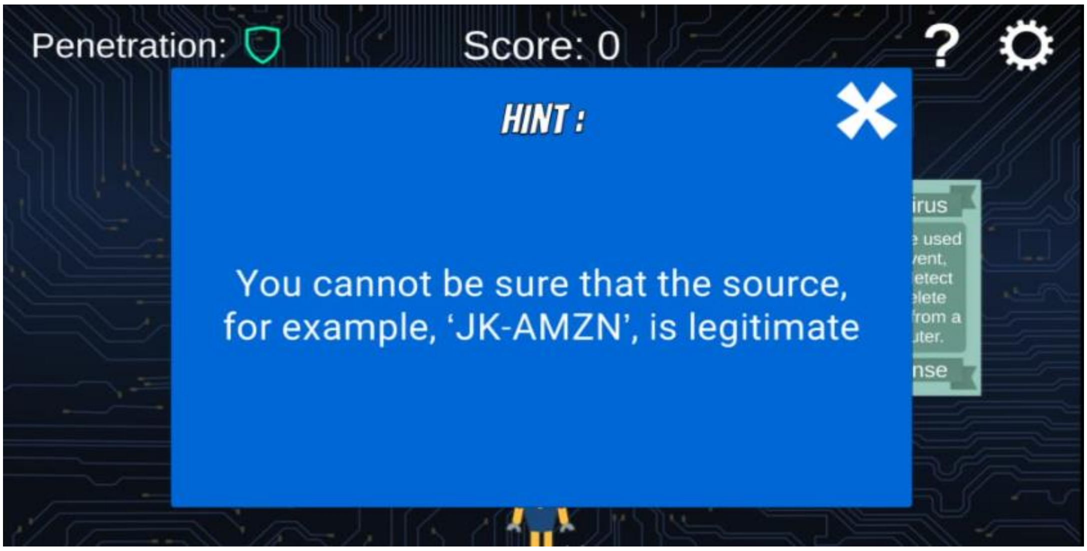
Figure 5.8 Level 1 (1)



You receive an SMS from "Amazon" saying that you've won a voucher worth rs. 10000 and you are asked to click on the link to redeem the voucher

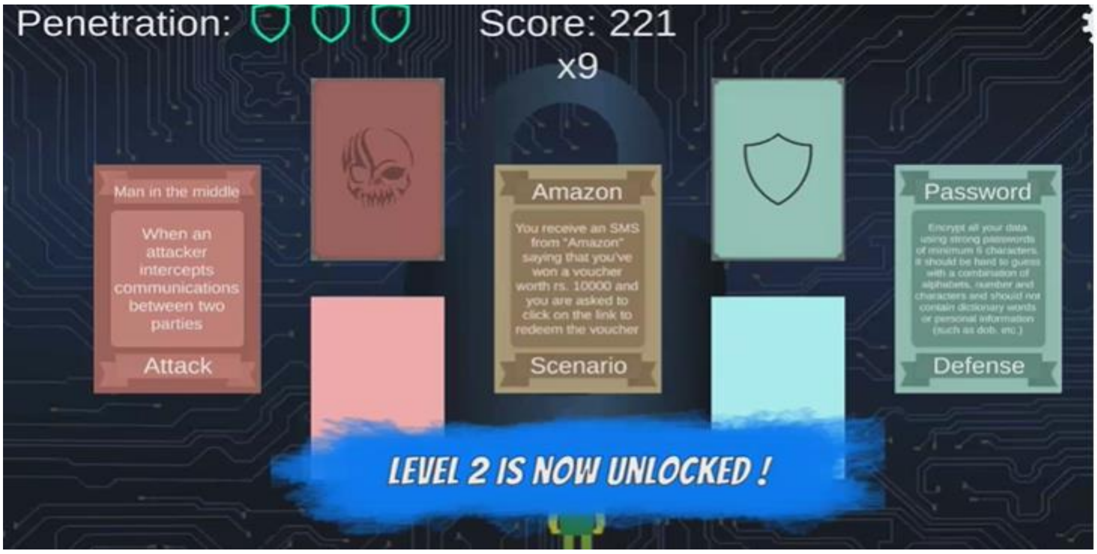
Software used to prevent, scan, detect and delete viruses from a computer.

Figure 5.9 Level 1 (2)



You cannot be sure that the source, for example, 'JK-AMZN', is legitimate

Figure 5. 10 Level 1 (3)



When an attacker intercepts communications between two parties

You receive an SMS from "Amazon" saying that you've won a voucher worth rs. 10000 and you are asked to click on the link to redeem the voucher

Encrypt all your data using strong passwords of minimum 8 characters. It should be hard to guess with a combination of alphabets, number and characters and should not contain dictionary words or personal information (such as dob, etc.)

Figure 5. 11 Level 1 (4)

5.7 Level 1 Tutorial

The user is given an option to view the Level 1 Tutorial on the Main Menu. The tutorial will allow the user to gain an insight into how the game works and how it is played in order to avoid any confusion.

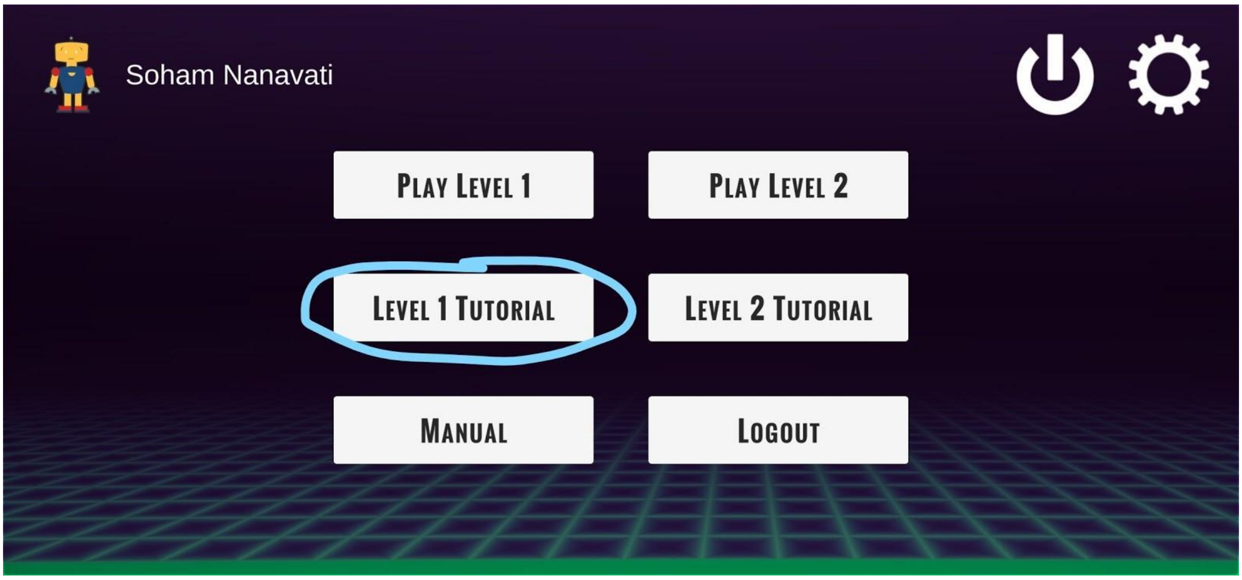
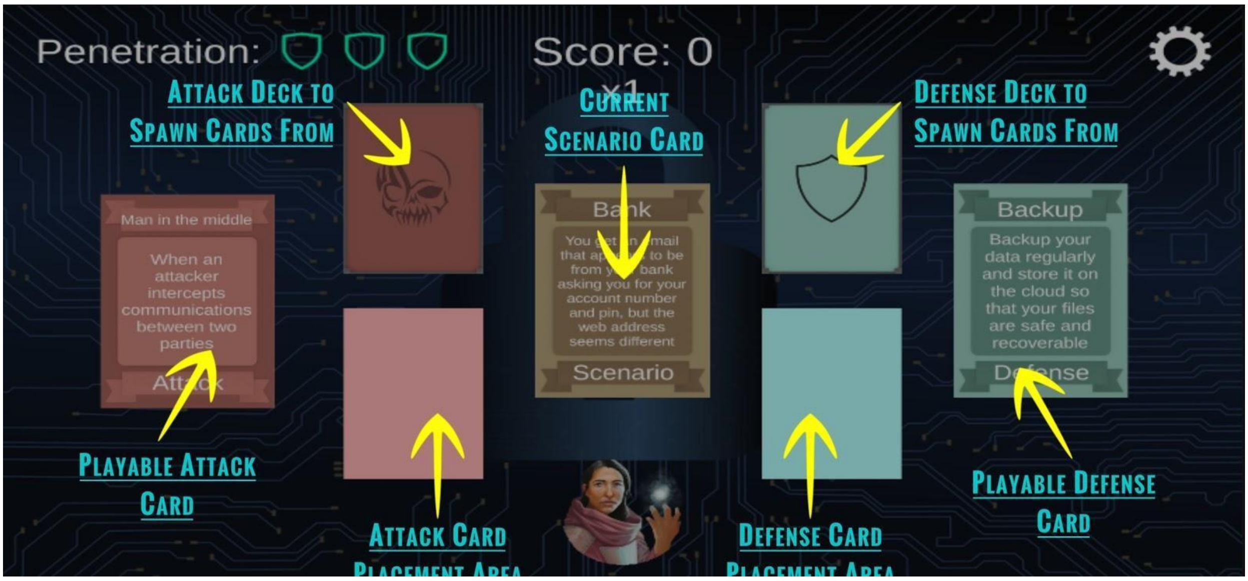


Figure 5. 12 Level 1 Tutorial Main Menu



When an attacker intercepts communications between two parties

You get an email that appears to be from your bank asking you for your account number and pin, but the web address seems different

Backup your data regularly and store it on the cloud so that your files are safe and recoverable

Figure 5. 13 Level 1 Tutorial

5.8 Level 2

After the user scores at least 200 points in Level 1, the “Play Level 2” option for the user is unlocked. The user can now play Level 2 which is a multiplayer 1v1 attack defense game, where the user needs to play the attack and defense cards against each other based on the asset being displayed.

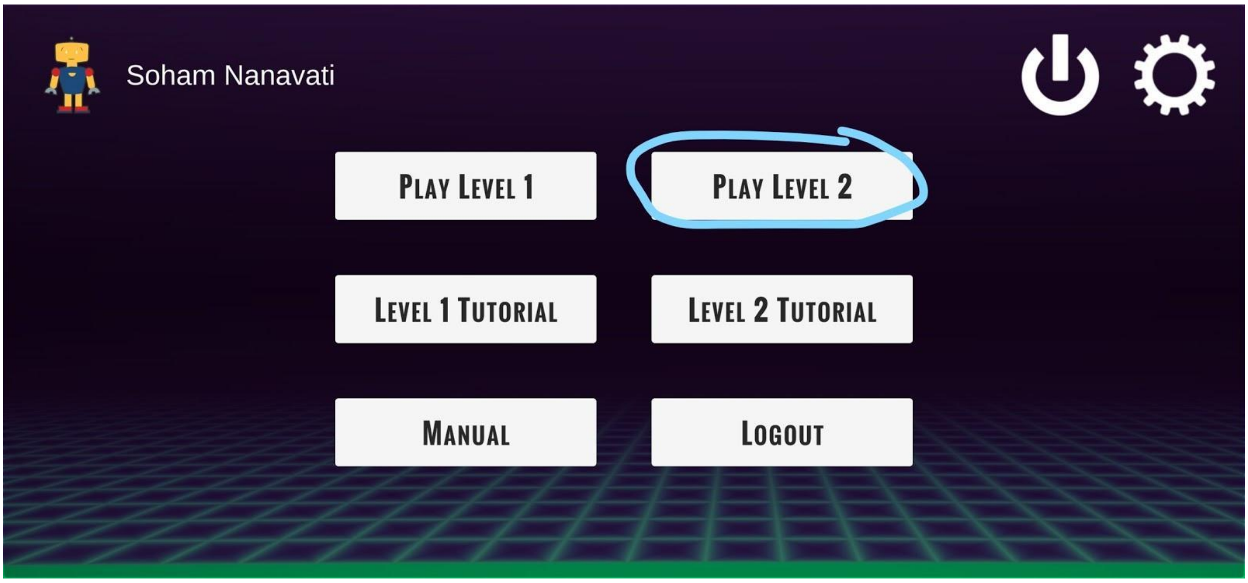


Figure 5.14 Level 2 Main Menu

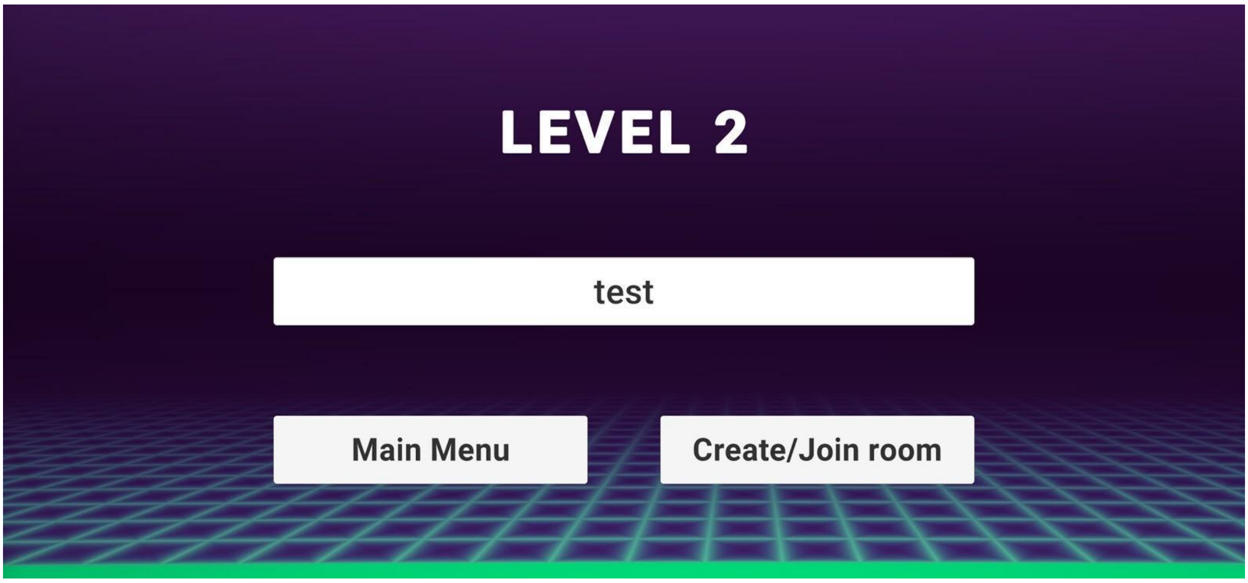
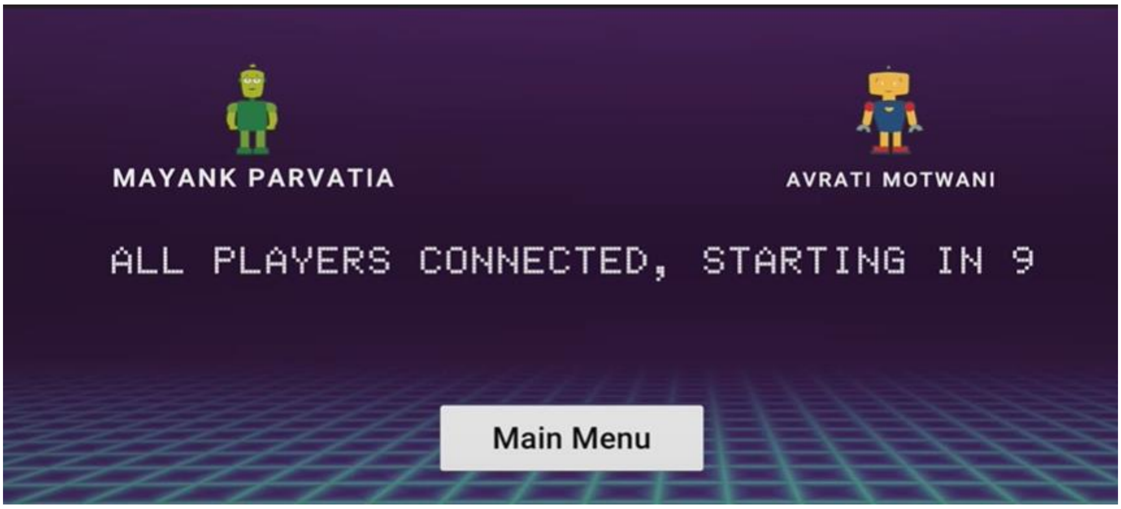
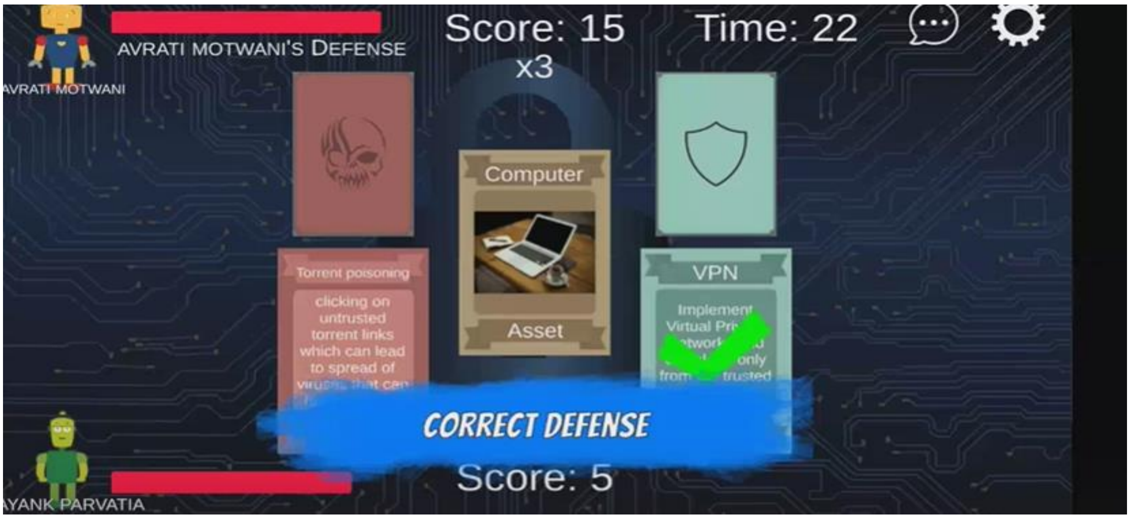


Figure 5.15 Level 2 Room Creation



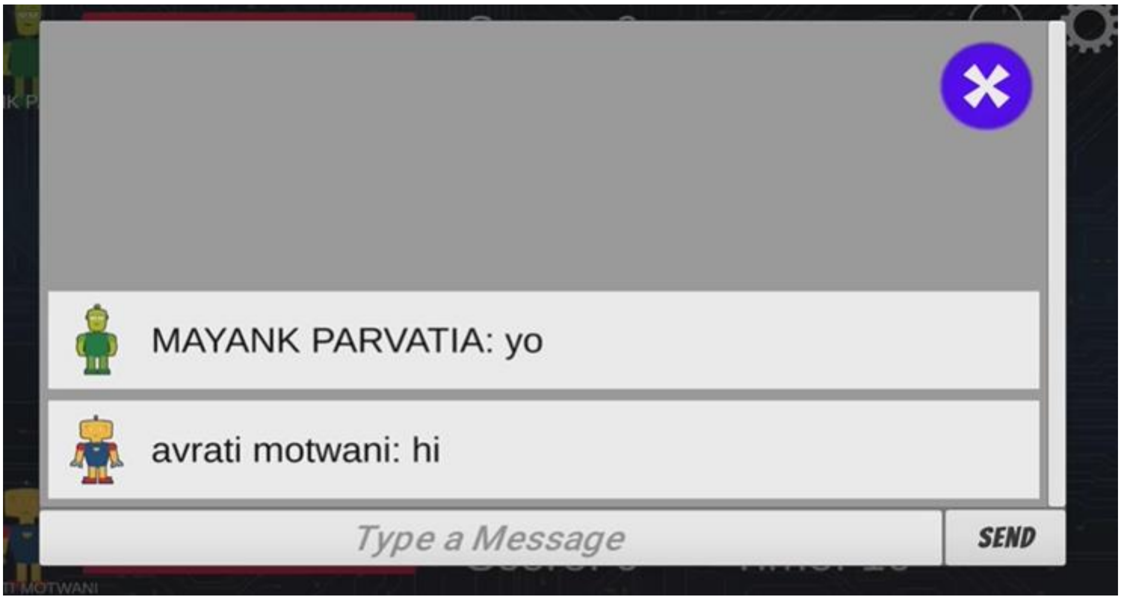
ALL PLAYERS CONNECTED, STARTING IN 9

Figure 5.116 Level 2 Loading Screen



AVRATI MOTWANI'S DEFENSE
AVRATI MOTWANI
Score: 15
x3
Time: 22
Computer
Asset
Torrent poisoning
clicking on untrusted torrent links which can lead to spread of viruses that can
VPN
Implement Virtual Private Network ... only from trusted
CORRECT DEFENSE
Score: 5
MAYANK PARVATIA

Figure 5.17 Level 2 Gameplay



MAYANK PARVATIA: yo

avrati motwani: hi

Figure 5.18 In-Game Chat

5.9 Level 2 Tutorial

After the user scores at least 200 points in Level 2, the “Level 2 Tutorial” option is unlocked on the Main Menu for the user. The user can view the Level 2 Tutorial in order to gain insight into how the game is played so as to avoid any confusion while playing the game.

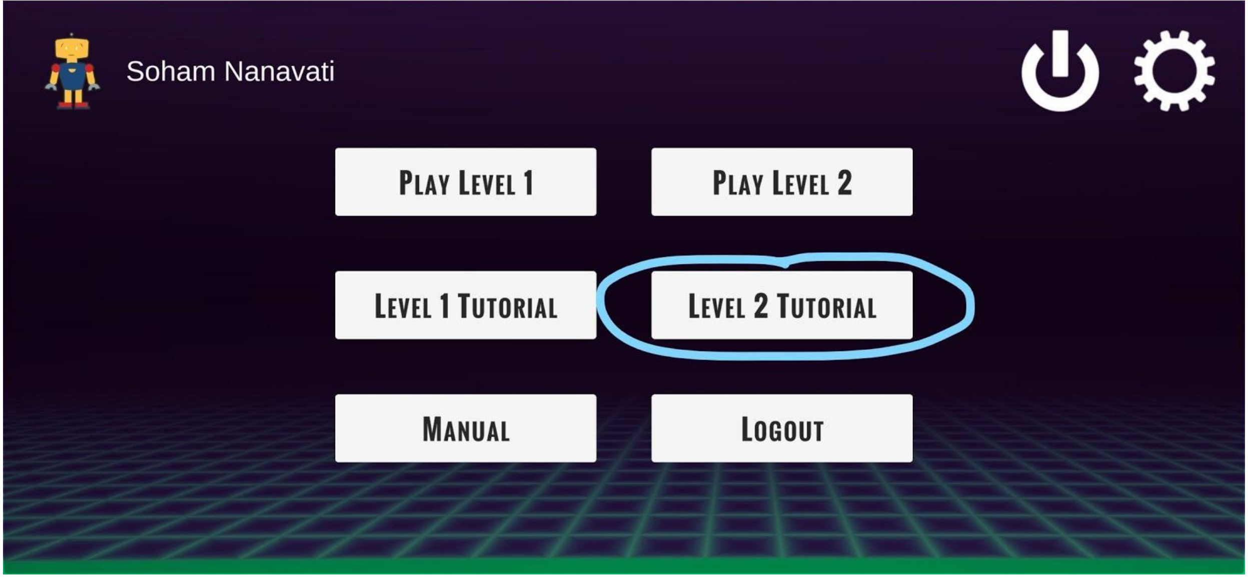


Figure 5.19 Level 2 Tutorial Main Menu

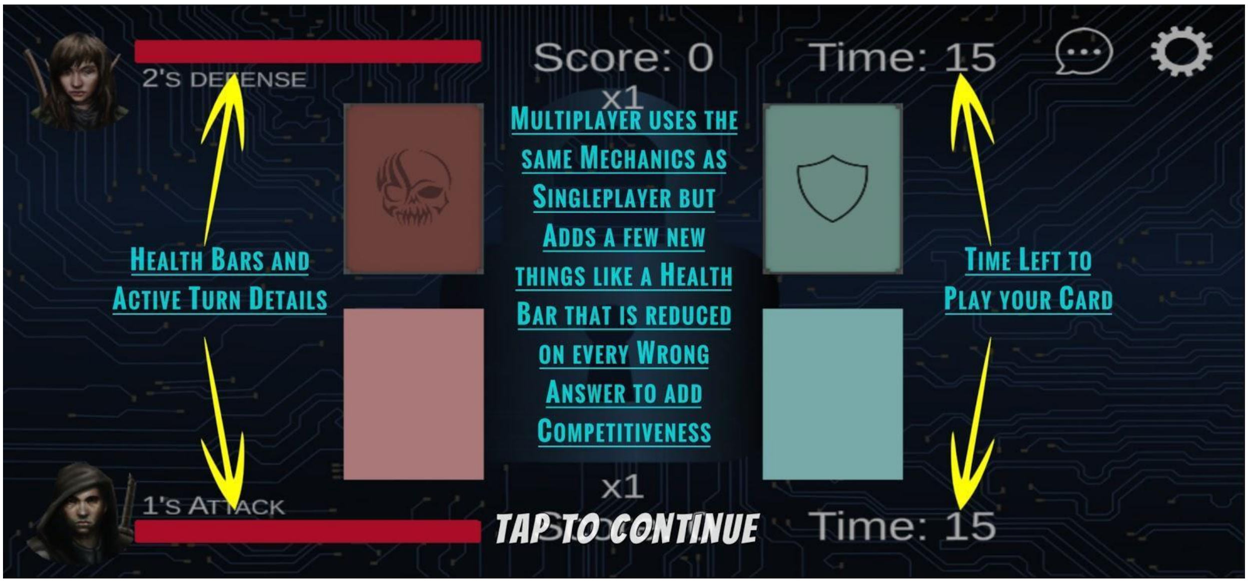

2's Defense
Score: 0
Time: 15
x1
Health Bars and Active Turn Details
Multiplayer uses the same mechanics as Singleplayer but adds a few new things like a Health Bar that is reduced on every Wrong Answer to add Competitiveness
Time Left to Play your Card
x1
1's Attack
Tap to Continue
Time: 15

Figure 5.20 Level 2 Tutorial

5.10 Game Manual

The user has a "Manual" button on the Main Menu that redirects the user to the Manual, which is a scrollable document that contains in depth details about the various cyber security attacks and defenses that are covered in this game. This helps the users to gain more knowledge into various cyber security attacks and defenses.

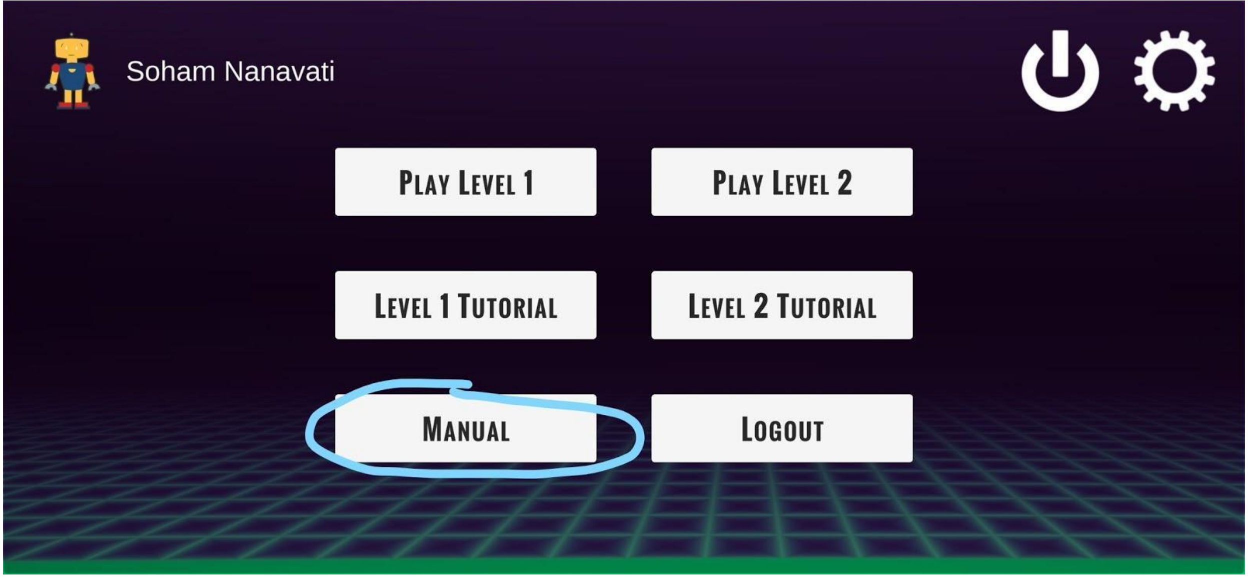
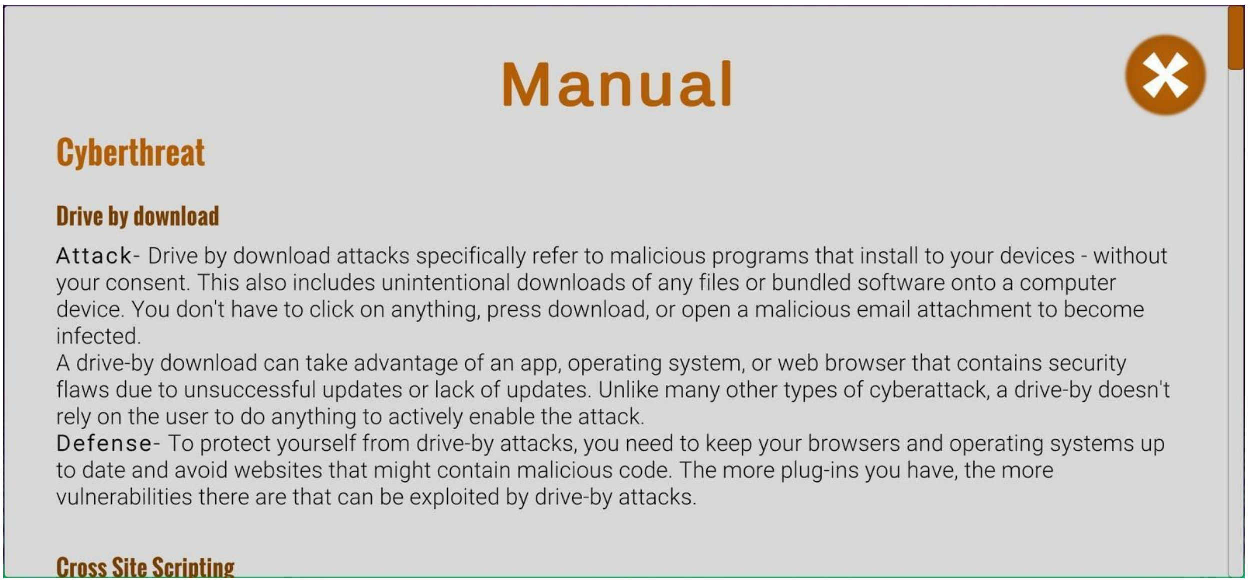


Figure 5.21 Manual-Main Menu



Manual

Cyberthreat

Drive by download

Attack- Drive by download attacks specifically refer to malicious programs that install to your devices - without your consent. This also includes unintentional downloads of any files or bundled software onto a computer device. You don't have to click on anything, press download, or open a malicious email attachment to become infected.
A drive-by download can take advantage of an app, operating system, or web browser that contains security flaws due to unsuccessful updates or lack of updates. Unlike many other types of cyberattack, a drive-by doesn't rely on the user to do anything to actively enable the attack.
Defense- To protect yourself from drive-by attacks, you need to keep your browsers and operating systems up to date and avoid websites that might contain malicious code. The more plug-ins you have, the more vulnerabilities there are that can be exploited by drive-by attacks.

Cross Site Scripting

Figure 5.22 Game Manual

5.11 Game Settings

The settings option is displayed on the top left corner of the screen that allows the user to adjust the game volume settings according to his preference.

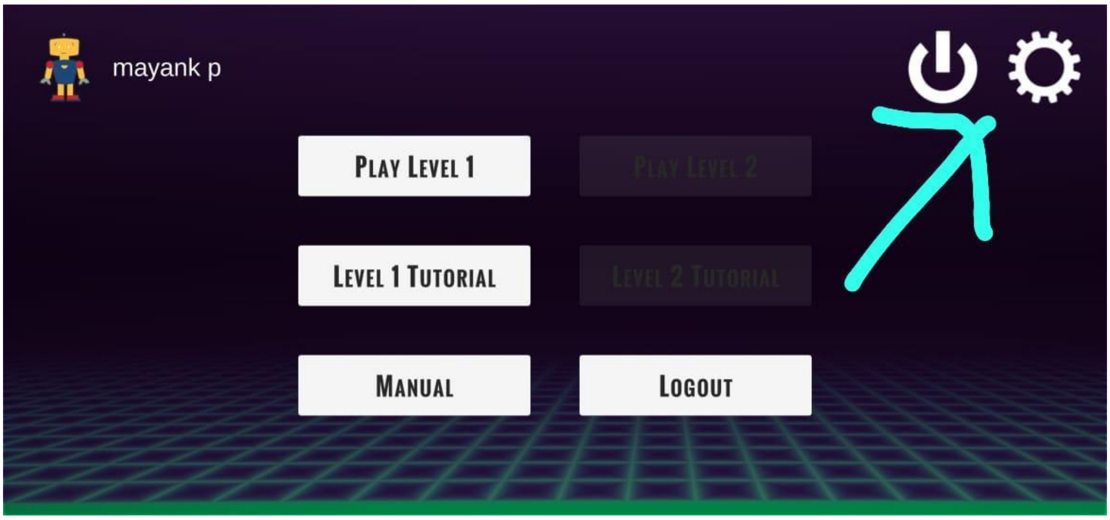


Figure 5.23 Settings Icon

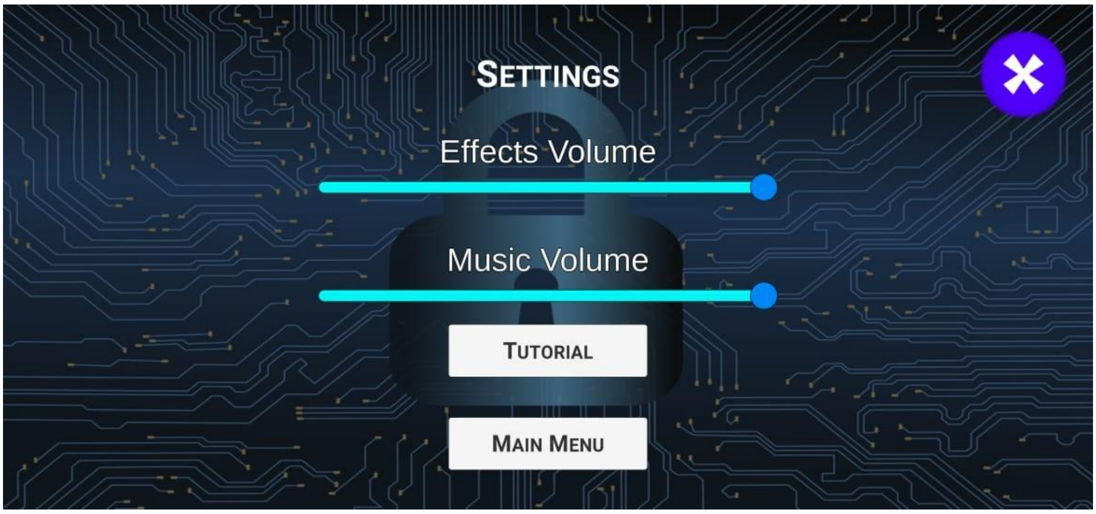


Figure 5.24 Settings Page

5.12 Logout

The “Logout” button on the Main Menu logs the user out of his account. The user is redirected to the Login page from where he/she can login to the application again.

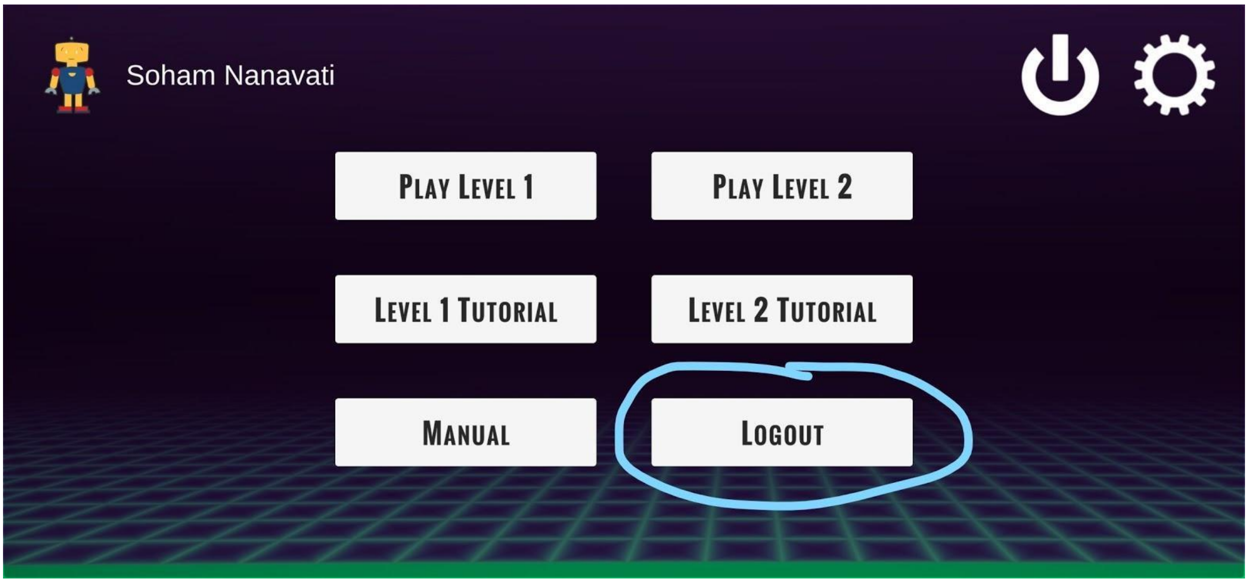


Figure 5.25 Logout Option

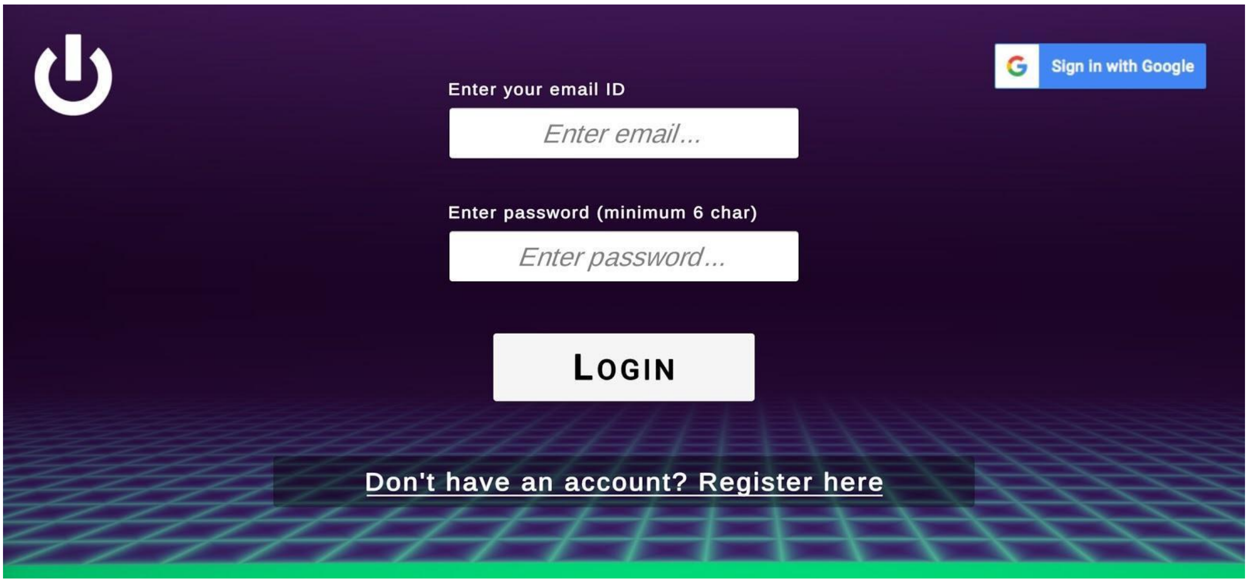


Figure 5.26 Back to Login Page

Chapter 6: Project Test Report:

Sr No.	Test Case	Description	Output
1.	Invalid email/no email while logging in	If the email entered is not in the correct format (missing @, .com/.in, etc.) or if no email has been entered by the user.	Error message “Invalid Email Format”
2.	Invalid password/no password input while logging in	If the password entered is less than 6 characters or if no password has been entered by the user.	Error message “Password needs to be 6 characters long”
3.	Invalid email/no email while registering	If the email entered is not in the correct format (missing @, .com/.in, etc.) or if no email has been entered by the user	Error message “Invalid Email Format”
4.	Registering with same email	If the email entered in the registration form is already in use by another account, i.e. there is already an account with the same email address	Error message “This email address is already in use by another account”
5.	Invalid password/no password while registering	If the password entered is less than 6 characters or no password has been entered by the user	Error message “Password is too short, minimum 6 characters required”
6.	Different inputs in password and confirm password fields while registering.	If the string entered in the password field does not match with the string entered in the confirm password field.	Error message “Passwords don’t match”
7.	Username field blank while registering	If no username has been entered by the user while registering.	Error message “Username cannot be empty”
8.	Successful Registration	Valid inputs given by the user for all the input fields in the registration form	Successful registration, details stored in database, and successful redirection to the main menu.
9.	Login: Using the Registered Email	Valid inputs given by the user for all input fields in the login form and the email and	Successful login and redirection to the main menu

		passwords match with the details in the database	
10.	Login-Google Sign-in	When the user clicks on the Google sign in option to log in via his/her google account	User is given a choice to choose the Google account he wishes to sign in with. For successful log in, redirect to the main menu.
11.	Successful Sign In	After successful login to the application.	Successful redirection to the main menu. Level 2 is initially disabled and will be unlocked after the user scores at least 200 points in Level 1. Users can view his/her scores by clicking on the profile picture. Initial scores are 0.
12.	Level 1-Correct Attack Card Played	On placing the correct attack card for the particular scenario in Level 1	Correct attack pop up is shown on the screen. The user gains points (5*scorestreak value) and the scorestreak is increased. The defense section gets unlocked for the user to play the defense card.
13.	Level 1-Correct Defense Card Played	On placing the correct defense card for the particular scenario in Level 1	Correct defense pop up is shown on the screen. The user gains points (5*scorestreak value) and the scorestreak is increased. A new scenario card is then loaded.
14.	Level 1-Wrong Attack Card Played	On placing the wrong attack card for the particular scenario.	Wrong attack pop up is shown on the screen. 2 points are deducted from the score and the scorestreak is reset. The penetration number (number of attempts) reduces by one.
15.	Level 1-Wrong Defense Card Played	On placing the wrong defense card for the particular scenario.	Wrong defense pop up is shown on the screen. 2 points are deducted from the score and the scorestreak is reset. The penetration number (number of attempts) reduces by one.
16.	Level 1-Hint Used	After two wrong attempts by user (only one attempt is remaining)	The hint (?) icon gets activated on the top right corner of the screen that the user can refer to view the hints for the attack/defense card for the particular scenario.
17.	Level 1-All Attempts utilized	When the user is not able to match the cards and all the attempts have been consumed.	The correct attack and defense cards are displayed to the user automatically and the next scenario is loaded.

18.	Level 2 Unlocked Notification	When the user scores at least 200 points in level 1	The user receives a pop up that Level 2 is now unlocked. The “Play Level 2” and “Level 2 Tutorial” buttons are activated on the Main Menu and the user can now play Level 2.
19.	Level-2 Room Creation/Lobby Page	When the user clicks on Play Level 2 button on Main Menu when it gets activated.	The user is redirected to the lobby where he/she has to enter the room name to either create a new room, or join an existing room.
20.	Level 2-No Room Name Entered	When the “Enter room name” field is left blank by the user when the user is trying to “Create/Join room”	Error message “The room name cannot be empty” is shown.
21.	Level 2-Creating and Joining a Room	When the user enters the room name and clicks on “Create/Join room”	If the room has not been created, a new room is created and the user has to wait for the opponent to connect. If the room is already created, the user joins that room and the countdown for the game starts.
22.	Level 2-Wrong Defense Card Played	When the attacker places the attack card, and the defender places the wrong defense corresponding the attack	“Wrong defense” pop up is shown on the screen. The attacker gains points (5*scorestreak value) and his scorestreak increases. The defender loses health. The attacker and defender roles are then switched for the next round.
23.	Level 2-Correct Defense Card Played	When the attacker places the attack card, and the defender places the correct defense corresponding to the attack.	“Correct defense” pop up is shown on the screen. The defender gains points (5*scorestreak value) and his scorestreak increases. The attacker loses health. The attacker and defender roles are switched for the next round.
24.	Level 2-No Attack Card Played	When the attacker does not play any attack card before the timer runs out.	The defender plays any defense card and an error message “No attack played” pops up. The attacker loses health and the defender gains points (5*scorestreak value). The attacker and defender roles are switched for the next round. Also, the scorestreak for attacker resets to 1 and for defender the scorestreak multiplier increases.
25.	Level 2-No Defense Card Played	When the defender does not play any defense card before the timer runs out.	The error message “No defense played” pops up. The defender loses health and the attacker gains points (5*scorestreak value). The

			attacker and defender roles are switched for the next round. Also, the scorestreak for defender resets to 1 and for attacker the scorestreak multiplier increases.
26.	Level 2-No Cards Played	When neither the attacker nor the defender play any card before the timer runs out.	The error message “No card played” pops up and that round is discarded. Also, the scorestreak multiplier resets back to 1. We then move on to the next round.
27.	Level 2-End Game	When one player’s health reduces to zero.	The end game screen is displayed which displays the winner of the game and the points of both the players.
28.	Level 2- In-Game Chat	The players can send messages to each other via the chat option.	The player can view the message and at the same time can send a message too.
29.	Level 2- Player Disconnected	When either of the two players leaves the game.	The game ends and a screen showing “Player disconnected” is displayed.
30.	Main Menu-Updated Player Profile	Progress and score after playing Level 1 and Level 2.	The user can view his/her updated scores after playing level 1 and level 2 by clicking on the profile picture at the top left corner of the screen. The scores and progress are updated there.
31.	Main Menu-Level 1 Tutorial	On clicking the Level 1 Tutorial button on the Main Menu	The user will can view a detailed overview of how to play Level 1 to get an overview
32.	Main Menu-Level 2 Tutorial	On clicking the Level 2 Tutorial button on the Main Menu after it has been activated.	The user will can view a detailed overview of how to play Level 2 to get an overview
33.	Main Menu-Manual	On clicking the Main Menu.	The user can view the Manual which gives the user insights into the various cyber security threats and their defenses.
34.	Main Menu-Logout	On clicking the Logout button on the Main Menu.	The user will be logged out of his/her account and will be redirected to the login page.

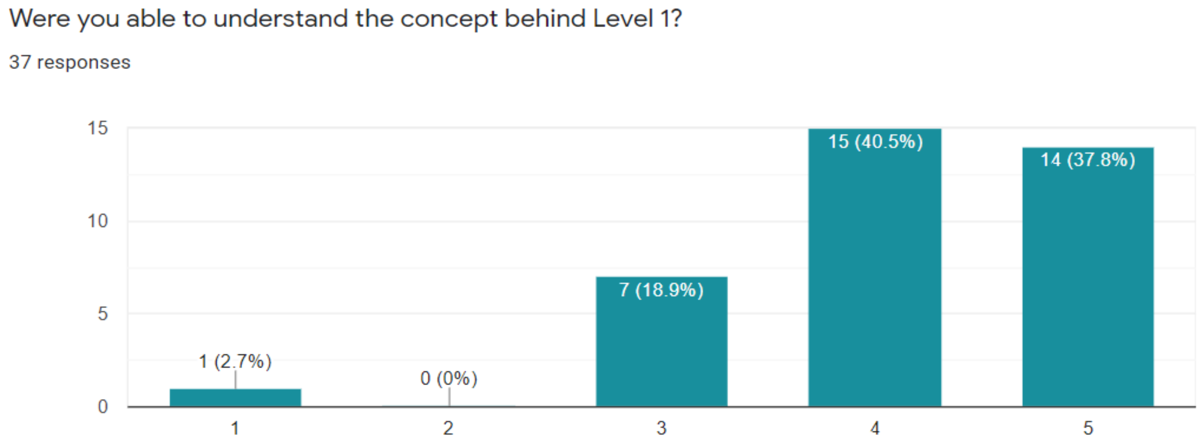
Table IV Test Cases

Chapter 7: Results and Discussions

On completion of development and system testing of the mobile application, it was tested by external users. User testing was also carried out to ensure all requirements were met. The testing involved giving the users a particular task to be performed on the application (e.g., Play Level 1 to unlock Level 2 and play with other users) and gauging their comfort and ease of use by conducting a think aloud study while they were performing the task and a post-task interview. Based on this testing, the users were able to successfully perform the tasks given to them (Shuffle the deck and match the cards according to the scenario) and were comfortable with the layout and navigation of the application.

Thus, users were receptive of the application, particularly its Multiplayer feature. Users were able to find out the matching pair out of their respective decks from which they had an idea about which attack and defense would be correct. Hint feature in the Level 1 helps them to think more in case they played 2 wrong cards and can move on to the different scenario. Overall, the user had a great experience using the game application and they were happy about the features and functions that were used in the app.

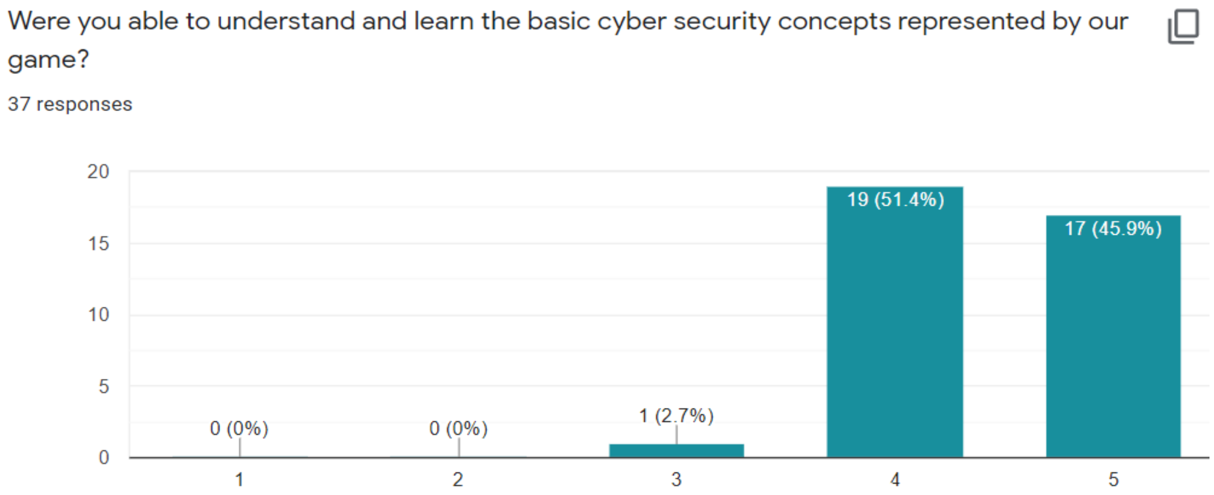
Following are the results of a survey which was conducted for Level 1.



Were you able to understand the concept behind Level 1?

37 responses

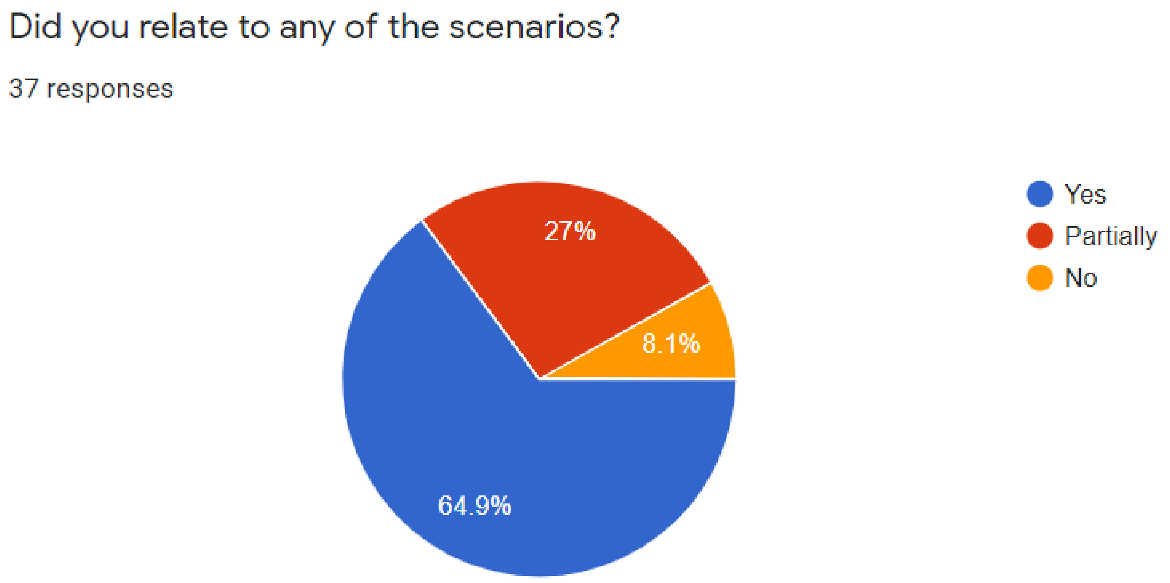
Figure 7.1 Survey Q1



Were you able to understand and learn the basic cyber security concepts represented by our game?

37 responses

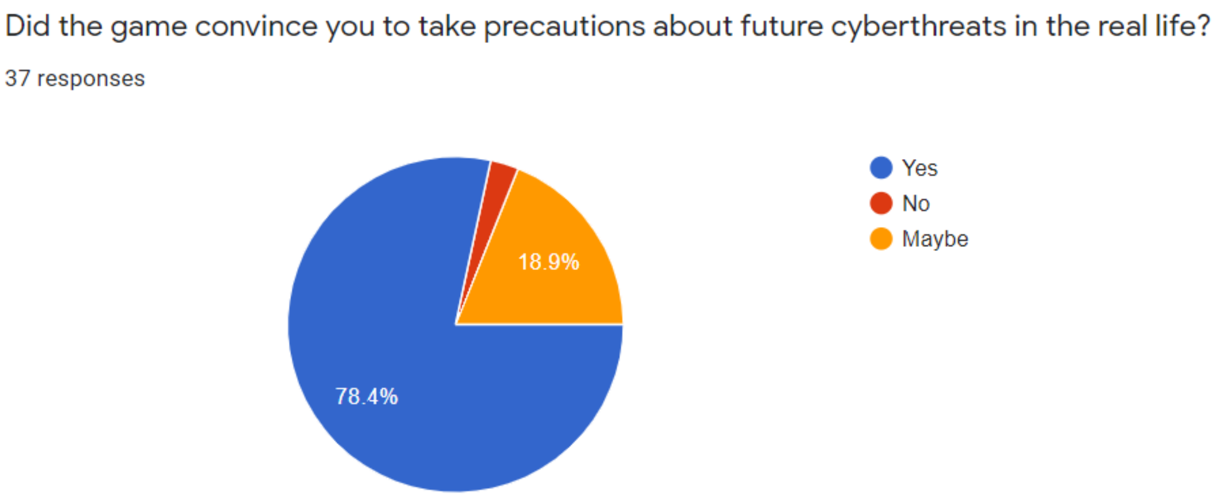
Figure 7.2 Survey Q2



Did you relate to any of the scenarios?

37 responses

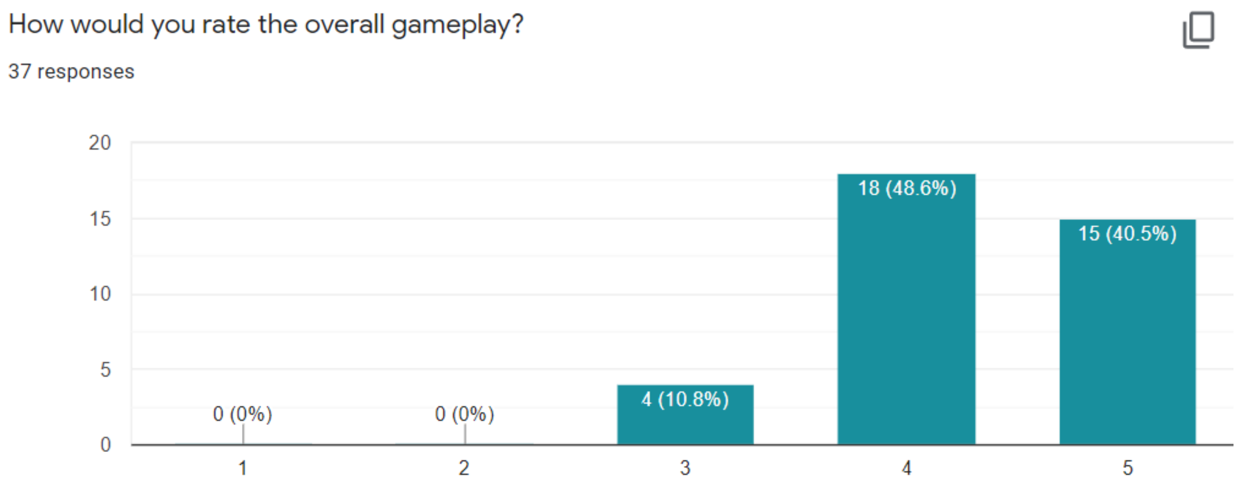
Figure 7.3 Survey Q3



Did the game convince you to take precautions about future cyberthreats in the real life?

37 responses

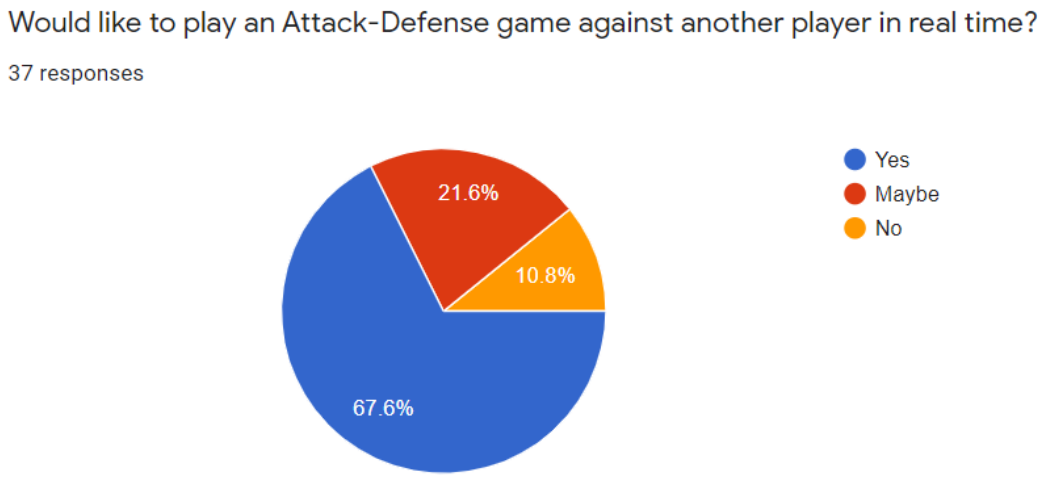
Figure 7.4 Survey Q4



How would you rate the overall gameplay?

37 responses

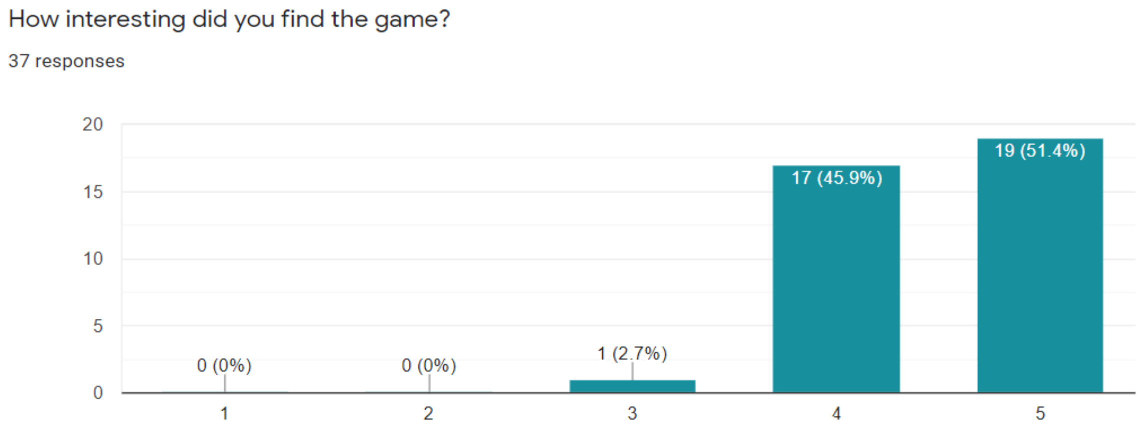
Figure 7.5 Survey Q5



Would like to play an Attack-Defense game against another player in real time?

37 responses

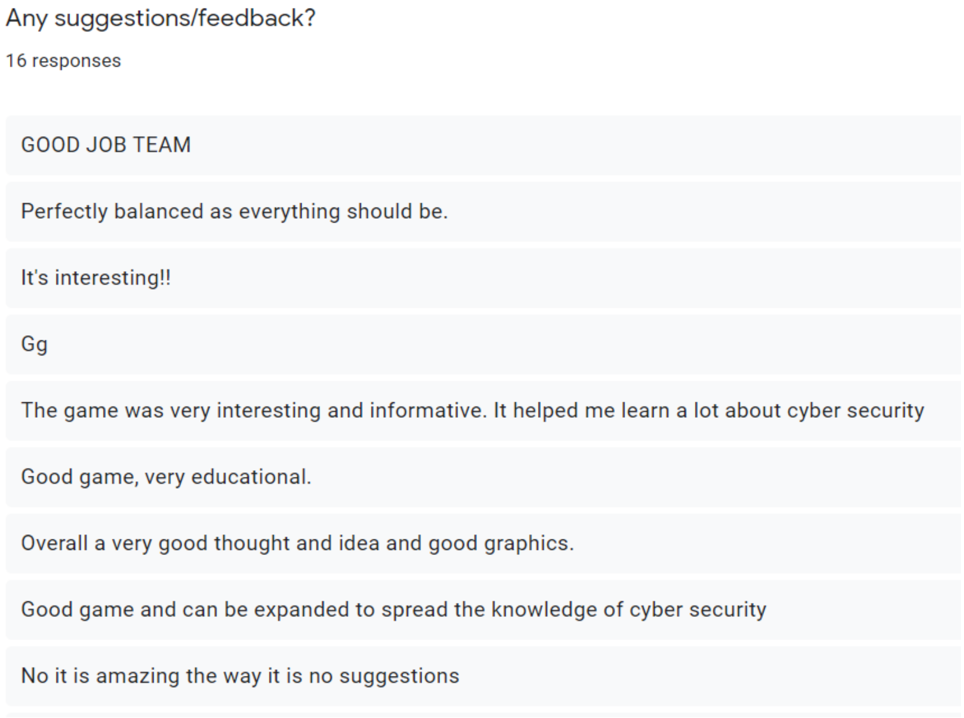
Figure 7.6 Survey Q6



How interesting did you find the game?

37 responses

Figure 7.7 Survey Q7



Any suggestions/feedback?

16 responses

GOOD JOB TEAM

Perfectly balanced as everything should be.

It's interesting!!

Gg

The game was very interesting and informative. It helped me learn a lot about cyber security

Good game, very educational.

Overall a very good thought and idea and good graphics.

Good game and can be expanded to spread the knowledge of cyber security

No it is amazing the way it is no suggestions

Figure 7.8 Feedback

Chapter 8: Conclusion

8.1 Conclusion:

Nowadays, we can see that there are many cyber threats happening around the world and majority of the people are unaware of these attacks and somehow become the victim. Clearly there is need for this application which will help the users to learn about the threats (Phishing and its types, Malwares, Web based attacks, Denial of Service, Ransomwares, Identity thefts, Physical manipulation, Password attacks and Man-in-the-middle attacks) and how to defense against or probably take future precautions in order to avoid mishaps.

8.2 Future Avenues:

So far, we have created a Cyber Security Awareness Card Game that has the following features: -

- Registration and Login, Including Google OAuth sign in.
- Two Levels: -
  - Level 1, to learn about the major cyber security attacks and defenses.
  - Level 2, 1v1 multiplayer attack-defense game with chat option enabled.
- Identified most common cyber security attacks and defenses on the basis of recent trends.
- Level 1 and Level 2 tutorials.
- Progress tracker.
- Compatible on Android devices and Desktop.

For our future work, we tend to do add the following functionalities: -

- Make the game compatible on iOS and MacOS devices as well.
- Come up with a system to automatically spot latest trends in cyber security attacks and defenses to allow us to keep our game up-to-date.
- Add various multiplayer options such as: -
  - 2v2
  - Playing in teams, etc.

References

[1] F. Alotaibi, S. Furnell, I. Stengel and M. Papadaki, "Enhancing cyber security awareness with mobile games," 2017 12th International Conference for Internet Technology and Secured Transactions (ICITST), Cambridge, 2017, pp. 129-134, doi: 10.23919/ICITST.2017.8356361.

[2] L. Tobarra et al., "Game-based Learning Approach to Cybersecurity," 2020 IEEE Global Engineering Education Conference (EDUCON), Porto, Portugal, 2020, pp. 1125-1132, doi: 10.1109/EDUCON45650.2020.9125202.

[3] K. Graham *et al.*, "Cyberspace Odyssey: A Competitive Team-Oriented Serious Game in Computer Networking," in *IEEE Transactions on Learning Technologies*, vol. 13, no. 3, pp. 502-515, 1 July-Sept. 2020.

[4] A. Yasin, L. Liu, T. Li, R. Fatima and W. Jianmin, "Improving software security awareness using a serious game," in *IET Software*, vol. 13, no. 2, pp. 159-169, 4 2019.

[5]H. Kettani and P. Wainwright, "On the Top Threats to Cyber Systems," 2019 IEEE 2nd International Conference on Information and Computer Technologies (ICICT), Kahului, HI, USA, 2019, pp. 175-179, doi: 10.1109/INFOCT.2019.8711324.

[6]https://www.udemy.com/course/learn-to-code-trading-card-game-battle-system-with-unity-3/

[7] https://www.udemy.com/course/build-multiplayer-games-with-unity-and-photon-pun-2/

[8] https://www.udemy.com/course/unitycourse/

[9] https://www.coursera.org/learn/introduction-programming-unity

[10] https://www.coursera.org/learn/more-programming-unity

Appendix

Rules

Level 1

- Upon initiating the Level, the user's Score is 0, the Penetration value (number of attempts) is 3 (which is reset after every new scenario is loaded), and the Scorestreak multiplier value is 1.
- The user is presented with a Scenario Card in the center.
- Based on the scenario, the user first needs to choose an Attack card from the deck indicated by a skull head.
- If the user chooses the correct Attack, the user gains points equal to 5*the Scorestreak multiplier value, the Scorestreak multiplier value increases by 1 and the user can move on to choose the Defense Card.
- If the user chooses the wrong attack, the user loses 2 points, the Scorestreak multiplier is reset to 1, and the Penetration value decreases by 1. If the number of attempts has not been exhausted by the user, the user needs to choose another Attack Card.
- After the user chooses the correct Attack Card, the user needs to choose a Defense card from the deck indicated by the shield.
- If the user chooses the correct Defense card, the user gains points equal to 5*the Scorestreak multiplier value, the Scorestreak multiplier value increases by 1 and a new scenario card is presented to the user.
- If the user chooses the wrong Defense card, the user loses 2 points, the Scorestreak multiplier is reset to 1, and the Penetration value decreases by 1. If the number of attempts has not been exhausted by the user, the user needs to choose another Defense Card.
- After the number of Penetration reduces to 1, the user is provided with an option to view a Hint to help them choose the correct Attack/Defense Card.
- After all the Penetration attempts have been exhausted by the user, the correct Attack and Defense Cards for the particular scenario are shown to the user.
- In the next round, a new Scenario Card is presented to the user, the Penetration values are reset, and the user needs to repeat the same process again.
- The user needs to score at least 200 points in Level 1 in order to unlock Level 2.

Level 2

- The users need to Create/Join a room.
    - If the user enters a room name that has already been created by some user and has only one player, the user joins that room and the game starts.
    - Else, if the user enters a room name that does not exist, a new room is created while the user needs to wait for another player to join.
- Upon initiating the Level, each user's Score is 0, each user's Health Bar (indicated with a red bar) is full, each user's Scorestreak multiplier value is 1, and each user is given a Timer of 30 seconds on every turn.

- The players are presented with an Asset Card in the center (Phone, PC, Wi-Fi router).
- Now both the players are assigned with a role, either Attack or Defense which is indicated during their turn. These roles keep on switching on a turn-by-turn basis.
- The player with the Attack Role(P1) selects and plays any Attack Card based on the Asset Card. Here a free choice is given to the player.
- Upon the completion of P1's turn, the player with the Defense Role(P2) selects and plays a Defense Card to defend against P1's attack.
- When both the players have completed their turns, the scores are evaluated as follows: -
  - If the Defense Card played by the player with the Defense Role(P2) is not the correct defense for the Attack played by the player with the Attack Role(P1), P1 gains points equal to 5*the Scorestreak multiplier value, P1's Scorestreak value increases by 1, and P2's Health Bar reduces.
  - If the Defense Card played by the player with Defense Role(P2) is the correct defense for the Attack played by the player with the Attack Role(P1), P2 gains points equal to 5*the Scorestreak multiplier value, P2'sScorestreak value increases by 1, and P1's Health Bar reduces.
- Both the players need to play their respective Attack and Defense cards within the time limit which is 30 seconds, else the players will not gain any points.
- After the completion of each round, the user roles are switched and a new Asset Card is presented on the screen.
- The game ends when either of the player's Health Bar reduces to zero. The other player wins the round by default.

How to Play

Level 1

- The user is presented with a Scenario Card in the center.
- Upon reading the scenario, the user first needs to select an Attack Card from the deck indicated by a skull head.
- If the Attack Card is right, there will be a message indicating the same
- But if the Attack Card is wrong then the user needs to select another card from the deck.
- Upon selecting the correct Attack Card, the user now needs to select a Defense Card indicated by the shield.
- If the Defense Card is right, there will be a message indicating the same
- But if the Defense Card is wrong then the user needs to select another card from the deck.
- If both the cards are correct then the user is assigned scores and the user is presented with a new scenario.

Level 2

- This level is an online 1v1 Multiplayer level, meaning players will compete among themselves in real time
- A player needs to create or join a lobby room.
- After both players have joined the room, the game starts.
- The players are presented with an Asset Card in the center (Phone, PC, Wi-Fi router)
- Now both the players are assigned with a role, either Attack or Defense which is indicated during their turn. These roles keep on switching on a turn-by-turn basis.
- The player with the Attack Role(P1) selects and plays an Attack Card. Here a free choice is given to the player.
- Upon the completion of P1’s turn, the player with the Defense Role(P2) selects and plays a Defense Card to defend against P1’s attack.
- When both the players have completed their turns, scores will be assigned and the Asset Card in the center will change.
- Furthermore, the roles for players P1 and P2 will also change. So now P2 has to play the Attack Card and P1 will play the Defense Card.
- The players also have been provided with an In-game Chat option.