Voice Control Turn-Based Role Playing Game Development Using Unity Speech Recognition

Sebastian[1], Harya Bima Dirgantara[2]
[1,2] Informatics, Institut Teknologi dan Bisnis Kalbis, Jakarta, Indonesia
harya.dirgantara@kalbis.ac.id

Accepted 31 October 2023
Approved 08 January 2024

***Abstract*— This research endeavors to create and develop a turn-based role-playing video game utilizing speech recognition as an innovative means to introduce a novel experience in video games. The game is designed and implemented using the Unity game engine. The game prototype produced in this study is intended to provide a unique experience when playing games, namely using voice recognition to try to beat opponents. The researcher adopts the Extreme Programming methodology, compassing four stages: Planning, Design, Develop and Testing/Release. The culmination of this research is a turn-based role-playing game featuring speech recognition controls intended for Windows computers equipped with a microphone. The game offers players fresh and innovative gameplay by incorporating speech recognition. Based on the closed beta trial on three respondents, the result was that there was an influence from the playing room. Two words are difficult for the game to detect: "molten spear" with an average detection rate of 40% and "chakra magic" with an average detection rate of 33%.**

***Keywords*— turn-based, role-playing video game speech recognition, extreme programming**

I. Introduction

Games usually require the user to engage in new and exciting systems, to manipulate the form and content displayed on the screen in accurate or real-time. In this way, the player's relationship with the media shifts from a more passive recipient of information to a more active involvement [1]. They all involve the player interacting with the on-screen action via multiple device inputs, which change depending on the platform the game is being played. Most computer games use a mouse and keyboard, and other games use a joystick/gamepad, but other controllers also exist. A joystick or gamepad is designed primarily for games on a game console, a device to play games other than computers [2].

Game development itself is defined as a medium of fictitious activity, unpredictable and not productive with rules, with time and space limits, and without obligations [3]. These elements are: players, goals, procedures or methods, rules, resources, conflicts, boundaries, then results [4].

Game genres are specific game categories that are related to similar game characteristics. Genre is usually not defined by the setting, story, or media of the game but by how players interact with the game. An example of an RPG game can be interpreted as a Role Playing Game, and players must control a character in a fictional theme. The character's purpose usually depends on the narrative or theme of the game. In RPGs, players usually have a strategic gameplay method by providing resources such as levels, types of magic, weapon types, and other mechanics. Conflicts in regular RPGs are in the form of computer-controlled enemies and even other players if the game is online. Conflict resolution and an understanding of the limits or boundaries of the game will result in the player's goals or goals, which can be in the form of victory or the end of the story if the game has narrative elements [5][6].

Game development itself cannot be separated from development in the world of computing, especially in the world of computer software development. One example is the use of artificial intelligence or artificial intelligence. Artificial intelligence in most modern games fulfills three basic needs: the ability to move the character, decide where to move, and think tactically or strategically [7].

The horror game Phasmopobhia uses the same AI properties, only now it uses speech recognition, a form of machine learning. Ghosts in this game can respond based on the words spoken by the player through the microphone, making players more careful with what they say [8]. The history of speech recognition implementation in games was done by shouting orders to their squad mates, who followed the player's orders [9].

Most games still use vital combinations on the joystick and keyboard/mouse to communicate with and control our characters on the screen. Things like this can provide the potential for a new way of playing the game; due to the previous statement that most games still use button-shaped controls, the following games that will be developed can provide a new exploration of usage: game controller or controller [10].

History of the implementation of voice recognition or speech recognition in games has played barking orders to their squad mates, and they follow the player's orders. Almost all games still use a combination of buttons on the joystick and keyboard/mouse to communicate with and control our characters on the screen. Things like this can provide the potential for new ways of playing games, due to the previous statement, that most games still use button controls, the game that will be developed below can provide a new exploration in the use of game controllers.

Based on the background described, this research's main problem is building a turn-based RPG game with Desktop-based speech recognition elements.

II. Methodology

A. State of The Art

Several previous studies are relevant and have become a reference for this research. Research by Ahmad et al. [11]. This research developed an educational game for learning English pronunciation based on speech recognition. This research underlies how speech recognition can be a control medium in games. Further research by Mustaquim [12], this paper aims to find a standard way of using voice commands in games that uses a speech recognition system in the back end, and that can be universally applied for designing inclusive games.

Research by Aguirre-Peralt et al. [13] stated that persons with disabilities have limitations in accessing certain types of hardware. Therefore speech recognition can be a medium that can help them use the software. Further research by Jung et al. [14], the development of voice user interface (VUI), which has become popular recently, presents new possibilities for human-computer interaction, especially in game development.

Based on these four studies, these research produces a game that uses voice recognition media as the primary control medium. The game genres from these research vary; therefore, the authors designed a desktop-based turn-based role-playing game in this study. The game prototype produced in this study is intended to provide a unique experience when playing games, namely using voice recognition to try to beat opponents. The development of this game uses the extreme programming framework.

B. Turn-based Game

In a turn-based game, players only run the action on the turn or turn away. Action can generally be restricted; for example, a player can only do two any action in turn. A turn-based strategy allows players to think without a limited time. An essential aspect of Turn Based Strategy Games is the presence of opponents or enemies. Enemies are also considered by players in making strategic decisions, and players who play this game will always show that they are smarter than their opponents [15].

C. Role-playing Game

In a role-playing game, the player can control a character according to his role in a fictional world. The role in question is the functionality of the character, such as a wizard, sniper, knight, or healer. RPG genre games usually have elements of action, adventure, or strategy [16].

Role-playing games allow the player to immerse themselves in the character's situation. Role Playing Games (RPG) continue their rich history in storytelling by embracing innovative ways to vary and report stories. Characters tend to be rich, the gameplay is extended, and character management is technical in RPGs [17].

D. Speech Recognition

Speech recognition, also known as Automatic Speech recognition (ASR), is a technology applied to software to receive input in spoken words. This technology allows a device to recognize and understand spoken words by digitizing the words and matching the digital signal with a particular pattern stored in a device [18] [19] [20].

E. Windows Speech Recognition

Microsoft has developed the Speech API since 1993. The Microsoft team has released the Speech API (SAPI) 5.3 with Windows Vista, which was very powerful and valuable. This allows developers to easily speech-enable Windows Forms applications and apps based on the Windows Presentation Framework [21].

F. Extreme Programming

Extreme Programming, commonly called XP, is a rapid software development approach. The XP method focuses on object-oriented development; this method has four frameworks for software development activities: the Planning Phase, the Design Phase, the Develop Phase, and finally, the Testing Phase before returning to the Planning Phase to start the next sprint [22]. The XP process is shown in Fig. 1.

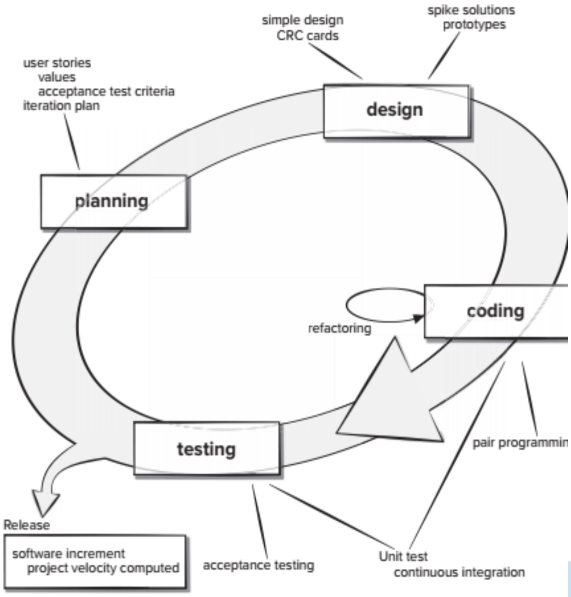


Fig. 1. The extreme programming process [22]

The Extreme Programming processes are as follows [22]:

- Planning: The planning activity (the planning game) begins with a requirements activity called listening. Listening leads to creating a set of "stories" that describe the required output, features, and functionality for software to be built.
- Design. XP recommends immediately creating an operational prototype of that portion of the design.
- Coding. After user stories are developed and preliminary design work is done, the team does not move to code but instead develops a series of unit tests that will exercise each story to be included in the current release (software increment).
- Testing. XP acceptance tests, also called customer tests, are specified by the customer and focus on overall system features and functionality that are visible and reviewable by the customer. They are derived from user stories implemented as part of a software release.

III. Result and DiscussiON

A. Planning Phase

This stage generates game formal elements as well as dramatic elements. The game's formal elements are shown in Table I.

TABLE I. Game Formal Element

Element	Description
Players	Single Player vs. Game, players defeat enemies controlled by artificial intelligence, taking turns. Players will use a microphone and speech to act.
Objectives	Players try to defeat the enemy with all their might, a skill that can be cast until the enemy's health runs out; all at once, try not to make the player's health run out.
Procedures	The game is seen from a two-dimensional side view or side scroller. The player can see himself and the enemy. At the time of the player's turn, the player can say the word through the microphone or see what skills the player has before saying it. Players can also see players' and enemies' health and mana bars.
Flow	If the player wins, the player will move on to the next level, with more difficult enemies.
Resources	Players have a health bar. Players have a list of skills that can be chanted to perform actions.
Boundaries	Players are limited by a system of taking turns or turn-based with the enemy after acting.
Outcome	Players will find the game's final level to fight the last enemy; if the player manages to defeat him, the player will win. If the player loses against the enemy, the player will repeat the match (restart) without returning to the initial level (unless the player exits the game application).

In addition to formal elements, this game also has dramatic elements. Dramatic elements are elements that can show the emotions of players when playing a game. The game's dramatic elements are shown in Table II.

TABLE II. Game Dramatic Element

Element	Description
Challenge	In the game, the player has to defeat artificial intelligence-controlled enemies. Players are given several skills that must be spoken through the player's microphone to fight or defend against enemy attacks. Players will take turns with the enemy acting. If successful, the player will advance to the next level and fight more difficult enemies; players will also get new skills every time they level up.
Theme	Turn-based role-playing game
Premise	Defeat enemies using the power of sound.
Character	a. The protagonist is a wizard with his spell book. b. Antagonists are monsters of various forms.

B. Design Phase

This stage describes several game design elements, including determining the words players will speak when engaging in turn-based combat. This stage will cover the game architecture plan or design pattern, the use of game assets, and the technology used. The important thing in the controller of this game is the pronunciation of keywords of some skills because it is an essential tool in developing this game. The following is a list of the skills in this game, and their keywords are shown in Table III.

TABLE III. GAME KEYWORDS FOR SPEECH CONTROL

Keyword	Type	Description
Explosion	Damage	*Summons an explosive fire*
Black hole	Damage	*Summons an unknown force of power*
Molten spear	Damage	*Summons a molten spear from hell*
Splash	Damage	*Spawn a geyser of water*
Spike	Damage	*Summon earth spikes*
Tornado	Damage	*Summons a wind tornado*
Bless	*Charge/Heal*	*Heals player for 500*
Hand of God	*Charge/Heal*	*Heals player for 1000*
Chakra Magic	*Charge/Mana*	*Restore mana for 300*
Power up	*Buff/Attack*	*Increase Attack by 150 for five turns*
Protect me	*Buff/Defense*	*Increase Defense by 20 for five turns*
I am motivated	*Buff/Critical Chance*	*Increase Crit Chance by 25% for three turns.*
I need more power	*Buff/Critical Damage*	*Increase Critical damage by 100% for three turns.*
Curse you	*Debuff/Attack*	*Decrease Attack by 100 for three turns*
corruption	*Debuff/Defense*	*Decrease Defense by 15 for three turns*

Table III displays the keywords used as game controls. This keyword was tested on three users to find out whether the system read this keyword correctly or not.

At this stage, the design of game wireframes is also carried out. Wireframe is the basic appearance of a software-based interface or user interface. The game itself is an artwork with interactive computer software, and the wireframe on the game project must be designed to understand how the game looks and to make an impression and message, reinforcing the elements and premise of the game that will be played. The wireframes are shown in Fig. 2 to Fig. 7.

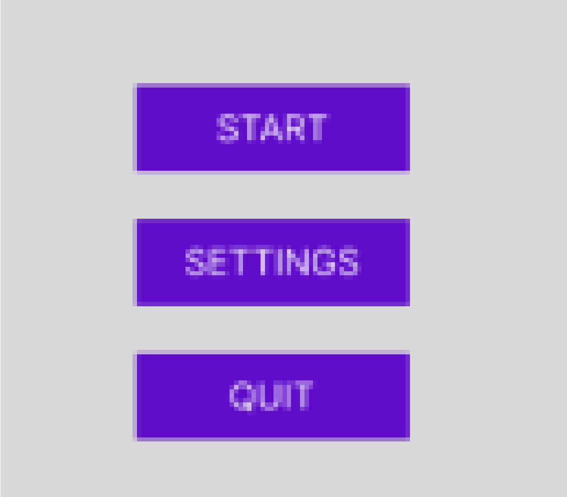
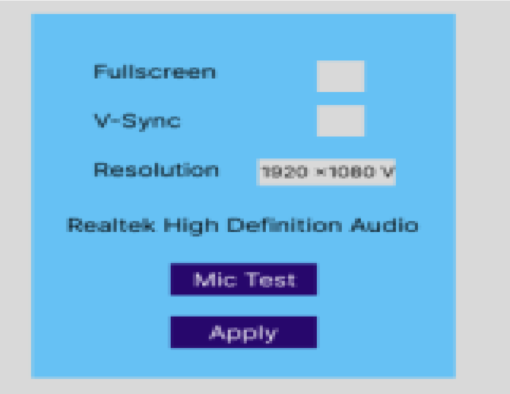


Fig. 2. The game main menu

Fig. 2 displays the start menu display of the game starting. The interaction between the player and the game is given buttons to navigate the display. Fig. 3. displays the settings menu. This menu is used so players can adjust the game's appearance, starting from the resolution size, full screen, and others. Players can also test the microphone before playing on this menu.



Fullscreen □

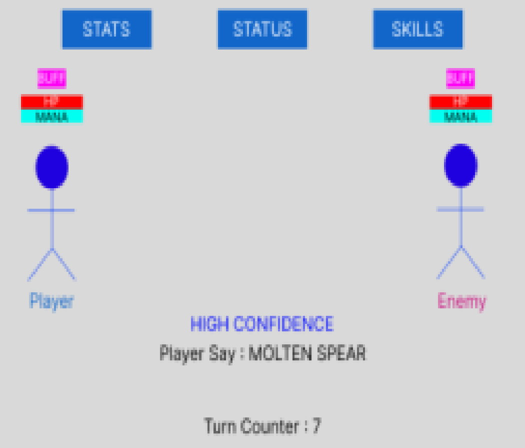
V-Sync □

Resolution 1920 × 1080 ∨

Realtek High Definition Audio

Fig. 3. The game setting menu

Fig. 4 shows the battle scene. This figure shows the main game; the player positioned on the left will face the enemy on the right. Complete display with menus - other menus that are useful for the course of the game. This display is also complete with subtitles to help players learn several things, for example, the words the player says.

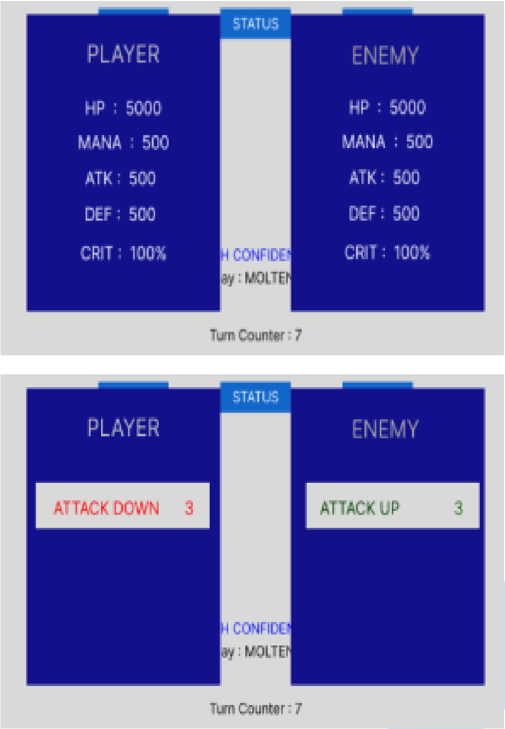


HIGH CONFIDENCE

Player Say : MOLTEN SPEAR

Turn Counter : 7

Fig. 4. The battle scene



PLAYER

HP : 5000
MANA : 500
ATK : 500
DEF : 500
CRIT : 100%

ENEMY

HP : 5000
MANA : 500
ATK : 500
DEF : 500
CRIT : 100%

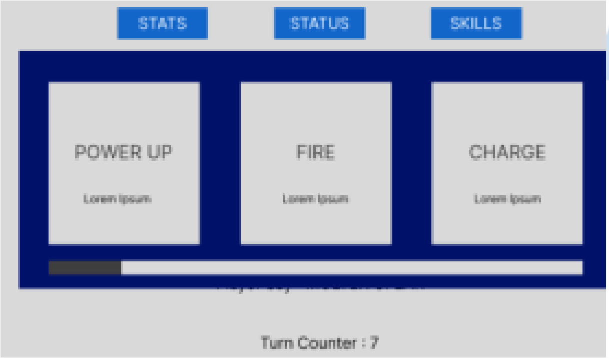
ATTACK DOWN 3

ATTACK UP 3

Turn Counter : 7

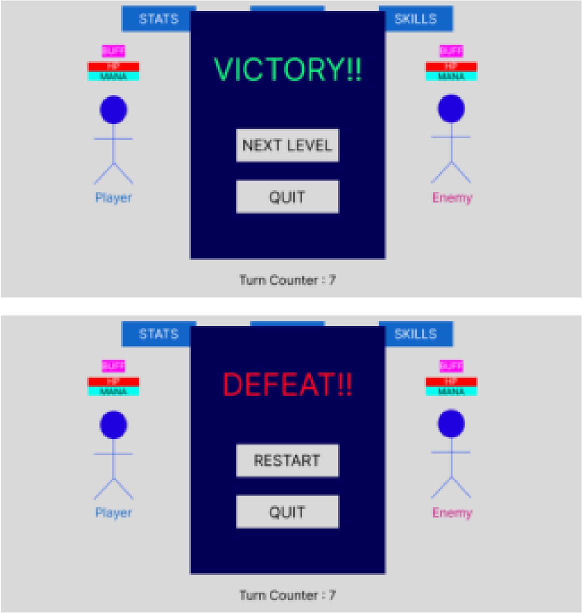
Fig. 5. The status effect menu

Fig. 5 shows the menu status and the status effects. This view focuses on providing additional information on the game, such as stats or statistics from players and enemies, for example: comparing the amount of strength of players and enemies and seeing what adverse effects are on both characters. Fig. 6 shows the skills list. This display shows the player's skills and all the information.



Turn Counter : 7

Fig. 6. The skills list menu



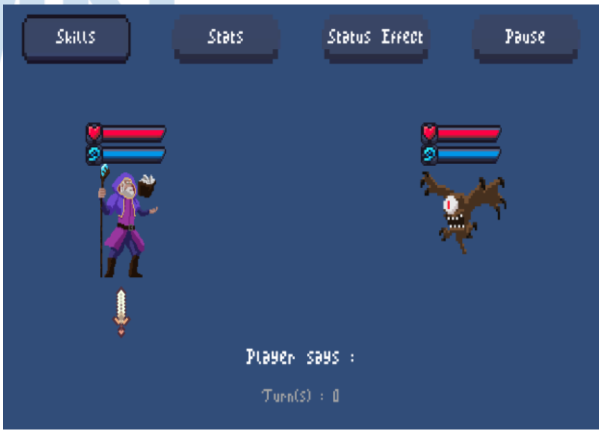
Turn Counter : 7

Fig. 7. The result scene

Fig. 7 shows the result scene. This display will appear if the player wins or loses. If the player wins, the player is given the option to continue to the next level. If the player loses, the choice is replaced by repeating the game.

C. Coding Phase

This stage produces the game development process, which already includes implementing the results from the two previous stages that must be programmed into the game. The result of the coding phase is shown in Fig. 8 to Fig. 12.

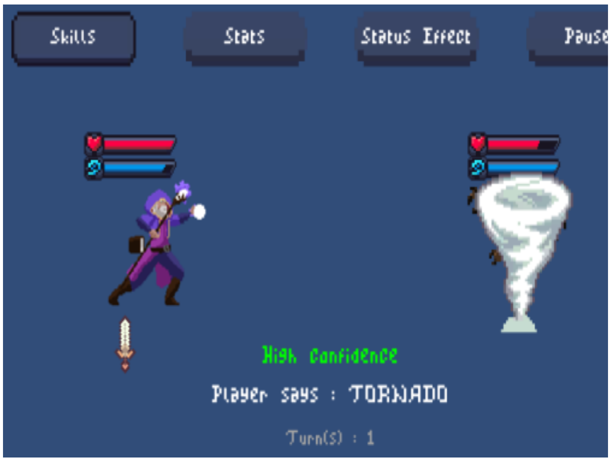


Player says :

Turn(s) : 0

Fig. 8. The game start battle

Fig. 8 shows the game start battle. The player's character is on the left while the enemy is on the right side. Fig. 9 features players attacking using "Tornado" keyword. A high-confidence statement indicates that voice input is heard very well.

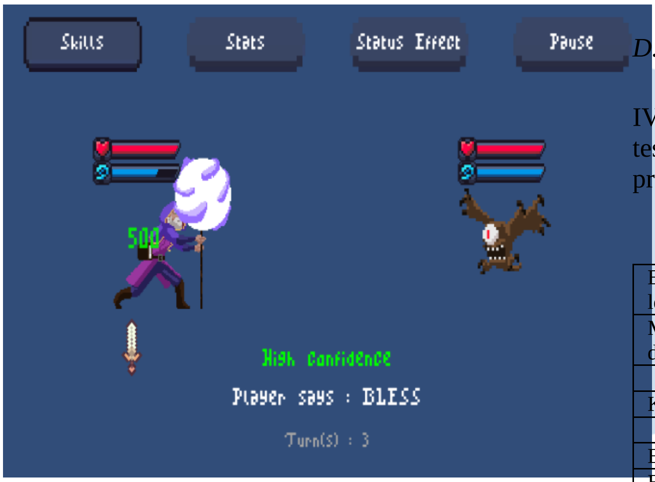


High Confidence

Player says : TORNADO

Turn(s) : 1

Fig. 9. The in-game battle (1)



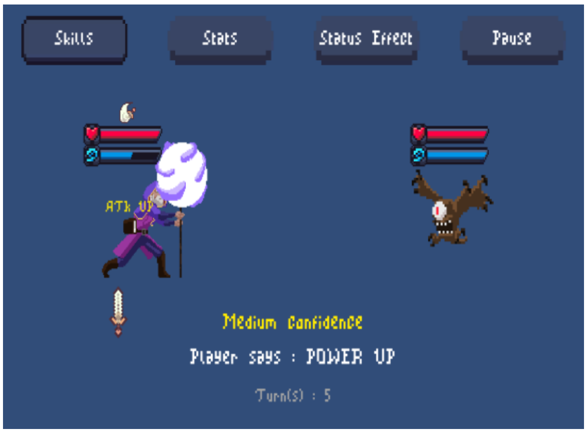
High Confidence

Player says : BLESS

Turn(s) : 3

Fig. 10. The in-game battle (2)

Fig. 10 features player buff self using "Bless" keyword. A high-confidence statement indicates that voice input is heard very well. Fig. 11 features player buff self using "Power Up" keyword. A medium-confidence statement indicates that voice input is heard quite well.



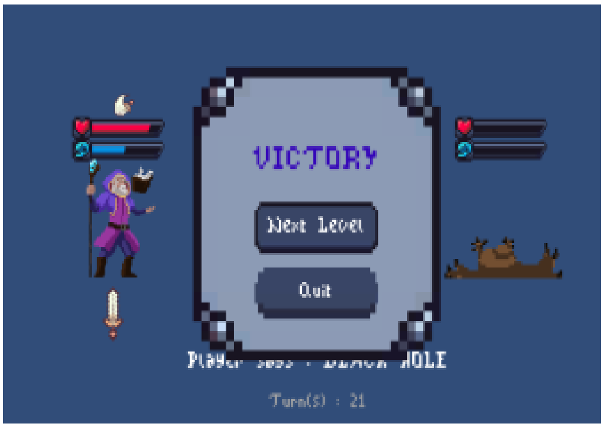
Medium Confidence

Player says : POWER UP

Turn(s) : 5

Fig. 11. The in-game battle (3)

Fig. 12 shows the winning result. When the enemy's health bar is empty, the player wins.



Player says : BLACK HOLE

Turn(s) : 21

Fig. 12. The in-game battle (4)

D. Testing Phase

At this stage, a closed beta test will be tried. Tables IV, V, and VI are examples of the results of closed beta testing by three testers, aiming to try the keyword pronunciation performance of the game specifically.

TABLE IV. FIRST TESTER RESULT

English level	Frequent speaker						
Microphone device	Steelseries Actris 5 (tested in a quiet public room)						
Keyword	Trial #					Result	Complexity
	1	2	3	4	5		
Explosion	v	v	x	x	x	40%	Easy
Black hole	v	x	v	x	x	40%	Easy
Molten spear	x	x	x	x	x	0%	Hard
Splash	x	x	x	x	x	0%	Hard
Spike	v	v	v	v	v	100%	Easy
Tornado	v	v	v	v	v	100%	Easy
Bless	v	v	v	v	v	100%	Easy
Hand of God	v	x	v	v	x	60%	Easy
Chakra Magic	x	v	x	x	v	40%	Hard
Power up	v	v	v	v	x	80%	Easy
Protect me	v	v	x	x	v	60%	Easy
I am motivated	v	v	v	v	x	80%	Easy
I need more power	v	v	v	x	x	60%	Easy
Curse you	v	x	v	x	x	40%	Neutral
Corruption	v	v	v	v	v	100%	Easy

TABLE V. SECOND TESTER RESULT

English level	Frequent speaker						
Microphone device	Rexus Fonix F26M (tested in a private room)						
Keyword	Trial #					Result	Complexity
	1	2	3	4	5		
Explosion	v	v	v	v	x	80%	Easy
Black hole	v	v	v	v	v	100%	Easy

Molten spear	x	v	v	v	v	80%	Easy
Splash	v	v	v	v	v	100%	Easy
Spike	v	v	v	v	v	100%	Easy
Tornado	v	x	x	v	x	40%	Hard
Bless	v	v	v	v	v	100%	Easy
Hand of God	v	x	x	x	v	40%	Hard
Chakra Magic	v	v	x	x	x	40%	Hard
Power up	v	v	v	v	x	80%	Easy
Protect me	v	v	v	v	v	100%	Easy
I am motivated	v	v	x	v	x	60%	Neutral
I need more power	v	v	v	v	v	100%	Easy
Curse you	v	v	v	v	v	100%	Easy
Corruption	v	v	v	v	v	100%	Easy

TABLE VI. THIRD TESTER RESULT

English level	Native speaker						
Microphone device	Steelseries Actris 5 (tested in a crowded private room)						
Keyword	Trial #					Result	Complexity
	1	2	3	4	5		
Explosion	x	x	v	x	x	20%	Hard
Black hole	v	v	v	v	x	80%	Easy
Molten spear	x	v	x	x	v	40%	Hard
Splash	v	v	v	v	v	100%	Easy
Spike	v	v	v	v	v	100%	Easy
Tornado	v	v	v	v	v	100%	Easy
Bless	v	v	v	v	v	100%	Easy
Hand of God	v	v	v	v	v	100%	Easy
Chakra Magic	x	x	x	x	v	20%	Hard
Power up	v	v	v	v	v	100%	Easy
Protect me	v	v	v	v	v	100%	Easy
I am motivated	v	v	v	v	v	100%	Easy
I need more power	v	v	v	v	v	100%	Easy
Curse you	v	v	v	v	x	80%	Easy
Corruption	v	v	v	v	v	100%	Easy

IV. CONCLUSION

The development of the application prototype has been completed. From this development process, it can be concluded as follows. There is a statement regarding the quality of sound input. This is influenced by the pronunciation of the word and the quality of the microphone used. Based on closed beta testing from three respondents, it can be concluded that the keywords that are difficult to detect in this game are "molten spear" with an average detection rate of 40%, and "chakra magic" with an average detection rate of 33%.

V. REFERENCES

[1] N. D. Bowman, *Video Games: A Medium That Demands Our Attention*. Routledge, 2018. doi: 10.1088/1751-8113/44/8/085201.

[2] A. H. Cummings, "The Evolution of Game Controllers and Control Schemes and their Effect on their games," *17th Annu. Univ. Southhampt. Multimed. Syst. Conf.*, pp. 1–8, 2007, [Online]. Available: http://mms.ecs.soton.ac.uk/2007/papers/6.pdf

[3] N. Esposito, "A short and simple definition of what a videogame is," *Proc. DiGRA 2005 Conf. Chang. Views - Worlds Play*, 2005.

[4] L. Nacke, "The Formal Systems of Games and Game Design Atoms," *Acagamic*, 2014. http://acagamic.com/game-design-course/the-formal-systems-of-games-and-game-design-atoms/

[5] E. Adams, *Fundamentals of Game Design*. San Francisco: Pearson Education Inc., 2014. doi: 10.1016/j.lfs.2005.03.030.

[6] N. Tringham, "Science fiction video games," *Sci. Fict. Video Games*, pp. 1–506, 2014, doi: 10.1201/b17460.

[7] I. Millington, *AI for Games*, 3rd ed. CRC Press, 2019. doi: 10.1201/9781351053303.

[8] K. Games, "Phasmophobia," 2020. https://sea.ign.com/phasmophobia (accessed Feb. 20, 2023).

[9] Summalinguae.com, "Voice Controlled Games: The Rise of Speech Technology in Gaming," 2021. https://summalinguae.com/language-technology/voice-controlled-games/ (accessed Nov. 14, 2022).

[10] D. Strzałko, "Voice Controlled Games – The approach and challenges of implementing speech recognition and voice control in games," *Position Commun. Pap. 16th Conf. Comput. Sci. Intell. Syst.*, vol. 26, pp. 229–230, 2021, doi: 10.15439/2021f143.

[11] F. I. Ahmad, T. Afirianto, and M. A. Akbar, "Perancangan Game Pembelajaran Pengucapan Bahasa Inggris Berbasis Pengenalan Suara," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 9, pp. 9299–9304, 2019.

[12] M. M. Mustaquim, "Automatic speech recognition- an approach for designing inclusive games," *Multimed. Tools Appl.*, vol. 66, no. 1, pp. 131–146, 2013, doi: 10.1007/s11042-011-0918-7.

[13] J. Aguirre-Peralta, M. Rivas-Zavala, and W. Ugarte, "Speech to Text Recognition for Videogame Controlling with Convolutional Neural Networks," no. Icpram, pp. 948–955, 2023, doi: 10.5220/0011782900003411.

[14] H. Jung, S. So, C. Oh, H. J. Kim, and J. Kim, "TurtleTalk: An educational programming game for children with voice user interface," *Conf. Hum. Factors Comput. Syst. - Proc.*, pp. 1–6, 2019, doi: 10.1145/3290607.3312773.

[15] F. S. Sulaeman and D. P. Aji, "Turn Based Strategy Games to Hone Your Knowledge of Indonesian Culture Based on Android," *Proc. 1st Paris Van Java Int. Semin. Heal. Econ. Soc. Sci. Humanit. (PVJ-ISHESSH 2020)*, vol. 535, pp. 47–50, 2021, doi: 10.2991/assehr.k.210304.011.

[16] J. P. Zagal and S. Deterding, *Definitions of "Role-Playing Games,"* no. May 2020. 2018. doi: 10.4324/9781315637532-2.

[17] L. Grace, "Game Type and Game Genre," 2005.

[18] R. Pahwa, H. Tanwar, and S. Sharma, "Speech Recognition System – A Review," *IOSR J. Comput. Eng.*, vol. 18, no. 04, pp. 01–09, 2016, doi: 10.9790/0661-1804020109.

[19] A. Kumar and V. Mittal, "Speech recognition: A complete perspective," *Int. J. Recent Technol. Eng.*, vol. 7, no. 6, pp. 78–83, 2019.

[20] R. Matarneh, S. Maksymova, V. V Lyashenko, and N. V

Belova, "Speech Recognition Systems: A Comparative Review," *IOSR J. Comput. Eng.*, vol. 19, no. 5, pp. 71–79, 2017, doi: 10.9790/0661-1905047179.

[21] V. Këpuska and G. Bohouta, "Comparing Speech Recognition Systems (Microsoft API, Google API And CMU Sphinx)," *Int. J. Eng. Res. Appl.*, vol. 07, no. 03, pp. 20–24, 2017, doi: 10.9790/9622-0703022024.

[22] R. S. Pressman and B. R. Maxim, *Software Engineering*, 9th ed. New York: McGraw-Hill Education, 2019.