[EPICAST]

By

[Yueshen Li] yueshen.20@intl.zju.edu.cn

[Jianli Jin] jianli.20@intl.zju.edu.cn

[Zihao Zhu] zihaoz.20@intl.zju.edu.cn

[Di Wu] di.20@intl.zju.edu.cn

Design Document for ECE 445, Senior Design, [Spring 2024]

Mentor: [Yushi Cheng] TA: [Guozheng He]

Abstract

"Dungeons & Dragons (D&D)" is celebrated for its boundless creativity and immersive gameplay yet faces limitations in visual and interactive expression of today's implementation. To address this, the "Epicast" is introduced, aiming to revolutionize the D&D gaming experience. Epicast integrates advanced technologies like GPT-powered AI and real-time data processing to offer players unprecedented immersion and engagement. The system features an AI Dungeon Master for intelligent narrative creation, an overhead projector for visual display, gesture recognition for intuitive control, and ambient lighting for mood setting. Epicast transforms traditional D&D sessions into dynamic, interactive adventures, enhancing player enjoyment and creativity. The design incorporates various modules such as Support, I/O, Control, Power, Game Instruction Decoder, and Game Logic State Machine, each fulfilling specific functions crucial to the system's operation. Challenges in hardware and software subsystems are addressed, ensuring compatibility, real-time data processing, and logical consistency. Ethical considerations and safety measures are also discussed to ensure a positive gaming experience for all users.

Contents

1. Introduction

Dungeons & Dragons (D&D) is a game that thrives on the breadth of imagination and the depth of interaction. It empowers players to construct elaborate worlds and characters that stem from the vast expanse of their creativity. This unfettered freedom not only offers a canvas for creation and interaction but also makes each gameplay experience profoundly personal and distinct. However, the richness of these imagined scenes and environments is often bottlenecked by the need for verbal expression and lacks an intuitive, sensory display. This limitation hampers the visualization of the game's full potential, presenting a hurdle for some players and constricting the game's broader allure. In addition, the Dungeon Master (DM) which is one of the players serving as the game's narrative architect, is often burdened with extensive preparatory work, juggling game mechanics with storytelling, which can be arduous and time-consuming. To address these issues above, project "Epicast" is a comprehensive system designed to enhance the Dungeons & Dragons gaming experience.

1.1 Functions

Central to our "Epicast" system is a GPT-powered AI that serves as an automated Dungeon Master, guiding gameplay with intelligent narrative creation and player interaction.
The visual aspect is handled by an overhead projector, capable of displaying intricate game scenes and animations. Its height is adjustable for optimal image quality. Gesture recognition is enabled through a sophisticated camera, allowing for intuitive control and the ability to capture memorable moments. Audio immersion is provided by integrated speakers and microphones for voice commands, narrative flow, and dynamic sound effects.

1.2 Benefits

Epicast transforms traditional Dungeons & Dragons (D&D) sessions into immersive, interactive experiences. Through the integration of advanced technologies like GPT-powered AI, sophisticated projection, and real-time data processing, Epicast opens new realms of imagination and engagement for players. Here's a scenario depicting the enhanced experience Epicast offers:

Imagine a group of friends gathered around a table, ready to embark on a D&D adventure. As they settle in, the lights dim and the Epicast system comes to life. The room is filled with the ambient sounds of a bustling medieval town, setting the stage for their quest.

With a wave of a hand, one player casts a spell. The gesture is instantly recognized by Epicast's sophisticated camera system, and the overhead projector vividly displays the spell's effects across the gaming surface – a brilliant burst of light enveloping their foes. Laughter and cheers fill the room, not just for the spell's success, but for the seamless, intuitive interaction that made it feel real.

The GPT-powered AI Dungeon Master, with its vast narrative capabilities, guides the adventure with dynamic storytelling. It generates rich descriptions of their surroundings, intricate plot twists, and detailed NPC interactions based on the players' actions and decisions. The players are no longer just imagining the dark, eerie forest; they see it projected in front of them, hear the sounds of unseen creatures, and feel the tension of the unknown.

As the adventure progresses, the ambient lights shift to match the mood, further drawing the players into the game. A serene blue glow accompanies their arrival at a tranquil lake, while a sudden shift to red signals the imminent danger of a dragon's attack. These cues, synchronized with the game's events, enhance the emotional and sensory engagement of the players.

With Epicast, every action and decision have an immediate, visible impact. The system's real-time processing and projection capabilities allow for a dynamic game world that evolves with the players' journey. The traditional boundaries of the board game are expanded, offering a more enriching and captivating experience.

1.3 Related Background

For a more immersive board game experience, several teams and companies have developed AR (Augmented Reality) board games for mobile devices such as phones or tablets. One notable example is the renowned game team known as "Mirrorscape,"[1] dedicated to crafting augmented reality platforms for tabletop role-playing games (TTRPGs). They have made available an AR D&D board game on both the Google and Apple app stores, accessible for free. Each player needs to position their device on a support with the cameras directed towards the table. Through their devices, they can witness virtual 3D game interfaces overlaid onto the real table. However, this implementation has its limitations. AR technology, as it stands, isn't flawless; there are instances where devices struggle to accurately recognize the table's plane, resulting in an unstable AR 3D interface. Additionally, players are required to view the interface through their device screens, which may detract from the gaming experience. Furthermore, with devices and supports occupying table space, players may find it uncomfortable to rest their arms.

1.4 Features

For a more immersive board game experience, several teams and companies have developed AR (Augmented Reality) board games for mobile devices such as phones or tablets. One notable example is the renowned game team known as "Mirrorscape,"[1] dedicated to crafting augmented reality platforms for tabletop role-playing games (TTRPGs). They have made available an AR D&D board game on both the Google and Apple app stores, accessible for free. Each player needs to position their device on a support with the cameras directed towards the table. Through their devices, they can witness virtual 3D game interfaces overlaid onto the real table. However, this implementation has its limitations. AR technology, as it stands, isn't flawless; there are instances where devices struggle to accurately recognize the table's plane, resulting in an unstable AR 3D interface. Additionally, players are required to view the interface through their device screens, which may detract from the gaming experience. Furthermore, with devices and supports occupying table space, players may find it uncomfortable to rest their arms.

1.5 Visual Aid

Figure 1 the Visual Aid of the Project.

The provided image serves as a visual aid to illustrate the EPICAST system.

- Interactive Display: At the center of the setup, a large, interactive display is shown, which projects the game board and various elements. This display serves as the primary visual interface for the game, allowing players to see the game environment and interact with it dynamically.

- Gesture Control: One of the players is shown using hand gestures to interact with the game. This highlights the gesture recognition capability of the EPICAST system, which allows players to control game elements through physical movements, enhancing the immersive experience.

- Voice Commands: Another player is depicted speaking, with annotations indicating the use of voice commands. This feature enables players to control the game and communicate actions verbally, integrating natural language processing for seamless interaction.

- Integrated Hardware: The image shows various hardware components, including the Raspberry Pi, which acts as the central processing unit, handling input signals from the camera and microphone, and managing output signals to the projector and speakers. The system's ability to handle real-time data processing is crucial for maintaining an immersive and responsive gaming experience.

- Real-Time Updates: The projection on the table includes dynamic elements such as character models, dice rolls, and terrain changes, demonstrating the system's capability to provide real-time updates and visual feedback to the players, keeping the game flow smooth and engaging.

- Collaborative Play: The setup encourages collaborative play, with players gathered around the table, each contributing to the game through their unique inputs. This highlights the social and interactive aspects of the EPICAST system, making the gameplay more engaging and participatory.

Overall, the visual aid effectively showcases the EPICAST system's innovative features, including gesture and voice recognition, real-time data processing, dynamic visual displays, and an immersive gaming environment. These elements work together to revolutionize the traditional D&D gameplay, making it more interactive, engaging, and enjoyable for players.

1.6 High-level Requirements List

Physical Subsystem:

- Motor Control: Ensure precise and subtle motor control with unit angular displacement as 0.1125 degrees. And The response time to the forward, reverse, and stop commands is within 0.5 seconds

- Camera Rotation: Rotate camera continuously for visual coverage with speed of 3.0 degrees per second. And cycle through 270 degrees.

- Portability: light weight of 4 kg for main body, adapt to most board-gaming environment.

Interaction:

- Realtime Gesture Interaction: Detect landmarks of hands. Recognize hand gesture. Stop and activate camera rotation. Low latency and fluency of live stream video detected output frame (exceed 10 FPS) and within 0.5 second for recognition of landmarks and gesture of hands. Also, latency of motor control should within 1 second.

- Realtime Voice Interaction: Utilizes STT (speech-to-text) and TTS (text-to-speech) technologies. Maintains low latency with a processing speed of at least 2.5 words per second for STT and a response time of less than one second for TTS. Accuracy is preserved at a level that does not disrupt the game flow.

Gaming and Style:

- Game Completion: Deliver a fully functional and playable game.

- Local Deployment: The system can be deployed entirely locally without dependency on any external web APIs.

- D&D Style: Support for Dungeons & Dragons characteristics.

2 Design

2.1 Block Diagram

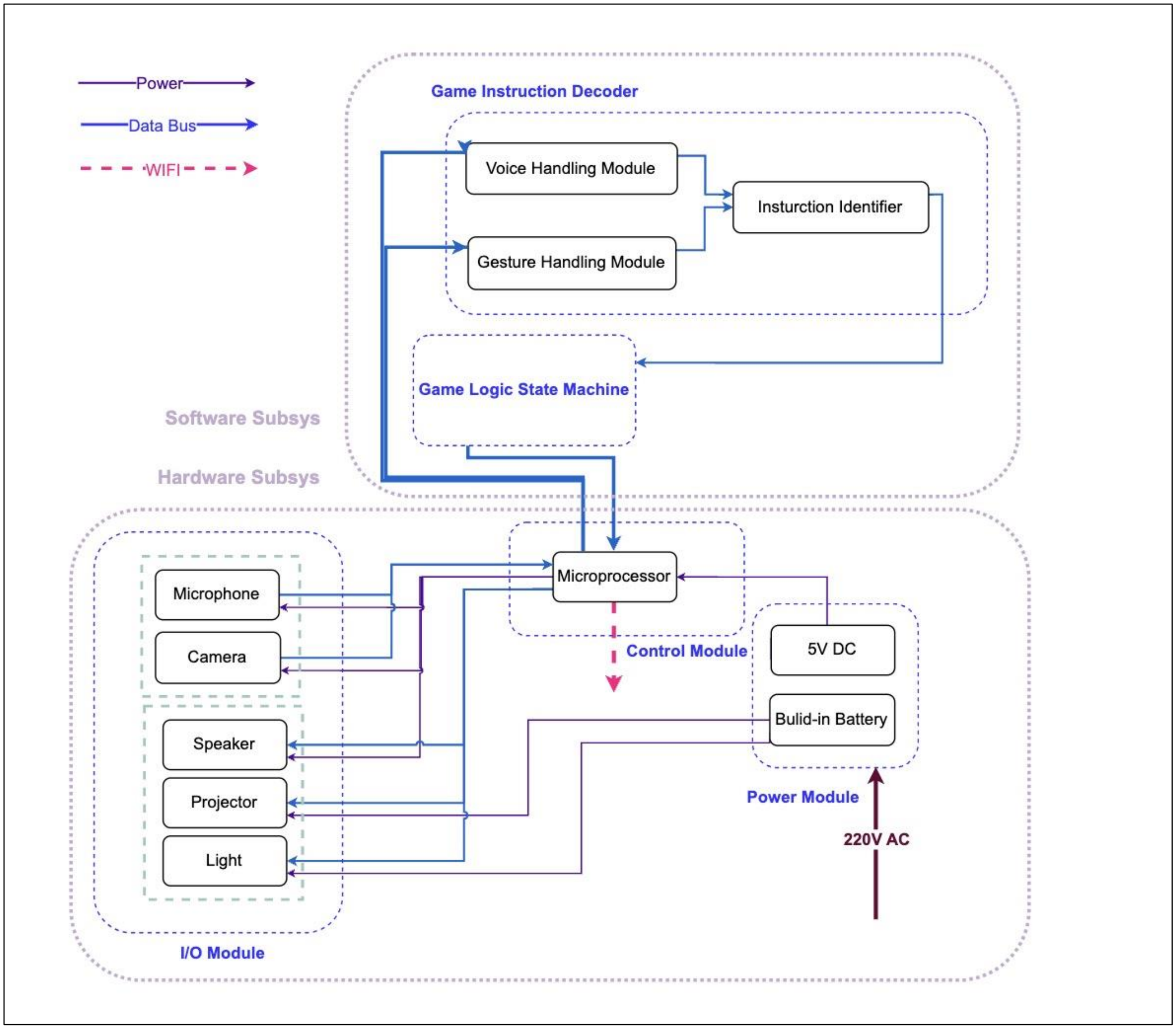

Figure 2 the Overall Block Diagram.

2.2 Physical Diagram

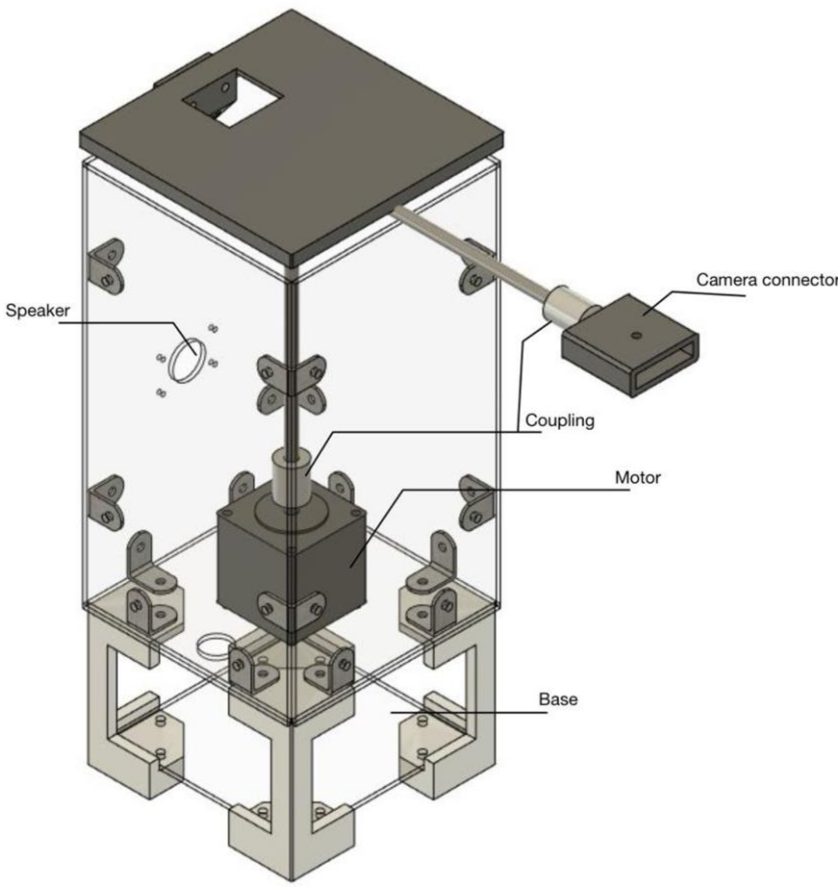

Figure 3 the Model of Support Module.

2.3 Block Description

2.3.1 Support Module

The support module is a holder of physical parts like camera, microphone, raspberry Pi. During the game time, this device will be placed at the middle of the table, while four players sit at any position beside the table. When a player wants to manipulate his role in game, he just needs to reach out one of his hands into the sight of the camera. Once the camera detects player's hand, the rotating camera will stop, and the player can use gesture or voice to manipulate the role in game.

There will be a motor inside the device which can rotate the camera so that our camera can capture user's hands accurately. A camera with microphone will be attached to the support module by the camera connector shown in the figure, The camera is lifted and slightly facing downward so that players can put their hands on the table instead of raising them towards the camera which can add more ease to their manipulation. A light will be placed near the lens so that it can point out the zone where the camera is capturing. A joint is used to connect the camera with the motor. With the joint we can slightly change the angle of the camera so that our device can fit the tables with different sizes.

Requirements:

1. Support Module can hold all physical parts including I/O Module, Motor Drive Module and Power Module tightly without bulk or break, have a safety factor larger than 5.
2. The motor can provide enough torque during the operation.

Verification of Safety Factor:

Simulate parts to find out the quantity and location of the stress they suffered and the strain they occurred. Calculate the safety factor of the mechanism.

After analysis, we believe the part which is likely to fail is the camera connector, since the protrusion of the stick will cause stress concentrations. Also, when adjusting the angle of the camera, an external bending will be applied on it. Given it's 3D printed which are more likely to fail, it's necessary to make some simulations.

In order to simply the simulation, I assume the connector will suffer a force of 30N downward which distributed normally and greater than the actual case.

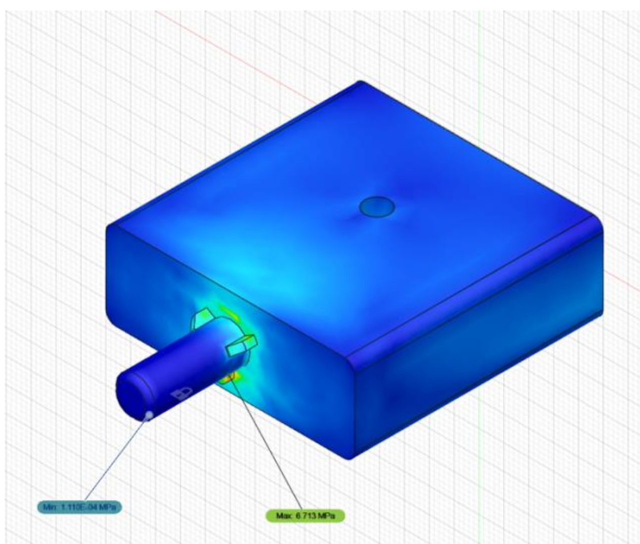

Figure 4 Von Mises Stress Suffered by Camera Connector.

As we can see from this figure, the stress is mainly contributed at the joint of the bar and the holder. In order to prevent this from failing, some reinforcement structures are added.

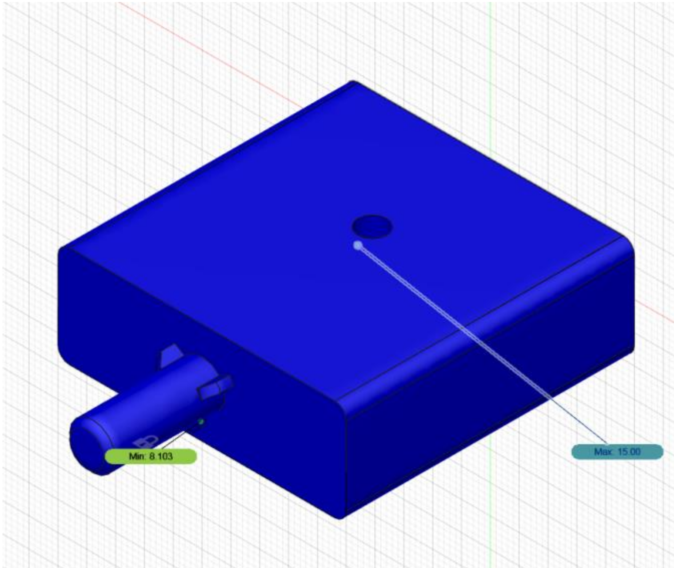

Figure 5 The Safety Factor of Camera Connector.

This holder has a minimum safety factor of 8.103, which satisfies the requirement of engineering. Also, this is obvious that this part will have a lowest safety factor among all components during operation. Therefore, I verified that the support module is stable during the operation.

Verification of Motor Torque:

The model we select can provide a torque of 1.2Nm.

Assume, the camera can be treated as a point mass.

Therefore：

The moment of inertia of camera: $I_1 = ml^2 = 0.3kg * 0.04m^2 = 0.012kg * m^2$ *(0.3kg is heavier than the actual case)*

The moment of inertia of horizontal bar: $I_2 = \frac{1}{3}ml^2 = \frac{1}{3} * 0.316kg * 0.04m^2 = 0.004kg * m^2$

$$The\ total\ moment \text{ moment of inertia r: } I = I_1 + I_2 = 0.016kg * m^2$$

Assume, the acceleration of motor is a uniform acceleration motion (I assume the time needed to fully complete acceleration is 0.1s) .

Therefore:

$$\tau = I * \alpha = I * \frac{\omega}{t} = 0.016 * 35 * \frac{2\pi}{60 * t} = 0.59N * m$$

It turns out the motor we choose can fully satisfy the requirement theoretically. Also, under our real test, we verified that when given a stable voltage of 24V, the motor can take the camera as well as the microphone to rotate continuously.

2.3.2 I/O Module

The I/O module is a signal management and dispatch center. It connects with support module and software subsystem. The center of the Electrical subsystem is a Raspberry Pi broad. It receives real-time video messages from camera and microphone by Universal Asynchronous Receiver/Transmitter. After receiving the vision and audio signal, it will send them through Controller Area Network to software subsystem to accurately process. The application of virtual network cards will assist in online processing and operation of information and output. When specific images are output by software, it is also responsible for projector output signal transmission. Speaker is also applied to broadcast voice signal. For EPICAST, we have chosen the Raspberry Pi 4B/8G as our microprocessor. Serving as the software foundation, Raspberry Pi will operate on Linux, supporting the game instruction decoder and game logic state machine. The control module will also feature a data bus connecting all I/O components to facilitate interaction with players.

Choose to use a more integrated camera module instead of camera components. This makes the module output more stable and provides better mechanical strength with the help of packaging materials, reducing the need for strength considerations in later physical construction. The camera and microphone input are integrated in the component, which simplifies the component composition and makes the design smoother and more efficient.

Figure 6 Integrated Camera and Microphone.

To set up Raspberry Pi, begin by gathering the necessary equipment such as the Pi board, microSD card with an installed OS, peripherals like a keyboard and mouse, and a power supply. Install the OS onto the microSD card using software like Etcher, then insert the card into the Pi and connect peripherals. Boot up the Pi and follow on-screen instructions to set up the OS, including language, time zone, and network settings. Explore the Pi's capabilities for various tasks like programming or media center use. Learn the operating environment of the Raspberry PI. Through the configuration of the mirror system to achieve the basis of internal program editing, the Raspberry PI system learning and configuration process, the successful realization of the camera call, for gesture recognition and other visual input system to provide the foundation.

Figure 7 Integrated Camera and Microphone.

We chose a projector that fits needs, set it up, and connected it to my laptop and smartphone to test its functionality, including Bluetooth capabilities. After adjusting basic settings like brightness and focus, we assessed its performance, ensuring the projected image was clear and visible from different angles. Once satisfied, we finalized the setup, ready to use the projector for various purposes. The speaker also applies I2C signal from raspberry pi to recognize test from game machine and output host voice in short time.

Figure 8 Projector.

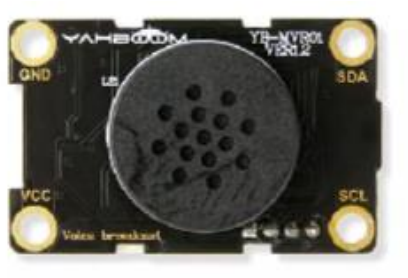

Figure 9 Speaker.

2.3.3 Motor Drive Module

EPICAST image input initializes all player gesture information and uses the camera to capture round table player gestures in real time during gameplay for in-game interaction. The camera module will be equipped with a circular motion camera track for continuous monitoring. After a specific range of player gestures is obtained, the movement stops, and the player interaction is focused until the player completes the current turn. To this end, a stable motor is needed to drive the camera mobile module to serve the player image input and recognition.

The decision to use a stepper motor for our project is driven by its precision control capabilities, cost-effectiveness, simplicity in signal control, and additional benefits such as high torque at low speeds and durability. These attributes make stepper motors the ideal choice for achieving the desired performance and reliability in our project.

Figure 10 Stepping Motor Data.

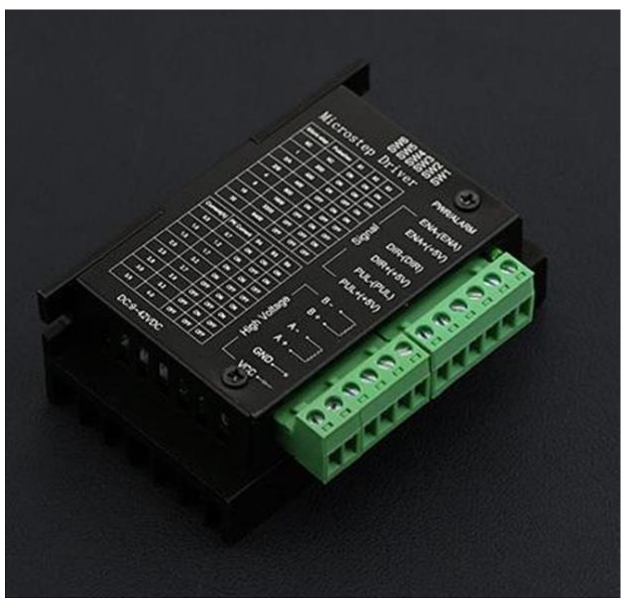

Figure 11 TB6600 Motor Driver Chip.

We write code in raspberry Pi to control a stepper motor using a TB6600 stepper motor driver. A key aspect of this control involves generating precise Pulse Width Modulation (PWM) signals, which are essential for managing the motor's movements. The Raspberry Pi utilizes its GPIO pins to send these PWM signals to the TB6600 driver, which interprets them to control the stepper motor's operation.

In the setup phase, the GPIO pins on the Raspberry Pi are configured for output. These pins are connected to the TB6600 driver's input pins, specifically the pulse and direction controls. The pulse control pins (PUL+ and PUL-)

receive the PWM signals, while the direction control pins (DIR+ and DIR-) determine the rotation direction of the motor. This setup ensures that the Raspberry Pi can effectively communicate with the driver to manage the motor's movements.

The core functionality of the code revolves around generating the PWM signals required to drive the motor. This is achieved by rapidly toggling the state of the GPIO pins, creating a series of pulses. The timing and frequency of these pulses are crucial, as they directly influence the motor's speed and position. By adjusting the delay between pulses, the code controls the speed at which the motor steps. A shorter delay results in faster motor rotation, while a longer delay slows it down.

Table 1 Circuit Connection.

Driver Port	TB6600 Driver Port	Raspberry PI BCM code	Description
BLACK	A+	/	Motor A phase positive
GREEN	A-	/	Motor A phase negative
RED	B+	/	Motor B phase positive
BLUE	B-	/	Motor B phase negative
/	PUL+	16	Step pulse signal positive
/	PUL-	18	Step pulse signal negative
/	DIR+	13	Direction control signal positive
/	DIR-	15	Direction control signal negative

Power subsystem includes two distinct power adapters. The first power adapter is responsible for converting the 220V AC input from the mains supply to a 24V DC output, which is used to power the TB6600 driver chip. The second power adapter is an integral component of the Raspberry Pi setup. It converts the 220V AC input to a 5V DC output, which is essential for powering the Raspberry Pi board.

2.3.4 Game Instruction Decoder

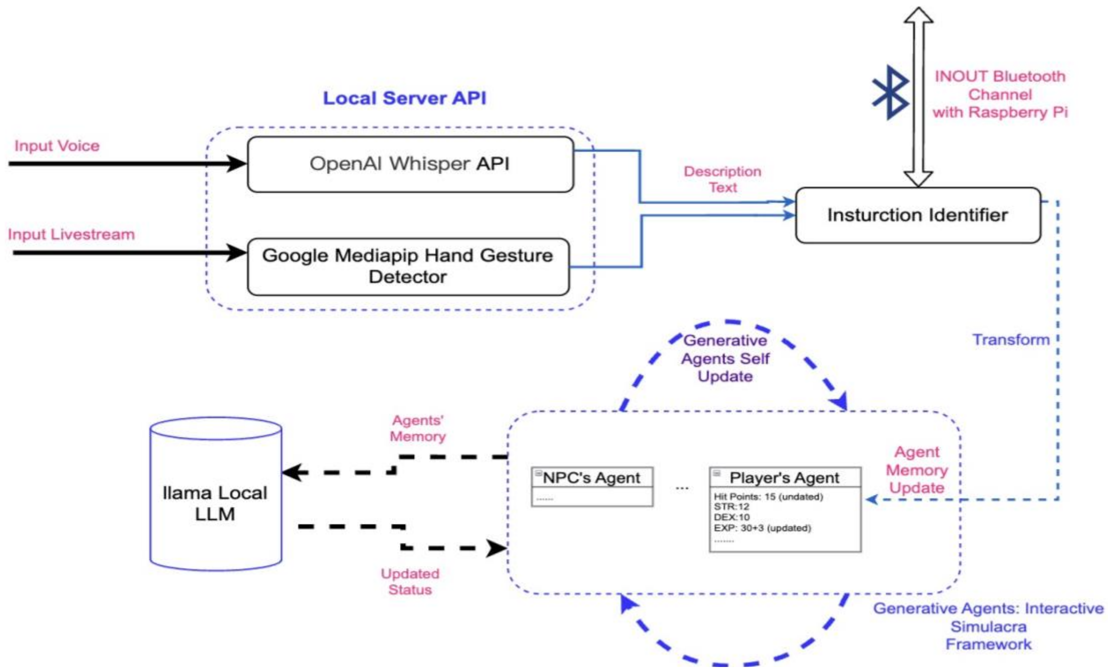

Figure 12 Integrated Diagram of Decoder with GLSM.

The Game Instruction Decoder translates high-level voice and gesture commands from players into a set of clear instructions that the Game Logic State Machine (GLSM) can understand. It utilizes separate Deep Learning networks (Based on OpenAI Whisper and Google MediaPipe Gesture Recognizer [8]) to process sound and image data, respectively, and convert them into text. The decoder then interacts [9] with the game cycle to feed the corresponding texts and commands into the right place. For example, player's voice input should feed into his/her ongoing dialogue with the NPCs and gesture command should control character movement and various other behaviors.

Requirements:

1. Successfully recognize players' gestures from various angles
2. Interpret player voice commands correctly despite variations like Inverted sentence order, Pauses and Rich player vocabulary.
3. Translate speed of players' voices should be within 3 words per second. The detection of one gesture should be within 1 second.
4. Successfully transmit signal to Raspberry Pi through GATT [6] [10] [11] Bluetooth protocol.

2.3.5 Game Logic State Machine (GLSM)

The GLSM acts as the system's core, managing the internal state of each player (items, health, stats) and the environment (map, NPCs, triggers). It ensures the game's logic flow remains consistent throughout the experience.

Requirements:
1. Guide and maintain the correct game progression despite variations in AI-generated content (e.g., narratives, scenarios).
2. Maintain low latency when contacting the server for the Large Language Model (LLM), ensuring smooth gameplay.
3. Modular design that allows for easy modification and adjustment to expand or change game content.

Upon thorough evaluation, we have adopted the "Generative Agents: Interactive Simulacra of Human Behavior" AI framework as the foundational structure for our Game Logic State Machine (GLSM). This model is selected for its capacity to maintain a long-term memory state for each character and to support intricate interactions among characters, thereby enhancing the dynamism and continuity of the game environment. This integration is visualized in the conceptual graph from the original paper (Park et al., 2023, Figure 1)."

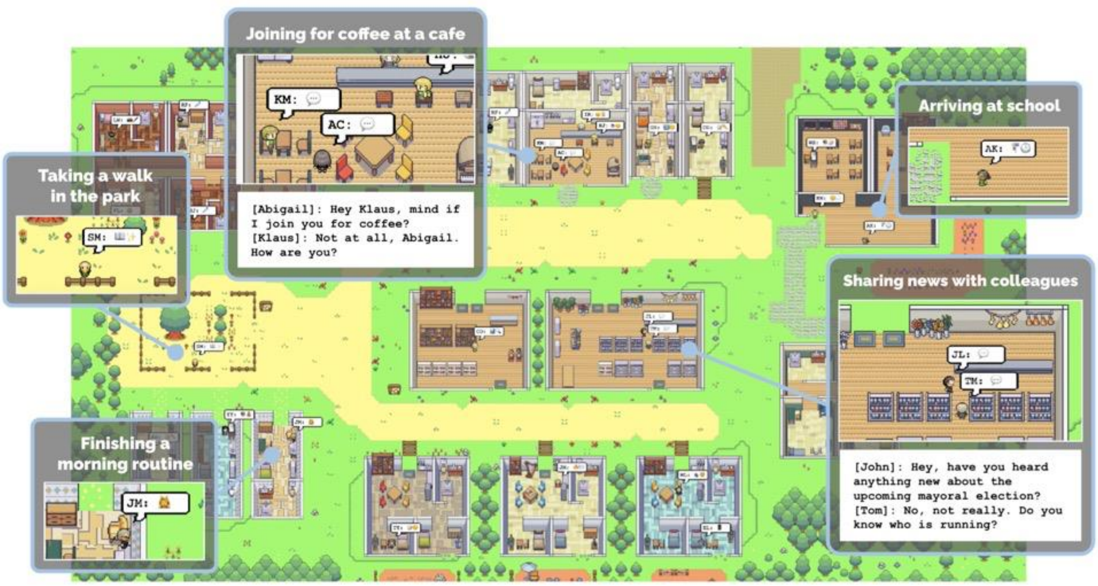

Figure 1: Generative agents are believable simulacra of human behavior for interactive applications. In this work, we demonstrate generative agents by populating a sandbox environment, reminiscent of The Sims, with twenty-five agents. Users can observe and intervene as agents plan their days, share news, form relationships, and coordinate group activities.

Figure 13 Visualization of the Conceptual Graph.

2.4 Risks & Challenges

Support Subsystem:

- Lightweight and Movable: The physical subsystem should be light and easy to handle with, which can make it fit more scenarios and can meet the needs of more customers, like can be played even in outdoors. This requires ingenious mechanism.

- Precise Motor Control: Since our camera aims to put users' hand in the middle of its sight, but the camera is rotating, it can be hard for the motor to drive the camera to an appropriate angle. Also, there is a connector at one edge of the support, so we need to put a limit and find a path for the motor to follow to avoid it to hit the connector.

Hardware Subsystem:
- Diverse Sensor Types: Different sensors may have different data formats, communication protocols and interface requirements. Hardware systems need to be able to adapt and handle a variety of different types of sensors, including microphone signal input and camera signal input. This requires hardware systems to be flexible and compatible, capable of effective communication and data exchange with a variety of sensors.
- Data Conflict and Redundancy: When multiple information sources provide data at the same time, data conflict or redundancy may occur. Hardware systems need to be able to handle these conflicts and redundancies, selecting the right data for processing and analysis to provide accurate results. In addition, there are high requirements for real-time data. The hardware system needs to be able to process and respond to the input instructions of the sensor in time under the real-time requirement to ensure the stability and reliability of the system.
- Stable Motor Drive: Because the moving camera module is an important part of the game interaction, the running speed and stability of the module will greatly affect the player interaction experience. The battery drive needs to be well matched with the identification output, improve the fluidity of movement, and complete the preset operation logic requirements to achieve practical goals.

Software Subsystem:
- Maintaining Logical Consistency: A core challenge lies in maintaining the internal logical consistency of the game world despite its reliance on AI-generated content. The system must seamlessly integrate the AI-powered narrator's descriptions, visual elements generated by the projector, and game logic updates managed by the GLSM. This requires implementing deterministic code alongside the AI components. Deterministic code refers to pre-defined rules and algorithms that guarantee consistent and predictable outcomes, even when handling the inherent variability of AI-generated content.
- Handling High Player Freedom: Dungeons & Dragons is renowned for its emphasis on player freedom, allowing players to make diverse choices and engage in creative actions. However, this presents a challenge for the system, as it needs to translate this vast range of player commands into a finite set of understandable game instructions. To address this, the Game Instruction Decoder and Game Logic State Machine together plays a crucial role.
- Game structure: Due to both 1 and 2, it's challenging to develop a satisfiable game structure with comprehensive character behavior definition and victory conditions. As with the logic flow being controlled by the AI, it's indeterministic for when and how the game will terminate and how to assign players' rewards properly during the game process. For now, we can only develop an open-ended game that's mostly chat-based and keep very little player-object interaction. But we firmly believed these features can be updated for future works.

3 Verification

As mentioned above in this report the requirements for project Epicast could be classified into three categories: Hardware, Interaction, Gaming and Style. In this section, we will go into more details for these requirements and explain the corresponding method we used for verification.

3.1 Hardware

3.1.1 PWM Control: Impulse Number and Rotation Range

This test aims to examine how varying the number of impulses affects the motor's rotation range. For the purpose of this test, we set the parameter of chip by 1600 impulses are required for one full rotation (360 degrees).

Test Steps:

1. Command the motor with different numbers of impulses (e.g., 400, 800, 1200, 1600).
2. Measure the corresponding rotation range for each impulse number using a high-resolution rotary encoder.
3.Repeat the test for each impulse number to ensure consistency.

Test Results：

Table 2 Impulse Number and Rotation Range Test Results.

Test Cycle	Number of Impulses	Expected Rotation (degrees)	Actual Rotation (degrees)
1	400	90	90
2	800	180	180
3	1200	270	270
4	1600	360	360

The test results demonstrate a direct correlation between the number of impulses and the motor's rotation range. As expected, increasing the number of impulses proportionally increases the rotation range. This linear relationship confirms that the motor accurately responds to the PWM control signals, ensuring precise control over the rotation range.

Strengths:

- The motor exhibits consistent rotation for a given number of impulses, ensuring predictable performance.
- The system's ability to handle varying numbers of impulses allows for fine-tuned control over the motor's position.

Weaknesses:

- The test assumes ideal conditions without significant external disturbances, which may not always be the case in practical applications.

Limitations:

- The test did not account for the effects of mechanical wear and tear over prolonged use, which could impact the motor's performance over time.

3.1.2 PWM Control: Impulse Time Gap and Rotation Speed

This test aims to examine how varying the time gap between impulses affects the motor's rotation speed. The impulse time gaps tested are 0.001s and 0.0001s.

Test Steps:

1. Command the motor to rotate with impulse time gaps of 0.001s and 0.0001s.
2. Measure the rotation speed using a high-resolution rotary encoder.
3. Repeat the test for each time gap to ensure consistency.

Test Results:

Table 3 Impulse Time Gap and Rotation Speed Test Results.

Test Cycle	Impulse Time Gap (seconds)	Rotation Speed (degrees/second)
1	0.001	3.6
2	0.0001	36.0

The test results show a significant impact of the impulse time gap on the motor's rotation speed. A shorter time gap between impulses results in a higher rotation speed, as demonstrated by the increase from 3.6 degrees per second at 0.001s to 36.0 degrees per second at 0.0001s. This indicates that the motor can achieve high-speed rotations with appropriately timed PWM signals.

Strengths:

- The motor responds well to changes in impulse time gaps, allowing for precise control over rotation speed.
- The system's ability to handle high-speed rotations is beneficial for applications requiring rapid movements.

Weaknesses:

- High-speed rotations may introduce additional challenges such as increased wear and tear on mechanical components.

Limitations:

- The test did not explore the upper limits of the motor's speed capabilities, which could provide a more comprehensive understanding of its performance range.

3.1.3 Motor Precision

The motor control test aims to ensure precise and subtle motor control with a unit angular displacement of 0.1125 degrees. Additionally, the response time to forward, reverse, and stop commands should be within 0.5 seconds.

Test Steps:

1. Command the motor to move forward by 0.1125 degrees.
2. Measure the actual angular displacement using a high-resolution rotary encoder.
3. Command the motor to move in reverse by 0.1125 degrees.
4. Measure the actual angular displacement again.
5. Command the motor to stop and measure the response time.
6. Repeat steps 1-5 for 10 cycles to ensure consistency.

Test Results:

Table 4 Motor Precision Test Results.

Test Cycle	Forward Displacement (degrees)	Reverse Displacement (degrees)	Response Time (seconds)
1	0.1125	0.1125	0.4
2	0.1125	0.1125	0.4
3	0.1125	0.1125	0.4
4	0.1125	0.1125	0.5
5	0.1125	0.1125	0.4
6	0.1125	0.1125	0.4
7	0.1125	0.1125	0.5
8	0.1125	0.1125	0.4
9	0.1125	0.1125	0.4
10	0.1125	0.1125	0.4

The motor precision control test demonstrated that the motor could achieve a precise angular displacement of 0.1125 degrees consistently across all test cycles. The response time to forward, reverse, and stop commands was consistently within the target of 0.5 seconds, with most cycles achieving a response time of 0.4 seconds. This level of precision and responsiveness is critical for applications requiring accurate and quick movements, such as robotic arms and precision instrumentation.

Strengths:
- High precision in angular displacement, ensuring accurate positioning.
- Quick response time to control commands, enhancing real-time control capabilities.

Weaknesses:
- The test was conducted under ideal conditions without significant external disturbances. In practical scenarios, factors such as load variations and environmental conditions could affect performance.

Limitations:
- The motor's performance under varying load conditions was not extensively tested, which could provide a more comprehensive understanding of its capabilities.

3.1.4 Camera Rotation

The camera rotation test ensures continuous visual coverage with a rotation speed of 3.0 degrees per second and a cycle through 270 degrees.

Test Steps:
1. Command the motor to rotate the camera at a speed of 3.0 degrees per second.
2. Measure the rotation speed using a high-resolution rotary encoder.
3. Command the motor to rotate the camera continuously for a cycle of 270 degrees.
4. Record the time taken to complete the rotation.
5. Verify the camera's visual coverage during the rotation.

Test Results:

Table 5 Camera Rotation Test Results Test Results.

Test Cycle	Rotation Speed (degrees/second)	Total Rotation (degrees)	Time Taken (seconds)
1	3.0	270	90
2	3.0	270	90
3	3.0	270	90
4	3.0	270	90
5	3.0	270	90

The camera rotation test showed that the motor could maintain a rotation speed of 3.0 degrees per second and complete a 270-degree cycle consistently in 90 seconds. This consistent performance is essential for applications requiring continuous visual coverage, such as surveillance and automated inspection systems.

Strengths:

- Consistent rotation speed, ensuring smooth and uniform camera movement.
- Adequate time management for completing the required rotation cycle.

Weaknesses:

- The camera's visual coverage was verified visually, and more precise measurement tools could provide better validation of the rotation accuracy.

Limitations:

- The test did not account for potential mechanical wear and tear over prolonged use, hich could impact the motor's performance over time.

3.2 Portability

For the Support Module, we need it can be transported and handled easily by a single person, which can bring it more application scenarios. Therefore, we want to be small and light in weight.

Its final dimension is a rectangular prism of 15.4cm * 15.4cm * 40.1cm. The smaller size and rectangular shape make it easier to transport and carry. Also, after measuring, the total weight of the whole mechanism is 3.106kg, which is less than 4kg as required by the high-level requirement list.

3.3 Interaction

3.2.1 Voice Interaction

In this module, we need to implement a real-time STT server for player voice transcription and a real-time TTS server for NPC voice synthesis (STT and TTS stands for Speech-to-Text and Text-to-Speech respectively). Accuracy and latency of course should be our primary concern.

After conducting a real-world simulated Dungeons & Dragons game experience, we determined that the latency requirement for speech-to-text (STT) should be at least 2.5 words per second, which aligns with the average speaking speed of an English-speaking adult in the US. Additionally, the text-to-speech (TTS) server must initiate synthesis within less than one second after receiving the NPC message.

We initially tested these modules individually, and they easily met the required benchmarks. During the subsequent integration testing of all modules, we encountered issues with lag and stalling in the STT and TTS servers. This was primarily because a significant portion of our computational resources was being utilized by the large language model (LLM) and the game engine, as the entire project was deployed locally. However, after switching to a smaller language model (tiny.en), the latency once again met the expected standards. While this change slightly impacted accuracy, we were pleased to find that these minor inaccuracies were ignored by our LLM-supported AI agent when incorporated into the game. Thus, from an operational perspective, the accuracy was still within acceptable limits. We measured the latency of speech recognition and synthesis by engaging in direct conversations with NPCs and reciting long paragraphs in the game. Our tests showed that we achieved a TTS recognition speed of 2.87 words per second and an STT server response time well below one second.

3.2.2 Gesture Interaction

In this model, our primary objective is to achieve real-time gesture interaction and hand landmark recognition. First, when a player's hand is detected by the camera, the running motor stops, meaning the start of the player's turn. Using Google's landmark annotation, we extract the location of landmark node 8 [7] of the left hand, representing the tip of the index finger, to control a mouse-like pointer on the game interface. We map specific right-hand gestures to game operations: a "closed palm" signifies a click, a "thumb up" indicates scrolling up, and holding a "victory" gesture ends the player's turn, reactivating the motor.

In a real game scenario, the latency of these operations must be minimal. The frames per second (FPS) should be high enough to ensure smooth gameplay. A basic requirement for video games is at least 10 FPS, so our system needs to support this minimum to maintain fluent game operation. Given that our Epicast game is turn-based rather than action-oriented, a latency of 0.5 seconds for hand gesture and location detection is tolerable. Additionally, motor operation latency is crucial since the camera's stop position directly affects the player's ease of use.

We modified the Google MediaPipe model to record system time and added code to log the output frame rate. We achieved an average of 12.9 FPS for detected output frames. To measure hand gesture detection latency, we introduced a hand at the first frame of the live video stream, recorded the landmark start time, and then noted the output frame where the correct gesture was detected. The average latency was approximately 0.47 seconds under optimal lighting conditions. Motor control signals are sent via Bluetooth GATT protocol, meaning the Bluetooth transmission latency between the laptop and Raspberry Pi must be within 0.53 seconds (1 second minus 0.47 seconds). Testing with the "nRF Connect" app showed a Bluetooth latency of 122 ms (0.12 seconds), which is well within acceptable bounds. Thus, our gesture interaction latencies are all within optimal limits.

3.4 Gaming and Style

This part of requirements required us to deliver a fully functional, playable D&D style game. To ensure this, we sought comprehensive testing that would provide user feedback on whether our interaction method was intuitive and on their experience with the D&D world view and game logic flow created by AI. Therefore, we invited a fellow student, who was not a member of the project team, to play and experience the game.

After explaining the basic gestures and their corresponding actions, the student quickly got the hang of it and mastered the game's interaction mode within five minutes. According to the student's feedback, our interaction method was intuitive and easy to learn, significantly enhancing the gaming experience. Additionally, the student successfully created a story about the Eastern Magic Forest in the D&D world, with conversations between the character and various NPCs maintaining logical coherence within the game. This confirmed that our AI agent effectively managed the game's progress.

4 Cost

Figure 14 Final Product.

Material cost:

The material cost contains all subsystems listed below.

Table 5 Table of Cost.

Part	Cost(¥)
Integrated Camera and Microphone	259
Raspberry Pi 4B	889
Projector	299
Speaker	42
Corner Connectors	7.84
Couplings	35
Screws	10.71
Steel Bars	9.25

5 Conclusion

5.1 Accomplishments

Our project, Epicast, successfully integrated gesture and voice interaction, complete game logic, local deployment, and maintained the Dungeons & Dragons style. The gesture interaction module allowed players to control the game using hand movements, while the voice interaction module enabled real-time speech-to-text and text-to-speech processing for dynamic NPC interactions. The complete game logic and AI Dungeon Master provided a coherent and immersive gaming experience. The system was deployed locally, ensuring independence from external web APIs, which is crucial for uninterrupted gameplay. Additionally, the project retained the essence of D&D, enhancing it with modern technology to create a unique and engaging experience.

5.2 Uncertainties

Despite our successes, several uncertainties remain. Firstly, the gesture recognition module and real-time STT server exhibited insufficient robustness when multithreading, occasionally leading to misinterpretations. Secondly, the game lacked sufficient elements and statistics to fully replicate the D&D experience. Thirdly, the limited computational resources on local deployment caused unexpected stalling and an unsmooth user experience. Lastly, we were unable to implement ambient lighting, which was intended to enhance the immersive atmosphere.

5.3 Future Work

To address the uncertainties and further enhance Epicast, we propose several areas for future work:

Logic of Motor Operation:

Logic needs to be optimized to have a smooth performance in the start and stop stages to avoid the user feeling blocked. It needs to communicate more with software recognition.

Raspberry Pi:

For the Raspberry PI multi-input and output processing environment, it is necessary to optimize the running speed, and improve the production efficiency of the project.

Refinement of AI Model:

Improve the AI Dungeon Master by applying extensive prompt engineering and regular language shaping to enhance the content generated by the locally deployed LLM.

Game Feature Enhancement:

Add comprehensive statistics, interactions with objects, game levels, tiers, termination conditions, and victory criteria to enrich the gameplay.

Interaction Module Improvement:

- **Gesture Interaction:** Develop more robust gesture controls that require less precision and map them to more intuitive actions.

- Voice Interaction: Integrate wake-word techniques to allow more fine-grained voice commands.

- Additional Interaction Methods: Explore the inclusion of other interaction methods such as eye-tracking and body movement detection.

Hardware Support Enhancement:

- Ambient Lighting: Implement ambient lighting to enhance the immersive experience.

- Hardware Interaction: Include hardware components as part of the interaction, such as voice commands to control camera movement.

5.4 Ethics and Safety

Mechanic Part: We conduct a comprehensive risk assessment when designing mechanical equipment and prioritize the use of intrinsic safety measures to fundamentally eliminate the existence of hazards. We will add enough protective measurements to make sure the safety of our user like we have adopted a lightweight design to reduce the safety risk. These measures include enabling the machine to automatically prevent misoperation, as well as automatically detaching from the faulty area and safely transferring to the backup section or stopping operation in the event of a malfunction. We'll also fully consider the issue of human-machine matching to make the machine suitable for various human operations. We take the convenience of product installation into fully consideration.

Privacy Part: One paramount ethical consideration revolves around the privacy of user data within EPICAST. Utilizing cameras to detect player gestures and behaviors, and microphones to capture player voices, may raise privacy issues. The IEEE Code of Ethics underscores the significance of safeguarding others' privacy [2]. In our EPICAST implementation, while players' operation data is essential for game progression, we are committed to preserving privacy. Upon the completion of each game session, EPICAST will promptly delete the current game status and eradicate all residual players' data obtained from real-time voice and gesture inputs. Additionally, EPICAST transmits data to Google and Mozilla AI service platforms over the Internet, subjecting it to their respective privacy terms. Google and Mozilla have meticulously defined privacy protocols [3][4], thereby becoming our privacy policy dependence accordingly. Considering potential data theft during Internet transmission, we pledge to establish secure and encrypted channels for EPICAST's Internet access, ensuring the utmost protection of players' data privacy.

Artistic Copyright: With the implementation of AI in Epicast, we are entering a realm where creative expression and copyright laws intersect in new and complex ways. The generation of images and narrative content through AI systems raises important ethical considerations regarding artistic copyright. The use of such technologies must respect the original creators' rights and avoid infringing upon existing copyrighted works. Therefore, we have proposed the Ethics Code for AI-Generated Artistic Content:

- Respect for Originality: Ensure that all AI-generated content is sufficiently original and not a direct copy of existing copyrighted works. When inspiration is drawn from copyrighted materials, the AI should transform the inspiration in a significant way that results in a new, original piece.

- Transparency: Clearly disclose to all players that the visual and narrative content is generated by AI, and where feasible, provide information on the data sources used to train the AI models.

- Fair Use Compliance: Operate within the bounds of 'fair use' policy, where AI-generated content is used for commentary, criticism, reporting, or teaching in a way that does not diminish the market value of the original works.

- Continuous Monitoring: Regularly review and update the AI's programming to ensure ongoing compliance with copyright laws as they evolve, and promptly address any unintentional infringements.

- Creator Collaboration: Whenever possible, collaborate with artists and creators to obtain licenses for the use of their work as inspiration for AI-generated content, thus supporting the artistic community.

- Responsiveness to Concerns: Establish a protocol for responding to copyright holders' concerns about AI-generated content, including a clear process for reviewing and addressing potential infringements.

- Promotion of Ethical Use: Advocate for the ethical use of AI in artistic creations across the industry, setting a standard for responsible innovation that honors and upholds the rights of creators.

By adhering to these principles, Epicast will not only foster a fair and legally compliant gaming environment but also contribute to the responsible development of AI technologies in respect to artistic expression. The project will continuously strive to balance the innovative use of AI with the utmost respect for intellectual property rights, ensuring a positive and sustainable integration of technology and creativity.

References

[1] Mirrorscape Team. "Introduction to Mirrorscape Team." Mirrorscape, www.mirrorscape.com. Accessed: March 5, 2024.

[2] IEEE, "IEEE Code of Ethics," June 2020. [Online]. https://www.ieee.org/about/corporate/governance/p7-8.html. Accessed: March 7, 2024.

[3] Google. "Google Privacy Terms," [Online]. Available: https://policies.google.com/privacy?hl=en-US. Accessed: March 7, 2024.

[4] Mozilla. "Mozilla Privacy Terms," [Online]. Available: https://www.mozilla.org/en-US/privacy/. Accessed: March 7, 2024.

[5] Park, J. S., O'Brien, J. C., Cai, C. J., Morris, M. R., Liang, P., & Bernstein, M. S. (2023). Generative Agents: Interactive Simulacra of Human Behavior. arXiv. https://doi.org/10.48550/arXiv.2304.03442

[6] Kevin T, "Introduction to Bluetooth Low Energy", Mar 2014. [Online]. Available: https://learn.adafruit.com/introduction-to-bluetooth-low-energy/gatt Accessed: April 28, 2024.

[7] Google. "Mediapipe Guide," [Online]. Available: https://developers.google.com/mediapipe/solutions/vision/gesture_recognizer/index. Accessed: May 1, 2024

[8] Google. "Mediapipe API," [Online]. Available: https://developers.google.com/mediapipe/api/solutions/java/com/google/mediapipe/tasks/vision/gesturerecognizer/GestureRecognizerResult. Accessed: May 1, 2024

[9] Python. "PyAutoGUI Documentation," [Online]. Available: https://pyautogui.readthedocs.io/en/latest/. Accessed: May 1, 2024

[10] dlech. "bleak," [Online]. Available: https://github.com/hbldh/bleak/. Accessed: May 1, 2024

[11] bluepy. "a Bluetooth LE interface for Python," [Online]. Available: https://ianharvey.github.io/bluepy-doc/index.html. Accessed: May 1, 2024

[12] AI Town. Available: https://github.com/a16z-infra/ai-town

[13] Avrae. "Avrae Documentation." [Online]. Available: https://avrae.io/. Accessed: May 24, 2024.

[14] AI Dungeon. "Play AI Dungeon." [Online]. Available: https://play.aidungeon.com/. Accessed: May 24, 2024.

[15] Thomason, J., Greenwood, M., Knight, A., & Hessel, J. (2023). "Generative Agents: Interactive Simulacra of Human Behavior." AIIDE. [Online]. Available: https://ojs.aaai.org/index.php/AIIDE/article/view/27534. Accessed: May 24, 2024.

[16] DigitalOcean. "How to Make a Web Application Using Flask in Python 3." [Online]. Available: https://www.digitalocean.com/community/tutorials/how-to-make-a-web-application-using-flask-in-python-3. Accessed: May 24, 2024.

[17] OpenAI. "Whisper Discussion #608." GitHub. [Online]. Available: https://github.com/openai/whisper/discussions/608. Accessed: May 24, 2024.

[18] SYSTRAN. "Faster Whisper." GitHub. [Online]. Available: https://github.com/SYSTRAN/faster-whisper. Accessed: May 24, 2024.

[19] KoljaB. "RealtimeSTT." GitHub. [Online]. Available: https://github.com/KoljaB/RealtimeSTT. Accessed: May 24, 2024.

[20] KoljaB. "RealtimeTTS." GitHub. [Online]. Available: https://github.com/KoljaB/RealtimeTTS. Accessed: May 24, 2024.

[21] Sundar, A. V., He, J., & Wu, Y. (2023). "Exploring Efficient Language Models: State-of-the-Art Architectures and Performance." arXiv. [Online]. Available: https://arxiv.org/abs/2403.18365. Accessed: May 24, 2024.

[22] NVIDIA. "Building Your First LLM Agent Application." [Online]. Available: https://developer.nvidia.com/blog/building-your-first-llm-agent-application/?ncid=pa-srch-goog-289887&_bt=692291200545&_bk=llm%20ai&_bm=p&_bn=g&_bg=168012264468&gad_source=1&gclid=CjwKCAjw9cCyBhBzEiwAJTUWNZ9UzT4aVGZn5IgKcR5bVaAagECdUAhKU1tvzYTCtV2rknMwru30jhoC5IsQAvD_BwE. Accessed: May 24, 2024.