A self assessment

Game Engine Basic

Team. 6
5585253 Gwak Woojin

1. Introduction

- As the sole developer of this project, I took on the responsibility of designing and implementing all aspects of the game, including story development, UI design, and game mechanics. This required a multidisciplinary approach and a deep understanding of various facets of game development.

2. Goal Setting and Achievement

- **Goal:** The primary objective of this project was to develop a board game-style application using Unity, leveraging 3D graphics to create an engaging and interactive experience.
- **Achievement:** I successfully implemented the core functionalities of the game. Through this process, I became proficient in managing 3D objects, integrating animations, and handling UI interactions within the Unity engine. This project significantly enhanced my technical expertise in game development and improved my ability to manage multiple development domains independently.

3. Achievements

- **Developing a Natural Movement System :**
  - Implemented a character controller to achieve smooth and natural character movement, seamless interactions with objects, and integration of responsive animations. This provided an immersive player experience by ensuring consistent and intuitive control dynamics.
- **Advanced Dialogue Parsing and UI Integration**
  - Designed and developed a **DialogManager**, enabling efficient parsing of dialogue data and its seamless connection with the UI system. This improved the narrative delivery by allowing real-time updates and visually engaging interactions.
- **Object Interaction and Data Persistence**
  - Built an **ObjectController** that effectively stores information about objects and facilitates diverse interactions based on the values assigned to each object. This feature added depth to the gameplay by dynamically altering interactions according to contextual data.

- **Game Resource Management Across Scenes**
  - Utilized a **GameManager** to manage in-game resources, ensuring their persistence across scene transitions. This eliminated resource resets and provided a continuous and cohesive gaming experience.
- **Comprehensive NPC Interaction System**
  - Developed an **NPCManager** to handle individual NPC movements and interactions with both the player and in-game objects. This allowed for complex NPC behaviors and meaningful interactions, enriching the game's dynamics and engagement.
- **Custom Answer Validation System**
  - Created an **AnswerManager** to validate user input and transfer the resulting data to other scripts, enabling its use across different scenes. This functionality streamlined the game's logic and provided a robust mechanism for cross-scene variable management.

4. What I Learned

- **The Importance of a Clear Development Plan**
  - Learned that having a well-defined design document is essential in game development. Accurately specifying details using visual references ensures that the project follows its initial plan, even when revisiting it later during the development process.
- **Prototyping for Efficient Development**
  - Discovered the value of creating simple prototypes rather than striving for perfection from the start. By testing basic functionality and outcomes early on, I was able to reconfigure and redevelop more efficiently, leading to better resource management and improved workflow.
- **Prioritizing Through a Comprehensive Proposal**
  - Realized the necessity of drafting a thorough proposal before beginning development. This allowed for clear allocation of required features and time, helping to prioritize tasks effectively and avoid potential obstacles during later stages of development.

5. Challenges and Solutions

- **Parsing Issues**
  - **Challenge:** While parsing CSV files, the use of ＂ or ， to separate rows and columns caused issues where quotation marks (＂) could not be used in character dialogues.
  - **Solution:** Improved the algorithm by changing the parsing method and keywords to distinguish fields, using alternatives such as the backslash

    (\). This allowed ＂ and ， to be parsed and displayed as part of the text.

- **Button Click Issue**
    - **Challenge:** Despite completing the script flawlessly, the button failed to respond to clicks.
    - **Solution:** Discovered that the Canvas Group in the button's inspector can block interactions. Resolved the issue by removing the Canvas Group component from the button's inspector
- **Scene Transition Issue**
    - **Challenge**: User data and parsing results were being reset or altered whenever transitioning between scenes.
    - **Solution**: Utilized Unity’s PlayerPrefs system to save data persistently across scene transitions. Additionally, ensured that PlayerPrefs-stored values were reset when the game ended to prevent unwanted data retention.

6. Limitations and Areas for Improvement

- **Limited NPC Functionality**
    - While I intended to implement active NPC behaviors, such as locking and unlocking areas autonomously, technical limitations resulted in NPCs remaining largely passive. Although I successfully developed the functionality to lock areas, further enhancements are needed to make NPCs appear more dynamic and responsive, simulating interactions akin to playing with real people.
- **Resource Constraints**
    - Due to a lack of resources such as UI assets and object models, development faced certain limitations. Moving forward, I plan to either purchase assets or develop skills in modeling and art to address these issues.
- **2.5D Integration Challenges**
    - Although I combined 2D images with 3D background objects to highlight the advantages of a 2.5D environment, the results did not meet expectations. I realized the need to study shaders and camera angles further and plan to explore these topics to achieve better visual outcomes.
- **Storage System Missteps**
    - Unaware of the storage limitations of Unity's PlayerPrefs, I mistakenly implemented an insufficient saving system. In the future, I aim to utilize external files and JSON-based systems to create more robust and scalable storage solutions.
- **Time Management Issues**
    - Due to insufficient time, I could not properly integrate background music into the project. This highlighted the importance of creating a

more detailed and realistic development schedule to ensure that lower-priority tasks are not neglected due to time constraints in the future.

7. Future Goals

- **Developing Advanced AI and NPC Interactions**
    - I aim to focus on creating games that emphasize interaction with NPCs by studying NPC algorithms and level design to implement more sophisticated AI systems.
- **Long-Term Game Development Planning**
    - To pursue long-term game projects, I plan to create detailed development lists and prioritize tasks based on importance, enabling me to design and develop games with a slower, more deliberate tempo.

project code on github : https://github.com/Buzadagi/LastTest