Maze Project Status Update - November 2023

**Overview**

The Maze Project, a 3D game developed solely by myself using raycasting, has progressed significantly following the specified guidelines and requirements.

**Technical Details**

**Platform and Compilation**

All project files have been successfully compiled on Ubuntu 14.04 LTS, employing gcc (Ubuntu 4.8.4-2ubuntu1~14.04) 4.8.4, adhering strictly to the designated gcc flags (-Wall -Werror -Wextra -pedantic).

**Code Structure and Functionality**

The entire codebase maintains adherence to the outlined organizational structure. All functions are meticulously commented, ensuring clarity and understanding. Function sizes are limited to a maximum of 40 lines, adhering strictly to one statement per line and an 80-column limit, fostering readability and maintainability.

**File Organization**

The repository structure remains well-organized, featuring a clear separation of source files into the 'src' directory and headers into the 'inc' folder, in accordance with best practices.

**Compliance and Quality Assurance**

**Betty Compliance**

The entire repository has undergone rigorous Betty checks, ensuring compliance with the specified coding standards and guidelines.

**File Management**

No object files (.o), temporary files (\*~), or unused source files have been pushed to the repository, maintaining a clean and efficient codebase.

**Progress and Milestones**

**Functionalities Implemented**

Key functionalities of the game, including raycasting for 3D rendering and maze generation, have been successfully implemented and tested.

**Testing and Debugging**

Rigorous testing and debugging procedures have been conducted, addressing potential bugs and enhancing overall stability.

**Challenges Faced**

1. *Compilation Challenge in wall.*c: During the development phase, a significant challenge arose specifically in the 'wall.c' file during compilation. It required a nuanced resolution wherein I needed to replace the use of `abs` with `fabs` due to a compilation issue. This adjustment ensured seamless compilation and functionality within the codebase, highlighting the importance of adaptability and troubleshooting in the development process.
2. *Solo Development*: Embarking on this project solo presented a notable non-technical challenge. The intricate nature of developing a 3D game using raycasting proved to be an arduous task. However, the project's inherent difficulty didn't deter progress. Despite navigating complexities alone, perseverance prevailed, and overcoming hurdles became a testament to embracing challenges with resilience and determination.
3. *Embracing the Process*: An overarching principle that guided the project was the embrace of difficulty as an inherent part of the development journey. Finding joy in tackling complex tasks and persistently striving to overcome challenges contributed significantly to the project's progress and eventual success.

**Collaboration/Communication**

**Contribution from Harrison Nyabuto**

Collaboration played a pivotal role in the early stages of this project. Harrison Nyabuto's input was instrumental in formulating an effective framework to approach the complexities of the Maze Project. Through collaborative discussions and brainstorming sessions, we devised a structured plan outlining key strategies and methodologies for tackling the various aspects of the project.

**Framework Development**

Harrison Nyabuto's contributions facilitated the establishment of a comprehensive framework that delineated the project's scope, division of tasks, and an organized approach toward implementing the game's functionalities. Their insights and suggestions significantly influenced the project's initial direction, enabling a more streamlined and efficient development process.

**Importance of Collaboration**

The collaborative effort underscored the importance of open communication and leveraging collective expertise. Harrison's input served as a catalyst in shaping the project's early stages, emphasizing the value of collaboration in navigating challenges and setting a solid foundation for the subsequent development phases.

**Project Rating and Progress:**

**Self-Assessment**

The current progress of the Maze Project is rated at 8/10. This assessment stems from the substantial achievements attained independently, particularly in the realm of code functionality. The rating is based on the realization of significant milestones amidst the inherent complexities of the project. Despite its challenging nature, considerable progress has been made, contributing to the high rating.

**Criteria for Rating**

The rating scale reflects the functionality and robustness of the codebase. Achieving an 8/10 signifies substantial accomplishments in coding functionalities, overcoming hurdles, and navigating complexities effectively. The self-assigned rating acknowledges the project's difficulty while emphasizing the substantial strides made in development.

**On-Time Completion**

Furthermore, the confidence in the project's completion within the specified timeline contributes to the positive rating. Maintaining progress in line with the anticipated schedule adds to the assurance of delivering the project within the defined time frame.

**Future Steps**

**Refinement and Optimization**

Focus will be on refining code quality, optimizing performance, and potentially adding additional gameplay features.

**Documentation**

Continuation of comprehensive documentation to ensure future maintainability and ease of understanding for potential contributors.

**Conclusion**

The Maze Project stands at a pivotal juncture, having achieved significant milestones in line with the outlined specifications. The next phase will emphasize refinement, optimization, and potential expansion to further enhance the gaming experience.

