Designing and Implementing a 2D Arena Boss Game Using the Unreal Engine

Ryan Peterson

Abstract

With the use of the Unreal Engine, a game engine capable of easing stages of the video game development process, and additional programming in C++, it is possible for an independent designer to implement a complete video game. To accomplish this, my mentor, Dr. Zoppetti, and I used assistant functionality built into the Unreal Engine, wrote supplementary code, and researched concepts of game development to execute a game of our own design. This research included an equal emphasis on the study of game design, as well as on the programming techniques necessary to create a game. We achieved familiarity with both the Unreal Engine and the C++ language, and experienced all stages of the software development life cycle. As a result of in depth research, design, and implementation, we have completed a 2D arena boss game from opening title screen to closing credits.

With no prior experience with the Unreal Engine, we were unsure of what features could be implemented in a one semester time frame. So, we decided to implement the most basic features first and only add details and polishing if time permitted. Fortunately, some of the time spent on the game development process was reduced by features of the Unreal Engine. In the end, more functionality was implemented than ever was anticipated.

Commentary

The finished game contains one roaming enemy, two intelligent enemies, and a player character with actions and animations for walking, rolling, swinging a sword, throwing a boomerang, throwing darts, and shooting a bow and arrow. The concepts for the enemies came from many years of experience with all sorts of classic enemy actions, and a focus was applied to designing and implementing distinct boss enemies that made use of classical artificial intelligence patterns. These enemies, along with the player actions, were implemented primarily with the built-in blueprint tool of the Unreal Engine. Additional functionality was developed in C++ to allow for elements specifically tailored for this game's implementation. The pixel art was incorporated from sprite sheets based off of The Legend of Zelda. Menu screens, in game graphical user interfaces, lighting simulations, background music, sound effects, a title screen, and ending credits

were added to create a complete and polished product.

Having no prior knowledge of the Unreal Engine, a lot of research was put into discovering what tools the engine had that could assist the development process. One of the toughest challenges of the project came with learning the 2D functionalities of the engine, which, at the time, were just beginning to be implemented. This resulted in there being a lack of documentation of the engine's capabilities. But with enough research and experimentation, the potential of the engine was unlocked, and this built-in functionality certainly eased the making of animations, sounds, menus, title screens, and scrolling credits.

Conclusion

Through researching the Unreal Engine, the ability to create a complete video game from start to finish was fully realized. We successfully designed and implemented a 2D arena game with intelligent bosses, multiple player actions, and many polished details, all as an independent developer within a single semester time frame. Through this project, we experienced all stages of the software development life cycle, implemented features with C++, and learned a lot about the Unreal Engine. In addition to gaining software development experience and learning a tool used in the current game development industry, we got to pay a great tribute to the games we grew up playing and loving.

Recommended Citation
Peterson, R. (2016). Designing and Implementing a 2D Arena Boss Game Using the Unreal Engine. *Made in Millersville Journal*, 2016. Retrieved from https://www.mimjournal.com