PROG 445, Application Design

**Testing Phase of SDLC in Game Development**

When a development team is producing software, the structure in which they normally follow is called the Software Development Life Cycle, or SDLC. This applies to all forms of software development, from applications to websites and even games. The typical SDLC has the following phases: planning, defining, designing, building, testing, deployment, and maintenance (*Software Development Life Cycle*). Depending on the organization, some of these phases are merged together or excluded entirely - this commonly occurs with the maintenance phase as some companies leave that step to their clients.

In game development, one of the most important phases is the testing phase, as it catches any bugs that players may encounter and helps to inform gameplay features such as balancing. Members of a development team known as Quality Assurance testers perform rigorous repeated tests (Eden). Some developers make software through Test Driven Development, or TDD, which integrates testing into the building phase of development. Ash Davis of Real Serious Games describes the benefits of TDD as being oriented towards high-quality outcomes (Davis, 2016). Testing allows for teams to discover if they are meeting their own definition of success, and if they are not then they will discover why that is.

There are many different types of tests that QA Engineers perform in order to identify any pressing issues when it comes to the game’s development. Some of these are automated tests -which include unit tests, smoke tests, and integration tests – and these often test the functionality of key components of the game’s code (Davis, 2016). These automated tests do not require employees to invest time and effort into testing game components over and over, unlike manual tests. Most manual tests focus on repetitive actions and locate bugs that the development team would not have ever considered. QA testers will perform actions that most players would not even perform, like running into walls repeatedly or spamming controls, to discover any hidden glitches. A soak test, another kind of manual test, lets developers know if there are any performance issues related to keeping the game running for a long period of time, like memory leaks (Davis, 2016).

While it might not be the most fun part of the software development process, testing is extremely important – especially for the end users and clients. Developers can never anticipate everything that can go wrong with their software or how users will actually use it, which is how a lot of bugs exist in the first place. This is why the testing phase of the Software Development Life Cycle is absolutely required and should not be glossed over. It is also important to start testing early, as some bugs can take an unexpected amount of time to fix before release. If you ignore the significance of testing your game, you may end up shipping a game that barely even works!

**Works Cited:**

Davis, Ash. (2016, July 26). *Testing for Game Development*. Game Developer. Retrieved 18 Feb, 2023, from <https://www.gamedeveloper.com/programming/testing-for-game-development>

*Software Development Life Cycle.* W3 Schools. Retrieved 18 Feb, 2023, from <https://www.w3schools.in/sdlc/software-development-life-cycle-sdlc/>

Eden, Martin. *Role of Testing in Game Development.* Melior Games. Retrieved 18 Feb, 2023, from [**https://meliorgames.com/game-development/role-of-game-testing-in-game-development/**](https://meliorgames.com/game-development/role-of-game-testing-in-game-development/)