**Postmortem**

**TEAM #**

**Derek Finch, Josh Kopp, Jordon Kopp, Vince Smeraldo**

**10-22-2012**

**Team Lead:**

**Vince Smeraldo**

**[GSP362: Applied Development Project II]**

**Instructor: [Professor Sharpe]**

**POSITIVE ASPECTS OF PROJECT DEVELOPMENT**

* **Experience gained with Game engine**
  + This project was our first experience with a commercial engine. We have done some prior work with a small engine that was developed by students but this was our first hands on experience using libraries for a game created by a legitimate development team. We learned the engine by going through tutorials as well as making small mock up programs to test various features out.
* **Team communication**
  + Team cooperated well together to complete some tasks. We had several forms of contact and made time for our meetings. Throughout this project we would meet during, after, and outside of class times to work on this project and bring each other up to speed on what was going on and where we were headed. There were also cases where individuals would get stuck and ask a team member for help and then they would collaboratively work together on the task. We would bounce ideas off of each other as well as figure out a solution and then divide and conquer to implement it.
* **Learned to use version control system**
  + Version control is a huge part of game design and software development in general so we took this opportunity to gain some experience working with this tool. It eliminated the need to have a flash drive that we would pass around that compiled all of our work together, this was an online storage with easy access and monitored changes so you could also see what people changed since the previous commit. It also sped up the coding process because everyone could easily access the most recent build of the project.
* **Learned Java**
  + This project gave us the chance to learn another language other than C++ . Java was easy for us to pick up, because its syntax was very similar to C++. We are no longer dependent on using C++ in our projects and have expanded or options for future work.
* **Went through the entire process of the game development process**
  + This was the first project oriented class that allowed us to go through the complete game development process. We went from start to finish with concept and design, to prototyping to actual development. We also understand the idea of crunch time better now. We realized that some of our original ideas could never make it into the final game and that some things do change along the way of development.
* **Learned Microsoft Project**
  + Microsoft Project gave us the ability to easily break down the tasks for a given week and know a general time frame. This is a very powerful tool that gives you the ability to plan out a project and project completion times. It gives you the ability to see where the project may bottleneck so you can better distribute your resources. Tracking progression is also an extremely useful feature.

**NEGATIVE ASPECTS OF PROJECT DEVELOPMENT**

* **Team communication**
  + There were definitely times when communication was at a low and people weren’t exactly sure what the rest of the group was working on. This slowed the process because some people would wait on others because of dependencies.
* **Designing and planning the game**
  + The person we found that would make models and animate models never provided any work. There were also instances early on when everyone wasn’t on the same page and there were misunderstandings of what was going to be in the actual game. Also more design could have went into the actual coding structures as well as providing more documentations because sometimes people would look at code and be unsure as to what it was doing, this could have been avoided.
* **Too dependent on JMonkey**
  + No backup plan for different engine if JMonkey had gotten licensed. We were heavily invested in the JMonkey engine and its libraries. If it were to become licensed then we would have to search for a new engine with similar capabilities so everything transferred smoothly.
* **Procrastination**
  + Sometimes we would start on a task late and then it would take longer than we had anticipated. This would cause everything to fall behind. There were also lapses of time when everyone would get lazy and their productivity would be at a crawl slowing down the entire project.

**How can we repeat our success in future projects?**

Have good team communication. We need to keep each other notified on what we are working on and anywhere we are having difficulties so that if someone finishes their task early on they can go back and help someone who is struggling.

**How we can prevent our troubles in future projects?**

* We can become completely familiar with the tools that we are using. We would test out the tools in JMonkey and NiftyGUI but we are certain they can be utilized better and more efficiently. Instead of learning along the way, we could set aside time to fully understand the tools at our disposal and save wasted time spent trying to learn new features on the fly.
* Fully Design everything. We should have went through and created a class diagram for everything and set up the flow of the program before we even started coding. We ran into a few issues because sometimes we would change an parameter of a function or constructor and then we would need to go back and find every case we called that function.
* Learn all the features of version control software. We ran into a few issues where someone’s local copy of the project would be conflicted with the copy saved online, causing them to have to reset their local copy and manually copy over any changes they made. If we had a full understanding on how to use the software these occurrences could have been avoided.