Game 541 Individual Milestone Report

Milestone	*Engine Proof*
Name	*Trevor Harron*
Team Name	*Orbital Launch*
Primary Role	*Producer*
Additional Role	*Game Designer (General Programmer)*
Self-Score	*+1.0%*

Coding

Filename	Description of code you added	~# Lines	Functional?	Complete?
GameEngine.cpp	Data-driven Configuration & ensure console would be used	15	yes	yes
FileManager.h/cpp	Centralization of I/O Operations	100	yes	Yes & No (needs saving, reading is complete)
InputFacade.h/cpp	Provided abstraction layer for Input manager allowing for the use of Controller vs Keyboard and implemented in replacing the uses of InputManager directly	240	yes	yes
InputMangager.h/cpp	Provided functionality to ensure gamepad input would be accepted as well as mouse input	200	Yes	Yes
GameObjectFactory.h/cpp	Fixed bug in ensuring that there are not issues in reading Level files and GameObject files	10	Yes	Yes
MemoryManager.h/cpp	Created manager to ensure all objects are deleted after use properly	40	Yes	Yes
GraphicsManager.cpp	Enabled Flipbook animations and background images	30	Yes	Yes
Sprite.h/cpp	Enabled Flipbook animation	35	Yes	Yes
Body.cpp	Initially enabled gravity and collisions for platforming	35	Yes	Yes
Spikes.h/cpp	Created new component for spike object	25	Yes	Yes
LevelState.h/cpp	Tied player collision to the completion of a level	10	Yes	Yes
Controller.cpp	Edited to use the InputFacade	10	Yes	Yes
`stdafx.h/cpp`	Created and set up Precompiled headers	35	Yes	Yes

Approx. Hours Spent Coding: 50

Approx. Hours Spent Designing/Architecting: 90

Approx Hours Research/Debuging: 70

Debugging

For my debugging efforts I relied on the console output as well as stepping through the code and debugger in instances of exceptions being through. My philosophy was to use the debugger for issues with the code itself while the console would be used to track behaviors and ensure that everything was being initialized correctly.

Optimizing

For this milestone there was not any significant effort in optimizing the engine with the exception of ensuring that all events and objects that are created get cleaned up properly with the MemoryManager. This prevented the memory leak caused by events not being cleaned up previously as well as centralizing game object cleanup.

Game Designing

I designed the sample level and have spent about 20 hours on design concepts and working to ensure that the engine's capabilities would cover these concepts. I designed the sample level to show off the different states of the engine running from the simple playstate, to the character death, and character success. All of the assets and level files for the level (excluding the music) I designed and implemented.

Testing

As I was implementing new features, such as the Input Facade and accepting controller input, I attempted to test iteratively, add a single feature, test to see that it works. This was also extended to testing the sample level to ensure that the different features that the engine needed would also work properly.

Art

For the simple flipbook animation about 1 hour was spent in creating the sprite map. Other assets were either created within 30 min in a simple image editor or already present in the game engine.

Research

For this Milestone, I spent a fair bit of time studying game programming patterns (Game Programming Patterns) as well as Game Engine Design by Jason Gregory to attempt to (re)architect the engine most effectively to ensure a more solid overall design. Additional research was also done looking up documentation for SDL and OpenGL to architect input handling and (eventually) level creation and to ensure that the correct functionality would be obtained. In addition to this, time was spent studying how to best create a scripting engine (with lua) using the book Game Programming in C++: Start to Finish. Needless to say scripting has yet to be implemented. Outside of (this) class, I have also spent time reading and researching game design as a whole with The Art of Game Design, Rules of Play, and Digipen's own Engagement Papers.

Organizing meetings & events

As the Producer I also spent time trying to organize the team with Trello, task management, as well as setting up meetings 4 out fo 7 days a week to go over our progress and work together. I estimate that this took 10 hours of time or so to ensure that the tasks were created, assigned, tracked, and completed.

Helping other teams

Since my engine was the base for our team, I spent about 20 hours helping my teammates understand the structure of the engine, conventions used, philosophy, and implementation of different parts. As a person who has used GIT my entire professional life, I also taught my teammates how to use Git and help troubleshoot most of the issues that we encountered. As the producer, I wanted to help enable my team to preform at their best in helping debug code, handle schedule differences, or even if they needed water.

Getting help

Outside of the Orbital Launch Team, I would like to provide special recognition to Dhrumil who was critical to helped me understand the steps required to implement flipbook animations.

Preparation for presentations

For the Presentation, I created all of the slides in Engine and created the sample level to show off the Engine's basic abilities as well as the Presentation. This was handled by changing the backgrounds of the different states and cleaning up the assets that were not needed for the sample level as well as creating a few new components and assets. With the team, we outlined what we needed talk about as well as ensuring our engine had all of the basic required features.

Other

My engine from CS529 was the base engine Orbital Launch's Basic Engine is built upon. This engine only had used basic collision and reflection physics previously and needed several features added to it to be ready for the Milestone. Initially we planned on building a new engine from the ground up but were finding that there were configuration issues that would stubbornly refuse to be resolved (despite appearing to be correct). After seeing that we would fall drastically behind if we continued, as the producer I made the tough decision that we needed to start with one our engines as a base instead. While this did result in a loss of work it also meant that we ended up becoming more productive as a result from not having to repeat unnecessary work. As an important note, we had a GIT issue that affected 2 members (myself and another) for 4 days and the other member ended up being affected for a week plus caused by something on the IT side. During this period we consistently worked/emailed IT to bring about a swift resolution as well as resorting to sharing files via USB to ensure that productivity could continue although hindered significantly. For my milestone report I would admit that I came up a bit short in some of my personal goals (having a scripting engine and level editor) but will work to have those completed shortly to ensure the smooth creation of the game prototypes.