Pet Roll: A Game Project

Ben Willcox

ITE 505

Dr. Manish Wadhwa

Overview:

- Objectives

- Explanation of project change

- Problem Specification

- Amendments

- Solution Design

- Tool List

- Time Schedule

- Grading Scheme

- Post-Mortem

- Demonstration of Project

- Deliverables

**Objectives**

When I came up with this project a goal of mine was to build an understanding of Unity as a game development platform. I was curious as to what its abilities and limitations were, and how much I could learn about it within the timeframe I had. Along with Unity, came C#. C# is the programming language used by Unity when it comes to scripting objects and the game itself.

I was also curious to see where my previous game development experience (in Game Maker and a couple of other less ambitious platforms) would come in handy and/or prove itself useful. With these things in mind, the main goal was to combine all of this and create a game with what I had previously learned and was in the process of learning.

**Explanation of Project Change**

Edwin and I originally planned on collaborating to complete a blockchain-oriented project using AWS. We called it “Grocery Store Blockchain”. This would have been a partner project, and we came to learn that we would need a lot of university funding for this project for it to proceed at all. We were both not comfortable with spending the school’s money in such large amounts, especially since we did not know quite how much we were going to need.

Our schedules when coming up with a project idea during the ITE501 semester were also basically the opposite of each other. This is a factor that probably would have continued into this semester and would have impacted our work since we would have struggled to find time where we could have both worked on it together. Both the money and time factors made us realize that we should switch up the project, so we split off and each worked on individual projects for this semester.

**Problem Specification**

I wanted to create a 3D game, and once I did some exploring, I landed on a recreation of games like Hamster Ball or Monkey Ball. Hamster Ball was a game I played when I was younger on a much older computer in the public library. I also wanted to gain an understanding of C#, and in general acquire the skill of being able to troubleshoot code using the Web and past user’s experiences. I also wanted to learn how material, texture, model design came together into the game in a specific way. For example, Blender is the main software of choice when it comes to 3D modelling in the professional world today.

**Solution Design**

Since I based my game off Hamster Ball, I started with making a ball roll in my first game level and then worked from there. I designed some initial obstacles and ramps in Blender, and later on I ended up finding Unity’s Pro Builder, an asset package that allowed me to create 3D models in Unity without having to go through the process of exporting them out of Blender and into Unity. I then turned to the Web and found several copyright-free sources. I found my game’s soundtrack, as well as some textures to use. I did my own sound design, (although there is not a lot of it) and recorded it using Audacity.

I was starting from scratch when it came to coding in C#. This meant that I had to look up a lot of stuff, and then derive what I could use and combine it all in a meaningful way. The only downfall of using other people’s games as a reference is that their game is not your game; not everything works the same way when it comes to the scripts. Beyond variable names and layout, different games are going to interact differently. For example, a health bar and a player.

**Tool List**

Unity, with ProBuilder (Unity asset package). I believe I also used an asset package that allowed me to have the camera follow the player around without going through too much hassle. I also used Visual Studio Code to create and edit scripts which were programmed with C#. Blender was used for a lot of the 3D modelling, but particularly for the player model. Audacity was used for sound design as well as clipping the soundtrack to start at the right point.

**Time Schedule (proposed at the beginning of the semester)**

Week 1:

* Initial decision making and finalization

Week 2:

* Learning and viewing documentation and tutorials. Initial knowledge acquisition, as well as starting the game and programming player movement.
* Start creating models (continued throughout the project for the most part.)
* Laying out framework for the game

Week 3 (Jan. 30th-Feb. 5th):

* Level design
* Models for levels (Blender)
  + Such as cubes, blocks, ramps, etc.
* Also using ProBuilder addon for Unity
* Model for player
* Main menu
  + If there is time, do options menu
  + Specific details for that include FOV, sound

Week 4 (Feb. 6th-12th):

* Start looking into saving game to drive/basic save function per player
* Continue pause/main menu work
* Learn more about fun level features
* Moving platforms, lava pits/spike traps,

Week 5 (Feb. 13th-19th):

* Animation work for main player model, and any other additional ones I feel I have time for
* Study and implement sound design (movement, damage sound, etc.)
* Increased attention to detail (materials, textures)
* Will be using a lot of free textures to save time

Week 6 (Feb. 20th-26th):

* Combine textures/materials to new level and player mechanics
* Combined work of weeks 4 and 5

Week 7 (Feb. 27th-March 5th):

* Goal is to have at least 3 working levels done, as well as:
  + 3 working levels includes a mostly polished tutorial level
* 2 player models
* Implement a menu for choosing unlocked content
  + If I have time, this will be unlocked additional playable characters

Week 8 (Mar. 6th-12th):

* Continue implementing a save feature for the user (for unlockable characters)
* Work on user manual

Week 9 (Mar. 13th-19th):

* Vacation week
* “Bonus time” that I will spend on bug fixing and starting some beta/newer levels
* Work on user manual (finalize if possible)

Week 10 (Mar. 20th-26th):

* Continued work on additional levels and unfinished work
* Additional levels tend to be time consuming

Week 11 (Mar. 27th-Apr. 2nd):

* Ensure .exe and related files to run the game are in order
* If not, add modules to the Unity Explorer/figure it out
* Finalize game sounds
* Finalize game menu

Week 12 (Apr. 3rd-9th):

* Continued game work
* Finalize user manual
* Finalize project report

Week 13 (Apr. 10th-16th):

* Continue work on final submissions (listed in the week below)
* No more new content in the game for now, just work on finalizing a product

Finalize project journal (for the most part)

Week 14 (Apr. 17th-23rd):

* Project report, commented source code, user manual, project journal, post-mortem, and presentation are all due on the 25th
  + Finish all of these

**Time Schedule (Closer to what really happened)**

Weeks 1-3:

Bulk of 3D model work and level design. Thought up the player model.

Weeks 4-8:

Levels were implemented in these weeks, as well as adding in more. Many game objects were added, and textures were found.

Weeks 8-12:

Combining everything together in the code so that everything worked well. Final touches on level design took place, and more attention to smaller details.

Weeks 12-14:

I worked on the paperwork doing these weeks. I also added some final UI/UX elements to make the user experience more friendly, such as level indicators and a way to return to the first level.

**Grading Scheme (from updated proposal)**

40% - Workable ‘final’ product for a game/playability

20% - Managed and organized workflow (minimal/no spaghetti code, organized directories)

10% - Minimal/no bugs in delivered game

10% - Meetings

10% - Report

10% - Presentation

**Post-mortem (from original file, slightly updated)**

When it came to the original project proposed, the initial concern with the money that would be required to proceed with the project that Edwin and I came up with last semester was brought up by Edwin, and he raised a good point. He was uncomfortable with the fact that the project would take a lot of the university’s money, and we also did not know how much that would end up being. This would have put a lot of pressure on the both of us to sync up our schedules and get a lot done to ensure that we were putting the school’s money to good use. This would have been too stressful, so we both ended up switching to new, individual projects.

I ended up writing up a new proposal in January, and I also wrote up a new time schedule. In the time schedule, I was too optimistic about what I would end up having time to accomplish in the game. The first 2-3 weeks of the project were the most productive and the most time I managed to spend on the project daily was about 7 hours.

After that, I referenced the time schedule purely to ensure that I knew what I should aim for next. A lot of steps took longer to accomplish than I would of liked, and many steps were skipped because when I wrote up the time schedule, it was before I had experienced what it was like working in unity. Starting out with a save feature, I did not implement this. I implemented some serialization (which is saving game data) when it came to coins between levels. That was it, I did not implement a new/load game feature because once I started researching all of it I came to the conclusion that it was going to be too much work for the amount of manpower and time that I had to work with. I also did not get around to additional playable characters, because getting the first and only character model done was already a hassle and the animations did not load into Unity correctly. In the future I would enjoy working with a group of people on a game if I had the opportunity to.That way each person I involved would work on a certain part of the game, like one or two programmers, a designer/artist, and someone on sounds and music maybe.

If I remember correctly, way back when we were initially thinking of project ideas Dr. Wadhwa brought up the idea of designing a game using Unity, and me and Edwin could have gone with that. I think this could have been a good idea because each of us could have focused on different aspects of the game. Obviously, I cannot change the past, but I believe that both of us could have collaborated here and come up with a game that is 2x better than the one I produced because we would have both been contributing to the project.

Overall, though, I should note that I am happy with the progress on the game that was made. I got a lot of game objects and level work done, and it was no easy task to come up with a new design for a level every time or figure out what game object would make the level more interesting.

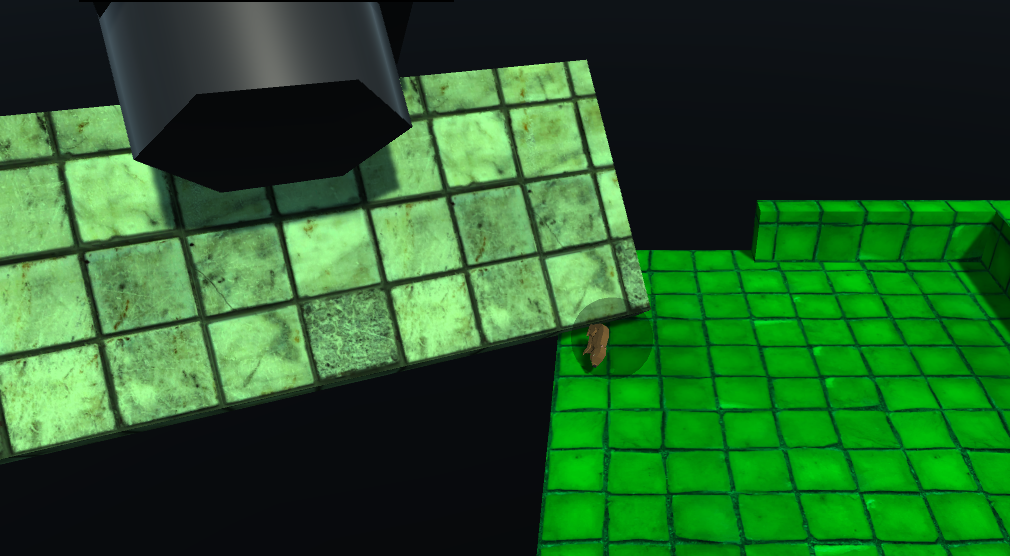
**Demonstration of Project**

[Link to Mass. URC video](https://www.youtube.com/watch?v=BUjfEWqs8sk)

My Game Manual has a few screenshots of different game objects for reference, as well as instructions on how to play. Here are some screenshots from the game.

A screenshot of a computer

Description automatically generated with medium confidence



Above: You can see the coin element, health bar, as well as level indicator as elements of the UI. In the game itself you can see the player object, as well as some obstacles. In the second image, we get a closer view of one of the obstacles, which is what I call a “crusher”. This object can be used to attempt to push the player off a ledge, or to boost a player, like an elevator. In this way, it has many purposes. More descriptions of various game objects can be found in my game manual.

**Deliverables**

● amendments to original proposal/time schedule

● appropriately commented source code (C#).

● documentation of project functionality (final results including screenshots, some video)

● user's manual

● executables and/or projects

● report and presentation files for ITE505 (can provide old 501 files if needed)

● complete project journal

● project post-mortem

- a list of what areas of the proposal/time schedule were not completed, and why

- Reasons for switching projects