Gameplay Prototype Report

Project Sound - Dark Energy

Progress Report

For the past two weeks we worked on the Gameplay Prototype, which was created using GameMaker. The prototype was, in essence, a complete mock-up of a level in our final game. We showed off all of our basic gameplay elements in the prototype, such as the monster AI, player pinging mechanism and light and shadows mechanism. We tested out our prototype and made three revisions to the prototype according to the feedback.

In addition to the prototype we also wrote the Milestones document. We also generated sound effects and music for our game (including basic atmospheric music, boss-fight music and sound effects for pinging). Our basic visual assets (graphics for player, pings, wall tiles and monsters) were also created and revised for a more polished effect. In addition, we had group tutorials which consisted of the senior programmers helping the junior programmers with specific elements such as GUI, event firing, etc.

Activity Breakdown

Nick - Helped test out, provide feedback for and debug the gameplay prototype. Taught junior programmers in XNA tutorials.

- Gameplay Prototype (input and debugging) - 2 hours
- Sound Assets - 1 hour
- Tutorials - 2 hours
- Milestones document - 1 hour
- 2-week report - 1 hour

All of the activities were a valuable use of time because they contributed to the group and the game significantly. Especially of value were the GUI tutorials, since the junior programmers had never used C# to build a user interface before. Showing them around the various classes and interfaces available was much more time-efficient than leaving them to figure it out themselves.

Elle - Helped test out the gameplay prototype and suggested areas of improvement on the prototype. Contributed to the milestone document by adding in the Risk Assessment parts to the document, and fixing the Test for Assessment parts. Started learning how to implement GUIs in preparation for the Technical Prototype.

- Gameplay Prototype - 1 hour
- Milestone Document - 3 hours
- 2-week report - 2 hour
- Learned about GUIs and Windows Forms on XNA- 5 hours

These activities were a valuable use of time, especially the time I spent learning about GUIs as this will help with the completion of the next milestone. Also playing the gameplay prototype has provided me a better understanding of how our game will be played, along with the interactions of the player and the enemies.

Rena - Helped test out the gameplay prototype. Suggested areas of improvement on the prototype. Provided support on milestone documents and gameplay specifications. Learned how to make use of XNA's visual framework (this will be useful for creating the GUI for Level Editor later on).

- Gameplay Prototype - 2 hours
- Milestone Document - 1 hour
- 2-week report - 1.5 hours
- Implementation of GUI and event firing with XNA - 5 hours

All the activities were a valuable use of time, especially learning GUI implementation. The XNA visual framework will be very useful in the future for creating the level editor. Critiquing and discussing the gameplay prototype did not only make our concepts of the game more well defined, it also ensured that everyone on the team has the same expectations of the game. I should have spent some time coding in XNA to set up the basics of the game so that we would have less work for the technical prototype.

Natalie - Found an alternative winning strategy in the gameplay prototype, unexpected by the level designer. Also made simple graphics to be used in the prototype, then adjusted them for sizing and user feedback. Filled in production dates, tests for acceptance, and contributions for the milestones document, then converted the file to PDF for submission.

- Initial Graphics - 30 minutes
- Playtesting/Cheating - 20 minutes
- Milestone Document - 1.5 hours
- Edited Graphics - 2 hours
- 2-week report - 1 hour

Creating the initial graphics was necessary for the gameplay prototype to be constructed, so sharing those assets quickly allowed for Detian to begin work in GameMaker early. Playtesting proved valuable because I discovered that it was possible for players to explore the world off-screen to reach the goal without encountering very many obstacles or enemies. Editing the graphics may not have been crucial, but it allowed me to think more about how to provide feedback to users through visual cues. The walls were a result of testing perspective at an imaginary angle, critters were given bunny-like qualities to show their harmlessness, and enemies were given exclamation marks whenever they become aware of player presence. Sizes were also adjusted to improve maneuverability and demonstrate the level of threat of each character. Not all the updated graphics were incorporated in the presentation prototype, but I can still use what I learned from making them in future graphics assets.

***Anthony* -** Playtested and provided feedback for the gameplay prototype. Worked on the Milestones document. Created a Gantt Chart that plots our future deadlines for the Milestones.

- Milestone Document - 1.5 hours
- Gameplay Prototype - 1 hour
- Gantt chart creation/planning schedule - 2 hours
- 2-week report - 2 hours

All of the activities were a valuable use of time, especially the creation of the Gantt chart and milestone document. Each sprint phase consists of multiple deliverables that sometimes are interdependent (in addition to scheduled assignments, reports and labs). Reading through the future deadlines and creating a schedule as well as a gantt chart allows me to better understand what deliverables have critical and dependent deadlines. In addition it allows me to better understand what each group member should be working on at a moment in time.

***Detian*-** Implemented and adjusted the gameplay prototype according to feedback received.

- Created Gameplay Prototype - 6 hours
- Modify gameplay prototype to fit expectations - 2 hours
- Milestone Document - 1.5 hours
- 2-week report - 2 hours

This sole activity was very valuable as it produced the gameplay prototype. The team as a whole is very satisfied with the outcome. Although the prototype was created in GameMaker (and therefore we will not be able to reuse the code), since we managed to mock up a complete level with winning and losing conditions, it gave us a very good round of initial user testing and feedback. We changed a lot of mechanisms in our game that was imprecisely defined before.

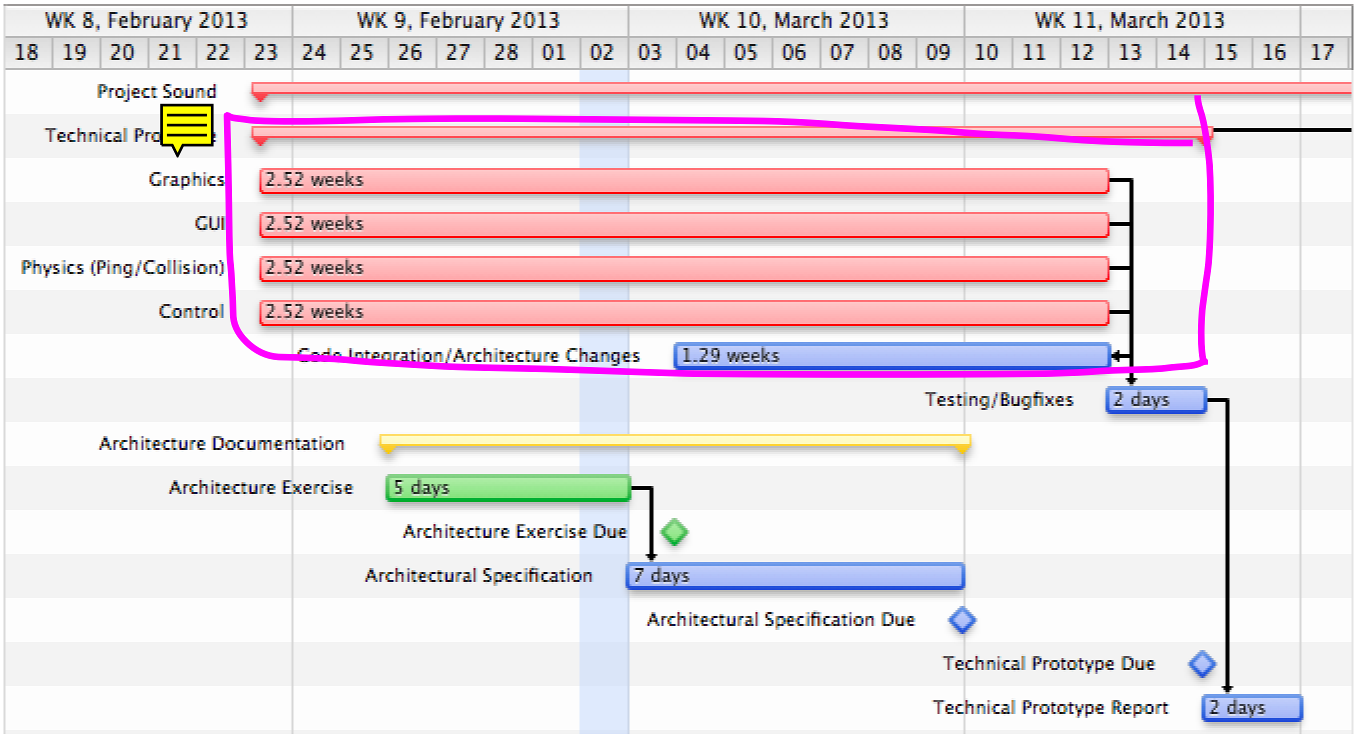
Productivity Analysis

Since this is the first two-week report we will make comparisons based on projections set in the Milestone Development document. According to said document the only deliverable was a playable prototype of one level of our game. Specifically, the prototype was planned to have rough graphics and sound. Testing the gameplay prototype was also planned.

We were moderately productive during this sprint phase. Our first version of the Gameplay Prototype was implemented by Detian more than a week before it was due. As a result the whole team had lots of time for testing and improving the prototype. Because we were able to finish the main deliverable quickly (in less time than we expected) we were able to spend more time testing the game and learning how our proposed game mechanics would function. Topics such as balancing and resource management were discussed during the testing phase. This is beneficial for future responsibilities as discussion regarding balancing was planned for later phases. Graphics and sound were also implemented as projected for this phase.

Our prototype now provides us with a basic idea of how our game will work. Even though we have yet to program in C#, we are actively working together to make sure our technical prototype is on schedule. Both the prototype and the graphics took less time than expected. Nothing took more time than expected. Responsibilities in the future still remain unchanged as we did not encounter any challenges that would affect our planned task distribution.

Milestone Predictions



Above is a Gantt chart for the upcoming milestone that details the deliverables due in the next milestone and our projected completion date. The main deliverables are the technical prototype and architectural documentation. Subsystems of the technical prototype include updated graphics, the game GUI, ping and collision physics and user control.

Test for Acceptance

Are the player actions implemented and working? This includes pings interacting with other objects, objects showing up in the graphics engine when pinged, the collision engine working, and the player moving. Also is the basic GUI implemented? Elements such as the sound-bar, and notifications should be visible.

Risk Assessment

One of the main risks that we face is not being able to deliver on time. Since we did our gameplay prototype in GameMaker, we will not be able to reuse any of that code. Implementation in XNA will also be different than it was in GameMaker. Hence, most of the other groups would have gotten a head start in comparison to us since we will have to start entirely from scratch. Also, since the gameplay prototype was entirely created by our lead programmer it might be hard for other members to start implementing elements since they are not familiar with the challenges and logic behind it. In addition, all the programmers are assigned tasks to complete concurrently, and the tasks must be combined and tested before the final submission. Besides delays due to bad time management issues may arise when combining code and fixing bugs.

Activity Breakdown

Elle - Will create the GUI. This involves making the sound bar and the menu screen. Will also start experimenting with Windows Forms in preparation for making the Level Editor.

- Group Meetings - 7 hours
- GUI Implementing - 4 hours
- Code Integration - 1 hour
- Architectural Specification - 2 hours
- Plan out level editor with Windows Forms - 6 hours

Nick - Will be working on implementing ping physics, running Will be working on ping physics and providing support for various coding tasks as needed.

- Group Meetings - 7 hours
- Architectural Specification - 3 hours
- Ping physics - 6 hours
- Teaching - 4 hours

Anthony - Will be working on collision physics, writing supporting documents and providing support for debugging and various coding tasks as needed.

- Group Meetings - 7 hours
- Collision physics - 6 hours
- Code Integration/Debugging - 3 hours
- Project Management - 3 hours
    - ○ Gantt chart adjustments, planning for future sprints - 2 hours
    - ○ Communicating with programmers and lead programmer - 1 hour
- Documentation - 3 hours
- Architectural Specification - 1 hour

Rena - Will be working on mapping the user controls on the keyboard to corresponding movements on the GUI (moving left, right, up, down and pinging). Will be synthesizing player's movements with collision physics and ping physics and making sure that they run smoothly. Will also be responsible for setting reasonable constraints and fixing bugs in player movements (e.g. making sure player doesn't phase through walls). Will also be exploring Windows Forms for Elle in preparation for making the Level Editor. Will start on the Level Editor if time permits.

- Group Meetings - 7 hours
- Architectural Specification - 1 hour
- User Controls - 2 hour
- Player Movement - 2 hours
- Animation - 3 hours
- Windows Form - 5 hours

Natalie - Will be working on updating graphics with preliminary sprite sheets for player, hunters, critters, and backgrounds. Possibly other assets as interface is created. May also help with various coding or scripting tasks as needed.

- Group Meetings - 8 hours
- Architectural Specification - 1 hour
- Character Designs - 6 hours
- Character Sprite Sheets - 2 hours
- Background Designs - 3 hours
- Miscellaneous Graphics Assets - 0 to 2 hours
- Ad hoc coding/scripting - 0 to 3 hours

Detian - Will be working on creating the framework for project so that others can begin working, including design of basic game logic, creating control structures, standardizing API's, and setting coding style. Will be making sure everyone is on the same page and following the same coding conventions.

- Group Meetings - 8 hours
- Architectural Specification - 1 hour
- Team coordination - 1 hour
- Technical Prototype - 10 hours
    - Framework - 2 hours
    - Game logic - 5 hours
    - Control structure - 2 hours
    - API/code maintenance 1 hour