

4/24/2006

Team #7: Pez

Project: Empty Clip

Members: Alan Witkowski, Steve Huff, Thos Swallow, Travis Cooper

Document: Final Team Report

1. Product-related information:

Current status of product

The game is currently fully playable. There is a story mode and an adventure mode, both of which have been extensively tested. Deathmatch is also implemented, though it has not been tested as well as the rest of the game. Due to time constraints, the story is not as well developed as initially desired, and plans to implement cooperative network play were scrapped.

Recommended work

The game could be enhanced by adding more levels, testing and adjusting balance, and implementing cooperative network play.

Advice to teams continuing this project

Use the database directory to change level content – no more coding should be needed in this regard. Familiarize yourself with the control flow of the code. Learn how to use the map editor. Play the game a lot to understand what the game is about and how it works.

2. Project team information: (Postmortum analysis)

Management objectives and priorities

The team leader will make sure all team members are completing their work, integrating correctly, and getting all the necessary information that they need. Individual team members are responsible for prioritizing their work, getting help when needed, updating their logs, and helping other members when needed.

Our goals for the project are to have fun, produce a functional and interesting game. We aspire to add character movement, monsters, levels, weapons, sound, multiplayer, puzzles, and good graphics.

Final team structure

Alan Witkowski – Team leader, graphics, collision detection

Thos Swallow – Networking, audio, A.I.
Steve Huff – Art
Travis Cooper – Menu system, GUI

The only role change was that Artificial Intelligence was switched from Steve to Thos.

The final structure proved to be good, and we would still use it if starting again.

Scheduling and planning

We had a good design from the beginning. We had a team leader that made sure everything that needed to be complete was covered and integrated according to our standards. We met every Monday to talk about what we had accomplished the previous week and we also made new assignments in order to help our project advance throughout the entire time. We also met on Wednesdays as was necessary to do additional documentation or pair programming.

Our main problem was our format. It took us awhile to find a format for file structures that were good enough. Certain tasks needed to be better defined before work was started. Sometimes there was some miscommunication and it was hard to talk through email to get a good enough description of what was needed to be done. As a result these tasks took longer than needed, and there was some work that was duplicated.

We used our website and e-mail. Also some side by side programming as needed.

If we had an agenda of things that we needed to talk about that was sent out before the meeting, team members could review necessary items and think about them before hand so that we could be better prepared for certain tasks. This would have helped us in having a more specific definition of what was needed. Especially this would have helped in the layout of our structures.

Support functions

We had good quality assurance – there are very few bugs left that we are aware of. For defect tracking, we had a web page linked from the main project site that described in detail each defect as it was discovered, and recorded progress made on fixing the defect. We were fairly successful in using this defect tracking tool, but occasionally we would just use email. Using something like Bugzilla that is designed for tracking issues would likely be easier and more effective.

Work with the clients

We didn't have a specific client to answer to; however, we did have many beta testers to help us find and fix bugs. Working with these beta testers was very simple because we had so many, and when there was an issue they let us know and we discussed these issues in team meetings.

There is nothing really we could have done differently given the nature of our project.

Work with project mentors

There was very little interaction with the project mentors. Most interaction was done via email, and mainly had to do with feedback on submitted documentation.

This can be made more useful by having in-person meetings, and feedback on the project itself rather than just our documentation.

Email was fine for documentation purposes.

Other issues

N/A

3. Feedback from the mentors

The only feedback we received from mentors was from the professor and ta. They were helpful in letting us know when we weren't keeping up our logs, and in which documents needed to be revised in order to meet expectations. This documentation was very, very helpful in finish the project.

4. Three general pieces of advice to future students

1. As they start out they should finalize modules within the project as much as possible before doing actual coding. This includes, but is not limited to, writing things out on paper, discussing them as a group, etc. This will help minimize the miscommunications in the group.
2. Give plenty of time in order to do a good job on documentation. This will be a valuable asset in making sure you have a successful project.
3. Do plenty of pair programming so that multiple people understand all the aspects of the code.