

Senior Project: AutoRPG

Mark Gius

Advisor: Dr. Zoë Wood

February 22, 2011

1 Introduction

Most video games on the market today are geared towards players with fast reflexes and favor quick action over careful planning. This trend has particularly affected the Console Role Playing Game. Where in the past the pace of the game afforded time to choose an action and select it, newer games progress at a breakneck pace, leading to less direct player interaction and more scripted actions. One such game, Final Fantasy XII, allowed the player to provide sets of actions for their characters to perform automatically in battle in addition to manually entered actions. [1] This preserved a significant amount of tactical planning in the game. A well-designed set of actions could often make the difference between victory and defeat.

A game that only allowed battle actions through pre-defined character tactics would force the player to think deeply about their opponents and characters in order to develop a winning strategy.

2 Previous Work/Related Work

There are several other games that feature similar gameplay mechanics.

RoboCode [2]

“Robocode is a programming game, where the goal is to develop a robot battle tank to battle against other tanks in Java or .NET.” Robocode contains a similar “AI” approach to player interaction, but requires the user to create their AI in a supported programming language. This limits the potential audience to programmers and those willing to learn.

Final Fantasy XII [1]

Final Fantasy XII is the primary influence for AutoRPG. Final Fantasy XII allows for the player to override the pre-programmed strategies at any time. This allows a sufficiently fast player to react to battle events as they occur rather than predicting them ahead of time.

What other solutions have been proposed or implemented for this problem? Why are they not good enough? You must prepare the readers to believe that your solution (which you have not yet said anything about) is better than these other attempts. This is also the section where you describe any work that your solution is built on top of. Include references to relevant related work.

3 Algorithm

(a) Overview of Solution. Describe in broad terms your project and solution (b) Main Algorithm or Interesting Details. Either describe the main algorithm in your project or if appropriate, Pick some of the key areas of your project that you are particularly proud of (that were either complex or tricky or just the most interesting to you). Describe in detail your solution to these aspects of your overall problem. Think about someone else trying to implement what you just did, what information would be most useful for them?

4 Results

Describe your results. Please include at least 3-6 screen shots of your results, which highlight the capabilities of your project.

5 Future Work

What is the next stage? What didn't you get done that you should have? Now that you're done, what should the next person do to carry on the project? Frequently senior projects are a portion of a larger project, and you have to make sure that your work is understandable by the next person to work on the project.

6 Appendices

(as needed) for proofs, user manuals, etc.

Please remember that this is a formal report that will remain in the library after you graduate. As such, it should be written with formal language and clear exposition. This report should demonstrate your mastery of the topic you have chosen for your senior project. It should be something you are proud of in the end.

References

- [1] Gambit system. <http://www.finalfantasyxii.com/>. Flash Website. Select "How to fight back," then "gambit system".
- [2] Robocode home. <http://robocode.sourceforge.net/>.