

# **Agile Game Development And Fun**

**Andrea Gibson**

**Andrea.G.Gibson@gmail.com**

**Advisor: Susanne Sherba**

**Sherba@cs.colorado.edu**

# Table of Contents

|  |
|--|
| <b>1. Introduction – Page 3</b>  |
| <b>2. Background – Page 4</b>  |
| <b>3. Related Woks – Page 8</b>  |
| <b>4. Experiment – Page 9</b>  |
| <b>5. Results – Page 11</b>  |
| <b>6. Experiment Conclusions – Page 18</b>                               |
| <b>7. Future work – Page 20</b>  |
| <b>8. References – Page 21</b>   |
| <b>Appendix A: HRC Application – Page 22</b>                             |
| <b>Appendix B: Player Responses – Page 28</b>                            |
| <b>Appendix C: Developer Metrics – Page 46</b>                           |
| <b>Appendix D: Developer Iteration Scrum meeting responses – Page 47</b> |
| <b>Appendix E: Developer Exit Survey – Page 53</b>                       |
| HRC protocol: 1106.15-01   |

# 1. Introduction

Video game creation is one of the many sub-types of software development and poses both unique problems and solutions. One of the main differences between video game development and conventional software development is that video games are mostly created for entertainment rather than practical or calculation purposes. Also, video games have a tendency to be complex, large projects which can make them difficult to create, because development can take years. Agile development, on the other hand, is designed to adapt and evolve along with the project. This type of development can help projects clearly see what is bogging them down, as well as see where they are going. Applying the agile process in the form of Scrum can potentially solve many of the problems that persist within game development.

This document will present what problems currently exist with game development, such as the complexity and non functional requirements that video games have. Also, it will go over a general overview of Agile, and a more in depth look at Scrum, the particular Agile method used in the experiment. Also, it will discuss the experiment process and recruitment process that was used. Along with the experiment process, it also explains what signature qualities were looked for.

## 2. Background

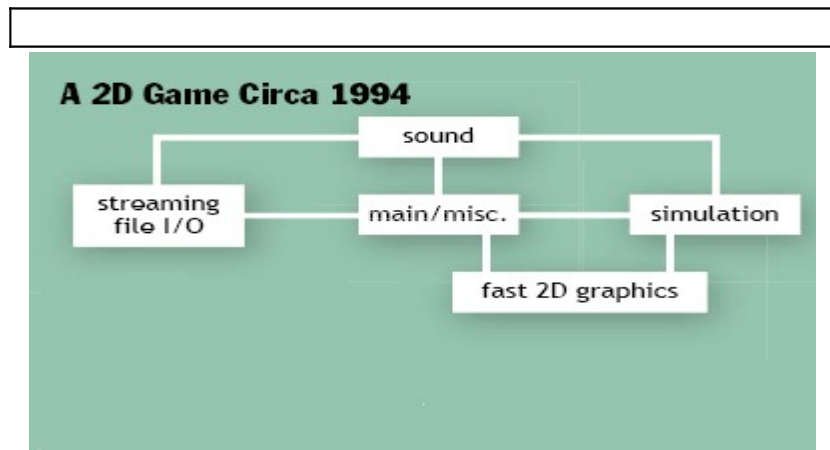
In this section we look at the problems associated with game development, as well as how the Agile process works.

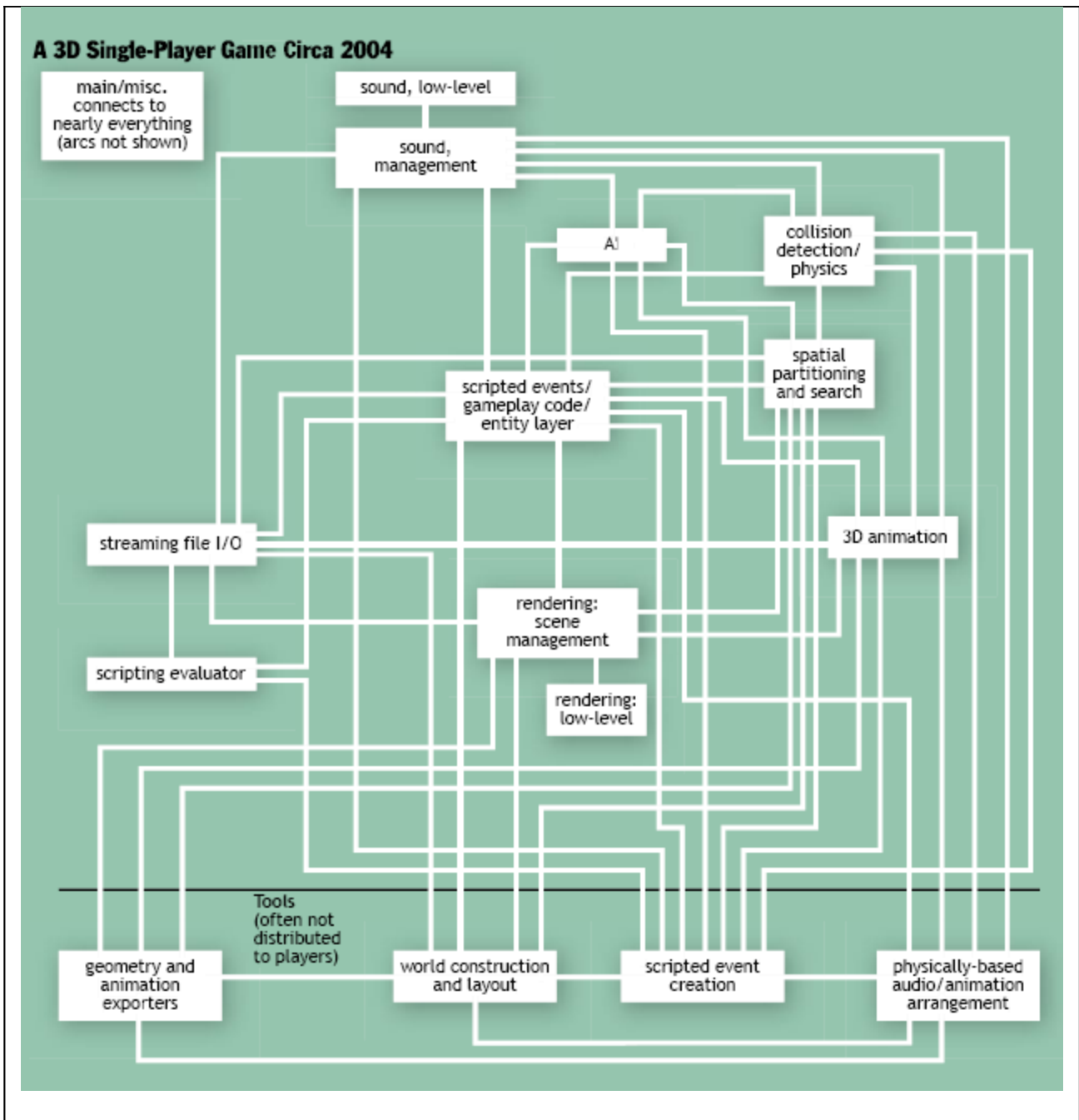
### 2.1 Game Development Problems

There are many problems with game development, one of the main ones being increased complexity from the size of game. Additionally, there exists the problem of it being hard to say what exactly is fun and what is not fun and how to attain it. Finally, there are issues with the communication between the various people involved with developing the game including programmers, designers and artists.

#### 2.1.1 Complexity

First, looking at the general complexity of code required in a video game, one can see that there are lots of different components that have to communicate in order to work properly. In the following graphs there are several charts that show the growth of these games and the number of components. [1]





Where in 1994 there were only 5 components, now there can be as many as 12 components in a single player game, not including the tools used to create content specifically for the game. For example, some of the more complex systems include the 3d graphics, the collision detection that goes with it, and the AI. Also, since there are so many different components, there is complexity that arises from the communication and dependencies between the various components.

### **2.1.2 Fun**

As we can see, while the functional requirements of games can be difficult, the fun factor presents a unique challenge to game design. As discussed in [2]. 'fun' is an experience between the user and the game. Since users all interact with the game differently, it can be hard to tell beforehand what the majority of target users will enjoy. For example, as discussed in [3] fun can include a sense of uncertainty about being able to win. With this definition of 'fun' there has to be an appropriate scaling of difficulty, so that the user can win, but won't always. There are several ideas outlined in this work including having the difficulty scale with level, randomness, the player not having all the information, and more than one goal. Creating this kind of content can be very hard, and requires lots of play testing to see if it will work.

### **2.1.3 Communication**

Communication is another important aspect to any software project. However, unlike typical business based applications, video games require input from people across many disciplines. Designers, programmers, artists all work together to make the final project As described in [4], there can be many different technologies being used in a single project. With all of these various types of people on the same project, and all with radically different facets of the same product communication paths can stack up very quickly..

## **2.2 Agile Software Development**

In this section, we will define the Agile process for better understanding. Additionally, we will define Scrum, an Agile methodology. Agile follows the Agile Manifesto found at [5] and it states:

Individuals and interactions over processes and tools  
Working software over comprehensive documentation  
Customer collaboration over contract negotiation  
Responding to change over following a plan

Basically, Agile is designed to be able to adapt to change. Products designs change throughout the project, and Agile helps to manage that change and keep everything under control. If the control is not possible, then it would be at least known that it is out of control.

### **2.2.1 Principles of Agile**

In [7], Martin outlines twelve principles that define the core of what agile is.

- 1) Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- 2) Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- 3) Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- 4) Business people and developers must work together daily throughout the project.
- 5) Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

- 6) The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- 7) Working software is the primary measure of progress
- 8) Agile processes promote sustainable development. The sponsors, developers, and users, should be able to maintain a constant pace indefinitely.
- 9) Continuous attention to technical excellence and good design enhances agility.
- 10) Simplicity- the art of maximizing the amount of work not done – is essential
- 11) The best architectures, requirements and designs emerge from self-organizing teams.
- 12) At regular intervals, the team reflects on how to become more effective then tunes and adjusts its behavior accordingly.

### **2.2.2 Scrum**

In [8] they lay out a great basic idea of Scrum, “The basic premise is that if you are committed to the team and the project, and if your boss really trusts you, then you can spend time being productive instead of justifying your work.”

Scrum is one of the methodologies based off of the Agile principles. The Scrum process revolves around iterations. It is recommended that you have these periods be shorter rather than longer, such as having them be two weeks. At the beginning of each iteration you gather a list of story points that the users want to be able to do, this list is gathered from both the client and yourself. With these you estimate a relative point value for each of them, this does not necessarily need to translate into real time. With these points you estimate how many you can do in one iteration and that is how many you try to do. This will start out fairly accurate but as you understand the project more, you will get more accurate with the estimates. With the user stories, you break those down into tasks and give those tasks and assign those hour estimates of how long they will take, estimate these until you have enough tasks to last the iteration. When you have established what you are going to work on for that iteration, that is what you work on and nothing else, this can be working on the design for the components, implementing or testing, as long as you are working on the user stories decided upon.

During the iteration you keep track of what you have completed on a burn down chart. With at the beginning of the iteration it at full points, and working towards 0 points. An item is not considered completed until it has been coded, and tested, an item is either 0% complete or 100% percent complete. At the end of the iteration, you do a reflection period to see what you felt was working and what wasn't working. Using this reflection as well as seeing how accurate your estimates were, you estimate for the next iteration and keep moving.

### 3. Related Work

In [6], Keith relates work that he has done in his company with agile. He relates his experience with switching to using Scrum halfway through the project *Darkwatch* and states, “Scrum is about making things visible so that you can make commonsense decisions.” Applying agile, and Scrum to game development is not a new idea, but it is something that would be nice to explore in more detail, especially in regard to bringing more ideas out early and often to help create more fun and unique ideas.

In [9] Rory McGuire lays contrasts and compares the Scrum process and the Waterfall process in regards to game development. Like Keith, McGuire also worked on the *Darkwatch* project, and shares the same belief that Scrum is much better for game development. One of the main reasons that he points out is that with the waterfall process there is a lot of stop and go because various departments are at varying levels of completion that the other departments are waiting on. While in Scrum it is possible to work on their own components independent of the other departments. The example he gives is creating a level around AI. Where in waterfall, the entire process would have to be designed first, then implemented and then tested, where during the design process, the developers wouldn't have anything to do. While in Scrum that process would be little pieces get designed, then passed off to be implemented, and if the design changes then the developers would be given that change. Also, if the developers found that a certain piece of AI was impossible, they could report back to the designers to change that aspect.

Another point that McGuire brings up is that the ones who know the most about various field are the ones who work with it. Basically stating that the programmers know the most about coding, the animators know the most about animations. And this includes knowing the most about how to do things, as well as what is impossible and what land mines could come up. Then with Scrum it gives the power to those in their own field, not giving it to management to tell them how to do it. McGuire states that, “Designers aren't programmers or artists so presuming we know the best path in all methods of technology and art is foolish.”

Also, even not looking at Scrum, there is other research that looks at involving players in the design process. In [10], Ermi and Mayra discuss the creation of a mobile game, and how they used player feedback in the process. They also bring up one of the bigger reasons to do this, “Designers may so easily fall in love with their own ideas and creations that they lose the ability to evaluate them critically.” Bringing in players helps to alleviate this problem, as they can critically analyze the work. Another thing that it brings up which can be a problem for using users to create the product is that for commercial products, the goal is to make the product easy to use, and simple. However with games, what makes it fun a lot of the time is solving puzzles and pushing the user. It can't get boring.



## **4. Experiment**

### **4.1 Recruitment Process**

To recruit people, an email was sent out to the CS students list. This email stated that both players and developers were needed and that both would receive some type of compensation for their participation. It was explicitly stated that the players would have a strong influence on the outcome of the game, that they would be able to shape it by their input, and that the game was being made specifically for them. The developers received independent study credit, which included working on the project as well as other work toward learning the Scrum process and Agile development process as a part of the independent study.

The documents used for applying for HRC approval and the emails sent out to recruit the players are included in Appendix A.

### **4.2 Experiment Process**

In this experiment, we recruited players and developers for an experiment with Agile game development and fun. The developers worked on programming and designing a game as a part of an independent study. The development team also included the Primary Investigator and met twice weekly for Scrum meetings to see if there were any impediments for the developers. Following the Scrum process the team released a new version of the game every two weeks for the players to test. At these release meetings the players played the game and provided feedback as to the fun factor of the game. This was recorded via a questionnaire that was handed out to each of the players at the release. A discussion was also recorded and held so that the players could also talk amongst themselves to give more feedback for both the developers, in making the game, and the investigator, to gather data as to how the process may help increase the quality of the game.

This continued over several months starting January 30, 2007, during which time the primary investigator observed how the development team worked with the process and how Scrum helped to alleviate potential problems as they occurred. This went on until April 17, when the developers were asked to complete a final survey about their experiences after the final release of the game.

### **4.3 Signature Qualities to Look for**

The main aspect of the development that is being examined is how the Scrum process affects the degree of fun in the game. This is tested by determining whether the players find the game more fun and desirable to play after each iteration. If the feedback that the players provided was able to be used by the developers in the next iteration, and the game became more fun because of this feedback, the ability of the players to improve the game demonstrates that the Agile process worked. If the final release of the game proved to be fun then Agile was working effectively. If the process wasn't working, then at the iterations the players wouldn't be giving feedback, or able to give feedback. Also, the game fun factor wouldn't be increasing with the iteration releases, and at the end the game would not be fun. Determining how the process affects fun is something that does not require feedback from the developers, but only from how the players respond and react to the product made by the developers.

The developers, on the other hand, were watched to see how Scrum affects the actual development side of production. While not being the primary focus of this study, the problems of complexity and

communication can also be examined to see if Scrum helps improve developer organization and capability. With both complexity and communication, the primary indicator that the process is working is that problems and impediments are brought to light and solved, especially early in the process. Along with this, if Scrum is working properly, the estimations should change to become more accurate as the project progresses.

There are various metrics that may or may not be useful in helping to determine how helpful the processes were in game development. These metrics were recorded and saved in the hope that they may be able to provide some insight into Scrum's usefulness in helping making games more fun. The metrics include:

Fun:

- Fun ratings at each iteration
- How happy people are with changes
- How active players are in playing and in the discussion
- Amount of suggestions from players
- How similar suggestions are from players

Complexity and Communication:(Recorded on a per iteration basis)

- Number of issues brought up at Scrums
- Lines of code
- Number of classes
- Number of functions per class
- Length of functions

## 5. Results

### 5.1 Game Development

This section discusses the effectiveness of the development process. It will go over observations of the development life cycle, and analyze how the Scrum process affected development.

The team started with four developers, plus the primary investigator as the Scrum master. During the first iteration, one of the developers left because of personal reasons, and the rest of the project was completed with the remaining three developers. The team met twice a week for an hour each time, and each developer on the team aimed to do six to nine hours of independent work on the game each week. All of the developers involved were students at the University of Colorado (Boulder) at the time of the study. The metrics in the code are in Appendix C, lists of accomplishments at each meeting in Appendix D and developer responses to the study in appendix E.

#### 5.1.1 Iteration One

In the first iteration, the primary focus was bringing the team up to speed. There was some trouble in the beginning, as all of the team members stated that they were unsure as to what to do. Several had never programmed in Java, and none of them had worked with Scrum before. During the first half of the iteration the team was brought up to speed in meetings and going over code. At this point, the team was still not able to self-organize, and things were moving slow as the team members were unsure what they could be doing. This brings out one of the weaknesses of Scrum – the team at this point felt that there was a lack of guidance going on in the early portion of the process. They were unsure of the requirements, as well as how to implement them. However, even with this, by making the Scrum meetings open to criticism this problem was overcome and more guidance was given in order to give the team what they needed in order to get work done.

Towards the end of the iteration, one of the developers had to leave the study. This did not greatly affect the team, however, because he did not take away much knowledge.

During the first iteration, the team also switched which graphics framework was going to be used. One of the developers had a lot of experience with graphics development, and found that the original idea of using jME would be too difficult. Based upon his advice the team switched to using JOGL, with minimal problems. In this instance, Scrum helped the developers, permitting the one who specialized in graphics to advise as to which framework to use for the best advantage of the project. The process of switching over involved a minimal amount of refactoring as the project was still small, and then the development continued as normal, with no one looking back, and happy with the new framework.

At the end of the first iteration, with a team of three developers, and one Scrum master, the team had a working start menu. Development went slower than originally estimated in the first iteration, probably due to the fact that the team was new to the project, process, and language and was still getting up to speed.

#### 5.1.1 Iteration Two

At the end of the second iteration the developers were still not much further than the first iteration. There was a lot of progress made on the front end of the program. The team was getting more comfortable with the tools, and progress was being made, however slowly. One of the developers had a good amount more experience with graphics and programming in general and was able to be more

productive. The other two were catching up, but still required help and instruction.

During this time, the team was more willing to ask questions, and admit to having problems understanding the tools, or not sure how to implement various parts. They were still able to work on the game, and did produce more code than the first iteration in this period.

### **5.1.3 Iteration Three**

By the third iteration the team was up to speed. They understand the direction of the project, as well as the tools and designs used for it. By this point they were also much more comfortable with asking questions about which directions to go in, as well as for direction with how to implement user stories. During the meetings there is a good amount of discussion between the developers. This shows some of the strengths of Scrum of being able to bring a team up to speed quickly. The team involved in this project worked approximately six to eight hours a week. By the end of the third iteration, it has been six weeks, so each developer has put in approximately thirty-six to forty-two hours. This is equivalent to one week of a developer working full time. During this iteration the team got the most new lines of code programmed, about twice as much as the previous iteration (see Appendix C). This is fairly significant in that a team of students got up to speed in what would have been one week for a professional development team, and after everyone was on track the team was able to work on new material at full capacity.

At the end of the third iteration, the game was playable, if slightly buggy. There were options for both player vs. computer and player vs. player battles, the mode determined by the player. The game had originally been focused exclusively on player vs. computer, but during discussion one of the players suggested implementing a player vs. player mode to be able to control the gameplay for the purpose of testing, as well as a good way to demonstrate new features. As far as gameplay, the players were able to set up their units on their respective sides of the game map with two available types of units. From this, when all human players have pressed a ready button, the game begins. In this release, there were two simple AIs available to each unit – attack nearest enemy and defend position.

### **5.1.4 Iteration Four**

At the end of iteration four, the game was playable and, based on feedback from the players, more information was presented on the screen. A log which stores all of the attacks and actions was added so that the player could examine previous turns. Unit death and damage animations were added to notify the players what is going on. This iteration was focused on revealing more about what was happening in the game to the players so that they could see what was happening at each step of play.

The development in this iteration went well with the team up to speed, and with most of the basic functionality in place it was easy to add new features. Having these short iterations helped to guide us to add these features, and as developer one stated, "any criticism, positive and negative, helped guide the development forward" (Appendix E). During this phase, there were still a good number of questions being asked, mostly in regard to how to work in to the current framework, which were able to be resolved quickly.

### **5.1.5 Iteration Five**

During iteration five, a lot of game content was added. It was able to be added fairly quickly, even with

a reasonable part of the team unable to contribute up to their full potential due to external classes. At this point the basic framework for functionality had been set. Work in this iteration was mostly tweaking existing features, as well as using the framework to add new content. More AI capability was included, new units were created, and special abilities were introduced. The AIs that were added allowed the units to attack the nearest enemy, defend their current space, assist the nearest ally, hold position as a sentry for the area, use a ranged attack, buff an ally's armor and defend their own hive. The final units in the game were the fire ant, killer bee, beetle, spider, mosquito, and the hive.

At this point in development, there weren't many problems that needed an extended time to figure out. It was mostly just implementing the content. Several of the AIs took only minutes to create with the framework that was already set up, as well as with new units. Some of the specials took a bit longer, but all in all, it was easy work to add new features. What made this easy was using good programming practices – objects were kept up to date with what they needed to have, refactoring them whenever a new change was made. For example, many functions were added and changed in the `InsectUnitImpl` class over this iteration. Instead of just using power for attacking as it did prior to these changes, the game now also executes a random special along with it.

### **5.1.6 Overall Conclusions**

This section will present the conclusions from the usage of the Scrum process in game development.

Over the course of the development process, the team adapted Scrum to work with it as best as possible. With this in mind, it is necessary to understand that not all teams are the same. The methods that the team involved with this project used won't work for all teams, and may not have been the optimal use of Scrum to begin with. However, what was adapted for the team turned out to be rather effective.

One way in which the team adapted the process was that instead of estimating each task and assigning it in the backlog, the team would just work on whatever they felt would be the next thing to do that the players wanted. The reason behind this, and the reason why it worked, was that the time that the team had together was minimum due to student schedules, and it was better spent discussing development as well as developing, and bringing forth any questions that came up. The lack of a backlog was compensated for by the fact that the player release meetings were so often compared to the time put in to production. There was enough time for some changes to the system to be made, but nothing major could be done. So, the players could give enough specific guidance to last for the next iteration.

This kind of adaptation and flexibility is what made Scrum so useful during development. The ability to change directions and not lock the development of the process on a previous decision has been a great asset of Scrum. Between each of the iterations, new directions were taken, albeit the number of changes got smaller as the game progressed and the team and players came to more of an agreement on what the game should look like. For example, in at the end of the third iteration, in terms of win conditions, there were many ideas, for the insects to control a certain area, collect resources, kill all opposing units. However, at the end of the fourth iteration, the players and developers came to a consensus to have the win condition be to destroy an enemy hive. This flexibility in features happens because of the just in time decision making of Scrum, meaning that when the basic concept of the game is being developed, the exact units, or AIs, or even mechanics are not looked at except superficially. These elements are only designed and implemented when they are needed in the game functionality. In this example, the win condition was not even considered until the third iteration.

Another aspect of Scrum that was helpful was the Scrum meetings. During these, it was easy to see the velocity of the project and make predictions of what will be done when, even without the burn down charts. These helped to bring forward obstacles quickly once the development team realized this. An example of a problem being solved with this was where one of the developers didn't understand the reason behind using an API with the implementation behind it. This was causing some issues for the front end, as it had access to methods it shouldn't. These functions were functions that would modify players points left, units controlled, and the sort that would effectively allow the player to cheat if they wanted to. This was quickly taken care of in a meeting where it was explained why it is done like that.

The need for developers to ask questions and be open, was a problem in the beginning of development; since the team didn't know what to do, and were afraid to ask question, the project got a slow start. Which brings forward one of the weak points of Scrum in that the team has be willing to work and put investment into it. The process doesn't use any strict deadlines, so it is up to the team itself to enforce those upon themselves.

Some things that Scrum did not directly effect:

-Code quality/developer quality: The code itself is not directly affected that much by the process. The code that developed can only be as good as the developers writing it. Scrum did not provide any mechanisms for improving code quality past the capability of the developers. However, it does allow for the developers to improve from guidance from more senior members with code reviews. Also, with Scrum being so flexible and able to adapt, it is always possible to refactor once problems with the code are noticed.

## **5.2 Fun Analysis**

This section will look at and analyze how effective the Scrum process was in terms of affecting fun. It will look at how the players reacted to the process, as well as to the game. The players responses can be found in Appendix B.

The player interviews and releases occurred every two weeks, with the first one being before any work was done. Also, in iteration four, the iteration lasted three weeks due to spring break and there would not be many people at the meeting, nor much work being done on the project by developers. The players were members of the CU Game Development club, most of whom are undergraduate students at CU, with a few that are not.

### **5.2.1 Before First Iteration**

During the first meeting the discussion was pretty simple, stating that the players wanted a game, and went fairly fast, lasting only a little bit more than 5 minutes. Another thing with this meeting, is the players didn't have much of an idea of what they wanted as a type of game. Most of the comments that were given were along the lines of just making a game. 13 of the 18 inputs from the first meeting are along those lines. The remaining 5 make a suggestion beyond just a game, like some ideas for a game. The one thing that the group was able to agree on during discussion was that they wanted an insect game.

From all of this, I believe that in order to get truly valuable feedback is that you need to go in with an idea. Then you can get ideas and thoughts from the players based on that, rather than try to let them come up with an idea, because with the difference of opinions within the group. As we presented more

ideas in later meetings, the players were able to provide input specific to the ideas.

### **5.2.2 After First Iteration**

At the second release meeting, things went much better. The players were really happy to see something, and the average rating of happiness with improvement was 7.8. Like in the first session, the main comment was wanting more. There were several comments in the discussion about idea's of what they want in the game, most of the discussion was about turn styles. It was in this that the idea for setting the AI was born. One of the players suggested assigning a "state" to each of your units, after several people spoke up stating that unit control can get really complicated. Also talk about various potential abilities that units could have, like a unit that could control other units, units that could move obstacles, etc etc. All of which from players in the discussion.

The comments were really helpful in developers designing the game, as the various bits that people wanted could be put together. At this point, the idea's were pretty vague, and nothing in terms of actual game play was really suggested. Rather it was more of "Be careful with the every unit moves every turn thing." or other similar comments, like in the discussion one player mentions with the kind of game play we were thinking about, it could take 10-15 minutes per turn. And in latter iterations it gets to more specifics about the game play.

### **5.2.3 After Second Iteration**

With the basics of the game set up in the previous two meetings, the players generally liked the direction of the game at this point. It wasn't entirely playable yet, but the players liked the concepts that the developers had come with, as well as the basics for setting up units. For example one of the typical responses from the players was "Good progress" (Player 7).

Nothing new in terms of content came out of this meeting, as the development side was just coming up to speed, and the players couldn't actually play the game. However, it was good encouragement for the developers to know that they were on track with what was going on (Player 3 even said "On the right track here"). The direction of production did not change from this meeting.

### **5.2.4 After Third Iteration**

After the third iteration, the big suggestion in the direction that the team take is improving the way information is given to the player. From as specific as "There needs to be some way to distinguish between the two sides. Also when something happens to one of the agents"(player 3) to as general as "more feedback on what's going on."(player 4). They also wanted more information regarding who was attacking, some of this was in animations, and some of this was in text. Using this feedback, we were able to change the priorities of what we were working on from making content, to more feedback for the players.

### **5.2.5 After Fourth Iteration**

It was during this iteration that most of the suggestions to change the actual game play occurred. The end goal of the game was changed from kill all enemy units to kill enemy hives "Kill the base, protect yours"(Player 6). More control over the units was asked for, as well as more unit types. Some people were just tired with the looks of the ants and bees, and some people suggested units like spiders. This

changed the primary focus of the development team back to working on the content of the game. There were some specific suggestions toward the content itself, one of the main ones being that the players didn't like how corpses stayed where they died and that units couldn't move through their spaces. This was one of the big topics of discussion, ending in the conclusion that they didn't like the corpses being permanent because it became possible to deadlock the game. So the players suggested decay, as well as flight and ranged attacks. Players also indicated that they wanted to see more special abilities.

### 5.2.6 After Final Iteration

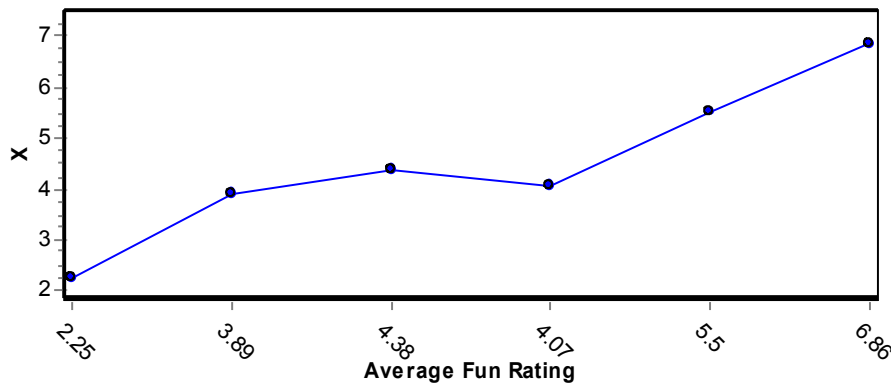
Unfortunately, due to time constraints at the end of the semester, development had to stop at this phase. However, even though the game was not able to get to a point where it could be considered finished, by this point it was really beginning to be a game. The players liked the direction that the game was headed, though they also agreed that there were certain things that were preventing it from being fun. One of the big problems was automation within the game, which made them feel that they set up their units, then let the game play for them. As Player 8 put it, they wanted "more skill required by the player." This is something that would be addressed in the next iteration, by giving the player more control over the units, being able to state where they could attack, or move a bit more directly. One aspect that players did like was the flexibility of the system, which made it possible to add units and enter game setup without recompiling the game itself. "Customization is good"(Player 3).

### 5.3 Conclusions

As we continued to use the Scrum process, I noticed that players were very good at providing input on concrete details. But when it came to more abstract concepts, they are unable to provide much useful feedback. As before the first iteration, the players provided little feedback about what they would like to see in a game. By this point in the development, it would probably be better to go to the players with a game idea, and get responses on what they think of it, as opposed to talking with them to come up with an idea.

Table a:

X Chart: Average Fun Rating





**Table b:**

**X Chart: Average happiness with changes**

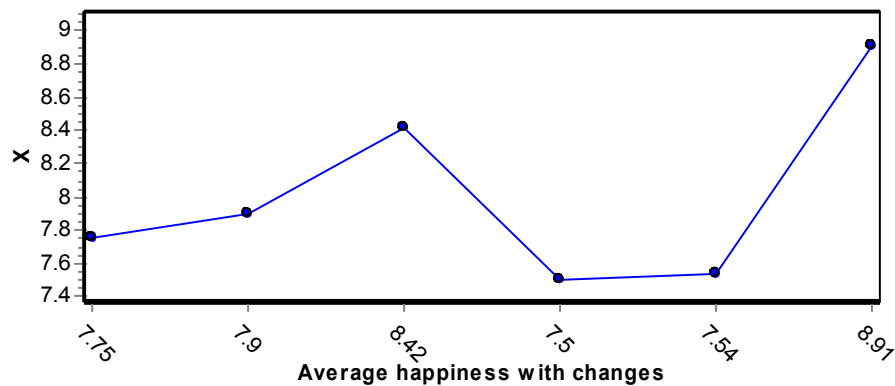


Table A shows that the fun throughout the production of the game was increasing. It was consistently going up, aside from after iteration 3. Also, how happy the players were with changes, generally tended to remain fairly high.

With the player ratings of the game, there is an almost constant increase in how fun they think that the game is. Aside from just after iteration 3, where the game became actually playable where the decrease is very small, the fun factor of the game is always going up. Also, with each of the iterations, the happiness with the changes stays around the same, I do not think that there is much correlation between how happy the players are with the changes and with the current fun factor to an extent. As long as the happiness with changes stays above a certain level, the fun factor will continue to go up, however how much it goes up does not depend on how happy they are with the changes. For example, some of the changes, like for after iteration four, the players didn't like the changes as much as they wanted to see however, because of the changes the game was more fun.

## 6. Experiment Conclusions

From the perspective of both the programmer and the player, the Scrum process has proven to be very effective and useful in the development process. This section will go over how this process was utilized to help game development from both the developer stand point, as well as fun.

### 6.1 Development Conclusion

Scrum was effective in helping developers overcome problems and work efficiently. All of the developers agreed that Scrum was able to effectively bring problems out into the open where they could be solved. If a developer found a bug, because communication was made so open, it was easy to let to the person who is capable of fixing it know about it. Having the communication paths open was essential for this to work. In order to have Scrum be effective, it is necessary to have the developers able to trust and work with each other, as a part of this process requires them being able to say that they don't understand something and ask for help in dealing with it.

One thing that I think would have helped, especially in the beginning, was a bit more initial planning. Having an initial architecture, that at least had the basic functionality explained, would have been really helpful for this team. During the first iteration, one of the biggest problems was the team not knowing what to do. Creating the basic APIs and setting up some basic design documents would have allowed to team to get a better idea what the program would look like earlier in the process. Also, I think that it would have been better for this team to keep better documentation throughout at least for one developer. The other two did not seem to have as much trouble with the minimal documentation. The documentation that did exist was focused around the Scrum meeting notes, the discussions between individual developers, and especially the source code and unit tests.

### 6.2 Fun Conclusion

Scrum does not create fun in a game by itself, but by using an initial idea from the design team it can be modified to better suit the players. The player meetings were very effective at allowing the players to provide helpful input to the developers at every iteration. It helped to keep the developers on track, or to change the direction of the game. With the iterations being only two weeks, this monitoring of the game was able to keep the developers working on what the players wanted. As one of the developers put it, "I didn't mind if the testers said they didn't like something that was added since the last play-test because at most I would have spent a week or so on that part, so it was not a huge waste of time if they didn't like it." The developers' ability to change the game based on the players input is what made this aspect of the experiment so successful.

The information from this experiment shows clearly, as the reading in [9] suggests, that the feedback of all parties involved, not just the players, is important to the process. In the release meetings, the players were able to give feedback as to what they didn't like about the existing game. However, when it came to new ideas and such, the players tended to be able to give feedback during a discussion, and provide some new ideas or concepts based on the new ideas from the development team. The design of the game needs to come from designers who have experience with creating games, with adjustments made to suit the players. Both parties are necessary to get a good idea. Programmers should also be involved in the discussion as, like designers, they are used to dealing with players.

Players are needed to make sure design stays on track and doesn't fail. They're also important to

helping the design team stay on track by using their feedback to focus the team on particular issues. Programmers are needed to provide their insight, as well as to create some more ownership for them, and more reason to program. Designers are needed to create base designs and ideas for a game from experience and then modify it based on the input of the entire group.

## 7. Future Work

This section will cover possible studies that could be done in the future based off of this research, or continuing this research.

### 7.1 Continued Study

I would like to continue this study in the future, but in a different setting. I would substitute a student team for a professional one. The students were not consistently able to meet, and had responsibilities to other classes that prevented them from focusing on this project. Especially in the last iteration, some of the developers were unable to do the amount of work they were supposed to due to exams and papers in their regular classes.

Along with this change of setting, it would be helpful to see the project to completion, as opposed to having to cut off development due to time constraints. This would allow a more accurate understanding of how quickly the team gets up to speed, as they would be together for the full time. Since they would always be together in the 'war room', it would be easier to see if that is helpful to them. Another reason to do this in an industry setting would be to look at the fun factor as to how that applies to large scale games. The main question to be answered would be, Does this process work in regards to fun for both large scale games as well as small?

### 7.2 Additional Studies

One interesting note in this was that many of the students involved said that they had learned more than they had in previous courses. With them two of them sophomores and one a junior, I think it may be effective to use Scrum, or game development in general, in teaching programming courses. It would be interesting to know how using this kind of project with students could potentially be used for recruitment and retention, as well as to provide students early opportunities to work on projects. Another aspect that would be interesting to look at with this would be to see how the group dynamic would come into play for a project like this one. It could potentially be a good experience for juniors and seniors to get lead development experience, as well as providing the freshmen and sophomores a colleague that they can learn from.

## 8. References

- [1] Game Development: Harder Than You Think  
Jonathan Blow,  
<http://www.acmqueue.org/modules.php?name=Content&pa=showpage&pid=114>
  
- [2] Requirements Engineering and the Creative Process in the Video Game Industry  
David Callele, Eric Neufeld, Kevin Schneider  
13th IEEE International Conference  
Volume , Issue , 29 Aug.-2 Sept. 2005 Page(s): 240 - 250
  
- [3] What Makes Things Fun to Learn?: Heuristics for Designing Instruction Computer Games  
Thomas W. Malone  
Proceedings of the 1982 conference on Human factors in computing systems  
Gaithersburg, Maryland, United States  
Pages: 63 - 68
  
- [4] Fun and Games: With a Multi-Language Approach  
Andrew M. Phelps, David M. Parks  
Volume 1 , Issue 10 (February 2004)  
Game Development: Serious Business, Serious Coding  
SPECIAL SECTION: Queue Focus: Game Development  
Pages: 46 - 56
  
- [5] Agile Manifesto  
[agilemanifesto.org](http://agilemanifesto.org)
  
- [6] Game Development Enters the Scrum  
<http://biz.gamedaily.com/industry/interview/?id=11418>
  
- [7] Agile Software Development  
Martin, Robert C.  
Agile Software Development: Principles, Patterns, and Practices. Prentice Hall, 2003.
  
- [8] Control Chaos  
[www.controlchaos.com](http://www.controlchaos.com)
  
- [9] Paper Burns: Game Design With Agile Methodologies  
[http://gamasutra.com/features/20060628/mcguire\\_01.shtml](http://gamasutra.com/features/20060628/mcguire_01.shtml)
  
- [10] Player-Centred Game Design: Experiences in Using Scenario Study to Inform Mobile Game Design  
[http://www.gamestudies.org/0501/ermi\\_mayra/](http://www.gamestudies.org/0501/ermi_mayra/)

# Appendices

## Appendix A: HRC Application

### Appendix: Forms and Letters part of Process

#### Recruitment Letter

Subject: Request for Subject Participation in Game Development Study

We are looking for participants to participate in a study of how Agile Development applied to game development affects fun. You are being asked to be in the study because you are a student in the Computer Science Department at the University of Colorado in Boulder. Participation in this study is entirely your choice. The study will be conducted over the 2007 Spring Semester.

If you agree to take part in this study, you will be asked to participate in one of two roles, as a Player or as a Developer. As a Developer you will be asked to participate and contribute to creating a game based on the Players input. At the end of the study you will be asked to fill out a survey as to your experience as a developer on this project. As a Player you will be asked to play the game every two weeks, complete a questionnaire and participate in a group discussion about what aspects of the game are fun, and what you would like to see changed.

Participating in this study as a developer you will be asked to put in what would be equivalent to a 3 credit hour course. You will have the option of receiving independent study course credit for your work. Participating in this study as a player should take about 60 minutes of your time every other week. We will ask you to come to a room in the Engineering Building that is available based on all participant's availability. Players will not be paid for this study. Players will not receive any class credit for participating in this study.

If you have any questions and/or are willing to participate in this study as a player or developer, please contact Andrea Gibson at [Andrea.Gibson@colorado.edu](mailto:Andrea.Gibson@colorado.edu).

#### Player Iteration Questionnaire

How fun do you think current game is(1-10):

Comments:

How happy are you with the changes since the last release(1-10):

Comments:

What would you like to see changed in the current game(Fun, game play, plot, control, etc etc):

What would you like to see added to the game:

#### Player Iteration Discussion Questions

1)What does everyone think about the current game? What do you think is fun? What do you think is not fun?

- 2)What would you like to be added into the game? What would like you to see changed in the game?
- 3)Anything else?

### Developer End Questionnaire

Did you feel that the process brought problems to knowledge quickly?  
What do you like and dislike about the meetings with the players?

### Player Consent Form

Agile Game Development and Fun  
Principal Investigator Andrea Gibson

### PARTICIPANT INFORMED CONSENT FORM 11-15-2006, Version 1

Please read the following material that explains this research study. Signing this form will indicate that you have been informed about the study and that you want to participate. We want you to understand what you are being asked to do and what risks and benefits—if any—are associated with the study. This should help you decide whether or not you want to participate in the study.

You are being asked to take part in a research project conducted by Andrea Gibson, an undergraduate student in the University of Colorado at Boulder's Department of Computer Science, 430 UCB, Boulder, CO 80309-0430. This project is being done under the direction of Professor Susanne Sherba, Department of Computer Science, 430 UCB. Andrea Gibson can be reached at (303)718-1972. Professor Susanne Sherba can be reached at (303)492-6380.

#### **Project Description:**

Within game development, fun is something that can be hard to test, and creating a fun game is hard. The Agile Development Process is a process based around releasing early and often to the customers, in our study that is the players. The purpose of this project is to study if the Agile Development Process can be used to make the final version of a game fun. You are being asked to be in this study because you are a student in the Computer Science Department at the University of Colorado in Boulder. It is entirely your choice whether or not to participate in this study.

#### **Procedures:**

If you agree to take part in this study, you will be asked to play a video game over its development of 12 weeks. You will be asked questions about your experiences playing the game, and whether or not various aspects of the game are fun, or not.

Participating should take 60 minutes every other week of your time. Participation will take place at an available room in the engineering center.

Participation in this research may include audio taping. These tapes will be used for analyzing the

changes in fun, and for the developers to use to continue to change the game and will be retained until the end of spring semester. Those individuals who will have access to these tapes will be the developers and researchers.

**Risks and Discomforts:**

There are no foreseeable risks associated with this study.

**Benefits:**

The benefits of being in this study are that the game created will be highly influenced by the players.

**Ending Your Participation:**

You have the right to withdraw your consent or stop participating at any time. You have the right to refuse to answer any question(s) or refuse to participate in any procedure for any reason. Refusing to participate in this study will not result in any penalty or loss of benefits to which you are otherwise entitled.

**Confidentiality:**

We will make every effort to maintain the privacy of your data. No names will be taken the questionnaires, or survey, nor will your name be put into the final report.

Other than the researchers and developers, only regulatory agencies such as the Office of Human Research Protections and the University of Colorado Human Research Committee may see your individual data as part of routine audits.

**Questions?**

If you have any questions regarding your participation in this research, you should ask the investigator before signing this form. If you should have questions or concerns during or after your participation, please contact Andrea Gibson at (303)718-1972.

If you have questions regarding your rights as a participant, any concerns regarding this project or any dissatisfaction with any aspect of this study, you may report them -- confidentially, if you wish -- to the Executive Secretary, Human Research Committee, 26 UCB, Regent Administrative Center 308, University of Colorado at Boulder, Boulder, CO 80309-0026, (303) 492-7401, Sheryl.Jensen@colorado.edu.

**Authorization:**

I have read this paper about the study or it was read to me. I know the possible risks and benefits. I know that being in this study is voluntary. I choose to be in this study. I know that I can withdraw at any time. I have received, on the date signed, a copy of this document containing 3 pages.

Name of Participant (printed) \_\_\_\_\_  
Signature of Participant \_\_\_\_\_ Date \_\_\_\_\_.



(Also initial all previous pages of the consent form.)

## Developer Consent Form

Agile Game Development and Fun  
Principal Investigator Andrea Gibson

### PARTICIPANT INFORMED CONSENT FORM 11-15-2006 Version 1

Please read the following material that explains this research study. Signing this form will indicate that you have been informed about the study and that you want to participate. We want you to understand what you are being asked to do and what risks and benefits—if any—are associated with the study. This should help you decide whether or not you want to participate in the study.

You are being asked to take part in a research project conducted by Andrea Gibson, an undergraduate student in the University of Colorado at Boulder's Department of Computer Science, 430 UCB, Boulder, CO 80309-0430. This project is being done under the direction of Professor Susanne Sherba, Department of Computer Science, 430 UCB. Andrea Gibson can be reached at (303)718-1972. Professor Susanne Sherba can be reached at (303)492-6380.

#### **Project Description:**

Within game development, fun is something that can be hard to test, and creating a fun game is hard. The Agile Development Process is a process based around releasing early and often to the customers, in our study that is the players. The purpose of this project is to study if the Agile Development Process can be used to make the final version of a game fun. You are being asked to be in this study because you are a student in the Computer Science Department at the University of Colorado in Boulder. It is entirely your choice whether or not to participate in this study.

#### **Procedures:**

If you agree to take part in this study, you will be asked to develop a game based on players input using the agile development process.

Participating should take the equivalent of a 3 credit hour course of your time. Participation will take place at an available conference or class room in the engineering center. You will be observed and notes and audio taping may take place during the production of the video game. You will be asked to meet with the Primary Investigator twice a week to discuss progress, as well as to plan out future work to be done on the game. You will also be asked to continue working on the game outside of these meetings as to what would be appropriate to a 3 credit hour independent study.

At the end of the study you will be asked questions about your experience with using the agile process.

Participation in this research may include audio taping. These tapes will be used for writing the final report and will be retained until the end of spring semester.

**Risks and Discomforts:**

There are no foreseeable risks associated with this study.

**Benefits:**

The benefits of being in this study are that you will receive independent study course credit for your work.

**Subject Payment:**

You will receive 2-3 for your participation in this study. If you chose to withdraw, an alternate means for completing the independent study.

**Ending Your Participation:**

You have the right to withdraw your consent or stop participating at any time. You have the right to refuse to answer any question(s) or refuse to participate in any procedure for any reason. Refusing to participate in this study will not result in any penalty or loss of benefits to which you are otherwise entitled.

**Confidentiality:**

We will make every effort to maintain the privacy of your data. Your privacy will be protect as much as possible with this study. Your names will not be used in the final report, and the audio recording will be erased at the end of the study.

Other than the researchers, only regulatory agencies such as the Office of Human Research Protections and the University of Colorado Human Research Committee may see your individual data as part of routine audits.

**Questions?**

If you have any questions regarding your participation in this research, you should ask the investigator before signing this form. If you should have questions or concerns during or after your participation,

please contact Andrea Gibson at (303)718-1972.

If you have questions regarding your rights as a participant, any concerns regarding this project or any dissatisfaction with any aspect of this study, you may report them -- confidentially, if you wish -- to the Executive Secretary, Human Research Committee, 26 UCB, Regent Administrative Center 308, University of Colorado at Boulder, Boulder, CO 80309-0026, (303) 492-7401, Sheryl.Jensen@colorado.edu.

**Authorization:**

I have read this paper about the study or it was read to me. I know the possible risks and benefits. I know that being in this study is voluntary. I choose to be in this study. I know that I can withdraw at any time. I have received, on the date signed, a copy of this document containing 3 pages.

Name of Participant (printed) \_\_\_\_\_

Signature of Participant \_\_\_\_\_ Date \_\_\_\_\_.  
(Also initial all previous pages of the consent form.)

## Appendix B: Player Responses

Base questionnaire

How fun do you think the current game is(1, worst-10,best):

Comments:

How happy are you with the changes since the last release(1, worst-10,best):

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

What would you like to see added to the game:

### January 30th, 2007

1) 18 M

How fun do you think the current game is(1, worst-10,best):

Comments:

How happy are you with the changes since the last release(1, worst-10,best):

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

What would you like to see added to the game:

Story, Story bending, Animation type graphics(like a comic book), Guns, Ninjas, Pirates, Cowboys

2) 18 F

How fun do you think the current game is(1, worst-10,best):

Comments:

How happy are you with the changes since the last release(1, worst-10,best):

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

What would you like to see added to the game:

Good user interface, graphics, plot, swords, ninjas, zombies, unicorns, good scenery(waterfalls, mountains, clouds, etc), problem solving, challenges... like Zelda/RPGs, MMOs pirates, stuff that can be cut up with ones swords (flowers, pots, furniture), fly, strategy)

3) 24 F

How fun do you think the current game is(1, worst-10,best): n/a

Comments:

How happy are you with the changes since the last release(1, worst-10,best): n/a

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

n/a

What would you like to see added to the game:

Pirates, Samurai or combination of the two(in insect form!)

An RTS(2D0 would probably be really fun.

4) 19 M

How fun do you think the current game is(1, worst-10,best): negative infinity

Comments: Needs giant robots

How happy are you with the changes since the last release(1, worst-10,best):

Comments: Disappointed, Too much paper, not enough game.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Everything

What would you like to see added to the game:

game play

Bug based civilization:

RPG Elements:

Control character with stars move through your "city"

multiple interactive plot lines.

Other characters

RTS/Sim city elements:

Build up your city from a birds eye view improving aspects.

Civ:

Diplomacy with other cities (possible on line multi player/MMO/spore implementation)

3rd/1<sup>st</sup> person shooter: [Or tactile RPG]

Front line combat against other cities when you go to war with them.

Also: try to work in exoskeleton giant robot upgrading system.

5) 28 M

How fun do you think the current game is(1, worst-10,best): 1

Comments: Not so much

How happy are you with the changes since the last release(1, worst-10,best): -

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

More cowbell, please

What would you like to see added to the game:

Something shiny

6) 25 M

How fun do you think the current game is(1, worst-10,best): 1

Comments: Huh?

How happy are you with the changes since the last release(1, worst-10,best): 1

Comments: Not happy.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

More interactivity

What would you like to see added to the game:

More of everything

7) 19 M

How fun do you think the current game is(1, worst-10,best): 1

Comments: Kinda.. non interactive

How happy are you with the changes since the last release(1, worst-10,best): N/A

Comments: N/A

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

I would like see a little more substance

What would you like to see added to the game:

I'm really not sure. I'd like to see something actiony.

8)

How fun do you think the current game is(1, worst-10,best): 1

Comments: Needs more graphics

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: N/A

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

N/A

What would you like to see added to the game:

Giant robots

9) 25 F

How fun do you think the current game is(1, worst-10,best): 1

Comments:

How happy are you with the changes since the last release(1, worst-10,best): N/A

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

More of all

What would you like to see added to the game:

Something cool. Candy land.

10) 23 M

How fun do you think the current game is(1, worst-10,best): 3

Comments: I like nothing (nihilist.)

How happy are you with the changes since the last release(1, worst-10,best): No changes

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Fun, gameplay, whats the difference.

What would you like to see added to the game:

A rich environment for exploration, interaction and play. Lollipops!!

11) 38 M

How fun do you think the current game is(1, worst-10,best): 1

Comments: No Game yet

How happy are you with the changes since the last release(1, worst-10,best): N/A

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Some kinda game, rather than no game.

What would you like to see added to the game:

Fun, gameplay, plot, control, etc.

12) 18 M

How fun do you think the current game is(1, worst-10,best): 1

Comments: Need a game

How happy are you with the changes since the last release(1, worst-10,best): ?

Comments: Need more game.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

More game!

What would you like to see added to the game:  
More!

13) 22 M

How fun do you think the current game is(1, worst-10,best): N/A

Comments:

How happy are you with the changes since the last release(1, worst-10,best): N/A

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Game play – add something

What would you like to see added to the game:

Arcade game

14) 20 M

How fun do you think the current game is(1, worst-10,best): \

Comments:

How happy are you with the changes since the last release(1, worst-10,best): N/A

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc): \

What would you like to see added to the game:

I think it would be very interesting if you could make a cooperative game.

15) 22 M

How fun do you think the current game is(1, worst-10,best): 1

Comments: Non existent.

How happy are you with the changes since the last release(1, worst-10,best):

Comments: Error self destruct in 10 seconds

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Yes it needs those things. (Arrow pointing to the (Fun, gameplay, plot, control, etc etc))

What would you like to see added to the game:

I would like to know more about the project before adding more features to this already bloated software.

16) 37 M

How fun do you think the current game is(1, worst-10,best): 1

Comments: Game doesn't actually exist yet.

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: See above

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Having something that actually exists.

What would you like to see added to the game:

Exploding ninja penguins. Escalating power levels.

17) 20 F

How fun do you think the current game is(1, worst-10,best): 5

Comments: It's a little low on content, but it doesn't have any bugs. :)

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: Awesome!

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):  
More game.  
What would you like to see added to the game:  
I like role playing games! And costumes!

18) 21 F

How fun do you think the current game is(1, worst-10,best): 10

Comments: zomgwtfbfq chicken salad best game EVAH!

How happy are you with the changes since the last release(1, worst-10,best): 6.25124

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):  
No way its awesome

What would you like to see added to the game:  
a game perhaps but not necessary

## February 13<sup>th</sup> 2007

01) 25 F

How fun do you think the current game is(1, worst-10,best): 7

Comments: Cool ideas

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: Awesome progress + no I'm not being sarcastic. Engine, concept + code.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):  
Basic characters, like 2 kinds.

What would you like to see added to the game:

02) 23 M

How fun do you think the current game is(1, worst-10,best): 1

Comments:

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: Something is better than nothing

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):  
We need some gameplay

What would you like to see added to the game:

3) 38 M

How fun do you think the current game is(1, worst-10,best): 1

Comments:

How happy are you with the changes since the last release(1, worst-10,best): 5

Comments: Something > nothing

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):  
add stuff to do

What would you like to see added to the game:  
more stuff!

-control insect behavior state instead of specific actions?

-brain controlling fungus



4) 18 F

How fun do you think the current game is(1, worst-10,best): 2

Comments: Could be better :)

How happy are you with the changes since the last release(1, worst-10,best):

Comments: First time I've seen it.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Insects are cool.

Still vote for the randomized bird attacks

What would you like to see added to the game:

Sparkles and gameplay

5) 24 M

How fun do you think the current game is(1, worst-10,best): 1

Comments: Lacks fun

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: Progress Made

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

More plot and gameplay.

What would you like to see added to the game:

More obscure insects.

Surprise me with fucked up aspects of the insect world.

6) 22 M

How fun do you think the current game is(1, worst-10,best): 2

Comments: It has a menu which is more than I can say for many of my games.

How happy are you with the changes since the last release(1, worst-10,best): 4

Comments: Even though it wasn't much it is very nice to see progress.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Its fine as it is

What would you like to see added to the game:

The rest of it.

Be careful with the every unit moves every turn thing. That can end up making very long turns. See Civ for prior art.

7) 18 M

How fun do you think the current game is(1, worst-10,best): 10

Comments: wwwwoooowww a blue screen with start game

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: Lots of change... we have a start screen!

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

I'll answer when we get to play

What would you like to see added to the game:

Gameplay: Dung beetles with a straight shot downhill

Queen Insect to create armies

bomber ants

I forgot what else I said.

8) 21 F

How fun do you think the current game is(1, worst-10,best): infinity

Comments: Absolutely amazing as usual

How happy are you with the changes since the last release(1, worst-10,best): 3.14159

Comments: There is an amazingly awesome start screen of doom!!!

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

More ninja and pirate bugs

What would you like to see added to the game:

A game would be nice

sparklies

9) 18 M

How fun do you think the current game is(1, worst-10,best): 8

Comments: I like the idea

How happy are you with the changes since the last release(1, worst-10,best):10

Comments: Anything is better than nothing

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Need a cooler start menu (pictures of insects)

What would you like to see added to the game:

-Make lots of obscure insects/lots of unit choices

-Maybe make everybody set their AI and then have all of the units move at once

10) 23 M

How fun do you think the current game is(1, worst-10,best): 3

Comments: Good premise, needs polish, more content

How happy are you with the changes since the last release(1, worst-10,best): 9

Comments: I was a little hopeful for more

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Display different player's units in different colors, show fights between individual units, flesh out spectral attacks for units, add more types of units, bigger text terrain elements, unique interaction between unit types (flowers heal bees, ants can burrow into dirt), more types of orders, maybe direct control

What would you like to see added to the game:

## **February 27<sup>th</sup>, 2007**

1) 20 M

How fun do you think the current game is(1, worst-10,best): 5

Comments: Very reminiscent of Heroes of Might and Magic. I'd like to see its next incarnation.

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: I can see what the game will be vaguely like, which is good.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Make it playable – first priority.

What would you like to see added to the game:

An element of hidden information might be interesting.

2) 22 F

How fun do you think the current game is(1, worst-10,best):

Comments: ... It doesn't work

How happy are you with the changes since the last release(1, worst-10,best): 5

Comments: B/c it has bugs.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

What's the point of the game? Kill everything? or... what?

What would you like to see added to the game:

Animations

3) 25 M

How fun do you think the current game is(1, worst-10,best): 1

Comments: Looking better though.

How happy are you with the changes since the last release(1, worst-10,best): 9

Comments: On the right track here.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

A fleshed out prototype, with game mechanics, controls and a victory condition.

What would you like to see added to the game:

See above.

4) 25 F

How fun do you think the current game is(1, worst-10,best): 6

Comments: Cool characters. Nice map

How happy are you with the changes since the last release(1, worst-10,best): 6

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Gameplay

What would you like to see added to the game:

A short round, you can play against another human.

5) 21 F

How fun do you think the current game is(1, worst-10,best): 8

Comments: There is gameplay now, woot.

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: See above

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Maybe have more back story/objective or terrain changes

What would you like to see added to the game:

Evil muffins with plots for world domination.

6) 18 M

How fun do you think the current game is(1, worst-10,best): 8

Comments: Needs a storyline, eventually

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: The Tick!! SPOOOONNN!!!!

I was expecting different graphics for the insects (Love the tick)

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Possibly the bitmaps for the insects, eventually, keep making the main game

What would you like to see added to the game:

Special abilities unique to each insect

Range: Stinkbugs, grasshoppers

7) 19 M

How fun do you think the current game is(1, worst-10,best): 1

Comments: Still not playable, n/a

How happy are you with the changes since the last release(1, worst-10,best): 7

Comments: Good progress

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Getting it to a playable state would be A1-super right now

What would you like to see added to the game:

Special abilities for bugs would be neat.

8) 37 M

How fun do you think the current game is(1, worst-10,best): 3

Comments: Doesn't DO anything yet

How happy are you with the changes since the last release(1, worst-10,best): 7

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Keep the Tick!

What would you like to see added to the game:

Make the goal fighting over food?

9) 18 M

How fun do you think the current game is(1, worst-10,best): 3

Comments:

How happy are you with the changes since the last release(1, worst-10,best): 7

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Alternative victory conditions

What would you like to see added to the game:

Special abilities

Subclasses of races ( ant: worker, warrior, queen)

game play

Structures ( webs, nests, colonies, large pieces of food)

10) 23 M

How fun do you think the current game is(1, worst-10,best): 1

Comments: Not playable yet

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: Anything is better than just a menu, I like the unit pictures

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Get it playable, more unit types/subclasses

What would you like to see added to the game:

Gameplay, sound effects, music, more crazy not really applicable insect associated things like the tick

11) 18 M

How fun do you think the current game is(1, worst-10,best): 8

Comments: Reminiscent of a bug's life and advance wars.

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: More stuff is always güd.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):  
A bit more variety in units, and special abilities  
e.g. Spiders web, poison black widows, playing maims cripples bug) (Reduced movement)  
What would you like to see added to the game:  
Unit reproduction!  
(Pause 1 or two turns, produce new unit.)

12) 22 M

How fun do you think the current game is(1, worst-10,best): 3

Comments:

How happy are you with the changes since the last release(1, worst-10,best):

Comments: Didn't see the last release

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Gameplay – add some

What would you like to see added to the game:

13) 30 M

How fun do you think the current game is(1, worst-10,best): 5

Comments: Have you looked at the old DOS game Battle Bugs? Your game reminds me of it.

How happy are you with the changes since the last release(1, worst-10,best): N/A

Comments: Didn't see prev version

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

RTS beyond just survival(motive/plot), how/why these different insects are banded together

What would you like to see added to the game:

Gruesome insect death scenes :)

Diplomacy

14) 20 F

How fun do you think the current game is(1, worst-10,best): 5

Comments: You can't do anything, but it looks cool. UI is good.

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: Yay stuff!

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Gameplay, balance prices

What would you like to see added to the game:

Gameplay

-special abilities for insects

-move/defend/attack/ability

-base on interactions

**March 13<sup>th</sup>, 2007**

1) 20 M

How fun do you think the current game is(1, worst-10,best): 2.5

Comments: It doesn't seem too interactive in terms of gameplay, the game looks like it plays itself while the players watch

How happy are you with the changes since the last release(1, worst-10,best): 5

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):  
Much more feedback to the players, bigger map maybe lower the turns between orders a bit.

What would you like to see added to the game:

easier way to inspect multiple characters at once

batch orders? Telling 10 ants to attack at once looks tedious, a way to group units and give commands would alleviate some of that.

2) 18 M

How fun do you think the current game is(1, worst-10,best): 8

Comments: Its very basic but potentially addicting

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: So far, gameplay looks clean

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Control: Right-click menus to change stances/AI behavior

Horde bonuses? ie. 2+ units attacking 1 gets attack or defense bonus?

Lower # of steps per turn

What would you like to see added to the game:

Colors to differentiate between friendly + hostile units (colored hexes?)

Ability to attack a specific unit!

More variety in terrain, ie ant hills (def bonus for ants), spider webs (anything but spiders immobilized!), beehives (double attack and defense for bees?)

3) 31 M

How fun do you think the current game is(1, worst-10,best):

Comments: Hard to say how fun it is in its current form. I like the idea of choosing different Ais and pitting them against each other.

How happy are you with the changes since the last release(1, worst-10,best):

Comments: This is the first release I've seen.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

There needs to be some way to distinguish between the two sides. Also when something happens to one of the agents (e.g. When its stats change) there should be some visual indication, such as the sprite changing color.

What would you like to see added to the game:

More feedback to the user in general.

4) 25 M

How fun do you think the current game is(1, worst-10,best): 3

Comments: I can see it coming together.

How happy are you with the changes since the last release(1, worst-10,best): 8

Comments: Very nice changes.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Better AI. Complete gameplay. Move units.

What would you like to see added to the game:

See above.

Plus more feedback on what's going on. ie. Attack animations or something.

5) 22 M

How fun do you think the current game is(1, worst-10,best): 3

Comments: Its an interesting concept.

How happy are you with the changes since the last release(1, worst-10,best): 7

Comments: Great Progress! It's playable!

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

I'd like to see more unit types and more strategies. I feel like I do not have enough control – there are not enough degrees of freedom – to build an effective strategy.

What would you like to see added to the game:

Allow orders with specifications; “general” moves, rather than square moves, for example. I like the gameplay model; 5 turns, give orders, 5 turns, etc.

6) 20 M

How fun do you think the current game is(1, worst-10,best): 2

Comments: Looks promising, but needs more functionality

How happy are you with the changes since the last release(1, worst-10,best): 7

Comments: Stability is more important than features, stabilize the game before you innovate further.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

-more info for the player

-graphical representation of hit point on each unit

-more AI options

-strike acknowledgments(animations at best, scrolling text at worst)

What would you like to see added to the game:

-The ability to select groups of units and order them collectively.

7) 20 F

How fun do you think the current game is(1, worst-10,best): 5

Comments: Reasonably fun, still incomplete

How happy are you with the changes since the last release(1, worst-10,best): 8

Comments: Yay! It works! You can play!

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Control & feedback

Gameplay

What would you like to see added to the game:

Distinguish between players

Let them know when automated steps are done

bigger font

more AI options

way of knowing AI style/damage without clicking (more feedback)

8) 33 M

How fun do you think the current game is(1, worst-10,best): 5

Comments: The basics are great. I think this will be a great game.

How happy are you with the changes since the last release(1, worst-10,best):

Comments: I didn't see the last release.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Distinguish between sides(change colors, number units), be able to select units on one side

What would you like to see added to the game:

A defensible position(ie, hive, flower patch)

**April 3<sup>rd</sup>, 2007**

1) 23 M

How fun do you think the current game is(1, worst-10,best): 3

Comments: Lacks Depths

How happy are you with the changes since the last release(1, worst-10,best): 7

Comments: Changes seem mostly cosmetic and ui-functional

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

More gameplay depth

What would you like to see added to the game:

Choices don't seem meaningful, so either expand the significance of the choices available or add new, interesting choices

2)

How fun do you think the current game is(1, worst-10,best): 8

Comments: Lots

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: Yay!

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Better ai or more controlling

graphics: bee wings = odd, ants look weird one directly above

What would you like to see added to the game:

corpses

attack animations

some sort of animation that makes moment visual trackable, teleporting hex to hex hurts eyes

3) 21 F

How fun do you think the current game is(1, worst-10,best): 8

Comments: Really cool and good animation but a little too random

How happy are you with the changes since the last release(1, worst-10,best): 7

Comments: A lot more gameplay now but didn't there used to be more kinds of bugs? Like the tick

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

more interactivity and more type of insects (the tick!!!!)

What would you like to see added to the game:

The tick!!!or if you can't violate those annoying copyright laws, maybe a nice dragonfly or something and special powers. Woot

4) 25 M

How fun do you think the current game is(1, worst-10,best): 3

Comments: Looking better, but no where near finished

How happy are you with the changes since the last release(1, worst-10,best): 4

Comments: Some improvements but still more work is needed

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

more depth, fix AI, flesh out bees and ants

What would you like to see added to the game:

more ants, but fixing the game as is is more important.



5) 33 M

How fun do you think the current game is(1, worst-10,best): 6

Comments: Maybe more unit types? Spiders?

How happy are you with the changes since the last release(1, worst-10,best): 8

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Control - being able to send units to specific hexes

What would you like to see added to the game:

A ranged attack of some kind

6) 25 F

How fun do you think the current game is(1, worst-10,best): 5-6

Comments:

How happy are you with the changes since the last release(1, worst-10,best): 7

Comments: 2-players, AI, its getting better

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

What would you like to see added to the game:

-special powers, bees fly over corpses

-target a specific enemy to attack

\*Kill the base, protect yours

7) 18 M

How fun do you think the current game is(1, worst-10,best): 8.5

Comments: eez gud

How happy are you with the changes since the last release(1, worst-10,best): 9

Comments: eez gud

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Needs a couple more unit types + a little balancing

What would you like to see added to the game:

Ability to command units to attack selectively.

8) 20 M

How fun do you think the current game is(1, worst-10,best): 6

Comments: Usable interface combat log helps – might be useful to label tiles on the board to help in finding units listed on the board

How happy are you with the changes since the last release(1, worst-10,best): 8.5

Comments: Much easier to see what is going on, easier to issue orders, fun animations, could color X's to see how many units of the other team you've killed

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Perhaps add more strategy for the user, more AI options – perhaps something to target a location

Bigger map to avoid line of corpses with five units

What would you like to see added to the game:

Special powers, more distinction among units

(Phase 5) storyline

9) 37 M

How fun do you think the current game is(1, worst-10,best): 5

Comments: The “impassable corpses” are a problem. More control over your bugs would be good.

How happy are you with the changes since the last release(1, worst-10,best): 6

Comments: More stuff going on is good.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Lose the impassable corpses -

More control over bugs – tag enemies to kill? Places to go?

What would you like to see added to the game:

Goals beside 'kill the other side'?

Movement rate difference for different bug types. Big maps?

“Nuke” equivalent - “Kid with magnifying glass”

10) 38 M

How fun do you think the current game is(1, worst-10,best): 5

Comments: Don't know how it plays, haven't played it

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: Lots of progress

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

bigger map – swarms of insects!

What would you like to see added to the game:

-Easy selecting of lots of insects,

-Maybe keep selection persistent across turns. Then you can change strategy on selected group.

-Make multiple groups, can toggle each groups strategy. Like make an attack group, toggle to flee if trouble.

11) 31 M

How fun do you think the current game is(1, worst-10,best): 5

Comments: Ok, I never was a fan of RTS's

How happy are you with the changes since the last release(1, worst-10,best): 7

Comments: Last time I saw it there were no units that moved on the screen.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

1 Plot. Doesn't seem to be any.

2 Fix bugs

What would you like to see added to the game:

Units. If there aren't more than ant + bees. Add more units

12) 23 M

How fun do you think the current game is(1, worst-10,best): 3

Comments:

How happy are you with the changes since the last release(1, worst-10,best): 7

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

More fun (as for everything)

Control – seems like game is fairly automated, players should be more involved or more strategy should be needed between rounds.

What would you like to see added to the game:

**April 17<sup>th</sup>, 2007**

1) 23 M

How fun do you think the current game is(1, worst-10,best): 8

Comments: Numerous units and Unit Differentiation

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: Solidified game features, added interesting content

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Better feedback of in-game evens like unit interaction, maybe a little less automation

What would you like to see added to the game:

Terrain features like water and rocks, a little bit more customizability for game modes, with in-game options.

2) 33 M

How fun do you think the current game is(1, worst-10,best): 8

Comments:

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: I love the new units

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

More detail on battles

What would you like to see added to the game:

More choices on Ai

3) 18 M

How fun do you think the current game is(1, worst-10,best): 9

Comments: Customization is good

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments: Eez cul

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Needs a little more micro-manageability

What would you like to see added to the game:

Damage animations! (Attacker + defender a la pokemon)

4) 20 M

How fun do you think the current game is(1, worst-10,best): 7

Comments: Looks like a game now, lacking in visual feedback, flexibility is nice

How happy are you with the changes since the last release(1, worst-10,best): 8

Comments: Actually seems playable, lot less buggy

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

More visual feedback, like showing ranged attacks graphically, stronger AI

What would you like to see added to the game:

Perhaps some pre-determined challenge match ups against the computer

5) 22 M

How fun do you think the current game is(1, worst-10,best): 5

Comments:

How happy are you with the changes since the last release(1, worst-10,best): 5

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):  
More user interaction – don't just have strategy in beginning and player just watches for the rest.  
What would you like to see added to the game:  
A bit more gameplay related graphics – who attacked who. Better AI

6) M

How fun do you think the current game is(1, worst-10,best): 8  
Comments: I like the game more  
How happy are you with the changes since the last release(1, worst-10,best): 10  
Comments: I really like the new units and AI's  
What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):  
More options to give more options.  
What would you like to see added to the game:  
More battlefield obstacles.

7) 25 M

How fun do you think the current game is(1, worst-10,best): 5  
Comments: Looking Good  
How happy are you with the changes since the last release(1, worst-10,best): 7  
Comments: Very good. Its looking like a game  
What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):  
Polish, better AI  
What would you like to see added to the game:  
See above

8) 19 M

How fun do you think the current game is(1, worst-10,best): 7  
Comments: Not quite my cup of tea as far as games go, but rather interesting  
How happy are you with the changes since the last release(1, worst-10,best): 10  
Comments: Last release I saw was very basic, this is a huge leap from that  
What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):  
More skill required by the player. Board is rather small.  
What would you like to see added to the game:  
A “move to” order A larger board. Require more maneuvering. More animation to show what is going on.

9) 18 M

How fun do you think the current game is(1, worst-10,best): 8  
Comments: Terrains, possibly more animations  
How happy are you with the changes since the last release(1, worst-10,best): 10  
Comments: I wasn't here for the last release ;\_  
What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):  
More options as to what a unit does (where to move or who to attack)  
Possibly an option to change management option  
What would you like to see added to the game:  
More shiny or sparkles :P

10) 22 M

How fun do you think the current game is(1, worst-10,best): 3-6

Comments: Maybe. Haven't played it. It would be nice to see some deep strategy. I have yet to see a game and think "good move."

How happy are you with the changes since the last release(1, worst-10,best): 8

Comments: Lots of progress, Hives + bigger board is a big improvement.

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Less randomness

What would you like to see added to the game:

11) 20 M

How fun do you think the current game is(1, worst-10,best): 6

Comments: Huge improvement

How happy are you with the changes since the last release(1, worst-10,best): 10

Comments:

What would you like to see changed in the current game(Fun, gameplay, plot, control, etc etc):

Better AI.

Better representation of combat

What would you like to see added to the game:

Resources?

## **Appendix C: Developer Metrics**

### **After iteration 1, Feb 13<sup>th</sup>,2007**

Lines of code: 841

Number of classes: 30: 26 classes + 4 interfaces

Number of functions per class: Average number of methods of 3.13

Length of functions: Average function length of 5.85

### **After Iteration 2, Feb 27<sup>th</sup>, 2007**

Lines of code: 1997

Number of classes: 53: 43 classes + 10 interfaces

Number of functions per class: Average number of methods of 4.28

Length of functions: Average function length of 6.17

### **After Iteration 3, March 13<sup>th</sup>,2007**

Lines of code: 4209

Number of classes: 99: 83 classes + 16 interfaces

Number of functions per class: Average number of methods of 4.46

Length of functions: Average function length of 6.94

### **After iteration 4, April 3<sup>rd</sup>, 2007**

Lines of code: 4715

Number of classes: 105: 87 classes + 18 interfaces

Number of functions per class: Average number of methods of 4.59

Length of functions: Average function length of 7.17

### **After iteration 5, April 17<sup>th</sup>, 2007**

Lines of code: 6495

Number of classes: 135: 117 classes + 18 interfaces

Number of functions per class: Average number of methods of 4.43

Length of functions: Average function length of 7.73

## Appendix D: Developer Iteration Scrum meeting responses

### Iteration One: Didn't start jotting down developer responses till iteration two

#### Iteration Two:

Monday Feb 19<sup>th</sup>

Have done:

developer one: Added functionality to quad. Implemented 3d board. Fixed texture bugs. Reworked vector2f and 3f.

developer three: Trying to learn OpenGL working on setup screen.

developer two: Got 2 tests to work. Added testcase(Not entirely working)

Questions:

developer one: Doing well

developer three: None this week

developer two: None

For next time:

developer one: Get 3d board done. Setup screen. Place units onto board.

developer three: Work on OpenGL Movings units around setup menu.

developer two: Design multiple units, 10ish. Read units config from a file. Make it not hardcoded.

Thursday 22nd.

Done:

developer one: Added Hexes, scaling for buttons for units. Zoom for board.

developer three: Worked with getting text display for setup window. Some setup window stuff.

developer two: Finished XML parser. Changed units to return info for the units.

Problems:

developer one: Good

developer three: Doing good.

developer two: Good

Next Time:

developer one: Mouse input to move units onto the board. As well as board to board.

developer three:

developer two: In unit setup, set area for player to be able to put stuff, and verify placing in that area.

Monday Feb 26<sup>th</sup>

What have you done:

developer one: Hexes scale properly with map. Started work on picking Hexes

developer three: Setup stats bar. Points available options and exit.

developer two: implemented different game types.

Problems:

developer one: None  
developer three: None  
developer two: None

For next time:

developer one: Picking hexes  
developer three: Starting game, selecting units AI  
developer two: Refactor game to include players. Computer unit set up.

### **Iteration Three:**

Mar 1st, 2007

What have you done:

developer one: Researching mouse conversion  
developer three: Options  
developer two: Work on game player stuff

Questions:

developer one: Mouse 3d stuff.  
developer three: None  
developer two: Trouble understanding pattern, understand now though.

For next time:

developer one: 2d board, and mouse recognition  
developer three:  
developer two: Finish up game player stuff.

March 5th, 2007

Done:

developer one: Converted board to 2d, knows where mouse is and can place units onto the board.  
developer three: Working on implemented ai window.  
developer two: Refactored game players.

Problems:

developer one: None  
developer three: None  
developer two: None

To Do:

developer one: Implement class keeps track of which player is currently doing stuff. Move units on the board.  
developer three: Transition to start playing.  
developer two: Refactor XML



March 8th, 2007

Done:

developer one: Redesigned game board singleton, for diff colors. Move units, switch units. Fixed up playing window

developer three: Start game.

developer two: Rewrote XML parser to do the AI and specials.

Problems:

developer one: None

developer three: none

developer two: None

To Do:

developer one: Get game playing started, be able to select

developer three: Buttons scale with text.

developer two: Work on unit placement, computer cant be placed through api and limit where players can put stuffs

Monday March 12

Done:

developer one: Left bar AI added buttons, scales

developer three: Rescaling buttons, implementing animations

developer two: Programmed computer player

Problems:

developer one: None

developer three: With unit death

developer two: Computer player calling, for unit setup

To Do:

developer one: Re orient board, be able to look at unit stats

developer three: Animations, graphics for win

developer two: Win conditions + letting comp player know

#### **Iteration Four:**

Thursday March 15<sup>th</sup>

Done:

developer one: Worked out some bugs, cleaned up some front end

developer three: Win screen done,

developer two: Win conditions, computer player stuff

Problems:

developer one: None

developer three: None

developer two: Mock listeners

To Do:

developer one: Start working on text problem

developer three: Animations

developer two: Game Log

Monday March 19<sup>th</sup>

Done:

developer one: Added shift+click assign multiple orders, researched front end text

developer three: Destroy animation display in right place, display damage

developer two: Started game log

Problems:

developer one: opengl + text = bad

developer three: None

developer two: Figuring out how game log works

To Do:

developer one: Import FenGui to get graphics working

developer three: Work on front end display log

developer two: Finish GameLog

Thursday March 22<sup>th</sup>

Done:

developer one: FengGui installed and working with it.

developer three: Display log at the bottom.

developer two: Game Log

Problems:

developer one: FengGui windowing, figuring out api

developer three: None

developer two: None

To Do:

developer one: Continue Fengui

developer three: Game Over display proper winner, display enemy health, scroll through game log

developer two: Refactor GameLog to be a part of the Game. Implement an Ai(Sentry)

Monday April 2nd, 2007

Done:

developer one: Units Face each other

developer three: List page up and down.

developer two: Sentry Ai(Needs to be regotten) Game player refactoring.

Problems:

developer one: None

developer three: None

developer two: None

To Do:

developer one: Body decomposition, start diffusion ai.

developer three: Make damage float, Start instructions menu

developer two: Work on debugging Ai

### **Iteration Five:**

Thursday, April 5<sup>th</sup> 2007

Done:

developer one: Prototype for diffusion

developer three: Basic instructions menu(Not working, fixed though)

developer two: Learned basic of Ai

Problems:

developer one: None

developer three: None

developer two: Busy

To Do:

developer one: Control groups, gray scale death picture

developer three: Start on animating moves

developer two: Work on Hive stuff

Monday, April 9<sup>th</sup> 2007

Done:

developer one: Double click for units, AI can select shared AIs, dead units, hold shift and deselect AI units

developer two: Worked on Hive

developer three: At test, couldn't make it to meeting

Problems:

developer one: None

developer two: Runtime exception that couldn't debug

developer three:

To Do:

developer one: Control groups, Implement some Ais

developer two: Add units for Beginner game

developer three:

Thursday, April 12<sup>th</sup> 2007

Done:

developer one: Unable to attend

developer two: None

developer three: Movement + Defend Hive

Problems:

developer one: Lack of time due to other classes  
developer two: Lack of time due to other classes  
developer three: Lack of time due to other classes

To Do:

developer one:  
developer two: Content generation  
developer three: Bodies decomposing, Content generation

Monday April 16<sup>th</sup> 2007

Done:

developer one: Working with resizing hexes, Removing a unit  
developer two: Added beetle and Tarantula units, and area effect attacks  
developer three: Rebalancing, Parasite AI/special

Problems:

developer one: Couldn't get it to work.  
developer two: None  
developer three: None

To Do:

None, end of iterations.

## Appendix E: Developer Exit Survey

### Developer One:

Did you feel that the process brought problems to knowledge quickly?

On the development side: I think it helped to know who to contact when a bug was found. So if I found a bug on the backed of the program I knew I could either contact developer two or the principal investigator. This was also true for them when they found a bug on the front end as they knew to contact either me or developer three.

On the play tester side:

I wouldn't say they helped find problems, but rather the feedback which the play testers gave was very helpful in guiding the development of the program.

What do you like and dislike about the meetings with the players?

I liked the whole thing. This could be a biased opinion as the play testers are all friends of mine; but any criticism, positive and negative, helped guide the development forward. As a side note: I didn't mind if the testers said they didn't like something that was added since the last play-test because at most I would have spent a week or so on that part, so it was not a huge waist of time if they didn't like it.

-Comments-

I enjoy the Agile Methodologies in general and I think Scrum is the easiest to conform to, especially when compared to extreme programming. I liked the feedback from players because after I knew what they wanted, I could add it. I also liked the daily (bi-weekly in our case) Scrums because it let me know what everyone had implemented since the last scrum. The meetings also allowed me and developer three to divide the work quickly and get working on what we needed to get done for the next scrum.

(I think he said this in the Scrum book, if he didn't then don't worry about adding this) The one part of the Scrum methodology I would have liked to see use implement was that no one touches someone else's code.

### Developer Two:

Did you feel that the process brought problems to knowledge quickly?

As far as the problem of figuring out what the players wanted I think that problems where determined almost immediately. Also since our meetings were twice weekly all developmental problems were discussed in a short amount of time since discovering them. So yes I think that this process was good at bringing problems to knowledge.

What do you like and dislike about the meetings with the players?

I liked hearing their input, mainly because they had ideas that I hadn't thought of. Features that would make the game more interesting and easier to use were things that I hadn't thought of since I only cared about making it functional. There is nothing I disliked about them. One thing I did feel however was that some of the players did not understand what kind of resources where available to us for development and so some of the requests they were making were unreasonable in some ways.

Comments about the process?

I think that the idea of doing only what is requested from the customer when it is needed is okay for maintaining or improving an already existing system. However I really like structure and planning, and I think that I could have programmed more efficiently with a well drawn out design of what the program should look like. I tend to get lost without being able to see the whole picture, and doing things bit by bit was a little bit confusing at times. I actually really started to develop the big picture of how the architecture of the program should be structured towards the end of the project. I also learned that working as group is much different than working individually and I think that this process was good for communication and sharing ideas among the group.

**Developer Three:**

Did you feel that the process brought problems to knowledge quickly?

We were able to address the problems quickly, as a team, and did not waste a lot of time on little bugs.

What do you like and dislike about the meetings with the players?

Liked knowing all the things that had been completed by the different people, and deciding on what to do next, so there was no overlap.

-Comments-

This was a really great first experience and I see that where one person is lacking, there was plenty of help to bring the understanding up to par, and this greatly improved the efficiency of our development.