Software Requirements Specification
For
Ant-World Game

Version: Final
Monday 12th March 2012

Software Engineering Group 11

Requirements Specification

Version: Final

12th March 2012

Approved by: _____ _____
Robert Johnson Date

Approved by: _____ _____
Mark Merriman Date

Approved by: _____ _____
Oliver McCarthy Date

Approved by: _____ _____
Andreas Nicolaou Date

Approved by: _____ _____
Mark Purser Date

Approved by: _____ _____
Joanne Robert Date

Approved by: _____ _____
Eleanor Shakeshaft Date

Approved by: _____ _____
Ben Watt Date

Document Revision History.

Date	Section	Changes	Author
02/03/12	All	Set up template	E.Shakeshaft
03/03/12	Introduction	Wrote sections	E.Shakeshaft
	System Requirements	Outlined requirements	E.Shakeshaft
05/03/12	Design & Implementation Constraints	Written	E.Shakeshaft
06/03/12	Introduction	Edited.	A.Nicolaou
	System Requirements	Updated	A.Nicolaou
07/03/12	Design & Implementation Constraints	Updated	E.Shakeshaft
	User Classes and Characteristics	Written	E.Shakeshaft
	System Requirements	Use Case Diagram	E.Shakeshaft
	System Requirements	Finalised Requirements 1-7	E.Shakeshaft
	CRC cards	Produced.	A.Nicolaou
	Sequence Diagram	Produced.	A.Nicolaou
	Class Diagram.	Produced.	A.Nicolaou
08/03/12	External Interface Requirements & Other Non-functional Requirements	Completed.	E.Shakeshaft
	Overall Description	Completed.	E.Shakeshaft
	System Requirements	Completed.	E.Shakeshaft
	Scenarios.	Scenarios written.	E.Shakeshaft
	System Requirements	Updated Use Case Diagram	E.Shakeshaft
09/03/12	All	Changed sections based on meeting on 09/03/12	E.Shakeshaft
	System Requirements	Added some req. that need incorporating (in bold)	M.Purser
	Sequence Diagram.	Re-done.	A.Nicolaou
	State Diagram.	Produced.	A.Nicolaou
	Scenarios	Edited	A.Nicolaou
12/3/12	All	Updated and finished all sections	A.Nicolaou & E.Shakeshaft

Table of Contents.

1. Introduction.

1.1 Purpose.

This Software Requirements Specification (SRS) identifies the requirements for the Ant-World Game (AWG) for the Software Engineering Module.

1.2 Document Overview.

Section 2 is an overview of the descriptions of requirements for the AWG, and the requirements are categorised and defined in section 3. Sections 4 and 5 contain general information to aid in the understanding of this specification.

1.3 Intended Audience.

This document is primarily intended for the design developers, but also as a reference for the project managers, programmers, QA testers, and documentation writers. Read over the overview sections and proceed through the sections that are most pertinent to your requirements.

1.4 Project Scope.

Our customer (referred to as the "client") is a games designer who has an idea for a new computer game, and wants us to provide a solid implementation of it. The game is a competitive two-player strategy game where the players ("users") design and upload an ant-brain to pit against each other. The ants are placed into two ant colonies within a randomly-generated world simulation which also contains two anthills, some food sources, and several obstacles. The ant world will then simulate the behaviour of both kinds of ants; they must explore the world, find food and bring it back to their anthill. The winner is the anthill which has the most food at the end of 300,000 rounds. The highest scores and players will be put on a website along with statistics. At the basic level, all the users require is the outcome of the game and statistics; however we will implement an attractive graphical interface to make the game more engaging.

1.4.1 Out of Scope

The software will not allow user profiles and it will not store ant-brain codes after the game(s) have completed; only the statistics and user aliases will be stored.

The system will not allow games to be played concurrently.

1.5 References.

Customer Requirements: http://www.sussex.ac.uk/Users/mfb21/se/project/require.html
Quality Manual: http://www.sussex.ac.uk/Users/mfb21/se/project/quality.html
Configuration Management: http://www.sussex.ac.uk/Users/mfb21/se/project/config.html
Project Website: http://dl.dropbox.com/u/12957105/Website/home.html
Project Plan: http://dl.dropbox.com/u/12957105/Website/Project_Plan.pdf
Google AI Challenge: http://aichallenge.org/

2. Overall Description.

2.1 User Classes and Characteristics.

Users (Players)	High frequency and primary (only) user, will have varying technical expertise but will not be required to have any technical skills apart from ability to code ant-brains. Target audience is expected to be but not limited to males aged between 16-40.

Users only involved in managing the AWG, the "Client".

Client Manager	Low frequency user, does not need any technical skills but will be expected to understand the licensing agreement. Has high privilege levels but will not be a user of the software or databases.
Client System Administrator	High frequency user, will have technical expertise with servers and databases (but not necessarily gaming), will be able to install and administer the software using only the Installation Guide. They will need to understand basic web administration to update the game website if needed, however this can be outsourced by the company.

2.2 Operating Environment.

The software will be designed to operate on PCs only and be will be cross-platform. Users will not be required to hold any software on their PCs as the game will be played as an applet within a browser; however they will have the option to run as a stand-alone program.

Chrome will be the recommended browser to use. The highest scores will be displayed on a website.

The Client will need a server and webhost.

2.3 Design and Implementation Constraints.

Programming language:	Java.
Website:	HTML, CSS & PHP – designed primarily for Chrome.
Graphical User Interface:	2D only required, must be designed with the target audience in mind. User must be able to adjust the speed of the simulation, as well as pause it. Statistics must be displayed alongside simulation. All information will be in English (UK) only.
Version Control:	GitHub - to be used on every update, and log all changes.
Programming Standards:	Client's organisation will be responsible for maintaining the delivered software and helping user's with issues.

2.4 User Documentation.

The client will be given a document, along with the software, which will contain two sections:

Installation guide: This is written for the system administrator (who can be assumed to have reasonable computing knowledge, but knows nothing about gaming in general this software in particular) installing the program. It must be possible to install the software and get it to run from the installation guide only.

User Manual: This documentation is to be read by users of the program (who cannot be assumed to have any particular computing knowledge), and shall describe the user interface and functionality of the system, in terms that the user understands.

It will also detail the licensing agreement for the software.

The client will also be given a website which will host the highest scores and statistics, user manual and where users(players) can run the game applet.

3. System Requirements

A high level use case diagram of the functions of the AWG with regards to the Users' interactions with the GUI, Website and Software provided in *Figure 1* consisting of: 1) User opens the website where the game applet can be viewed, or a jar file of the game can be downloaded. They may also download the User Manual 2) User chooses to play a single game or a tournament. 3) User uploads their ant-brain code – this is parsed and the results fed to the GUI, 3) If the ant-brain code is verified then the User can start a new game or join a game. 4) if the user is the first player in a game then they will either upload an ant-world to play in – which the software parse - or select the default world if it is a single game; if playing a tournament they will select the number of players (between 3 and 12) but will not upload a world. 4) The software runs the simulation(s) then sends the data to the GUI to be displayed to the user, and also sends data to the website for the highest scores.

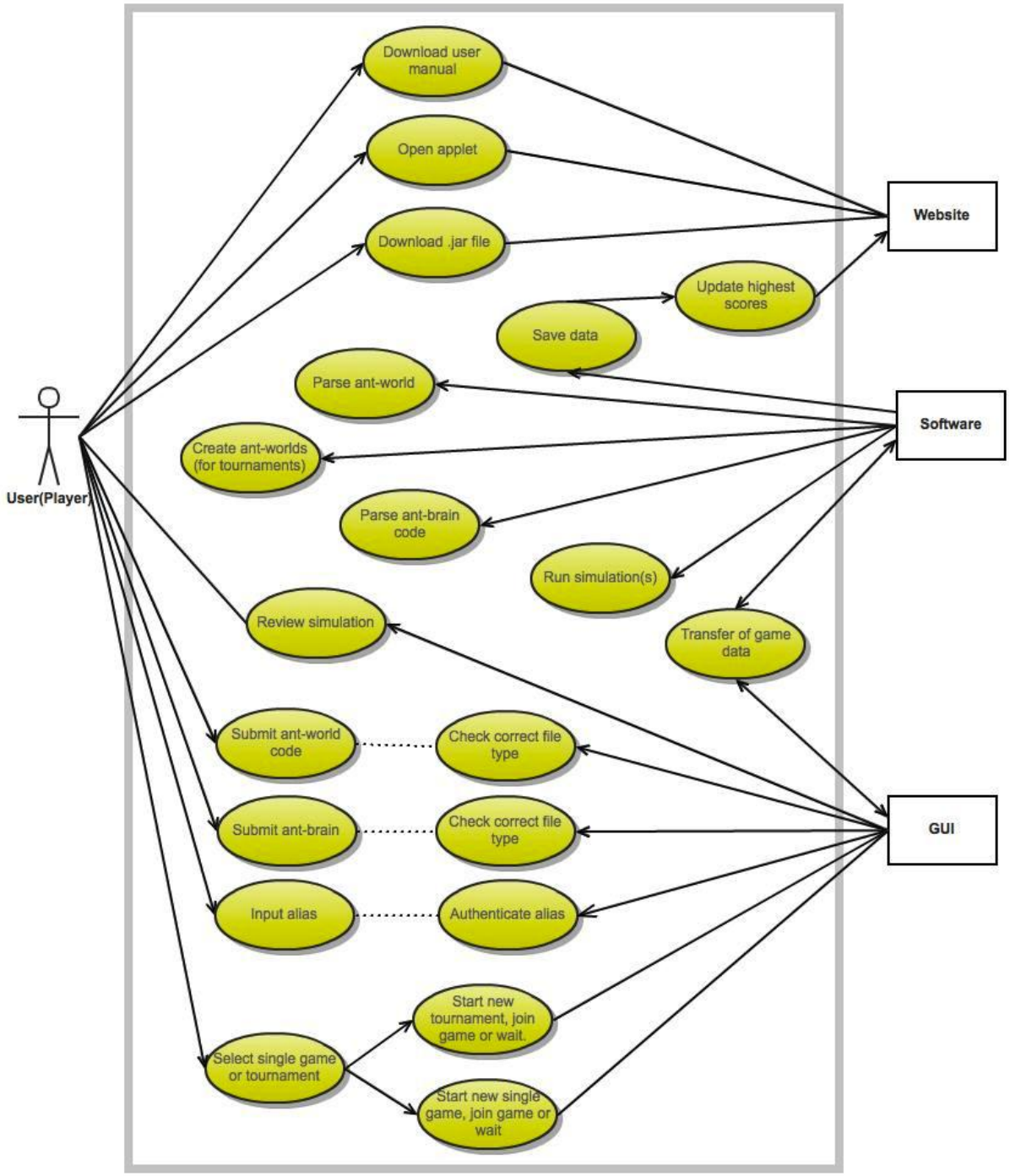

Figure 1 – High Level Use Case Diagram for the Ant-World Game.

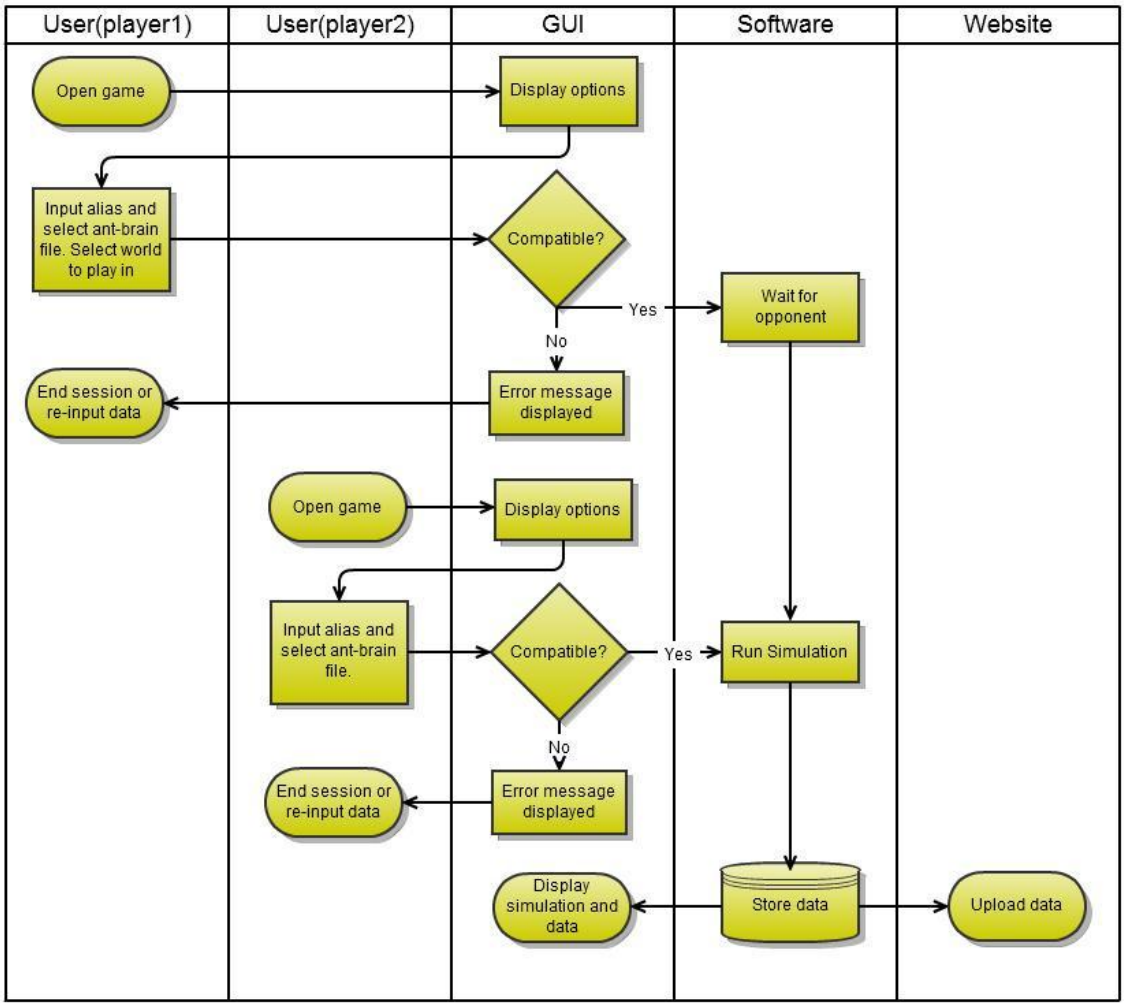

Figure 2 – High Level Swimlane Diagram for a single game in the Ant-World Game.

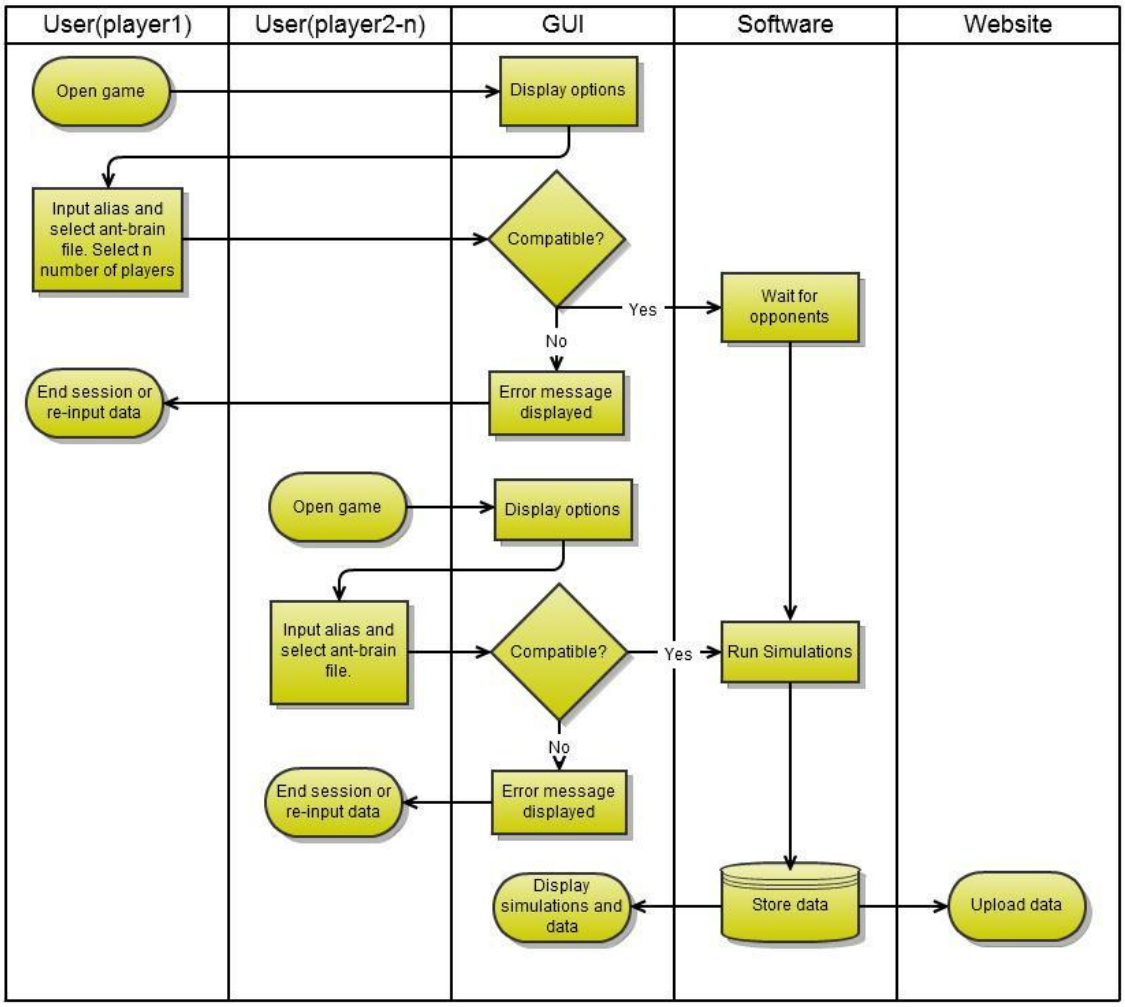

Figure 3 – High Level Swimlane Diagram for a tournament in the Ant-World Game.

3.1 Open website.

Description and Priority.

User (Player) opens up the game website where they can start the game applet.
Priority = high.

Stimulus/Response Sequences

Stimulus: User goes to website page.
Response: Website loads.
Stimulus: User starts game(applet).
Response: Game loads.

Functional Requirements

Website.Compatibility	The website and game should display properly on Chrome.
Website.Updates	The website should contain up-to-date data.

3.2 Single game.

Description and Priority.

User (Player) chooses to play a single game.
Priority = high.

Stimulus/Response Sequences

Stimulus: User is first player.
Response: Ask user to enter alias, upload ant-brain code, select default ant-world or upload ant-world code.
Stimulus: User is second player.
Response: Ask user to enter alias and upload ant-brain code.
Stimulus: A game is in session (either single or tournament)
Response: User cannot enter a game and must wait for current session to end.

Functional Requirements

SingleGame.FirstPlayer	Enter alias, ant-brain code and selects to use default world or upload their own.
SingleGame.SecondPlayer	Enter alias and ant-brain code.
SingleGame.GameinSession	GUI display message that a game is in session and to wait.

3.3 Tournament game.

Description and Priority.

User (Player) chooses to play in a tournament.
Priority = high.

Stimulus/Response Sequences

Stimulus: User is first player.
Response: Ask user to enter alias, upload ant-brain code and select number (*n*) of players.
Stimulus: User is any player up to and including *n*.
Response: Ask user to enter alias and upload ant-brain code.

Stimulus: User is player *n+1* or higher.
Response: User cannot enter the tournament and must wait for current session to end.
Stimulus: A game is in session (either single or tournament)
Response: User cannot enter the tournament and must wait for current session to end.

Functional Requirements

Tournament.FirstPlayer	Enter alias, ant-brain code and selects number of players between 3 and 12 (*n*).
Tournament.UpToNplayers	Enter alias and ant-brain code.
Tournament.GameinSession	GUI display message that a game is in session and to wait.

3.4 Authenticate alias and ant-brain code file-type.

Description and Priority.

User (Player) fills in an alias for high-score table and uploads their ant brain code.
Priority = high.

Stimulus/Response Sequences

Stimulus: User fills in alias name and selects file to upload
Response: System validates if alias conforms, and file is correct type. If correct allows user to submit code.
Stimulus: User submits code.
Response: Systems responds that code is being verified.

Functional Requirements

Authenticate.Alias	The system shall ensure maximum length of 8 characters and will respond with error message it the file is not.
Authenticate.FileType	The system must only accept plain text files, and will respond with error message it the file is not.

3.5 Parse ant-brain code.

Description

Once ant-brain code is uploaded the software must check if it is syntactically correct.
Priority = high.

Stimulus/Response Sequences

Stimulus: User submits code.
Response: System parses the code to check if it is syntactically correct.
Stimulus: Code is verified.
Response: Save code and send confirmation to User (Player) via GUI.
Stimulus: User (Player) receives confirmation of code.
Response: Game stars.
Stimulus: Code is not verified.
Response: Delete code and send confirmation of error to User (Player).

Functional Requirements

ParseAntBrain.CheckSyntax	Check syntax of code.

3.6 Single game Ant-world.

Description

The first user must generate a random ant-world for the game to take place, or choose the default world.

Priority = high.

Stimulus/Response Sequences

Stimulus: First user has opted to play a single game.

Response: Display option for upload of world, or choose default.

Functional Requirements

SingleGameWorld.Dimension	Upper limit size: 150 x 150 cells. Lower limit size: 10 x 10 cells.
SingleGameWorld.AntPositions	All ants are initially facing in direction 0.
SingleGameWorld.Positions	There must always be at least one empty cell between non-food elements. No elements must overlap.
SingleGameWorld.Ants	Each cell in the red anthill is populated with a red ant and each black anthill cell with a black ant.
SingleGameWorld.Food	No limit.
SingleGameWorld.Rocks	No limit.
SingleGameWorld.Anthills	2 anthills (hexagonal, size dependent on world size).
SingleGameWorld.Perimeter	The cells on the perimeter are always rocky.

3.7 Tournament Ant-world.

Description

The software must generate a random ant-world for the game to take place.

Priority = high.

Stimulus/Response Sequences

Stimulus: 2 Users (Players) have been allocated to play.

Response: Randomly create Ant-world and populate with ant-colonies.

Functional Requirements

TournamentWorld.Dimension	150 × 150 cells
TournamentWorld.Perimeter	The cells on the perimeter are always rocky.
TournamentWorld.Anthills	2 anthills (hexagonal of side length 7).
TournamentWorld.Rocks	14 rocks
TournamentWorld.Food	11 blobs of food (5-by-5 rectangle, with each cell containing 5 food particles)
SingleGameWorld.Ants	Each cell in the red anthill is populated with a red ant and each black anthill cell with a black ant.
TournamentWorld.Positions	Chose randomly. There must always be at least one empty cell between non-food elements. No elements must overlap.
TournamentWorld.AntPositions	All ants are initially facing in direction 0.

3.8 Parse Ant-world.

Description

Once ant-world is created the software must check if it is correct as per the specifications.
Priority = high.

Stimulus/Response Sequences

Stimulus: An ant-world has been created and populated.
Response: System parses the world layout to check if it is correct as per the specifications.
Stimulus: Ant-world is verified.
Response: Send information to GUI to display to Users(Players) that a battle is commencing.
Stimulus: Ant-world is not verified.
Response: Send information to GUI to display to Users(Players) that the world does not meet the requirements, and the first user must upload a new world or pick the default.

Functional Requirements

ParseAntWorld.CheckLayout	Check layout of Ant-world conforms to specifications

3.9 Run simulation.

Description

The game will commence and the entire simulation will run through.
Priority = high.

Stimulus/Response Sequences

Stimulus: An ant-world has been verified.
Response: First simulation is run.
Stimulus: First simulation is finished.
Response: Run simulation again with ant colonies/colours swapped.

Functional Requirements

Simulation.Length	Simulation will run for 300,000 turns.
Simulation.Repeat	Repeat the first simulation in the same world, but with the ant colony colour/ant-hills swapped
Simulation.AntColours	For the first simulation, the first user is red. For the second simulation the first user is black.
Simulation. AntCharacteristics	Can communicate or leave trails by means of chemical markers in order to find their way back to the anthill and store their food. Can sense but not modify the markers of the other species. Different ant species can attack each other by surrounding them; If an ant ever finds itself adjacent to 5 (or 6) ants of the other species, it dies When the ant dies as a result of an attack it becomes 3 food particles.

3.10 Game statistics.

Description

The game winner, statistics and points will be stored and the website will be updated.
Priority = high.

Stimulus/Response Sequences

Stimulus: All simulations for the game/tournament have been finished.
Response: Calculate winner and scores, save data.
Stimulus: Data about simulation(s) has been saved.
Response: Update website.

Functional Requirements

GameStatistics.Winner	Winner is User (Player) with the most food particles. Only the number of food particles currently in the anthill cells of each color are counted, food being carried by an ant does not count, even if it is standing on its own anthill
GameStatistics.Scores	Winner gets 2 points. If in a case of a draw, both players' get 1 point.
GameStatistics.WebScores	For highest-score table on website, this will be from total number of food particles counted. There will be a separate highest-score table for tournaments; this will show the number of wins, score and number of food particles. It will be sorted first by score, then number of wins and then buy number of food particles.
GameStatistics.Statistics	How many ant deaths. How many opponent deaths made.
GameStatistics.Website	The website will be updated with the new data.

3.11 Users(Players) view simulation.

Description

Users will be able to review their game/simulation.
Priority = high.

Stimulus/Response Sequences

Stimulus: Simulation has been completed.
Response: Send simulation to users in game.

Functional Requirements

ViewSimulation.View	Simulation will be shown on screen, with running statistics beside it.
ViewSimulation.Navigation	User(Players) will be able to speed up or slow down simulations, as well as pause it.

4. Scenarios.

4.1 User input – Primary.

The user inputs alias and uploads ant-brain file via GUI, clicks submit and the GUI responds that the ant-brain code is being verified.

4.1.1 User input – Secondary

The user inputs alias and uploads ant-brain file via GUI, clicks submit and the GUI responds that the alias does not conform to the specifications and/or the file is not a compatible file type.

4.2 Entering a single game – Primary.

The user wishes to play a single game. They upload their ant brain and wait for an opponent player to enter the game. The first user defines what world to play in.

4.2.1 Entering a tournament game – Secondary

If there is a game in process the players cannot play a game.

4.3 Entering a tournament game – Primary.

The user wishes to enter a tournament. They upload their ant brain and wait for other players to enter the game. The first user defines the number of players in the tournament.

4.3.1 Entering a tournament game – Secondary

If there is a game in process the players cannot play a game.

4.4 Ant-brain code – Primary.

The ant-brain code is syntactically correct, the GUI responds that it has be verified and the user will proceed to be entered into a game/tournament.

4.4.1 Ant-brain code – Secondary

The ant-brain code is not syntactically correct, the GUI responds that it has be not been verified and the user should review their code.

4.5 Ant-World c for single game – Primary.

The user uploads an ant-world, the system verifies it complies with the regulations and the simulation runs and completes.

4.5.1 Ant-World simulation for single game – Secondary

The user uploads an ant-world, the system rejects it as the layout does not comply with the regulations, and a new world is created.

4.5.2 Ant-World simulation for single game – Secondary

The user opts to use the default ant-world.

4.6 Ant-World simulation for tournament – Primary.

The software produces a random world for users to play in which meets the requirements.

4.6.1 Ant-World simulation for tournament – Secondary

The world does not meet the requirements.

4.7 Calculate winner and upload statistics – Primary.

A simulation starts and a list of statistics is created. As the simulation plays out, the statistics are updated. They contain then data such as food gained and the number of ants that have died on each team.

5. CRC cards.

The GUI class displays the game on the website.

Class: GUI	
Responsibilities	**Collaborators**
User uploads ant-brain	World
User starts the game	Ant-Brain

The State class updates and holds the actions of the ant.

Class: State	
Responsibilities	**Collaborators**
Update the ant's state	Ant
Update the world	Ant-Brain
	World

The World class shows the cells in the map where the game takes effect.

Class: World	
Responsibilities	**Collaborators**
Shows the world	Ant
The map of the game	Ant-Brain
Show ants markers	Cell

The Cell class shows what is implementing in the world class.

Class: Cell	
Responsibilities	**Collaborators**
Food	World
Anthills	
Rocks	

The Ant-Brain class shows the actions of an ant.

Class: Ant-Brain	
Responsibilities	**Collaborators**
Directions to move	Ant
Pick/Drop food	World
Identify opposite species (enemies)	GUI

The Ant class depends on the Ant-Brain been uploaded.

Class: Ant	
Responsibilities	**Collaborators**
Colour	Ant-Brain
Carrying food to anthill	World
Fight with other species	
Movement (Direction)	

The Tournament class simulates a competition game between 2 players or more/

Class: Tournament	
Responsibilities	**Collaborators**
Checks for more than 1 player	Ant-Brain
Show the winner	World

The High Score class depends on the winner of the game. After the end of each game a winner will be chosen and its high score will be uploaded to the website.

Class: High score	
Responsibilities	**Collaborators**
Checks for high score	Tournament
Shows the winner	
Upload to website	

6. Sequence Diagram.

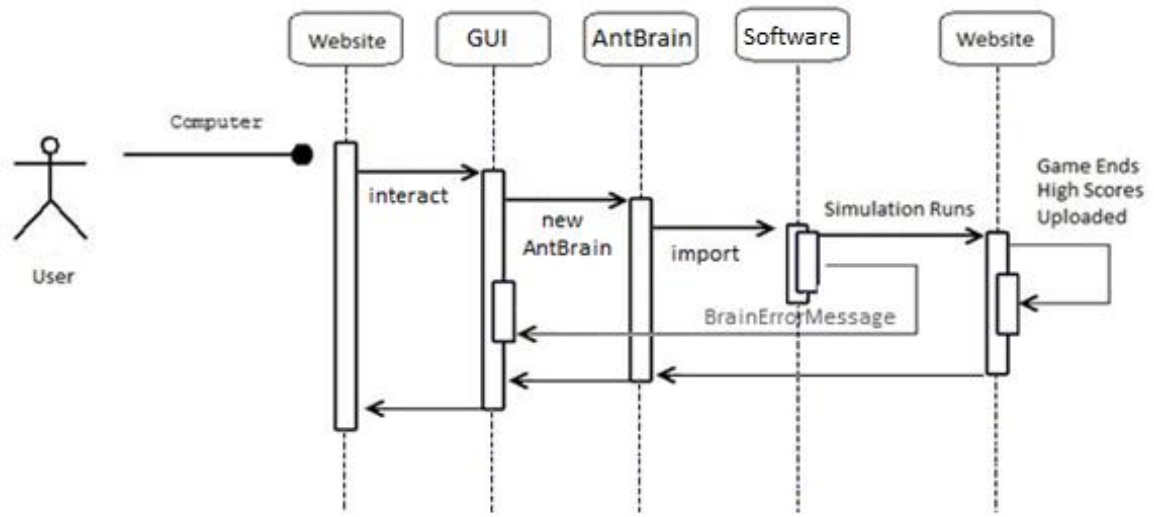

Figure 4 – High Level Sequence Diagram for the Ant-brain import for the Ant-World Game.

7. State Diagram.

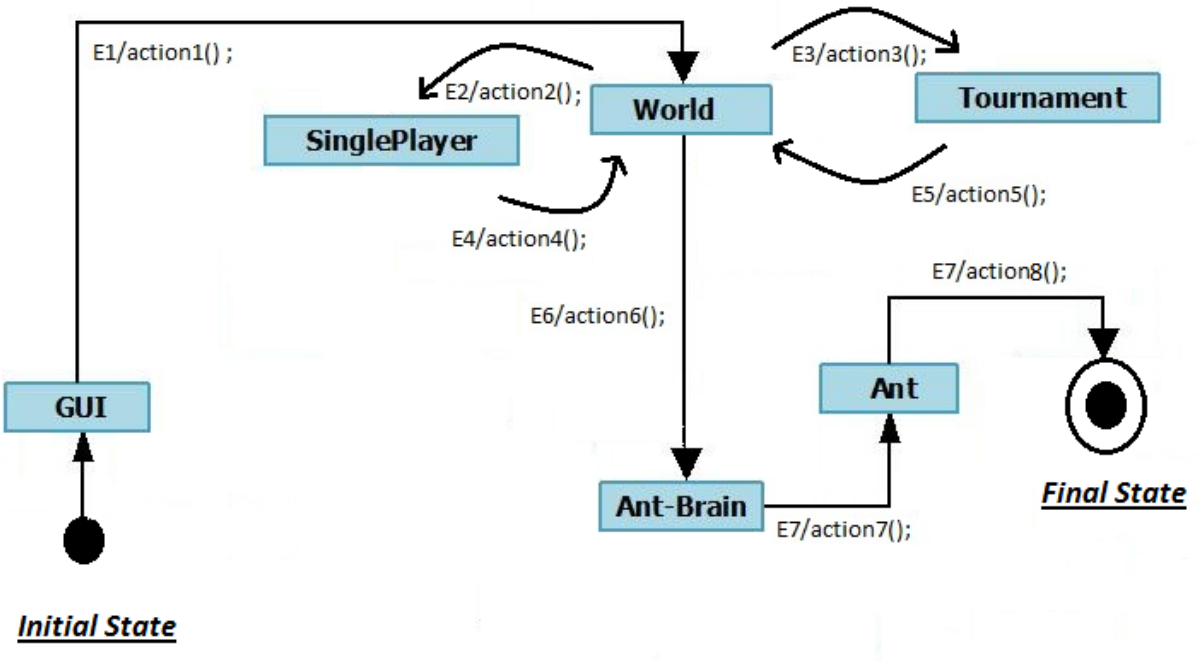

Figure 5 – High Level State Diagram for the Ant-World Game.

8. External Interface Requirements & Other Non-functional Requirements

8.1 User Interfaces.

UI-1: The system shall provide a help link from the main page; this will navigate to a help page where users can download the User Manual as a PDF file.

UI-2: The website and GUI shall contain text of at least a font size of 10pt.

UI-3: The website and GUI shall use colours which do not interfere with the readability of the page

8.2 Hardware Interfaces.

No requirements have been identified.

8.3 Software & Communications Interfaces.

SCI-1: The software must communicate with the website which will store the statistics.

SCI-2 The game state after each round shall be output to a trace file in an identical format to the customer 'dump' files

8.4 Performance Requirements

PE-1: The system shall accommodate 12 users at a time.

PE-2: All webpages be fully downloadable in no more than 20 seconds.

PE-3: If the connection between the user and the system is broken prior to the user submitting a verifiable ant-brain then they will be required to re-open the game. If the ant-brain has been verified then the simulation will continue to run but the user will not be able to view the simulation, the other user (opponent) will though. If in a tournament the user's contribution/points will not be counted at the end.

PE-4 The visualization update rate shall be no slower than 10 frames per second.

8.5 Safety Requirements

No requirements have been identified.

8.6 Security Requirements

SE-1: Users will not have log-ins so no data is stored or kept about users.

SE-2 The interface protocol shall be an unencrypted TCP/IP connection.