Coursework

Analysis

Idea:

Ant simulator

The game will simulate a colony of ants which can act independently to gather food and explore the surrounding area. Users will be able to take manual control of an ant or group of ants at once to manipulate their movement and other behaviour. To add complexity to the game, ants should be able to communicate with each other using pheromones – chemical trails which signify a message. For example, if an ant finds food which is too large to carry back, it will create a specific pheromone trail back to the colony, so other ants can come help.

I believe that a survival-based strategy game would be most engaging, as the game is likely to be designed for teenagers who will be experienced in gaming and will find little difficulty in learning the controls. However, other modes such as a sandbox or higher difficulty levels may also be desirable depending on the audience playing, so this is subject to change. In a survival mode, users would have to gather food and expand the colony while also managing threats such as rival ant colonies and natural events e.g. floods. Whereas a sandbox mode would allow users to experiment without worrying about resources, as they would be unlimited. Unique variants of ants could also be implemented with different sets of abilities, such as fighters that are effective at battling other colonies, and food gatherers that can carry a larger mass of food and move faster.

I believe setting the game to take place in a forest would be appealing as it would be a natural scene for ants with vibrant graphics and would have a slower pace to allow for simpler and less frustrating gameplay for users. Other settings could also be used, such as an urban area where food gathering may involve scavenging for food dropped by humans, or a desert where food is rarer, and a moisture/temperature level could be added for an additional challenge in resource management. Research will be required to decide which of these settings is best to implement and whether there should be multiple levels or just a default one. Additionally, research is needed to decide on whether the game should be displayed using a top-down, or an isometric view, which may be harder to implement, but more visually appealing and engaging.

I would also like for the game to have some educational quality in illustrating the concept of emergent behaviour. This is when lots of simple predictable actions on a small scale can act together to incite “intelligent” behaviour at a large scale. This leads to a holistic view that the whole (colony) is more than just the sum of its individual parts (ants). For example, the previously mentioned pheromone trails which lead to food would seem to an observer like intelligent behaviour, as the ants find food, and form a line to carry it back very quickly. However, each ant simply sees a trail and follows it. I hope that this game will make the concept more intuitive to users, as they can see it simulated in real time. This concept is also very important to understanding consciousness in neuroscience, as scientists want to describe how a brain made of simple components such as neurons is capable of high level thought.

Research:

Similar games:

[Empires of the Undergrowth](https://en.wikipedia.org/wiki/Empires_of_the_Undergrowth) (released in 2024)

A screenshot of a video game

AI-generated content may be incorrect.

A screenshot of a video game

AI-generated content may be incorrect.

A screenshot of a video game

AI-generated content may be incorrect.

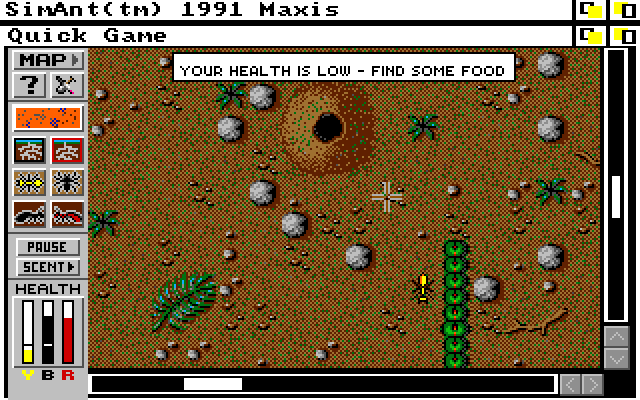
(Fire Ant Trailer - Empires of the Undergrowth Early Access, 2025)

This is a similar game involving management of an ant colony, where the user controls a colony of fictional “gene stealer” ants which can steal the genetic data of other colonies and create ants of the same type. The gameplay consists of many side missions to unlock new ant variants.

The game has many features which I will be taking into my project, as they add more variety and new mechanics to it, however I feel that some features are unnecessarily complex, and will not be worth the time to implement as a solo developer such as a storyline and side missions. I like the contrast of using both 2d and 3d graphics, however 3d would be difficult to implement without a lot of research in my GUI software, so will not be possible in the time frame of the project.

|  |  |
| --- | --- |
| Features to implement | Features I do not like |
| Tile based system as shown in first screenshot.  This will be simpler to implement than 3d graphics, as I am not experienced in using my chosen GUI, so this would not be worth the effort required to learn and create 3d effects. | Storyline for player to follow  I believe this is not necessary for my game, as players will gain greater satisfaction from learning to play by themselves, and it would be time consuming to create multiple specially curated levels for the user to play. |
| Underground and Overground parts of map(first 2 screenshots)  This is an idea which I had not considered, but could improve the game, as it would be more realistic, and would add more variety to the gameplay, as the user can manage the internal or external parts of the colony | Skill points to customise ant stats (3rd screenshot)  This feature may make the game too complex for beginners and does not significantly affect the gameplay.  My game may include different varieties of ants however to achieve a similar effect. |
| Rival ant colonies  This is a critical feature to the survival element of the game and will create a challenge for the user to expand against opposition, so that the game isn’t trivially easy and uninteresting. | Cosmetics  The player can customise their ants with different colours, or hats. I feel that this is not important enough to spend time developing, and it would be unrealistic. |
| Other animals  The game includes animals such as spiders, frogs and insects. I feel that this would add more immersion to the game for the user and would make it more engaging. |  |

[SimAnt](https://en.wikipedia.org/wiki/SimAnt) (released on windows in 1991)

(Universal Videogames List, 2025)

|  |  |
| --- | --- |
| Features to implement | Features I do not like |
| Different modes of gameplay  SimAnt includes 3 modes: A quick game, full game and experimental game.  The quick and full game will be like my survival mode, as the player must eliminate a colony of rival ants which are red, however I think I will add a difficulty setting to survival mode, rather than separate it further, so that users have more control over the length/difficulty of the game and can therefore adjust it if they are not enjoying the intensity of the game  The experimental game is like my sandbox mode, as the player will be able to place walls, pheromone trails, food, ants and other objects which will allow them to experiment with different strategies without consequences if they aren’t effective. | Win condition  In this game, the player wins if they eliminate the rival ant colony, and human owners of the house.  I would prefer for my game to have no winning end condition, so that the user can play if they survive, however the game can end if they are eliminated.  This means that the user can compete with themselves to try to survive for a longer time, or a scoring system could be implemented for the user to try and reach a high score.  This will provide more positive feedback and reduce frustration if the user is not able to survive as they won’t “lose”. |
| Control of an ant  In SimAnt, the player controls a yellow ant which can be changed to another ant by double clicking. I would like to implement this feature to add interactivity but also include the ability to control a group of ants at once rather than just one as this will allow the user to control the colony on different levels and will help with the educational aspect of showing how the colony operates. | Humans  In SimAnt, the game is set in the garden of a house where humans live, and eliminating the humans is one of the conditions required to win.  My game will not focus on human interaction with the ants, as it should focus on their natural behaviour to be more realistic, and will have a forest as the default setting.  However, an urban area could be added as an extra map to play on if there is time at the end. |
| Natural hazards  The game includes hazards such as rain, the lawnmower and human footsteps. This provides a new level of challenge as the user must adapt to new events, at least for the first few attempts where they have not seen the hazards before. | Mouse based controls  The majority of controls in the game involve double clicking with a mouse to select an area to travel to. While this is an intuitive and easy to learn method for beginners, it can become tiring and inconvenient after only a few minutes of gameplay.  For this reason, I will be using keyboard controls for the game, as they will allow for faster, more convenient controls, and my game is likely to be played by teenagers who are experienced to some extent in games, therefore they should conform to common standards in gaming, such as using the wasd keys for movement. Keyboard controls will be optimal as they will allow for more complex control without straining the user over a long game. |
| Realism  The developer of the game, Will Wright, attempted to make the game mimic the complexity of real ant colonies, and called it a form of “self-directed learning”  (Wikipedia, 2025).  I would like for my game to have the same quality of being true to real life, so that users can create an accurate image of how colonies function. |  |
| Pheromone trails  The game implements trails of pheromone chemicals which the player or their computer-controlled ants can leave to communicate the location of food.  My game will also be implementing this feature, as communication between the ants will help to ensure a realistic representation of a colony, while also adding functionality to the game, so the user has a range of mechanics at their disposal. |  |

[Spore](https://en.wikipedia.org/wiki/Spore_(2008_video_game)) (released 2008)

(Wikipedia, 2025)

A screenshot of a video game

AI-generated content may be incorrect.

(Image of late cell stage of gameplay)

Designed by Maxis, the same development studio as SimAnt, Spore is a game where the user controls a species throughout the stages of its evolution. It takes place over 5 main stages: cell, creature, tribe, civilisation, and space. While this game is less similar to my project than the other two, it is highly popular, achieving a 91% score from PC Gamer UK (Wikipedia, 2025), and was created by a large development team, therefore will more accurately depict what features my target audience of teenage gamers would enjoy.

|  |  |
| --- | --- |
| Features to implement | Features I do not like |
| Concurrent control of multiple organisms  In the last 3 stages of spore (tribe, civilisation and space), the user no longer controls a single member of their species, but a group of them.  In my game, I would like for the user to be able to control multiple ants at once to give them a better idea of the function of an ant colony on a high level and it will improve the user experience, as there is a higher skill cap, allowing them to continue seeing improvement for a long time. | Separate stages of gameplay.  Spore includes 5 main stages which the user must play through, leading to excessively long games ranging anywhere from 12 to more than 100 hours. This could be implemented on a smaller scale to reduce the time frames to more reasonable amounts, however I don’t believe this is suitable for my game and would not significantly improve user engagement enough to justify implementation. |
| Scrolling 2d graphics  I think the style of graphics used in spore would be the ideal style for my project. A 2d top down view is used, and scrolls to follow the movement of the user. This would be relatively simple to implement compared to other styles such as the use of 3d or isometric graphics, and is very visually appealing. | Customisation of creatures  At the start of the creature stage, the user is able to create their own customised creature to control. This provides more interactivity to the game, however it will not be used in my game.  It would be time consuming to implement in my project, and a similar effect will be achieved using separate varieties of ants if there is time e.g. fighter ants, food gathering ants or a queen ant |
| Food gathering  A key aspect for the majority of spore is gathering enough food to grow your creature/group of creatures.  In my game, it will be necessary to find food to expand, and it will be a core aspect of gameplay providing a simple challenge to start the user off, which can be added to later by other additional challenge | Side missions  Spore contains additional missions for the user to complete which will grant them rewards such as an ability or money. I will not include this in my game, as it would not make sense to include in a realistic simulation, and it would make the gameplay more complicated for users to learn, especially if additional abilities were granted as rewards. I believe it will be better to keep the core gameplay mechanics simple, but demanding of the user to act quickly and effectively for a more engaging experience. |
| Allied species  In spore, the user can make allies with another species by imitating their behaviour. These species will not attack the user like they normally would. I believe that it could be an interesting feature to add to my game for an increased level of strategy, however it would be difficult to balance correctly, as the user shouldn’t be able to make allies too easily. |  |

Stakeholders:

My users should be familiar with using a keyboard to control movement and should be able to react fast enough to perform adequately in a survival scenario due to the nature of the game. While I considered adding features for young children to have a better experience with the game, they will be less experienced and may find it difficult or frustrating to play in a survival mode. However, they may like to play the sandbox mode if it is implemented, since there will be less pressure. Older audiences would be more likely to find the gameplay difficult, as they would be inexperienced with fast paced digital games and may find it hard to pickup controls without playing for a moderate amount of time. The game should be somewhat educational, so the users should be interested in the concept of the game, how ant colonies function, to get the most benefit from it. For this reason, I believe the most suitable audience for my game will be teenagers with some experience in gaming and ideally will be interested in the topic of my game.

I plan to gain feedback from my target audience of teenagers who enjoy gaming, for guidance on the initial user requirements, and for evaluation at the end of the project. I will send screenshots for graphical aspects for feedback on the aesthetics of the game, and videos of gameplay, or a prototype of the game, for the final evaluation of the gameplay.

For my initial user requirements, I researched my audience’s view of the following aspects of the game:

A blue circle with black text

AI-generated content may be incorrect.

All responses were yes, so this survey will represent my target audience very well. Responses were collected from within my college, so should only include people in the age range 16-18

How difficult do you like survival games to be? 1-10

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a graph

AI-generated content may be incorrect.

While I initially believed that a storyline would not be beneficial for my game, it appears that my target audience disagrees, so I will change my game to include a storyline rather than being open to the user.

How important are the graphics of the game? 1-10

A screenshot of a computer

AI-generated content may be incorrect.

Wide range, however generally lower than the scores for gameplay

How important is the gameplay? 1-10

A screenshot of a computer

AI-generated content may be incorrect.

Very high scores, implying that the gameplay is a very important aspect, and should be the main focus of my time developing

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a graph

AI-generated content may be incorrect.

The “other” response was WASD HJKL which I will treat as WASD. Almost all users wanted WASD controls, with only one exception out of seven responses

A screenshot of a graph

AI-generated content may be incorrect.

Wide range of responses, but mostly 20 minutes to an hour. This will be the default time that I will aim for, but settings could be introduced to alter this by changing difficulty (time will be largely dependent on the skill of the player and the difficulty of the game.)

A screenshot of a computer

AI-generated content may be incorrect.

Fairly even split, so may implement both if there is time at the end

A blue circle with black text

AI-generated content may be incorrect.

All responses were positive, so difficulty will constantly increase.

A blue circle with black text

AI-generated content may be incorrect.

Majority voted for three or more maps, so this should be a focus despite potentially being time consuming

A screenshot of a computer screen

AI-generated content may be incorrect.

Very positive response, so sandbox mode will be implemented.

Results:

[Stakeholder survey.xlsx](https://bhasvic-my.sharepoint.com/:x:/r/personal/theo_thomson24_students_bhasvic_ac_uk/Documents/Stakeholder%20survey.xlsx?d=w29125c793ab44adfb35c0588e11a7b7d&csf=1&web=1&e=bDjbJW)

Requirements:

|  |  |  |  |
| --- | --- | --- | --- |
| Requirement | Explanation | Justification | Priority level |
| Main menu | Menu for navigating between settings and starting game | This will allow the user to get ready before starting the game, and is present in the similar games that I have researched | Essential |
| Control of an ant | The ability for the player to control the movement and other behaviour of an ant. | This gives the user input to the game, and is how they will interact with the surroundings. | Essential |
| Control of multiple ants | The ability to simultaneously control many ants. This will probably have less sophisticated control compared to the individual ant. | This allows the user to operate at a higher level, and see the influence of their behaviour on the colony at a larger scale. | Highly desirable |
| Food to be gathered | Piles of food should spawn around the map which the user has to navigate to survive | This is the main aspect of survival, and provides a basic challenge to the user, which will be enhanced once more conditions are required to survive. | Essential |
| Other resources to gather | Could include resources such as water, or materials to build a simplistic structure for protecting the colony | This would add additional challenge to the game, as many users wanted a high level of difficulty, but is not essential as this can be implemented through other means. | Desirable |
| Pheromones | Chemical trails which ants use to communicate a message, such as the presence of a nearby rival colony, or food. | While not completely necessary for the gameplay, it is vital for the educational aspect of the game, and so I will place higher priority on it than similar aspects. | Highly desirable |
| Multiple vari*ants* of ant | This would include ants with different strengths and weaknesses, which the player can take advantage of to complete tasks more effectively. Examples could include food gatherers, fighters and a queen ant. | This will more effectively parallel the function of a colony in real life, allowing it to be more effective at simulating it for the user. Also, it will implement more strategy into the game, which I feel may not be possible to implement to the extent that I originally planned, as my users would prefer a more skill based game. | Highly desirable |
| Natural hazards | Random events such as rain or predators which are likely to significantly damage the population of the player’s ant colony | While I did not gather data in my survey about this feature, I believe that it will be appreciated by my audience, as it will randomly increase the pace of the game, which means that the user will have to react faster and play more effectively for a short period. This will be more engaging than having a constant amount of challenge throughout the whole run. | Desirable |
| Rival ant colonies | Enemy ants, which will consume food, and can fight the player’s colony | This adds a more advanced challenge than just searching for food, and creates the possibility of multiplayer play, though I doubt that will be possible in the time frame. | Desirable |
| Win condition | Eliminating all of the rival ant colonies will give the player a win, and end the game. | The majority of survey respondents preferred this option over a game which lasts indefinitely, until the player has no resources. This will also provide a sense of accomplishment to the user if they manage to win after a close game, or on a high difficulty. | Highly Desirable |
| Endless mode (no win condition) | No win condition; the player is able to keep playing as long as they survive. Difficulty will increase over time to ensure a reasonable time frame. | My audience were closely split on the issue of whether there should be a win condition, so it may be worthwhile to implement both options. All of the respondents who wanted this option voted for increasing difficulty over time. | Desirable |
| Scoring system | An algorithm to determine the player’s success as a numerical value based on stats such as food eaten and time survived. | This will give a player some idea of how their run compares to other runs that they have done in the past, and can allow them to see their progress, improving engagement. | Possibility |
| Settings | Customise aspects of the game, such as the mode of play and difficulty | My survey has shown a range in the wants of my audience (questions 2 and 6). I believe settings should be feasible to implement within a reasonable time frame, and will make the game accessible and engaging to a larger number of people. | Highly desirable |
| Sandbox mode | A mode where the user has access to unlimited resources and cannot lose. | My survey has shown a high demand for sandbox mode with over 80% of responses saying they would like it | Highly Desirable |
| High difficulty | The game should be challenging to survive, and have a high skill cap. | Most users in my survey gave scores above 8 for difficulty they would like, and the lowest response was 7. This is what I was expecting, as my audience is experienced in gaming. | Desirable |
| Customisable difficulty | A setting to alter the difficulty of the game, which will also likely affect its timespan | There was a slight range in the timespan that stakeholders in my survey voted for, and the difficulties that they wanted (questions 6 and 2). Increasing the difficulty will make it harder to survive, and therefore decrease the length of the game and visa versa. | Possibility |
| Keyboard controls using wasd keys | Control of movement in each of the 4 directions using the keys W,A,S and D | Of the 7 responses, 6 wanted movement to be controlled using the WASD keys, with the other response voting for mouse controls | Desirable |
| Detailed graphics | High quality aesthetics which make the game more visually appealing. The game is likely to be shown from a top down view, with 2 dimensional assets | This will give the game a cleaner and nicer look, which could attract new players, and improve the user experience. However, my audience gave relatively low scores for graphics they would like compared to gameplay. This means that it should take up only a small amount of development time. Using previously made two-dimensional assets from other creators is therefore likely to be the most effective solution if suitable ones can be found, as it will be simpler to add. | Possibility |
| Multiple maps | A set of different settings which the player can select from. These may have slightly differing gameplay, and will have different graphics. These could include a forest, desert and urban area. | This will add a slight variation to the game, allowing players to try different maps to avoid gameplay becoming repetitive or boring. Many users want this feature, with 57% of responses voting for 3 or more maps. On the other hand, this is likely to be relatively time consuming, and add comparably low amounts of content to some features. Therefore it is important, but not essential. | Desirable |
| Storyline | A story which will guide the player, explaining where they are, and how to play. | Despite initially believing this feature would not be worth the effort to implement, users have shown a high demand for it. Also, this can act as a tutorial to teach players how the game works, hopefully reducing the number of players who lose interest early due to its high difficulty. | Highly desirable |

|  |  |  |  |
| --- | --- | --- | --- |
| Pause game | The option to stop the game midway through, and continue it later. | Users may need to leave as the game will take a reasonable amount of time to complete by default (20 minutes to an hour). Therefore, they must be able to pause, or it could be frustrating to lose progress. | Essential |
| Save game | The option to store the game state as a file, allowing the user to play again later, even if they close the app or restart their computer. | This would allow users to complete a run of the game over multiple days, without having to leave the application open. However, it may be quite difficult to implement for such a small feature, therefore it will be left until later stages of development in case there is not time to implement it | Desirable |

Limitations:

As a solo developer, I wont be able to implement every feature that is labelled as desirable. Therefore, I will prioritise the essential features first, and focus my attention to the aspects which stakeholders value most.

|  |  |
| --- | --- |
| Limitation | Explanation |
| Variety of ants | There are likely to be only a few varieties of ants added to the game, as each one would require additional time to develop new features for, and including too many may make it confusing for the user to keep track of. |
| Graphics | My survey showed that graphics were less important than the gameplay to my audience, so they will be quickly made, or I may look for existing assets which I can use. Also, realistic graphics would be likely to cause performance issues, if features such as raytracing were used to simulate lighting and shadows. |
| Performance/Scale of colony | My game may be unable to run on low end computers, as it is simulating a large number of ants at once, which is likely to take up a significant amount of memory to store all their data. I will try to optimise memory usage throughout the project to avoid having too high requirements. Making the game run smoothly will be more visually appealing to the user, and allow them to run it with other applications open |

|  |  |
| --- | --- |
| Multiplayer | I do not intend to include multiplayer functionality in my project unless there is significant time left after developing the essential and desirable features.  It would not be effective for my game, as it will be designed to take between 20 minutes and 1 hour, so users may not have time to do a full game online which would lead to the other player being left alone. However, in singleplayer the user could pause the game or save it instead, if a saving feature is added. |
| Other types of animals | The game is unlikely to feature other insects or animals despite the locations where it will be set, such as a forest, likely having them in real life. It would be more time-consuming to implement than simply adding other varieties of ant, since they would have completely different behaviour, and would need new graphics. |
| Number of maps | My survey showed that most users want at least 3 maps, and preferably more. This may be time consuming to create, as each one will need to be unique and use different assets. This could potentially be made easier by procedurally generating maps so that I do not need to create large ones manually and will only need to make assets that can be placed such as trees and bushes. |
| Mobile version | The game will not be able to run on mobile, as its large variety of controls would be difficult to implement intuitively with only a touchscreen, and it would require all of the code to be made in a way that is compatible with mobile devices. |

Hardware and software requirements:

My project will be coded in java, using a windows pc, but users will be able to use any operating system that can run java.

Software

Users will need to be able to run java by installing a JDK. This is not typically included in the operating system; therefore, users will have to install one if they do not have it.

No other software is required to play the game, other than a graphical interface.

Hardware

The game will not include a mobile version as this would be too time consuming to code on a different platform, and the controls will be complex, so difficult to implement with only a touchscreen; therefore, the user must have a desktop or laptop computer (laptops might have less powerful performance and run slower).  
A keyboard and mouse/touchpad will be required to navigate menus (mouse) and to control the user’s movement in the game (keyboard).

The game may be unable to run quickly on low end computers, if the game contains too many ants at once which could use a large amount of memory to store their data. However, it is unlikely to cause issues on most systems, and I will try to reduce memory wastage where possible so that the user is able to have a smooth experience while using a less powerful client, or having multiple windows open simultaneously.

Computational methods:

I believe my project is very well suited to computational methods, as it is a game which would be too complex to run in any other way than with a computer, due to the scale of processing.

* Many hundreds or thousands of ants and other objects will be simulated at once, therefore a computer will be required to process their behaviour
* Abstraction will be necessary in this project to model the behaviour of ants without replicating them exactly which would take a significant amount of resources. For example, a predictable algorithm will be used to decide where they go instead of simulating their brain.
* OOP - The project is suitable for an object oriented language, since I can use inheritance to make many similar subclasses of ants, and instantiate them.
* Concurrency – all parts of the game will be taking place at once (unless it is paused), so it must be able to run in real time, requiring a reasonably powerful computer.
* Visualisation – A GUI will be used to display the gameplay in real time to the user, by rendering all of the relevant objects such as food and ants near to them, and a map.

Design

Structure diagram:



Stages of development:

1.Main menu: functional buttons to navigate to what the user wants to do e.g. settings, gameplay, saves

2. Settings menu: buttons to select a map, difficulty, or gameplay mode.

3.Initial map design and creating assets: Basic map designs for locations such as a forest, desert, or city as well as sprites for food, ants and pheromones

4.Player ants and movement controls: should be visible and controllable by the user

5. Food spawning: Food should spawn in random locations on the map (required to test collecting food in next step)

6. Additional ant behaviour/controls: Ability to use pheromones and collect food.

7. Enemy ants (essential) and different variants (desirable): These should be easier to develop with the player ants as a starting point, but will require logic to make decisions automatically. Player ants will use the same logic when not actively controlled. Variants could include fighters, gatherers and a queen ant.

8. Population system: Population should grow when the player has enough resources and passively shrink at a slow rate otherwise. This will require food and ants to be functional first for testing.

9. Check for win: If all enemy ants are eliminated, then the player will win, and the game will end.

10. Sandbox mode: Player has the ability to spawn in resources or ants, and can experiment with the game and different strategies.

11. Natural disasters (optional): Random events to increase difficulty, such as rain or a predator. I would like to develop this before scoring so that it can give bonus score for surviving them.

12. Scoring: A system to score the player’s run based on factors such as the time played, the size of their colony and the size of the enemy colony. Should be developed near the end so that all finished features can have an impact on the final score.

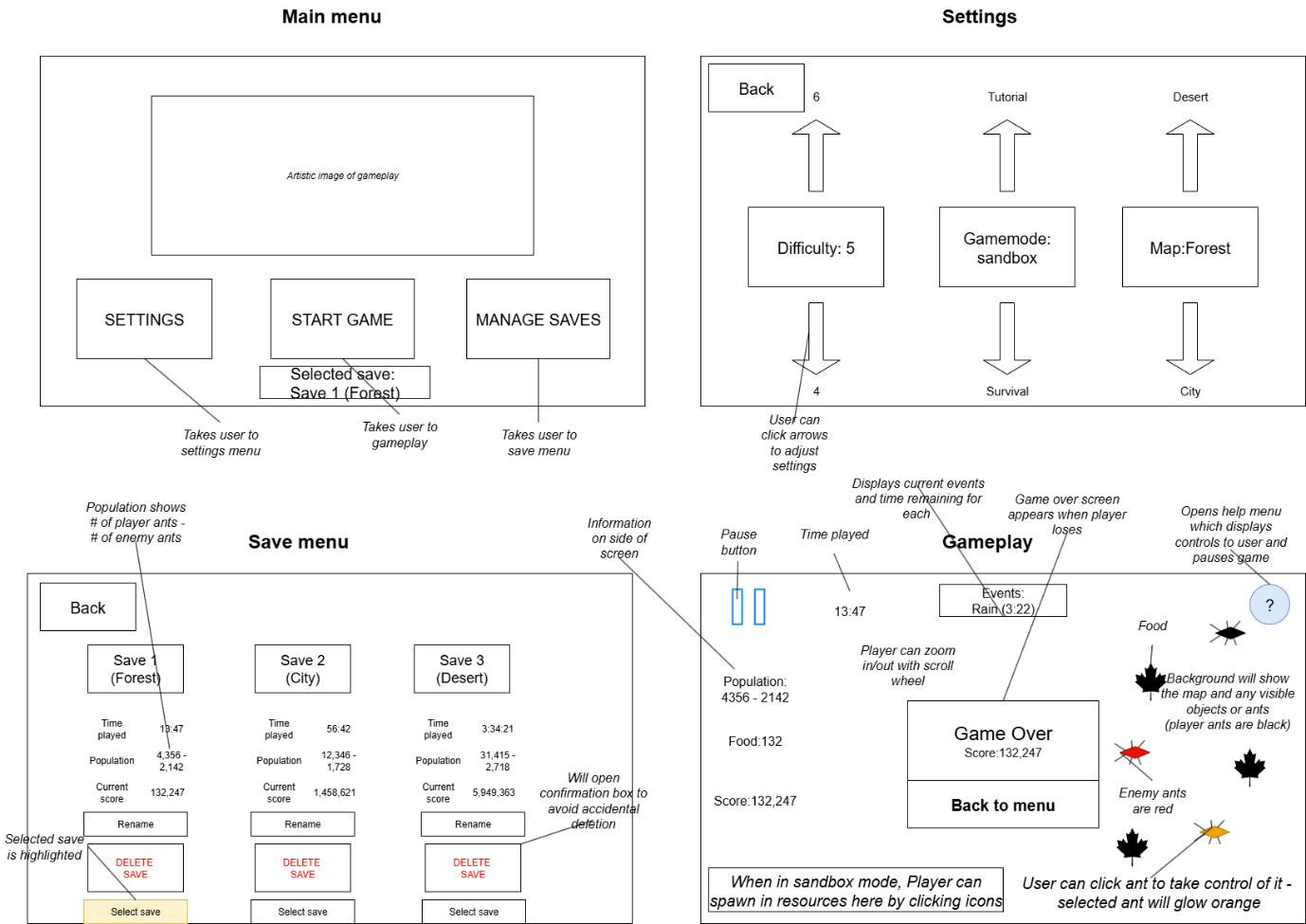
13. Tutorial (optional): This feature is not required and should cover all the developed features, so will be created at the end if there is enough time.

14. Saves (optional): This is likely to be difficult to implement, and is not required for the project, so I will complete it if there is enough time at the end

Test Plans

|  |  |  |  |
| --- | --- | --- | --- |
| Stage | Feature tested | Test data | Expected result |
| 1 | Main menu buttons | Press the buttons for settings, gameplay and save menu | The user will be directed to the correct screen |
| 2 | Settings menu – adjustable difficulty, map and game mode | Use the arrows for each feature | The selected setting will be changed |
| 2 | Settings menu – adjustable difficulty, map and game mode | Keep pressing the same arrow | The setting should reach a final state and the arrow will be grayed out |
| 4 | Controlling player ants | Select an ant and use WASD keys to move | The ant should move in the direction that the player chooses |
| 5 | Food spawning | Run the game and see where food appears | It spawns in a random location which is different each time the game runs |
| 6 | Placing Pheromones | Select an ant and use the associated key for pheromones (spacebar) | A pheromone will be placed at the ant’s location |
| 6 | Pheromones affecting behaviour | Place a pheromone near to ants | They should follow or move away from it depending on the type of pheromone |
| 6 | Collecting food | Control an ant to walk towards a piece of food | The food will disappear and the players food variable should increment |
| 7 | Automatic ant behaviour | Spawn in an enemy ant and a player ant | They should both automatically act to collect food and wander around. If the player takes control of their ant it will stop moving on its own. |
| 7 | Combat | Place an enemy and player ant near to each other | They should fight and one of the ants will be killed |
| 7 | Different variants of ant | Spawn in one ant of each variant | They should all behave more effectively at their specific task e.g. fighters will be more likely to win combat and food gatherers can move faster |
| 8 | Population growth | Give the player a large amount of food | Their ant population should grow, and some food will be consumed – new ants should appear on the map |
| 8 | Population decay | Make the player run out of food | Their population should slowly shrink – ants will disappear from the map |
| 9 | Check for win | Kill all enemy ants | The game should end and display a victory screen |
| 10 | Sandbox mode | Select sandbox mode in settings and run the game | The player should have a functional menu to spawn in resources or ants |
| 11 | Natural disasters | Run the game for a while | Eventually a random event will occur which influences the gameplay |
| 12 | Scoring | Play a run of the game | At the end, the player should be given a score which is calculated using their stats |
| 13 | Tutorial | Select tutorial mode in settings and run the game | The player should be shown information on how the gameplay works and the controls |
| 14 | Creating saves | Pause during a game and save and exit | A save should be created with all of the current data |
| 14 | Loading saves | Select a previously made save and run the game | The game should continue from the point where it was saved initially |
| 14 | Renaming saves | Press the rename button and enter text for the name of the save | The save should be given a name which is displayed to the user in future |

Gui Design:



Gui designs for 4 main screens: menu, settings, saving, and gameplay

I decided to display the selected save in the main menu, so that the user can check it is the correct one immediately, without having to navigate further. This will save them time as they can just load the game and play if it is correct; although they will need to visit the save menu to select a different one.

Arrows are used for the settings, so that the user can easily cycle through the options. The difficulty arrow will be grayed when it is equal to 1 or 10, as these are the lowest and highest values, so the user should not be able to use the arrows further.

The population will always be displayed in the same format for consistency,

[player ants - enemy ants]

This will provide the player with the relevant population information for both colonies in a simple format.

During gameplay, ants will be given different colours based on their state, so that the user can identify them.

Black – player ant

Red – enemy ant

Orange – player ant that is being controlled by user

Information is displayed around the edge of the gameplay screen so that it is not distracting, but can still be easily checked by the user.

Iterative development

Data dictionary:

Game (stored statically in Main) – Stores code to run the game and menus

|  |  |  |  |
| --- | --- | --- | --- |
| Variable name | Data Type | Example values | Explanation |
| screen | int | 0 (Main menu),  1 (Settings),  2 (Gameplay),  3 (Save menu) | The menu that the user is currently on |
| frames | Arraylist<JFrame> | Main menu, Settings menu, Gameplay, Save menu | Stores the 4 menus so that they can be accessed by the function getFrames() |
| difficulty | int | 0 to 9 inclusive | Difficulty of the game – affects features such as food spawning |
| gamemode | int | 0 (survival)  1(sandbox)  2(tutorial) | The gameplay mode that a new game will use |
| map | int | 0 (forest)  1 (desert)  2 (city) | The map that will be used |

GameplayGrid – Panel used to display gameplay

|  |  |  |  |
| --- | --- | --- | --- |
| tiles | Arraylist<JPanel> | Ant  Empty tile  Pheromone  Food | stores all the tiles of the game, reading left to right in each row.  To access a tile at (x,y), the index is width\*y+x |
| width | int | Initialised at 10 and can increase to 20 before being capped. | The width of the grid |
| height | int | Initialised at 10 and can increase to 20 before being capped. | The height of the grid |
| CornerX | int | 0,1,2 …  (width-columns-1) | The x coordinate of the bottom left corner of the grid, used to change which area is displayed |
| CornerY | int | 0,1,2… (height-rows-1) | The y coordinate of the bottom left corner of the grid |
| menu | GameMenu | GameMenu | The GameMenu which contains the grid |
| (local scope)  rows | int | 10-20 inclusive | Number of rows that the grid displays |
| (local scope)  columns | int | 10-20 inclusive | Number of columns that the grid displays |

GameMenu – Menu used to display gameplay as well as pause menu and information

|  |  |  |  |
| --- | --- | --- | --- |
| game | Game | Game | The game class |
| ants | Arraylist<Ant> | Ant1, Ant2 | The ants currently in the grid |
| newants | Arraylist<Ant> | Ant3, Ant4, Ant5 | Ants to be added to the grid when it expands. They cannot be added instantly as the game is looping through ants for their movement, so updating the array throws an error |
| grid | GameplayGrid | GameplayGrid | The grid used to display tiles |
| food | int | 0,1,2 … | The amount of food that the player currently has. Increments when a food tile is collected, and passively decreases |

Ant – superclass of player and enemy ants

|  |  |  |  |
| --- | --- | --- | --- |
| x | int | 0,1,2 … width-1 | x coordinate of the ant |
| y | int | 0,1,2 … height-1 | y coordinate of the ant |
| playing | Boolean | false, true | Whether the ant is controlled by the player |
| menu | GameMenu | GameMenu | The GameMenu – needed to use methods such as addAnt() to keep track of all ants. |
| grid | GameplayGrid | GameplayGrid | The grid used to display ants and other tiles |

Food – A collectable tile which will increment the food variable by 1

(No attributes)

Pheromone – A tile which can be placed to communicate with other ants

|  |  |  |  |
| --- | --- | --- | --- |
| edge1 | int | 0-3 inclusive | The edge which the pheromone will start from.  0 = top  1=right  2=bottom  3=left |
| edge2 | int | 0-3 inclusive  cannot be equal to edge1  Edges will automatically swap so that edge1 is always greatest | The edge which the pheromone will finish at. |
| image | ImageIcon | 6 possible cases:  new ImageIcon(file)  where file can be:  PheromoneTopRight  Pheromone  PheromoneTopLeft  PheromoneBottomRight  PheromoneHorizontal  PheromoneBottomLeft | The image that the grid will display for the pheromone. Varies depending on orientation |

Stage 1 - Main menu

The main menu should have buttons to redirect the user to one of 3 other screens: Settings, gameplay or saving

A diagram of a computer program

AI-generated content may be incorrect.

Created a Main menu screen with 3 buttons

A screen shot of a computer

AI-generated content may be incorrect.

Added functionality to the settings button with print function as placeholder code

A screen shot of a computer program

AI-generated content may be incorrect.

Added image placeholder to match GUI layoutA screenshot of a computer

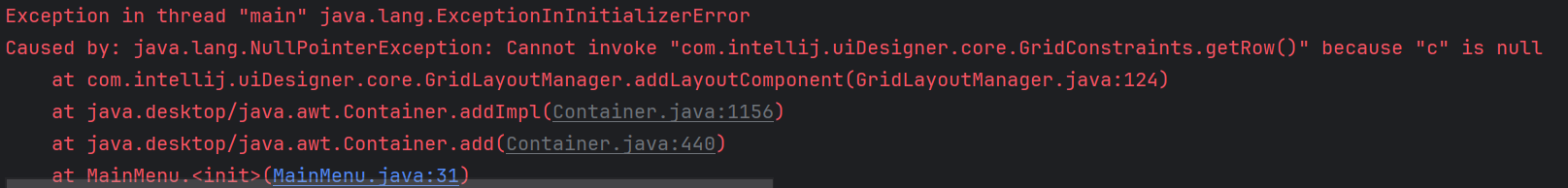
AI-generated content may be incorrect.

Errors:

A screen shot of a computer

AI-generated content may be incorrect.

Error trying to read image file, as exception is thrown. Fixed using a try/catch statement



Encountered an error trying to add an image to a JPanel.

Fixed by switching its layout manager to a FlowLayout

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Stage | Feature tested | Test data | Expected result | Actual result |
| 1 | Main menu buttons | Press the buttons for settings, gameplay and save menu | The user will be directed to the correct screen | Pass  User is directed to correct screen |

Stage 2 - Settings Menu

Pseudocode

int difficulty = game.getDifficulty()

int mode = game.getMode()

int map = game.getMap()

Arraylist Arrows= new Arraylist(Button)

difficultyUp = New Button(“difficulty”)

Arrows.add difficultyUp

gameModeUp= New Button(“GameMode”)

Arrows.add gameModeUp

mapUp= New Button(“Map”)

Arrows.add mapUp

difficultyDown= New Button(“Difficulty”)

Arrows.add difficultyDown

gameModeDown= New Button(“GameMode”)

Arrows.add gameModeDown

mapDown= New Button(“Map”)

Arrows.add mapDown

method update():

FOR button in Arrows:

Button.setGray(false)

NEXT button

if(difficulty=9):difficultyUp.setGray(true)

if(difficulty=0): difficultyDown.setGray(true)

if(map=2): mapUp.setGray(true)

if(map=0): mapDown.setGray(true)

if(mode=2):gameModeUp.setGray(true)

if(mode=0):gameModeDown.setGray(true)

Encountered an error trying to update the settings menu in its constructor. The update function required information from the game, however the game had not yet been constructed, as the settings menu is instantiated within the constructor.

A black screen with colorful lights

AI-generated content may be incorrect.

This was fixed by passing the game as a parameter to the settingsMenu constructor rather than accessing it from a static context. I made a similar change to all menus to avoid the same problem.

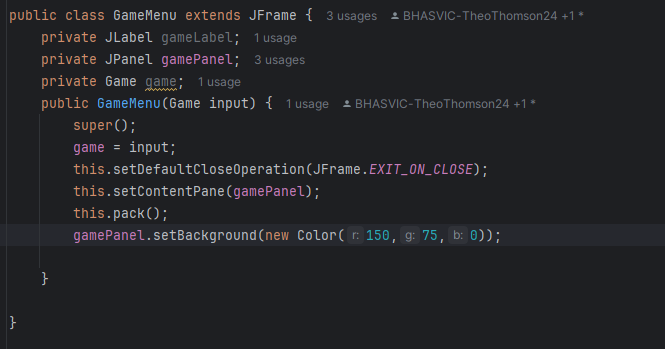
A screen shot of a computer

AI-generated content may be incorrect.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Stage | Feature | Test | Expected result | Actual result |
| 2 | Settings menu – adjustable difficulty, map and game mode | Use the arrows for each feature | The selected setting will be changed | Pass  The value is changed and the new result is displayed to the user |
| 2 | Settings menu – adjustable difficulty, map and game mode | Keep pressing the same arrow | The setting should reach a final state and the arrow will be grayed out | Pass  The arrow turns gray at the limiting value, and reverts to black when the setting is changed back |

Stage 3 – Map design

I decided to use a brown colour for the background to look like dirt, specified with the rgb values: 150,75,0



I implemented a pause feature when the esc key is pressed

A computer screen shot of a program code

AI-generated content may be incorrect.

Initialising the gamePlay grid as 10x10A computer screen shot of a program code

AI-generated content may be incorrect.

I created a class for each type of tile that the game will use: Ants, food, pheromones and empty tiles.

A screenshot of a computer

AI-generated content may be incorrect.

The Ant class inherits from JButton, as it will need to detect clicks once gameplay is added

A screen shot of a computer screen

AI-generated content may be incorrect.

The pheromone class has 6 possible assets as it leads between two distinct edges out of the four possible, and direction can be ignored. The edge data is first validated so that they are both between 0 and 3, not equal, and edge 1 is lowest, then the correct asset is found to display

A computer screen with text and images

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

Screenshot of what game menu looks like with pause menu open and example tiles:A green leaf on a brown surface

AI-generated content may be incorrect.

Stage 4 – Ant movement

Move function – sets current tile to empty then sets new tile to the ant after validating to ensure position is allowed.

A diagram of a computer

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

Ants could move into other ants or food, erasing them from the screen, so I added a check to fix this

A screen shot of a computer program

AI-generated content may be incorrect.

Adding control of multiple ants, toggled by control key

A screen shot of a computer program

AI-generated content may be incorrect.

A computer screen shot of a program code

AI-generated content may be incorrect.

I then implemented an extendable grid, so that new tiles are generated when ants move past the edge.

A computer screen shot of a program code

AI-generated content may be incorrect.

A computer screen shot of a code

AI-generated content may be incorrect.

Finally, I capped the size of the grid to 20x20 tiles and implemented a subroutine to move the corner, so that the game can scroll and will not run out of space on the screen.

A computer screen shot of a program code

AI-generated content may be incorrect.

The movement now works as intended, but throws an error when trying to move past the top left corner.

A black screen with text

AI-generated content may be incorrect.

This was fixed by adding a try catch loop to the getTile() function