

Requirements

Requirements Elicitation Procedure

To determine the requirements for our system we conducted two interviews with our major stakeholders, they went as follows:

Stakeholder Interview 1:

This interview was an initial discussion with our stakeholders to clarify points from the brief and to present our ideas from the initial brainstorm. The transcript can be found at [1]. Following this interview we elicited some key requirements and constructed two paper prototypes to quiz our stakeholders on features we were unsure of.

Stakeholder Interview 2:

This interview involved our demo-ing of a hotseat UI and a side by side UI to determine which method of multiplayer was preferred. The prototypes allowed us to adapt the UI on the fly and come up with a design solution that suited the stakeholders and audience

Alongside our stakeholder interviews we conducted research into resource management games [7] to feed into our ideas for UI and the game experience and mechanics overall. Focussing on determining a list of positive and negative points for each game we refined what features are enjoyed by the community and what features and factors make games less enjoyable to play. We also conducted a questionnaire [4] to communicate with our audience and get a feel for their experience of videogames and resource management games. Finally we did brief research into the law of supply and the law of demand [5,6] to allow the market to represent a real world market in a basic way.

Presentation of Requirements

The main requirements have been listed in bold and numbered. Each main requirement has sub-requirements which provide finer details and justifications for that requirement. Any Numbers in [] at the end of a justification for a requirement refer to the bibliography.

Research was carried out to determine how our requirements should be presented. The format is loosely based on the IEEE standard [8]. Originally we adopted a tabular format for the requirements, this was later replaced due to the amount of space that the structure of the format needed. The list format is more appropriate with our space constraints but still allows for a convenient numbering system.

Each requirement's associated risks were also discussed and considered thoroughly: these risks, along with alternative requirements that can be implemented should they present issues, are described on the project's website and can be accessed through the site's contents' page.

User Requirements

1. The game will not be convoluted and complex to play

- a. The game's element of challenge should be limited and aimed towards entry level players rather than masters of the game
 - i. Both our user and audience enjoy playing games to relax [1.2.1][4.3.1]
 - ii. It will need to be suitable for play by prospective students and other members of the public [3]
 - iii. The game should teach via gameplay and not have a tutorial or manual as a barrier to play the game, otherwise it won't be suitable for the intended environment. [1.8.1]
 - iv. The computer opponent should be easy to win against [1.1.1.4]
- b. The timeframe for the game should be 10-20 minutes
 - i. Players will only be allowed to play the game in short sessions, so they must be able to complete it [1.4.1.2]
 - ii. Customer does not enjoy long games [1.7]
- c. The UI should be clear and beginner friendly
 - i. Text and UI graphics should be a reasonable size [7.4]
 - ii. The UI should link to other pop up screens for information like market prices and roboticon upgrades to promote a simple and structured experience [2.4]
 - iii. Only the current player's resource information should be visible [2.8]
 - iv. The statistics of tiles should be hidden until they can be acquired, so the user can apply intuition to their strategy [2.9]

2. The game should have a graphical user interface which presents a map of the University.

- a. There must be at least three visibly identifiable landmarks
 - i. For university marketing purposes [3]
 - ii. Buildings may have unique bonuses for the player that occupies them [1.1.1.2, 1.1.1.3]
- b. The map should be subdivided into 16 isometric square plots of land (4 by 4) [1.8.2]
 - i. Plots may partly or totally include buildings
 - ii. Plots do not need to be identical in terms of resource allocation and buildings
 - iii. Square plots are favoured by the customer; they make the game easier to play for users [2.1]
- c. Plots must be all unallocated at the beginning of a game [3]

3. The game should support two opposing players (one of which may be computer-controlled) [1.4.1]

- a. The user must be able to clearly distinguish between each player and their plots [1.5.1]
 - i. Each player should have different colour schemes to make the ownership of tiles clear [2.4.1]
- a. Players should take alternating turns to play in a hot-seat style of gameplay [1.4.1]

4. The game should take the general form of a resource-management simulator

- a. The game will have a scoring system with final score based off of resource amounts at the end of the game [3]

5. The player should have individual reserves of money and three different resources

- a. The resources are to be: Ore, Energy and Food [3]
- b. Resources can be acquired, spent, sold, and bought (with players' money) [3]
- c. Each player shall start with a small amount of money [3]
- d. Player can win or lose money via "gambling" [3]

6. The game should support random effects that come into play during a round [3]

- a. These effects should vary from low impact effects to larger more meaningful displacements in gameplay. Events should never be 'unfair' and only used judiciously. [1.6.1, 1.6.1.1]

7. The game should have a market which will be used to buy and sell resources and equipment

- a. The market will portray "Supply and demand economics" as described in [5.6]. [3]
 - i. The higher the amount of a resource in the market, the lower the price players will be able to sell that resource at.

- b. The market will initially have 16 units of Food, 16 units of Energy and no Ore.[3]
- c. The market will have 12 Roboticons (robotic assistants) at the start[3]
- d. The market will have a bar where money can be won or lost via gambling[3]
- 8. **The player should be able to purchase a robot ("Roboticon") to produce resources[3]**
 - a. Resources cannot be produced without Roboticons.[3]
 - b. Players can upgrade their Roboticon according to an upgrade path.[1.1.1.2, 1.1.1.3, 4.5.1.2, 2d)]
 - c. Roboticons are assigned to a tile.[3]
- 9. **The game should support rounds where each player has one turn. The round will have 5 phases: [3]**
 - a. Acquisition: each player may acquire a previously unoccupied plot of land.
 - i. Will not include random elements [1.6.1]
 - ii. Only plots adjacent to plots the player currently owns may be acquired [2.3]
 - b. Purchase and Customisation: each player may purchase a Roboticon and customise them to support food, energy or ore production.
 - i. Customisation will cost money. [3]
 - ii. Phase will be time limited [3]
 - iii. Will not include random elements [1.6.1]
 - c. Installation of Roboticon: the customised Roboticon may be installed on a plot of land owned by the player.
 - i. Phase will be time limited [3]
 - ii. Will not include random elements [1.6.1]
 - d. Production: the players' colonies produce (or fails to produce) Food, Energy and Ore [3]
 - i. Suitable location for random effects applied judiciously [1.6.1]
 - e. Auction: resources can be sold at an auction. Players may sell their resources to the other player or to the market at a fixed price. [3]
 - i. It will not be necessary for the player to sell or buy resources at each auction phase
- 10. **The game should end at the end of the round in which the last plot of land has been allocated**
 - a. At the end of a game, the player with the highest score(4.a) is pronounced the winner [3]
- 11. **The player can get information about their owned plots of land and roboticons installed on them**
 - a. The player should be able to get information about resources that it is generating [1.5.1]
 - b. The player should be able to see the type of roboticon, it's current level and the number of resources that it produces [1.5.1]
 - i. The player can upgrade the roboticon using their money. The upgrade will enhance roboticon allowing it to generate more resources [1.1.1.3]
- 12. **The game should provide a brief background story**
 - a. Customer and audience enjoy playing games with a narrative [1.3.1] [4.4.1]

System Requirements

- 13. **The main menu should offer the following options..**
 - a. **Start Game**
 - i. This should bring the player to a menu where they can choose their college, whether they would like to play against a computer-controlled opponent or a human being, and an AI difficulty level (if applicable)
 - ii. Information regarding each college's unique attributes should be shown on this screen
 - iii. If the player chooses to play against someone else, another area of the UI will open up to allow for the other player to select their college
 - iv. Both players will not be able to select the same college to play as
 - b. **How to Play**

- i. *Brings up a flipbook-style sequence of pages that introduces the game's story/context to players and teaches them how to play it*
- ii. *This should be concise and primarily convey information graphically [1.8.1.1]*

c. Leaderboard

- i. *Displays a list of the top 5 overall scores in the game and a simplistic name attached to each one*

14. The game's map, which will be assembled manually and not through random generation, will be grid-like in nature and presented to players from a non-isometric bird's-eye perspective

- a. **Players should be allowed to acquire tiles for their chosen colleges as the game goes on**
 - i. *Only tiles that are adjacent to players' existing acquisitions will be acquirable*
- b. **Tiles on the map should include:**
 - i. *A value for Ore, Water and Food resources on that tile (possibly 0)*
 - 1. *Values unseen before acquiring*
 - ii. *At least three University departments and landmarks should be clearly visible [3]*
 - iii. *In accordance with 1.c.iv. The tiles resources should be representative of the graphic on them. I.e. A Lake will have a fair amount of water etc.*

15. The player will interface with the game via the use of a mouse and keyboard

- a. *Our game research showed that the mouse and keyboard combination is the much preferred method for playing any kind of strategy game due to the easier navigation and additional functionality [4.7.1]*
- b. *There will be keyboard shortcuts to access various menus quickly.*

16. A heads-up display (HUD) should display the following information whilst playing

- a. *Counters for current player's resources*
- b. *A button to access the market to, only show robot inventory in the market until auction phase [2.5]*
- c. *Button for the current user to end their turn*
- d. *A button in the corner which leads to the pause menu (19)*

17. Players should be able to select tiles using the control scheme described in point [5] and plant available Roboticons onto them through tool-tip menus bound to each tile

- a. *A tile's tooltip will describe/present the following (provided they own it)*
 - i. *The type of tile that it is; The type of resources that can be gained from it; what kind of Roboticon is working on it (if one's working on the tile at all); options to assign or withdraw Roboticons from each tile and an option to upgrade the Roboticon working on the tile*

18. Each tile should be visually distinguishable and follow a consistent art-style

19. Players should be able to pause the game at any time while it ensues

- a. *Given that turns are timed, players should be able to attend to conversations or other tasks without it affecting the game.*

20. The game needs to be playable on university computers

- a. *It should not require a dedicated graphics accelerator other than those included with the processors powering the university's PCs*
- b. *It should not consume more than 4GB of RAM under any circumstances*
- c. *The combined file-size of the game and all of its resources should be small enough to fit on the university's servers comfortably*
- d. *Windows 10 should be supported as these are the operating systems that the university computers run*
- e. *It should not exceed a resolution of [1680 × 1050], as this is the resolution at which the monitors in the university's CS department operate*