**Adam Fraser AH CS project**

**Multi-screen remote controlled asteroids**

**Project proposal**

**Perceived need for system**

Currently there are no systems I am aware of that allow as many people as are available to play the classic arcade game asteroids across a large “arena” much less control it with their phones. this system will allow users to use whatever compatible devices they have to hand as a potentially giant arena to competitively play asteroids.

**End-users**

End users will likely be relatively experienced with computer systems due to the hassle of setting up the server and database. They will probably play the game in large groups since it requires many devices and will be most fun that way.

**System overview**

this system is a modern take on the classic arcade game “asteroids”. While in the original users generally played alone only interacting with other users through a leader board this version will …

* Allow users to engage in multiplayer games across a large arena comprised of many screens
* Give users the ability to control their “ship” from any device with the latest browser technology but preferably a mobile / touch device
* Add lots of interesting game mechanics like powerups (e.g. speed boosts) and the ability to setup mining colonies on asteroids (give a certain number of points per cycle)

**Key features / objectives**

* The game should be responsive with latency low enough over a local network to be unnoticeable
* There should be an intuitive way to initiate a game with other users
* There should be a simple to use and scalable leader board
* The game should display the score as it is being played
* The game should include interesting features like power ups, point boosts and obstacles
* The server should have a useful and intuitive dashboard showing such information as current number of games, total games and server uptime
* The server should keep extensive logs in an external persistent file
* The server should make use of a database to store scores
* The controller interface should be optimized for mobile devices
* The transition of entities from one screen to annother should be seamless or at least only involve a few ms of delay
* The asteroids automatically generated for the game should be “interesting” shapes and not just simple circles
* It should be easy for each player to follow their “ship” around the arena
* When a new powerup appears users should be alerted with a sound since it may be out of their sight

**Feasibility**

**technological**

* There may be significant challenges keeping the game in sync across all screens
* For a system that involves 4 distinct parts (displays, controllers, server and database) it will be challenging to ensure that unexpected behaviour (e.g. controllers quitting) does not lead to total system failure or damage to persistent systems like logs and the database
* It’s been a while since I’ve worked with socket.io or mongodb but I’m confident I can revise the necessary skills in a short timeframe
* The potential computational burden is massive due to the number of individual entities spread out over a massive area so a way to distribute the load between devices (likely each individual display) will need to be implemented
* The range of mobile devices is large and so developing a consistent and friendly controller interface will be difficult
* Any system that acts as the server will need to be able to respond quickly and handle all the clients connected to it. It will also need to have the neccesary packages installed for the node server and be able to access the database that stores the leader board

**Financial**

* there are potential financial concerns since to run the arena effectively multiple devices with large screens are neccesary however computers are available
* for the game to be played effectively multiple controllers should be used this could be a concern however due to the controller being based in a browser any device can be used not just phones

**Legal**

* Any images used will be created using paint.net, Microsoft paint or be from royalty-free sources like Wikipedia
* Asteroids is not an original idea so a direct copy would be a legal problem but this project will be a significant alteration on the original version so there should be no concerns
* Since user data will be stored in the form of a leader board and this could potentially be used to identify that person there may be concerns relating to the data protection act
* Since there is nothing to stop users entering obscene or offensive names they will need to be informed of their responsibilities in relation to the communications act i.e. not to use obscene or offensive names

**Time**

* There are constraints in place to keep the project on track and I am confident it can be finished on time

**Timescale (assuming 5 hours work a week)**

**Estimates of time needed**

|  |  |  |
| --- | --- | --- |
| Task | time | Weeks done |
| Analysis | 5 hours | 1 |
|  |  |  |
| Design | 21 hours | 6 |
| *Arena-backend interface* | *4 hours* |  |
| *arena* | *5 hours* |  |
| *backend* | *5 hours* |  |
| *controller* | *3 hours* |  |
| *HCI* | *2 hours* |  |
| *Game mechanics* | *2 hours* |  |
|  |  |  |
| Revising and learning | 5 hours | 7 |
| *mongodb* | *1 hour* |  |
| *Socket.io* | *2 hours* |  |
| *express* | *1 hour* |  |
| *templating* | *30 min* |  |
| *Optimising for mobile* | *30 min* |  |
|  |  |  |
| Implementation | 25 hours | 12 |
| *backend* | *7 hours* |  |
| *arena* | *8 hours* |  |
| *controller* | *4 hours* |  |
| *Connecting everything* | *3 hours* |  |
| *Interesting mechanics* | *3 hours* |  |
|  |  |  |
| Testing | <15 hours | 15 |
|  |  |  |
| Evaluation | ~ 7 hours | 16 |
|  |  |  |
| Total time | ~78 hours | 16 |