**Adam fraser AH CS log**

**27/08/2018**

* **­­**decided to change project from multi-player pong to multi-player asteroids where the game is spread out across multiple screens attached to different devices and each player controls their “ship” with a mobile device through a browser interface. The reasons for this are 1. I don’t think the old idea was original enough (or at all) 2. The old idea wouldn’t have taken much time 3. After finding out about what other people have done for their AH CS projects online I don’t feel like my idea was anywhere near good enough and most importantly 4. The new idea is so much cooler 😄
* Wrote down the main objectives of the project to have clear criteria for success and goals to aim for. This also helped to clarify some of the biggest challenges (latency, handling disconnects etc.)
* Looked into the feasibility of the project (economic, legal, technological, time) after this I feel more confident about being able to complete the project and do it on time since there are no immediate reasons for it to be infeasible
* Wrote out a timescale for the project (stages and how long each stage will take) after this I am very confident the project will be completed on time and also have a much clearer idea in my head of what needs to be done and in what order
* Setup a github repo to store the paperwork for the project and the project itself. This is so I can easily see where things went wrong when they inevitably do

**17/09/2018**

* Started work on Gantt chart.
* this has lead to a revaluation of how long each step will take and what it will involve
* it also gave me a clearer idea of what modules will go where and how they will interact

**27/08/2018**

* thought about asteroid intersection / collision detection and how it could be implemented so as to be immersive and still efficient. Here is my thought process
  1. thought about a simple radius detection system whereby if the radius of the asteroid + the radius of the radius of the object being checked for intersection are sufficient to bridge the distance then a collision state is triggered. I knew this would be un-realistic since asteroids, players and plasma bolts will not be perfect (or even close to) circles.
  2. Then I thought about a more immersive algorithm based on adjacency where there are a few pixels in each direction like a buffer zone where if the player intersects there will be a collision event. I realised this would be extremely computationally intensive and that the maths involved would give me a headache
     + I also thought about a bounding box-adjacency hybrid system where a more efficient bounding box algorithm is used to detect if an entity is close then switch over to the more intensive adjacency algorithm but this still requires a lot of difficult maths
  3. Taking inspiration from how 2D games (and some 3D games) handle collision with bounding boxes I now plan to create an algorithm to generate detailed bounding boxes for asteroids and players
     + I want to create a outer-inner bounding box system where the more intensive multiple small bounding boxes are only used once an entity intersects the larger bounding box

OOOH IDEA A SHIP SPEC PAGE?