Orbit project document

-Aim

Create a casual/social game that can be played easily, used to fill time etc. Completed using Unity with c# scripting language, based on prototype developed for comp1917 something awesome

-Keys

* Graphical
  + Strength of orbit represented by arrows around planet with light circle, arrows initially undergo a startup animation then rotate around planet slowly, counter to the planets standard rotation.
  + Based on standard, clear cut, 2D graphics.
  + Backdrop possibly an animated star system, twinkling stars everynow and then
  + Menu simple, Daily level (need name), Play (continues from last completed level), level selection.
  + On level completion quick briefing, star assignment, Large retry button, next to large next button both beneath summary which incl fuel used, orbits used.
  + Planets magnify into proportion on level start
  + Double check legality of current assets in use.
  + Camera moves to points in relation of planet with most gravitational affect
* Mechanical
  + Complications involving moving planets/orbiting each other,”heavier” planets, refuel stations, alien ships (still emiting an orbit)
  + No upgrade system etc, want to keep it simple for casual
  + Standard format of using planets gravity/orbits to slingshot into a target
  + Limited fuel preventing simple completion
  + Daily levels received from server, constant for all users, best scores easily shared to social media sites.
  + Score determined by amount of fuel remaining, stars awarded for amount of fuel conserved in a bracket type system
  + Start docked with some stationary object which releases on first touch screen contact. Removes current problem of lack of knowledge
  + Need sound effects, along with basic background music, however not critical and needs to be playable with own music in background.
  + Must have quick replayability, very easy to restart if failed level
* Code
  + Based on standard code developed for comp1917
  + Use tag system for planets, develop other tags for different sized planets or bodies then assign to array for calcs
  + Develop gui for fuel return
  + Start with 30 preset levels
  + Daily levels pushed through server communication, sends data to be interpreted by application. However can also push into a monthly update, with 30 levels pre-assigned. Would solve issue of server communication/level download.
  + Datasheet for each level, not needed if monthly update
* Marketing
  + Daily levels as mentioned in mechanical, utilises social media spread, can also possibly link to platform specific leaderboards to match each day?
  + Run through afternight brand, publicise on Mist brand for initials, directive to afternight brand
  + Post updates etc to afternight, afternight not as important till later developments.
  + Look into templates for afternight logo
  + Trailer developed and deployed to afternigh youtube publisised by Mist