ComS 437 Asteroids 3D

In this project you will utilize 3D constructs to create a first person asteroids game. There is a list of required elements of the game that you must implement. You will also have a list of optional features to implement. You must implement at least 3 of these optional features. Note that a two-player game is an optional requirement.

Required Elements

1. The game must be written in C# and Monogame.
2. The universe is a sphere or other shaped contained space in which you, your opponent, asteroids, and other objects reside inside.

3. If the ship collides with the edge of the universe (the edge of the sphere) the ship bounces off. You may choose how the bounce works.

4. If an asteroid and ship collide, the ship is destroyed.

5. Asteroids come in at least 3 different sizes (and masses).

6. Your game must have at least two different models for each size asteroid.

7. Asteroids may not be a simple sphere for a model.

8. When 2 asteroids collide, there is a user settable switch that defines the probability that asteroids are damaged. If this is the case, the result is that each asteroid is decreased by 1 in size. If an asteroid of the smallest size collides, it is removed from the game.

9. When 2 asteroids collide, any resultant asteroids rebound (bounce) with appropriate direction and speed, based on collision velocity and mass.

10. A player has three ships to achieve the game goal.

11. In single player mode, the goal is to achieve the fastest time to destroy (removed from the game) 75% of the initial asteroids.

12. A ship has an infinite supply of photon torpedoes that it may use to blow up asteroids. However, a torpedo may only be fired at a maximum rate of 2 per second.

13. Torpedoes have a velocity vector and must be tracked by the game until they either collide with an asteroid or the edge of the universe.

14. When a torpedo collides with the edge of the universe, a brief explosion is displayed.

15. When a torpedo collides with any ship, the ship is destroyed and a brief explosion is displayed.

16. When a torpedo collides with an asteroid of any size, the asteroid is destroyed by the torpedo, a brief explosion is displayed.

17. The ship has jets that allow it to rotate to any orientation and a thruster that will accelerate it in the direction it is pointing. You may choose how to control these from either the keyboard or a game pad. Note that a thruster adds (angular) acceleration in the direction of the thrust, only while the thruster is burning.

18. Ships have an infinite amount of fuel.

19. The number of initial asteroids and the size (volume) of the universe is configurable at the start of the game. Note that the size of the universe need not be an exact number, it may allow the player to select between 1 to 5 different sizes, for example.

20. Asteroids are initialized with a random position, orientation, spin, and velocity.

21. You must use a skybox, or some other effect to display a "star field" in the background. (The edge of the universe is invisible.)

22. The ship is displayed from a "first-person" point of view (behind the ship view is OK) .

23. It is highly recommended that you use quaternions to manipulate orientations of objects. It makes things MUCH easier.

24. The game must include cool sounds for explosions, fired torpedoes, etc.

25. The elapsed time (for single player games) and the number of ships left for any players must be displayed appropriately.

Optional requirements. (Pick 3 to implement)

1. Implement a two-player mode using split screen.

2. Instead of a split screen, the game is networked between the two players on two different machines. (This single option may count as 2 options.)

3. Ships have a limited amount of fuel. Players may "dock" at fuel depots by running over them to receive more fuel. These depots are fixed in space unless hit my something that

start them moving. They have special protection so that they don't explode when hit and have stabilizers to slow them back to a motion free state.

4. If 3 is implemented, there may be fuel specials that appear at random times and may be run over for extra fuel. These specials disappear once used.

5. Provide ships with a limited number of shield uses that protect the ship during a collision or torpedo.

6. If 6 is implemented then implement shield specials similar to fuel specials.

7. Allow the mass of asteroids to pull ships and torpedoes toward them.

8. Implement black holes and white holes. If you enter a black hole, you will emerge from the white hole. If traveling through this type of wormhole, implement a probability of destruction of the ship. Also note that the black hole will also act as an attractor of your ship and the white hole will act to repel a ship.

9. Implement specials/power-ups. When a ship runs over a power-up, the ship is given a special ability for a period of time, or a limited resource. Example power-ups include better torpedoes (heat seeking for example), invisibility, auto pilot, nebula cloud, etc.

10. Allow co-op in "single player" mode by having one controller provide steering control and the other controller provide shot control.

11. Propose an option for approval and then implement.