Battle Hover Tanks Game

I. How to run it, and how to play it

To run it: From the command line, navigate to the folder called bin and enter the following -- java networking.server.GameServerTest.java. This will start the server for the game. Next, from the same bin folder, type the following -- java battleHoverTanks.Starter. This will start the game. It will first ask you for the server's ip address, the port number, your name, the option to play in windowed or fullscreen mode and allow you to choose the style and color of your tank avatar. After you hit 'OK', it will take a minute for the game to load. Once it does your view should be behind your avatar and you are ready to play.

To play it: You can use the A,W,S,D keys to move your tank left, forward, back, and right respectively. The U,H,J,K keys will move the orbit camera around your avatar. If you are using a control pad, the left joystick will move your tank and the right joystick will move the orbital camera. To fire missiles from your tank use the 'F' key or the right trigger on the control pad. The object of the game is to get more points than the other players before you die. To do this, you must hit enemy turrets and other players with your missiles. Every enemy turret hit you get is worth 12 points and hit on opposing players will get you 25 points. Getting hit by enemy turrets will lose you 2 health points and a hit from an enemy tank will do 4 points of damage to you. Once you run out of health your game is over and you must wait until all the other players but one has died. To exit the game at anytime, press the 'ESC' key or the 'back' button on the control pad.

II. How to use scripting

To use the scripting for this game, first navigate to the scripts folder. Inside open the javascript file named CreateWorld.js. There are three variables that can be changed. One for the skybox, one for the speed of the avatar and one for the color of the avatar. Comments in the file will tell you the parameters that are needed for each variable.

III. Available controls

This game can be played with either a keyboard or X-box controller.

IV. Machine requirements.

This game has only been tested on Windows machines, that are using NVidia graphics cards. The ability to use on other systems is unknown.

V. An explanation of how the game satisfies each of the requirements.

Skybox and Terrain. Game includes a skybox with multiple sets of textures. Terrain is implemented with a height map. Everything operates on and moves about the terrain.

Networked Multi-player. Game communicates avatars, turrets, missiles and endo of game. It also communicates changes through javascript updates.

Scripting. Game implements javascript for scripting. Changes to the skybox textures and changes to the ghost avatars can be made in the javascript file. As well as changing the speed of the avatar.

External models. Models used in the game are missiles, explosions, aliens, two tank types, which are made up of a cannon and tank base, and turrets, which are made of a turret cannon and turret base. Both tank models are selectable in the beginning with different colors available. All models were created in blender. Kevin created both pieces of the turret. Tim created the missiles, aliens, both tanks and cannon, and the explosion object.

Skinning. The alien, missiles, explosions and turrets have one skin. Both tanks have 6 skins

Events. The missiles have an impact event that causes an explosion sound when contacting another tank, turret, or terrain object.

HUD. HUD includes time for game, players name, score, and their health.

Heirarchical SceneGraph. Turrets are implemented with the base as part of the top group and turret cannon is part of a second group attached to the top group.

3D Sound. Sounds are attached to the tanks, ghost avatars, turrets, and missiles.

Animation. The alien model has two skeletal animations using keyframes. One animation is the alien walking with arms and legs moving back and forth. The second animation raises its arms above its head and wiggles them about with legs and head moving back and forth.

NPCs. Turrets are controlled by the server with an AI controller. They target the closest player and set a target lock on that player until they move out of range. When turret has that target in line of sight it will fire a missile every six seconds.

VI. The contributions of each team member, including who designed which model(s).

Contributions by Tim Johnson:
Most of the models were done by Tim, including the two tank models, the missile model, the explosion and the alien animated model. He also did the animation for the animated model. As well as the skinning for his models. He also did the skybox and terrain and got the skybox to be changeable through scripting. He also did the work for the missile and explosions.

Contributions by Kevin Richter:
He did two models, the turret base and the turret cannon. He made them a hierarchical scenegraph so that the cannon on the turret could be rotated. The skins for these two models were also done by him. He also did the majority of the networking, along with the AI for the npc enemy turrets, and the 3d sound for all the objects.

There were a lot of other parts not covered here that were worked on equally by both team members.