Cody Lanier
Cody Prior
CSc165 A3

StarGunner

i. To play, navigate to the stargunner/bin/ directory and type 'java a3.Starter'. The game is a top down shooter. The world auto-scrolls and your job is to take out as many UFOs as possible. The UFOs will come at you in waves and they can fire lasers at you. You have your agility and your missiles to protect yourself and rack up a nice score. Players earn a point for taking down a UFO, but lose 10 for being hit by an enemy bullet.
ii. Scripting is utilized via the initPlayer.js file. A player can edit the attributes of their ship (Size, fire rate, ship speed) to better serve gameplay.
iii. No known specific requirements. The game can be played with a USB controller or the keyboard. See page 5 for a list of all available controls.
iv. Again, no known hardware constraints.
v. Game Requirements
    a. Skybox and Terrain: We have a space Skybox and are using a red rock texture for our world terrain. The terrain has various heights which were implemented using a Hill Height Map. To better view these objects, you can pan the camera around with the WASD keys.
    b. Networked Multi-player: A prompt at the beginning of the game will ask the players how many players there will be, which will determine whether they are playing a regular single game or a multi-player networked game. At the time of submission, the game can successfully be played with 2 people on the same local network. A ghost avatar is created and displayed in each player's world that is moved around freely by the other player. The scrolling feature of the game waits for both players to be connected and ready, so that the world appears almost perfectly synchronized. The host acts as a server and client while the non-hosting player is strictly a client. Currently, each player's world has their own independent UFOs and T-Rex boss characters, which is not how we wanted to turn it in, but we ran out of time and it seems that the networking requirement is basically met as described in the instructions. Much of the actual gameplay came together during the last 48 hours available to code, which is why networking fell behind from when it was previously working well with what was currently implemented before that time.
    c. Scripting is implemented successfully and is described in part ii, above.
    d. Custom Models: We both made one model from scratch. Cody Prior made the UFOs and Cody Lanier made the Fighter Jet. An F-22 blueprint from

thebluepprints.com was used to aid in the shape of the model, but it does deviate from the blueprint quite a bit. There is also a T-Rex model in the game as a NPC boss character. This model was not made by either Cody or Cody, and credit is given in the 'Additional Notes' section of this document.

e. Skinning: Both our custom models have been appropriately skinned. We made seams, UV-unwrapped, painted, imported back into blender as material textures, and applied the painted file as a texture in the game code. This was done to the T-Rex as well, but unfortunately not to the standard that was intended. As of this writing, the monster's body, eyes, mouth, and teeth all use the exact same texture.
f. Events: Events are used to play sounds when the player fires, an event is passed to a listener containing the location of where the shot was fired which is used to play a laser sound.
g. The HUD displays time elapsed and the player's score.
h. Scene graphs are used in limited capacity for the background.
i. 3D Sound: The T-Rex, that was previously mentioned, has a growl associated with it and also a roar. Both of these sounds can be heard when the player is 200 z-units from the T-Rex's z-axis value, and the sound's volume is maximized when the player gets within 15 units. The goal was for it to be very ominous being able to hear the dinosaur from far away. Background music is also provided, but the volume must be turned way up to hear it. Sage's sound constructer volume property did not have any effect whatsoever on the volume of the background music, but it is certainly there and playing.
j. Animation: Animations were done on the T-Rex model. The T-Rex was brought in specifically for animations due to the fact that we couldn't figure out good animations to use with our jets and UFOs. If we had tried to do animations on the original models, the animations would not have been as comprehensive as they turned out. As previously stated, the model was downloaded from blenderswap.com and actually came fully rigged. It was over-rigged and OGRE did not like that, so we removed everything from the model and started with nothing but the model itself. Once fully rigged, there was a lot of trouble getting the vertices tied to the bones correctly. Anyway, after much frustration and almost giving up and going to a new model, we got the animations to successfully work. The T-Rex has 2 animations; the simple one happens when walking, in which just the legs are moving. The more spectacular looking one happens at the end of the walk, when the dinosaur roars. Its head neck and jaw are all involved with various rotations and angles. It actually turned out pretty nicely.

k. NPCs: The T-Rex is an NPC and so are the UFO's. Both use AI controllers and the T-Rex uses behavior trees. The UFOs are used for the main part of the gameplay. The T-Rex is pretty dumb and it just wanders around, walking in a random direction every 10 seconds. There is some limited code in place in the behavior tree to determine if one of the players is in range of the dinosaur. If so, there is also skeleton code for attacking a player. It just did not get fully implemented, as there were actual unfinished requirements or basic gameplay that needed to be finished before working on that. The UFOs attempt to track the player only if they are somewhere in front of them and they occasionally fire down at the player. If they lose sight of the player, they simply drift off into space.
l. Additional Notes:
    i. The game is a 3rd person camera view.
    ii. The player is prompted whether or not they want to play in windowed mode or full screen exclusive mode at the beginning of the game. NOTE: The pop up dialog box pops up behind all other active windows. If you do not know this, it can appear that the game doesn't work, but you must selected windowed of FSEM before it goes further.
    iii. IP address of the server is displayed on the host's prompt and the client must enter that address to play networked.
    iv. This should be considered a fat server TCP approach.
    v. The author of the T-Rex model is MatiasCNT and the link to it is http://www.blendswap.com/blends/view/73329.
    vi. Single player is the best way to currently play this game, because networked requires some more work. We hope to have it working much better for our demo.
    vii. All resources are included and are only accessed using relative paths, so the game should work fine on any common OS with Java.
    viii. The game has executed fine on RVR 5029 machines all semester, and the generic top-level package name will forever be 'a3.Starter'.
m. Extra Credit:
    i. The T-Rex has 2 separate animations and both use a combination of several bones. If more time is found between now and the demo, the walk animation will be improved to include a head bob, torso movement, and tail sway, but this extra credit requirement appears completed as is.

n. Contributions by each team member:

1. Skybox and Terrain: Cody Prior
2. Networked Multi-Player: Cody Lanier
3. Scripting: Cody Lanier & Cody Prior
4. Custom Models & Skins:
    - Fighter Jet: Cody Lanier
    - UFO: Cody Prior
5. Events: Cody Prior
6. HUD: Cody Prior
7. Heirarchical Scenegraphs: Cody Prior
8. 3D Sound: Cody Lanier & Cody Prior
9. Rigging & Animations: Cody Lanier
10. AI & NPCs:
    a. T-Rex: Cody Lanier
    b. UFOs: Cody Prior
11. Other Gameplay Aspects: Mostly Cody Prior
12. Documentation: Mostly Cody Lanier

Game Controls

COMMAND	BUTTON	KEY
Zoom Out		S
Zoom In		W
Pan Left		A
Pan Right		D
Move Forward	Left Stick Forward	Up arrow
Move Backward	Left Stick Backward	Back arrow
Move Left	Left Stick Left	Left arrow
Move Right	Left Stick Right	Right arrow
Fire	A	Z
Look Up	Right Stick Up	
Look Down	Right Stick Down	
Look Left	Right Stick Left	
Look Right	Right Stick Right	