CST-320 Technical Report - HCI Project Part 2

Period (10/25/21-11/12/21)

Student Name: Diego Guerra, Ryan Scott, Andrew Esch

Faculty Name: Isac Artzi

Project Topic: 5

Current task(s) (refer to the tasks listed in LoudCloud and/or instructor directions; detailed bullet list):

- Objective: Students will design and test interactions that are driven by mathematical and/or scientific concepts.
- From this list, here is current task deliverables for this CLC assignment:
    - Complete the technical report by demonstrating the current team progress for the Space Trooper project as it relates to mathematical and/or scientific concepts.
    - Explain the current progress for this week in the Space Trooper unity project by demonstrating screenshot, code, and project examples.
    - Film and present on the technical report and project in a loom video.

Activities performed this week (bullet points with explanations):

- In-class team activities: Math and Science project
    - AtomsBinding Scene
        - Added new camera views, more atoms to the scene, modified atom collision barriers and script to bind with no more than two hydrogen atoms.
    - Growth Scene
        - Edited this scene to visually demonstrate a normal distribution curve (from concepts of Statistics). It features:
            - Pillars that display the standard deviations from -3 to 3 (by integer)
            - Sliders that control the sample mean, standard deviation, and a chosen z-value
            - When the "Run simulation" button is clicked, it updates the pillars through a growth animation, displays the value for each standard deviation, and forms a normal distribution. Moreover, it generates, displays, and grows a pillar using the user-chosen z-value.
    - Logic Scene
        - Added the ability to complete random challenges within the scene. The scene starts by generating a random position and two random objects, and displays the challenge text to the user on screen. Moreover, the user can click the "Start a Challenge" button to generate a new type of challenge. Once the user thinks that they have completed a challenge, they can verify the information using the "Check Logic" button, which will increment a counter and generate a new challenge if the user successfully completes it.
    - Statistics Scene
        - Modified to calculate and display the blue sphere's displacement, velocity, and acceleration. When the user clicks on the "Execute" button, it will calculate these variables manually based on principles of Physics. Moreover, we modified the scene by adding elements of drag and a rotating ramp (using lerp function).
- Worked on adding mathematical and scientific concepts to the Space Trooper project based on a set of goals and deliverables for Topic 6 (see below).

Overall progress (describe new knowledge acquired, successes, ideas generated, etc.):

- Skill Inventory Table

Item #	Who gained the skill/knowledge?	Explain Skill/Knowledge Gained
1	Diego, Ryan, Andrew	Learned how to effectively lerp in unity
2	Diego, Andrew	Learned how to more effectively change from different scenes
3	Diego, Ryan, Andrew	Learned how to use Parent and Children objects more effectively in C#
4	Ryan	Learned how to put audio continuously throughout the scenes
5	Diego	Learned how to change primary code editor on Unity
6	Andrew	Created a really effective main menu that has a really great UI
7	Andrew	Learned how to Edit Skybox
8	Andrew	Learned how to use Blender
9	Andrew, Diego	Mouse Over Function on Unity
10	Diego, Andrew, Ryan	Coroutines
11	Andrew	Creating sound and editing sound files directly in unity
12	Ryan	Basic Player movement for Space Invaders (Lerping to set transform positions).

Space Trooper Project Plan (from Topic 5)

- Gather and create assets
  - 3D Objects
  - Scripts (if any)
  - Sprites
  - Materials
  - Particle Effects
  - Lighting
  - Sounds
    - Music
    - Sound Effects
  - Plugins & Packages
    - Post-processing
  - Animations

        - Rigging
        - Projectiles
    - UI
        - Menus
        - Scoreboards (Points, Timers, etc.)
        - Game Over Screen
        - Skybox
- Initial Scene and Asset Layout
    - General Assets
        - Space Skybox (Ryan)
        - Space Music Ambiance (Ryan)
        - Draft Loading Screen (Andrew)
        - Astronaut Player (Diego)
    - Intro Scene (Andrew)
        - Rocket Ship
            - 3D Object
            - Smoke Particle Effect
            - Explosion Particle Effect
        - Timer UI
    - Main Menu Scene (Andrew)
    - Space Station (Diego)
        - Hallway
        - Doors
        - Windows
        - Ceiling Lights
            - 3D Object
            - Effects
    - Space Kitchen (Diego)
        - Pancake
            - 3D Object
            - Multiple Materials to Simulate Cooking
        - Cooking Script
        - Frying Pan
        - Stove
            - 3D Object
            - Stove Flame Particle Effect
    - Space Walk (Diego & Ryan)
        - Space Station 3D Object
        - Space Walk Controls Script
    - Space Fruit Slicer (Ryan)
        - Various 3D Objects
            - Fruit
            - Toast
            - Candy
            - Fast Food
        - Slicing Particle Effects
            - Juice Effect
            - Food Effect
        - Fruit Spawning Script
        - Mesh Slicing Script
    - Fire Extinguisher Obstacle Course (Andrew)

- Fire Extinguisher
  - 3D Object
  - Fire Extinguisher Emission Effect
  - Controls Script
- Hoops
- Portal Cube
- Space Invaders (Andrew, Ryan and Diego)
  - Prop Spaceship
  - Cannonball
  - Cannon
  - Cannonball Spawning Script

Space Trooper - Topic 6 Goal & Implementation Deliverables

- Goal for Topic 6: Build simplistic versions of every scene using concepts from Math and Science. Moreover, to build a final product, an additional goal for this project is to add Oculus VR functionality and navigation between all scenes.
- Andrew
  - Intro Scene
    - Utilize a rocket model with gravitational motion (features gravity vectors and gravitational rotation), and v science to simulate a realistic rocket launch webcast.
  - Main Menu Scene
    - Acts as the primary hub for launching to the Space Station, settings, and exiting the game
    - Create a spherical room GameObject with a functioning control panel
      - The control panel features several buttons that users can interact with using the OVRPlayerController or mouse.
  - Fire Extinguisher Obstacle Course Scene
    - Build a basic hoop object and start adding functionality for randomly generating hoops within the scene to create an obstacle course.
  - Adding Navigation Between Scenes
    - Add multiple types of loading screens (using async coroutine), Main Menu scene, and more.
- Diego
  - Space Station Scene
    - Build out the hallway, and add basic navigation between scenes using doors.
  - Space Kitchen Scene
    - Simulate the visualization of cooking a pancake using a pan object, a lerp function and real-life cook times
  - Explore Oculus Quest VR Avatars
    - Creating Models that work with OVRPlayerControllers
      - Astronaut
      - Player Hands
- Ryan
  - Space Fruit Slicer
    - Make progress towards mesh slicing, spawning fruit in a spherical pattern, and destroying fruit objects upon collision with another object (e.g. sword, arrow, etc.).
  - Space Invaders

        - Build basic functionality Cannons w/ Firing Mechanism Cannonball
    - Adding Persistent Sounds across all scenes
        - For example, if background music is playing in one scene, and the scenemanager switches to a scene with that same background music, it will keep playing the music from where it originally left off.

Issues that need to be resolved (bullet list):

- Adding window gameobjects to scenes (e.g. main menu scene, space station scene) to allow the user to view space from inside another object.
- Creating an OVRPlayerController prefab with long-distance pointers (that acts like a mouse).
- The loading screen graphic and loading bar are inconsistent between different screen resolutions.
- The mesh slicing script is potentially functioning, but now we need to make it work in VR.

Next steps (how will you mitigate the issues listed above; bullet list):

- Look into adding window objects in Blender or creating a universal window asset for the Space Trooper project.
- Explore Oculus integration package in Unity for further guidance on implementing long-distance pointers in Unity.
- Test and edit Canvas or Image configuration settings to resolve the screen resolution problem.
- Attempt a different approach to mesh slicing for the Fruit Slicer scene.

Other comments:

- GitHub Link: https://github.com/drewesch/CST-320
    - Our Unity Scripts are located under /Scripts/Topic-6
- Loom Link: https://www.loom.com/share/a5d7d5bf28ea49aa868ffa9cbba1f208