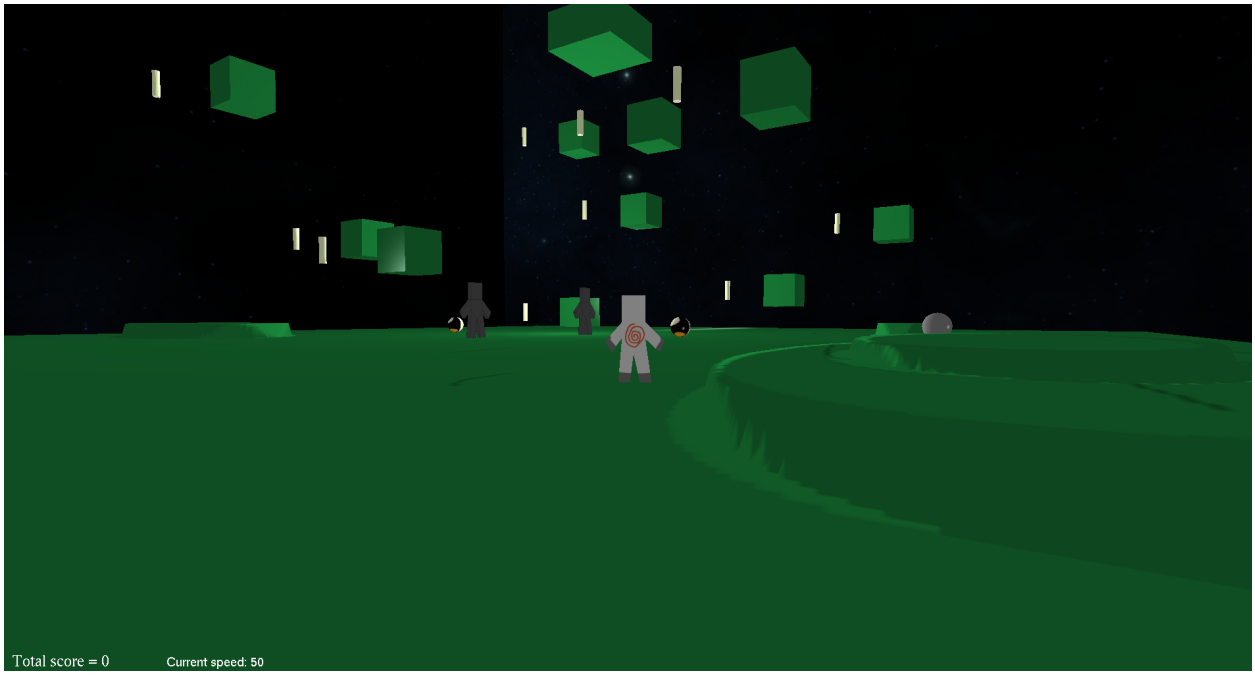
1. Craftmine Parkour, Package name for game is craftmine. Jarod Shavor and Joseph May, Section - 02

2. 

3. To run game:
   Run compile file
   Run compileServer file
   Run server file
   Run run file

4. None

5. The player can move forwards and backwards and move the character left and right. They can also jump. The goal is to jump up the platforms to make it to the top platform where they gain a point. In order to gain more points they must touch the ground and then make their way back to the top platform again. The player can only jump once they have collided with something, meaning you can also jump off the sides of the platforms. The shadow enemies will launch you, so you need to watch out for them.

6. W: accelerates the player forward
   S: accelerates the player backwards
   A: turns the player left
   D: turns the player right
   Space: The player jumps
   X: jumps the player, but downwards instead of up.
   Q: increase player move speed
   E: decrease player move speed

P: toggles candle lights
J: azimuth orbit camera left
L: azimuth orbit camera right
U: zoom in
O: zoom out
I: elevate orbit camera up
K: elevate orbit camera down

Gamepad:
Y axis: move forwards/backwards
X axis: rotate avatar left/right
Button 0: jump
Button 1: jump down
Button 2: zoom in
Button 3: zoom out
Button 4: increase speed
Button 5: decrease speed
Button 6: toggle lights
Axis RX: Azimuth camera
Axis RY: change elevation of camera

7. Movement and turning speed for the player
   Initial points
   Avatar's ability to jump initially
   Avatar can score or not initially
   Strength of force on avatar's movement
   Lights are on or off
   The mass of objects affected by gravity

8. No changes to the TAGE networking.

9. Added a getRigidBody method to physics object
   Added nodeControllers FlyController, StretchController, and ShrinkController
   Added straightMovement, panVerticalMovement, panVerticalMovement, panHorizontalMovement, yaw, and pitch to Camera.java
   Added the CameraOrbit3D class
   Added straightMovement, yaw, and pitch to GameObject.java
   Added removeLight to SceneGraph.java

10. Genre: Platformer
    Theme: Nighttime plains
    Dimensionality: 3D
    Activity: Parkour

11. Input controls: Player can control movement of avatar, and control movement of camera, using both keyboard and gamepad
    Networking: Other clients that connect will be represented by their ghost with their selected avatar.
    Two models: The avatar/shadow model, and candle model
    Single player mode: If only one player is present, the player can jump around and get to the top to score points
    Choose avatar: Players have a choice of 4 avatars. The client will send that data to the server and then the server will send that to all the clients for the ghost creation.
    Scripting: certain variables are initialized using a script
    Skybox: the game area is encompassed by a skybox.
    Terrain: The game has a flat grass plane for the avatar to stand on, and physics objects to bounce off of
    Lighting: We have a global light that lights up the area, as well as multiple candles that each have spotlights that circle the platforms and aim at the ground.
    3D sound: background bird sounds. Sound effect when player jumps. Sound effect when a player collides with something
    HUD: Displays players current movement speed and their current score
    Hierarchical scenegraph: The candles are children of the terrain cube platforms in the air and translate around them.
    Animation: as the avatar moves around it has a walking animation
    (in game): The animation is visible in game as the player moves.
    NPCs: The shadow enemies are the NPCs.
    AI for NPCs: They walk back and forth by default, but when the behavior tree sends a location message and is told the avatar is near, it calculates the straightest route to the player and walks directly towards them.
    Physics: The player is affected by gravity, there is a ground plane, two balls the player can interact with, and many cubes that float around that are physics objects for the player to jump off of.
    collision detection: The player avatar can bounce off of the ground, balls, and the many platforms around, and all of them use collision to do so. The balls also interact with the ground plane accordingly Collision is also used to detect the scoring for each player.

12. The avatar does not follow the terrain’s heightmap

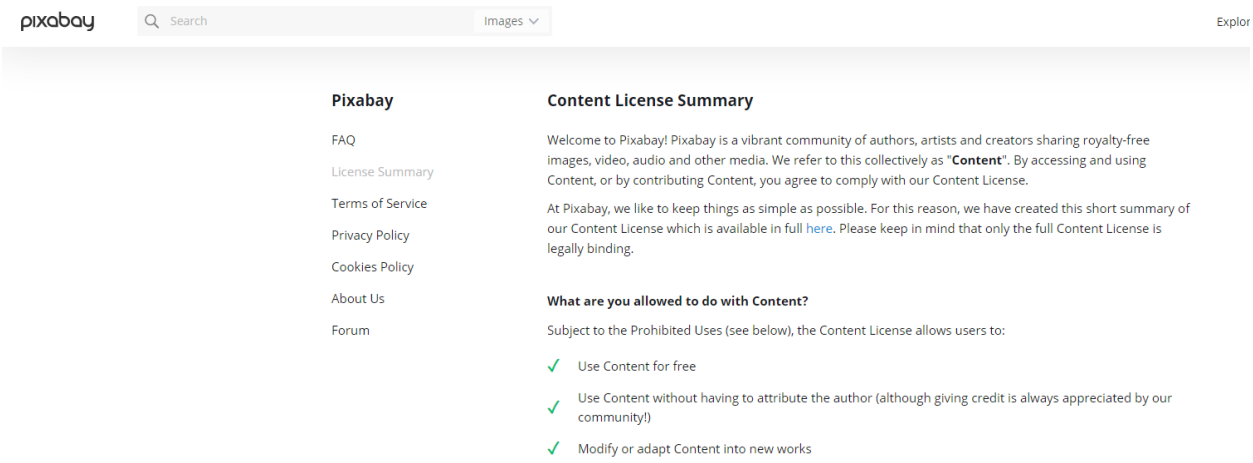
13. The avatar can jump around, we have 4 avatar textures to choose from.

14. Jarod Contributions:
    Physics (jumping, movement, various platforms and objects)
    Sound
    Scoring system
    Player model
    AI texture

HUD
Scripting

Joseph Contributions:
Networking
NPC/AI
Animation
Candle textures/Model
Avatar Textures
Lighting

15. Candle.png, grass.jpg, hills.jpg, AItexture.png, candle.obj, player.rka, player.rkm, player.rks. avatarUVskin1.png, avatarUVskin2.png, avatarUVskin3.png, avatar1.png, avatar2.png, avatar3.png, avatar4.png
16. Sounds from https://pixabay.com/
    Jumping sound: https://pixabay.com/sound-effects/toy-button-105724/
    Landing sound: https://pixabay.com/sound-effects/human-impact-on-ground-6982/
    Bird noises: https://pixabay.com/sound-effects/birds-singing-in-the-morning-nature-sound-146832/
    Permission to use: https://pixabay.com/service/license-summary/



    Space skybox from https://opengameart.org
    Skybox link: https://opengameart.org/content/space-skybox-1
    This skybox has a CC0 license, meaning it has been dedicated to the public domain and anyone can freely use it. The details are here: https://creativecommons.org/publicdomain/zero/1.0/
    The Dolphin_HighPolyUV.png was given to the students for the earlier projects in the course
17. PACMAN and FALLOUT