Week 4 Presentation
Team 3

By Emily, Carson, & Mat

Game Recap

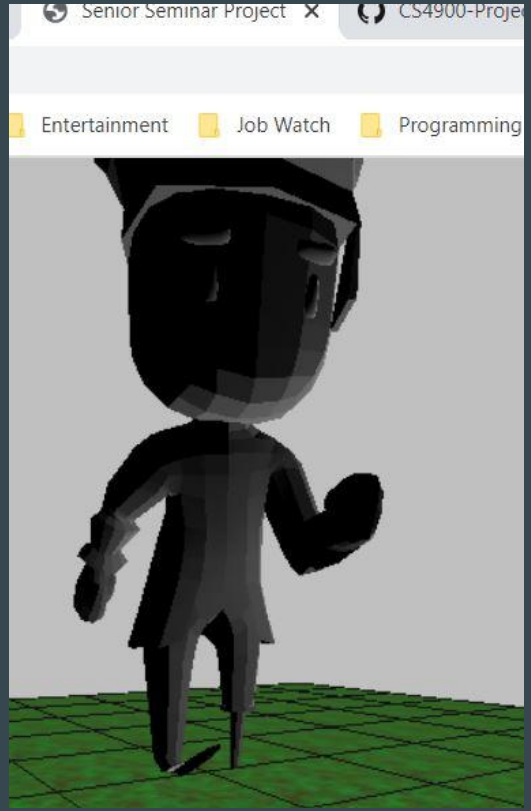
- Game: Rogue-like, Tactical RPG
- Think of Games like XCOM, Fire Emblem, Final Fantasy Tactics

Replacing models (@quaternius)

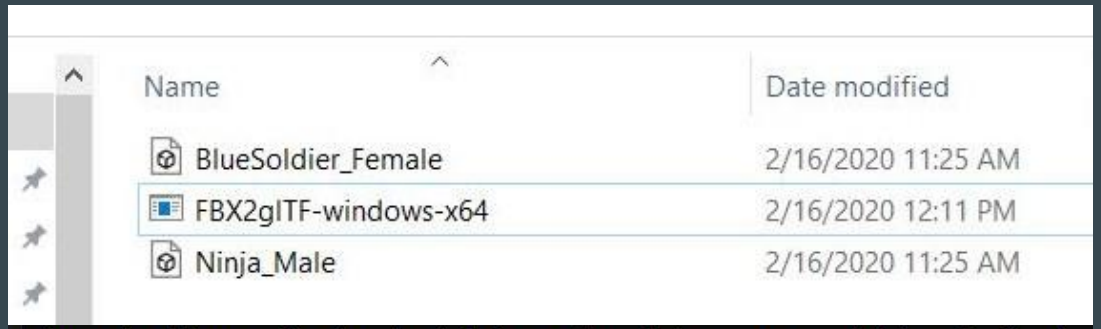
https://www.reddit.com/r/gamedev/comments/dwqsl9/ive_made_50_free_rigged_and_animated_lowpoly/

Converting to glTF

- Exporting from blender
  - Imported as .blend and exported as .glb
  - Didn't work
- https://blackthread.io/gltf-converter/
  - Web form that accepts certain file types and converts them
  - GLTF viewer gave >3000 errors
- The conversion that finally worked?
  - Facebook's FBX2glTF utility



FBX2glTF - https://github.com/facebookincubator/FBX2glTF

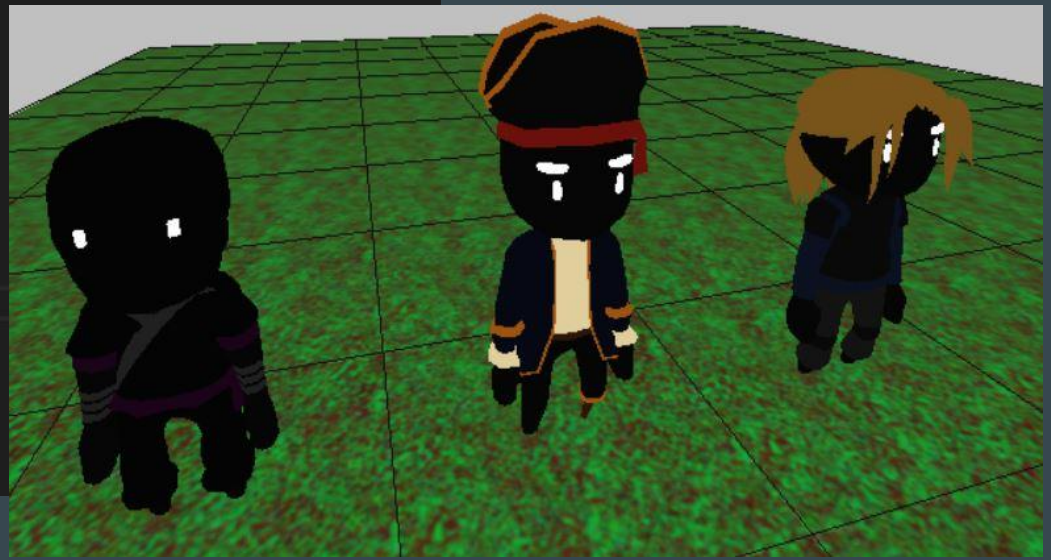



```
C:\Users\Mathieu Davidson\Desktop\gltf>FBX2glTF-windows-x64.exe ./Ninja_Male.fbx -b
Animation CharacterArmature|Defeat: [0 - 60]
Animation CharacterArmature|Idle: [0 - 100]
Animation CharacterArmature|PickUp: [0 - 30]
Animation CharacterArmature|Punch: [0 - 18]
Animation CharacterArmature|RecieveHit: [0 - 15]
Animation CharacterArmature|Shoot_OneHanded: [0 - 13]
Animation CharacterArmature|SitDown: [0 - 23]
Animation CharacterArmature|StandUp: [0 - 31]
Animation CharacterArmature|Victory: [0 - 45]
Animation CharacterArmature|Walk: [0 - 30]
Animation CharacterArmature|Walk_Carry: [0 - 30]
Wrote 453861 bytes of binary glTF to ./Ninja_Male.glb.
```

Finally, implementing LoadingManager with glTF

```
const models = {
  melee:    { url: './models/Pirate_Male.glb', name: 'melee', pos: 1.5 },
  ranged:   { url: './models/Ninja_Male.glb', name: 'ranged', pos: 2.5 },
  defender: { url: './models/BlueSoldier_Female.glb', name: 'defender', pos: 0.5 },
};
const gltfLoader = new THREE.GLTFLoader(manager);
for (const model of Object.values(models)){
  gltfLoader.load(model.url, (gltf) => {
    const root = gltf.scene;
    root.name = model.name;
    root.position.set(model.pos, 0.01, -3.5);
    root.rotation.y += Math.PI;
    root.scale.set(.34,.34,.34)
    //root.visible = false;
    scene.add(root);

  });
}
}
```



Imports and Exports

- This week, I worked on some refactoring
  - Imports and exports are a great way to make your code more readable and flexible
  - Functions are essential for modularization as well as importing and exporting
  - Visual Studio Code also has a useful color scheme that shows you if something is unused

```
import { scene, mapTopZ, mapRightX, mapBottomZ, mapLeftX, highlights } from '/main.js';
import { createHighlight } from './worldGeneration.js';
```

```
export { scene, mapTopZ, mapRightX, mapBottomZ, mapLeftX,
    highlights };
```

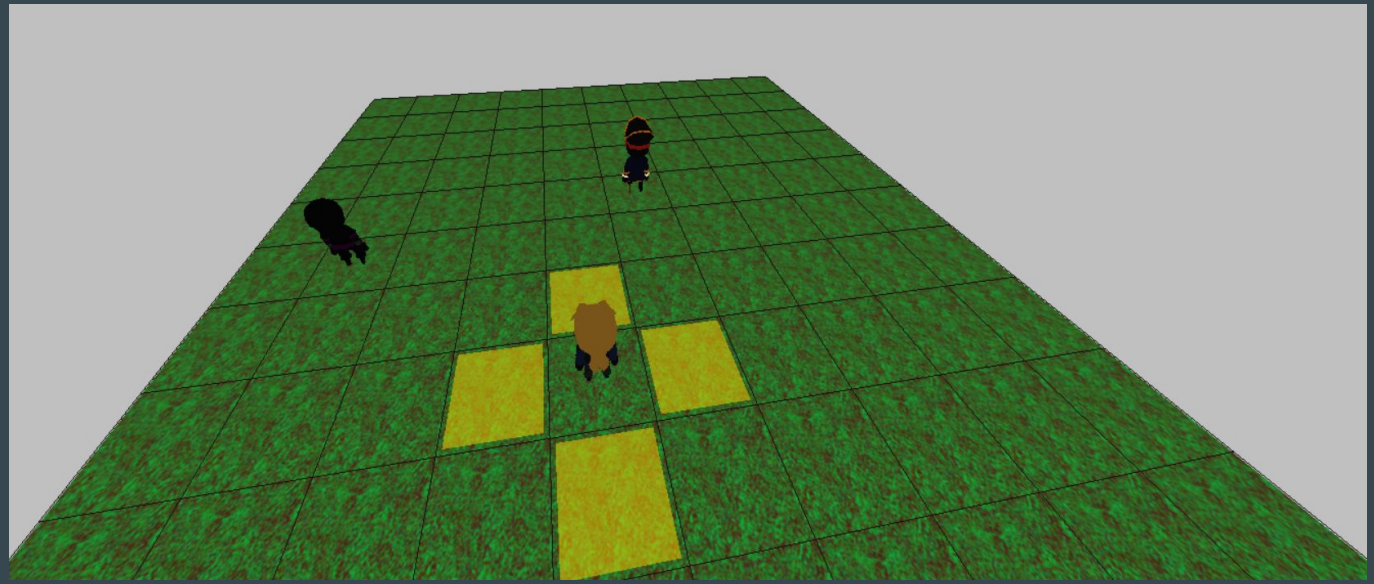
Characters

- I utilized Mat's new model creation and edited my functions to reflect this

- Each character now has a certain number of moves
  - A *turns* attribute was added within the previous object creation
  - Thanks to a while loop, the event handler only allows them to move if they have moves left
    - Otherwise, the character counter increments, changing the movable character by using *scene.getObjectByName*

Characters

- Highlight positions and visibility are reset using *resetHighlights()*
- Cycling characters is in progress
  - *We would like to utilize a Linked List to implement this*



How are we doing terrain?

- Similar idea to Group 1
- How we are changing it:
  - Using a 3D matrix, where we have:
    - X Position on Grid
    - Y Position on Grid
    - Properties of the block
      - Height
      - isOccupied
  - Dedicated spawn locations and dedication terrain positions (Plot and Parcel)

Sample Matrix

[

[[x, y], [0, 1]],

[[x, y], [4, 0]],

[[x, y], [2, 1]]

]

Trial and Error

- First thought for terrain generation:
  - One static model with holes that we fill in later
- Issues:
  - Not easy to manipulate plane geometry (i.e create holes in it)
  - Would have to figure out how to push objects into the created holes
- Tools that looked useful, but weren't
  - CSG.js
  - THREE.shape

This week's useful findings...

- CSG.js
  - A quick way to create custom shapes
  - https://evanw.github.io/csg.js/
- Imports and exports
  - Keeps code neat and tidy
  - Great way to allow different files to interact
- FBX2glTF
  - https://github.com/facebookincubator/FBX2glTF
- @quaternius - 3d Model Maker
  - https://twitter.com/quaternius?lang=en

Our next steps

- Emily
  - Cycle between characters using a Linked List
  - Add highlight “trails”
  - Add enemies
- Carson
  - Work with Mat on generating terrain
- Mat
  - Work with Carson on generating terrain
  - Add animation code into the model factory (AnimationMixer)