Cruiser Chaos

Ethan Dorow, Nile Henry, Kyle Wybranowski

Game Objective

Navigate the cruiser while avoiding obstacles along the way. If you hit an obstacle (goose, roadblock) a passenger will get off the cruiser. Pick up passengers to gain more health. If you run out of students, you lose. See how long you can last!

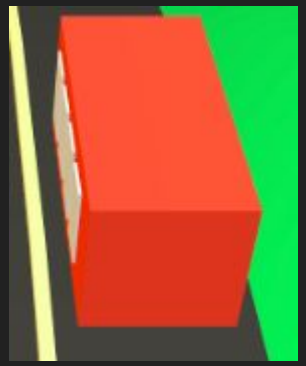
Backend Design (Engine,GameObject,Component)

- ECS + Callback
- The game’s underlying structure is an engine which manages all aspects of the gameplay. Engine consists of functions such as adding/removing instances of certain game aspects of game, updating gameplay at each frame, etc.
- GameObject
  - Contains a render (which holds the model for the specific object)
  - Contains other game logic which applies to given object
  - Contains a transform which transforms from local to global coordinate system
- Component
  - Aspects individual to specific object. Ex. Timer, BoxCollider

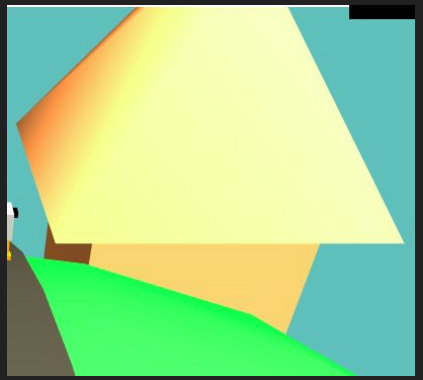
```
var empty = new GameObject(engine,
    new Transform(-laneWidth + (i * laneWidth), 0, -worldRadius - render.height),
    render);
var despawnBox = new BoxCollider(10, 10, 10, DefaultTransform(), COLORS.yellow);
var hitBox = new BoxCollider(laneWidth, laneWidth, laneWidth / 2);

despawnBox.RegisterOnCollision( (_this) => empty.Destroy(engine) );

hitBox.RegisterOnCollision( (_this) => cruiser.TakeDamage(engine) );
hitBox.RegisterOnCollision( (_this) => empty.Destroy(engine) );

empty.AddComponent("DESPAWN_BOX", despawnBox);
empty.AddComponent("HIT_BOX", hitBox);
```

GameObject Models

Cruiser



Building



Goose

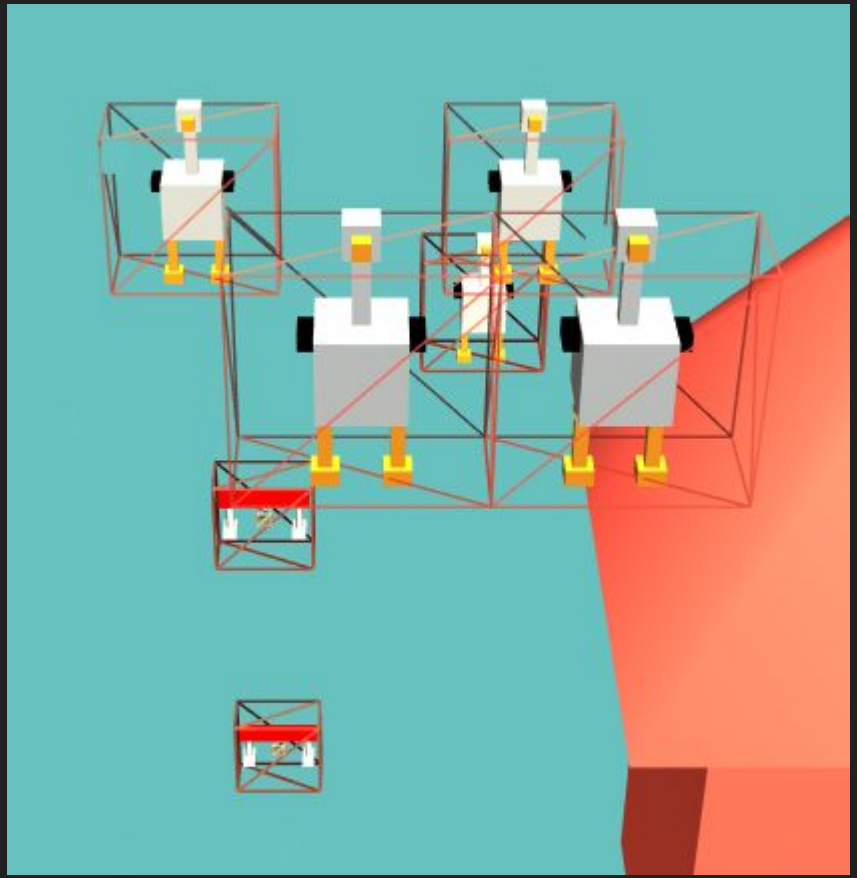
Roadblock

World/Cruiser/Obstacle Movement

- The world (centered at x: 0, y:0, z:0) rotates about the x axis at a designated speed.
- Obstacles rather than being attached to the rotating world are distinct from the rotating world and instead are placed at a position in the global space. They then rotate about x:0,y:0,z:0 at a designated speed
- The cruiser remains in the same z-position while all the other objects rotate to convey that the cruiser is moving
- Cruiser switches lanes with key inputs
- Sky rotates around world, changing lighting from day to night

Collisions

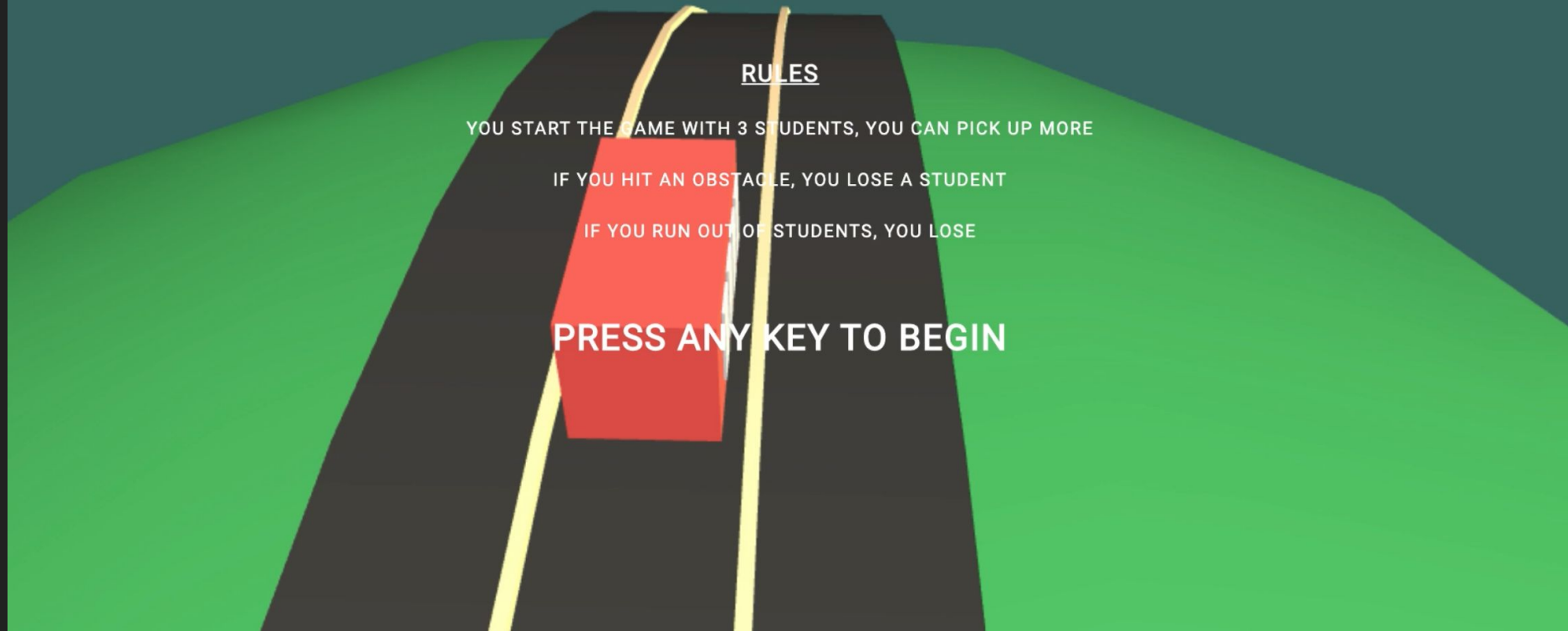
- Collisions allow objects to interact with each other
- Each interactable GameObject has a CollisionBox component



UI/UX

- BeginGameMenu
  - Handle key pressed
- GameUI
  - Score, Health - backed by GameController data (HTML -> JS)
  - Cruiser hit/heal sends callback to GameController to update HTML
- GameOverMenu
  - Handle key pressed

CRUISER CHAOS



Lessons

- No rolling world
- More graphics components (lighting, shapes, etc.) for free
- More defined roles
- More thorough planning (now we know time commitments)
- A bigger team

The End