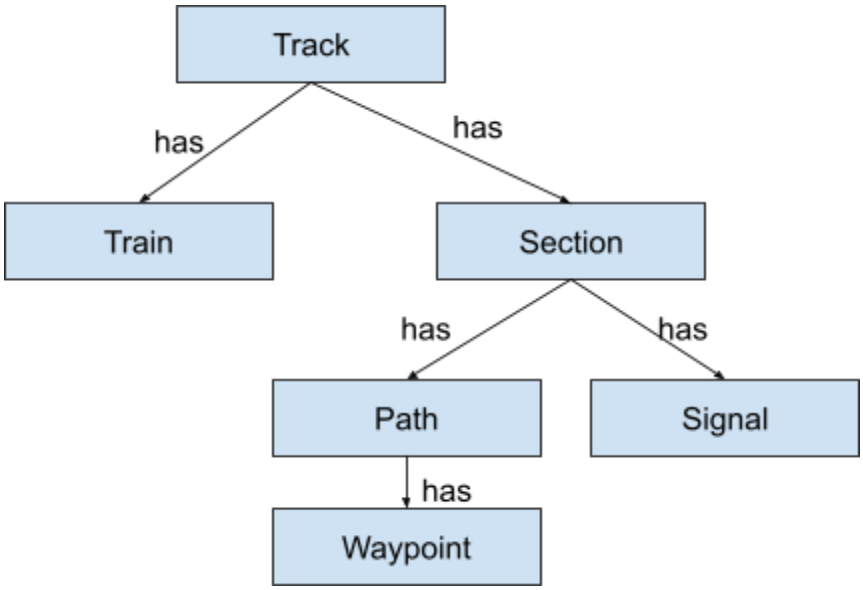
Purpose:
This document is designed as a way to ensure that I as a programmer am able to keep track of the overall design and functionality of TOS.

Object Orientation:



This diagram is meant to show the way in which objects are held. One track has many trains and many sections on it. Each section has a path to follow and two signals at either end. Paths are made up of Waypoints.

Track:
In this local build of the game track serves as the stand in for the server, it is the main driver and acts as the TOS. In order to serve as the TOS the Track object will need to hold an array of all the trains on track and an array of all the sections of the track. Because Track is serving as the TOS Track needs to control the updating of everything. The onUpdate() function in Track will drive the movement of each train. Because it is not threaded at this time each train will be updated sequentially and there should not be any racing condition issues (yet).

Train:
A Train only knows a few things. It knows its location exactly, the index of the Section it is currently in, and its speed. Track informs Train when to move and where. Train informs Track when it needs a new target for movement.

Section:
A Section makes up a portion of the track. Sections have a Path, an int lock, and two Signals. Sections tell the Track what the next Transform in their Path is when asked. Sections unlock and lock when told to by Track. Track will manage them such that only one Train is allowed on a Section at a time.

Path:

Path holds a list of waypoints in a given order and provides the next in their order upon request by Section.

Signal:
Signal is a sprite which indicates visually if a Section is locked. One is placed at the beginning and end of the Path in a given Section.

Waypoint:
A Waypoint is a transform in the world, a location that can be pathed to. All Waypoints are parented by a Path.