

Zombie/Bug Train Idea

Creative Vision Document

Zombie Game Reference(s):

- Black Ops: Cold War Zombies
- Final Station
- Spiritfarer

Final Station:



Train Reference(s):



3Key:

- Horror, Action & Survival
- First Person Perspective
- Add Horror/Occult Shit???

Engine: Unity or UnrealEngine

(Not sure how Unity's Engine would handle both Zombie and Train Physics.)

Goal:

- UNKNOWN

Objectives:

- Find parts to equip the train.

(Recurring) Objectives:

- Find fuel for the train.
- Find scrap (metal junk) for the train.

Interactions:

- The player can choose a stop for the train to go to, the player will be redirected to that route (however, the train won't go backwards)
 - Incentive: The player will plan routes ahead.
 - Risks: The engine will be able to be destroyed.
- The player will be able to find scrap to armor and upgrade the train.
- If a player dies they can be respawned at a different outpost.
- If the player's engine is damaged, the player can opt to repair it at an outpost.
- The player has to go to a specific Outpost; that will somewhat always have a minimal amount OR no zombies -- this one will allow the player to do extensive repairs and upgrade the Train. (As in add turrets(which should be instant upgrades) and overall upgrades.)

Game States/Ideas:

- Each time the player goes to a specific zone/outpost, the amount of zombies will be increased. (nMaxHordeDensity will go up.)
- Emission Blowout/Like Stalker, if the player is inside of a trigger volume for the train, they should be protected against it.
- Zombie/Bug/Creature Giga-Horde -- (Also maybe an on train specific event??) Now the way the Emission and Giga-Horde works is there is a bool to determine if an event will happen. Then there will be three chances that could happen: X Chance for Giga-Horde, X Chance for Emission and X Chance it was just an odd feeling (as in nothing bad happens). (Create a Health & Radiation system for the Player)

Fail States:

- The engine is destroyed.
- The players are all dead.
- The train runs out of fuel.

Stops (The Secondpiece)

- Zombies/Creatures should be able to sneak onto the train while it's stopped -- and won't go after until the train is moving. (presents an additional risk to the train being stopped.)
- There are special zones that must be routed from a specific outpost.
- Stops have area-specific loot?????

Train (The Centerpiece)

- When the train's fuel is less than 25% the speed of it is slowed until halted. (Which may increase the chance of the player rerouting to a closer place rather than passing it by.)
- The player can fuel up the train with one-time use fuel, however, this will cause the engine to be damaged -- and it only adds a little bit of fuel.
- While the train will automatically stop, at the nearest stop, the player will need to start it back up. (Prototype Train?)
 - Caboose (Not Important but an Idea(If Done Well, can implement))
 - The player will have a caboose and can add on a caboose.
 - The player can find objects to pack into a crate; and sell the caboose's contents to one of the special outposts. As in, if they lack specific resources it can be sold for whatever scrap they have lying around.
 - If the Train has enough cargo (over encumbered) it should either: go slower or the player can choose to make it go faster, thus consuming fuel at a slower rate at the expense of damaging the engine.

Because of the nature of the Train, it might be better for it to be controlled via an AI StateMachine. When it reaches the intended NavPoint, TrainState.TrainStop() should be called. When a route is created, the intended NavPoint.transform, should be updated for it but it should move into TrainState.TrainGo() is called.

When the train is moving after being in Train.Stop(), there should be a gradual increase in speed into a constant speed. Which will have its own rate?? To make it go slower, or faster depending on fuel consumption.

The player should be able to vote, (if more than one player) for an emergency stop. As in, if the engine is heavily damaged. They can temporarily reinforce it, then take out the Giga-Horde or until it's passed. However, the chance of zombies "sneaking" onto the train is increased by ten percent of what the normal x rate is.

When inputting the RouteCode, if the string of the input matches a route, it should be locked and logged. Which means that I should create a separate function of where the player can input something and then the route will be set.

Health & Radiation:

- **Radiation:**

- Damage:
 - When Radiation damage is low, there is no significant problem to the player.
 - When Radiation damage is medium, the player's speed will likely be reduced.
 - When Radiation damage is high, the player will actively take damage from the health pool.
 - Radiation is mitigated by using radiation consumables.
 - Even though Radiation can be 0 or 100, Radiation reaching 100 will not mean the player automatically dies. Just that they need to use a RadConsumable.
 - If player's listen to the audio/visual cue, they will be able to take a consumable item which will not allow their rads to go up as fast as normal. (Slower Radiation gain)

- **Health:**

- Health exists as normal as with any other game.
- Health is replenished via consumables.

Backstory Idea:

YOU are Void Researchers who have volunteered to be sent on a planet to survey the way things died the way that they did. Somewhere along the line, split blood had allowed the things to know where they were. You must find the corpses of the first agents who died and recover what they have found out -- then leave the planet.