Prototype Documentation

November 13, 2023

Contents

Preface

This documentation satisfies the Checkpoint 2: Prototype written requirements: an updated game pitch, project synopsis, objectives, and mechanics documentation. The document here is based on our own internal documentation. I understand the game is in its infancy, but working in 3D is significantly harder than in 2D. In short time, while juggling classes, and while managing work as I can find it, I've had to learn new programming skills, animation, mapping, texturing, blueprinting, UI, and more. Over the following weeks it will evolve significantly.

1 Game Pitch

Terfenol plunges players into a dystopian future where corrupt tech elites unleash a Terfenol-D based nanotech, unintentionally turning the surrounding environment into a monstrous, sentient fog. Navigate apocalyptic landscapes guided by a specialized tablet revealing hidden horrors and unstable weather anomalies; equip yourself with tools of survival like nanotech disruptor shields and guns. Embark on a harrowing mission to deactivate the technological foundation of this nanotech plague: 6 signal repeater towers relaying the commands of a central tower, battling evolving nanotech beasts that shroud the world in darkness, each success intensifying the challenge. Survive, adapt, and unveil the truth in a strategic struggle against a technological apocalypse designed by humanity's own hubris.

2 Project Synopsis

Game takes place in a not-so-distant future where major tech companies formed a conglomerate (or just got purchased by one big company). In an attempt to control weather, the food supply, and satisfy a god-complex - as well as for a deeper insidious reason - the company's big-wigs developed a nano-technology controlled via long range magnetic waves. The working title - and possible final title - of the game is Terfenol, a reference to the magnetostrictive substance used in the nanobots construction; using magnetic waves you could influence the Terfenol nenoatech and thus the (swarm) behavior of nano-bots. The nanobots are unwittingly released onto the environment to "beneficially" improve the ecosystem but instead cause havoc. A computer in the central tower running an AI-model uses the nanobots as long-range sensors and data input to make complex decisions. In a short time all the citizens, fauna, and a large portion of the flora are wiped out in a mass extinction, within a defined radius of the tower and 6 repeaters. A black fog covers the affected region, densest where the signal is strongest. A combination of already failing socioeconomic systems, poor government response, external actors taking advantage of the situation, and the (obvious) damages of the event, result in an inability to stop the nanobots. Most of the developers on the project die in the process, so much knowledge is lost. Four main characters - and some side characters - attempt to destroy the 'plague' of nanobots. Our main cast form a team of intrepid explorers and venture in. This group either worked on the technology while it was being developed, has a unique background that lends them to fighting the plague, or a niche motivation; all have their own agenda. Additional characters may be met along the way, all who have a similar background or skills.

For development, at a high level we can divide the tools of our project into level design, character design, animation, particle effects, UI, and game logic. The levels will be designed in Unreal Engine with some additional help from Gaea and miscellaneous heightmapping tools. The foliage in the levels was largely sourced from Quixel Megascans free, online library, as were some models. Lighting and audio both employ the tools provided by Unreal Engine, but all settings and audio were built by the team. We are not using Nanite or Lumen, preferring instead to focus on low-LOD assets that can be run on our old computers. For character design we're relying heavily on MetaHuman and Blender; the former allows us to relatively easily self-insert, while the latter requires a steep learning curve and will be used to model the monsters. Animation is also completed in Blender, as well as Unreal Engine and Metahuman. Particle effects are accomplished with Unreal Engine, using its Cascade and Niagara particle systems. UI is created mostly in Unreal Engine, with some additional icon help from Photoshop. Game logic is completed with Unreal Engine's blueprint scripting and C++.

In addition to the development tools mentioned above we have tried figuring out a good project management/content management system. Unfortunately this has been absolute hell, occupying 40 hours of wasted time. From GitGUI to GitHub Desktop, almost nothing seems to be doing what we want. GitHub is far to finicky and confusing for long term use, and the fact you need to have local copies of the entire repository makes game development almost impossible. The day of Checkpoint 2 we discovered Preforce, an alternative similar to GitHub that strikes a balance with Google's Piper. Preforce makes group work on

massive projects significantly easier, and it appears to have integration with Unreal Engine; the most obvious benefit is we can check out individual folders or groups of folders, letting each of us work on what's absolutely necessary without needing all assets ever. We also discovered Unreal's own collaboration tools today.

3 Objective

The goal - which plays into the mechanics of the game - is to destroy all 6 repeater towers, and reach the central tower. As our characters traverse the open world to destroy repeaters they'll encounter synthetics. Once they're at the central tower they'll have to 'climb', making their way through the horrors of the company, discovering vile truths, finally culminating in a way to defeat the plague. The tower climb will almost certainly be the subject of a second game if the first is successful.

In practice this appears as a large map with players stationed near the limits. Each day there's a routine action pattern: wake up, scout, sleep. In the wake up action phase the player can customize their gear, and is given the option to participate in some event that will boost stats. In the scouting phase the character needs to scout the map as much as possible, and try to identify the towers while avoiding monsters. In the sleep phase the character is once again given the opportunity to customize and interact.

4 Mechanics Documentation

4.1 Preface

Below is an 'anthology' of the various mechanics we've come up with. These are in no way complete, many of them not even started, but represent a demonstration of the thought we've put into the game. The higher priority, 'more complete' will be near the top. The later mechanics, but still to be included, will be at the bottom.

At its core this prototype serves to demonstrate player movement, and monster-chasing AI.

4.2 Destruction of Repeaters and Scary Monsters

As mentioned previously, there will be a central tower, around which 6 repeaters are arrayed. The repeaters provide a strong signal for the nanotech in the environment; to inhibit the nanotech's reach you must destroy the towers. Note that destroying the towers will not kill the nanotech, just cause them to redistribute to a region of stronger signal. Currently the 'tower' is a simple block with minimal interactivity, only there to pace the game. In the future this will be replaced with its own model and a far more meaningful interaction.

At the destruction of each tower a new beast will be formed, something to hunt you. It will not appear directly near the player, but somewhere close in the environment. The beast will be empowered in regions of better signal. The player has the opportunity to flee from the beast (stealth and running are options), but they could also fight it. Choose to fight and you may kill it permanently; don't kill it and the beasts will compound as you destroy more towers. The beast will avoid being killed and flee if it feels it's weak. Currently we've only been able to implement about half of the AI of the beast, and we're representing it with a floating orb. We still have some improvements to make to the perception and roaming of the beast, but most of the work is in the aesthetics zone.

The beasts are manifestations of the swarm intelligence of the nanotech, controlled by the AI. As a result it acts less like an individual animal or being with ego, and more like some extension of nanotech. Early beasts will be more crude, simple, and unshapely in their design; but, as you destroy successive towers they will become more precise, larger, intelligent, and violent. The rationale is that the nanotech isn't initially sure if it was your fault that the tower got destroyed, so it simply sends monsters to investigate and progressively hunt you.

4.3   Tablet

This mechanic, as important as it is, was not included in the prototype. Though I've created a stand-in for the tablet static mesh, and have even implemented a rudimentary version of the camera functionality, it's not in good enough form to include with the game. If you'd like to see a demo of the tablet's interaction model, consult Assignment 2.

The core objective of the tablet device is to facilitate players in navigating the environment, identifying nanotech manifestations, and interacting with various elements and objectives ad hoc. It acts as a lens, a guide, and a crucial component for decision-making processes, thereby playing an essential role in the player's journey and survival.

The tablet comes equipped with a spectral sensor or camera allowing players to almost peer into an altered visual dimension. To understand what I mean by this, imagine you're taking a picture, settings like ISO, F-stop, and WB can be used to correct the image in undesirable settings, sometimes enhancing your human vision beyond what's natural. Camera capabilities on the tablet in game will permit environment scanning - enhanced by sliders for Specularity, Density, and Temperature - and allow players to adapt their perception, unveiling hidden monstrous threats while navigating through obscured terrains.

The device provides a dynamic mapping feature, updating in real-time as the player traverses the landscape. Essential locations, terrain types, and objective points are marked, providing a comprehensive navigational guide.

Beyond the mapping and visualization, the tablet will also act as a status informer, and as a random wisdom generator (like a fortune cookie feature). Both must be explored further later.

4.4   The Day Cycle, and Engage vs Meditate

The player will engage with the game in cycles: meditative/relax phases, and engage phases. During the meditative phase (early morning and before bed at night), the player will meditate with spiritual scripture, discuss with fellow team members, see cinematics, and more. Their actions here will influence the lore they're exposed to and possibly even status effects.

During the engage phase (brunt of the game), the player will traverse in one of two ways: split scouting or group scouting. In group scouting the player takes the two AI (brains and brawn) with them into the nanotech 'fog' and forest. They must scout new terrain and attempt to accomplish their goals. This mode of scouting will be slower but more resilient. We should attempt to make the AI engage with the player realistically and logically, taking on combat formations during fights, walking and talking casually during down time, and/or running off nearby to engage with the environment as they see fit. The split scouting mode

will see the player pinning exploration points on the map for the AI to travel to, while they themselves move out individually. The AI can't travel unrestricted, and like the player will be limited by the battery power they have; each pin will come with a percentage likelihood of living (based on some algorithm we'll make that considers distance, terrain conditions, etc). The AI will scout in the same time it takes the human to go and return, and will mark interesting locations on their map. It's possible the AI gets captured, at which point it's up to you and the remaining crew to save them. I may make it so that if you don't save them within some time they die permanently. In its current state, the scouting method is only individual.

The passage of time must happen somewhat seamlessly. I imagine we'd have the player wake up, perform some potential routine, and then when they're ready to leave a black change screen, jumping to them packing out. They will scout at the cross of day into night, such that half the level is played in each. They must return to base camp within their allotted time (which we'll associate with the battery running out). When they return another black screen transition suggesting a wind down. They will finally have the opportunity to participate in a night routine before they sleep and pass to the next day. As I've mentioned elsewhere there may be a dream cinematic after sleeping.

4.5   Sleeping, Dreaming, and Reading

Every time you finish a scouting cycle you can choose to read, eat food, interact with your team, and then you must perform a sleep cycle. Depending on your actions, location, resource status, or other factors, you'll be presented with a cinematic. The cinematic may be an innocuous warning confronting you about brash gameplay, it might provide insight to the future, it may foreshadow, or improve your spiritualism.

Reading is one of the many actions that will improve dream performance. Reading will also influence gameplay, mostly positively. The readings themselves feed into the story to an extent. The player can either choose to read according to a schedule (which we will establish and which will inform the story), or 'open randomly', (which like the dreams will inform or foreshadow). Whether the player chooses to read themselves or not, it will need to factor into the story via cinematics or otherwise. The AI synchronized nanotech will 'learn' by listening to the reading the player does, not actually, just in the narrative.

4.6   Adrenaline Rush

Adrenaline will be a binary feature, either being on, or off. The player will have some default fall or mistake percentage that governs when they trip or mess up (maybe a shot) on the scouting mission. When the player is being chased by the monster, this default setting will be multiplied or nullified in various ways, simulating the effects of adrenaline. After the player exits the monster's significant range, their adrenaline will turn off.

Proper eating habits, sleeping, reading scriptures and meditation, and talking with friends will improve the player's performance, in both regards. Get bad at everything and you'll be tripping and missing shots frequently.

Parameters that will be affected are damage taken (less during adrenaline

rush), the blur around the screen (visibility), possibly random blur, shot steadiness/miss chance (or just success of hits), fall likelihood.

4.7 Weaponry

DRG is a pretty prototypical example of what I mean by this: 4 classes, each semi-customizable with their own set of weapons. The user should be able to customize the weapons of all members of their party and play how they'd like. Certain weapons might be better for stealth, AoE, direct fire, etc, and improve the players chances at success. This doesn't mean we'll have massive variety and precision in design, more like they can pick from one of three options.

Your gun will either be drawn or holstered, the two states impacting movement speed. Aiming down sights will also impact movement speed. This is done to play into the fight interactions with monsters: the player has to strategize when to shoot and what impact this will have on their escape. There might be some pause as the gun is being drawn or aimed.

4.8 Resource Management

No part of these mechanics have been implemented. As I said before, they sit here for reference to what will happen with the future of the game.

To keep the game simple the only resources you'll need to manage are battery, munitions, sleep, and health. Battery refers to both the suit battery and the camp battery; the former dictates how long you can scout, the latter dictates how many days you have to accomplish your mission. We must come up with a good idea for the generator equipped to big dog, which with minimal fuel can reliably power a mission this long. Munitions I suppose are another form of battery, being that your gun is an energy weapon. Sleep is crucial to survival as you'll perform poorly and be exposed to less lore without quality sleep. Health is understood in a traditional sense.

The battery can be easily leveraged as a component of the difficulty. Harder difficulties give you less battery overall, and make your suit and weapons drain more. I'm also imagining we could make different actions sap your battery much quicker: running vs walking; fighting vs sneaking, etc. The battery of the gun could also be dynamic, depending on how you shoot will impact the battery consumption. All batteries will be recharged at base, so by limiting how much charge you get from base we can increase difficulty.

Sleep can be improved by engaging with spiritual content, eating the right foods at the right time, and talking with group mates. Talking extensively or not at all, eating right before bed, and disregarding spiritual content will give you the worst possible sleep. Eating first, talking moderately, and consulting with a spiritual routine right before bed will grant good sleep and possibly even dreams (which would just be cinematics). Sleep may also contribute to the luck and coordinate of the player on the field. By coordination we may have the player trip or make more mistakes with poorer sleep.

Health is as you'd expect. It's a bar that has a maximum, and taking damage from the environment lowers it. When the bar runs out you die. We could implement save points, and then have a hardcore, no death mode. Hard mode might allow you to die a certain amount of times - for which your team

will 'save' you - but after which you die permanently. I think it's important to have a permadeath mode to incentivise people to engage in a challenging play.