

CSCI 265 Initial Project Proposal

Team name: We Be Daves

Project/product name: See a Neevle, Hear a Neevle

Contact person

- Dave Narealdave, nareal@somewhere.ca

Project Overview

See a Neevle, Hear a Neevle (SaNHaN hereafter) is a team multiplayer puzzle/combat game, played on a top-down 2D map. Players on a team have to cooperate to survive and move on, as the makeup of each being means they are unable to sense certain other things, effectively meaning each player has some things they cannot see and some things they cannot hear. The game is best suited for teams that either communicate/cooperate very well or teams that have a sense of humour when things go hideously wrong.

The premise of the game is that players are inhabitants of the planet Neevle, where all life forms are energy based. There are three core forms of energy, Gamma, Zeta, and Theta (*those names are tentative and likely to change before release*) and life forms of each core type are unable to see beings of one other type and unable to hear beings of one other type. (*Gamma cannot see Zeta or hear Theta, Zeta cannot see Theta or hear Gamma, Theta cannot see Gamma or hear Zeta*).

Players on each team are evenly divided across the three core energy forms, thus each player is unable to see/hear certain players on their own team and unable to see/hear certain enemies. This means players on a team need to coordinate with each other to survive/overcome opponents. Similarly, puzzles (involving gates, switches, locks, etc) revolve around specific energy bases, meaning players need to coordinate to solve puzzles that permit moving deeper into the game world.

To summarize the impact:

- you can see and hear anyone (friend or enemy) who is of the same energy base but you cannot hurt them (and they can see/hear you but not hurt you)
- if you can see anybody else then you will not be able to hear them, while they can hear you but not see you (and it'll hurt if you bump into each other)
- if you can hear anybody else then you will not be able to see them, while they can see you but not hear you (and it'll hurt if you bump into each other)

The objective is to either become the first team to reach **THE GREAT CORE**, an energy source granting virtually limitless power to the first team to reach it. Alternatively, a team can win by eliminating all other teams in the game, giving teams the choice to win by success in combat or by evading combat and rapidly solving puzzles to leave your opposition behind.

Target audience and motivation

The game is intended as a relatively light-hearted top-down 2D team game, with a heavy emphasis on team coordination and a mix of combat and puzzle solving elements. It is expected to appeal to groups who either cooperate effectively or who can find the humour in watching their friends perish to things they can't see coming. *(We all have friends like that, you know who you are!)*

Everyone on our team is an avid game player and we have all followed discussions/videos discussing the design strengths/weaknesses of our favourite games, but none of us has yet attempted actually designing or building an actual game. We're all looking forward to the project as a way of learning more about the ins and outs of game development from an applied perspective.

Targeting a multiplayer game right out of the gate adds a big extra layer of challenge, so we'll be keeping many of the other game aspects simple (relatively simple puzzle/combat rules, maps, and story progression). We still hope the basic nature of the game will render it fun for casual players.

The game controls are currently expressed in keyboard form, but console or touch-screen versions and controls might be considered as a long term stretch goal.

Key features and discussion

As mentioned earlier, the game will consist of opposing teams of players working their way through a map, fighting opposing teams (and some other hostile inhabitants) and solving puzzles in the hopes of being the first to reach **THE GREAT CORE**. With that in mind, some of the key features/aspects of the game are broken down below.

Teams

- initially SaNHaN will target two teams of three players each
- one player will host the game, the other five players joining as clients
 - the host chooses game settings such as map, friendly fire, and whether or not to apply the rules regarding what players can/cannot see/hear
- each team has one slot for each of the three core energy types
- each team will have a unique symbol to represent its players, and each energy base is assigned a colour (thus the symbol+colour tells you which team a player is on and what energy type they are)
- a secondary goal will be to fill empty player slots with NPCs
- a stretch goal will be to support more than two teams and more than three players per team

Energy levels

- each being in the game has a current and maximum energy level (somewhat akin to health)
- energy is expended in combat by taking damage (really an energy drain) or firing projectiles (see below)
- energy slowly replenishes over time
- energy can also be replenished by item pickups during the game
- all beings have a critical energy level, below this level they cannot take any action until their energy replenishes above the critical level
- if a beings energy level reaches zero they dissipate: for player characters this means they can no longer participate in the game (but can continue to watch if desired)

Maps and movement

- there will be a single large map with numerous walls, items, obstacles, traps, puzzles, creatures, etc
- each team will start in an isolated section of the map, getting practice with some puzzle solving before reaching a common area with a mix of puzzles and team-v-team combat
- stretch goals include having multiple maps for the host to choose from when setting up a game and/or having procedurally-generated maps
- players move around the map using WASD controls, with the player facing the direction of the most recent movement (up/down/left/right)
- players can look around the map without moving their character using the arrow keys, zoom in/out using +/-, and center the map back on their character with the c key
 - the ability to 'look around' the map will be partially limited to prevent players from getting sneak peaks at what is further down the road

Sounds and hearing

- ultimately the goal would be to have high quality directional headphone sound, but this is likely far outside the scope of what our team can tackle this semester
- our alternative for the initial version is to display sounds in a subtitle field
- to capture the sound/puzzle elements of the game, a circle will appear on the map indicating the rough area the sound is coming from
- the circle is smaller when the source of the sound is closer, giving the player a clearer idea of where the sound is coming from
- a later enhancement might be to allow different sound volumes, e.g. a player could yell, talk, or whisper, letting players attempt to communicate without being overhead by enemies or alerting possibly hostile creatures

Combat

- whenever two beings of different energy bases come in contact they will both experience an energy drain, hand to hand combat thus amounts to running into another being and hoping they take drain faster than you do or that you have enough energy to outlast them
- players can also use the space bar to periodically fire energy projectiles that drain beings of other energy bases (projectiles travel in the direction the player is facing, and stop when they hit something of a different energy base)
- when setting up the game, the host has the option of turning off friendly fire (so you can't accidentally clobber or shoot an ally)

Puzzle ideas

- throughout the game there will be a variety of traps, triggers, gates, switches, pads, etc that the team must avoid or pass through to reach the next map area
- puzzles may consist of multiple components (e.g. several switches that must be flipped to open a gate), each of which is keyed to a specific energy type
- some components might only be detectable by sound, some are visible but make no sound, and the usual energy-base rules apply (which energy types can see/hear which others)
- *puzzle example:*
 - a long wall contains a gap that the players must go through
 - an energy gate bars the gap and two energy pads are nearby, all three items are of different energy types

- two players must stand on the pads while the third opens the gate, but they all must be on/at whichever item is invisible to them (i.e. the other two players must talk them into the correct position)
- *(of course, a similar puzzle might appear later where players must be on/at the one that is visible to them or they get zapped, or another puzzle might be sound-based rather than visible...)*
- there could be long winding pathways the players have to traverse with sections that are invisible to certain energy bases
- there could be gates to shortcuts to bypass certain areas/obstacles, where a neutral guardian provides the first person to arrive with a password to get through the gate. That person can tell the password to others on their team, but might risk being overheard by the opposition. (Or possibly a stronger enemy could spare them from destruction in return for the password?)
- there could be gates or passages that cycle through the energy types, you can only go through while its current energy type aligns with your own
- there could be friendly creatures that guide you to passageways or valuable items (or unfriendly ones that are actually guiding you to a trap)

Levelling up

- players gradually accrue experience as they solve puzzles and succeed in combat
- each time players reach set experience thresholds they are able to improve their ability in one of several areas:
 - increasing their maximum energy level
 - improving the rate at which they regain energy
 - increasing the amount of energy they drain from an opponent in combat
 - improving their control so they drain *less* energy from allies

Items

- items will be scattered around the map (randomly or at preset points)
- each item is energy-based, so only visible/audible to the appropriate energy bases
- common items restore a portion of current energy
- rare items might increase a player's experience level
- trapped items might be disguised to look like restorative items but actually drain energy
- lore markers might provide backstory for the planet and its inhabitants (possibly also providing experience?)

Creatures

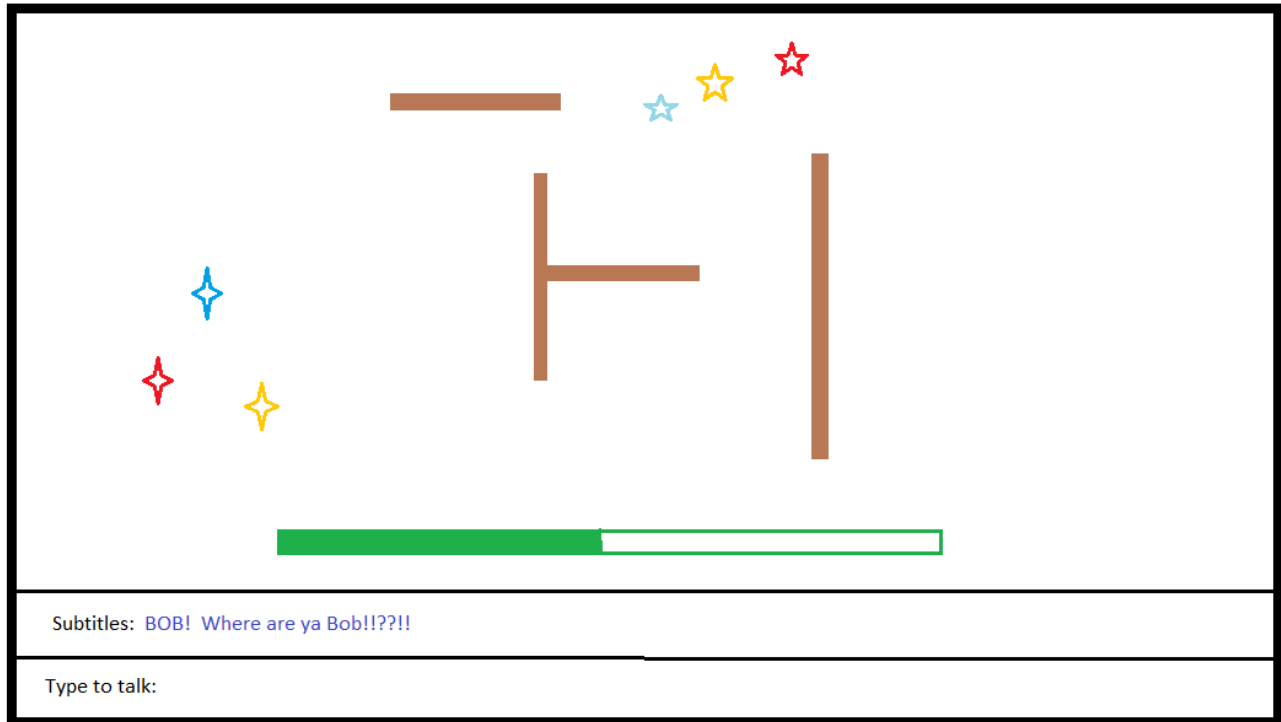
- a variety of NPC creatures will populate the world, some at fixed points and some randomly wandering
- these creatures (as with everything else) are of one of the three energy types
- the creatures will have different levels of hostility, intelligence, current and maximum energy

Preliminary interface sketches

Mock-ups of the main map and some of the menus are shown below, though these are to be treated as *very* preliminary and will certainly change as we get deeper into the design process.

First sketch of the combat screen

The main interface is shown below, depicting a map segment with two teams (represented by the different symbol types), the different energy bases (represented by colour), some wall segments, the player energy bar (in green), a subtitle segment and a place for the player to type messages (though the message will only appear to the team member that can 'hear' them)



Early menu mock-ups

Early drafts of four key menus are shown below:

- the start menu a player sees when opening the game
- the options menu presented to configure a game
- the help menu to provide user guidance
- the pause menu (probably a poor choice of name since this is a multiplayer game and likely should not pause the game for everyone?)

<p>HELP MENU</p> <ul style="list-style-type: none"> Gameplay Map Combat Gameplay Critter catalogue Puzzles 	<p>PAUSE MENU</p> <ul style="list-style-type: none"> Resume Help Options Quit Upgrades 	<p>OPTIONS</p> <ul style="list-style-type: none"> Control config (A) Subtitles (on) Friendly fire (off) See/hear all (off) 	<p>START MENU</p> <ul style="list-style-type: none"> Options Host game Join game
---	--	---	--

Map construction

The preliminary plan is to have a single large map that is segregated into sections (levels?) by long walls containing a single gate/gap to the next section. Within each section there will be creatures, traps, puzzles and items (some common to all sections, some unique to that particular section).

In the default map, the teams will start in separate sections while they master the puzzle solving techniques and basic game mechanics. Once they get through the opening segment they will enter a shared section where team-vs-team combat is an option.

Depending on the progression of design and play-testing, from there the teams may move from section to section, sometimes separating the two teams into areas unique to the team and sometimes in mutual combat situations. To avoid premature ends to the match and promote the plot-line, it is possible that sometimes the teams will be forced to end combat and move on to separate sections before meeting again later.

The number of sections in the initial map and the depth of the plot-line will depend on how the overall development process goes: if the team is making good progress then we'll be able to devote more time to extending the map and the challenges within it.

Scaling options

As this is our first attempt at game design and development (and a multiplayer game at that!) there is a great deal of uncertainty around what we'll be able to successfully complete.

As a result, we have identified a core set of features we absolutely want to complete, a secondary set that we certainly hope to complete, and a set of stretch goals we can work on if/when everything else is finished.

Core aspects

- support for six human players divided into two teams, where the host provides the players with the necessary information to connect to them (IP and port?)
- one fixed map with at least several traps, obstacles, creatures, pickup items, and working puzzles
- the combat and energy-recovery system as described in the key features section
- at least two 'levelling up' opportunities for the players on each team
- if the host drops then the game ends, if another player drops they are 'dead'
- players are randomly assigned to the teams/energy slots

Secondary features

- a wider variety of maps to select from, with differences in size and difficulty
- a wider variety of puzzles, traps, items, and creatures
- configuration options to turn on/off friendly fire and turn on/off energy based hearing and visibility
- a mechanism for the host to automatically communicate the necessary connection information
- options for the host to impose some form of throttling on update rates to deal with potential networking issues
- filling empty player slots with AI (allowing the game to be played with 1-6 players)
- either the host or the joining player can pick their team/energy slots (first come first served?)

Stretch goals

- procedurally generated maps
- support for more than two teams and more than two players per team
- allow hosting to switch to another player if the host drops and allow an AI to take over if client player drops
- good AI control of NPCs
- allow the host or joining players to supply avatars/map images
- add more variety/depth to the combat system

Risks and potential issues

Because we are all new to this there are many risks and potential issues...

From the pure playability aspect, we aren't sure yet that the game design will be fun for a wide range of people. Hopefully, however, we can bring our own sense of what is/isn't fun to the table and come up with something that is genuinely fun, and the plan is to conduct play-testing exercises with mock-ups of the game over the next several weeks to catch any glaring weaknesses.

With respect to actual game programming, as we get further into the process we'll need to pick a set of game development tools/languages that provide us with the features we need (including multiplayer/networking support) without overwhelming us with the amount we need to learn to use them effectively. To alleviate this somewhat, each member of the team is going to spend time over the next couple of weeks investigating one or two of the following tech options to see if it seems like a good choice for us:

- Unity
- Unreal5
- Gamemaker
- C++ with an ascii/terminal format or with gtk for a graphical format

- Python with Turtle or Pygame
- Javascript/nodejs/html5/webgl

With respect to scope/scaling, the competing risks are that either we bite off too much too soon and wind up not getting anything working or that we attempt too little without an adequate plan for expansion. We believe that the core/secondary/stretch goals outlined in the previous scaling section will allow us to reach a good balance here.

(Risks and potential issues with respect to the team composition and functionality are handled in the [team charter document](#).)