

CS179N Project Report - Team SLAKN

Lacey Kisselburgh, Kiana McCullough, Noah Christensen, Salvador Esquivias, Arturo Perez

Overview

Don't Die is a 2D survival RPG where the player must survive in the forest for five days until a rescue party comes to save them. The player is free to explore and make use of natural resources around the map to survive.

Game Description (e.g., levels, characters, game play)

The player is stuck in the forest with only a tent, fishing pole, and an axe. Using these materials and the environment, the player must try to survive also while in enemy territory. There are three kinds of enemies: bats, wolves, and bears. In order to survive the five days, the player must keep track of their health, hunger, thirst, as each of these deplete little by little over time. The temperature fluctuates depending on the time of day and can hurt the player if it gets too cold. The hotter the temperature, the more the player's thirst decreases. The player can observe these stats from the inventory. The inventory can be pulled up by pressing "space". From here, the player can see their stats and the items they have so far. The player initially starts out with a tent, fishing rod, axe, one berry, and one full canteen of water. The inventory will update as the player collects items.

Hunger and health can be kept in check by eating berries and meat. Berries are scattered across the map and can be collected by walking up to it. They restore a small amount of hunger and health. Meat can be collected from enemies or fish. When an enemy is successfully beaten, they will drop meat. Meat restores more health and hunger than berries do.

Thirst levels can be maintained by collecting and drinking water. The player can fill their canteen by going in any body of water and pressing "F". A maximum of 10 drinks can be kept in the canteen at a time.

Player temperature can be: freezing, very cold, cold, normal, hot or very hot. To maintain ideal / normal temperature, the player can build a fire. Fire can be built from the inventory under "wood". Building a fire requires three pieces of wood. Keeping warm is especially important at night. At night, when the player is away from a fire, their body temperature is very cold. If they

are very cold for more than 20 seconds, their body temperature drops to freezing. If they are freezing, their health slowly depletes until they get to a fire. Once near a fire, their temperature will be just cold. Fires will disappear after 20 seconds, as the fire goes out.

The “levels” are days to survive and there are five in total. The days run on a day / night system. It is harder to survive at night as the player’s temperature becomes more susceptible to the cold. At the end of each day, the player can use the tent to sleep and proceed to the next day. Only one tent can be set up at a time. When the player sets up a tent, the player’s position is saved and can be loaded at any time in-game. The player can do this via the pause menu or by pressing “L”. The player can also survive through the night without sleeping in the Tent if the player chooses to. A prompt on the screen will just appear letting the player know it is the next day.

The player can fish at the ledge of a body of water. Pressing “J” will cast the line and when a fish is hooked, the player must keep pressing “J” until the bar is full. If the player fails to get the bar full after a certain amount of time, the fish will escape. Successful captures of fish will show in the inventory as meat. The player can also fish whilst on a boat. Building a boat costs thirty pieces of wood. After selecting “build boat” from inventory, the player will see the boat icon in their inventory. Now, if the player goes into the water, they will be on a boat. The boat is much faster than swimming and can get players across water quickly.

Combat system. The combat menu occurs when the player is close enough or “touching” an enemy (bat, wolf, or bear). Both the player and the enemy are frozen and the player can see both their own health and the enemies health. The bats have the least health, then the wolves, then the bears and the player. The objective for the player is to finish off the enemy or flee before they are finished themselves. The player has four options to choose from. These options are: Charge, Brace, Intimidation, and Flee. Charge is based on a random number generator for both the enemy attack and the player attack making it high risk high reward. Brace uses set values for the attack on both sides making it often a safer option. Intimidation has a random chance of stopping the enemy for the turn it is used and the following turn, but failing does nothing and results in a lot of damage to the player. Finally, fleeing is given as an option to escape a fight the player is unlikely to win and stops the enemy movement for a short period if it succeeds.

The enemies were generated by a spawner. The spawner creates 10 bear clones, 30 wolf clones, and 30 bat clones randomly in the map. We spawn such large numbers since our map is so large. The player needs to be able to encounter enemies regularly, but not too often. These numbers are a nice balance so that the player does not have to hunt for enemies.

The player is also given a skill menu that they can access at any point. The skill menu is based on the combat system and gives the player the option to improve any of the combat options. The player can improve the given combat options through Charge, Brace, Intimidation, and Luck(Flee). Finally there is a defense option which mitigates damage taken for all the combat options. Each skill has four levels and the player has to unlock the previous level to continue leveling the skill. Skill points, which are used to level the skills, are given upon defeating any enemy and each skill level requires a different amount of points. As an additional feature, upon highlighting anything in the skill menu, info on the highlighted thing will appear in the info box to provide clarification to the player.

Implementation (e.g., game engine, scripts, algorithms, tools)

We used Unity engine and its collaboration feature Unity Teams for our game, which required all of us to get a Unity Pro subscription through our UCR emails. Unity uses C# so we extensively implemented C# scripts attaching them to the necessary game objects within the game scene.

Project Post Mortem

What tasks were accomplished?

From the scrum sheet, the tasks / features we completed are: setting up unity, main menu, pause menu, creating and integrating the main character, creating and integrating enemies, map creation, RNG encounter system, inventory, save feature (tent), load feature, day and night cycle, player ability to swim, player and enemy health bars, collecting items (meat, wood, berries, etc.), boat crafting, game clear objective, skill system, temperature system, building fire, game over screen, music, fishing, combat system, settings, story intro. Some of these tasks / features were broken down into smaller tasks / features. Some tasks / features from the scrum sheet involved fixing or improving existing tasks / features.

What planned tasks were not done?

We were able to finish all of the tasks on our scrum sheet and they were, for the most part, bug free. However, we had discussed adding features such as game difficulty. This would have adjusted how quickly hunger and thirst deplete, and how many days you must survive. This would also affect variables such as enemy stats and player health. We also would have liked to redo the map. In the end, we weren't satisfied with the aesthetics of the map and would have liked to create something from scratch. We also discussed that, if we had time, we would have liked to make the main character customizable.

How did scrum work for you?

The scrum sheet worked well for the team. The scrum approach helped us keep track of the tasks / features that must be completed and tested. It helped us stay organized and on track. Each person was able to pick a task that they were either familiar with or interested in. Ultimately, it helped create a better group environment and let each teammate feel more motivated about their contribution.

What would you do differently?

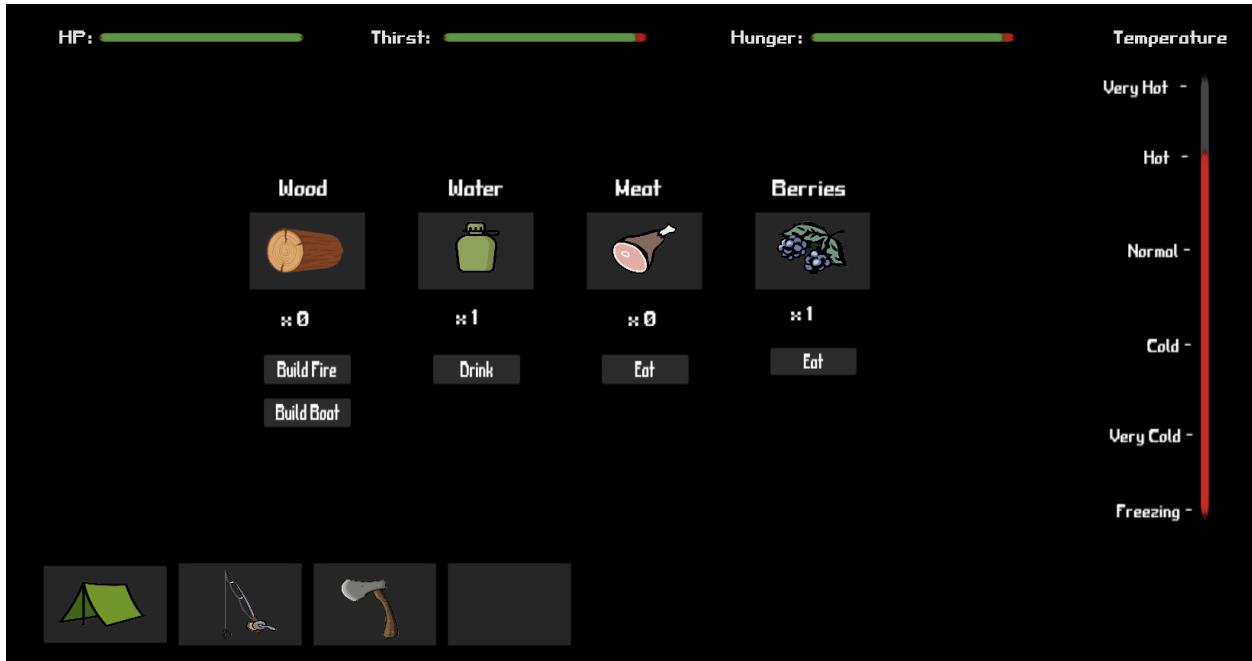
If we had more time, we would have liked to create our own assets. We do not own any of the assets in our game but some have been edited / modified to fit our game. Art can be very time consuming and so our team focused more on the gameplay aspect. In our game, we have music for day and night, victory, and game over. If we had the time, it would have been nice if we created our own music. We also would then be able to have a consistent aesthetic across all of our menus and the game.

Link to Video

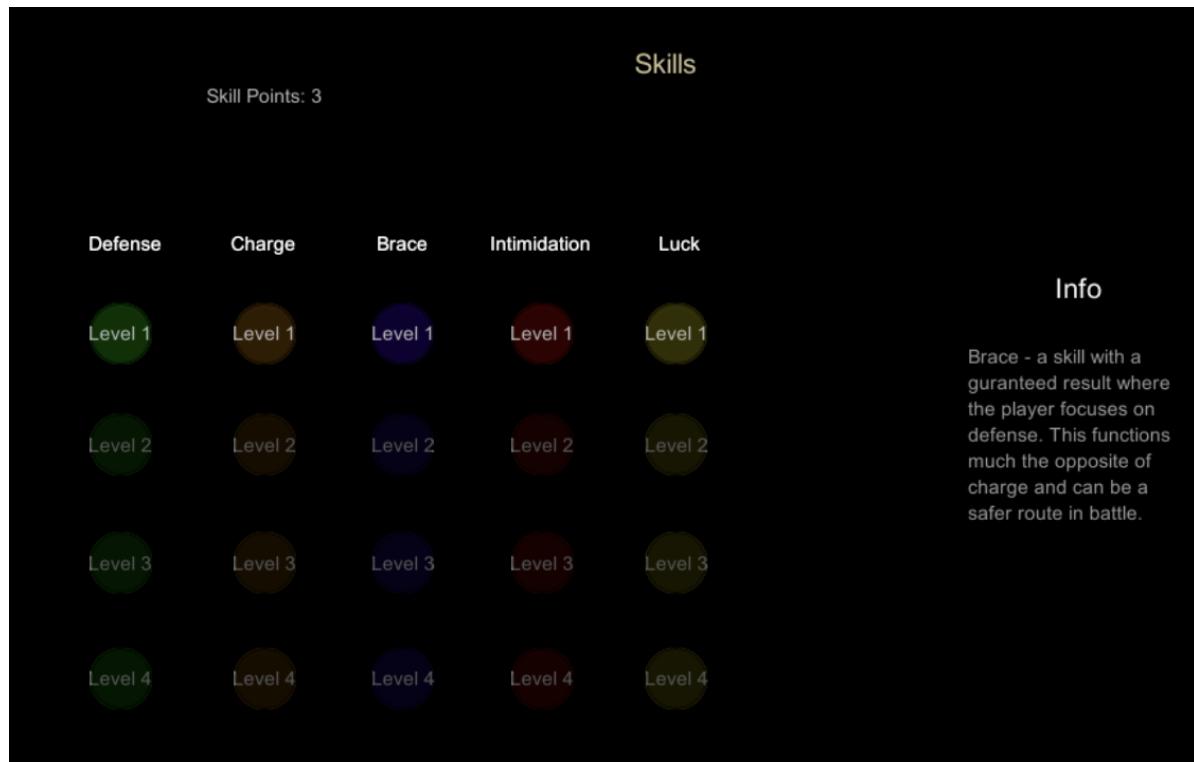
[Don't Die Recording - CS179](#)

Images

Inventory:



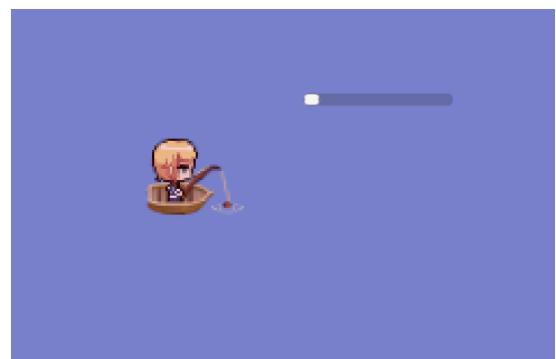
Skill Menu:



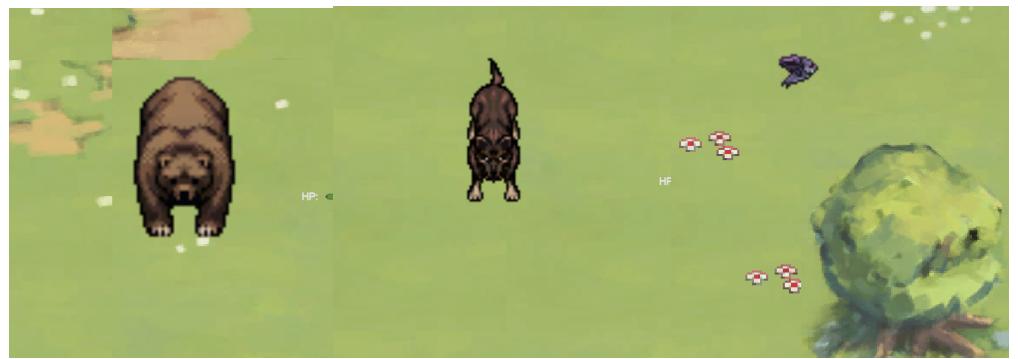
Fishing at ledge:



Fishing on boat:



Bear, Wolf, and Bat:



Full Map:

