

Thursday, October 30th, 2019.

Gordon Gray

Sabine Rosenberg,

Cart 351.

NetLife prototype

Netlife is a temmachi inspired game where the user must take care of an ill pet on its deathbed. The user is tasked with ensuring that the pet passes on peacefully without suffering when it comes to the end of a pet's life. The user acts like a vet who removes virus that appear within the lungs heart and brain. These viruses appear occasionally every half hour and do damage to the netlife pet every hour. Heath does not recover when virus are removed as the intention is to have the pet's health in decline. Therefore, the length of the pet's life cycle will be determined on how frequently the user returns to the game. This is also intentional as I want keeping the pet alive to be a frustrating task and demaing on the user's time. The game is supposed to challenge the user's commitment to how long they are willing to take care of their pet.

At this stage I have completed most of the core mechuques for the game.

- Creating a random animal
- navigating between the animals lungs, brain and heart
- spawning viruses with in the brain/heart/lungs
- Damaged colcatatlion.
- Death window.

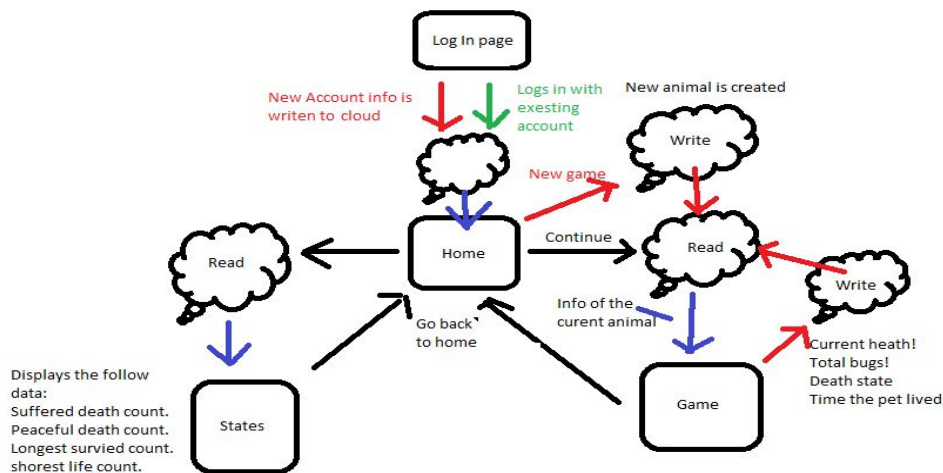
What is left to do is the following:

- Login page.
- Storing the animal to the cloud.
- Storing heath to the cloud.
- Storing big count to the cloud.
- Increase the bug count every half hour with a real time function.
- Call the damage function with a real time function.
- Page that shows user states.

Tester feedback.

- Making removing the bugs a more teddest task so to anyone the user more.
- Feedback system to show that the user has some impact on the animal wherever they remove the bugs from the pet.
- One of the questions I had asked was if I should use real life animal images or my own sprites. Everyone I asked said sprites.

How all the data will be transferred?



Login page.

- Creating a new account will create the user data for that account.
- Logging in will gather that data from the cloud.

Main page

- New game: Creates a new animal and writes the data to the cloud before going to the game page.
- Continue: reads the data from the current animal in use and then goes to the game page.
- States: sends the player to the states page and reads the data of the account users states.

Game page:

- Reads the animal data when started up.
 - Animal type(dog,cat etc).
 - Name.
 - Hobbie.
 - Birthplace.
 - Zodiac sign.
 - Current Hp
 - Current bugs
- Writes data to cloud when:
 - A bug is removed. (lowers bug count).
 - When a bug is added. (total bug count increased)
 - When damaged is calculated. (updates animals hp to cloud)
 - When an animal dies. (removes current animal data, time the animal lived for, and death condition for the states page.
 - At the end of all calculation reads the new information to the page.
- Home button.
 - Takes the user to home page.

States page:

- Reads and display info on the cloud.

- Number of suffered death.
- Number of peaceful happy deaths.
- Deaths total
- Longest time survived
- Number of new game starts.
- Home button.
 - Takes the user to the home page.