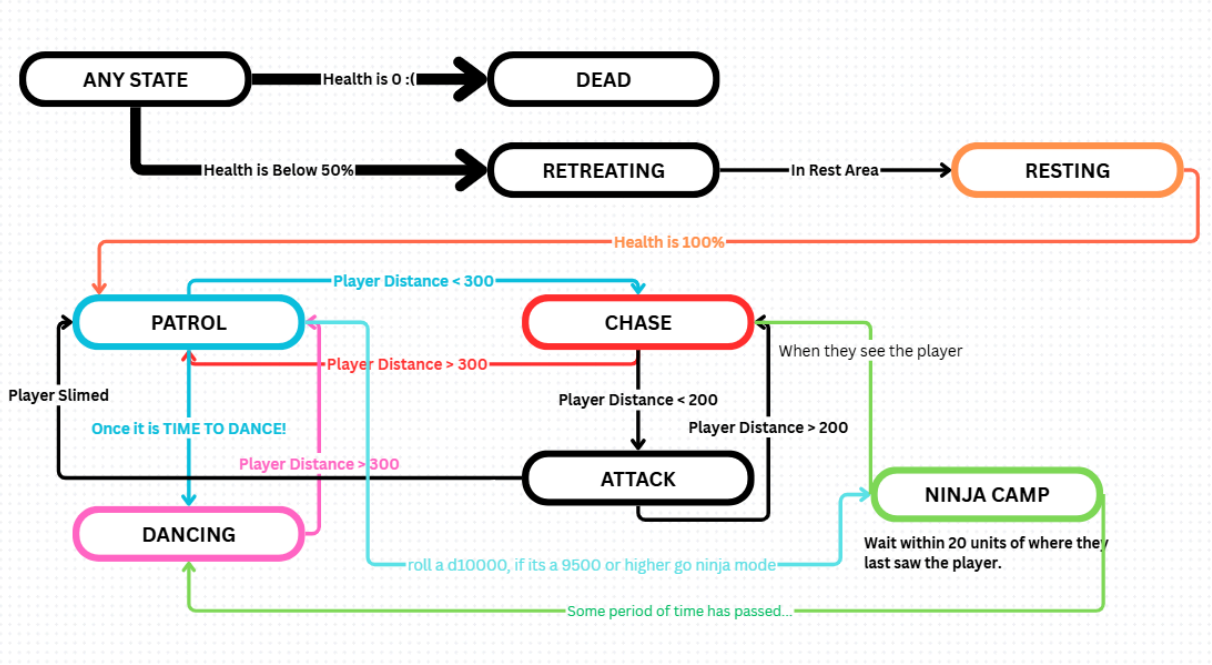

Ninja State

1. Make A New Ninja State
    - Make a Ninja state and add it to the FSMState enum.
2. Make the 'Ninja' Action
    - Make a new function for updating the ninja state. The code inside this function should define the ninja behavior.

    - Making the prediction:
    - Track the player's position each frame and calculate velocity.
    - Use a Kalman filter to predict the next position, then correct it with the actual measured position.
    - Estimate the player's future position by projecting velocity forward by a configurable time.
    - Expose this predicted position for enemies to aim or camp at
    - In the ninja action state: move to the predicted location, and upon arrival, wait there

3. Add an Entry Condition
    - Modify the UpdatePatrolState() function so that randomly the script tells the AI to switch from Patrol to the new Ninja state.
4. Add Exit Conditions
    - Modify the function for updating the ninja state to exit when the player is close. The AI stops holding position and starts attacking.

Implementing the Resting State

1. Create the physical area that the Tanks will go to heal.
    a. The area must have a Sphere Collider (trigger) that destroys all projectiles fired into it, the tanks cannot be damaged once they finally make it to the healing zone, so this is the implementation of that.
2. Make it so that when the tanks drop below a chosen “Critical Health”, they transfer to the resting state automatically.
3. While in the Resting State, the tanks travel to the Tank’s healing Zone, zigzagging to attempt to make it harder to hit them.
4. Then, once at the healing zone, they regain health overtime until they are at their starting health, then they go back out on patrol.

Implementing The Dancing State

1. Create the timeToDance variable before entering the Patrol state, this variable will be equal to the timeElapsed + a random number of seconds ranging between 2 and 5 seconds.
2. Create the StopDanceTime Variable, similar to above except it determines when the Tank goes back on Patrol assuming it isn’t interrupted by seeing the player. This variable must be created wherever the code transitions into the Dance State.
3. In the Dance Update Loop (create one, and hook it into the FSMUpdate switch statement), code a simplistic dance animation where the whole tank rotates on the Y axis (It does this each frame)
4. If ElapsedTime >= StopDanceTime, Transition back to the Patrol state, and reset both Time Variables, the StopDanceTime to 0, and the startDanceTime = elapsedTime + a random amount of time between 2 and 5 seconds.