

1. Brief introduction _/3

For the group project I will oversee the general chicken system of our game (The enemy of our game).

This would include:

Base Enemy Class

Chicken "Split" Functionality (Higher tier chicken would split into lower chicken tiers)

Enemy Health (Deals with tower damage to chicken entity)

Enemy Damage (To the player health)

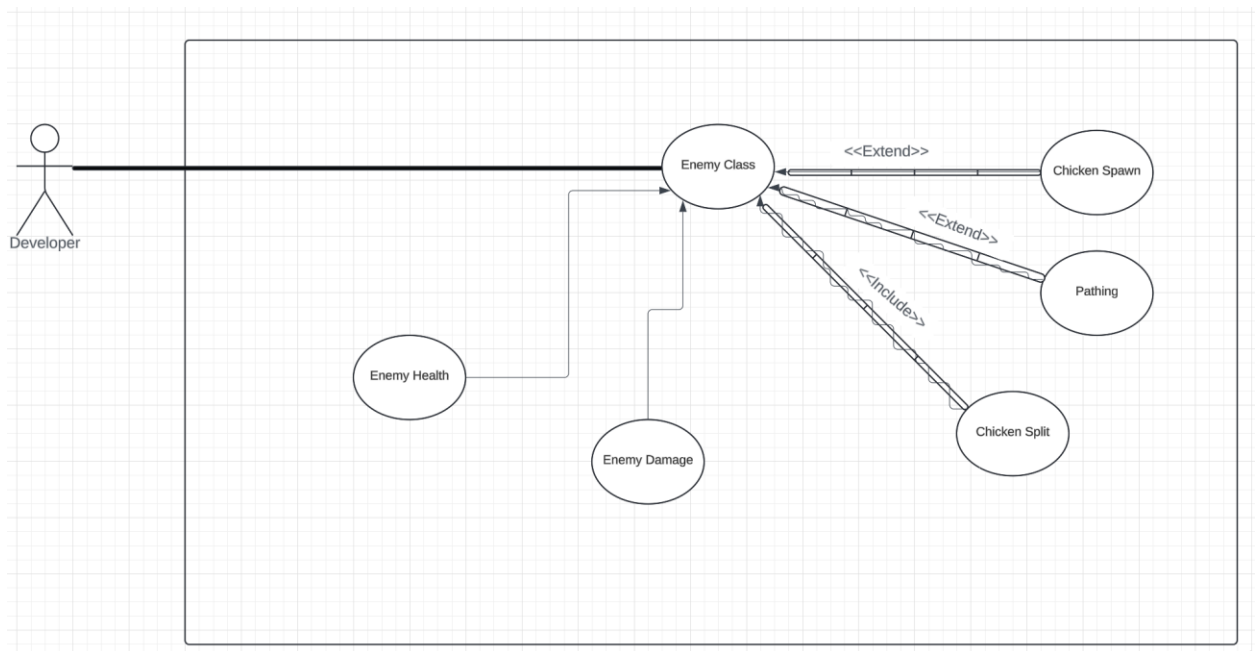
Path-finding System (Dependent on map)

Chicken Spawn (covered by Nathan for round management, but will still talk about as an extend case)

2. Use case diagram with scenario _14

Use Case Diagram

Note: The dotted Lines are represented by the 1. zigzags over bars and 2. line through bars



Scenarios

Name: Enemy Health

Summary: Defines health for various tiers of chicken entities

Actors: Developer

Preconditions: Chicken entity must be created

Basic sequence:

Step 1: Chicken is spawned

Step 2: Tier of chicken is checked

Step 3: Health is assigned accordingly

Step 4: Health is decremented according to tower damage

Exceptions:

Step 1: When chicken is not attacked by tower, health does not decrement

Step 2: Do not assign health if there is no chicken

Post conditions: Chicken makes it to the end or is killed by a tower then despawns

Name: Enemy Damage

Summary: Defines the Damage for the chicken entity

Actors: Developer

Preconditions: Chicken entity must be created

Basic sequence:

Step 1: Chicken is spawned

Step 2: Tier of chicken is checked

Step 3: Damage is assigned accordingly

Exceptions:

Step 1: If chicken does not make it to the end, damage does not matter

Post conditions: Chicken reaches the end of level, and does damage to player according to tier

Name: Chicken Split

Summary: Splits the chicken into multiple lower-tier chickens

Actors: Developer

Preconditions: Tier 1+ chicken must die so that new lower tier chickens can spawn

Basic sequence:

Step 1: Chicken Dies

Step 2: Depending on tier, the function will spawn a fixed number of lesser chickens

Step 3: It will repeat until the lowest level of chicken is reached

Exceptions:

Step 1: Chicken is tier 1 and does not split

Post conditions: Chickens will split and spawn on the path and continue until they reach the end of the level or get eliminated

Name: Pathing

Summary: Will keep the chickens on the path until they die or reach the end

Actors: Developer

Preconditions: Chickens must be spawned

Basic sequence:

Step 1: Chicken spawns in

Step 2: Chicken determines the path it can take

Step 3: Chicken follows the path until the end

Step 4: Chicken de-spawns

Exceptions:

Step 1: Chicken will de-spawn if it is eliminated

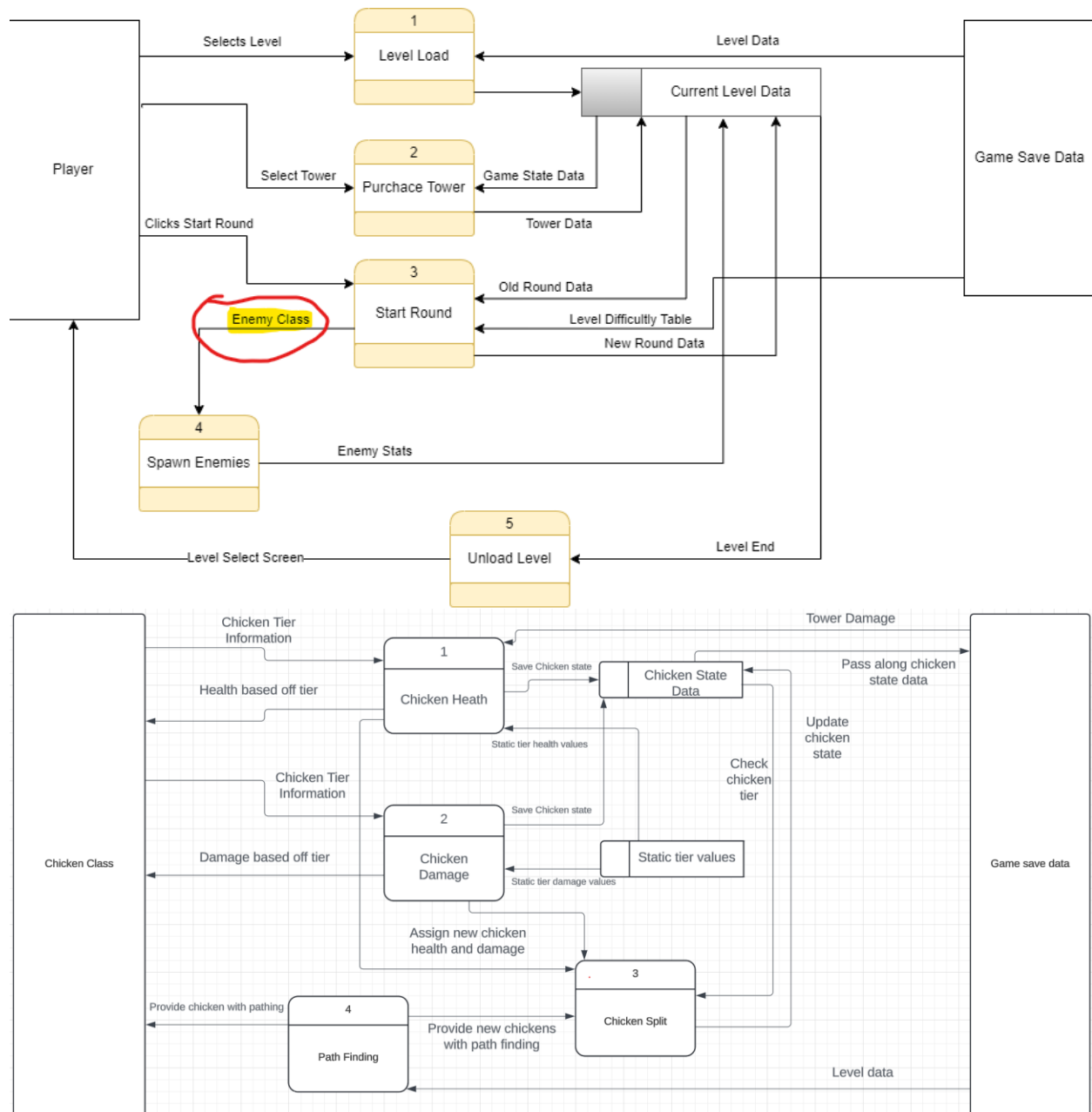
Post conditions: Chicken follows path properly

Name: Chicken Spawn

(Not my own feature, refer to Nathan's Champion for an explanation of how it will work)

3. Data Flow diagram(s) from Level 0 to process description for your feature ____14

Data Flow Diagrams



Process Descriptions

Chicken Class:

Create Chicken():

Add chicken health()

Determine chicken tier()

Compare tier of chicken to pre-assigned health value of tier;

Assign health value to new entity;

Decrement chicken health()

Check tower damage to chicken to decrement health;

If (health == 0 || chicken tier > 1)

Chicken Split()

Else

Remove chicken entity;

Save chicken state;

Add chicken damage()

Determine chicken tier()

Compare tier of chicken to pre-assigned damage value of tier;

Assign damage value to new entity;

Save chicken state;

Chicken Split()

Int Chicken split amount;

For (int i = 0; i < chicken split amount; i++)

chicken spawn(tier -1) #part of Nathan's code

ensure that new health and damage values are assigned to each new chicken
spawned

Chicken pathing()

#Mostly will be done through the game engine without the need for much script

Determine path of map using level data

Pass path information to chicken entities using Chicken spawn() function

Check that no towers are being collided with, and that no chicken is off the path

If (chicken off path)

Remove chicken;

4. Acceptance Tests _____9

Chicken Split feature:

Spawn in 1 max tier enemy 10+ times

The following should be the case:

- Appropriate amount of lower tier chickens are spawned in
- Pathing features work properly for newly spawned chickens
- New chicken health and damage are properly assigned
- Towers can attack new chickens

Other tests that need to be conducted

- Perform stress test on the system to determine max number of chickens before crash

Enemy Health Feature:

Spawn in 5 chickens 10+ times

Following should be the case:

- Health values should be equal amongst all chickens of same tier
- Health should decrease when attacked by a tower

Enemy Damage Feature:

Spawn in 5 chickens 10+ times

Following should be the case:

- Damage values should be equal amongst all chickens of same tier
- Player health should decrease at rate of damage per chicken when they make it to the end of the path

Path Finding Feature:

Spawn in 5 chickens 10+ times:

Following should be the case:

- Chickens spawn and stay on the path until the end of the level.
- Chickens disappear when they reach the end of the level
- Towers and chickens do not intersect or overlap
- For Chicken Split function -
 - New chickens follow the path

Other tests:

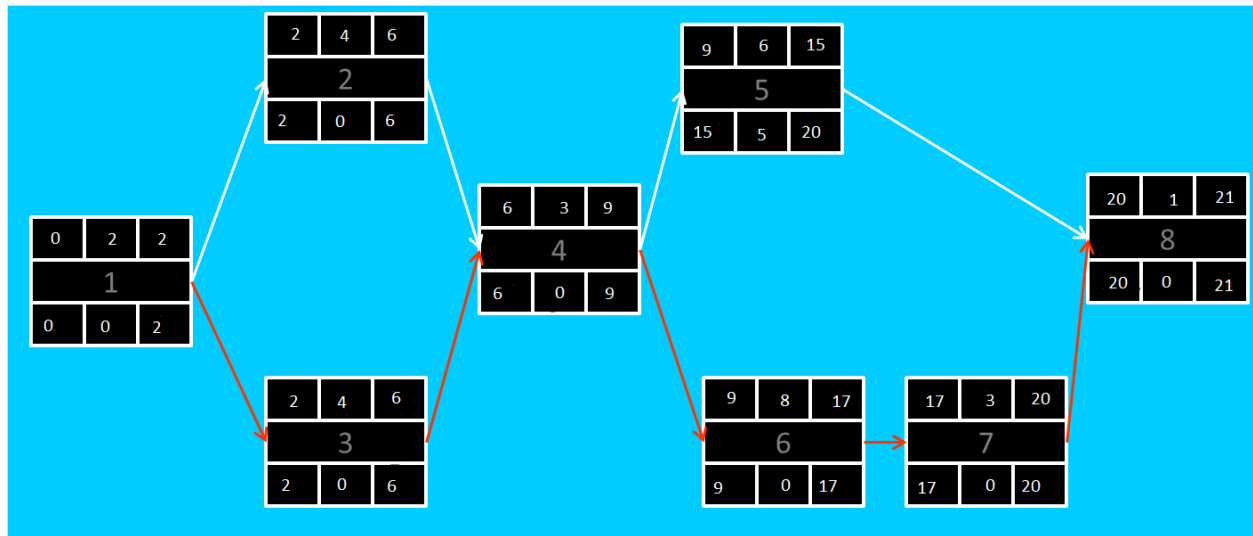
- Spawn in 50+ chickens at once to determine if pathing will work with significant number of enemy entities.

5. Timeline ____/10

Work items

Task	Duration (PWks)	Predecessor Task(s)
1. Define needed Functions	2	-
2. Collab with team	4	1
3. Organization of workspace / File system management	4	1
4. Character design	3	2, 3
5. User Documentation	6	4
6. Programming	8	4
7. Testing Features	3	6
8. Installation / Final Integration	1	5, 7

Pert diagram



Gantt timeline

