Farm Battle Project Proposal

**Description:**

1. Name: Farm Battle
2. Genre: multiplayer grid-based PVP
3. Game Play:
   1. Two Parties controlled by multiple players
   2. The farmer:
      1. Knock the moles when they show up
      2. Support multiple farmers
   3. The moles:
      1. Player click on the fields to make a mole pop up
      2. Stay x seconds before disappearing (x related to the level of difficulty)
      3. If being hit, dazzle for y seconds and disappear (y related to the level of difficulty)
      4. Each mole player can control multiple moles or Us game AI to play supplement
   4. Battle:
4. 2 minutes campaign
5. Scoring:
6. When farmer hit a mole, farmer score + 1, hit on dazzle moles gains no scores
7. When mole waits for x seconds and disappear without being hit, mole score + 1
8. Map: randomly generated mix of field tiles and grass tiles (Moles only shows on the field tiles)

**Competitive Analysis:**

Whac-a-Mole is a popular arcade game. It was invented in 1976 and evolve through times until now. The core game mechanics remains the same. The moles pop out in the game map and the player hit the moles when they pop up to earn scores. When the game gets harder, the mole will remain up for shorter time, which requires the player move faster. There is multiplayer version, when multiple players play together and compete on the number of moles they hit.

Farm Battle is inspirited by Whac-a-Mole, but different in three points:

1. The player can choose to play the farmer or the moles. The competition would happen among the farmer players and the mole players.
2. Cooperation mechanics. The game would support multiple farmers vs multiple moles. Players in the same party should work together to fight against the other party
3. It made us of the network so that the players can play on their individual machines.

**Structural Play:**

1. Three new classes:
   1. Gamemap:
      1. Data:
         1. the dictionary recording the tiles of the map
         2. positions of the UI elements
         3. Scores and game states
         4. Player mode (farmer/mole)
      2. method:
         1. Randomly generate the map
         2. Convert the mouse position to fit in the tile
         3. Game state check
   2. Moles
      1. Data:
         1. Number of the moles a client can handle
         2. List of the position of all the moles
         3. The time showing up for each mole
         4. Sprites of the moles
      2. method:
         1. Check the viability of the clicks
         2. Check the hit
   3. OpeningScene:
      1. Data:
         1. Options/buttons
      2. Method:
         1. Load different mode based on the players’ selection
2. Other Functions:
   1. Functions to handle synchronizations
   2. Functions to draw the visual elements

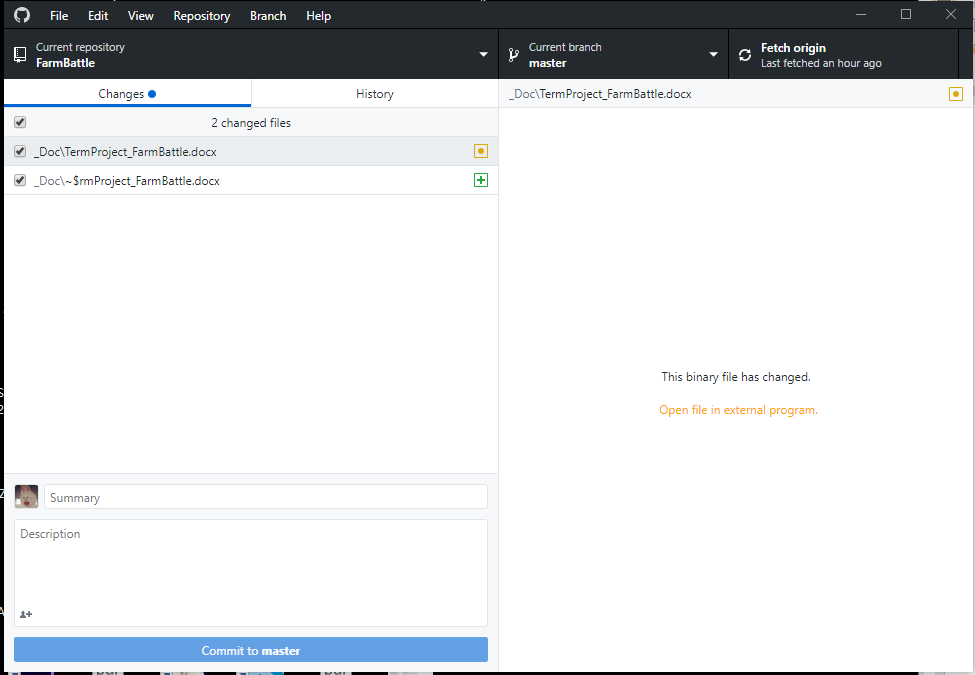
Algorithmic Plan:

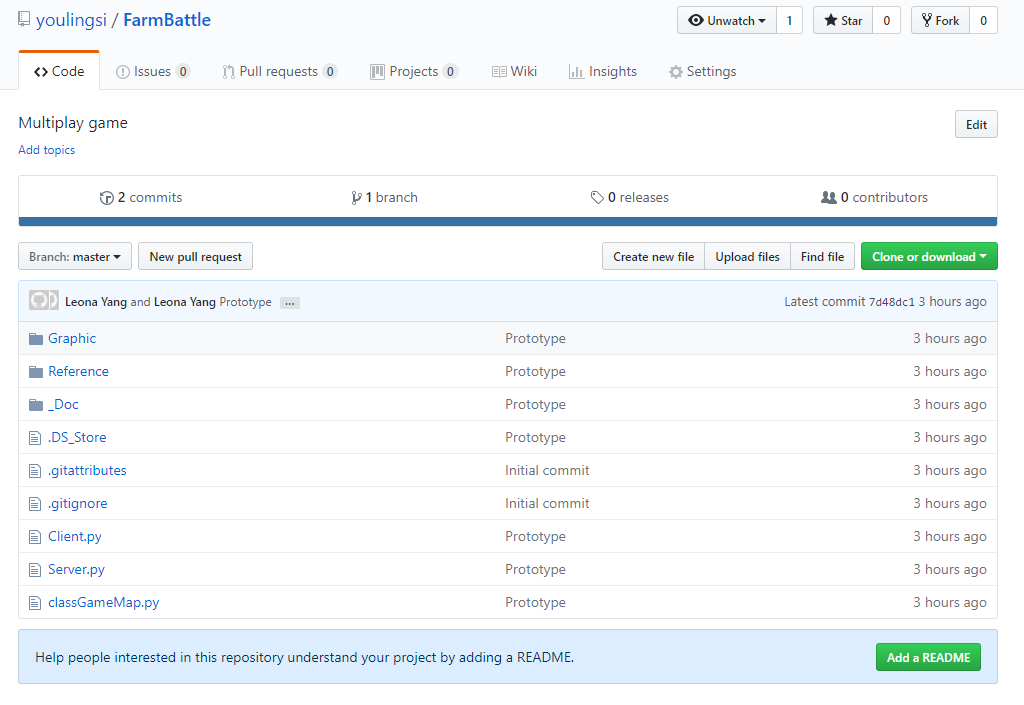
**Timeline Plan:**

1. Spend 2-4 hours every day to implement
2. Start from the core feature than extend to other feature

**Version Control Plan**

1. Use Github Desktop to back up the code
2. Make commit every day after finishing the work:

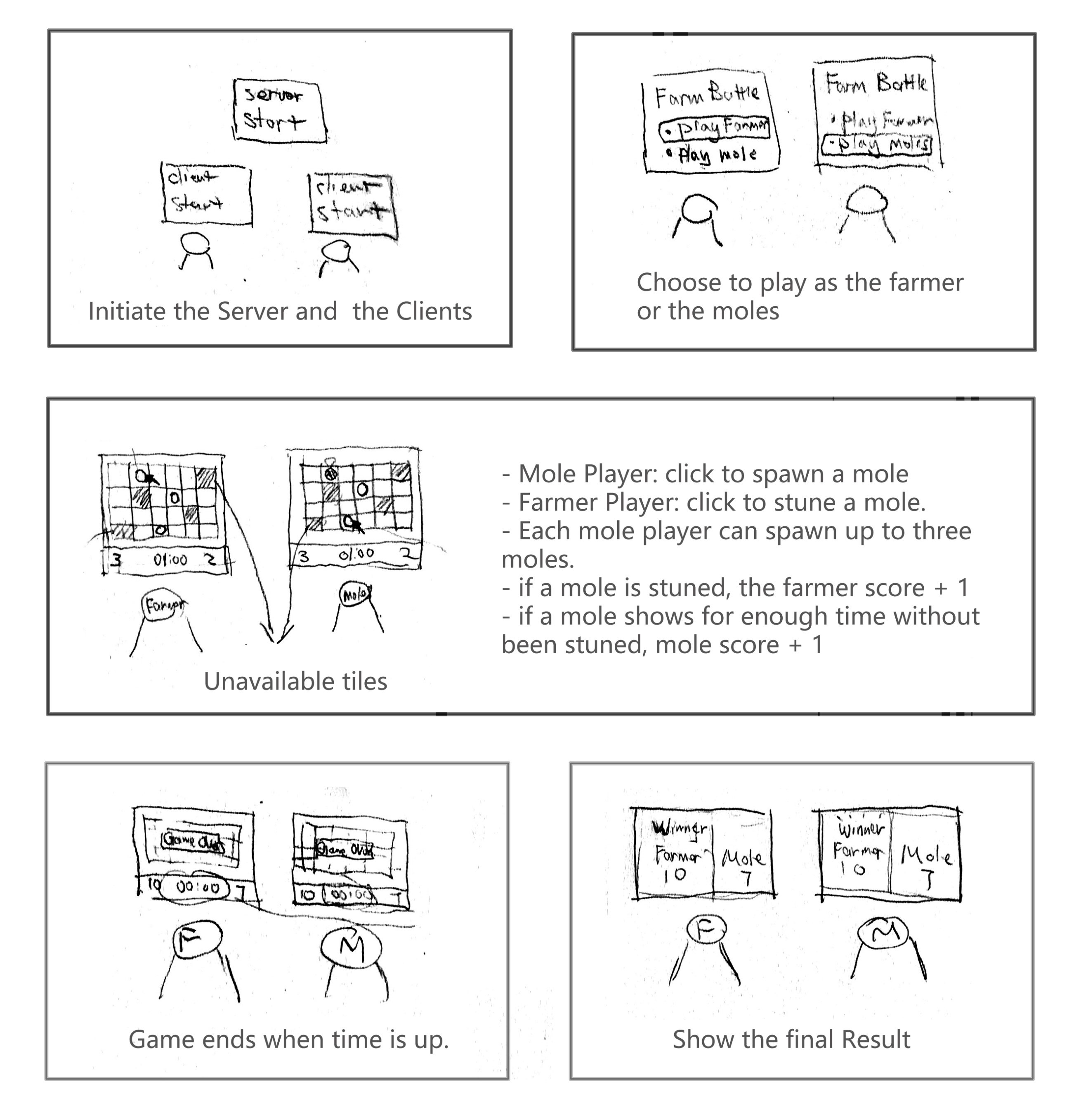




**Module List:**

1. Pygame
2. Socket

**Storyboard:**

****