2016 Adithi's High Level Design (HLD) Document- Software Project Manager

1. Introduction

-The Story-

There was once a nest of ants outside Mobeen's home. He got really sick of them entering his house, but soon he became fascinated by how all of the ants knew where to come. He began researching and figured out that ants communicate using pheromones. Using this knowledge, we will create a model to simulate how ants look for and bring bag food. This model will start off with a nest that has 'x' number of ants. A 'x' number of the ants in the nest will leave the nest and random walk around looking for food or the strongest pheromone trails to food. The ants randomly walk around looking for food or strong pheromone trails and all of the ants will leave behind a broken trail of pheromone themselves. The more ants that pass over a spot the stronger the pheromone trail will become. Over time, the pheromone trail in an area will fade if no ants go over that area. A person (the user) will be able to place a drop of sugar that contains 1000 – 5000 grains of sugar. When an ant finds the sugar it will pick up 3 sugar grains and start to follow the broken trail back to the nest leaving a solid pheromone trail on its return. When the ant return homes, the ant will inform 3 other ants to follow the strongest pheromone trail. When all the food is gone, the last ant comes back to the nest and signals the rest of the group that the food is gone and so the foraging ants will return to randomly foraging.

2. Subject Matter Experts Agreement List

Name	Title/Role	Mandatory Reviewer (y/n)	Approved
Adithi Rao	HLD Writer	Y	
Caroline Anderson	Software Manager	Y	
Ernie Edinboro	Supervisor	Y	
Apprentices	Client	Y	

3. Timeline

- **Project is Due July 1st**.

4. Desired Behavior / Components

- Website
  - Description of your project
  - Working Model created by another student
  - Create link to the following files on your website
    - HLD document, it will be a downloadable file
- Create a model that is assigned to you using that person's HLD
- **The Model**

  -Agents-

  1. The ants
  2. The World
  3. The nest
  4. Sugar

  - Agent behaviors-

  1. The ants randomly move around on the world.
  2. The ants leave behind a broken pheromone trail when they are moving.
  3. When more and more ants pass over a place on the world, the pheromone trail at that point will get stronger and stronger.
  4. Time is a major factor- the pheromone will fade over time. After 10 seconds, the trail will disappear.
  5. When/if the user places a drop of sugar in the world and the ants find the sugar, the ant will pick up 3 grains of sugar.
     i. When an ant finds sugar, it will begin following its own broken pheromone trail back to its nest.
     ii. As the ant retraces its path back to its nest, it will leave a solid pheromone trail in the other path's place.
     iii. When an ant that picked up some sugar returns home to its nest, it will recruit 3 more ants to follow the strongest trail of pheromone.

iv. When all of the sugar is gone, the last ant will return to the nest and tell all the other ants that all the food is gone- this will give those ants the signal to go back to randomly foraging

6. The world is home to the nest and the ants travel on it.
7. The nest is where all the scavenger ants come back to if they find food.

-Variables-

1. Array of Ant Objects
2. Each ant Object
    a. x-coordinate (number)
    b. y-coordinate (number)
    c. Color- Black (string)
    d. Speed (number)
    e. Width (number)
    f. Height (number)
3. Other Variables about Ants
    a. Population = User should be able to set it (should be around 4,000 to 10,000, which may need to be reduced to be able to use agentcubes).
    b. *Speed array-- Only if we want to give the ants different speeds
    c. *Colors array[black, brown]--Only if we want to make it easier for the user to know which ants are out randomly moving and which ones are out following the path for food. In this case, the color of the ants who found the food would change to brown, whereas the rest of the ants would be black.

AGENTCUBES OR JAVASCRIPT

List of all functions

1. Ant() (constructor)
2. Animate()
3. FoodCollection()
4. BrokenTrail()
5. SteadyTrail()
6. FollowLeader()

7. PheromoneFade()
8. BackToRandom()
9. Food()
10. Interruption()
11. *ChangeColor() --Optional

Descriptions of Functions

1. Ant() function (constructor)
    a. set x coordinate to random number
    b. set y coordinate to random number
    c. set color (black)
    d. *set random movement speed from Speed array- may not be necessary if there is one speed for all ants
    e. set width = Size
    f. set height = Size
2. Animate() function
    a. Initiate the random movement of the ants.
3. FoodCollection() function
    a. Makes the ant pick up 3 pieces of sugar when it finds the sugar source
4. BrokenTrail() function
    a. Creates a broken trail behind an ant that is randomly travelling
5. SteadyTrail() function
    a. Creates a steady trail behind an ant that is coming back to the nest after finding food
6. FollowLeader() function
    a. Makes 3 ants follow the most recent (darkest) trail whenever an ant that found food returns to the hill to the
7. PheromoneFade() function
    a. Makes the pheromone trail left by ants disappear after a certain number of time steps
8. BackToRandom() function

   a. Tells the ants to go back to roaming randomly after the food source has been exhausted

9. Food() function
   a. Creates a food source with around 1,000 to 5,000 grains of sugar that will be in the spot where the user clicks down
10. Interruption() function
   a. Takes into account any interruptions that may happen when an ant is out trying to bring back food to the nest- this is a delay in the time it takes for a specific percent of ants that are outside of the nest
11. *ChangeColor() function (optional)
   a. Changes the color of the ant when they are moving specifically towards the food source

VENSIM OR EXCEL

List of rates at which ants find food, variables for sugar value, and algorithms

1. Becoming Scavengers
2. Getting Food
3. Random Movement
4. Find Food
5. Travel Time
6. Sharing trail with other ants
7. Back to Work

Description of rates at which ants find food, variables for sugar value, and algorithms

1. Ant births function
   a. This must take into account the current number of ants that exist and an ant birth rate fraction- .06.
2. Ant deaths function
   a. This must take into account the current number of ants that exist and an ant death rate fraction- .03.
3. Becoming Scavengers function

   a. This must take into account the current number of ants in the nest and the percent of them that becomes scavenger ants.
4. Getting Food function
   a. This must take into account the number of scavengers out looking for food, the amount of sugar that is placed somewhere in the world (1,000-5,000 grains of sugar) and the collection rate fraction (should be 3 grains of sugar per ant). This is what will decrease the food as more and more scavenger ants come to take food.
5. Random Movement function
   a. This must use the amount of scavengers and the fraction of those ants that are randomly moving (this should be 1 or 100% when there is no food on the ground). Some of the scavenger ants should always be randomly moving, even if there is a food source.
6. Find Food function
   a. This must factor the amount of scavengers that are out looking for food and a fraction that represents the chance of them finding the food source- this will only apply when there is a food source out there. Think logically about the percentage- it depends on the distance between wherever you want your ant hill and your food source.
7. Travel Time function
   a. Once one of the ants finds food, and gets back to the hill, it will send out 3 more ants that will be focused scavengers (100% of them will get food during the time it takes to travel that distance). It is important to have a time step because ants actually take time to go to the food and come back to the nest.
8. Back to Work function
   a. Makes it so that when all of the food is gone, all ants EITHER go back into the ant nest or just randomly move around.

"***Days*** *of debugging saves* ***hours*** *of design and planning*" - Charlie Peck

--What tool or programming language will be used?--

1. Agentcubes or Javascript
2. Vensim or Excel