

Sprint 2 Backlog

We aim to improve our GUI implementation for the more detailed options, have a basic alpha and beta pruning, and a simple pie rule implementation. Therefore, the backlog for Sprint 2 looks like this:

1. The program has a basic min-max tree set up that checks all the valid moves
2. Provide graphics for game board
3. Work on visualization of home screen
4. Evaluate and test the utility evaluation function with a working min-max
5. The program checks if the leaf is at a final state
6. If at a final state, then assign the max/min possible utility value
7. If the leaf is not final state, then evaluate the state of the board to select the best possible move (using recursion/0)
8. Complete the implementation of the minimax tree
9. Integrated the iterative deepening implementation with a basic alpha pruning function
10. Integrated the iterative deepening implementation with a basic beta pruning function
11. Evaluate the same move from all the chance nodes
12. If Max node take the max value if Min node take the min value
13. Reevaluates the utility using the now found alpha and beta and min/max value
14. Game checks if the max number of moves has been exceeded
15. Game checks if the total time allocated for the game has been finished
16. Board evaluation function checks if a winner has been figured out
17. User input for number of holes of game board
18. Implement number of holes in board game based on user input
19. User input for the range of number of seeds in each hole
20. Implement number of seeds in each hole
21. User input for the random distribution of seeds on each side
22. Implement random distribution of seeds
23. Computer is modified to play first
24. User gets to choose if to do pie rule
25. If user chooses to do pie rule, implementation of board with switching positions
26. If user chooses not to do pie rule, implementation of not allowing pie rule any longer
27. User option for being timed or not
28. If user chooses timer, implementation of time options
29. Implementation of time on the game board, displaying a timer and losing turn if player does not make a move
30. Set up the protocol for client server architecture
31. Set up the basic configuration for client server
32. Set up the client connection
33. Integrate the server
34. Integrate the client server architecture with the game
35. Integrate the remote client services with the game developed
36. Implement number of holes in board game based on server configuration
37. Implement number of seeds in each hole based on server configuration
38. Implement random distribution of seeds based on server configuration

39. Basic configuration properties files in order to pass in seeds, houses, random distribution, and time limit per move
40. Setup program to host a server and multiple clients
41. Client connection to server should be acknowledge with WELCOME
42. Client should receive and print out INFO statement with game configuration including type of player

Task Assignment

1. The program has a basic min-max tree set up that checks all the valid moves
 - a. Sneha Santani
 - b. Completed
2. Provide graphics for game board
 - a. Yerania Hernandez
 - b. Removed
3. Work on visualization of home screen
 - a. Yerania Hernandez
 - b. Completed
4. Evaluate and test the utility evaluation function with a working min-max
 - a. Sneha Santani
 - b. Completed
5. The program checks if the leaf is at a final state
 - a. Sneha Santani
 - b. Completed
6. If at a final state, then assign the max/min possible utility value
 - a. Sneha Santani
 - b. Completed
7. If the leaf is not final state, then evaluate the state of the board to select the best possible move (using recursion/0)
 - a. Sneha Santani
 - b. Completed
8. Complete the implementation of the minimax tree
 - a. Sneha Santani
 - b. Completed
9. Integrated the iterative deepening implementation with a basic alpha pruning function
 - a. Sneha Santani
 - b. Completed
10. Integrated the iterative deepening implementation with a basic beta pruning function
 - a. Sneha Santani
 - b. Completed
11. Evaluate the same move from all the chance nodes
 - a. Sneha Santani
 - b. Completed
12. If Max node take the max value if Min node take the min value
 - a. Sneha Santani
 - b. Completed
13. Reevaluates the utility using the now found alpha and beta and min/max value
 - a. Sneha Santani
 - b. Completed
14. Game checks if the max number of moves has been exceeded
 - a. Sneha Santani
 - b. Completed
15. Game checks if the total time allocated for the game has been finished
 - a. Sneha Santani

- b. In Progress
- 16. Board evaluation function checks if a winner has been figured out
 - a. Karl Lawson
 - b. Completed
- 17. User input for number of holes of game board
 - a. Yerania Hernandez
 - b. Removed
- 18. Implement number of holes in board game based on user input
 - a. Karl Lawson
 - b. Removed
- 19. User input for the range of number of seeds in each hole
 - a. Yerania Hernandez
 - b. Removed
- 20. Implement number of seeds in each hole
 - a. Karl Lawson
 - b. Removed
- 21. User input for the random distribution of seeds on each side
 - a. Yerania Hernandez
 - b. Removed
- 22. Implement random distribution of seeds
 - a. Karl Lawson
 - b. Completed
- 23. Computer is modified to play first
 - a. Karl Lawson
 - b. Removed
- 24. User gets to choose if to do pie rule
 - a. Yerania Hernandez
 - b. Completed
- 25. If user chooses to do pie rule, implementation of board with switching positions
 - a. Karl Lawson
 - b. Yerania Hernandez
 - c. Completed
- 26. If user chooses not to do pie rule, implementation of not allowing pie rule any longer
 - a. Karl Lawson
 - b. Yerania Hernandez
 - c. Completed
- 27. User option for being timed or not
 - a. Yerania Hernandez
 - b. Removed
- 28. If user chooses timer, implementation of time options
 - a. Yerania Hernandez
 - b. Removed
- 29. Implementation of time on the game board, displaying a timer and losing turn if player does not make a move
 - a. Yerania Hernandez
 - b. Removed

Additional Components Added:

30. Set up the protocol for client server architecture
 - a. Yerania Hernandez
 - b. Completed
31. Set up the basic configuration for client server
 - a. Yerania Hernandez
 - b. Completed
32. Set up the client connection
 - a. Yerania Hernandez
 - b. Completed
33. Integrate the server
 - a. Yerania Hernandez
 - b. Completed
34. Integrate the client server architecture with the game
 - a. Yerania Hernandez
 - b. Completed
35. Integrate the remote client services with the game developed
 - a. Yerania Hernandez
 - b. Completed
36. Implement number of holes in board game based on server configuration
 - a. Karl Lawson
 - b. Completed
37. Implement number of seeds in each hole based on server configuration
 - a. Karl Lawson
 - b. Completed
38. Implement random distribution of seeds based on server configuration
 - a. Karl Lawson
 - b. Completed
39. Basic configuration properties files in order to pass in seeds, houses, random distribution, and time limit per move
 - a. Yerania Hernandez
 - b. Completed
40. Setup program to host a server and multiple clients
 - a. Yerania Hernandez
 - b. Completed
41. Client connection to server should be acknowledge with WELCOME
 - a. Yerania Hernandez
 - b. Completed
42. Client should receive and print out INFO statement with game configuration including type of player
 - a. Yerania Hernandez
 - b. Completed

SCRUM Meetings

Our meetings will be held at the following times along with our backlog updates:

1. Sprint 2 Backlog: 3/21/2017
2. Scrum Meeting 1: 3/22/2017
3. Sprint 2 Backlog Update: 3/23/2017
4. Scrum Meeting 2: 3/24/2017
5. Sprint 2 Backlog Update: 3/26/2017
6. Scrum Meeting 3: 3/27/2017
7. Sprint 2 Backlog Update: 3/28/2017
8. Scrum Meeting 4: 3/29/2017
9. Sprint 2 Completion: 3/30/2017
10. Sprint 2 Retrospective: 3/31/2017