

Anika Payano
Colvin Chapman
Connor Novak
Emily Lepert
Sophia Nielsen

Software Design Final Project

1. The Big Idea:

Main Idea: We want to create a game where organisms are controlled by AIs that evolve over time. The user would have the option to set up the board how they like.

Topics to explore:

- Game Generation
- Artificial Intelligence
- Evolutionary Algorithms

Generated Product:

- Two-AI game, AI vs AI, with player interaction to lay out board
- Demonstration of AI gameplay with ability to choose AI “difficulty” represented by number of evolutionary iterations

Gameplay specifics:

- Capture the flag inspired
- Unknown section of the field that nobody can see
- Players can “discover” the field by sending units to explore
- Units have a radius of visibility around it that determines how far it can see
- Base generates units at a set rate of time.
 - Examples:
 - Teeny Unit: Low speed, low strength, low attack, low health -> 5s production time
 - Brute Unit: Low speed, high strength, low attack, high health -> 10s production time

MVP: Sophia

MVP:

- Two AI's, one evolved to play the game with a specific success rate
- User sets up game
- AIs need to discover the board as they go (send out units to gain information about map)
- Capture-the-flag gameplay with different types of units
- Multiple unit control

Stretch Goal:

- AIs have different personalities that change how they evolve
- Randomly generated set ups
- Implement different AI in addition to *astar, compare?

2. Learning Goals:

What are your individual learning goals for this project?

Anika: Work on making code neat and understandable. Also messing with AIs and more game based code.

Colvin: Collaborating as efficiently as possible with a larger team; learning about evolutionary algorithms; write comments as I write code instead of after.

Connor: Develop more robust and legible code; concise comments, concise code, easy for others to read, change, and use; good structure and layout

Emily: Learn more about AIs and evolutionary algorithms and implementing them. Learn more about game play implementation. Improve documentation and legibility of code. Work on good structure and layout of code.

Sophia: Learn more about AI's and evolutionary algorithms and their implementation. Learn about how to evolve for a specific goal.

3. Implementation Plan:

This will probably be pretty vague initially. Perhaps at this early juncture you will have identified a library or a framework that you think will be useful for your project. If you don't have any idea how you will implement your project, provide a rough plan for how you will determine this information.

Roughly following the toolboxes on the topics of AI and Evolutionary Algorithms, we will be using:

- pygame
- deap
- numpy

The AI will be built off of the astar* pathfinding algorithm, and the weights for the various actions/moves will be determined through evolution to build a better AI. One of the sections yet to be determined is how to go about weighing various moves and positions such that the astar* algorithm can be extrapolated to playing a game.

4. Project schedule:

You have 6 weeks (roughly) to finish the project. Sketch out a rough schedule for completing the project. Depending on your project, you may be able to do this in great specificity or you may only be able to give a broad outline. Additionally, longer projects come with increased uncertainty, and this schedule will likely need to be refined along the way.

March 20 - Project Proposal

March 23 - mapping during class/thinking about review

March 27 - [Architectural Review](#) (need to have a document prepared)

March 30 - Have (rough) visual gameplay that two users play + investigation into AI implementation

- Have discussion into weighting of tiles, etc...

April 3 - Connor and other implement basic AI into game

April 10 - Evolutionary part is done

April 17 - Stretch Goals!

April 24 - [Project Presentation](#) (Formal presentation) + other stretch goals

May 1 - Project Website & Poster Demo

5. Collaboration plan:

How do you plan to collaborate with your teammates on this project? Will you split tasks up, complete them independently, and then integrate? Will you pair program the entire thing? Make sure to articulate your plan for successfully working together as a team. This might also include information about any software development methodologies you plan to use (e.g. [agile development](#)). Make sure to make clear why you are choosing this particular organizational structure.

Combination of pair programming and individual programming. Each goal will require different methods (ie: for the first week, game play implementation might break down the game play into different classes, while AI discovery could be more pair programming). We will take into account outside courses putting a stressor on certain team members during certain weeks.

6. Risks:

What do you view as the biggest risks to the success of this project?

- Implementing the AI's "decision-making" process for optimal game play
 - Drawing movement destinations for the AI
- AI process being computationally expensive especially when we start evolving it
- Code becoming monolithic and difficult to read/sort out parts to work on and fix

7. Additional Course Content:

What are some topics that we might cover in class that you think would be especially helpful for your project?

- Implementation of AI's in general (even astar); Toolbox doesn't help to build *astar, merely provides example and asks to slightly change implementation.