CSPB 3112 – Digitizing an Older Board Game: Yggdrasil

Author: Daniel Simonson

Vision:

For this project, I am planning to turn the 2011 board game "Yggdrasil" into a digital format to be played through one's desktop. I am hoping this will deepen my understanding of different aspects of software development from start to finish.
The goal is not necessarily to 100% have a working game at the end of the semester, as there are just not enough hours in the day. The goal would be to create design documentation, diagrams as well as small components of the game to show functionality.

This would deepen my knowledge of different coding environments and languages. After a brief dive into game creation Pygame, Godot or Unity are all good options to look into.

Motivation: I have always been a huge board game fan and rarely have enough time or people to get games like this to the table. I chose this game for a number of reasons. First off, it is cooperative, so no competition or score tracking is required. There is a clear end game both to win and to lose and if neither are met, the game continues. The other reason is that this game can be played solo, so no need to create bot players to play if no one else can join.

Risks to completion:

The obvious risk is time. This class is labeled as 1 unit, and 3-5 hours a week. That is simply not enough time to fully create and test a full game as just one person.

The other risk is how much self learning is potentially needed here.
None of the courses I have taken in this program have specifically been tailored to anything in this project.

Mitigation Strategy:

To mitigate the time risk, I have set more realistic goals within this project and have no intention of presenting a full working game by May. These set criteria and deadlines will help keep me on task and on track to getting done what I need to get done.
As for the second major risk, I am fortunate to have a number of friends who work in the CS field and have some experience in game creation/alteration. They have already pushed me to look up certain tools and videos that will be beneficial in this project.

Project Assessment/Evaluation Criteria:

-Write a Requirements Document

-Write a Design document with case and class diagrams.

-Have at least one working component to the game, preferably one of the 10 different turn options, as well as the special dice that is included in the game.

Rough Weekly Schedule:

Weeks 1–2: Understand *Yggdrasil* and Plan the Digital Version

Objective: Break down *Yggdrasil* into core components and define how it will translate into a digital format.

Tasks:

- **Play *Yggdrasil* and document:**
    - Rules and mechanics, focusing on cooperative gameplay.
    - Game flow (turns, actions, event resolution, and victory/loss conditions).
    - Components (board, gods, enemy deck, tokens, resources, and action spaces).
- **Decide on the platform:**
    - Web-based (React + Flask), desktop (Pygame/Godot), or mobile.
- **Sketch initial UI/UX ideas:**
    - Digital board layout, player interactions, and cooperative decision-making.
    - Handling enemy advancement and action resolution.
- **Choose tools and frameworks:**
    - Python-based solutions (Pygame, Godot, Flask) for the backend and game logic.
    - UI/UX tools for wireframing (Figma, draw.io).

Deliverables:

- A **game design document** outlining rules, mechanics, and digital adaptations.

- **Mockups** of the game's digital interface (board, actions, enemy advancement system).

Weeks 3–4: Set Up the Development Environment

Objective: Prepare tools and begin coding the basic framework for *Yggdrasil's* digital version.

Tasks:

- Install necessary tools (e.g., Python, Flask/Pygame/Godot, VS Code, Git).

- Create a **GitHub repository** for version control.

- Develop the **basic project structure**:

    o Load the *Yggdrasil* board and action areas (even as placeholders).

    o Implement basic interactions (e.g., selecting an action, drawing an enemy card).

Deliverables:

- A **working prototype** with placeholder UI.

- Basic interactions, such as moving a god token or drawing a card.

Weeks 5–6: Implement Core Mechanics

Objective: Focus on implementing *Yggdrasil's* turn structure and main mechanics.

Tasks:

- **Program turn structure** (each player choosing actions before enemies advance).

- **Implement core mechanics:**

    o Drawing enemy cards and resolving effects.

    o Player actions (e.g., recruiting Viking warriors, gathering resources, fighting enemies).

- **Test rule enforcement** (ensuring cooperative mechanics work correctly).

Deliverables:

- **A working version of key mechanics** (player actions, enemy advancement).

- Basic **game logic for completing a full round**.

Weeks 7–8: Add Game State Management

Objective: Manage game state tracking and cooperative interactions.

Tasks:

- Implement state management for:

  - **Player actions & available resources.**

  - **Enemy progression** (handling how enemies advance towards Asgard).

  - **Game board updates** (real-time tracking of threats and progress).

- Add **win/loss conditions** (e.g., all enemies reaching Asgard results in a loss).

- Playtest different scenarios to ensure smooth state transitions.

Deliverables:

- **A functional game flow from start to end.**

- **State tracking system** (displaying enemy advancement, player turns, and resources).

Weeks 9–10: Polish the User Interface

Objective: Improve the visual design and player interactions for a better experience.

Tasks:

- Replace placeholders with **themed graphics** (board, characters, enemies).

- Add **animations and visual feedback**:

  - Highlight enemy movement.

  - Indicate available actions.

  - Show results of combat or resource gathering.

- Implement a **main menu and game setup screen** (player count, difficulty settings).

Deliverables:

- **Refined UI with improved interactivity.**

- **Smoother user experience** with animations and action feedback.

Weeks 11–12: Playtest and Plan Next Steps

Objective: Gather feedback, refine gameplay, and plan additional features.

Tasks:

- **Playtest** with friends or classmates.

- Collect feedback on:

    o **Game mechanics** – Does it feel balanced and true to the board game?

    o **UI experience** – Is it easy to navigate and play cooperatively?

    o **Multiplayer interaction** – Does cooperation feel smooth in a digital setting?

- Fix bugs and make minor improvements.

- Plan future development:

    o **AI-controlled teammates** for solo play.

    o **Online multiplayer** for cooperative play.

Deliverables:

- **A fully playable prototype.**

- **List of improvements and next steps.**

https://github.com/DSimonson1251/DSimonson1251.github.io