

Project Report: Delta Guardians

A Game of Strategy and Social Impact for the Vibe Coding Challenge

Capybara Talent Force

October 20, 2025

Contents

1	Introduction	2
2	Game Theme Topic Justification	2
3	Potential Impact	2
4	Technology Stack	3
4.1	Web Technologies	3
4.2	AI Tools and "Vibe Coding" Workflow	3
5	Overview of Game Mechanics	4
6	Reflection	4
6.1	The Development Process	4
6.2	Successes and Challenges	5
6.3	Future Improvements	5
7	Conclusion	6

1 Introduction

"Delta Guardians" is a single-player, turn-based strategy game where the player assumes the role of a community leader in the Mekong Delta of Vietnam. The primary objective is to achieve a target level of economic prosperity (Treasury) within a 20-year timeframe (20 turns). This goal is challenged by the persistent and escalating threat of climate change-induced saltwater intrusion, which damages crops and threatens the community's livelihood.

The game was conceived and developed to meet the criteria of the "Vibe Coding - Play to Impact" challenge, which called for a web-based game that highlights a pressing social issue in Vietnam or Australia. By simulating the difficult trade-offs between short-term economic gain and long-term environmental resilience, "Delta Guardians" aims to create an engaging and thought-provoking player experience.

2 Game Theme Topic Justification

The Mekong Delta is often called Vietnam's "rice bowl," responsible for a significant portion of the nation's food production and the livelihood of millions. This vital region is acutely vulnerable to the impacts of climate change, particularly sea-level rise.

Our choice of this theme is justified on several grounds:

- **Urgency and Relevance:** Saltwater intrusion is not a future threat but a present crisis. Each year, saltwater pushes further inland, rendering agricultural land barren and jeopardizing food security. This makes it a highly relevant and urgent social challenge.
- **Complex Trade-offs:** The problem is not simple to solve. Solutions like planting resilient mangrove forests come at a high initial cost and produce no direct income, creating a difficult strategic choice for farmers who need immediate revenue. This inherent tension is a perfect foundation for engaging gameplay.
- **Educational Opportunity:** While widely reported, the mechanics of this crisis (salinity, crop vulnerability, mitigation strategies) can be abstract. A game provides a simplified, interactive model that allows players to understand these dynamics firsthand.

By gamifying this issue, we transform a complex socio-environmental problem into an accessible and compelling strategic challenge.

3 Potential Impact

The primary goal of "Delta Guardians" extends beyond entertainment. We aim to achieve a tangible impact on the player's perspective and understanding.

1. **Educational Value:** Players learn about the core mechanics of the crisis, including the function of mangroves as a natural defense, the vulnerability of different crop types, and the constant, creeping nature of the salinity threat.

2. **Raising Awareness:** The game serves as a narrative vehicle, bringing a critical regional issue to a potentially global audience in an interactive format that is more engaging than a traditional article or documentary.
3. **Fostering Empathy:** By putting the player in a position of responsibility, the game encourages empathy for the real-life communities that face these impossible choices. Players experience the frustration of a failed harvest and the relief of a successful defensive strategy.

4 Technology Stack

The game was developed to be lightweight, accessible, and fully client-side, adhering to the challenge's technical constraints. The technology stack is divided into web technologies and the AI tools central to the "vibe coding" process.

4.1 Web Technologies

- **HTML5:** Provides the fundamental structure of the game screens and UI elements.
- **CSS3:** Used for all styling, layout, and animations. Modern features like Flexbox and Grid were used extensively to create a responsive layout that adapts to different screen sizes.
- **JavaScript (ES6):** Powers the entire game engine, including all state management, game logic, rule enforcement, and DOM manipulation. No external JavaScript libraries or frameworks were used.

4.2 AI Tools and "Vibe Coding" Workflow

The development process was heavily accelerated and creatively enhanced through the use of AI tools.

Gemini 2.5 Pro Used for conceptualization, content generation, and code scaffolding.

- **Game Design:** Brainstorming event ideas, balancing plant statistics, and writing the detailed "How to Play" guide.
- **Code Generation:** Scaffolding JavaScript functions and providing solutions to logical problems. For example, generating the initial structure for the 'endTurn()' function:

```
1 function endTurn() {  
2     // 1. Apply a random event.  
3     // 2. Increase global salinity.  
4     // 3. Calculate damage to crops based on protection.  
5     // 4. Calculate income from surviving crops.  
6     // 5. Update game state and re-render the UI.  
7     // 6. Check for win/loss conditions.  
8 }  
9
```

Listing 1: AI-scaffolded game loop concept

Nano Banana Used for the creation of all visual assets.

- **Game Sprites:** All plant types, the community house, and the empty land tile were generated with prompts like: *"pixel art game asset, a dense mangrove tree, top-down view, 2D sprite"*.
- **UI & Background:** The game's background image and favicon were also generated using descriptive prompts to create a cohesive visual theme.

5 Overview of Game Mechanics

- **Core Loop:** The game proceeds in turns, with each turn representing one year. The game ends after Year 20.
- **Player Actions:** During a year, the player can perform multiple actions, provided they have enough money. Actions include planting a Rice Paddy, Fruit Orchard, or Mangrove Forest on an empty tile, and using the "Purify Water" special action to reduce global salinity.
- **The Plants:** Each plant has a unique profile of cost, health, and production, creating strategic choices between reliable income (Rice), high-risk profit (Orchard), and defense (Mangrove).
- **Salinity and Damage:** Salinity is the primary antagonist. It increases slightly each year and can be drastically increased by events. At the end of each year, all non-mangrove plants take damage. Plants protected by an adjacent mangrove take half damage.
- **Income:** Income is calculated at the end of the year based on the production value of surviving plants, scaled by their remaining health percentage.
- **Events:** Each year begins with a random event that can be beneficial (e.g., subsidies) or detrimental (e.g., floods, pests that damage even mangroves).
- **Win/Loss Condition:** The player wins by meeting or exceeding their Treasury goal by the end of Year 20. There is no other loss condition, making the game a race against time and environmental decay.

6 Reflection

6.1 The Development Process

The "vibe coding" approach was transformative. Instead of getting bogged down in boilerplate code or asset creation, we could focus on the core gameplay loop and user experience. The process was highly iterative:

1. Have an idea (e.g., "Mangroves should have their own threats").
2. Prompt an AI for event concepts and code logic.



Figure 1: A mid-game screenshot of Delta Guardians, showing the game board and UI.

3. Receive and refine the AI's output.
4. Integrate the new feature and test its balance.

This cycle allowed for rapid prototyping and the implementation of complex features in a fraction of the time it would normally take.

6.2 Successes and Challenges

The biggest success was achieving a polished, playable, and complete game that effectively communicates its core message within the time constraints. The AI-driven workflow was instrumental in this.

The primary challenges were:

- **Game Balancing:** While AI can generate stats, the human touch is essential for balancing costs, production, and damage to ensure the game is challenging but fair. This required significant manual tweaking and playtesting.
- **AI Output Integration:** AI-generated code often requires refactoring to fit into an existing structure. Similarly, AI assets sometimes need manual editing to ensure perfect consistency.
- **UI/UX Refinement:** Initial layouts were functional but not intuitive. Iterating on the UI to add features like the permanent health display and responsive screen-fitting was a key challenge that was solved through conventional development practices.

6.3 Future Improvements

Given more time, "Delta Guardians" could be expanded with:

- A wider variety of crops and defensive structures.
- A technology tree for researching upgrades (e.g., salt-resistant crops).

- More complex events, including multi-choice narrative events.
- Sound effects and background music to enhance immersion.

7 Conclusion

"Delta Guardians" stands as a successful proof-of-concept for the "vibe coding" methodology and a meaningful execution of the "Play to Impact" theme. It is a complete, engaging game that effectively translates a complex real-world issue into an interactive strategic experience. The project demonstrates the power of combining human creativity with AI tools to rapidly build impactful software.