Urban Plastic: Saigon Cleanup

GenAI & Cybersecurity Hackathon 2025 — Challenge 3 (Vibe Coding)

Team: (Điền tên team) | Author: nguyentrithong

ased 2D browser game on reducing single-use plastic waste in V

Table of Contents

1. Introduction

Urban Plastic: Saigon Cleanup is a short, replayable browser game designed to raise awareness about single-use plastic in Ho Chi Minh City. Players act as a city planner and make policy choices over 12 turns. Each decision affects four metrics: Budget, Public Support, Environment, and Plastic Waste. The game is front-end only and works offline to satisfy hackathon constraints.

2. Problem Context & Social Impact

Urban Context (Vietnam)

- Fast urbanization and street-food culture increase single-use plastics (bags, boxes, bottles).
- Plastic blocking drains exacerbates seasonal flooding and pollution in canals and rivers.

Educational Goals

- Help players understand policy trade-offs and behavior change levers.
- Encourage sustainable practices (sorting, deposit-return schemes, levies, outreach).

Audience

- Students, young citizens, and anyone curious about sustainability solutions.

3. Game Design Overview

Game Loop

- Draw an event → choose Option A or B → metrics update → next turn.
- Survive 12 turns or until a fail condition occurs.

Win/Lose

- Win: high overall score with low Plastic Waste (configurable thresholds).
- Lose: Budget or Support collapses (≤0) or Waste hits 100.

Content

- Events: street-food plastics, sorting bins, river interceptors, retail bag levies, festival surges, etc.

4. Core Mechanics & Balancing

Metrics

- Budget (higher is better), Public Support (higher is better), Environment (higher is better), Plastic Waste (lower is better).

Scoring

- Final Score = average(Budget, Support, Environment, (100 - Waste)).
- Grades: A ≥ 85, B ≥ 70, C ≥ 55, else D.

Balance Considerations

- Infrastructure options reduce waste but may be costly (Budget down).
- Outreach and incentives can improve Support but require Budget.
- Levies may reduce waste but risk lowering Support.

5. Technical Architecture

Front-end Only

- All logic implemented in plain JavaScript; runs from file:// without CORS issues.
- Events embedded directly to avoid additional asset loading.

Code Structure

- index.html — entry point and scenes (Menu/Play/Results).
- styles.css — layout, meters, buttons, responsive tweaks.
- game.js — state machine, events, scoring, accessibility helpers.

Performance

- Minimal DOM churn, simple CSS animations, no heavy frameworks.

6. AI Vibe Coding Process

Prompts (see /prompts)

- Concept prompts for theme, events, JSON schema, win/lose conditions.
- Code generation prompts for scaffolding HTML/CSS/JS and accessibility.
- Refinement prompts for balancing, naming, and structure.

What AI Did

- Accelerated prototyping, ensured consistent tone and UI copy, suggested event ideas.
- Helped generate documentation scaffolding (README & report templates).

Human-in-the-loop

- Manual review, localized wording, gameplay tuning, and a11y testing.

7. Accessibility & UX

Accessibility

- Keyboard focus on scene change; Help modal has ARIA roles and Esc-to-close behavior.
- High-contrast palette; large tap targets; semantics in HTML.

UX Polish

- Inline tips, consistent button states, gentle shadows and rounded corners.

8. Playtesting & Findings

Observations

- Players quickly understood meters and trade-offs; short sessions work well in class settings.

Adjustments

- Reduced extreme Budget hits; increased effectiveness of deposit-return events.

9. Limitations & Future Work

Limitations

- Simplified model; not calibrated to real municipal data yet.
- No persistence/analytics due to front-end-only constraint.

Future Work

- District-specific scenarios; localized datasets; daily challenges; SFX/sprites; VN/EN toggle.

10. How to Run & Repo Structure

Run

- Open game_submission/game_app/index.html in a modern browser.
- Use GitHub Pages for a shareable URL if desired.

Submission Structure

- README.md — overview & instructions
- project_report.pdf — this report
- youtube_link.txt — one-line demo link
- prompts/ — prompts used during vibe-coding
- game_app/ — playable game (index.html + assets)
- screenshots/ — up to 5 screenshots

End of Report