RedCycle — Project Guide

This guide explains the project's goals, the design decisions, how the web app maps to those decisions, and answers the required deliverables for the Jezero Crater waste-reuse challenge.

Project summary — what we developed

RedCycle is an educational, web-based prototype that simulates a constrained Mars habitat recycling system focused on immediate reuse and recycling of inorganic waste. The app models:

- Inventory of available materials (fresh and recovered),
- a reusable **waste pool** that prioritizes reusing previously processed waste,
- processing modules that convert inputs into recovered products with configurable efficiencies,
- a map interface (Jezero Crater context) visualizing waste distribution and weighted centroids by material type,
- mass-balance metrics (recovered vs wasted) and resource tradeoffs (energy, water).

Why it matters: RedCycle focuses on in-situ resource utilization (ISRU) at a human-mission scale — maximizing material recovery while minimizing additional inputs (energy, water, crew time) and unusable outputs. The intended impact is to help mission designers and students reason quickly about reuse-first strategies and simple circular systems.

Challenge mapping — objectives answered

The competition asks for solutions that design a sustainable system to manage and reuse inorganic waste at Jezero Crater. RedCycle addresses this by:

- Providing immediate-reuse behavior: wastePool is consumed first when processing (reuse-first), which reduces new inputs required from initial inventory.
- Implementing secondary recovery: multi-pass processing recovers additional material from initial residues.
- Exposing resource tradeoffs through Eco/Ultra Eco modes that reduce water/energy consumption and change process efficiency, encouraging minimal inputs.

- Visualizing spatial aggregated data (weighted centroids) so mission planners can see where material accumulates and prioritize local collection.

Three scenarios and detailed reuse workflows

1) Residence Renovations

Problem: After inflating a habitat and installing a 3-D cube frame to support it, the frame and protective packaging become available as waste. The crew needs materials to produce habitat outfitting and small structural parts.

Potential recyclable/reusable elements:

- Aluminum struts and small frames
- Polymer matrix composites (carbon-fiber reinforced plastic)
- Foam packaging (Zotek F30-like)
- Bubble wrap and air cushions

Suggested workflow (implemented as a scenario in RedCycle):

1. Collect aluminum struts and sort by metallic vs composite. In the app, stage these materials into the 'Structural' module.
2. Run a recycling process (Full) that recovers metal stock for fasteners and small structural brackets. Secondary recovery converts leftover composite into filler material for interior panels.
3. Use recovered foam to create lightweight insulation panels or packaging for new equipment; combine with regolith if heavier structural mass is acceptable.
4. Leftovers after secondary recovery are placed into the wastePool for future reuse-first consumption.

Benefits: recovers high-value, structural mass, reduces shipped replacement parts, and uses foam as low-mass, high-insulation outputs.

2) Cosmic Celebrations

Problem: The crew wants to celebrate with decorations and small novelty items but has no dedicated party supplies.

Potential materials: clothing, washcloths, disinfectant wipes, plastic wrap, rehydratable and drink pouches, thermal pouches.

Suggested workflow:

1. Gather textiles and flexible packaging. Stage fabrics and plastic pouches into the 'Fabric & Soft Goods' module in the app.

2. Run a low-energy 'Quick' process that produces ribbons, buntings, and modular decor pieces from textile strips and laminated plastic pouches; this preserves energy and crew time.
3. Use recovered small containers (from pouches) for confetti holders or seed-starting trays for hydroponics experiments.
4. Prioritize reuse-first: if similar textile waste exists in the wastePool, use it first to avoid consuming fresh clothing kits.

Benefits: fosters morale with low resource cost, demonstrates creative reuse, and generates small utility items useful beyond the celebration.

3) Daring Discoveries

Problem: Research produces devices and carbon-rich residues (from oxygen extraction experiments) and some EVA-used items that can be reused.

Potential materials: EVA waste (Nomex, nylon), filter meshes, carbon residues, resealable bags, nitrile gloves.

Suggested workflow:

1. Collect filter meshes and carbon residues; stage into an 'Experimental Recovery' module that is tuned for carbon recovery and filter refurbishment.
2. Run Full processing to extract usable carbon (for composite filler, or as feedstock for small printed tools) and to refurbish meshes (cleaning, bonding).
3. Surplus carbon can be combined with binders (small amounts of polymer) to produce low-strength but useful composite parts or radiation shielding inserts by mixing with regolith.
4. Refurbished fabric (EVA, Nomex) can be repurposed into seals, insulation patches, or layered to make tool pouches.

Benefits: recovers experimental outputs into useful materials, reduces waste from science operations, and increases overall mission resource efficiency.

System design overview (how RedCycle maps to a real system)

At a glance, the app models a compact waste-management infrastructure with these components:

- **Collection & Sorting**: crew collects waste items and assigns material keys (aluminum, textiles, foam, carbon).
- **Waste Pool**: stores leftovers from processes and makes them available for the next reuse pass (reuse-first policy).

- **Processing Modules**: small-footprint workstations with tunable efficiencies. Each module consumes inputs, uses energy/water, and yields recovered products + residuals.
- **Secondary Recovery**: additional recovery pass applied to residual waste to maximize mass recovered before final unusable output.
- **Distribution**: recovered products are placed into product inventory for reuse in habitat outfitting.

Design considerations implemented in the prototype:

- Minimize crew time — the interface supports quick vs full processing choices. Quick reduces crew time and resource cost but lowers efficiency.
- Minimize water use — Eco/Ultra Eco modes lower water usage and push the system to rely on secondary mechanical or thermal recovery strategies where appropriate.
- Localize processing — map centroids help prioritize collection routes, reducing extra crew EVA time.
- Re-use outputs of processing where possible (e.g., waste heat for preheating processes, wastewater reuse) — described as extension items in the guide.

Data & implementation mapping (developer-oriented)

The prototype code maps the physical system to simple data structures and functions. Key files and shapes:

```
// src/hooks/useWasteManagement.ts
// - inventory: { [key]: { name, kg } }
// - wastePool: { [key]: kg }
// - products: { [key]: { name, kg } }
// - processMaterials(moduleId, staged[], mode: 'quick'|'full', optimize:boolean)
//   returns { recoveredKg, wastedKg, outputs }

// src/components/Mars3DMap.tsx
// - visualizes locations and computes weighted centroids per material type

// src/pages/RedCycle.tsx
// - UI wiring: drag/drop, staging, module actions, eco toggles
```

Mass-balance is explicitly tracked: the hook maintains `recoveredTotal` and `wastedTotal`, and all module runs update these counters and persist them to `localStorage` so the mission state survives page reloads.

Mass-balance and metrics

The core metric is simple: for any process run, input mass = recovered mass + wasted mass (modulo rounding and secondary recovery accounting). Implementation details:

- Processing applies an efficiency factor per module: recovered = input * efficiency.
- Secondary recovery reprocesses a fixed fraction of initial waste to extract additional low-grade material.
- Waste pool accumulation = sum leftover wastes across runs; the system consumes the pool first when staging new jobs.
- The HUD shows cumulative recoveredTotal, wastedTotal, and available-for-reuse (wastePool total).

Resource optimization & minimization

Key strategies used in the prototype to minimize inputs and unusable outputs:

- Reuse-first pool: prioritizes previously-processed material to avoid fresh inventory use.
- Eco/Ultra Eco modes: apply multipliers to energy/water cost and can change process choices; Ultra Eco reduces resource draw further with a modest recovery penalty.
- Secondary recovery: increases total recovered mass while often costing little additional water (mechanical sieving, sorting) — modeled in the prototype as a second pass with a smaller recovery fraction.
- Encourage local aggregation: the map centroids help plan collection to reduce EVA time.

Health & safety considerations

The project intentionally avoids any process that would create the following on Mars:

- Toxic emissions or harmful chemical byproducts
- PFAS generation or release
- Microplastic dispersal into wastewater or the habitat environment

Notes and mitigations:

- No open burning or incineration — all thermal processes are controlled and contained in hypothetical module enclosures.
- Wastewater handling must include filtration and reuse; any microplastic risk is handled by mechanical filtration and retention (not modeled in detail in the

prototype).
- Crew safety: crew tasks are minimized using quick versus full modes, and modules are imagined as glovebox-style, low-hazard enclosures for handling EVA and chemical residues.

How to use the web app (step-by-step)

1. Open the RedCycle application (served locally with Vite at `http://localhost:8080` when running locally).
2. Explore the Inventory panel: drag an inventory row to a Module Panel to stage material for processing.
3. Use the Module Panel to choose Quick or Full processing. Observe consumed energy/water impacts in the HUD.
4. Check the HUD for recovered/wasted totals and the wastePool available-for-reuse metric.
5. Open the Mars map: click a legend type to reveal per-location markers and view weighted centroids to prioritize collection.
6. Try the scenario pages (Residence, Celebrations, Discoveries) for suggested step-by-step examples and workflows preconfigured for the materials in each scenario.

Project summary & required narrative answers

What did you develop?

A small, interactive web prototype that models a reuse-first, multi-module recycling system for a Mars habitat at Jezero Crater. It includes inventory, a waste pool, processing modules with tunable modes, mass-balance tracking, and a map visualizer for collection planning.

How does it address the challenge?

By prioritizing reuse-first behavior, implementing secondary recovery, and making resource tradeoffs explicit, it demonstrates practical strategies to maximize recovery, minimize fresh inputs, and limit unusable outputs. Scenario workflows show how specific mission activities (habitat unpacking, celebrations, experimental residues) can feed into the system.

Why is it important?

Small design choices (reuse-first, secondary recovery, local aggregation) can significantly reduce resupply needs and increase mission resilience. The prototype lets stakeholders experiment with these choices quickly and see mass-balance outcomes.

What tools / code / hardware used?

- Frontend: React + TypeScript, Vite dev server
- Styling: Tailwind CSS utilities and shadcn UI components
- Map & visualization: Canvas-based component with DOM tooltip overlay
- Local persistence: browser localStorage
- Development environment: Node.js, npm

How is the project creative?

It adapts simple, real-world circular economy patterns to the constraints of an off-world habitat. The wastePool reuse-first policy and the map-centroid visualization make the prototype practical: they combine process optimization with spatial planning in a compact, educational tool.

What factors were considered?

Mass balance, energy and water tradeoffs, crew time, local collection logistics, regenerative reuse (wastePool), and safety (avoid incineration, avoid PFAS/microplastics). We also assumed continuous electricity and limited water available, matching the challenge guidance.

AI usage disclosure

Per the submission policy, we disclose how AI tools were used in the project:

- Code & logic: portions of the application code (hook logic, map updates, documentation) were drafted with assistance from an AI coding assistant to accelerate iteration. All generated code was reviewed and adapted by a human developer before being committed to the repository.
- Images & media: none of the core visual assets in the repo are AI-generated. If AI-generated images are added, they will be marked with a visible watermark per the rules.
- Text & documentation: this guide was produced and refined with AI assistance; this disclosure appears in the document metadata visible to reviewers.

If you plan to submit this project, include this paragraph in your written submission describing AI use and attach or embed any watermarked images as required by the competition rules.

Limitations and recommended next steps

- Replace the simple browser `prompt()`-based map editor with an in-app modal that stores explicit location→material mappings.
- Add per-material wastePool breakdown and a one-click "use all waste" action in the HUD.
- Model time and energy budgets per module for scheduling and capacity planning.
- Add unit and integration tests for `processMaterials` to validate mass-balance under edge cases (zero input, pool-only, large numbers).
- Explore automated PDF export in CI with Pandoc or headless Chrome to generate submission-ready artifacts.

Generated: October 5, 2025 — this file documents the RedCycle prototype, how it maps to the Jezero Crater challenge, and answers the project's required narrative prompts.