# Adventure Map Game - Design Document - Product Requirements Document (PRD)

Project Name: Adventure Map Educational Game

Version: 1.0 (Concept Demo)

Platform: Web Browser (Node.js Local Server)

## 1. Executive Summary

A browser-based, single-player adventure game designed for children. The game features a linear map where progress is gated by educational mini-games. The core differentiator is an **Adaptive Difficulty Engine** that adjusts challenge levels in real-time to prevent frustration, coupled with a **Mercy Rule** mechanism. The system includes a robust **Admin Dashboard** for parents/teachers to configure curricula and view performance analytics.

## 2. User Personas

### Primary Persona: The Player (Child)

* **Age:** 5–9 years old.
* **Characteristics:** Likes visual feedback, easily frustrated by repetitive failure, may have limited reading skills.
* **Goal:** Wants to reach the end of the map and unlock cool characters.
* **Pain Point:** Getting stuck on a hard math problem and being unable to play the rest of the game.

### Secondary Persona: The Administrator (Parent/Teacher)

* **Context:** Using the game for educational reinforcement (homeschooling or classroom station).
* **Goal:** Wants to reinforce specific skills (e.g., "Practice Level 2 Math") and track progress.
* **Pain Point:** Games that are too easy (boring) or too hard (tantrums), and lack of visibility into what the child is actually learning.

## 3. Detailed User Experience (UX) - The Child

### 3.1. The "Adventure" Flow

1. **Onboarding:**
   * **Visual:** Large, colorful "Start Adventure" button.
   * **Character Select:** Child scrolls through 3-5 avatars (e.g., Knight, Astronaut, Bunny). Selection triggers a fun sound effect and a specific "pose" animation.
2. **The Map View:**
   * The camera pans over a winding or circular path with 8-10 stops (Nodes).
   * **Audio Prompt:** Voiceover asks, *"Are you ready for your adventure?"*
   * **Action:** Child clicks the blinking "Next Node." Character walks to Node 1.
3. **The Challenge (The Modal):**
   * A window overlays the map. Background dims to focus attention.
   * **Instruction:** Audio & Text instructions play immediately (e.g., *"Match the cards!"*).
   * **Gameplay:** Child interacts with the mini-game.
4. **Feedback Loops (The Critical Path):**
   * **Scenario A: Success (1st-3rd try):**
     + *Visual:* Confetti explosion. Stars awarded (3 for 1st try, 2 for 2nd, 1 for 3rd).
     + *Action:* "Next" button appears. Modal closes. Character walks to next node.
   * **Scenario B: Failure:**
     + *Visual:* Gentle "Oh no" sound. No red 'X' marks (avoid negative reinforcement).
     + *Action:* "Try Again" button bounces.
   * **Scenario C: The Mercy Rule (After 3 Fails):**
     + *Logic:* The system detects the "stuck" state.
     + *Visual:* An encouraging character (Helper Bot) appears.
     + *Audio:* *"Good effort! Let's skip this one for now."*
     + *Action:* The path unlocks automatically. No stars awarded, but progress continues.

### 3.2. The Victory State

* **Level Complete:** Upon reaching the final node, a chest opens.
* **Reward:** Total stars are tallied. If the threshold is met, a new Character Skin is unlocked for the next session.

## 4. Detailed Teacher/Parent Experience (AX) - The Admin

### 4.1. The Dashboard (Landing Page)

* **Access:** A secure/hidden tab or button (e.g., "Grown-up Zone").
* **Overview:** Shows a list of active profiles (e.g., "Alex", "Sam").
* **Quick Actions:** "Start Session" or "View Report".

### 4.2. Configuration Tab (The Setup)

This allows the adult to tailor the game to the child's current capability.

* **Campaign Settings:**
  + *Map Theme:* Select "Forest", "Space", or "Random".
  + *Length:* Set number of levels (1-5).
* **Curriculum Control (Game Registry):**
  + List of available game types (Math, Memory, Pattern).
  + **Toggle:** Enable/Disable specific games (e.g., Disable "Reading" if the child is pre-literate).
  + **Difficulty Floor/Ceiling:** Set the starting difficulty (1-5) and the maximum difficulty.

### 4.3. The Analytics Report (The "Report Card")

A visual breakdown of the child's performance, stored in the local JSON files.

* **Skill Heatmap:**
  + *Green:* Skills passed on 1st try (e.g., Memory Lvl 3).
  + *Yellow:* Skills requiring retries.
  + *Red:* Skills that triggered the "Mercy Rule" (skipped).
* **Time Tracking:** Total time played vs. Time spent on puzzles.
* **Intervention Flag:** Highlight specific games where the difficulty level dropped due to adaptive logic (e.g., *"Alex struggled with Level 3 Math, system downgraded to Level 2"*).

## 5. Functional Requirements

### 5.1. Game Engine

* **FR-01:** The system shall render a responsive map on HTML5 Canvas with winding or circular paths based on configuration.
* **FR-02:** The system must support "pure JS" modals that block interaction with the map until resolved.
* **FR-03:** The system must preserve state (current node, stars, difficulty) to data/sessions/ after every node completion.

### 5.2. Adaptive Difficulty Logic

* **FR-04:** If a player fails a node 3 times, the MercyFlag must be set to true.
* **FR-05:** If MercyFlag is true, the *next* node's difficulty must be CurrentDifficulty - 1 (Minimum 1).
* **FR-06:** If a player passes a node on the 1st try, the *next* node's difficulty returns to BaseDifficulty.

### 5.3. Asset Management

* **FR-07:** The system must allow dynamic injection of "Mini-Game" scripts from the /minigames/ directory.
* **FR-08:** The system must use SVG placeholders for all graphics (Characters, Icons) to ensure zero external dependency for the demo.

## 6. Non-Functional Requirements

* **NFR-01 (Performance):** The game must load within 2 seconds on a standard browser (Chrome/Firefox/Edge).
* **NFR-02 (Accessibility):** All text instructions must have a corresponding Text-to-Speech (TTS) or pre-recorded audio trigger button.
* **NFR-03 (Usability):** All interactive touch targets (buttons) must be at least 48x48 pixels for child motor skills.
* **NFR-04 (Data):** All data must be stored locally in JSON format. No database installation required.

## 7. Success Metrics (KPIs)

* **Completion Rate:** % of sessions that reach the Final Node (Goal: >90% thanks to Mercy Rule).
* **Frustration Index:** Frequency of "Rage Quits" (closing browser mid-puzzle).
* **Teacher Efficiency:** Time taken to configure a custom lesson plan (Goal: < 2 minutes).