# **Product Requirements Document (PRD)**

## **Project Title**

**Cyberpunk Financial Dashboard**

## **Document Version**

v1.0

## **Author**

Joy

## **Date**

2025-05-29

## **1. Purpose and Overview**

The purpose of this project is to design and develop an interactive **Cyberpunk Financial Dashboard** that provides users with a clear and visually engaging summary of their personal finance data. The dashboard aims to present total income, total expenses, net balance, and a detailed breakdown of expenses by category.

This tool is intended for individuals who want to monitor and analyze their financial transactions in real-time, helping improve budgeting and financial awareness with a modern cyberpunk aesthetic.

## **2. Goals and Objectives**

* **Goal 1:** Create a visually appealing and user-friendly dashboard using modern web technologies (HTML, CSS, JavaScript).
* **Goal 2:** Integrate backend functionality via Google Apps Script to fetch real-time financial data from a Google Sheet.
* **Goal 3:** Display aggregated data metrics: total income, total expenses, and net balance dynamically.
* **Goal 4:** Provide detailed breakdown of expenses by category in a tabular format.
* **Goal 5:** Enable automatic periodic updates to keep financial data current without manual refresh.
* **Goal 6:** Ensure the dashboard is accessible via a simple URL (hosted on GitHub Pages).

## **3. Scope**

### **Included:**

* Frontend dashboard with cyberpunk styling and responsive design.
* Real-time data fetching from a Google Sheet backend via Apps Script API.
* Display of total income, expenses, and net balance.
* Expense categories displayed in a dynamic table.
* Auto-refresh of data every 60 seconds.
* Hosting on GitHub Pages for public access.

### **Excluded:**

* User authentication or multi-user support.
* Data input/editing through the dashboard interface.
* Advanced analytics or forecasting features.

## **4. Stakeholders**

* **Primary User:** Individual users wanting a personal finance dashboard.
* **Developer:** Joy (project creator and maintainer).
* **Reviewers:** Academic instructor or peers (if applicable).

## **5. User Stories**

* **As a user,** I want to see my total income, total expenses, and net balance on a dashboard so that I can quickly understand my financial status.
* **As a user,** I want to see a categorized breakdown of my expenses so that I can identify where my money is spent.
* **As a user,** I want the dashboard data to update automatically so I do not have to refresh the page manually.
* **As a user,** I want the dashboard to have an engaging cyberpunk theme for a futuristic and visually appealing experience.

## **6. Functional Requirements**

| **ID** | **Requirement Description** | **Priority** |
| --- | --- | --- |
| FR1 | Display total income value fetched from Google Sheets data | High |
| FR2 | Display total expenses value fetched from Google Sheets data | High |
| FR3 | Calculate and display net balance (income - expenses) | High |
| FR4 | Display a table showing expenses broken down by category | High |
| FR5 | Fetch financial data from Google Apps Script Web API | High |
| FR6 | Automatically refresh dashboard data every 60 seconds | Medium |
| FR7 | Use cyberpunk style CSS for UI components (cards, table) | Medium |
| FR8 | Deploy dashboard on GitHub Pages accessible via public URL | Medium |

## **7. Non-Functional Requirements**

* **Performance:** Data fetch and rendering should complete within 3 seconds on a standard broadband connection.
* **Usability:** Clear, readable fonts and high-contrast color scheme for accessibility.
* **Maintainability:** Codebase modular and commented for easy updates.
* **Security:** No sensitive user data stored in the frontend or exposed via API.
* **Availability:** Dashboard hosted on reliable GitHub Pages platform.

## **8. User Interface / Design**

* The dashboard features three main cards aligned horizontally showing Total Income, Total Expenses, and Net Balance.
* Below the cards, an expenses-by-category table is presented centered on the page.
* Colors: black background (#0d0d0d), neon pink (#ff00ff), cyan (#00fff7) to reflect cyberpunk style.
* Font: 'Orbitron' from Google Fonts for futuristic styling.
* Responsive layout with hover animations on cards for interactivity.

## **9. Technical Architecture**

* **Frontend:** HTML, CSS, JavaScript.
* **Backend Data Source:** Google Sheets storing financial transactions (date, category, description, type, amount).
* **Backend Logic:** Google Apps Script to aggregate financial data and serve it via RESTful Web API.
* **Deployment:** Static site hosted on GitHub Pages.
* **Data Flow:** Google Sheets → Apps Script API → Fetch JSON data → Frontend Dashboard UI

## **10. Success Criteria**

* The dashboard correctly displays up-to-date financial summaries reflecting the latest Google Sheets data.
* Users report satisfaction with the UI design and ease of understanding their finances.
* Data refresh occurs automatically without page reload.
* The site is accessible publicly via the provided URL:<https://joy-creator04.github.io/project2/>

## **11. Risks and Mitigation**

| **Risk** | **Mitigation** |
| --- | --- |
| API downtime or slow response | Implement error handling and retry mechanisms |
| Data inconsistency in Sheets | Validate data entries and sanitize inputs |
| Styling may not display well on mobile **12. Extension Goals**  * **Categorized Spending Support:** Enhanced table structure to display expenses by category. * **Dark Theme with Glassmorphism:** Applied modern cyberpunk styling for improved readability and aesthetic. * **Currency Selector (UI only):** Dropdown implemented; backend logic pending. * **Clarity Improvements:** Added tooltips and inline labels based on instructor feedback. * **Mobile Responsiveness:** Refactored layout using Flexbox to ensure proper display on smaller screens.   AI chat history: <https://chatgpt.com/share/683f0992-3254-8000-a1c4-88381452c98e>  <https://chatgpt.com/share/683f09d0-fe90-8000-9089-2605c7596184>  <https://chatgpt.com/share/683f0a05-cf60-8000-973b-765fe32e605c> | Use responsive CSS design |