**Internal Dashboard**

**TOC**

**BLUF**

**Executive Summary**

**Internal Dashboard Platform**

* **Core capabilities of the platform**
  + Consolidates projects, notes, and theme switching in a single-page interface.
  + Offers real-time visibility into ongoing initiatives, statuses, and categories.

**Analysis and Visualisation Layers**

* **Breakdown of presentation layers**
  + Project categorisation (e.g., All, Strategic, Tactical).
  + Notes panel for structured content (e.g., CSV-based details).
  + Dynamic theme selection (e.g., Day, Night, Coffee).

**Technical Architecture**

* **Explanation of the modular design and key technologies used**
  + Data ingestion (Papa Parse) for CSV-driven notes.
  + Theming via CSS variables.
  + Single-page JavaScript logic for rapid page switching.

**Project and Theming Hierarchy**

* **The structure of pages and customisation**
  + Surface-level views (e.g., category filters, quick stats).
  + Interactive notes panel for deeper insights.
  + Full theme overrides to switch interface look/feel.

**Operational Applications**

* **Use cases such as:**
  + Tracking internal projects and resource allocations.
  + Centralising notes for cross-departmental collaboration.
  + Enabling rapid UI adjustments for diverse user preferences.

**BLUF**

The **Internal Dashboard** is a **single-page application** that consolidates project tracking and thematic control under one unified interface. By leveraging dynamic data loading, a flexible colour‐theming engine, and a collapsible sidebar navigation system, it offers a highly customisable, centralised environment for managing project information and notes. This architecture promotes real-time accessibility, clarity, and swift navigation—providing a streamlined user experience ideal for organisational or multi-departmental use.

**1. Core Capabilities:**

The platform consolidates project data and user notes into a streamlined single-page application by:

* **Unified Project Tracking**: Displays multiple projects, each with a brief description, tags, and links for quick reference.
* **Dynamic Category Views**: Filters items into “All,” “Strategic,” or “Tactical” pages, providing straightforward navigation.
* **Real-Time Note Access**: Pulls structured notes from CSV files via Papa Parse, grouping them by section for deeper insights.
* **Interactive UI**: Collapsible sidebar, quick transitions between pages, and a notes panel that slides in on demand.
* **Theming Engine**: Switches instantly between pre-defined colour schemes (Day, Night, Coffee), tailoring the user experience.

**2. Technical Strengths:**

Built on a flexible, theme-oriented architecture, the platform delivers:

* **Papa Parse Integration** for straightforward CSV ingestion and structured note handling.
* **CSS Variable–Based Theming**, enabling quick UI transformations without rewriting core CSS.
* **Modular JavaScript** design (via window.projects array and showPage logic), ensuring maintainability and scalability.
* **DOM Manipulation and Event Handling** through a clear single-page application approach, enhancing performance and simplicity.
* **Expandable Infrastructure**, allowing easy addition of new pages, categories, and theming options.

**3. Analytical Layers:**

While primarily focused on organisation and presentation, it leverages layered data interactions to provide:

* **Page-Level Views**: Splits projects by category, surfacing relevant items for direct comparison or analysis.
* **Structured Notes**: Sections and sub-sections within CSV-based content reveal hierarchical insights about each project.
* **Interactive Elements**: Toggleable notes panel, tooltips, and dynamic download/export options for quick sharing or offline review.

**4. Operational Applications:**

This unified dashboard supports internal oversight and collaboration, including:

* **Project Oversight and Management**: Keeps stakeholders up to date on progress, changes, or relevant documentation in a single location.
* **Rapid Configuration**: Quickly adapts to changing organisational needs by adding categories, theme palettes, or CSV-based data sources.
* **Cross-Team Communication**: Centralises notes, links, and references, minimising confusion and misaligned data.

**This Internal Dashboard equips teams with a cohesive, user-friendly interface for project visibility.** By combining a robust notes system, real-time theming, and concise category views, it provides a scalable solution for internal coordination, data exploration, and responsive decision-making.

**Executive Summary**

This document outlines a **lightweight yet extensible** Internal Dashboard solution, providing a comprehensive view of multiple projects and associated information. By combining a **multi-page layout** (powered by JavaScript show/hide logic), **notes panel** (populated via CSV data), and **theming** (achieved through CSS variables), the platform offers a robust yet user-friendly interface for teams seeking real-time insights.

Key features include:

* **Page-based Navigation**: Users switch between “All,” “Strategic,” and “Tactical” project lists, or visit central “Home,” “Settings,” and “About” pages.
* **Dynamic Project Cards**: Each project is rendered with essential details, links, and a quick-access notes panel.
* **Theme Switching**: Multiple customisable colour schemes (e.g., Day, Night, Coffee) available at runtime, instantly shifting the UI’s look and feel.
* **Real-time Notes**: Projects pull in notes from a CSV (via Papa Parse), sorting them into user-friendly sections with export features.
* **Badge Counts**: Numeric counters on sidebar submenus display the number of projects in each category, providing quick overviews.

Designed for internal stakeholders such as **project managers**, **team leads**, and **administrative staff**, the platform ensures data centralisation, theming flexibility, and rapid insight generation, all in a single-page ecosystem.

**Internal Dashboard Platform**

**Core Capabilities**

1. **Consolidated Project Oversight**  
   Provides a clear, interactive catalogue of all organisational projects, neatly subdivided into categories (e.g., “All,” “Strategic,” and “Tactical”).
2. **Project-Specific Notes**  
   Leverages CSV-based storage for structured notes and key data points. A dedicated panel slides out on demand, offering deeper insights, export options (MD, PDF, DOCX, TXT), and screenshots.
3. **Adaptive Theming**  
   Enforces consistent design through CSS variables in :root. Multiple theme sets (Day, Night, Coffee) are triggered on demand, seamlessly switching background, text, and accent colours.

**User Interface and Theming**

* **Collapsible Sidebar Navigation**: The sidebar, normally 60px wide, expands to 250px on hover. Main navigation items toggle submenus, revealing pages or theme switches.
* **Responsive Pages**: A showPage function hides all .page elements, displaying only the user‐requested section. This pattern keeps the interface clean and context-specific.
* **Theme Switching**: A simple applyColorScheme('schemeName') call overwrites CSS variables. By referencing colorSchemes[schemeName], each theme fully updates colours, icons, and button states in real time.

**Data-Driven Insights**

* **CSV Integration**: The platform uses Papa Parse to ingest notes or project data from CSV files, structuring them by project ID and logical sections.
* **Badge Counts**: Real-time counters for “All,” “Strategic,” and “Tactical” automatically compute how many projects fall under each category, offering top-level status at a glance.
* **Export Options**: The notes panel supports direct downloads of content in multiple formats, encouraging both local archiving and easy sharing with external stakeholders.

**Analysis and Visualisation Layers**

While the Internal Dashboard focuses primarily on **organisation** and **presentation** rather than deep data analytics, it still offers several layers of interaction and insight:

1. **Project Listing Layer**
   * Displays standardised project cards, each featuring a title, description, tags, icons, relevant links, and optional CSV download buttons.
   * Implements a flexible filter system: “All,” “Strategic,” and “Tactical” categories for quick scoping.
2. **Notes and Documentation Layer**
   * Retrieves textual data from external CSV files, grouping notes by sections (e.g., “Research Methods,” “Findings,” “Implementation”).
   * Offers subordered bullet points to differentiate main ideas from deeper sub‐topics or sub‐points.
3. **Theme Customisation Layer**
   * Allows direct manipulation of CSS variable sets to apply new backgrounds, text colours, and hover states.
   * Transforms user experience, toggling between lighter (Day/Coffee) or darker (Night) styles to suit different usage contexts.

By structuring data presentation into these interactive layers, the dashboard ensures no key detail is overlooked, promoting clarity and efficient navigation.

**Technical Architecture**

1. **Single-Page Application (SPA)**
   * **HTML**: Structures multiple .page <div>s for Home, Settings, About, and each project category. A global .container encloses the collapsible sidebar and main content area.
   * **CSS**: Uses **CSS variables** as theming anchors, plus additional classes for layout, transitions, and interactive states.
   * **JavaScript**:
     + Orchestrates page toggles via showPage(), clearing and populating .project-list elements dynamically.
     + Integrates CSV parsing (Papa Parse) to load external notes.
     + Implements theming with a colorSchemes object and an applyColorScheme function that rewrites CSS variables.
   * **DOM Event Handling**: A DOMContentLoaded listener seeds initial data (window.projects), sets the default page, updates badge counts, and captures the baseline “original” theme.
2. **Notes Panel Slide-Out**
   * Uses a fixed <div class="notes-panel"> anchored off the right edge, toggling .active to slide in.
   * **toggleNotes(projectName)** populates project‐specific notes, handles CSV data sorting, and presents optional screenshot galleries or doc downloads.
3. **Data Management**
   * **window.projects**: A JavaScript array storing metadata like name, description, links, GitHub repos, icons, screenshots, and categories.
   * **Papa Parse**: Ingests project-notes.csv to build a nested object structure (projectNotes[projectId].sections[sectionName]) for each project.
   * **Export Tools**: Document, CSV, and PDF downloads triggered by dedicated JavaScript functions (e.g., downloadPDF(), downloadCSV(), etc.).

This integrated architecture is **modular** and **scalable**, allowing additional pages, note types, or new CSS themes without major disruption.

**Operational Applications**

1. **Departmental Projects Management**
   * Centralises varied departmental projects under one roof, enabling easy cross-reference and progress tracking.
   * Notes panel fosters knowledge sharing, collating strategic and tactical insights in a single place.
2. **Internal Stakeholder Visibility**
   * Executives, team leads, and analysts gain immediate visibility into project scope, categorisation, and resources.
   * Quick theme switching can adapt visuals for different usage contexts (e.g., presentations in well-lit or darkened rooms).
3. **Rapid Prototyping and Collaboration**
   * The flexible codebase supports swift adjustments to UI, theming, or CSV structures—ideal for an agile environment.
   * Integrations with further data sources or external documentation can evolve as organisational demands grow.
4. **Scalable Platform for Future Modules**
   * Because each page is triggered by a straightforward showPage() call, new pages (e.g., “Reporting,” “Metrics,” or “Audit Trail”) can be easily appended.
   * Additional theming can be introduced by adding new scheme definitions and a single submenu link.

The **Internal Dashboard** thus serves as a **foundational hub**: bridging high-level oversight, user-driven customisation, and straightforward expansion paths for deeper analysis or extended functionalities. Its focus on clarity, theming, and dynamic data ensures it remains **adaptable** to the evolving needs of modern teams.

**High-Level Design**

1. **HTML Structure (index.html):**
   * A **collapsible sidebar** (<nav class="sidebar">) contains **top-level nav items** (Home, Projects, Settings, About, Theme). Some have submenus for sub-navigation (e.g., “All,” “Strategic,” “Tactical”).
   * Each “page” is a <div class="page" id="somethingPage">, which by default is hidden with style="display:none;". Clicking on a nav item calls showPage('somethingPage'), setting it visible while hiding other pages.
   * A **notes panel** slides in from the right, triggered by toggleNotes(), showing project-specific details loaded from a CSV file.
2. **CSS & Themes (style.css):**
   * Uses **CSS variables** (e.g., --background-color, --text-color) in :root to define a default theme.
   * The .sidebar is initially 60px wide and expands to 250px on hover. Submenus are collapsed or expanded depending on .expanded.
   * Each page has a .project-list into which dynamic project cards are inserted.
   * Additional **colour schemes** are placed in the JavaScript colorSchemes object. By calling applyColorScheme('day') or 'night' (etc.), you overwrite these variables on the document.documentElement.style, instantly swapping the theme.
3. **JavaScript Behaviour (script.js):**
   * **Project Data**: An array of window.projects contains details like name, images, tags, category (strategic/tactical), etc.
   * **Page Switching**: showPage(pageId) hides all .page elements, then displays the requested one. If that page has a .project-list container, it filters window.projects by category (e.g., “tactical”) and populates the page with project cards.
   * **Notes Panel**: toggleNotes(projectName) fetches or parses project-notes.csv using Papa Parse, then builds the HTML for that project’s notes and displays it in .notes-panel.
   * **Theme Switching**: applyColorScheme(schemeName) loops over a dictionary of CSS variables (in colorSchemes[...]) and applies them to document.documentElement, changing backgrounds, text colours, button styles, etc.
   * **File Downloads**: Various functions (downloadCSV, downloadTechnicalDoc, downloadDOCX, etc.) fetch or generate files (CSV, PDF, docx) and trigger the browser’s download.
   * **Version & Release Date**: Inserted on the Home page by setting document.getElementById("appVersion").textContent and .releaseDate.textContent after the DOM loads.
4. **Additional Features & Nuances**:
   * **Submenus** expand or collapse when you call toggleSubmenu(this), adding or removing .expanded.
   * **Badge Counts**: “All,” “Strategic,” and “Tactical” each have a live numeric badge. updateBadgeCounts() calculates how many projects belong in each category.
   * **Tooltips**: A small snippet moves a tooltip if the user hovers over .maid-tooltip.
   * **Day/Night/Coffee** Themes\*\*: The “day” scheme is captured on load with getComputedStylesAsObject(), and the “day,” “night,” or “coffee” schemes override some or all of these variables as they are moved between.

**Takeaways**

This single-page application effectively **decouples** content, style, and dynamic behaviour:

* **HTML** for the structure and placeholders (pages, sidebar, notes panel).
* **CSS** variables let you seamlessly **switch** colour themes, each controlling text/background colour, hover states, and more.
* **JavaScript** handles all dynamic tasks: **loading** or **filtering** data, **toggling** pages and submenus, **enabling** downloads, and **applying** theme overrides at runtime.

The result is a cleanly modular design: you can easily add new pages, define new projects or theming, or incorporate additional logic (e.g., advanced filtering or user preferences) while keeping your code well organised.