

Advanced Project Journal — Resource & Cost Management

A comprehensive, single-file web application for freelance developers and project managers to track work entries, manage resources, calculate costs, and generate detailed financial reports. Built with a focus on the Israeli market (₪ / ILS currency, right-to-left Hebrew UI).

Overview

This tool serves as an all-in-one project management journal designed for freelancers, consultants, and small teams. It enables you to log work entries, track AI resource costs (Claude, ChatGPT, Perplexity, etc.), apply discounts, manage payments, forecast revenue, and calculate ROI — all from a single HTML file with zero dependencies.

No server required. Everything runs in the browser with data persisted to `localStorage`.

Features

Multi-Project Management

- Create and manage multiple projects via a tabbed interface
- Each project maintains its own independent data, resources, and entries
- Switch between projects with a single click

Resource Management

- Define custom resources with daily cost rates (₪/day)
- Pre-loaded with common AI tools (Claude, ChatGPT, Perplexity)
- Add, edit, or remove resources per project
- Select which resources were used per work entry

Work Entry Logging

- Log entries with start/end dates, topic, component built, and notes
- Automatic work-day and total-hours calculation
- Select multiple resources per entry with automatic cost aggregation
- Configurable hourly rate and trainee discount percentage
- **Date stamp** — each entry is automatically timestamped when added

- **Inline delete** — remove any entry with a single click on the  icon

Live Calculation Preview

- Preview the full cost breakdown *before* committing an entry
- Shows work days, total hours, hourly rate, gross charge
- Detailed discount calculation with before/after amounts
- Per-resource cost breakdown (cost/day × days)
- Grand total with all components

Data Table

- Scrollable, responsive table with all entry details
- Columns: #, Date Added, Start Date, End Date, Work Days, Hours, Topic, Component, Resources, Notes, Resource Cost, Work Charge, Discount %, After Discount, Total
- Inline delete button per row
- Whole-number formatting (no unnecessary decimals)

Summary Dashboard

- Summary cards: total work days, total hours, total resource cost, total income
- Grand total box with aggregated stats across all entries

Token Cost Calculator

- Designed for AI-powered SaaS projects
- Configure: tokens per treatment, tokens per search, price per 1K tokens, USD/ILS rate
- Input therapist count, treatments per day, daily searches
- Auto-calculates daily and monthly token costs
- **AI Recommendation Engine:** suggests optimal monthly token limits per subscriber based on target profit margins

ROI Calculator

- Input fixed monthly costs, token costs, subscriber count, subscription revenue
- Calculates net monthly profit and initial investment payback period

Monthly Maintenance Cost Tracker

- Track hosting, token costs, maintenance hours, programmer rates, payment fees

- Software subscription management (add/remove tools with monthly costs)
- Miscellaneous cost tracking
- Auto-totals all maintenance expenses

Revenue Forecasting

- Input subscriber count, subscription price, additional revenue
- Projects total monthly revenue
- Calculates minimum subscribers needed to break even
- Profit/loss report with ROI timeline

Financial Summary

- Gross income, resource costs, net profit, average hourly rate
- Payment tracking: invoice dates, amounts billed, amounts received
- Running balance due calculation

Data Persistence & Export

- **Auto-save** — optional 60-second auto-save to `localStorage`
- **Manual save** — save snapshots on demand
- **Report history** — view, reload, or delete past saved reports
- **JSON export/import** — full project data backup and restore
- **Email export** — generate formatted email with project summary
- **Print-ready** — clean print layout (hides UI controls)
- **Shareable link** — encode project data into a URL
- **WhatsApp sharing** — share summary via WhatsApp
- **Markdown export** — generate `.md` file for NotebookLM or documentation
- **HTML report** — generate a standalone, styled HTML report

Responsive Design

- Fully responsive: desktop, tablet, and mobile layouts
- Touch-friendly with minimum 44px tap targets
- Safe-area support for notched devices
- Horizontal scroll for data tables on small screens

Getting Started

Prerequisites

- Any modern web browser (Chrome, Firefox, Safari, Edge)
- No server, build tools, or dependencies required

Installation

1. Clone the repository:

```
bash  
  
git clone https://github.com/YOUR_USERNAME/project-journal.git  
cd project-journal
```

2. Open in browser:

```
bash  
  
open index.html
```

Or simply double-click `index.html` in your file manager.

That's it — the app runs entirely client-side.

Usage Guide

1. Managing Resources

When you first open the app, three default AI resources are pre-loaded (Claude, ChatGPT, Perplexity). To customize:

- **Add a resource:** Enter the resource name and daily cost (₪/day), then click .
- **Edit cost:** Change the cost directly in the resource list input field.
- **Remove:** Click the  button next to any resource.

2. Adding Work Entries

1. Fill in the entry form:

- **Start & End dates** — the app auto-calculates work days
- **Topic** — general subject area
- **Component** — what was built
- **Hours/day and Hourly rate (₪)**
- **Trainee discount %** — default 35%, adjustable
- **Select resources** — click resource tags to toggle selection
- **Notes** — optional free-text

2. Click  **ב' סה"כ** (Calculate Total) to preview the full breakdown.
3. Review the preview, then click  **הוסף רשותה** (Add Entry) to commit.

Each entry is automatically timestamped with the date and time it was added.

3. Managing the Data Table

- All entries appear in the data table with a sequential number and date stamp.
- Click the  icon on any row to delete that entry (with confirmation).
- The table scrolls horizontally on mobile devices.

4. Token Cost Planning (for AI SaaS)

Scroll to the planning section to:

1. Set token pricing parameters (price per 1K tokens, USD/ILS rate).
2. Enter usage data (therapists, treatments/day, tokens/treatment).
3. View calculated daily and monthly costs.
4. Get AI-powered recommendations for subscriber token limits.

5. Financial Forecasting

1. **Maintenance costs** — enter all fixed monthly expenses.
2. **Revenue forecast** — enter subscriber count and pricing.
3. **View the dashboard** — see profit/loss, ROI timeline, and minimum subscriber recommendations.

6. Saving & Exporting

Action	Button	Description
Auto-save	 Toggle	Saves every 60 seconds to browser storage
Manual save		Snapshot current state

Action	Button	Description
Email		Opens email client with formatted summary
Print		Print-optimized layout
Share link		Generates URL with encoded data
WhatsApp		Share via WhatsApp
Markdown		Export <code>.md</code> file for NotebookLM
JSON import		Load a previously exported report

Project Structure

```
project-journal/
├── index.html    # Entire application (single file)
└── README.md     # This file
```

The application is a single self-contained HTML file with embedded CSS and JavaScript — no build step, no dependencies, no external assets.

Data Storage

All data is stored in the browser's `localStorage`. This means:

- Data persists across browser sessions on the same device
- Clearing browser data will erase saved projects
- Use **JSON export** to create backups
- Use **JSON import** to restore data on a new device or browser

Browser Compatibility

Browser	Supported
Chrome 80+	

Browser	Supported
Firefox 75+	✓
Safari 13+	✓
Edge 80+	✓
Mobile Chrome	✓
Mobile Safari	✓

Mobile Support

The app is fully responsive with three breakpoints:

- **Desktop** ($> 768\text{px}$) — full grid layouts, side-by-side panels
- **Tablet** ($\leq 768\text{px}$) — stacked layouts, 2-column summaries
- **Phone** ($\leq 480\text{px}$) — single-column layout, compact spacing

Touch targets meet the 44px minimum for accessibility. The data table scrolls horizontally on smaller screens.

Contributing

Contributions are welcome! To contribute:

1. Fork the repository
2. Create a feature branch (`(git checkout -b feature/my-feature)`)
3. Commit your changes (`(git commit -m 'Add my feature')`)
4. Push to the branch (`(git push origin feature/my-feature)`)
5. Open a Pull Request

License

This project is licensed under the MIT License. See [LICENSE](#) for details.

Acknowledgments

- Built with vanilla HTML, CSS, and JavaScript
- Designed for the Israeli freelance developer community
- Optimized for AI-powered SaaS project cost management