

Study Tracker

A modern, feature-rich productivity dashboard built with pure HTML, CSS, and JavaScript. Perfect for tracking tasks, managing deadlines, and boosting productivity with built-in analytics and Pomodoro timer.



Figure 1: Version



Figure 2: HTML5



Figure 3: CSS3

✨ Features

Task Management

- **Create tasks** with title, description, deadline, category, and priority
- **Status tracking**: Pending, In Progress, Completed, Skipped
- **Edit tasks** inline with modal interface
- **Delete tasks** with confirmation
- **Search and filter** by status, category, or keyword
- **Sort tasks** by date, priority, or alphabetically

Analytics & Visualization

- **Task distribution chart** (Doughnut) - see breakdown by category
- **Weekly progress chart** (Bar) - track completion over time
- **Activity heatmap** (GitHub-style) - visualize daily productivity
- **Completion statistics** - success rate and task counts

Productivity Tools

- **Pomodoro Timer** - 25/5/15 minute intervals with visual progress
- **Dark Mode** - toggle between light and dark themes
- **Keyboard shortcuts** for quick actions
- **Confetti animation** on task completion



Figure 4: JavaScript

Data Management

- **LocalStorage persistence** - data survives page reloads
- **Export data** to JSON backup file
- **Import data** from JSON backup
- **Mobile responsive** - works on all devices

🚀 Quick Start

1. **Clone or download** this repository
2. **Open** index.html in any modern browser
3. **Start tracking** your tasks!

No build process, no dependencies (except Chart.js CDN), no server required.

📁 Project Structure

```
study-tracker/
  index.html # Main HTML file
  css/
    style.css # Main styles with CSS variables
    components.css # UI components & animations
    responsive.css # Mobile responsiveness
  js/
    app.js # Application entry point
    utils.js # Helper functions
    storage.js # LocalStorage management
    taskManager.js # Task CRUD operations
    charts.js # Chart.js integration
    timer.js # Pomodoro timer
    ui.js # DOM manipulation
  assets/
    icons/ # SVG icons (optional)
  README.md
```

⌨️ Keyboard Shortcuts

Shortcut	Action
Ctrl/Cmd + N	Focus new task input
Ctrl/Cmd + F	Focus search box
Ctrl/Cmd + D	Toggle dark mode
Escape	Close modal

🎨 Design System

Colors

- **Primary:** Indigo (#6366f1)
- **Success:** Emerald (#10b981)
- **Warning:** Amber (#f59e0b)
- **Danger:** Red (#ef4444)

Typography

- **Font:** Inter (Google Fonts)
- **Scale:** 12px to 32px

CSS Architecture

- **CSS Variables** for theming
- **Mobile-first** responsive design
- **BEM-inspired** naming convention
- **Smooth animations** and transitions

🛠 Technical Highlights

Clean Architecture

```
// Modular design with separation of concerns
TaskManager // Business logic
UI // Presentation layer
Storage // Data persistence
Charts // Visualization
Timer // Productivity tool
Utils // Helper functions
```

Modern JavaScript Features

- ES6+ syntax (arrow functions, destructuring, spread)
- Async/await for file operations
- Event delegation for performance
- Debounced search input
- Intersection Observer ready

Accessibility

- Semantic HTML5 elements
- ARIA labels for screen readers
- Keyboard navigation support

- Focus visible indicators
- Color contrast compliance

Data Schema

```
{
  id: "string", // Unique identifier
  title: "string", // Task name
  description: "string", // Task details
  category: "study|work|personal|health",
  priority: "low|medium|high",
  deadline: "YYYY-MM-DD",
  status: "pending|in-progress|completed|skipped",
  createdAt: "ISO date",
  completedAt: "ISO date|null"
}
```

Why This Project?

This project demonstrates:

1. **Frontend Architecture** - Modular, maintainable code structure
2. **State Management** - Client-side data persistence
3. **Data Visualization** - Interactive charts and graphs
4. **UX Design** - Intuitive interface with feedback
5. **Performance** - Optimized rendering and event handling
6. **Accessibility** - Inclusive design principles

Perfect for CS portfolio - shows understanding of:

- DOM manipulation
- Event handling
- Data structures
- Algorithmic thinking
- UI/UX design
- Code organization

Future Enhancements

- Drag & drop between status columns (Kanban board)
- Recurring tasks
- Task tags/labels
- Time tracking per task
- Goal setting and milestones
- Cloud sync (Firebase)

Service Worker for offline support

PWA install support

License

MIT License - feel free to use for your own portfolio!

Credits

- Chart.js for beautiful charts
- Inter Font by Google Fonts
- Icons via Unicode emoji

Built with ❤ for CS Portfolio