Chrome Starting Page Dashboard v1.0

Personal Chrome Start Page Dashboard | Generated 2026-02-23

> *Your personal Chrome start page dashboard — runs locally on Node.js with zero external dependencies.*

Overview

A fully-featured Chrome start page dashboard that runs on a local Node.js server. All data stays on your machine — no external servers, no data transmission, no accounts needed. Bookmark management, notes, todos, calendar, habit tracking, and more, all wrapped in a customizable glassmorphism UI.

Forked and extensively upgraded from the original chrome-starting-page project.

Features

• **Spotlight Search** — unified search across bookmarks, todos, notes, and D-Days with Korean initial consonant support (`Ctrl+F`)

• **Keyword Quick Search** — type `yt query`, `nv query`, `gh query` to search YouTube, Naver, GitHub instantly

• **Multi-Card System** — Bookmark, Notepad, Todo, Calendar, and Habit cards with drag & drop reordering

• **Multi-Page Tabs** — organize cards into separate pages with tab navigation

• **Smart Bookmarks** — drag & drop from browser, duplicate detection, NEW badges, cross-card movement

• **Theme Engine** — dark/light/auto modes, 8 accent colors, glass effect presets, background slideshow

• **Auto Backup** — smart retention policy (7-day full → 30-day daily → auto-delete, max 50 files)

• **Offline Support** — Service Worker caching keeps the dashboard viewable even when the server is down

• **Weather & D-Day** — current weather display and countdown to important dates

• **Pomodoro Timer** — built-in focus timer with statistics tracking

• **Profile Management** — save and switch between multiple configurations

• **Zero Dependencies** — pure Node.js built-in modules, no npm packages required

Tech Stack

Category	Technology
Server	Node.js ≥ 18 (built-in modules only, no npm packages)
Frontend	Vanilla JavaScript + CSS (no frameworks)
Data Storage	Local JSON files (`data/` directory)

Category	Technology
Caching	Service Worker
Port	1111 (configurable)

Prerequisites

- **Node.js** v18 or higher — Download LTS
- Verify installation: `node -v` → should show `v18.x.x` or higher

Installation

Windows Recommended Auto Setup

1. Place the project folder where you want it (e.g., `C:\Users\YourName\Dashboard`)
2. Double-click `setup_windows.bat`
3. When complete, open `http://localhost:1111`

The auto setup will check Node.js, start the server in background, and register it to Windows Startup.

macOS Recommended Auto Setup

```
cd ~/Dashboard
chmod +x setup_mac.sh
./setup_mac.sh
```

The auto setup will check Node.js, register a macOS LaunchAgent, and start the server.

Linux

```
cd ~/Dashboard
node server.js
```

For auto-start, register a systemd service:

```
sudo tee /etc/systemd/system/dashboard.service &lt;&lt; EOF
[Unit]
Description=Chrome Starting Page Dashboard
After=network.target

[Service]
Type=simple
User=$USER
WorkingDirectory=$HOME/Dashboard
ExecStart=$(which node) server.js
Restart=on-failure

[Install]
WantedBy=multi-user.target
EOF

sudo systemctl enable dashboard
sudo systemctl start dashboard
```

Manual Start All Platforms

```
cd /path/to/Dashboard
node server.js
```

The terminal will show `[Dashboard v7.2] http://localhost:1111/` when ready.

Usage

Set as Chrome Start Page

1. Navigate to `chrome://settings/onStartup`

2. Select **"Open a specific page or set of pages"**

3. Click **"Add a new page"** → enter `http://localhost:1111`

Optional: Install New Tab Redirect extension and set redirect URL to `http://localhost:1111` for new tabs.

Keyboard Shortcuts

Shortcut	Action
`/` or `Ctrl+K`	Focus search bar
`S`	Open settings
`E`	Toggle edit mode
`T`	Toggle theme (dark/light)
`Ctrl+F`	Spotlight search (search all bookmarks/todos/notes/D-Days)
`1–9`	Open nth bookmark in first card

Search

- **Default**: Type in search bar + Enter (switchable between Google/Naver/YouTube)
- **Keywords**: `yt query`, `nv query`, `gh query`, `g query`, `tw query`, `map query`
- **Spotlight**: `Ctrl+F` for unified search with Korean initial consonant support (■■ → "■■■")
- **Commands**: Type `>` in Spotlight for commands (`todo`, `timer`, `edit`, `focus`)

Bookmark Management

- **Add**: Click + button in edit mode, or drag a URL from browser onto a card
- **Reorder**: Drag & drop within a card
- **Move**: Drag bookmark to another card
- **Delete**: Hover → click ✕ (undo supported)
- **Duplicates**: Settings → Data → "Check Duplicate URLs"
- **NEW Badge**: Auto-shown for 7 days on new bookmarks

Card Types

- **Bookmark Card** — favorite link collections
- **Notepad Card** — quick notes (multiple cards supported)

- **Todo Card** — tasks with priority, tags, due dates, recurring (multiple cards supported)
- **Calendar Card** — monthly/weekly calendar with event management
- **Habit Card** — daily habit checklist

Card Management

- **Add/Remove**: + button at bottom / ✕ button on card (undo supported)
- **Collapse**: Click card title (state persists across sessions)
- **Resize**: Toggle 1x/2x in edit mode
- **Reorder**: Drag via ■ handle on left
- **Color**: Set per-card background color in edit mode

Settings S key

- **Theme**: Dark / Light / Auto, 8 accent colors
- **Glass Effect**: Transparency, blur intensity, presets (Transparent/Default/Opaque)
- **Background**: Solid color, image upload, slideshow
- **Weather**: Enter city name for current weather display
- **Search**: Change default engine, custom keyword search
- **D-Day**: Countdown to important dates
- **Data**: Export/Import, profiles, duplicate check, trash

How It Works

The dashboard runs a lightweight Node.js HTTP server on port 1111 that serves static files and provides a REST API for data persistence. All user data is stored as JSON files in the `data/` directory. The frontend is pure vanilla JavaScript and CSS — no build step, no bundling, no framework.

Key architectural decisions:

- **Zero npm dependencies** — the server uses only Node.js built-in `http`, `fs`, and `path` modules
- **Service Worker** — caches pages for offline access even when the server is stopped
- **Smart backup** — automatic periodic backups with a retention policy that keeps recent backups and prunes old ones
- **3-tier favicon** — Google Favicon → DuckDuckGo → auto-generated colored letter icons

Project Structure

```
Dashboard/
■■■ data/                        ← All user data (auto-managed)
■    ■■■ bookmarks.json          ← Bookmarks
■    ■■■ config.json             ← Settings
■    ■■■ notes.json              ← Notes
■    ■■■ todos.json              ← Todos
■    ■■■ events.json             ← Calendar events
■    ■■■ ddays.json              ← D-Day countdowns
■    ■■■ usage.json              ← Usage statistics
■    ■■■ trash.json              ← Trash / recycle bin
■    ■■■ pomo-stats.json         ← Pomodoro statistics
■    ■■■ icons/                  ← Uploaded icons
```

```
■    ■■■ profiles/              ← Profile data
■    ■■■ backups/               ← Auto backups (smart retention)
■■■ assets/                     ← Background images, PWA icons
■■■ server.js                   ← Server (edit only for port change)
■■■ index.html                  ← Page structure
■■■ style.css                   ← Styles (themes, glass effects, animations)
■■■ script.js                   ← Feature logic (2,800+ lines)
■■■ restart.bat                 ← Server restart (Windows)
■■■ run_server_background.bat    ← Background execution (Windows startup)
■■■ setup_windows.bat           ← Windows auto setup
■■■ setup_mac.sh                ← macOS auto setup
```

Security

- All data is stored locally — nothing is transmitted externally
- No system registry modifications
- No global config files or hidden folders outside the project
- The server only listens on `localhost:1111` — not accessible from other devices
- Memory usage is ~15MB when idle
- Works offline (except weather widget)

Troubleshooting

Symptom	Cause	Solution
"Port 1111 already in use"	Previous server still running	Double-click `restart.bat` (Windows) or `kill` the process
Server won't start	Node.js not installed	Run `node -v` to verify, install from nodejs.org
Page not showing	Wrong URL protocol	Use `http://localhost:1111` (not https)
Data not saving	Server connection lost	Restart server; check write permissions on `data/`
Bookmark icons missing	Favicon fetch failed	Auto-fallback: Google → DuckDuckGo → colored letters

Complete Uninstall

This program only needs two things removed: the project folder and the auto-start registration. It leaves nothing in system registry, global configs, or hidden folders.

Windows:

```
# 1. Remove auto-start
del "%APPDATA%\Microsoft\Windows\Start Menu\Programs\Startup\Dashboard_StartPage.lnk"
# 2. Stop server
taskkill /f /im node.exe
# 3. Delete project folder
```

```
# 4. Remove http://localhost:1111 from chrome://settings/onStartup
```

macOS:

```
launchctl unload ~/Library/LaunchAgents/com.dashboard.startpage.plist
rm ~/Library/LaunchAgents/com.dashboard.startpage.plist
rm -rf ~/Dashboard
```

Linux (systemd):

```
sudo systemctl stop dashboard &amp;&amp; sudo systemctl disable dashboard
sudo rm /etc/systemd/system/dashboard.service &amp;&amp; sudo systemctl daemon-reload
rm -rf ~/Dashboard
```

Acknowledgments

- Original project: sodam-AI/chrome-starting-page

License

This project is licensed under the MIT License — see the LICENSE file for details.