SONNET 4.5 PROJECT GENERATION PROMPT

You are my senior Go engineer.

Generate a complete, minimal but clean Go project for a macOS network usage monitor.

The project has two executables:

1. netmon-service → a background daemon that runs every second and stores network usage stats in SQLite

2. netmon → a CLI tool with commands like stats today, stats week, etc.

REQUIREMENTS

Language & Tooling

- Write everything in Go 1.22+

- Use modules (initialize with go mod init netmon)

- Use standard library wherever possible

- Use SQLite via:

import "modernc.org/sqlite"

and the standard database/sql driver shim

Architecture

Use this exact folder structure:

project-root/

■■ cmd/

■ ■■ netmon-service/

■ ■ ■■ main.go

■ ■■ netmon/

■ ■■ main.go

■■ internal/

■ ■■ collector/

■ ■ ■■ collector.go

■ ■ ■■ interfaces.go

■ ■■ db/

■ ■ ■■ db.go

■ ■■ stats/

■ ■■ stats.go

■■ go.mod

■■ README.md

netmon-service Requirements

- Run an infinite loop with a 1-second ticker

- On each tick:

- Read all network interfaces

- Extract inbound/outbound byte counters

- Insert into SQLite traffic_logs

- Graceful shutdown with SIGINT/SIGTERM

Database Schema

CREATE TABLE IF NOT EXISTS traffic_logs (

id INTEGER PRIMARY KEY AUTOINCREMENT,

timestamp INTEGER NOT NULL,

interface TEXT NOT NULL,

bytes_in INTEGER NOT NULL,

bytes_out INTEGER NOT NULL

);

netmon CLI Requirements

- Use standard library flag parsing

Commands:

- netmon stats today

- netmon stats week

- netmon stats interfaces

Computation Rules

- Use bytes_in/bytes_out deltas

- Totals = sum of deltas

- Peak throughput = max delta/second

Code Quality Requirements

- No global state

- Centralized DB connection

- Documented functions

- Clean CLI output

Expected Output

1. Full project structure

2. Complete source code

3. go.mod

4. README.md with build & run instructions