

Go + Gin + PostgreSQL Backend Learning Path

Phase 0 — Go Mindset (2 days)

Goal: Understand *how Go thinks*, not syntax memorization.

Learn

- Go philosophy (simplicity, explicitness)
- Project layout conventions
- Why Go avoids frameworks like NestJS

Must-know differences

- No classes → structs + methods
- No DI container → constructor injection
- Errors are values
- Interfaces are implicit

Resource

https://go.dev/doc/effective_go

Phase 1 — Go Language Essentials (3–5 days)

Topics

- Packages & modules
- Types & zero values
- Structs & methods
- Interfaces (VERY important)
- Pointers (when & why)
- Slices, maps
- defer
- Enums with iota

Phase 2 — Error Handling & Testing (2 days)

Go's quality comes from this phase.

- Learn
- Idiomatic error handling
- Custom errors
- Wrapping errors (fmt.Errorf, errors.Is)
- Table-driven tests
- Mocks using interfaces

Phase 3 — HTTP & Gin (4-5 days)

Now we move to real backend work.

Learn

- net/http basics (before Gin)
- Gin routing
- Context usage
- Middleware
- Request binding & validation
- Response patterns

NestJS mapping

NestJS	Go + Gin
Controller	Handler
Service	Service
Guard	Middleware
Pipe	Validation
Interceptor	Middleware

Phase 4 — PostgreSQL & Data Layer (4-5 days)

You know Postgres well, focus on **Go patterns**.

Start with

- **GORM** (easy transition)

Learn

- DB connection pooling
- Models & migrations
- Transactions
- Pagination
- Soft deletes

(optional but recommended)

- **SQLC** for type-safe SQL

Phase 5 — Architecture (NestJS-Style, Go-Native) (3-4 days)

Learn

- Manual dependency injection
- Interface-driven design
- Avoiding over-abstraction

Phase 6 — Auth, Security & Middleware (3-4 days)

Learn

- JWT authentication
- Middleware chaining
- Role-based access

- Request context usage
- Rate limiting

Phase 7 — Concurrency & Performance (3-4 days)

Learn

- Goroutines
- Channels
- Worker pools
- sync package
- context.Context (critical for APIs)

Practice

- Parallel DB queries
- Background jobs
- Graceful shutdown

Phase 8 — Create Project (Best Learning) (7-10 days)

Recommended project (perfect for you)

Rebuild one NestJS module in Go

Example:

- Auth + Users
- Catalog + Products
- POS staff service
(or you can suggest a project)

Stack

- Gin
- PostgreSQL
- GORM / SQLC
- JWT