bdd-with-godog

2-Day Hands-On Training Program: BDD, Cucumber, Gherkin, and godog (with Go)

> **Audience:** Proficient in Go, beginner in BDD and associated tools
> **Objective:** By the end of Day 2, you should be able to design, implement, and run BDD scenarios using godog in a real-world Go application
> **Outcome:** You will be able to:

- Write effective .feature files in Gherkin
- Implement step definitions in Go using godog
- Integrate BDD into your software lifecycle
  - pgsql
  - Copy
  - Edit

📅 Day 1: Foundations of BDD, Gherkin & Cucumber

🕘 Morning Session: Introduction and Theory

1. Welcome & Overview (30 mins)

- Introduce course structure, goals, and expectations
- Understand why BDD is critical in Agile projects

2. What is BDD? (1 hour)

- Difference between TDD, ATDD, and BDD
- Benefits: collaboration, clarity, client alignment

3. Introduction to Gherkin & Cucumber (1 hour)

- Gherkin syntax: `Feature`, `Scenario`, `Given`, `When`, `Then`
- Cucumber's role in executing BDD scenarios

☕ Coffee Break (15 mins)

4. Gherkin Syntax Deep Dive (1 hour)

- Keywords: `And`, `But`, `Background`, `Scenario Outline`
- Good practices: clarity, avoiding technical jargon
- ✍ *Exercise:* Write Gherkin scenarios for a "To-Do List" feature

5. Editor Setup (30 mins)

- Install plugins for Gherkin syntax in VS Code / GoLand
- Validate syntax, formatting, and readability

🕐 Afternoon Session: Hands-On with Gherkin & Feature Files

6. Writing Your First Feature Files (1.5 hours)

- Draft `feature` files for:
  - User login
  - Task creation
  - Error scenarios
- Peer review your scenarios

7. Understanding Step Definitions (1 hour)

- What are step definitions?
- Mapping natural language to executable code
- Cucumber flow: Feature file → Step definition → App code

Q&A and Homework (30 mins)

- Review key learnings
- 📝 *Homework:* Finalize your `.feature` files for Day 2 integration

---

📅 Day 2: Implementing BDD with godog in Go

🕘 Morning Session: Setup & Integration

Recap of Day 1 (30 mins)

- Open discussion on Gherkin clarity and use cases

8. Introduction to `godog` (1 hour)

- Overview of `godog` – Cucumber for Go
- Installing godog using `go install github.com/cucumber/godog/cmd/godog@latest`
- Project structure for godog

9. Project Setup & Environment (45 mins)

- Create sample Go app (`user login` or `task manager`)
- Add `*.feature` file under `/features/`
- Create and register step definition file in Go

☕ Coffee Break (15 mins)

🕑 Late Morning: Step Definitions in Go

10. Writing Step Definitions (1 hour)

- `func (s *Suite) iEnterUsername(username string) error { ... }`
- Matching Gherkin regex with Go functions
- Using godog Context and Data Tables

11. Running and Debugging (30 mins)

- `go test -v`
- Handling undefined steps
- Interpreting error outputs and failures

🕒 Afternoon Session: End-to-End Project & Best Practices

12. Mini Project: End-to-End BDD with Go - Web Service (1.5 hours)

- Project: Build a minimal "User Data Management Service" with REST APIs:
  - GET all existing Users
  - GET existing User by Id
  - CREATE new User
  - UPDATE existing User
  - Delete one User by Id
  - Delete all Users
- Steps:
  - Write `.feature` files
  - Implement Go steps
  - Execute godog tests

13. Mini Project: End-to-End BDD with Go - Web Service (1.5 hours)

- Project: Build a minimal "User Data Management Service" with Graphql APIs:
  - GET all existing Users
  - GET existing User by Id
  - CREATE new User
  - UPDATE existing User
  - Delete one User by Id
  - Delete all Users
- Steps:
  - Write `.feature` files
  - Implement Go steps
  - Execute godog tests

13. Mini Project: End-to-End BDD with Go - Standalone Application (1.5 hours)

- Project: Build a minimal "Task Manager" with features:
  - Create Task
  - List Tasks
  - Error validation
- Steps:
  - Write `.feature` files
  - Implement Go steps
  - Execute godog tests

14. Best Practices for Real Projects (45 mins)

- Keep feature files non-technical for stakeholder readability
- Separate test logic from core logic
- Use godog hooks (BeforeScenario, AfterStep)

15. Wrap-Up & Resources (30 mins)

- Key Takeaways
- Common mistakes and how to avoid them
- 📚 Recommended Reading:
  - https://cucumber.io/docs/guides/overview/
  - https://github.com/cucumber/godog
  - https://go.dev/doc/
- 💡 Next Steps:
  - Start integrating BDD into client project
  - Use CI tools like GitHub Actions to run godog tests

📎 Appendix: Tools and Setup Commands

```
# Install godog
go install github.com/cucumber/godog/cmd/godog@latest

# Run godog tests
godog run

# Example project structure
.
├── go.mod
├── main.go
├── features/
│   └── login.feature
└── stepdefs/
    └── login_steps.go
```