Advanced Programming 1
Assignment 4 – Project Milestone 2 (Core System Implementation)

Deadline & Submission
Deadline: 23:59 01.02.2026
Submission:

- ZIP archive named Assignment4_TeamName.zip
- Upload to Moodle
- Project must compile and run using `go run .`

Defense: All practice classes in Week 9 (02.02-06.02), you will be asked to download your assignment solution from the lms and demonstrate your results and answer questions.

Goal

Check whether the team has implemented a working core system based on the approved design from Assignment 3. This assignment evaluates backend implementation, data model consistency, and basic runtime behavior. It is NOT required to complete all planned features. This is Milestone 2 of the final project.

What Must Be Implemented

Based on the project proposal and architecture from Assignment 3, teams must implement:

- One running backend application (monolith, microservices are ok)
- Core domain models matching the ERD
- 3-5 core features or use cases
- Basic persistence (in-memory or database)

The system must start, accept input (from postman or frontend), process data, and return results.

Technical Requirements

1. Backend Application (30%)
    a. Implement HTTP server using net/http (or approved framework)

b. At least 3 working endpoints or entry points
c. JSON input and output

2. Data Model & Storage (25%)
    a. Data structures must match ERD from Assignment 3
    b. CRUD operations for at least one core entity
    c. Safe data access (no crashes during concurrent requests)

3. Concurrency or Background Processing (15%)
    a. Use at least one goroutine
    b. Examples: background worker, async processing, channel-based logic

4. Git Workflow & Team Contribution (15%)
    a. Feature branches per team member
    b. At least 2 commits per member
    c. Meaningful commit messages

5. Demo & Explanation (15%)
    a. Demonstrate running system
    b. Show implemented features
    c. Explain how implementation follows Assignment 3 design

What is NOT required (yet)
The following are intentionally NOT required at this stage:

- Complete feature set
- Authentication and authorization
- Full UI or frontend
- Error handling for all edge cases
- Performance optimization

These will be addressed during the final project submission.

Grading Rubric

- Backend Application: 30%
- Data Model & Storage: 25%
- Concurrency / Async Logic: 15%
- Git Workflow: 15%
- Demo & Explanation: 15%

Important Notes (points can be deducted if there is radical changes)

1. The implementation must follow the architecture approved in Assignment 3
2. Major design changes must be justified during defense
3. Code quality is less important than correctness and clarity at this stage
4. This assignment prepares your project for the final revision and presentation