

Lecture 33

Scalefocus Go Academy

Today's Agenda

— Final Project Guidelines

Final Project Guidelines

Final Project Guidelines

The final project will require you to develop a To-Do Web application.

We will provide you with a skeleton of the application. This application will have its own frontend (provided by us).

Your goal is to implement the rest of the application, so that it works as expected by the frontend.

Deadline: May 20th

ScaleFocus TODO

Lists

List name...	✓	✓
primary		
Groceries		×
Chores		×

Tasks

Please type text...	✓	Add
<input type="checkbox"/> Eggs		Remove
<input type="checkbox"/> Meat		Remove

Final Project Guidelines

This includes implementing:

1. all the necessary Web API endpoints (using Echo or Gin)
2. all the business logic
3. all the persistence
 - You are free to choose any library for working with a relational database, but you must store and fetch To-Dos from a DB
 - Setting up a DB server is not necessary. SQLite is perfectly fine for this project
 - Any relational DB is Ok, but please include the SQL schema
4. All the tests
 - ensure at least 70% code coverage

Final Project Guidelines

You can develop your application in multiple phases, in order to keep moving forward. We advise to do so.

1. Develop all the API endpoints, business logic, persistence, and tests with a single user in mind
2. Introduce support for multiple users through **Basic Authentication**. Both Echo and Gin have middleware that supports that. Feel free to research and one to your app. This will save you from having to implement login UI on your own
 - However, you still need to make sure that the rest of your app can work with multiple users
3. Use an external weather API to provide the current day's temperature
4. Add an API endpoint that will export all the tasks for a given user as a single CSV file.

Final Project Guidelines

Things to keep in mind:

- Code quality and proper separation of concerns.
- Let the tests guide you. If possible, design your code in such a way that you don't have to use a database during testing
- If you cannot, please, make sure that you set up the DB properly at the beginning of each test, and clean it up afterwards

Final Project Guidelines

How to start the server

If you decide to use Echo:

```
go run cmd/echo/main.go
```

If you decide to use Gin

```
go run cmd/gin/main.go
```

This will automatically start an HTTP server on port 3000, accessible via `http://localhost:3000`

NOTE: Please, always start both `main.go` files from the root of your project.

Final Project Guidelines

Available Endpoints

`/app` - this one starts the frontend. As you will see, it throws a bunch of errors.

`/swagger` - opens an interactive documentation web page, where you will be able to see what you need to implement. It will also allow you to send test requests to your API.

You will have to implement all the requests as part of the `/api` path.

Good Luck!
