



Programación Go

» Web Servers

// Práctica clase 2 - Go Web

Objetivo

El objetivo de esta guía práctica es que podamos afianzar los conceptos sobre las capas de una aplicación web, vistos en el módulo de Go Web. Para esto vamos a plantear una serie de ejercicios simples e incrementales (ya que vamos a ir trabajando y agregando complejidad a lo que tenemos que construir), lo que nos permitirá repasar los temas que estudiamos.

Forma de trabajo

Vamos a continuar trabajando con la temática elegida en la primera clase del módulo.

¿Are you ready? 😊👍



Ejercicio 1 - Generar paquete internal

Se debe separar la estructura del proyecto y como primer paso generando el paquete internal, en el paquete internal deben estar todas las funcionalidades que no dependan de paquetes externos.

Dentro del paquete deben estar las capas:

1. Servicio, debe contener la lógica de nuestra aplicación.
 - a. Se debe crear el archivo service.go.
 - b. Se debe generar la interface Service con todos sus métodos.
 - c. Se debe generar la estructura service que contenga el repositorio.
 - d. Se debe generar una función que devuelva el Servicio.
 - e. Se deben implementar todos los métodos correspondientes a las operaciones a realizar (GetAll, Store, etc..).
2. Repositorio, debe tener el acceso a la variable guardada en memoria.
 - a. Se debe crear el archivo repository.go
 - b. Se debe crear la estructura de la entidad
 - c. Se deben crear las variables globales donde guardar las entidades
 - d. Se debe generar la interface Repository con todos sus métodos
 - e. Se debe generar la estructura repository
 - f. Se debe generar una función que devuelva el Repositorio
 - g. Se deben implementar todos los métodos correspondientes a las operaciones a realizar (GetAll, Store, etc..)



Ejercicio 2 - Generar paquete server

Se debe separar la estructura del proyecto, como segundo paso se debe generar el paquete server donde se agregaran las funcionalidades del proyecto que dependan de paquetes externos y el main del programa.

Dentro del paquete deben estar:

1. El main del programa.
 - a. Se debe importar e inyectar el repositorio, servicio y handler
 - b. Se debe implementar el router para los diferentes endpoints
2. El paquete handler con el controlador de la entidad seleccionada.
 - a. Se debe generar la estructura request
 - b. Se debe generar la estructura del controlador que tenga como campo el servicio
 - c. Se debe generar la función que retorne el controlador
 - d. Se deben generar todos los métodos correspondientes a los endpoints