BACKEND—

// main.go

package main

import (

"github.com/gin-gonic/gin"

"github.com/graphql-go/graphql"

"net/http"

)

func main() {

r := gin.Default()

r.GET("/api", func(c \*gin.Context) {

c.JSON(http.StatusOK, gin.H{

"message": "Hello from Gin!",

})

})

// GraphQL setup

r.GET("/graphql", func(c \*gin.Context) {

// Define GraphQL Schema

fields := graphql.Fields{

"hello": &graphql.Field{

Type: graphql.String,

Description: "A simple hello field",

Resolve: func(params graphql.ResolveParams) (interface{}, error) {

return "Hello, GraphQL!", nil

},

},

}

schemaConfig := graphql.SchemaConfig{

Query: graphql.NewObject(graphql.ObjectConfig{

Name: "RootQuery",

Fields: fields,

}),

}

schema, \_ := graphql.NewSchema(schemaConfig)

// Execute GraphQL Query

query := c.DefaultQuery("query", "{}")

result := graphql.Do(graphql.Params{

Schema: schema,

RequestString: query,

})

c.JSON(http.StatusOK, result)

})

r.Run(":8080")

}