Nombre: Hernán Javier Aguilar Cruz

CC: 1006503110

Parcial Final - Arquitectura de Software (Práctica)

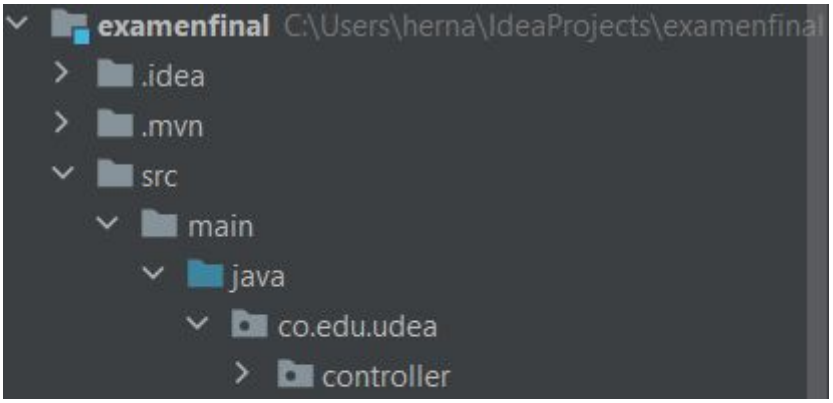
Una vez hemos creado el proyecto con Spring Web activado, ejecutamos el método main de la clase principal.

```java
package co.edu.udea;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class ExamenFinalApplication {

    public static void main(String[] args) { SpringApplication.run(ExamenFinalApplication.class, args); }

}
```

Una vez se termine la ejecución de la página, verificamos que quedan en los puertos 8080.

```
2021-01-23 10:30:02.370  INFO 8724 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port(s): 8080 (http) with context path ''
2021-01-23 10:30:02.380  INFO 8724 --- [           main] co.edu.udea.ExamenFinalApplication       : Started ExamenFinalApplication in 1.122 seconds (JVM running
2021-01-23 10:30:02.470  INFO 8724 --- [nio-8080-exec-1] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring DispatcherServlet 'dispatcherServlet'
2021-01-23 10:30:02.470  INFO 8724 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet        : Initializing Servlet 'dispatcherServlet'
2021-01-23 10:30:02.471  INFO 8724 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet        : Completed initialization in 1 ms
```

Procedemos a crear el paquete denominado controller

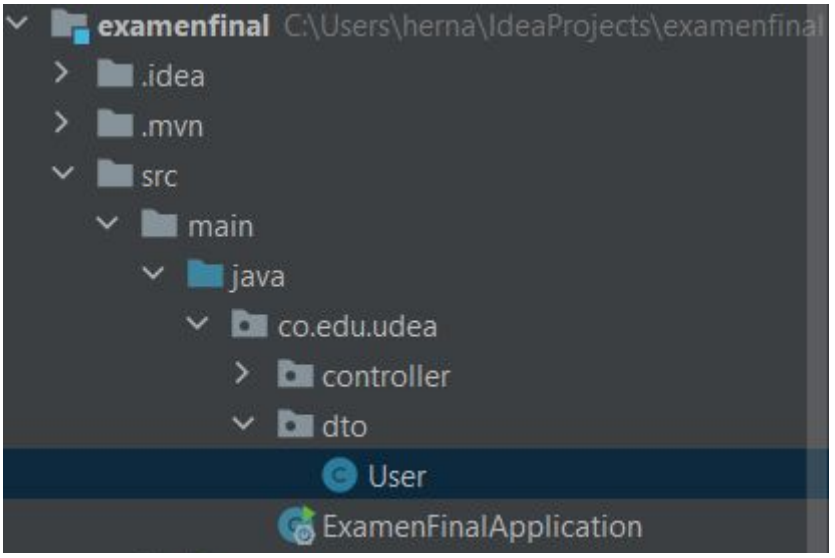


En él, generamos la clase HomeController.

```java
package co.edu.udea.controller;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.ResponseBody;

@Controller
@ResponseBody

public class HomeController {
    @GetMapping(value="/hola")
    public String holaMundo(){
        return ("Hola mundo");
    }
}
```

Verificamos su funcionamiento.

Una vez confirmamos, procedemos a crear el paquete dto con una entidad, en este caso User.



En él creamos el método constructor.

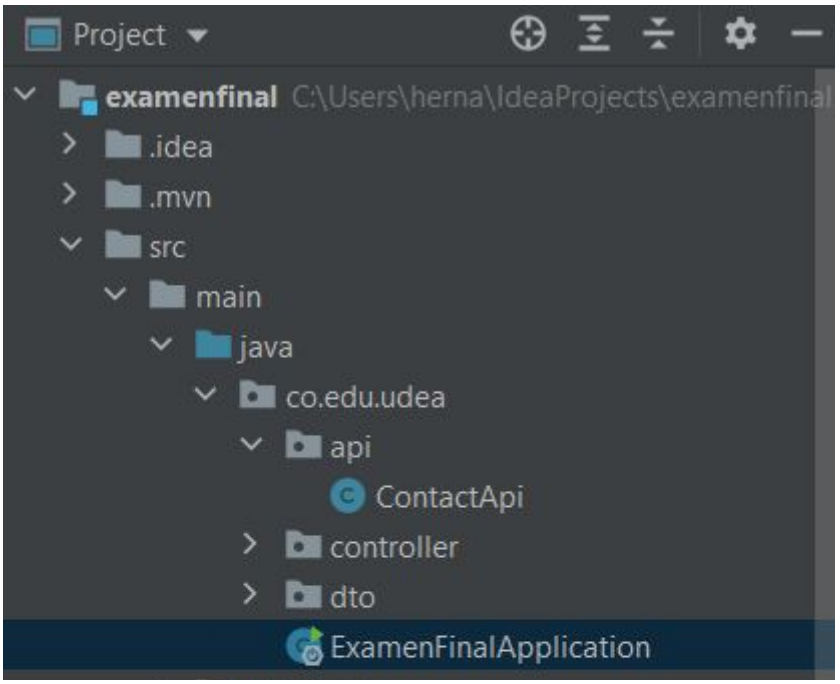
```java
package co.edu.udea.dto;

public class User {

    private long id;
    private String firstName;
    private String lastName;
    private String phoneNumber;
    private String email;

    public User(long id, String firstName, String lastName, String phoneNumber, String email){
        this.id = id;
        this.firstName = firstName;
        this.lastName = lastName;
        this.phoneNumber = phoneNumber;
        this.email = email;
    }
```

Y sus getters y setters.

```java
public long getId(){
    return id;
}

public String getFirstName(){
    return firstName;
}

public String getlastName(){
    return lastName;
}

public String getPhoneNumber(){
    return phoneNumber;
}

public String getEmail(){
    return email;
}
```

```java
public void setId(long id) {
    this.id = id;
}

public void setFirstName(String firstName) {
    this.firstName = firstName;
}

public void setLastName(String lastName) {
    this.lastName = lastName;
}

public void setPhoneNumber(String phoneNumber) {
    this.phoneNumber = phoneNumber;
}

public void setEmail(String email) {
    this.email = email;
}
```

Creamos el api y la clase ContactApi.




```java
package co.edu.udea.api;

import co.edu.udea.dto.User;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.ResponseBody;
import org.springframework.web.bind.annotation.RestController;

@RestController
@ResponseBody

public class ContactApi {
    @GetMapping(value="/contact")
    public User createUser(){
        User contact = new User ( id: 1006503110, firstName: "Hernan", lastName: "Aguilar",
                                  phoneNumber: "333", email: "hernan.aguilarc@udea.edu.co");
        return contact;
    }
}
```

Verificamos entonces su funcionamiento.