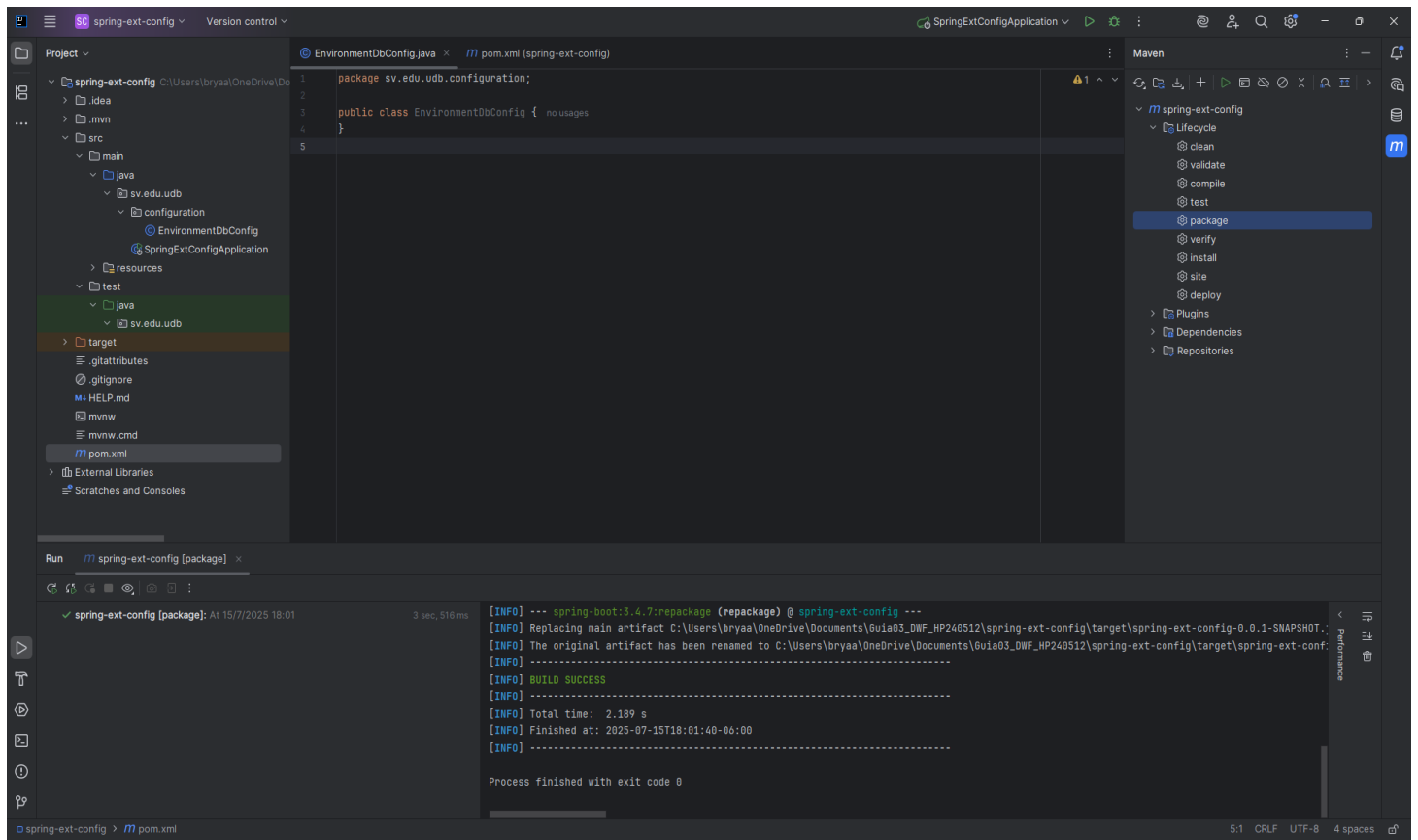
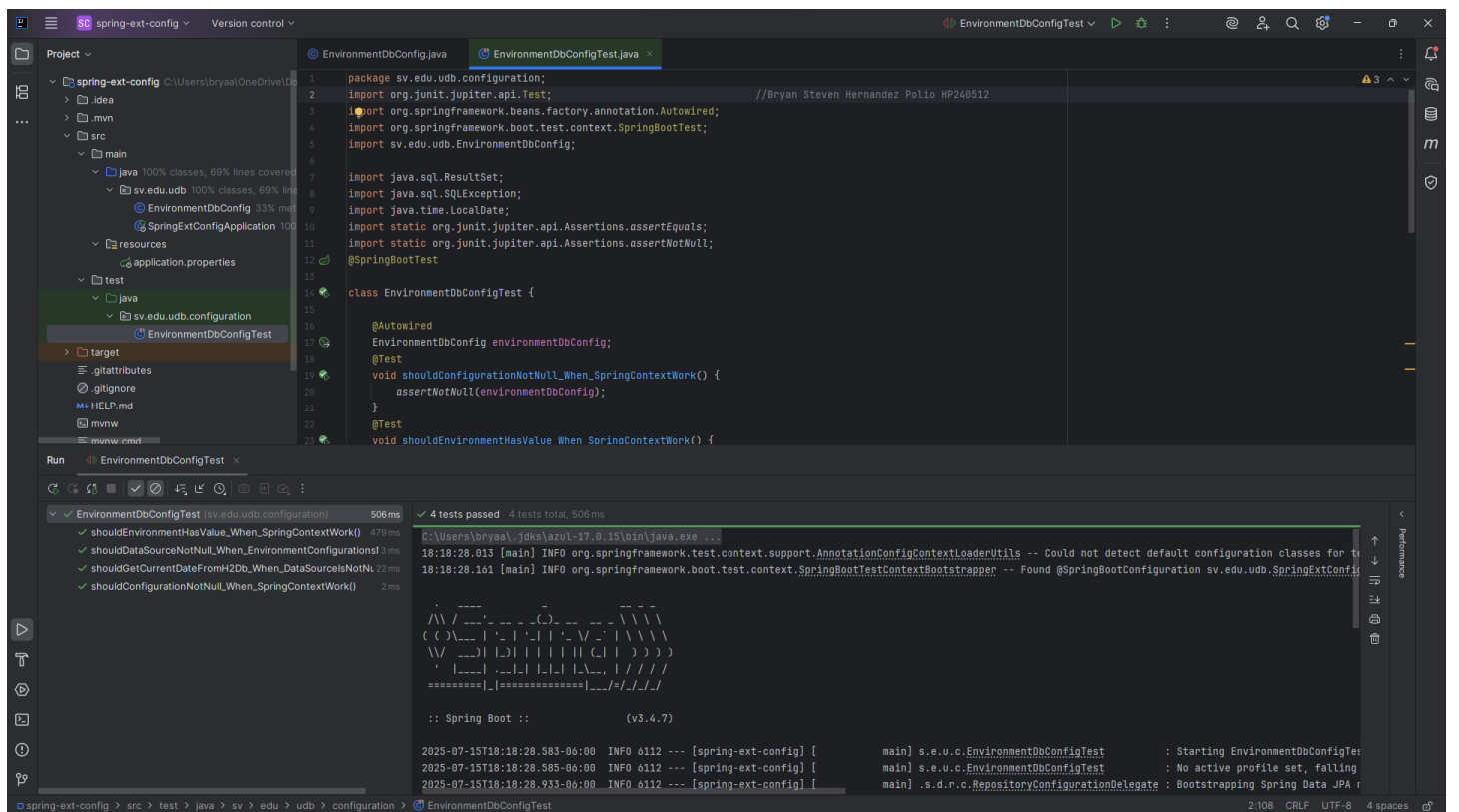


## PRIMERA PRUEBA PASO 9: (se me paso colocar el nombre y carnet aquí)

Bryan Steven Hernández Polio HP240512.



## PRUEBAS UNITARIAS DEL PASO 14:



## PRUEBAS UNITARIAS DEL PASO 16: EnvironmentDbConfig.java con anotacion @Value("\${key}") y Lombok

The screenshot shows an IDE with the following components:

- Project Explorer:** Shows the project structure with folders like `src`, `test`, and `resources`. The `EnvironmentDbConfigTest` class is selected in the `test` folder.
- Code Editor:** Displays the implementation of `EnvironmentDbConfig.java`. The code uses `@Value` for injecting database properties and `@Getter` for getters. It also uses `Objects.requireNonNull` for validation.
- Run Console:** Shows the output of the unit tests. All tests passed, and the console output includes the Spring Boot logo and version information.

```
package sv.edu.udb.configuration;
import lombok.Getter;
import org.springframework.beans.factory.annotation.Value;
import org.springframework.boot.jdbc.DataSourceBuilder;
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import javax.sql.DataSource;
import java.util.Objects;

@Configuration
public class EnvironmentDbConfig {
    private final String dbDriver;
    private final String dbUrl;
    private final String dbUsername;
    private final String dbPassword;

    public EnvironmentDbConfig(final @Value("${db.driver}") String dbDriver,
                              final @Value("${db.url}") String dbUrl,
                              final @Value("${db.username}") String dbUsername,
                              final @Value("${db.password}") String dbPassword) {
        this.dbDriver = Objects.requireNonNull(dbDriver);
        this.dbUrl = Objects.requireNonNull(dbUrl);
        this.dbUsername = Objects.requireNonNull(dbUsername);
        this.dbPassword = Objects.requireNonNull(dbPassword);
    }
}
```

PASO 20: Crearía que el error ocurre porque se esta activando en `application.properties` el perfil `dev` y trata de cargar el otro `application-dev.properties` al mismo tiempo y el otro también tiene esa línea `db.url=jdbc:h2:mem:dev` esa es mi respuesta.

The screenshot shows an IDE with the following components:

- Project Explorer:** Shows the project structure with folders like `src`, `test`, and `resources`. The `EnvironmentDbConfigTest` class is selected in the `test` folder.
- Code Editor:** Displays the implementation of `EnvironmentDbConfigTest.java`. The code uses `@SpringBootTest` and `@Autowired` for testing. It also uses `assertEquals` for assertions.
- Run Console:** Shows the output of the unit tests. One test failed, and the console output includes the Spring Boot logo and version information.

```
import java.sql.ResultSet;
import java.sql.SQLException;
import java.time.LocalDate;
import static org.junit.jupiter.api.Assertions.assertEquals;
import static org.junit.jupiter.api.Assertions.assertNotNull;
import org.springframework.boot.test.context.SpringBootTest;

@SpringBootTest
class EnvironmentDbConfigTest {

    @Autowired
    EnvironmentDbConfig environmentDbConfig;

    @test
    void shouldConfigurationNotNull_When_SpringContextWork() {
        assertNotNull(environmentDbConfig);
    }

    @test
    void shouldEnvironmentHasValue_When_SpringContextWork() {
        final String dbDriverExpected = "org.h2.Driver";
        final String dbUrlExpected = "jdbc:h2:mem:test";
        final String dbUsernameExpected = "SA";
        final String dbPasswordExpected = "";

        assertEquals(dbDriverExpected, environmentDbConfig.getDbDriver());
        assertEquals(dbUrlExpected, environmentDbConfig.getDbUrl());
    }
}
```

## PASO 22 PRUEBAS CON EL CAMBIO EN APPLICATION.PROPERTIES

spring.application.name=spring-ext-config

spring.profiles.active=prod

The screenshot shows an IDE with a project named 'spring-ext-config'. The 'Project' view on the left shows the file structure, including 'resources' with 'application.properties', 'application-dev.properties', and 'application-prod.properties'. The 'Run' view at the bottom shows the test results for 'EnvironmentDbConfigTest'. The test failed with the message: 'Could not detect default configuration classes for test class [EnvironmentDbConfigTest]. The following 1 profile is active: 'prod'. Bootstrapping Spring Data JPA repository:'. The test output also shows a Spring Boot banner and logs indicating the active profile is 'prod'.

```
import java.sql.ResultSet;
import java.sql.SQLException;
import java.time.LocalDate;
import static org.junit.jupiter.api.Assertions.assertEquals;
import static org.junit.jupiter.api.Assertions.assertNotNull;
@SpringBootTest
class EnvironmentDbConfigTest {

    @Autowired
    EnvironmentDbConfig environmentDbConfig;

    @Test
    void shouldConfigurationNotNull_When_SpringContextWork() {
        assertNotNull(environmentDbConfig);
    }

    @Test
    void shouldEnvironmentHasValue_When_SpringContextWork() {
        final String dbDriverExpected = "org.h2.Driver";
        final String dbUrlExpected = "jdbc:h2:mem:test";
        final String dbUsernameExpected = "SA";
        final String dbPasswordExpected = "";
        assertEquals(dbDriverExpected, environmentDbConfig.getDbDriver());
        assertEquals(dbUrlExpected, environmentDbConfig.getDbUrl());
    }
}
```

Run EnvironmentDbConfigTest (sv.edu.ldb.config) 512ms

1 test failed, 3 passed 4 tests total, 512ms

18:39:26.988 [main] INFO org.springframework.test.context.support.AnnotationConfigContextLoaderUtils -- Could not detect default configuration classes for test class [EnvironmentDbConfigTest]. The following 1 profile is active: 'prod'. Bootstrapping Spring Data JPA repository:'. Found @SpringBootConfiguration sv.edu.ldb.SpringExtConfigApplication.

18:39:27.127 [main] INFO org.springframework.boot.test.context.SpringBootTestContextBootstrapper -- Found @SpringBootConfiguration sv.edu.ldb.SpringExtConfigApplication.

2025-07-15T18:39:27.575-06:00 INFO 1812 --- [spring-ext-config] [main] s.e.u.c.EnvironmentDbConfigTest : Starting EnvironmentDbConfigTest using Java 17.0.15

2025-07-15T18:39:27.577-06:00 INFO 1812 --- [spring-ext-config] [main] s.e.u.c.EnvironmentDbConfigTest : The following 1 profile is active: 'prod'

2025-07-15T18:39:27.910-06:00 INFO 1812 --- [spring-ext-config] [main] s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repository:'. Found @SpringBootConfiguration sv.edu.ldb.SpringExtConfigApplication.

2025-07-15T18:39:27.926-06:00 INFO 1812 --- [spring-ext-config] [main] s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning:'. Found 1 repository bean(s).

## PRUEBA FINAL PASO 23:

The screenshot shows the same IDE as before, but the test 'EnvironmentDbConfigTest' has passed. The 'Run' view at the bottom shows the test results: '4 tests passed, 1 ignored 5 tests total, 516ms'. The test output shows the Spring Boot banner and logs indicating the active profile is 'prod'. The test code is the same as in the previous screenshot, but the configuration is now correctly loaded.

```
package sv.edu.ldb.configuration;
import org.junit.jupiter.api.Test;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.test.context.SpringBootTest;
import org.springframework.test.context.junit.jupiter.EnabledIf;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.time.LocalDate;
import static org.junit.jupiter.api.Assertions.assertEquals;
import static org.junit.jupiter.api.Assertions.assertNotNull;
@SpringBootTest
class EnvironmentDbConfigTest {

    @Autowired
    EnvironmentDbConfig environmentDbConfig;

    @Test
    void shouldConfigurationNotNull_When_SpringContextWork() {
        assertNotNull(environmentDbConfig);
    }

    @Test
    @EnabledIf(expression = "#{environment['spring.profiles.active']} == 'dev'",
        loadContext = true)
    void shouldEnvironmentHasValue_When_SpringDevContextWork() {
        final String dbDriverExpected = "org.h2.Driver";
        final String dbUrlExpected = "jdbc:h2:mem:dev";
        final String dbUsernameExpected = "SA";
        final String dbPasswordExpected = "";
        assertEquals(dbDriverExpected, environmentDbConfig.getDbDriver());
        assertEquals(dbUrlExpected, environmentDbConfig.getDbUrl());
    }
}
```

Run EnvironmentDbConfigTest (sv.edu.ldb.config) 516ms

4 tests passed, 1 ignored 5 tests total, 516ms

18:41:37.774 [main] INFO org.springframework.test.context.support.AnnotationConfigContextLoaderUtils -- Could not detect default configuration classes for test class [EnvironmentDbConfigTest]. The following 1 profile is active: 'prod'. Bootstrapping Spring Data JPA repository:'. Found @SpringBootConfiguration sv.edu.ldb.SpringExtConfigApplication.

18:41:37.917 [main] INFO org.springframework.boot.test.context.SpringBootTestContextBootstrapper -- Found @SpringBootConfiguration sv.edu.ldb.SpringExtConfigApplication.

2025-07-15T18:41:37.917-06:00 INFO 1812 --- [spring-ext-config] [main] s.e.u.c.EnvironmentDbConfigTest : Starting EnvironmentDbConfigTest using Java 17.0.15

2025-07-15T18:41:37.919-06:00 INFO 1812 --- [spring-ext-config] [main] s.e.u.c.EnvironmentDbConfigTest : The following 1 profile is active: 'prod'

2025-07-15T18:41:38.252-06:00 INFO 1812 --- [spring-ext-config] [main] s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repository:'. Found @SpringBootConfiguration sv.edu.ldb.SpringExtConfigApplication.

2025-07-15T18:41:38.268-06:00 INFO 1812 --- [spring-ext-config] [main] s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning:'. Found 1 repository bean(s).