

## **LAPORAN AKHIR PRAKTIKUM**

Mata Praktikum : Rekayasa Perangkat Lunak 2  
Kelas : 4IA18  
Praktikum ke- : 5  
Tanggal : 15 November 2025  
Materi : Spring Framework  
NPM : 50422215  
Nama : Andi Purnama  
Ketua Asisten : Haikal Abizar  
Paraf Asisten : -  
Nama Asisten : 1.  
2.  
3.  
Jumlah Lembar : 5 Lembar



**LABORATORIUM INFORMATIKA**  
**UNIVERSITAS GUNADARMA**

**2025**

**1. Jelaskan apa itu Spring Boot dan bagaimana perbedaannya dengan kode pada pertemuan sebelumnya. Apa keuntungan utama yang ditawarkan Spring Boot bagi pengembang aplikasi?**

**Jawab:**

Spring Boot adalah sebuah framework berbasis Spring yang dirancang untuk menyederhanakan proses pembuatan aplikasi Java dengan konfigurasi yang minimal. Jika pada pertemuan sebelumnya Anda menggunakan pendekatan manual dalam mengatur konfigurasi, membuat objek, dan mengelola struktur aplikasi, maka Spring Boot menawarkan pendekatan yang jauh lebih otomatis dan terstruktur. Spring Boot menyediakan auto-configuration, starter dependencies, dan embedded server sehingga aplikasi dapat langsung dijalankan tanpa perlu melakukan konfigurasi web server atau pengaturan Spring secara mendetail. Hal ini sangat berbeda dengan pendekatan sebelumnya yang biasanya mengharuskan pengembang menuliskan konfigurasi XML, membuat bean secara manual, serta mengatur koneksi dan struktur aplikasi secara eksplisit.

Keuntungan utama yang ditawarkan Spring Boot bagi pengembang adalah kemudahan dan kecepatan dalam membangun aplikasi. Dengan fitur auto-configuration, banyak komponen penting seperti koneksi database, JPA, REST controller, dan server web dikonfigurasi otomatis oleh Spring berdasarkan dependency yang digunakan. Selain itu, Spring Boot juga menyediakan starter dependencies yang mempermudah pengelolaan library dan menurunkan risiko konflik versi. Pengembang dapat langsung fokus pada logika bisnis tanpa terbebani konfigurasi yang kompleks. Fitur embedded server seperti Tomcat yang sudah terintegrasi memungkinkan aplikasi dijalankan hanya dengan perintah `mvn spring-boot:run` atau menjalankan file JAR saja, tanpa perlu instal server terpisah. Secara keseluruhan, Spring Boot membuat proses pengembangan lebih cepat, efisien, dan mudah dipelihara dibandingkan metode tradisional di pertemuan sebelumnya.

## **2. Jelaskan kode dan langkah-langkah program yang telah dibuat!**

**Jawab:**

### **1. File pom.xml**

Pada file pom.xml, struktur dependensi yang Anda gunakan sudah sesuai untuk aplikasi Spring Boot yang memerlukan JPA, MySQL, dan komponen web. Spring Boot Starter Parent versi 3.3.3 sudah tepat untuk mengelola versi seluruh dependency agar kompatibel dengan Spring Boot modern. Namun terdapat satu hal yang perlu diperhatikan, yaitu penggunaan driver MySQL masih menggunakan versi lama (mysql-connector-java) yang sebaiknya diganti menjadi mysql-connector-j tanpa menuliskan versi secara manual. Dengan menghapus penentuan versi secara eksplisit, Anda membiarkan Spring Boot mengatur versi driver yang paling kompatibel. Perubahan ini membuat dependency lebih stabil dan meminimalkan potensi error ketika aplikasi melakukan koneksi ke database.

### **2. File application.properties**

Konfigurasi pada application.properties sudah benar untuk menghubungkan aplikasi Spring Boot ke database MySQL. Anda sudah menggunakan URL JDBC yang sesuai dan menambahkan parameter seperti useSSL=false dan serverTimezone=UTC yang memang dianjurkan untuk stabilitas koneksi. Selain itu, konfigurasi Hibernate seperti spring.jpa.hibernate.ddl-auto=update memungkinkan Hibernate membuat atau memperbarui tabel secara otomatis berdasarkan entity yang Anda definisikan. Kode ini cocok untuk tahap pengembangan, namun perlu diingat bahwa pada tahap produksi sebaiknya menggunakan validate untuk menghindari perubahan struktur tabel yang tidak disengaja.

### **3. File Pertemuan5SpringBootApplication.java**

Pada class utama ini, Anda telah menggunakan anotasi @SpringBootApplication dan mengimplementasikan CommandLineRunner untuk menjalankan aplikasi berbasis menu console. Cara ini sudah benar karena Spring Boot menyuntikkan (inject) MahasiswaController secara otomatis melalui mekanisme dependency injection saat aplikasi berjalan. Namun hal yang perlu diperhatikan adalah konsistensi penamaan class dan file. Pastikan nama file dan nama class benar-benar

sama dengan konfigurasi di proyek Anda agar Spring Boot dapat melakukan scanning terhadap package aplikasi dengan baik dan mencegah error saat proses startup.

#### **4. File MahasiswaController.java**

Di dalam file controller ini, logika aplikasi console Anda ditulis dengan cukup baik. Controller ini berfungsi untuk menampilkan menu, menambah mahasiswa, menampilkan seluruh data, dan memeriksa koneksi database. Spring Boot secara otomatis menyuntikkan repository ke dalam controller menggunakan anotasi `@Autowired`. Cara ini sesuai dengan konsep Dependency Injection yang membuat controller tidak perlu membuat objek repository secara manual. Meski demikian, ada hal penting yang perlu diperbaiki, yaitu penggunaan Scanner dalam loop menu. Meskipun tidak menyebabkan error besar, penggunaan scanner diisi ulang di tempat tertentu bisa menyebabkan input tak terbaca jika tidak ditangani dengan benar. Secara umum, controller sudah berjalan dengan baik sebagai antarmuka console aplikasi ini.

#### **5. File ModelMahasiswa.java**

File entity ini sudah sesuai dengan aturan JPA karena Anda telah menggunakan anotasi seperti `@Entity`, `@Table`, dan `@Id` untuk menandai model sebagai representasi tabel dalam database. Namun ada dua kesalahan penting yang perlu diperbaiki. Pertama, tipe ID pada entity menggunakan int, tetapi repository Anda menggunakan Long, sehingga tidak konsisten dan harus disamakan. Kedua, method `toString()` yang ada di class ini salah mencantumkan nama field, karena semua field ditandai sebagai "nama=", sehingga output menjadi salah dan membingungkan. Anda harus memperbaikinya agar setiap field menampilkan label dan nilai yang benar sesuai dengan atribut mahasiswa seperti npm, nama, semester, dan IPK.

#### **6. File MahasiswaRepository.java**

Repository ini adalah interface yang memperluas `JpaRepository`, yang memungkinkan Anda mengakses database tanpa harus menulis query SQL secara manual. Struktur repository sudah benar, namun Anda menggunakan tipe ID Long sementara entity menggunakan int. Hal ini berpotensi menyebabkan error saat

operasi JPA dijalankan karena tipe ID harus konsisten antara entity dan repository. Anda perlu menggantinya menjadi Integer agar sesuai dengan deklarasi ID pada ModelMahasiswa. Setelah diperbaiki, repository akan bekerja optimal dan Spring Boot dapat mengelola entity mahasiswa dengan baik.

