

Spring Boot Web App Package Structure

1. Model (Entity Layer)

- Represents the business/domain objects.
- Typically mapped to database tables using JPA/Hibernate.
- Contains fields, relationships, and validation rules.

Common Annotations:

- From `javax.persistence` / `jakarta.persistence`:
 - `@Entity` → Marks a class as a JPA entity.
 - `@Table(name="...")` → Maps entity to a table.
 - `@Id` → Primary key.
 - `@GeneratedValue(strategy=...)` → Auto-generate IDs.
 - `@Column(name="...")` → Column mapping.
 - `@OneToMany`, `@ManyToOne`, `@OneToOne`, `@ManyToMany` → Relationships.
 - `@JoinColumn(name="...")` → Foreign key mapping.
 - From `jakarta.validation`:
 - `@NotNull`, `@Size`, `@Email`, `@Pattern`, `@Past`, etc. → Bean validation annotations.
-

2. Controller (Web Layer)

- Handles HTTP requests and responses.
- Defines API endpoints (`/api/...`).
- Uses the Service layer to handle business logic.

Common Annotations:

- From `org.springframework.web.bind.annotation`:
 - `@RestController` → Combines `@Controller` + `@ResponseBody`.
 - `@Controller` → For MVC views (Thymeleaf/JSP).
 - `@RequestMapping("/path")` → Base mapping for endpoints.
 - `@GetMapping`, `@PostMapping`, `@PutMapping`, `@DeleteMapping`, `@PatchMapping` → HTTP method-specific mappings.
 - `@PathVariable` → Extracts values from URL.
 - `@RequestParam` → Extracts query parameters.
 - `@RequestBody` → Maps JSON request to Java object.
 - `@ResponseBody` → Returns JSON/XML response.
 - `@CrossOrigin` → Enables CORS.
- Exception handling:
 - `@ExceptionHandler`, `@ControllerAdvice`.

3. Repository (Data Access Layer)

- Directly interacts with the database.
- Extends Spring Data JPA interfaces (e.g., `JpaRepository`).

Common Annotations:

- From `org.springframework.stereotype`:
 - `@Repository` → Marks as DAO component, enables exception translation.
 - From `org.springframework.data.jpa.repository`:
 - `@Query("JPQL/SQL")` → Custom queries.
 - `@Modifying` → Marks update/delete queries.
 - `@Transactional` → Manages transactions.
-

4. DTO (Data Transfer Object)

- Used to transfer data between layers (esp. Controller ↔ Service).
- Helps decouple API contract from entity model.
- Often used with MapStruct or manual mappers.

Common Annotations:

- Usually plain POJOs with Lombok for boilerplate:
 - `@Data` → Generates getters, setters, equals, hashCode, toString.
 - `@Getter`, `@Setter`, `@AllArgsConstructor`, `@NoArgsConstructor`, `@Builder`.
 - For JSON serialization:
 - `@JsonProperty("name")` → Custom JSON property name.
 - `@JsonIgnore` → Exclude field from JSON.
 - `@JsonInclude(JsonInclude.Include.NON_NULL)` → Exclude nulls.
-

5. Service (Business Layer)

- Contains business logic.
- Calls repositories, applies business rules, and returns data to controllers.
- Decouples controllers from repositories.

Common Annotations:

- From `org.springframework.stereotype`:

- @Service → Marks class as service component.
 - From org.springframework.transaction.annotation:
 - @Transactional → Transaction management.
 - Dependency injection:
 - @Autowired (though constructor injection is preferred).
-

6. Config (Configuration Layer)

- Used for application-level configurations, such as:
 - Security setup
 - CORS config
 - Bean definitions
 - External integrations (e.g., Swagger, Kafka, Mail)

✓ Annotations:

- From Spring Core:
 - @Configuration → Defines configuration class.
 - @Bean → Defines a bean managed by Spring IoC container.
 - @Component → Marks as Spring-managed component (generic).
 - @Autowired → Injects dependencies (constructor/setter/field injection).
 - For security:
 - @EnableWebSecurity → Enables Spring Security config.
 - @EnableGlobalMethodSecurity(prePostEnabled = true) → Enables method-level security (e.g., @PreAuthorize).
 - From Spring Boot:
 - @ConfigurationProperties(prefix = "...") → Binds external config (YAML/Properties) to a POJO.
 - @EnableConfigurationProperties → Enables binding.
-

Summary Table of Annotations by Package

Package	Common Annotations
Model (Entity)	@Entity, @Table, @Id, @GeneratedValue, @Column, @OneToMany, @ManyToOne, @JoinColumn, @NotNull, @Size, @Email

Package	Common Annotations
Controller (Web)	@RestController, @Controller, @RequestMapping, @GetMapping, @PostMapping, @PutMapping, @DeleteMapping, @RequestBody, @PathVariable, @RequestParam, @CrossOrigin, @ExceptionHandler, @ControllerAdvice
Repository (DAO)	@Repository, @Query, @Modifying, @Transactional
DTO	@Data, @Getter, @Setter, @Builder, @JsonProperty, @JsonIgnore
Service (Business)	@Service, @Transactional, @Autowired
Config (Configuration)	@Configuration, @Bean, @Component, @Autowired, @EnableWebSecurity, @EnableGlobalMethodSecurity, @ConfigurationProperties, @EnableConfigurationProperties
