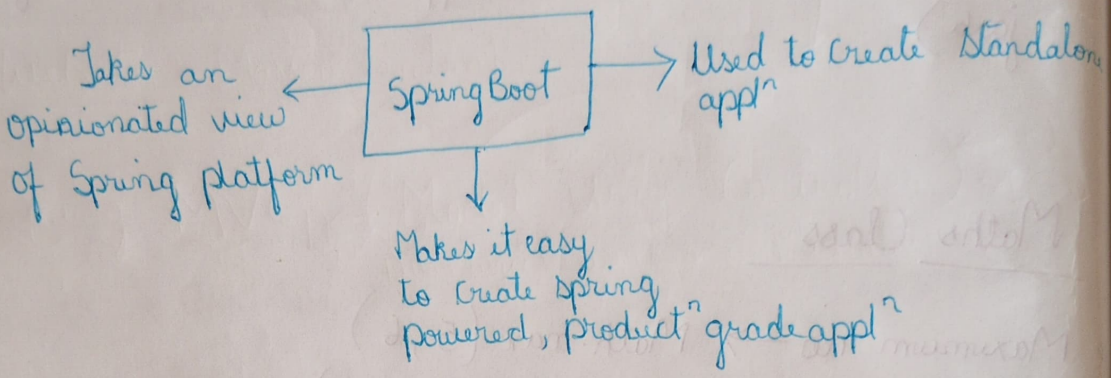
1 Spring Boot

Simplifies spring development by providing sensible defaults and ready to use feature.

* It is used to build web application in Java Programming language.



* Framework used for building Java appl^n.

Features

Spring CLI, Spring initializer, Spring Actuator

2 Spring Vs Spring Boot

(i). Takes time to have Spring appl^n up & running

(ii). Manage lifecycle of Java

(i). Shortest way to run Spring appl^n.

(ii). No need to worry about data source.

* Spring Boot provides the components in a pre-configured way, which is used for developing & running Spring-based appl^n.

3 Components :- It has around 23 lib helps us to build appl^n.

SpringBoot, Spring Data etc.

4 Spring Boot Starter (Dependencies)

- Starter is a pre-defined set of dependencies
- Helps to quickly set up project with minimal config.
- Instead of adding many lib one by one, we add1 starter, and it pulls all required lib.
- Combine the various dependencies arising from a Springboot project into a single dependency.

5 Spring Boot Auto Configurat°

- It is a feature that automatically sets up your appl° based on libs present in project.
- Provides default configurat°.

Why use it?

- Reduce boilerplate (less manual Setup).
- Faster development

6 Spring Boot Actuator

It is a tool that gives ready-made endpoints to monitor & manage the appl° while its running.

Actuator endpoints

/actuator/health → check whether app is running fine
/actuator/info → Basic app info.
/actuator/metrics → CPU, memory, HTTP stats.
/actuator/env → All env variables.

Annotations → Gives instruct°

@Controller
@RequestMapping

Spring Initializr

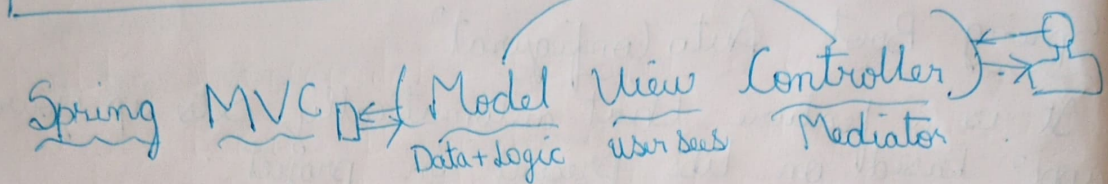
* Website where we can create our base appl^n
* Springboot website.

Dependencies

They are 3rd party lib / frameworks.

Website :- Maven Central

Spring MVC



Standalone application - No need of configuring tomcat Servers.

Maven

* It is build automation tools and simplify it
* It helps to manage dependencies.

Maven Repository helps to download the jar.

Spring IOC → (Inversion of Control)

It is a principle where the control of creating & managing object is given to a container / framework, instead of being handled in code directly.

Eg- (i) without IOC

object obj = new object();

(ii). with Spring IOC

@ Component
public class Classname { }.

#Structure of Spring Boot:

Annotations (Spring Boot)

- Annotations are special makers in code prefixed with @, that provides instructions to the compiler, framework, or runtime env.

- Spring uses annotations to configure beans, inject dependencies, handle HTTP requests & more.

@Controller → makes a class as a web controller.

@RestController → Combines @Controller & @ResponseBody. Used for RESTful web services.

@SpringBootApplication → Combines @Configuration, @EnableAutoConfigurat^n & @ComponentScan.

(i). Indicates main class of a Spring Boot Appl^n.

@ModelAttribute - automatically binds form data to a java object.

Dependencies Inject^n & Configurat^n

@Autowired: Automatically inject dependencies where needed.

@Configurat^n: Define a class as a source of bean definit^n.

Web & Rest

@RequestMapping: Maps HTTP requests to handle method / class.

@GetMapping @PostMapping @PutMapping @DeleteMapping @PatchMapping: Shortcut for HTTP methods

@Component → Add object of that class.

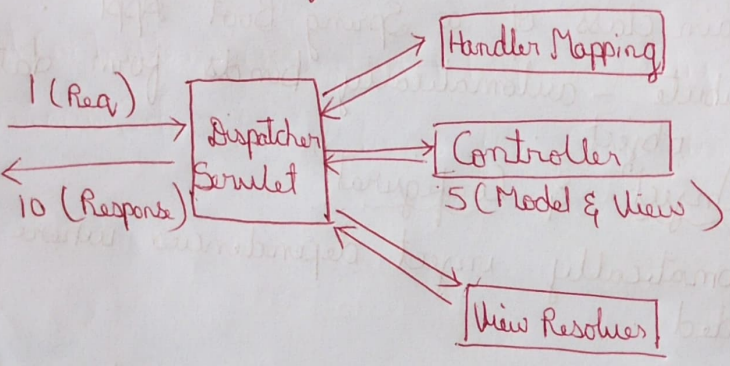
Spring MVC Framework detail

Spring MVC follows the Model-View-Controller architectural design pattern which works around the Front Controller i.e., Dispatcher Servlet. The Dispatcher Servlet handles and dispatches all incoming HTTP requests to appropriate Controller. It uses @Controller & @RequestMapping as default request handlers.

Model - It encapsulates the appl° data

View - Renders the model data and generates HTML output that the clients browser can interpret.

Controller - process user request & passes them to view for rendering.



Spring MVC frameworks works as follows :-

(i). All incoming req are intercepted by D.S which works as front Controller.

(ii). The DS retrieves entry of handler mapping from Config file & forward req to Controller.

(iii). The Controller process the req & return an obj of ModelAndView.

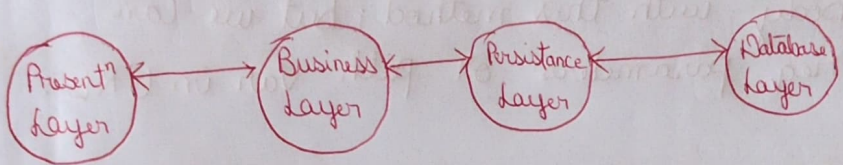
(iv). The D.S checks entry of view resolver in Config file & invokes appropriate view component.

Spring Boot simplifies Spring development by:

(i) AutoConfigurat^n - Spring Boot automatically configures many common beans based on libraries included in project which eliminates the need for a lot of manual config.

(ii). Rapid appl^n Development (RAD) - Spring Boot provides a quick & easy way to start with Spring projects. Can create Springboot appl^n with minimal config and run in minutes.

ARCHITECTURE OF SPRING BOOT



(i). Presentat^n Layer - Topmost layer, consists of REST Controllers that handles HTTP request (GET, POST, PUT, DELETE). It performs authenticat^n, validat^n & JSON desertilizat^n (Conversion of JSON to Java obj). After processing req. it forwards to business layer.

(ii). Business Layer - It is responsible for implementing appl^n core logic. Consists of service classes that:

(a). process & validate data (b). Handle authent^n & authoriz^n

(c). Apply transact^n manag^n using @Transact^{n}l

(d). Interact with Persistance layer to store or retrieve data.

(iii). Persistence Layer - It manage database transact^n & storage logic. It consist of repository class using Spring Data JPA, Hibernate for data access. Responsible for:-

(a). Mapping data obj to database recording

(b). Managing CRUD.

(c). Supp^n Relational & NoSQL database

(iv). Database layer - It contains actual DB where appl" data is stored.

Methods of HTTP

(i). GET - Reads existing data
(ii) PUT - Updates " "
(iii). POST - Creates new data
(iv). DELETE - Deletes data.

GET - Default req method for HTTP. We dont have any request body with this method; but we can define multiple req parameters or path var in URL.

~~RESTful~~

@ Rest Controller - Controller + Response Body.
@ Autowired - Annotations for Connection classes.

ORM.

Object Relational Mapping.

* No need to insert SQL Query by using ORM.

Class Name = Table name
Var = Columns.
Obj = Rows.
} — Mapping

ORM tool - Hibernate, Eclipse, JPA.

JPA - Java Persistance API.

* Using JPA Storing Data in DB becomes easy

H2 → Provides a fast in-memory database that supports JDBC API & R2DBC access, with a small (2mb) footprint.

- It supports embedded & server modes as well as a browser based console appl^n.