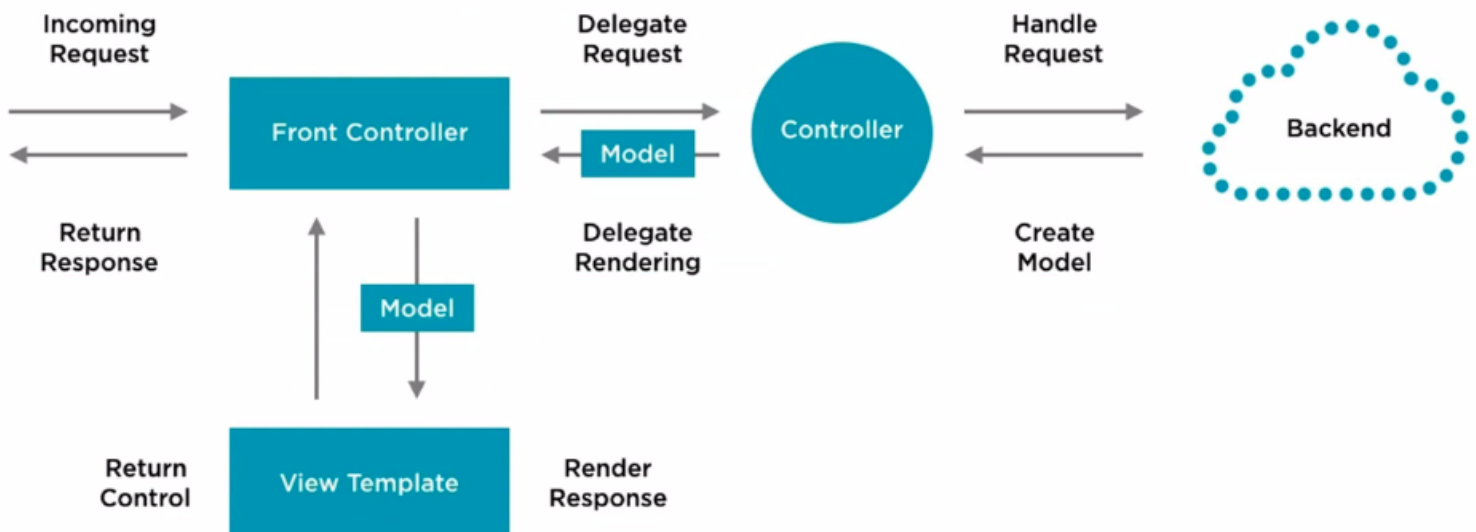


IoC

```
@SpringBootApplication
@EnableAutoConfiguration
@ComponentScan(
    excludeFilters = {@Filter(
        type = FilterType.CUSTOM,
        classes = {TypeExcludeFilter.class}
    ), @Filter(
        type = FilterType.CUSTOM,
        classes = {AutoConfigurationExcludeFilter.class}
    )}
)

public @interface SpringBootApplication {
```

Request / Response Lifecycle



MODEL-VIEW-CONTROLLER

```
@Component
public @interface Controller {

    @AliasFor(
        annotation = Component.class
    )
    String value() default "";
}
```

```
*****
APPLICATION FAILED TO START
*****

Description:

Web server failed to start. Port 15000 was already in use.
```

JAVA EXPRESS – SPRING BOOT MICROSERVICES ONLINE TRAINING

www.youtube.com/c/javaexpress

Contact Number : +91 8555929285 (New Batch Going to Start in this week – 8.30 AM/8.00 PM)

1. Spring Boot Internals

- 1) What is Spring Framework ?
- 2) Spring vs Spring Boot ?
- 3) Advantages of Spring Boot ?
- 4) Environment Setup
- 5) Spring Boot Application Creation
 - a) Spring Initialer
 - b) STS (Spring Tool Suite)
- 6) Spring Boot Starters
- 7) Main Class in SpringBoot
- 8) @SpringBootApplication
- 9) Autoconfiguration in SB
- 10) Spring Boot Internal Working
- 11) Spring IOC Container
- 12) Setter Injection
- 13) Constructor Injection
- 14) Field Injection
- 15) Autowiring & Qualifier
- 16) Component Scanning
- 17) Spring Boot Customer Banner
- 18) Runners in SpringBoot
- 19) Application Runner
- 20) Command Liner Runner
- 21) Stereotype Annotations

2. Spring Data JPA

- 1) What is Spring Data ?
- 2) Spring ORM vs Data JPA
- 3) What is Persistence Layer ?
- 4) ORM Basics
- 5) JPA Annotations
- 6) CRUD Repository
- 7) JPA Repository
- 8) Repository Methods
- 9) @Query Annotation
- 10) Native SQL Queries using @Query
- 11) HQL Queries using @Query
- 12) Positional & Named Parameters
- 13) Read Queries using Properties file
- 14) Find By Methods in Spring Data JPA
- 15) Pagination using JPA
- 16) Sorting using JPA
- 17) Custom Generators
 - IDENTITY
 - AUTO
 - SEQUENCE
- 18) Schema Generation
 - create
 - update
 - validate
 - create-drop

3. Spring MVC

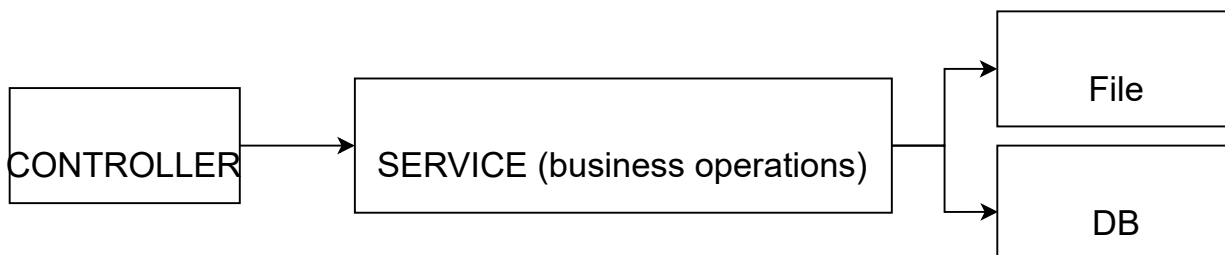
- 1) What is Web Application ?
- 2) MVC Architecture ?
- 3) Front Controller Design Pattern
- 4) Dispatcher Servlet
- 5) Handler Mapping
- 6) View Resolver
- 7) Model, Model and View
- 8) @ModelAttribute
- 9) @Controller Annotation
- 10) @ResponseBody
- 11) Send Controller to UI
- 12) Send UI to Controller
- 13) @RequestParam
- 14) @PathVariable
- 15) Embedded Containers
- 16) Embedded Database (H2)
- 17) Spring Boot Actuator
- 18) Exception Handling in Spring Boot
- 19) Thymeleaf Integration with Spring Boot
- 20) Email Sending in Spring Boot

4. Restful Services

- 1) What is Distributed Application ?
- 2) Distributed Technologies
- 3) Evolution of Web services
- 4) SOAP Based Web Services Architecture
- 5) What is WSDL File in SOAP ?
- 6) REST Architecture Principles
- 7) Rest Resource & Rest Client
- 8) HTTP Methods
- 9) Http Status codes & Messages
- 10) @Rest Controller
- 11) Difference between @RestController vs @Controller
- 12) Spring Annotations
- 13) Swagger Documentation
- 14) OPEN API Integration
- 15) Spring Boot Profiles
- 16) Spring Boot Logging
- 17) Rest Template, Web Client
- 18) Reactive Programming
- 19) Spring Boot Security
 - Form Based Auth
 - Http Basic Auth
 - Role Bases Auth
- 20) HATOES
- 21) Rest Client Using Node JS & Java
- 22) Running SB in Docker Container

5. Micro Services

- 1) Monolithic Architecture
- 2) Drawbacks of Monolithic Architecture
- 3) Introduction to Microservices
- 4) Challenges in Microservice's
- 5) Advantages of using Microservices
- 6) Microservices Architecture
- 7) What is Service Registry ?
- 8) Eureka Server
- 9) Eureka Client
- 10) API Gateway (Zuul Proxy)
- 11) Feign Client (Interservice Communication)
- 12) Spring Boot Actuators
- 13) Spring Boot Admin Server & Admin Client
- 14) Read Properties from YML
- 15) Centralized Configuration
- 16) Spring Boot Config Server
- 17) Spring Boot Config Client
- 18) Distributed Logging using Sleuth & Zipkin Server
- 19) Load Balancing using Ribbon Client
- 20) Hystrix (Circuit Breaker)
- 21) OAuth 2.0 SSO – GitHub Integration
- 22) Connecting Multiple DB's
- 23) Apache Kafka Integration



LESSON #03 - 04.11.2021

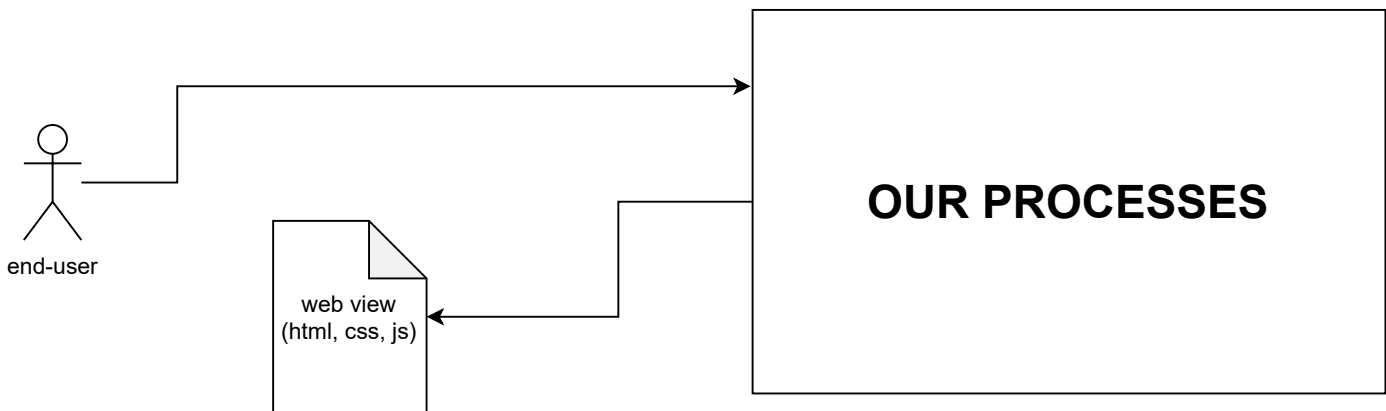
Controller

Service
Business Process

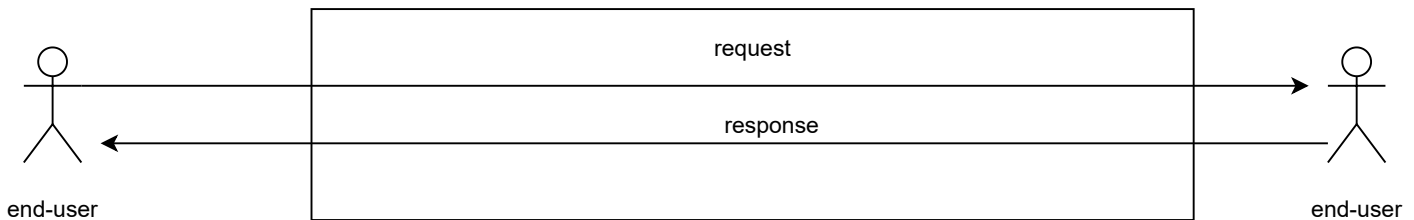
DAO
DAL

Text

LESSON #04 - 11.11.2021



DATA TRANSFER - WEB SERVICES



```
// http://localhost:8085/api/v1/persons
```

```
▼[[{"id": 1, firstName: "Elvin", lastName: "Taghizade", salary: 100},...]  
▼0: {id: 1, firstName: "Elvin", lastName: "Taghizade", salary: 100}  
  firstName: "Elvin"  
  id: 1  
  lastName: "Taghizade"  
  salary: 100  
▼1: {id: 2, firstName: "Pervin", lastName: "Etibarli", salary: 200}  
  firstName: "Pervin"  
  id: 2  
  lastName: "Etibarli"  
  salary: 200  
▼2: {id: 3, firstName: "Elvin #2", lastName: "Taghizade #2", salary: 300}  
  firstName: "Elvin #2"  
  id: 3  
  lastName: "Taghizade #2"  
  salary: 300  
▼3: {id: 4, firstName: "Elvin #3", lastName: "Taghizade #3", salary: 400}  
  firstName: "Elvin #3"  
  id: 4  
  lastName: "Taghizade #3"  
  salary: 400
```

```
▼<List>  
  <item>  
    <id></id>  
    <firstName>Elvin</firstName>  
    <lastName>Taghizade</lastName>  
    <salary>100</salary>  
  </item>  
  <item>  
    <id></id>  
    <firstName>Pervin</firstName>  
    <lastName>Etibarli</lastName>  
    <salary>200</salary>  
  </item>  
  <item>  
    <id></id>  
    <firstName>Elvin #2</firstName>  
    <lastName>Taghizade #2</lastName>  
    <salary>300</salary>  
  </item>  
  <item>  
    <id></id>  
    <firstName>Elvin #3</firstName>  
    <lastName>Taghizade #3</lastName>  
    <salary>400</salary>  
  </item>  
</List>
```