

Configuring a Spring MVC Application with Spring Boot



Kesha Williams

SOFTWARE ENGINEERING MANAGER & PROFESSOR

@KeshaWillz www.kesha.tech



Overview



Configure Spring MVC application

MVC design pattern

Useful annotations

Deploy application to Tomcat



```
<dependency>  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-starter-web</artifactId>  
</dependency>
```

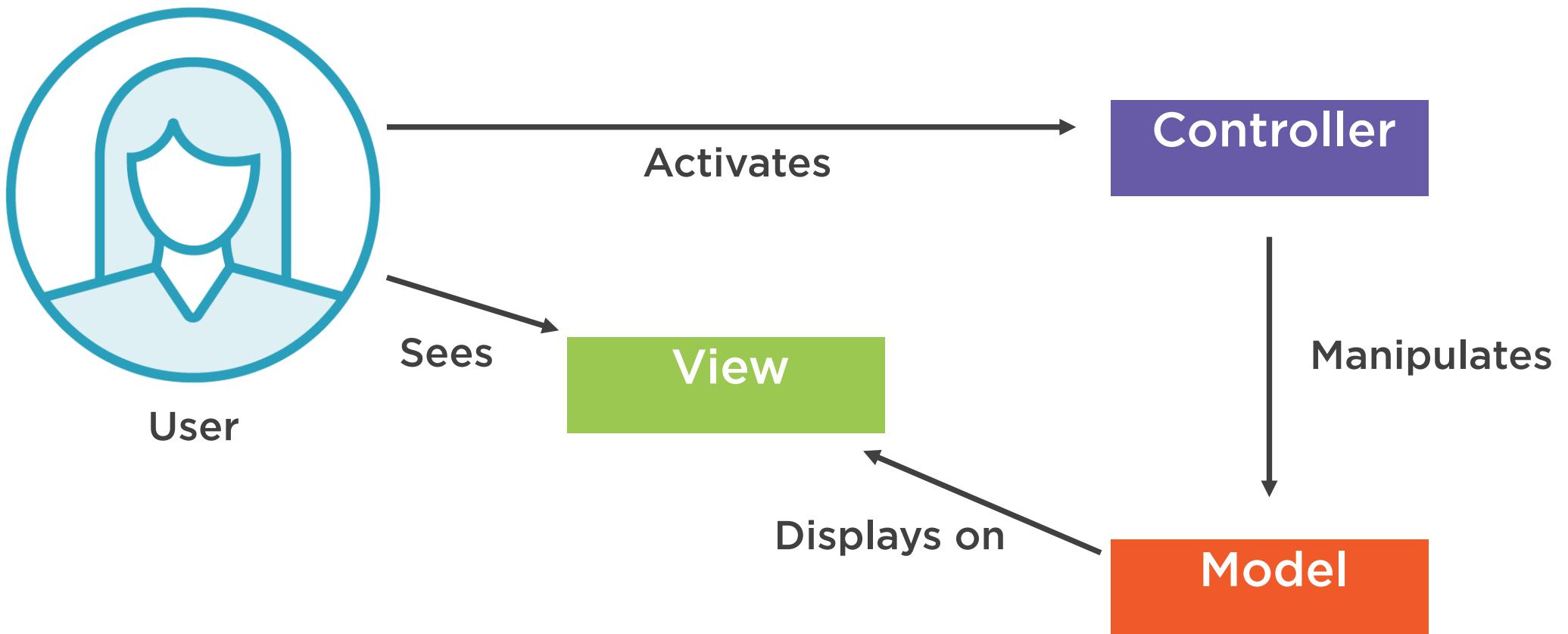
Spring MVC Dependency
Dispatcher servlet, error page, servlet container, jars



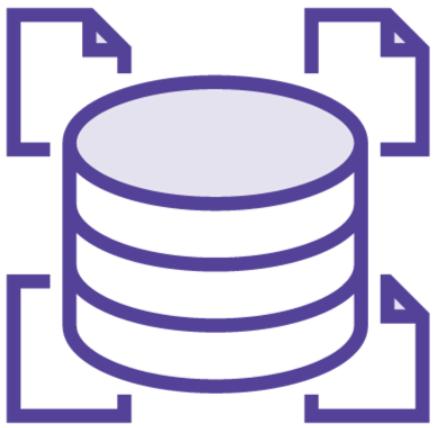
MVC Design Pattern



MVC Design Pattern

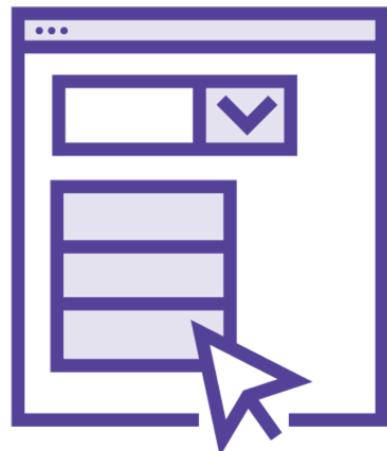


MVC



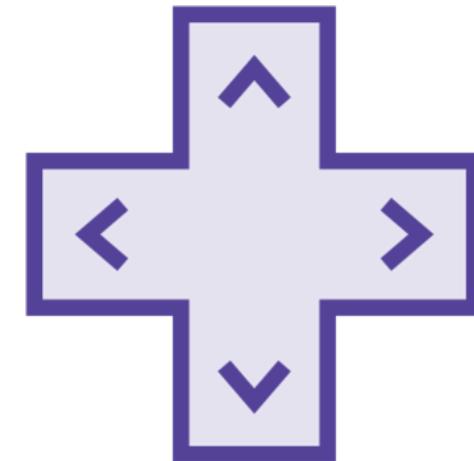
Model

Representation of
data in a system



View

Responsible for
displaying data

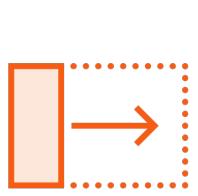


Controller

Directing incoming
user requests



ThymeLeaf



Fragments



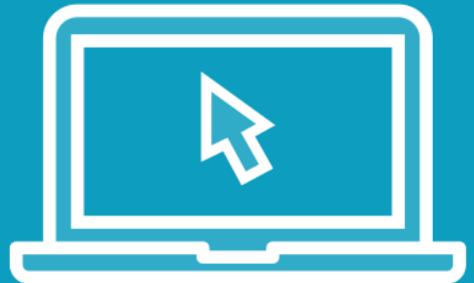
Repeatable chunks of code



Components reused across pages



Demo



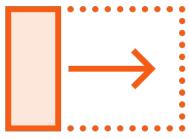
**Model View Controller (MVC) in
TrackZilla**



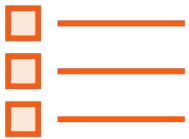
Spring Boot Packaging and Deployment



Packaging and Deployment



Packaged as traditional web application in a .war file



Trackzilla is a standalone application



Packaged in an executable JAR file with a main method

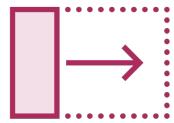


Spring

```
<build>
  <plugins>
    <plugin>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-maven-plugin</artifactId>
    </plugin>
  </plugins>
</build>
```



Spring Boot Maven Plugin



Repackages your .jar and .war files to be executable



Runs Spring Boot application



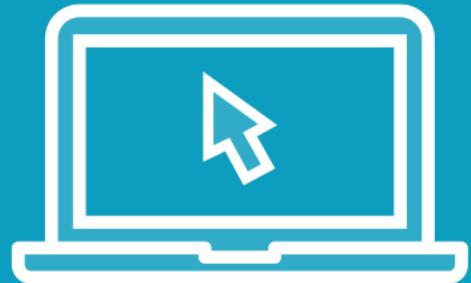
Provides built-in dependency resolver



Manages lifecycle of Spring Boot application



Demo



Application Layers

- TzaController
- Model
- Service
- Repository
- View



Summary



Configure Spring MVC application

Overview of MVC design pattern

Useful annotations

Automatic configuration

Deploy application to Tomcat

