Spring notes

Contents

[1. Spring MVC Project Structure 2](#_Toc85031648)

[2. Standard Project Structure for Spring Boot Projects 3](#_Toc85031649)

[3. Spring Boot non-web application example 3](#_Toc85031650)

[4. IDE for Spring project 3](#_Toc85031651)

[5. Spring maven dependencies 4](#_Toc85031652)

# Spring MVC Project Structure

Spring MVC project structure - <https://www.javaguides.net/2019/01/spring-mvc-project-structure.html>

Spring MVC Project Structure

In a typical Spring MVC web application, the web application packaging structure may look:

── pom.xml

└── src

├── main

│ ├── java

│ │ └── com

│ │ └── companyname

│ │ └── projectname

│ │ ├── domain

│ │ │ └── MyDomain.java

│ │ ├── repository

│ │ │ └── MyDomainRepository.java

│ │ ├── service

│ │ │ ├── MyDomainService.java

│ │ │ └── internal

│ │ │ └── MyDomainServiceImpl.java

│ │ └── web

│ │ └── MyDomainController.java

│ ├── resources

│ │ ├── META-INF

│ │ │ └── spring

│ │ │ ├── applicationContext.xml

│ │ │ └── database.properties

│ │ ├── logback-access.xml

│ │ └── logback.xml

│ └── webapp

│ ├── WEB-INF

│ │ ├── classes

│ │ ├── i18n

│ │ ├── layouts

│ │ ├── spring

│ │ │ └── webmvc-config.xml

│ │ ├── views

│ │ │ ├── myDomain

│ │ │ │ ├── create.jsp

│ │ │ │ ├── list.jsp

│ │ │ │ ├── show.jsp

│ │ │ │ └── update.jsp

│ │ │ ├── dataAccessFailure.jsp

│ │ │ ├── index.jsp

│ │ │ ├── resourceNotFound.jsp

│ │ │ ├── uncaughtException.jsp

│ │ │ └── views.xml

│ │ └── web.xml

│ ├── images

│ └── styles

├── site

│ ├── apt

│ ├── fml

│ ├── site.xml

│ └── xdoc

└── test

├── java

│ └── com

│ └── companyname

│ └── projectname

│ └── service

│ └── MyDomainServiceTests.java

└── resources

├── com

│ └── companyname

│ └── projectname

│ └── service

│ └── MyDomainServiceTests-context.xml

└── logback-test.xml

**Production**

* **src/main/java** – Java Source code packages and classes
* **src/main/resources** – NON-Java Resources, such as property files and Spring configuration
* **src/main/webapp** - deployment descriptor(web.xml), WEB-INF, JSP views under WEB-INF folder, static resources.

**Test**

* **src/test/java** – Test Source code packages and classes
* **src/test/resources** – NON-Java Resources, such as property files and Spring configuration

**Spring MVC - placing your JSP files in a directory under the 'WEB-INF' directory**

As a best practice, we strongly encourage placing your JSP files in a directory under the 'WEB-INF' directory so there can be no direct access by clients.

# Standard Project Structure for Spring Boot Projects

Standard Project Structure for Spring Boot Projects- <https://www.javaguides.net/2019/01/standard-project-structure-for-spring-boot-projects.html>.

# Spring Boot non-web application example

Read - <https://mkyong.com/spring-boot/spring-boot-non-web-application-example/>.

# IDE for Spring project

Install - <https://spring.io/tools>, Read - <https://github.com/spring-projects/sts4/wiki>.

# Spring maven dependencies

Read - <https://www.baeldung.com/spring-with-maven>.

# Overriding beans

<https://www.codelooru.com/2017/02/overriding-spring-beans-with-aliases.html>