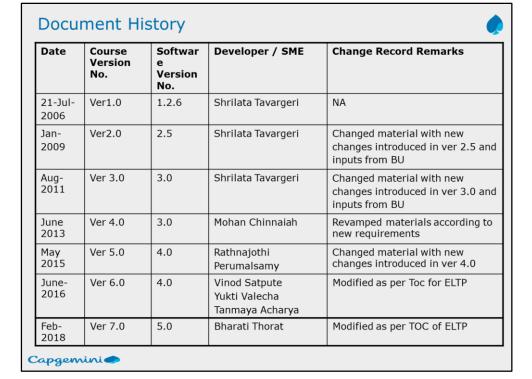
Instructor Notes:



©2016 Capgemini. All rights reserved.

The information contained in this document is proprietary and confidential. For Capgemini only.

Instructor Notes:



Keep this as a hidden slide.

Note to co-ordinators: Not to be printed for the class book.

Instructor Notes:

Course Goals and Non Goals



- Course Goals
 - Understand the benefits of using Spring
 - Understand the principles of IoC
 - Understand the Java Based Configuration
 - Connect business objects to persistent stores using Spring's DAO modules
 - Use the Spring MVC web framework to develop flexible web applications
 - Use the Spring MVC Java Based Configuration to develop flexible web application
- Course Non Goals
 - Design patterns, Spring Integration with different technologies



Instructor Notes:

Pre-requisites

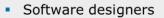


- Core Java , Java 8 features and JDBC
- XML, DBMS/SQL
- Servlets, JSP
- Concepts of MVC, Design patterns

Instructor Notes:

Intended Audience

All Java application developers especially
 Enterprise Java Programmers





Instructor Notes:

Day Wise Schedule



Day 1

- Lesson 1: Introduction to Spring Platform and environment, Spring Boot features
- Lesson 2: Introduction to Spring Framework, IoC

Day 2

- Lesson 3 : Spring Java Base Configuration
- Lesson 4: Spring MVC Framework

Day 3

Lesson 4: Spring MVC Framework

Day 4

Lesson 5 : Spring JPA Integration



Instructor Notes:

• Lesson 6 : Spring Boot

Day 5
• Lesson 7 : Spring RESTful
• Lesson 8 : Exception Handling in Spring RESTful

Instructor Notes:

Table of Contents



Lesson 1: Introduction to Spring Platform and Environment

- 1.1 Introduction to Spring Platform and environment
- 1.2 Spring 5 features
- 1.3 Spring Projects At a Glance
- 1.4 Spring IO Platform
 - 1.4.1 Spring Framework
 - 1.4.2 Spring Boot

Lesson 2: Introduction to Spring Framework, IoC

- 2.1 What is Spring Framework, Benefits of Spring
- 2.2 The Spring architecture
- 2.3 Dependency Injection
- 2.4 IOC Inversion of control, wiring beans
- 2.5 Bean containers, lifecycle of beans in containers, Bean Scopes
- 2.6 Customizing beans with PostProcessors
- 2.7 Annotation-based configuration

Instructor Notes:

Table of Contents



Lesson 3: Spring Java Base Configuration

- 3.1 Spring Application Using Java Base Configuration
- 3.2 Java Configuration Class

 @Configuration

 @Bean
- 3.3 Implementing Bean Lifecycle Callbacks and Bean Scope
- 3.4 Registering Configuration Using AnnotationConfigApplicationContext

Lesson 4: Spring MVC framework

- 4.1 Introduction: DispatcherServlet, Handler mappings, Resolving views
- 4.2 Annotation-based controller configuration



Instructor Notes:

Table of Contents



Lesson 5: Spring JPA Integration

5.1 Spring support for JPA

5.2 Implementing Spring JPA integration

5.3 Spring Data JPA

Lesson 6: Spring Boot

Lesson 7: Spring RESTful

Lesson 8: Exception Handling in Spring RESTful

Instructor Notes:

References



 Spring in Action, Fourth Edition, Manning publications by Craig Walls



 Spring-framework-reference.pdf from SpringSource (this is available in the downloaded Spring software)

Instructor Notes:

Software required



- JDK version 1.8 + with help, Netscape or IE
- MS-Access/Connectivity to Oracle database
- WildFly
- Eclipse Luna with Spring Tool Suite
- Spring 4.0 API with docs

Instructor Notes:

Other Parallel Technology Areas



- EJB 3.0
- PicoContainer
- NanoContainer
- Keel Framework
- Google Guice



- PicoContainer: is an exceptionally small DI (Dependency Injection) container that allows to use DI for your application without introducing any dependencies other than PicoContainer itself
- NanoContainer: is an extension to PicoContainer fro managing trees of individual PicoContainer containers.
- Keel Framework: is more of a metaframework, in that most of its abilities come from other frameworks that are all brought together under one roof.
- Google Guice: focuses purely on DI.