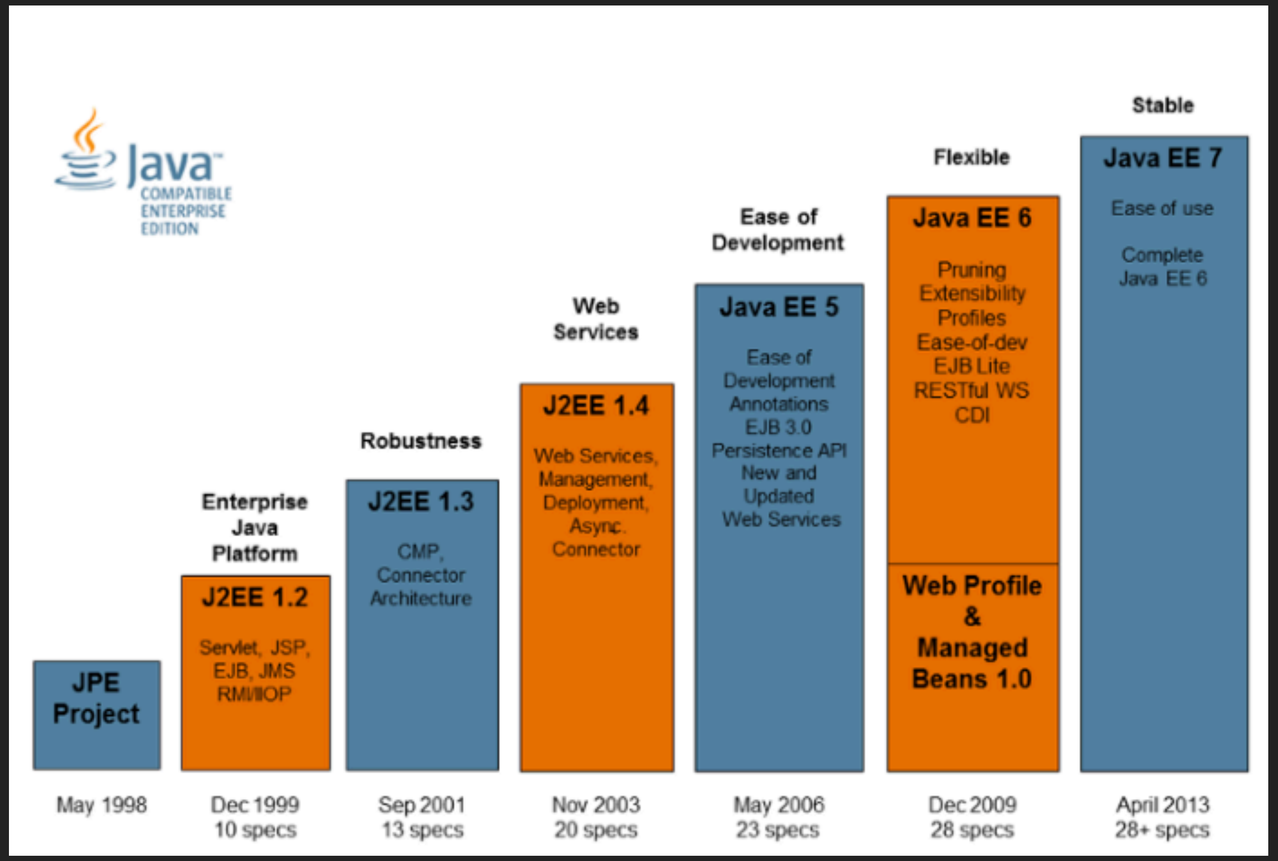
CPAN-252 WEB APPLICATION DEVELOPMENT FOR CPAN DIPLOMA. PROF. ANTON KOVUNOV

SPRING FRAMEWORK

SPRING HISTORY - JEE

- Java EE consists of a set of over 28 specifications and a runtime environment. It is a superset of the Java SE platform.This means that Java EE components can take full advantages of all Java SE APIs.
- This set of APIs build standard component-based multi-tier applications and that deploy in different containers offering a variety of services. Not only is it be used to developer monolithic application
- 1998 saw the release of the first incarnation of Enterprise Java, but soon after, important technologies soon joined the mix, such as Servlets, Messaging, and Enterprise Java Beans.

JEE


Java
COMPATIBLE
ENTERPRISE
EDITION
JPE
Project
May 1998
Enterprise
Java
Platform
J2EE 1.2
Servlet, JSP,
EJB, JMS
RMI/IIOP
Dec 1999
10 specs
Robustness
J2EE 1.3
CMP,
Connector
Architecture
Sep 2001
13 specs
Web
Services
J2EE 1.4
Web Services,
Management,
Deployment,
Async,
Connector
Nov 2003
20 specs
Ease of
Development
Java EE 5
Ease of
Development
Annotations
EJB 3.0
Persistence API
New and
Updated
Web Services
May 2006
23 specs
Flexible
Java EE 6
Pruning
Extensibility
Profiles
Ease-of-dev
EJB Lite
RESTful WS
CDI
Web Profile
&
Managed
Beans 1.0
Dec 2009
28 specs
Stable
Java EE 7
Ease of use
Complete
Java EE 6
April 2013
28+ specs

SPRING FRAMEWORK

- Spring came into being in 2003 as a response to the complexity of the early J2EE specifications. While some consider Java EE and its modern-day successor Jakarta EE to be in competition with Spring, they are in fact complementary. The Spring programming model does not embrace the Jakarta EE platform specification; rather, it integrates with carefully selected individual specifications from the traditional EE umbrella:
- Servlet API (JSR 340)
- WebSocket API (JSR 356)
- Concurrency Utilities (JSR 236)
- JSON Binding API (JSR 367)
- Bean Validation (JSR 303)
- JPA (JSR 338)
- JMS (JSR 914)
- as well as JTA/JCA setups for transaction coordination, if necessary.

SPRING PHILOSOPHY

- Provide choice at every level. Spring lets you defer design decisions as late as possible. For example, you can switch persistence providers through configuration without changing your code. The same is true for many other infrastructure concerns and integration with third-party APIs.
- Accommodate diverse perspectives. Spring embraces flexibility and is not opinionated about how things should be done. It supports a wide range of application needs with different perspectives.
- Maintain strong backward compatibility. Spring's evolution has been carefully managed to force few breaking changes between versions. Spring supports a carefully chosen range of JDK versions and third-party libraries to facilitate maintenance of applications and libraries that depend on Spring.
- Care about API design. The Spring team puts a lot of thought and time into making APIs that are intuitive and that hold up across many versions and many years.
- Set high standards for code quality. The Spring Framework puts a strong emphasis on meaningful, current, and accurate javadoc. It is one of very few projects that can claim clean code structure with no circular dependencies between packages.
-

SPRING DATA REST CUSTOMIZATION

- Spring data REST not only support GET requests, but also POST, PUT and DELETE with default semantics
- We can also customize base url by using spring.data.rest.base-path

```
spring.datasource.name=tekkenreborn
spring.data.rest.base-path=/api
```

- We can also use @RestResource(rel = "tekkenfighters", path = “tekkenfighters") to customize url (instead of table name fighers)

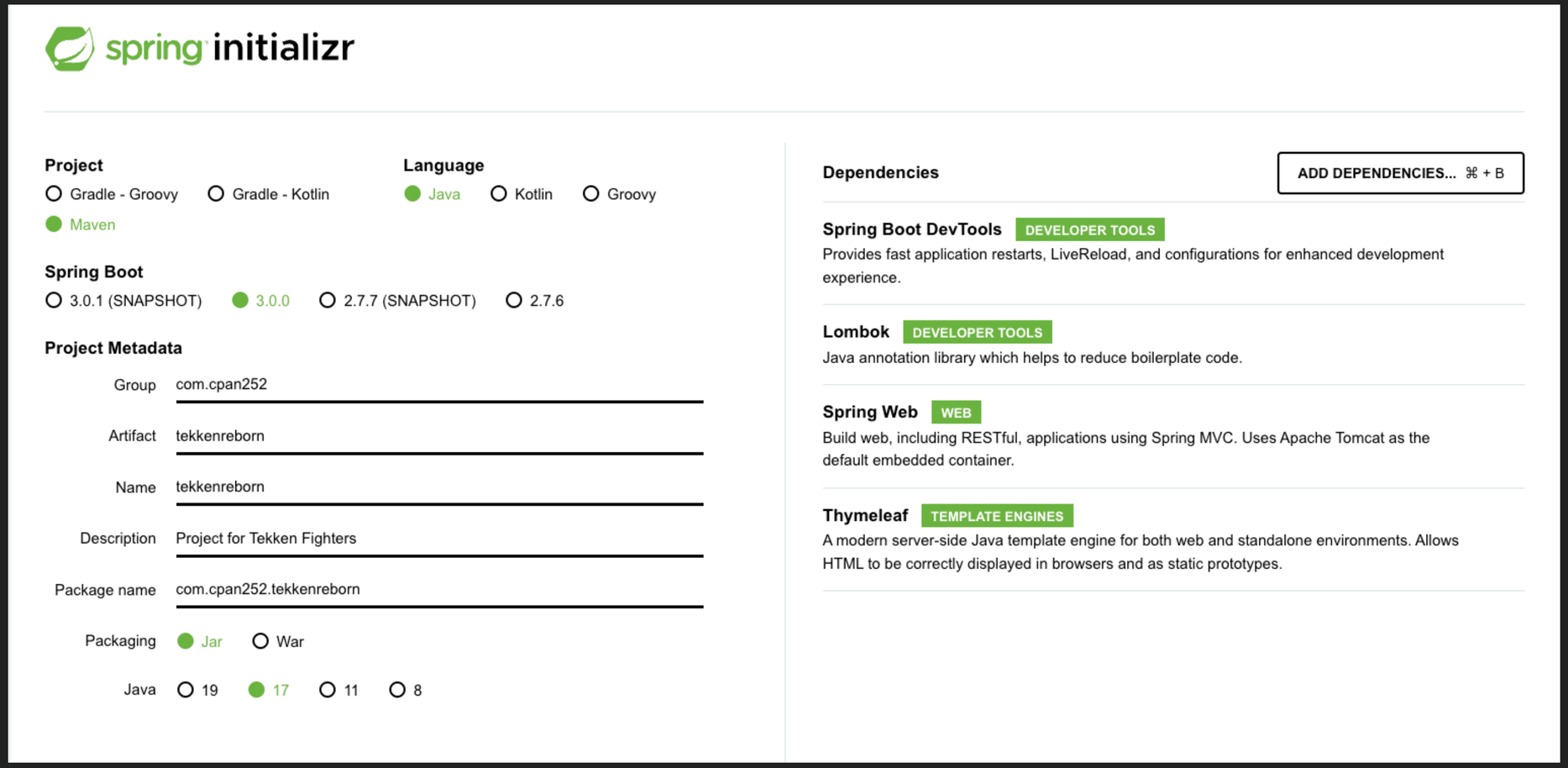
SPRING BOOT OVERVIEW

- Spring Boot makes it easy to create stand-alone, production-grade Spring based Applications that you can "just run".
- We take an opinionated view of the Spring platform and third-party libraries so you can get started with minimum fuss. Most Spring Boot applications need minimal Spring configuration.
-

SPRING INITIALIZR

- One more useful Spring Toolkit is called Spring Initializr, it helps to bootstrap a Spring Boot project with set number of dependencies.
- You will need to go to this website: https://start.spring.io/
- On the next page we going to see selection of dependencies

SPRING INITIALIZR



MAVEN DESCRIPTION

- Maven is a popular open-source build tool developed by the Apache Group to build, publish, and deploy several projects at once for better project management. The tool provides allows developers to build and document the lifecycle framework.
- Maven’s purpose is to provide developers with:
- A comprehensive, maintainable, reusable, and simple model for projects.
- A set of tools and plug-ins that can interact with the declarative model

MAVEN FEATURES

- Maven is loaded with many valuable and useful features, which goes a long way towards explaining its popularity. Here are some of Maven's more noteworthy features:
  - A huge, continuously growing repository of user libraries
  - The ability to set up projects easily, using best practices
  - Dependency management, featuring automatic updating
  - Backwards compatible with previous versions
  - Strong error and integrity reporting
  - Automatic parent versioning
  - Ensures consistent usage across all projects
  - It's extensible, and you can easily write plug-ins using scripting languages or Java.

PROJECT OBJECT MODEL (POM)

- Maven is so useful thanks to the Project Object Model (POM), which is an XML file that has all the information regarding project and configuration details. The POM has the description of the project, details regarding the versioning, and configuration management of the project.
- The XML file is located in the project home directory. When you execute a task, Maven searches for the POM in the current directory.
- Maven is chiefly used for Java-based projects, helping to download dependencies, which refers to the libraries or JAR files. The tool helps get the right JAR files for each project as there may be different versions of separate packages.
- After Maven, downloading dependencies doesn't require visiting the official websites of different software. You can visit mvnrepository to find libraries in different languages. The tool also helps to create the right project structure in struts, servlets, etc., which is essential for execution.

QUESTIONS?

On lab we will continue exploring Spring Boot project, diving deep into dependency injection and annotations, decomposing project structure and describing things that are happening behind the scenes