**TECNOLOGIAS A UTILIZAR EN APLICACIÓN WEB POS\_RESTAURANTE**

**ENTORNO DE DESARROLLO SPRING TOOLS**



### **Primavera consciente**

Diseñada para desarrollar aplicaciones empresariales con Spring Framework y Spring Boot, la nueva generación de Spring Tools proporciona soporte de desarrollo de clase mundial para sus aplicaciones Spring. Nuestras herramientas tienen un profundo conocimiento de Spring incorporado.

### **IDE agnóstico**

Las nuevas Spring Tools se pueden usar en varios entornos de codificación, desde Eclipse como un entorno de desarrollo integrado con todas las funciones hasta Visual Studio Code y Theia como editores de código livianos. Continúe usando su entorno preferido y agregue excelentes herramientas Spring.

### **Reconstruido desde cero**

La nueva generación de Spring Tools está construida en gran medida desde cero, incorporando tecnologías modernas y arquitecturas de herramientas para desarrolladores. Se ejecuta en procesos separados, se construye teniendo en cuenta el rendimiento desde el principio y conoce las últimas tecnologías Spring.

**FRAMEWOK SPRING BOOT 5:**

Spring Framework proporciona un modelo integral de programación y configuración para aplicaciones empresariales modernas basadas en Java, en cualquier tipo de plataforma de implementación.

Un elemento clave de Spring es el soporte de infraestructura a nivel de aplicación: Spring se enfoca en la "plomería" de aplicaciones empresariales para que los equipos puedan enfocarse en la lógica de negocios a nivel de aplicación, sin vínculos innecesarios con entornos de implementación específicos.

**Política de apoyo y migración**

Para obtener información sobre los requisitos mínimos, orientación sobre la actualización de versiones anteriores y políticas de soporte, consulte [la página wiki oficial de Spring Framework](https://github.com/spring-projects/spring-framework/wiki/Spring-Framework-Versions)

Caracteristicas

* [**Tecnologías principales**](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/core.html)**:** inyección de dependencia, eventos, recursos, i18n, validación, enlace de datos, conversión de tipo, SpEL, AOP.
* [**Probando**](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/testing.html)**:** objetos simulados, marco TestContext, Spring MVC prueba, WebTestClient.
* [**Acceso a datos**](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/data-access.html)**:** transacciones, soporte DAO, JDBC, ORM, Marshalling XML.
* [Spring MVC](https://docs.spring.io/spring/docs/current/spring-framework-reference/web.html) y [Spring WebFlux](https://docs.spring.io/spring/docs/current/spring-framework-reference/web-reactive.html) frameworks web.
* [**Integración**](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/integration.html)**:** comunicación remota, JMS, JCA, JMX, correo electrónico, tareas, programación, caché.
* [Idiomas](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/languages.html) : Kotlin, Groovy, lenguajes dinámicos.

**LENGUAJE DE PROGRAMACION JAVA:**

Java Enterprise Edition, Java EE en adelante, es un conjunto de estándares de tecnologías dedicadas al desarrollo de Java del lado del servidor. La plataforma Java EE consta de un conjunto de servicios, API y protocolos que proporcionan la funcionalidad necesaria para desarrollar aplicaciones basadas en web de varios niveles. Es decir, desarrollaremos aplicaciones empresariales distribuidas, con arquitecturas multicapa, escritas en Java y que se ejecutan en un servidor de aplicaciones.

Java es un lenguaje orientado a objetos de alto nivel. Sin duda, es uno de grandes lenguajes de programación de los que se disponen, con una alta demanda en el mercado laboral.

Necesitamos un buen nivel de Java que adquiriremos bajo la plataforma Java SE (Estándar Edition): Sintaxis, características, objetos, etc. Java EE incluye muchos componentes de Java Standard Edition (Java SE). Podéis consultar más información en la página oficial

Captura de pantalla de un celular

Descripción generada automáticamente

A la altura de las ambiciosas expectativas del mercado laboral, en el [curso](http://www.fundesemacademy.es/curso-experto-en-desarrollo-de-aplicaciones-web-con-java-ee-con-empleabilidad-garantizada-) abordamos otras tecnologías necesarias para el desarrollo completo de las diferentes capas de nuestras aplicaciones.

Cuando ya hemos establecido una buena base de Java se aborda el mundo de las Bases de Datos, el modelado de datos, el modelo relacional y como acceder e interrogar al sistema desde Java. Esto nos dejará preparados para abordar conceptos complejos como son las arquitecturas de software con las que ya empezamos a trabajar en la última parte del siguiente módulo “Desarrollo FrontEnd”.

Ahí, basándonos en proyectos, se programará en HTML, CSS y JavaScript como base primordial del desarrollo web en la parte cliente, incluyendo la librería JQuery. Comunicación con la tecnología AJAX. Presentación de datos con XML y JSON.

Se abordan conceptos tan importantes como los patrones de diseño: Modelo Vista Controlador, Modelo Vista Vista-Modelo. Cómo surgen, a qué necesidad responden. Y se empieza a trabajar con ellos desde el tan extendido y demandado framework javascript AngujarJS. Más adelante del curso, también trabajaremos con el framework javascript BackBoneJS.

Y ahora si estamos listos para abordar Java EE. Empezaremos presentando su arquitectura e instalando y configurando un servidor de aplicaciones. Ya desde el inicio, estaremos trabajando con el IDE Netbeans por temas de metodología, pero llegados aquí, se presentarán otros IDE como Eclipse, por ejemplo, y se trabajará también con ellos.

La forma de ir avanzando dentro del amplio mundo de Java EE será de lo más sencillo, como los Servlets, a lo más complejo, que es la razón de ser del temario propuesto.

Obtendrás unas bases sólidas para crear aplicaciones web con Java EE de inicio a fin mediante proyectos individuales y grupales.

**MOTOR DE BASE DE DATOS ORACLE 18C**

Oracle Database 18c es la última generación de esta tecnología; diseñada con la nueva funcionalidad Multitenant Architecture que funciona, esencialmente, con la base de datos que ejecuta tu data center, Oracle Cloud, Oracle Database y Engineered Systems, permitiendo mayor despliegue en el entorno In the Cloud, On- Premises y Hybrid Cloud.

Oracle Database 18c se continúa innovando, como el primer producto de Oracle Database que seguirá un modelo de lanzamiento anual. Esto resulta en una simplificación de las versiones con un producto que introduce nuevos cambios con mayor frecuencia.

Desde ya puedes experimentar las últimas características y mejoras del producto, aprovechando las capacidades nuevas y únicas.

**Capacitaciones Oracle Database 18c**

Oracle University se compromete a ayudar a las empresas y a que sus colaboradores aprendan Oracle desde Oracle. Estos cursos de capacitación utilizan lo último en tecnología de base de datos para la implementación segura y rentable de cargas de trabajo transaccionales y analíticas.

**CONTENEDOR DE APLICACIÓN “APACHE TOMCAT”**

Tomcat es un contenedor de servlets que se utiliza en la Referencia oficial de la implementación para Java Servlet y JavaServer Pages ([JSP](https://www.ajpdsoft.com/modules.php?name=Encyclopedia&op=content&tid=831)). Las especificaciones Java Servlet y JavaServer Pages son desarrolladas por Sun Microsystems cuyas especificaciones vienen dadas por la JCP (Java Community Process). Apache Tomcat es desarrollado en un entorno abierto y participatorio, bajo la licencia de Apache Software License.

Para simplificar, podríamos decir que Apache Tomcat (o Jakarta Tomcat) es un software desarrollado con Java (con lo cual puede funcionar en cualquier sistema operativo, con su máquina virtual java correspondiente) que sirve como servidor web con soporte de servlets y [JSPs](https://www.ajpdsoft.com/modules.php?name=Encyclopedia&op=content&tid=831).

Tomcat es mantenido y desarrollado por miembros de la Apache Software Foundation y voluntarios independientes. Los usuarios disponen de libre acceso a su código fuente y a su forma binaria en los términos establecidos en la Apache Software License. Las primeras distribuciones de Tomcat fueron las versiones 3.0.x. Las versiones más recientes son las 7.x, que implementan las especificaciones de Servlet 3.0 y de JSP 2.2. A partir de la versión 4.0, Jakarta Tomcat utiliza el contenedor de servlets Catalina.

Tomcat es un servidor web con soporte de servlets y JSPs. Tomcat no es un servidor de aplicaciones, como JBoss o JOnAS. Incluye el compilador Jasper, que compila [JSPs](https://www.ajpdsoft.com/modules.php?name=Encyclopedia&op=content&tid=831) convirtiéndolas en servlets. El motor de servlets de Tomcat a menudo se presenta en combinación con el servidor web Apache.

Tomcat puede funcionar como servidor web por sí mismo. En sus inicios existió la percepción de que el uso de Tomcat de forma autónoma era sólo recomendable para entornos de desarrollo y entornos con requisitos mínimos de velocidad y gestión de transacciones. Hoy en día ya no existe esa percepción y Tomcat es usado como servidor web autónomo en entornos con alto nivel de tráfico y alta disponibilidad.

**La jerarquía de directorios de instalación de Tomcat incluye:**

* bin - arranque, cierre, y otros scripts y ejecutables.
* common - clases comunes que pueden utilizar Catalina y las aplicaciones web.
* conf - ficheros [XML](https://www.ajpdsoft.com/modules.php?name=Encyclopedia&op=content&tid=800) y los correspondientes DTD para la configuración de Tomcat.
* logs - logs de Catalina y de las aplicaciones
* server - clases utilizadas solamente por Catalina.
* shared - clases compartidas por todas las aplicaciones web.
* webapps - directorio que contiene las aplicaciones web.
* work - almacenamiento temporal de ficheros y directorios.

**CONTROLADOR DE VERSIONES GIT**

Git, es un software de control de versiones diseñado por Linus Torvalds. La pregunta es ¿qué es control de versiones? Pues bien, se define como control de versiones a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración del mismo es decir a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración, y para los que aún no les queda claro del todo, control de versiones es lo que se hace al momento de estar desarrollando un software o una página web. Exactamente es eso que haces cuando subes y actualizas tu código en la nube, o le añades alguna parte o simplemente le editas cosas que no funcionan como deberían o al menos no como tú esperarías.

Y, entonces ¿a que le llamamos sistema de control de versiones? Muy sencillo, son todas las herramientas que nos permiten hacer todas esas modificaciones antes mencionadas en nuestro código y hacen que sea más fácil la administración de las distintas versiones de cada producto desarrollado; es decir Git.

**Git**

Git fue creado pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente, es decir Git nos proporciona las herramientas para desarrollar un trabajo en equipo de manera inteligente y rápida y por trabajo nos referimos a algún software o página que implique código el cual necesitemos hacerlo con un grupo de personas.

Algunas de las características más importantes de Git son:

* Rapidez en la gestión de ramas, debido a que Git nos dice que un cambio será fusionado mucho más frecuentemente de lo que se escribe originalmente.
* Gestión distribuida; Los cambios se importan como ramas adicionales y pueden ser fusionados de la misma manera como se hace en la rama local.
* Gestión eficiente de proyectos grandes.
* Realmacenamiento periódico en paquetes.

**Ordenes básicas**

Iniciar un repositorio vacío en unas carpeta específica.

git init

Añadir un archivo especifico.

git add “nombre\_de\_archivo”

Añadir todos los archivos del directorio

git add .

Confirmar los cambios realizados. El “mensaje” generalmente se usa para asociar al commit una breve descripción de los cambios realizados.

git commit –am “mensaje”

Revertir el commit identificado por "hash\_commit"

git revert “hash\_commit"

Subir la rama(branch) “nombre\_rama” al servidor remoto.

git push origin “nombre rama”

Mostrar el estado actual de la rama(branch), como los cambios que hay sin hacer commit.

git status

**COMPATIBILIAD DE LA APLICACIÓN WEB POS\_RESTAURANTE CON DISPOSITIVOS**

1. Telefonos movil ios y android
2. Tabletas
3. Computadoras con cualquier sistema operativo

**JASPERSOFT STUDIO**

Jaspersoft anuncia [Jaspersoft Studio](http://www.jaspersoft.com/es/jaspersoft-studio), el un entorno de diseño de BI de código abierto para Eclipse. Jaspersoft Studio es una plataforma de código abierto completa que aporta a los desarrolladores Java de Eclipse nuevas funciones de BI y les permite crear, securizar y compartir informes de BI de forma gratuita.

Eclipse constituye una de las mayores y más importantes comunidades Java, con una media de un millón de descargas mensuales. Jaspersoft, con más de 225.000 miembros registrados, es la mayor comunidad de BI de código abierto. Con Jaspersoft Studio, estas dos importantes comunidades de desarrolladores se ven reunidas con el fin de conseguir una mayor adopción de la inteligencia de negocio de código abierto entre los usuarios. En comparación con las herramientas de informes que hay actualmente a disposición para Eclipse, las funciones de BI que proporciona Jaspersoft a los desarrolladores de Eclipse son mucho más amplias.

Jaspersoft Studio supera las funciones de las herramientas de generación de informes actualmente disponibles para la comunidad de Eclipse, permitiendo a los desarrolladores crear informes en [Jaspersoft Studio](http://www.jaspersoft.com/es/jaspersoft-studio) y compartirlos con otros equipos o con su organización mediante el proyecto de código abierto [JasperReports Server](http://jasperforge.org/index.php?q=project/jasperserver). Jaspersoft Studio proporciona una verdadera plataforma de código abierto para crear, compartir y gestionar informes sin que los desarrolladores tengan que comprar software comercial. Los usuarios pueden diseñar y distribuir sus informes a partir de cualquier fuente de datos como bases de datos relacionales, “big data” y NoSQL o a partir de ficheros planos.

Guilio Toffoli, Fundador y Arquitecto Principal de iReport de Jaspersoft dijo: “Los desarrolladores pueden familiarizarse bastante rápido con las herramientas de generación de informes de Eclipse, pero lo malo es que no ofrecen muchas opciones y es poco lo que puede hacerse con ellas. Jaspersoft Studio está diseñado para ofrecer a la comunidad de Eclipse una experiencia de BI más completa. Ahora, si un desarrollador desea desplegar o securizar un informe de BI, puede utilizar nuestro servidor sin tener que adquirir una costosa licencia comercial.”

Además de poner a disposición de los desarrolladores de Eclipse las soluciones de Jaspersoft, la empresa anunció también que se ha convertido en un miembro oficial de Eclipse Foundation. Al incorporarse a esta comunidad, Jaspersoft amplía sus opciones y su capacidad para desarrollar software de código abierto y crear aplicaciones y productos de código abierto.

Brian Gentile, CEO de Jaspersoft dijo: “En Jaspersoft queremos contribuir al éxito de todos los desarrolladores con nuestro potente y probado software de código abierto. Al combinar las fuerzas de Jaspersoft y Eclipse impulsamos la inteligencia de negocio de código abierto, proporcionando a un número mayor de clientes las herramientas necesarias para mejorar su toma de decisiones.”