



Bloc A

- Característiques del desenvolupament web: web development stack, backend, frontend, API REST, peticions http, serveis,
 - Spring framework: característiques, utilitats, components, versions, entorns de desenvolupament adjunts
-

Característiques del desenvolupament web

En primer lloc quan parlem del desenvolupament web, el primer que ens apareix és el concepte de **Web Development Stack**. Aquest concepte fa referència a tot el que engloba a una pàgina web des del punt de vista de la seva creació i manteniment. Ho podem dividir en dos grans grups, el **BACKEND** i el **FRONTEND**.

FRONTEND

El **frontend**, és la part que va més relacionada amb el disseny que té la web, aquesta part definirà els estils i estructura que tindrà. És una part més enllaçada en atreure al públic, ja que és el que ells veuen. En el desenvolupament d'aquesta part, ens topem

amb 2 llenguatges, el **HTML** i el **CSS**. L'**html**, determina l'estructura amb que es presentarà l'informació a la web. El **css** determina l'estil amb que presentarem l'estructura presentada al html, és a dir, els colors, tipografia del text, etc.

BACKEND

El **backend**, és la part de la web que controla totes les dades i especifica la manera de funcionar de l'aplicació. Al contrari que amb el frontend, aquesta part és inaccessible per als usuaris. En aquest cas, el número de llenguatges que es solen usar, és més alt i divers. Podem trobar des de Python, PHP, C#, etc. Hi ha molta varietat de frameworks que tracten de maneres diferent els llenguatges, s'elegeix el que millor s'adapti al projecte que es vol dur a terme.

BASES DE DADES

A més a més d'aquests dos grans grups del Web Development Stack, també trobem, les **BASES DE DADES**. Totes les aplicacions de l'actualitat, requereixen d'una base de dades, sense importar de quin tipus sigui.

Els principals tipus de bases de dades que trobem a una aplicació, són, les **bases de dades relacionals** o les **bases de dades orientades a documents**.

Les **bases de dades relacionals**, pretenen emmagatzemar tota la informació de manera estructurada i enllaçant diversos tipus de dades entre sí.

Les **bases de dades orientades a documents**, al contrari que el tipus anterior ofereixen una estructura més “desordenada” que no relaciona la informació entre sí.

A l'hora de triar quin tipus de base de dades utilitzarem per a la nostra aplicació, ens guiarem depenent del tipus d'aquesta. En casos on la informació que emmagatzema l'app, no té relació entre sí utilitzarem les BD orientades als documents ja que si utilitzessim les altres, hauriem de modificar moltes parts per a que pugui funcionar bé. Si, en canvi, volem tenir un control més gran de les dades i no acabar tenint un desastre de dades haurem d'utilitzar les BD relacionals.

API REST

Si parlem de que són les **API**, veiem que, aquestes són conjunts de definicions i protocols utilitzats per integrar el Software de les aplicacions. Es a dir, la seva funció és permetre la interacció d'un sistema per obtenir dades o executar una funció, de manera que el sistema entengui les sollicituds i les compleixi.

Quan parlem de **REST**, no estem parlant d'un protocol ni un estàndard sinó d'un conjunt de límits de l'arquitectura. Límits que apliquen els desenvolupadors de les API de diferents maneres.

Per tant, un cop ja sabem que significa cada paraula per separat, podem dir que un API REST, és una interfície de programació d'aplicacions que segueix uns límits establerts.

Amb les api rest les empreses poden compartir els recursos que volen, conservant la seguretat, el control i l'autenticació, això fa que puguin restringir el contingut que pot accedir cada usuari.

PETICIONS HTTP

En primer lloc l'HTTP, és el protocol que permet enviar documents d'un lloc a un altre de la web, un protocol és un conjunt de regles que determina quins missatges es poden intercanviar i quins missatges són respostes apropiades a altres.

Dintre del HTTP, trobem el model de **CLIENT-SERVIDOR**, per norma general el client inicia la conversació amb el servidor, aquest respon en forma de bits de text.

L'HTTP, defineix un conjunt de mètodes de petició per indicar l'acció que es vol realitzar per a un recurs determinat. Trobem els següents tipus de peticions:

▼ GET

Sol·licita una representació d'un recurs específic

▼ HEAD

Demana el mateix que la petició GET, però sense el cos de la resposta

▼ POST

Per enviar una entitat a un recurs específic

▼ PUT

Substitueix totes les representacions actuals del destí amb la càrrega de la petició

▼ DELETE

Esborra un recurs en específic

▼ CONNECT

Estableix un túnel al servidor indicat

▼ OPTIONS

Descriu les opcions de comunicació per al recurs del destí

▼ TRACE

Realitza una prova de bucle de retorn de missatge al llarg de la ruta de destí

▼ PATCH

Serveix per aplicar modificacions parcials a un recurs

SERVEIS WEB

Els serveis web són una tecnologia que permet intercanviar dades entre aplicacions sense importar la plataforma sobre la que estiguin executades, el llenguatge amb que s'ha programat o el lloc on es situa geogràficament.

Un dels avantatges que els caracteritzen és l'estandarització dels formats de text, ja sigui en un llenguatge com XML, possible d'entendre tant per a màquina com per als humans.

Ara bè, aquest avantatge que acabo de comentar genera un inconvenient a la vegada, és a dir, estandaritzar les dades a un format de text comprensible per tothom, fa que aquestes dades ocupin un espai significativament gran, que pot desencadenar en que algun usuari que tingui una connexió lenta, tingui dificultats per accedir de manera ràpida.

SPRING FRAMEWORK

Spring Framework, és un entorn de programació Java, destinat a la programació web. Aquest s'ha fet molt famós per basar-se en la filosofia de "convenció sobre configuració", és a dir, intenta reduir al màxim els passos que ha de fer el programador a l'hora de la configuració inicial d'un projecte, fent que el programador només s'hagi de centrar en la part important del projecte i no tenir que repetir accions que ja ha fet en altres projectes a l'hora d'inicialitzar-los. D'aquesta manera s'aconsegueix estalviar temps i diners.

Les característiques que trobem són les següents:

- Spring és completament modular i té suport a diferents tecnologies com injecció de dependències, events, recursos, validació, enllaç de dades, conversions de tipus, etc.
- L'accés a dades soporta, DAO, JDBC, ORM, Marshalling XML
- Gestió de transaccions
- Integració de comunicació remota, JMS, JCA, JMX, correu electrònic, tasques, programació, caché
- Simulacre d'objectes

Punts a favor

Spring es divideix en mòduls, com ja he comentat. Cada mòdul fa una funció diferent que ens ajudarà a l'hora de desenvolupar un programa. Per exemple podem trobar mòduls que, substitueixin 50 línies de codi per 5.

La seguretat dels serveis web també cal destacar-la, ja que ho fa de manera senzilla i ocupant-se de totes les parts possibles.

En el cas de desenvolupament d'API REST, Spring ens ofereix moltes avantatges, ja que aquest gestiona les peticions, analitza el tipus de dades i decideix sobre quina classe de java treballar per poder processar aquestes dades.

Versions

0.9 (2002)

1.0 (2004) Agrega la llicència de Apache 2.0

1.2.3 (2004) Guanyadora del premi de productivitat Jold i a la innovació JAX

2.0 (2006) Soporta Java 5

2.5 (2007) Afegeixen noves funcions com configuracions de anotació

3.0 (2009) Soporta Java EE6

3.1 (2011)

3.2 (2012) Introdueix configuració de Java, soporta Java 7

4.0 (2013) Soporta Java 8

4.3.7 (2017)

5.0 (2017)

Entorns de desenvolupament

1. **Java 8:** Aquestes versions de Java ens aporten API's més completes i avançades, expressions lambda i més.
2. **Eclipse:** Aquest IDE facilita el desenvolupament de codi a través de diferents funcions integrades com poden ser:
 - a. Navegació de codi
 - b. Compilació a temps real
 - c. GIT
 - d. Maven
3. **Maven:** És un gestor de dependències i algunes coses més, permet gestionar algunes propietats com poden ser:
 - a. Nom
 - b. Versió
 - c. Tipus d'empaquetat
 - d. Perfils
 - e. Dependències
4. **MySQL:** Serveix per emmagatzemar la informació en una base de dades, també podem administrar totes les dades.
5. **Git:** Per el control de versions utilitzarem Git, està pensat de manera que sigui eficient en quan a confiabilitat i compatibilitat del manteniment de les versions d'aplicacions quan aquestes tenen un gran número d'arxius de codi font.
6. **Postman:** Es tracta d'una eina dirigida al desenvolupament web, que permet realitzar peticions HTTP a qualsevol API. És molt útil a l'hora de programar i fer proves.
7. **Tomcat:** És un contenidor de serveis que es pot fer servir per compilar i executar aplicacions web realitzades amb Java.