

<https://mensuel.framapad.org/p/architecture>

Formateur : Mohamed DERKAOUI

- mderkaoui@dawan.fr
-

Formation Architecture Logicielle

Plan de formation : <http://www.dawan.fr/formations/gestion-de-projets/systemes-d-information/architecture-logicielle--fondamentaux>

Support : <http://demo.dawan.biz/mohamed/architecture.pdf>
<http://demo.dawan.biz/mohamed/mapArchi.pdf>

Horaires :

9h30 - 17h30

9h-17h

Pauses : 2x15mn + repas

Composant logiciel = élément constitutif d'un logiciel destiné à être incorporé en tant que pièce détachée dans des applications

Service web : composant web déployé sur un serveur d'applications exposant (interface) un ensemble de fonctionnalités métiers
(sans ihm)

WeatherService

- récupérerTemperature(ville)
- récupérerHumidité(ville)
-

Micro-service : composant exposant 1 seule fonctionnalité métier

- Micro service CRUD
- Create
-

<http://demo.dawan.biz/mohamed/modele-dat1.pdf>

<http://demo.dawan.biz/mohamed/modele-dat2.docx>

<http://demo.dawan.biz/mohamed/modele-dat3.doc>

<http://demo.dawan.biz/mohamed/modele-dat4.pdf>

Objectif :

sortir les grandes lignes du contenu d'un dossier d'architecture technique

Contenu :

- Vision globale du projet => Lean Canvas
- Vue fonctionnelle :
Fonctionnalités de l'application
 - vision macro => diagramme de cas d'utilisation global
 - ou diagramme global par acteur
 - - Choix de l'architecture
 - + motivation du choix
- Vue de composants (organique) :
 - * Etat de l'art :
composants existants/interactions => diagramme de composants
infra ==> diag de déploiement
on peut combiner : diag combiné de composants et de déploiement
 - * Contraintes :
sécurité,
accès aux données,
matériel,
sauvegarde,
volumétrie, disponibilité...
stratégie existante/préconisée
 - * Flux de données (interactions des nouveaux composants)
- Vue des flux physiques
protocoles utilisés (et non les données)
 - flux externes
 -
- Vue de déploiement
architecture physique système => Diag de déploiement
serveurs d'applications, entrepôts de données, espaces de stockage
serveur de supervision
- Vue de réalisation
Processus de Logs des traitements ?
Schéma d'authentification ?
Mécanisme de reprise après incident
Mécanisme de réplication
- Diagramme de structure composite pour donner une indication sur l'architecture interne d'un composant
- Choix technologiques :

- * Frameworks + motivation
 - service NodeJS vs service JAX-RS (benchmark)
 - web : front JS ou un framework MVC back
 - Angular ou ReactJS ou VueJS
 - vs
 - Spring MVC
 - persistance :
 - Bdd vs NoSQL (volume de données, accès)

DevOps : mouvement permettant de rapprocher opérationnels et développeurs

Objectif : essayer de réaliser un déploiement en continue

URN : Uniform Resource Name

- (nom de la ressource)
-

URI : Uniform Resource Identifier
(identifiant unique d'une ressource)

URL : Uniform Resource Locator

- (adresse physique de la ressource)
-
- ---

 client =====requete1 =====> boutique.orange.fr
 - serveur crée un identifiant unique associé à l'utilisateur
 - sessionId=XXX

<===== rep1+cookie =====

le navig web stocke
le cookie

- ===== requete2+cookie =====>

- Le suivi de session est assuré par un identifiant unique véhiculé :

- soit dans un cookie
- soit dans une URL

- Web Service : composant web autonome et exécutable, interopérable (différents langages)
- --
- Application = ensemble de composants
- Styles architecturaux à retenir :
 - - en couches
 - - hexagonale (peut être vue comme une variante de couches)
 - - orientée objets
- ---
- Il existe 2 types de serveurs d'applications Java EE :
 - - Web Profile : 1 conteneur web - Tomcat, Jetty
 - - Full Profile : 2 conteneurs (web, EJB) - WebSphere, GlassFish, WebLogic, JBoss, WebFly
- ---
- CDN (Content Delivery Network) = Réseau de diffusion de contenu (RDC)
 - => ensemble d'ordinateurs reliés en réseau à travers le web
 - =====> dans l'objectif de mettre à disposition du contenu (des données)
 - Constitution :
 - - de serveurs d'origine d'où les contenus sont injectés (=> réplication)
 - - des serveurs périphériques (déployés à plusieurs endroits géographiquement distincts)
 - Types de CDN :
 - - CDN de streaming de média
 - - CDN de caching pour le téléchargement
-

- ---
- GraphQL : langage de requêtes (développé en interne par Facebook)
- Il offre une alternative aux API REST dont le stockage est éventuellement distribué
- ---
- CaaS (Container as a Service)
 - Une forme de virtualisation par conteneurs dans laquelle les moteurs, l'orchestration et les ressources sont fournies aux utilisateurs sous la forme d'un service.
- IaaS (infra+OS+stockage+ réseau)
 - une sorte d'offre du IaaS => CaaS
- PaaS (IaaS + envir. d'exécution)
 - FaaS : Function as a service
 - développer des applications et déployer des fonctionnalités sans gérer de serveur
- SaaS (logiciel final)
- ---
- Une user story doit être INVEST :
 - - Indépendante
 - - Négociable
 - - Verticale (ayant de la valeur)
 - - Estimable

- - Suffisamment petite
- - Testable
- _____
- Caractéristiques d'un composant :
 - - Réutilisable : conçu pour être réutilisé dans différentes situations dans diff. applications
 - - Remplaçable : doit pouvoir être librement substitué par un autre composant similaire
 - - Libre de contexte : le composant est conçu pour être exécuté dans différents environnements
 - - Extensible : on peut étendre un composant à partir de composants existants qui fournissent un comportement proche.
 - - Encapsulé : le composant possède des interfaces, qui autorisent l'appelant à utiliser ses fonctionnalités et n'exposent pas le détail des processus internes / état interne.
 - - Indépendant : conçu en ayant des dépendances min envers d'autres composants
-

•