**QUESTION ATELIER 1 :**

**Web Static et Web dynamique**

**-----------------------------------------**

**Qu’est-ce que le pattern MVC ?**

Le pattern MVC ou Modèle-Vue-Contrôleur est un motif d'[architecture logicielle](https://fr.wikipedia.org/wiki/Architecture_logicielle) destiné aux [interfaces graphiques](https://fr.wikipedia.org/wiki/Interface_graphique) lancé en 1978 et très populaire pour les [applications web](https://fr.wikipedia.org/wiki/Application_web). Le motif est composé de trois types de modules ayant trois responsabilités différentes : les modèles, les vues et les contrôleurs.

* Un modèle (Model) contient les données à afficher.
* Une vue (View) contient la présentation de l'interface graphique.
* Un contrôleur (Controller) contient la logique concernant les actions effectuées par l'utilisateur. Il sert de lien entre les deux autres modules.

**Quels avantages présente-t-il ?**

Un avantage apporté par ce modèle est la clarté de l'architecture qu'il impose. Cela simplifie la tâche du développeur qui tenterait d'effectuer une maintenance ou une amélioration sur le projet. En effet, la modification des traitements ne change en rien la vue. Par exemple on peut passer d'une base de données de type [SQL](https://fr.wikipedia.org/wiki/Structured_Query_Language) à [XML](https://fr.wikipedia.org/wiki/Extensible_Markup_Language) en changeant simplement les traitements d'interaction avec la base, et les vues ne s'en trouvent pas affectées. Nous avons durant cette année utilisé le modèle MVC dans le module COO (Conception Orienté Objet), cela permet notamment de respecter des bonnes pratiques de codes mais aussi de debugger le code de façon plus efficace.

**Qu’est-ce que le Web dynamique ?**

Un site dynamique désigne un site sur lequel certaines pages sont des pages dynamiques, c’est-à-dire des pages créées à la demande (on parle aussi de page générées « à la volée ») par le serveur. Le contenu des pages dynamiques va donc changer sans que cela soit le fait du webmaster. Les pages seront générées en temps réel à partir d’informations contenues dans des bases de données en fonction d’interaction avec les internautes. Dans le cas du projet, l’affichage d’une carte sera fait lorsque l’utilisateur cliquera dessus et la page web affichée sera fonction du prix, du nom de la carte ou encore de ses caractéristiques stockées en base de données.

**Pourquoi est-il intéressant ?**

Un site internet dynamique permet de modifier son site beaucoup plus rapidement et plus facilement. Cela va permettre à la personne mettant à jour le contenu de simplement s’occuper du texte des image et non de la forme qui sera déjà établi. Pour un exemple plus concret c’est ce qu’on peut observer lors de l’ajout d’un article dans une boutique de e-commerce.

**Comment sont récupérés les fichiers par le Web Browser en Web statique ?**

En Web statique, la récupération de fichier se fait de « façon directe » car il affiche le même contenu à chaque visite. Ainsi, le Web Browser exécute le script HTML et récupère les informations demandées coté serveur via une requête. Tant qu’il n’y a pas d’erreur, les informations de la page sont téléchargées puis affichées sur l’ordinateur.

**Quels sont les avantages d’utiliser du Web Statique avec des services REST ?**

**REST (pour REpresentational State Transfer)** ne décrit pas une implémentation mais un style d’architecture qui suit différents principes :

* Client-serveur : les responsabilités sont séparées entre le client et le serveur. L’interface utilisateur est séparée de celle du stockage des données. Cela permet aux deux d’évoluer indépendamment (contrairement à des applications qui réalisent des requêtes directement en base de données par exemple…).
* Sans état : la requête envoyée par le client doit être auto-suffisante : ne pas nécessiter de sauvegarder un état sur le serveur
* Mise en cache : la possibilité pour le serveur de spécifier les réponses pouvant être mises en cache
* Une interface uniforme : chaque ressource est identifiée unitairement, les ressources ont des représentations définies
* Un système hiérarchisé par couche : les états de l’application sont représentés par des ressources individuelles. L’ensemble de l’information n’est pas envoyé dans une ressource unique.

Le quatrième point permet de modifier l’affichage dans un site statique de façon plus simple afin de gérer le design des différentes pages.

**Qu’est que les architectures N-Tiers ?**

L'architecture N-tier (anglais tier : étage, niveau), ou encore appelée multi-tier, est une architecture client-serveur dans laquelle une application est exécutée par plusieurs composants logiciels distincts

**Quels avantages apportent-elles ?**

* Gestion flexible et multi-système.
* Le lien entre les niveaux est défini et limité à des interfaces.
* Les interfaces assurent la modularité et l’indépendance technologique et topologique de chaque niveau.

**Comment fonction l’AJAX ?**

L’AJAX n’est pas une technologie à proprement parler, mais un principe. Quand on demande une page web, vous émettez une requête à un serveur web, qui vous renvoie un résultat, sous forme d’HTML (ensuite formaté par du CSS et du JavaScript).

L’AJAX c’est pareil, mais sans que votre page web soit rechargé. Il s’agit d’une requête à un serveur web (souvent le même), mais programmée en JavaScript suite à une action (clic sur un bouton, scroll…).

**Qu’est-ce que JEE ?**

JEE (*Java Entreprise Edition*) est la version entreprise de la plate-forme "Java" qui se compose de l'environnement "JSE" ainsi que de nombreuses API et composants destinés à une utilisation "côté serveur" au sein du système d'information de l'entreprise. Il s'agit donc d'une évolution du Java.

**Qu’est-ce qu’un Web Container en JEE ?**

Un conteneur de servlets (*Servlet container* en anglais) ou conteneur web (*web container* en anglais) est un logiciel qui exécute des [servlets](https://fr.wikipedia.org/wiki/Servlet). Parfois, ils sont appelés moteur web ou moteur de servlets.

**Qu’est-ce que Springboot ?**

Spring Boot est un Framework de développement JAVA. C'est une déclinaison du Framework classique de Spring qui permet essentiellement de réaliser des microservices (ce sont la majeure partie du temps des services web qui sont regroupés en API).

**Quelles différences avec JEE ?**

JEE :

1. Java EE de l'industrie a approuvé la norme API basée sur le cadre
2. Il est principalement basé sur les annotations et CDI
3. JFC MVC Framework de développement web
4. Spécification JPA pour traiter DB opération
5. API JTA avec la mise en œuvre
6. Conteneur d'EJB et de POJO en fonction de la mise en œuvre
7. Oracle licence

Springboot :

1. Basé sur du CIO et de l'AOP
2. Basé sur un fichier XML de configuration (maintenant qu'ils sont en tirant parti de l'annotation)
3. Utilise Printemps DAO cadre (basé sur le Modèle de design pattern) pour se connecter à la base de données
4. Fournit une couche d'abstraction à l'appui de divers JTA mise en œuvre vendeur
5. S'intègre avec les différents Java vendeurs pour le support des fonctionnalités tels que struts, …
6. Fournit de bout en bout plate-forme pour construire des applications web atteindre le couplage à l'aide de DI et AOP
7. Licence Open source

**Qu’apport Thymeleaf à Springboot ?**

Le fichier modèle (Template file) de **Thymeleaf** est en substance un fichier de document ordinaire au format **XML/XHTML/HTML5**. **Thymeleaf Engine** (le moteur Thymeleaf) va lire un fichier modèle et le combiner avec des objets **Java** pour générer un autre document.

**Que signifie l’annotation @Controller, pourquoi est-elle importante lors de la génération de pas coté serveur ?**

Un contrôleur est une classe Java portant l’annotation [@Controller](https://docs.spring.io/spring-framework/docs/current/javadoc-api/org/springframework/stereotype/Controller.html). De manière générale, l’objectif d’un contrôleur est de réagir à une interaction avec l’utilisateur. Pour une application Web, cela signifie que l’utilisateur envoie une requête HTTP au serveur.

**Que représente le répertoire ‘src/main/ressources’ dans un projet SpringBoot ? Quelles sont les informations stockées à cet endroit ?**

Ce répertoire est important car c’est ici que nous avons toutes les ressources nécessaires pour le projet (templates, pages css, …).