Diagrammes Atelier 1

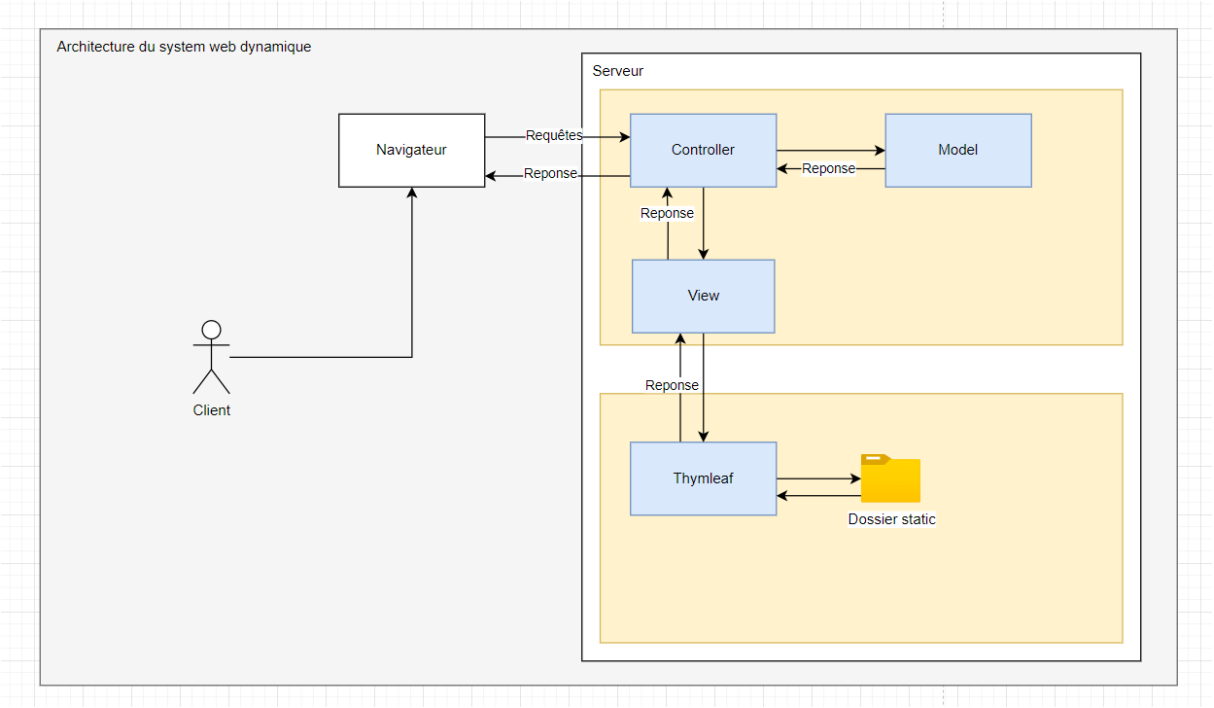

Figure 1 - Architecture du systeme web dynamique

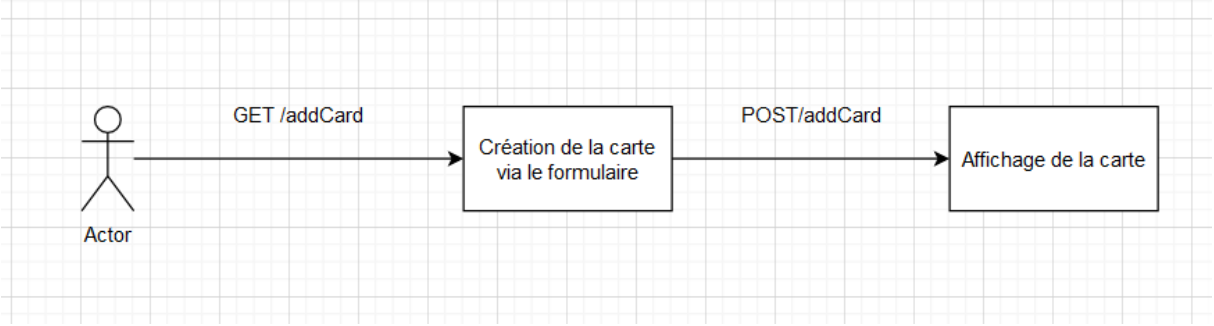

Figure 2 - Workflow d'ajout d'une carte

Tableau de comparaison MVC, SOA, MicroService

Après avoir réalisé les deux étapes de l'atelier, le site web statique puis le dynamique, nous avons pu constater que chaque solution a ses avantages et ses inconvénients. On note que les sites statiques charge plus rapidement et offre beaucoup plus d'interactions avec l'utilisateur sans recharger la page entière contrairement à une page dynamique qui ne contient pas de JavaScript. Cependant, le site dynamique offre un développement plus rapide, car plus simple, ainsi qu'un meilleur SEO, parce que le contenir de la page est généré côté serveur avant l'envoi.

Architecture	***Description***	***Avantages***	***Inconvénients***
MVC	application en trois composants interconnectés : Modèle, Vue, Contrôleur	- Facile à comprendre et à mettre en place - Un monolithe avec une couche réseau limitée	- Difficile à maintenir sur le long terme car l'architecture en monolithe rend les évolutions plus compliqués - Difficile à tester
SOA	architecture qui se concentre sur des services au sein d'une application lieu d'une conception monolithique. .	- Facilite la réutilisation des services - Facilite l'intégration de nouveaux services car une seule application et donc une seule base de donnée	- Complexité de mise en place - Difficulté à maintenir car monolithe
MicroService	architecture de plusieurs micros applications autonomes	- Facilite la maintenance, chaque partie de l'application est séparé - limite les failles car les applications sont autonommes (pas de base de données partagé par exemple)	- Complexité de mise en place car énormément d'échange réseaux, plusieurs base de données etc..

Mais on ne peut pas opposer ces architectures, car elles ne répondent pas aux mêmes besoins. En effet, elles ont chacune des avantages et des inconvénients qui les rendent plus ou moins adaptées à un contexte donné. Par exemple, le MVC est plus adapté pour des applications simples tandis que le MicroService est plus adapté pour des applications complexes. Il est donc important de bien comprendre les besoins de l'application avant de choisir une architecture.

Workflow git

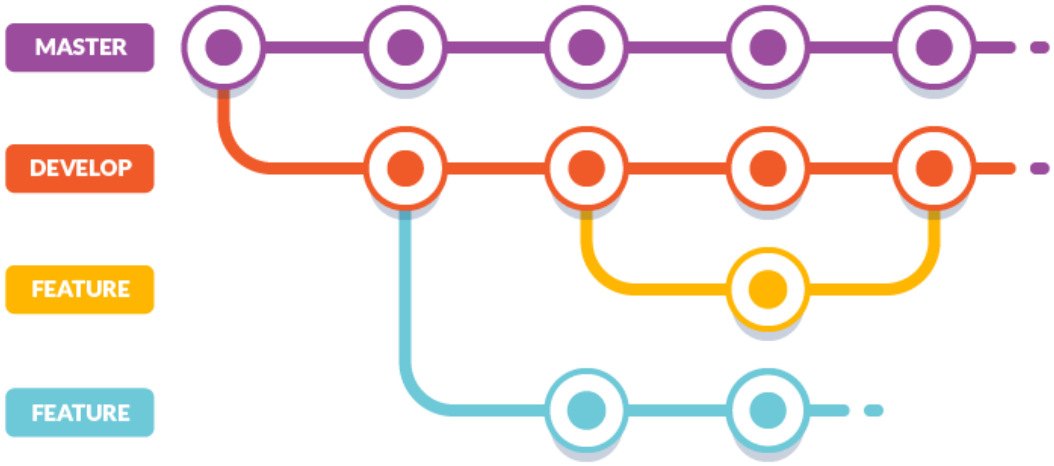

Figure 3 - workflow git