

Javascript

PROFESSEUR : Nicolas PRIEUR

BINÔME : Houda ALIABBAS & Nourdine MOUMINI

PROJET : POST-IT

Année : 2020

INTRODUCTION :

Dans le cadre de notre 3ème année de licence en Informatique à Paris 8, nous avons travaillé sur un projet en Javascript avec une architecture MVC.

TECHNOLOGIE :

FileSysteme : Nous avons choisi cette API car elle permet comme une BDD d'utiliser la méthode CRUD (Create, Read, Update, Delete).

Socket.io : Nous avons décidé de choisir socket.io car il permet des interactions en temps réel.

POST-IT :

Nous avons réalisé un post-it où il est possible de modifier le corps du texte, d'ajouter un titre à la création du post-it et de le supprimer. Nous n'avons pas pu créer une base de données et la lier aux activités des post-it, comme la sauvegarde des post-it, l'ajout de photo (grâce au BLOB). Nous n'avons pas pu faire de comptes d'utilisateur

ARCHITECTURE :

L'architecture MVC est assez complexe lorsqu'on a pas l'habitude de travailler en orienté objet, la chose la plus complexe est évidemment les appellent de fonction car elle fusent dans tout les sens. Une pratique est essentiel pour l'assimilation du concept. Mais il est claire qu'un bon modèle MVC rend l'application beaucoup plus simple a maintenir sur du long terme.

Le View est simplement pour le visuel de l'application.

Le Controller à un rôle un peu complexe car il peut empiéter sur le View par exemple en récupérer la valeur pour l'envoyer au Model, alors que View peut récupérer la valeur et l'envoyer à une fonction du View qui fera appelle au Controller qui l'envoi au Model.

Le Model gère lui la discussion entre serveur et l'application.

DIFFICULTE :

Les difficultés sont nombreuse mais 2 sortent du lot.

1 – La base de donnée : Nous n'avons pas pu implémenter une liaison Model <=> Server.

Récupérer les "values" d'un input et l'envoyer au model était simple mais la connexion server <=> model est plus complexe, dans notre cas la liaison était perdu entre le model et le server. Nous n'arrivions pas a déterminer

pourquoi.

2 – Post-it : Le post-it est complexe car nous devons gérer le déplacement et les superpositions.

Notre approche naïve des choses a était de vouloir récupérer l'id des éléments sur lequel on a cliquer et de travailler ainsi en changeant les ID a chaque événement clique.

Mais n'est-il pas plus simple de travailler sur une div ? Puisque nous dupliquons les éléments. Bien évidemment que oui. Il a été essentiel de donner une position absolue a notre div car sinon elle se faisait pousser dans tout les sens.

CONCLUSION : Le projet était un peu compliqué, il y a beaucoup d'améliorations. Mais c'était extrêmement enrichissant.