

Architecture du programme Post-it

Ghouibi Ghassen

19 novembre 2018

Table des matières

I	Introduction	2
1	Description du projet	2
II	Architecture du code	2
2	Objets du projet	2
2.1	Utilisateur	2
2.2	Post-it	2
2.3	Interface	2
3	Model View Controller	3
3.1	Model	3
3.2	View	3
3.3	Controller	3
III	Outils pour réaliser ce projet	3
4	Front-end	3
4.1	HTML + CSS + Javascript	3
5	Back-end	3
5.1	Nodejs + MySQL	3

Première partie

Introduction

On a tendance a oublié nos idées quand on a pas de bloc notes sur les yeux,ou simplement de décomposer nos idées en morceaux .
Personnellement ça fait trois ans que j'utilise des post-it pour mes études,rendez-vous ou des rappels (exemple : passer à la poste, résilier un contrat .. etc)et
Surtout des phrases de motivation !
La réalisation d'une API post-it serai génial !

1 Description du projet

Le but de projet est de disséminer des post-it virtuels sur un repère infinie,Ces derniers vont suivre les mouvements du curseur à l'aide de la souris pour être chargé par le navigateur.

Pour éviter les conflits on introduit une base de données,d'où la nécessité d'avoir un compte pour chaque utilisateur.

Deuxième partie

Architecture du code

2 Objets du projet

2.1 Utilisateur

L'utilisateur qui va se connecter sur notre API doit créer un compte.

Donc c'est nécessaire qu'il nous communiquer son nom,prénom,pseudo,mail,mot de passe.

2.2 Post-it

un tableau de Post-it ,chaque Post-it est représente avec un rectangle avec un champs text,des coordonnées x et y et la distance entre lui et l'origine et l'angle de déviation ,aussi il aura une couleur (vert,orange,jaune,rose).

On aura deux fonctionnalités qui s'applique a cet objet :

- Ajouter un Post-it
- Supprimer un Post-it

2.3 Interface

L'interface se décompose en deux partie :

- Premier interface d'inscription ou de connexion

-Deuxième interface les post-it qui vont occuper tout l'écran (visible par l'utilisateur) dans un repère infini et vont suivre le mouvement de la souris.

3 Model View Controller

3.1 Model

La création d'une base de données qui contient des informations par utilisateur et par post-it du coup le model traitera :

- DB **utilisateur** (nom,prénom,id-utilisateur,pseudo,mot-de-passe)
- DB **post-it** (id-utilisateur,nombre-de-post-it)
- DB **post-it-coordonnees** (id-utilisateur,nombre-de-post-it,couleur,coordonnées-x,coordonnées-y,distance,angle-x,tangente,text)

3.2 View

Les fonctions de View se décompose sur deux parties :

- Screen View les post-it disponibles dans le champ de l'utilisateur
- Post-it View les post-it placées sur le grand repère infini

3.3 Controller

Les fonctions de Controller se décompose sur deux parties :

- Screen Controller qui va gérer l'écran visible par l'utilisateur
- Post-it Controller il prendra en charge de déplacer les post-it dans le champ de l'utilisateur

Troisième partie

Outils pour réaliser ce projet

4 Front-end

4.1 HTML + CSS + Javascript

Pour l'instant je vais utiliser ces trois langage pour gérer tous ce qui est front-end .Vu que j'ai déjà utiliser avant .

5 Back-end

5.1 Nodejs + MySQL

Je voulais utiliser Nodejs tout simplement parce que je serai pas invité a apprendre un autre langage comme PHP par exemple et aussi la rapidité

de nodejs est tellement impressionnante . Pour connecter notre bases de données je vais utiliser l'application phpMyAdmin