

Rapport de projet de développement web

Yacine Izouine et Louis L'Haridon



Introduction

Contexte

Au cours de notre UE « développement web » encadrée par Marc Lemaire nous avons été amenés à développer un site afin d'évaluer nos compétences. Ce site a pour objectif d'explorer les données publiques des établissements d'enseignement supérieur en France recensés par l'ONISEP. Ce site web permettra de visualiser, consulter et mettre en valeur ces données.

Equipe

Pour ce projet nous travaillons dans un binôme composé de Yacine Izouine et de Louis L'Haridon. Tous deux étudiants en 2ème année de CMI SIC nous avons l'expérience du travail en groupe et il a été facile pour nous de nous organiser afin de travailler de manière efficace.

Répartition des tâches

Nous avons tout d'abord évalué les différentes tâches nécessaires à la réalisation de ce projet. Nous avons choisi de nous en répartir certaines et de faire ensembles d'autres. Voici le tableau de répartition des tâches :

Yacine	Louis
Historique	Charte graphique CSS
Exploration du jeu de données	

Planning prévisionnel

Le projet a été distribué le 19 mars, comme nous devons rendre notre travail le 16 mai il nous faut établir un planning de travail.



Solution adoptée

Nous allons développer un peu plus en détail la solution que nous avons établie pour la réalisation de notre projet.

Charte graphique

Afin de créer un site au style moderne et esthétique nous avons décidé d'adopter une charte graphique minimaliste.

Le minimalisme est l'art de jouer sur une esthétique simple, épurée et des touches parcimonieuses de couleur.

Palette de couleur

Comme notre site exploite les données des établissements du supérieur en France nous avons choisi pour couleur bleu, blanc et rouge.

Notre site étant minimaliste et moderne il nous faut des teintes de couleur fortes et modernes et simples. Le minimalisme privilégie les couleurs « pures » et mates.

Nous partons donc sur un blanc pur, un bleu et un rouge puissant et mat.

Pour le texte il nous faut une couleur simple et lisible. Nous partirons donc sur un noir « pur » et un gris marqué.



Typographie

La typographie a une place très importante dans le web design. Pour améliorer la lisibilité du site nous choisirons une police sans empâtement.

Une police simple pour un design minimaliste et élégant.

Nous utiliserons Google web font afin d'accéder à un large catalogue de polices libres et modernes. Nous retiendrons le choix de Roboto qui, moderne, sobre et lisible dans tous les corps possibles vient compléter à merveille les couleurs minimalistes que nous avons choisi.



Charte

Suivant le choix d'une charte minimaliste nous avons donc élaboré la charte suivante :

H1 (20px, roboto, centré, bleu)

H2 (18 px, roboto, aligné à gauche, gris)

Corps de texte (16px, roboto, justifié, noir)

Lien (16 px, roboto, justifié, rouge)

Nous faisons également le choix d'un bouton menu minimaliste pour accéder à la navigation. Nous travaillerons également les transitions afin de rendre la navigation de l'utilisateur fluide.

Lecture du jeu de données

Les données nous étaient fournies sur différents formats (csv, xml, json). Nous choisissons d'étudier les données via le fichier csv car ce format de fichier est facilement manipulable en php.

Affichage des informations d'un établissement

En étudiant la structure du fichier on repère que chaque établissement possède un code UAI. Ce code UAI (ou Unité administrative immatriculée) est un code unique possédé par chaque établissement d'enseignement supérieur en France.

Nous disposons donc d'un moyen d'identifier facilement un établissement.

Le fichier comporte ensuite pour chaque établissement des informations comme le nom, la tutelle, l'adresse, le type d'établissement...

Nous créons donc la page qui permet d'afficher les informations d'un établissement et, en fonction du code UAI passé par l'adresse, nous affichons les différentes informations de l'établissement en lisant le csv à la ligne correspondante.

Google Maps

Pour une meilleure visualisation des données nous choisissons par la suite d'utiliser l'API de Google Maps afin de visualiser dans un cadre Google Maps l'adresse de l'établissement.

Voici le code permettant d'inclure la recherche Google Maps :

```
$var_adress = $info["adresse"] . $info["CP"] . $info["commune"];  
$address = str_replace(" ", "+", $var_adress);  
  
echo '<iframe id="cusmap" src="https://maps.google.fr/maps?q=' .  
$address . '&output=embed"></iframe>' . "\n";
```

Avec ce script, à partir des données du CSV, on personnalise une recherche Google Maps dans une iframe.

Le cas « portes ouvertes »

En analysant le fichier csv on se rend compte que tous les établissements n'ont pas les dates de leurs portes ouvertes renseignées.

Ces renseignements ne seront donc pas disponibles pour tous les établissements. Nous réfléchissons donc à une solution afin d'avoir un affichage cohérent quel que soit l'établissement.

On choisit donc d'afficher les informations de l'établissement par section :

- Informations
- Contact
- Adresse
- Portes ouvertes
- Localisation

Ces blocs d'information seront disposés de la manière suivante :

Si portes ouvertes renseignées :

Informations		Contact	
Adresse	Portes ouvertes	Localisation	

Si portes ouvertes non renseignées :

Informations		Contact	
Adresse		Localisation	

Sélection de l'établissement

Afin de permettre à l'utilisateur de choisir facilement l'établissement souhaité nous devons élaborer les modes de sélection les plus performant.

Nous choisissons donc de base d'intégrer un sélectionneur alphabétique qui, en fonction de la lettre de l'alphabet n'affiche que les établissements commençant par cette dernière.

Pour récupérer la liste des établissements nous lisons simplement le csv du début à la fin en stockant dans un tableau à deux dimensions le nom et le code UAI de ce dernier. Nous affichons donc par la suite la liste de tous les établissements avec un lien vers la page d'affichage des informations selon leur code UAI.

L'avantage de cette méthode est son agilité, sa simplicité et sa rapidité d'exécution. En effet un tel script est très rapide à l'exécution ce qui fait de son utilisation un avantage pour le web.

Différentiation des types d'établissements.

En étudiant le csv on a repéré la présence de catégories d'établissement. Afin de faciliter le choix de l'établissement par l'utilisateur nous faisons le choix d'associer une icône à chaque type d'établissement.

Recherche par critère.

En plus de la simple sélection par nom nous avons décidé d'intégrer une recherche par critère. Nous avons choisi comme critère l'académie et la ville.

La méthode reste assez semblable, il s'agit, en plus de stocker le nom et l'UAI de stocker à la lecture la ville/l'académie et, via un formulaire php, de permettre à l'établissement de sélectionner une des académies ou villes disponibles et d'en afficher les établissements (qui seront toujours sélectionnables alphabétiquement).

Historique

Afin de permettre à l'utilisateur la meilleure expérience possible il est essentiel que le site possède un historique. Pour réaliser cet historique nous avons choisi d'utiliser les cookies.

En effet à chaque visite d'une page d'établissement on stockera son UAI dans un cookie, ce qui permettra dans une page d'historique d'afficher tous les établissements visités. On proposera la possibilité d'effacer cet historique en supprimant le cookie par un simple formulaire.

Afin de stocker plusieurs établissements dans un seul cookie nous avons choisi la méthode suivante :

Ecriture :

```
Si cookie historique existe
```

```
    Explorer la valeur de cookies en tableau de 8 char
```

```
    Si UAI absent de tableau
```

```
        Ajouter UAI à la fin de valeur de cookie historique
```

```
        mettre à jour valeur de cookie
```

```
Sinon
```

```
    Créer cookie historique ayant pour valeur UAI
```

Lecture

```
Si cookie historique existe
```

```
    Explorer la valeur de cookies en tableau de 8 char
```

```
    Lire le csv et stocker dans un tableau le nom et l'UAI  
    établissements dont l'UAI est dans le tableau
```

```
    Afficher comme pour une recherche standard les résultats
```

Sinon

Afficher « historique vide »

Voici des exemples de la valeur du cookie historique :

- null : aucun établissement n'est stocké dans l'historique
- 2E1D3T43 : l'établissement au code UAI 2E1D3T43 a été visité
- 2E1D3T433D97S234082DNS92 :
 - o Les établissements aux codes UAI 2E1D3T43, 3D97S234 et 082DNS92 ont été visités.

Mise à jour du jeu de données (extension)

Nous proposons également sur ce site la possibilité de mettre à jour le jeu de données en uploadant un nouveau fichier csv via un formulaire :

```
<form method='post' enctype='multipart/form-data' action='upload.php'>
  <div id="opt">
    Fichier: <input type='file' style="width: auto;"
name='file_upload'>
  </div>
  <input type='submit'>
</form>
```

Afin d'éviter les tentatives de hacking et de prévenir des scripts malveillants nous limiterons l'envoi de fichier à plusieurs critères. Seront acceptés les seuls fichiers répondant aux critères suivants :

- extension en .csv
- fichier de moins de 5mo

Nous renommerons ensuite le fichier uploadé correctement et remplaceront celui mis en ligne auparavant.

Après cela la liste des établissements est instantanément mise à jour.

Validation

Nous avons veillé à respecter les normes w3c et notre site est donc entièrement valide. Le code php est sans warning, le css valide niveau 3 et le php compilé en HTML5 valide.

Conclusion

Nous avons fini ce projet à l'heure de l'écriture de ce rapport et sommes assez fiers du travail fourni. Nous avons mis en ligne un site que nous trouvons esthétiquement agréable et fonctionnel.

Si nous devions l'améliorer nous choisirions de :

- le rendre entièrement responsive car seule les pages d'affichage de résultat et d'accueil le sont
- ajouter des fonctionnalités comme des critères de recherche ou la possibilité d'afficher des graphes d'analyse du fichier