

École Polytechnique de l'Université de Tours  
64, Avenue  
Jean  
Portalis  
37200  
TOURS,  
FRANCE  
(33)2-47-36-14-14  
[www.polytech.univ-tours.fr](http://www.polytech.univ-tours.fr)

# Parcours des Écoles d'Ingénieurs Polytech

Année 2022-2023

## **Introduction aux bases de données avec développement d'une application en PHP**

---

Étudiant(e)

**Siméon Joseph-Schenfeigel**  
**Youssef Hafidi**  
[simeon.joseph-schenfeigel@etu.univ-tours.fr](mailto:simeon.joseph-schenfeigel@etu.univ-tours.fr)  
[youssef.hafidi@etu.univ-tours.fr](mailto:youssef.hafidi@etu.univ-tours.fr)

---

Encadrant

**Claudine Tacquard**  
[claudine.tacquard@univ-tours.fr](mailto:claudine.tacquard@univ-tours.fr)



## Avertissement

Ce document a été rédigé par **Youssef Hafidi** et **Siméon Joseph-Schenfeigel** sus-nommés les auteurs. L'École Polytechnique de l'Université François Rabelais de Tours est représentée par **Claudine Tacquard** susnommé le tuteur académique.

Par l'utilisation de ce modèle de document, l'ensemble des intervenants du projet acceptent les conditions définies ci-après.

Les auteurs reconnaissent assumer l'entière responsabilité du contenu du document ainsi que toutes suites judiciaires qui pourraient en découler du fait du non-respect des lois ou des droits d'auteur.

Les auteurs attestent que les propos du document sont sincères et assument l'entière responsabilité de la véracité des propos.

Les auteurs attestent ne pas s'approprier le travail d'autrui et que le document ne contient aucun plagiat.

Les auteurs attestent que le document ne contient aucun propos diffamatoire ou condamnable devant la loi.

Les auteurs reconnaissent qu'ils ne peuvent diffuser ce document en partie ou en intégralité sous quelque forme que ce soit sans l'accord préalable du tuteur académique et de l'entreprise.

Les auteurs autorisent l'école polytechnique de l'université François Rabelais de Tours à diffuser tout ou partie de ce document, sous quelque forme que ce soit, y compris après transformation en citant la source. Cette diffusion devra se faire gracieusement et être accompagnée du présent avertissement.



# Table des matières

Introduction.....	4
1. Création de la base de données .....	7
1.1 Création du modèle conceptuel de donnée (MCD) .....	7
1.2 Création de la base de données avec l'outil PHPMyAdmin .....	8
2. Site statique .....	11
2.1 Création de la maquette.....	11
2.2 Écriture du code HTML / CSS de manière statique .....	12
3. Formulaires et résultats .....	14
3.1 Écriture du formulaire depuis le code PHP .....	14
3.2 Affichage des résultats .....	15
Conclusion .....	17
Bibliographie .....	19
Comptes-rendus hebdomadaires .....	21



# Introduction

Notre projet vise à nous initier aux outils de gestion de bases de données et à leur utilisation dans le cadre du développement d'applications web. Il consiste en la création d'une page web codée en PHP/SQL. Cette page web permet à l'utilisateur de filtrer différentes spécialités proposées par toutes les écoles Polytech et d'afficher un tableau contenant toutes les formations en fonction des filtres choisis.

Les utilisateurs peuvent filtrer les spécialités en fonction de différents critères, tels que la ville, le type de formation, ou encore le domaine d'études.

Une fois les critères choisis, toutes les formations correspondantes s'affichent dans un tableau où l'utilisateur peut accéder au site Web de chacune d'entre elles.

Le site Web de Polytech propose déjà un outil semblable, mais celui-ci n'est pas dynamique et ne prend pas en compte les paramètres choisis par l'utilisateur.

Cette page web est utile pour les étudiants en cycle préparatoire qui souhaitent trouver la spécialité qui correspond le mieux à leurs envies, mais aussi pour les écoles Polytech qui peuvent utiliser cette page web pour promouvoir leurs filières.

Le projet a été réalisé en suivant plusieurs étapes, depuis la création de la base de données jusqu'à l'affichage des résultats personnalisés sur la page web. Il est composé de deux parties distinctes.

La première partie consiste en une initiation à l'outil de gestion de bases de données MySQL, et à son utilisation dans le cadre du développement d'une application

web en PHP. Cette partie est dédiée à l'acquisition des connaissances de base sur les bases de données et à leur manipulation dans le contexte de l'environnement de développement web.

La deuxième partie du projet consiste en la réalisation d'un site web permettant aux utilisateurs de s'orienter parmi les différentes spécialités proposées par les écoles Polytech. Le site web sera une carte de la France, sur laquelle seront localisés tous les établissements Polytech. L'utilisateur pourra choisir une ou plusieurs villes particulières et accéder à la liste des spécialités proposées dans cet établissement. Il pourra également choisir de filtrer les résultats par domaine d'études et par type de formation. Ce projet permettra ainsi aux étudiants de mieux comprendre l'offre de formation proposée par les différentes écoles Polytech en France, et de faciliter leur orientation.

Pour réaliser ce projet, nous avons suivi différentes étapes. Tout d'abord, nous avons créé la base de données en utilisant le MCD sur Looping. Nous avons ensuite créé une base de données sur PHPMyAdmin et importé le schéma dans la base de données. Puis nous avons récupéré les informations utiles depuis le site Polytech et rempli la base de données avec ces informations. Après, nous avons commencé à développer la page web en créant une maquette et en écrivant le code HTML/CSS dans un fichier PHP de manière statique.

Nous avons par la suite créé un formulaire qui permet à l'utilisateur de filtrer les spécialités en fonction de différents critères. Pour cela, nous avons connecté PHP à la base de données et écrit les requêtes SQL utiles. Nous avons également récupéré les données à partir de ces requêtes pour créer le formulaire. Enfin, nous avons personnalisé les requêtes SQL en fonction des saisies de l'utilisateur et récupéré les données à partir de ces requêtes pour afficher les résultats sur la page web.

Au cours de ce projet, nous avons utilisé différentes technologies telles que PHP, SQL, HTML, CSS, et JavaScript. Nous avons également utilisé des outils tels que PHPMyAdmin pour créer et gérer la base de données, et Sublime Text pour écrire le code HTML, CSS, PHP et JavaScript.

Dans les pages suivantes de ce rapport, nous allons décrire en détail chacune des étapes de ce projet. Nous commencerons par une présentation des outils et technologies utilisés, puis nous décrirons en détail les différentes étapes du développement de l'application. Nous présenterons ensuite les fonctionnalités de l'application ainsi que les résultats obtenus. Nous allons également discuter des différentes décisions que nous avons prises et des difficultés que nous avons rencontrées tout au long de notre avancée.

Enfin, nous concluons en dressant un bilan du projet et en proposant des pistes pour des améliorations futures.

# 1

## Création de la base de données

### 1.1 – Création du modèle conceptuel de donnée (MCD)

La première étape de la conception du site web consistait à créer une base de données. Pour ce faire, nous avons commencé par réaliser un modèle conceptuel de donnée (MCD) depuis le logiciel Looping.

Le MCD est un schéma qui permet de représenter les données d'une entreprise ou d'un projet. Il permet de décrire les entités, les relations entre ces entités, ainsi que les attributs de chacune. Le MCD permet de définir la structure de la base de données (BDD). Il contient les tables, les champs et les relations entre les tables. Ce schéma est donc un document important pour sa création.

Looping est un logiciel de modélisation de données, outil essentiel pour la création d'une base de données. Il permet de créer des MCD, des MPD (Modèle Physique de Données) et des MLD (Modèle Logique de Données), puis de récupérer le code SQL correspondant à ces modèles.

Pour notre modèle conceptuel de données (*annexe 3*), nous avons tout d'abord créé la table « Spécialités », contenant comme clé primaire un identifiant propre à chaque formation. Pour ses attributs, elle contient la ligne « nomS » qui correspond au nom de la formation, ainsi que « lienSpecialité », le lien vers le site Web. Cette table contient aussi une clé étrangère « idEcole », reliant chaque formation à une école à partir de leurs identifiants. Une spécialité ne peut donc être liée qu'à une seule école ; en revanche, une école peut être liée à plusieurs spécialités.

La deuxième table créée est la table « écoles », contenant un identifiant, le nom de l'école, le lien, mais aussi ses positions « top » et « left » par rapport à la

carte de France, permettant de positionner la ville correctement lorsqu'elle sera affichée sur notre page.

La table suivante est la table « domaines », contenant les quinze domaines de formation possibles. Cette table possède un identifiant et un nom, mais aussi une image « imgD » correspondant au logo de chaque domaine.

Pour relier la table des domaines à celles des spécialités, nous avons créé la table de relation « spécialitéDomaine » contenant l'identifiant d'une formation ainsi qu'un identifiant de domaine. Ainsi, chaque formation peut être reliée à un ou plusieurs domaines, et chaque domaine à une ou plusieurs formations.

Enfin, il y a la table des formations « typeFormation » contenant un identifiant, une description, un nom « nomT » et un logo « imgT ». Cette table est liée à celle des spécialités à travers une table de relation « specialiteType », contenant l'identifiant d'une spécialité et d'un type de formation, mais aussi une ligne « infoC », qui permet d'ajouter des informations complémentaires si nécessaire. Malheureusement, nous n'avons pas utilisé cette ligne car nous n'avons pas trouvé d'informations pouvant être nécessaires au fonctionnement de notre projet.

## **1.2 – Création de la base de données avec l'outil PHPMyAdmin**

Une fois le MCD écrit, la deuxième étape était de créer une base de données sur PHPMyAdmin. Pour ce faire, nous avons récupéré le code SQL de notre MCD afin de l'importer dans notre base de données.

PHPMyAdmin est un outil de gestion de base de données MySQL. Il permet de créer, modifier et supprimer des tables, des champs et des enregistrements dans une base de données. PHPMyAdmin est donc un outil essentiel pour la création d'une base de données.

Nous avons ensuite récupéré les informations utiles depuis le site Polytech et avons rempli la BDD avec ces informations. (*Annexe 5*)



Pour éviter la tâche répétitive du remplissage de la base de données, nous avons écrit un script Python qui prend en paramètre tout le texte de la page web, que nous avons copié (contenant les noms des spécialités ainsi que leur ville) et qui extrait chaque information pour chaque spécialité ; nous avons rempli la base de données au fur et à mesure.

Malheureusement, nous n'avons pas pu éviter cette tâche avec les liens pour chaque spécialité. En effet, chaque Polytech a son propre site web (chacun très différent de l'autre) ; il en va de même pour les types de formation (Alternance au Etudiant). Nous avons donc dû écrire ces données une par une dans la base de données.

Enfin, après avoir entré toutes les données, nous avons dû vérifier et corriger la BDD ligne par ligne ; heureusement pour nous, il n'y avait que très peu d'erreurs.

Après avoir terminé la base de données, nous nous sommes rendus compte que le moteur de stockage n'était pas le bon, car nous utilisions un moteur nommé « MyISAM », alors que notre encadrante voulait qu'on utilise « InnoDB ».

InnoDB et MyISAM sont deux moteurs de stockage de bases de données relationnelles MySQL. La principale différence entre les deux moteurs est la façon dont ils gèrent les transactions et les verrous.

InnoDB est un moteur de stockage transactionnel qui prend en charge les transactions ACID (Atomicité, Cohérence, Isolation, Durabilité). Cela signifie que InnoDB garantit que les transactions sont traitées de manière cohérente et fiable, même en cas de panne système ou de problème de réseau. Il utilise des verrous de ligne pour gérer les conflits de lecture et d'écriture entre les transactions simultanées.

MyISAM, en revanche, est un moteur de stockage non transactionnel qui ne prend pas en charge les transactions ACID. Il utilise des verrous de table pour gérer les conflits de lecture et d'écriture entre les transactions simultanées, ce qui peut entraîner des problèmes de performance dans des environnements à forte concurrence.

Pour changer de moteur, nous avons importé le code SQL de notre BDD depuis PHPMyAdmin, puis l'avons modifié manuellement en remplaçant par InnoDB. (*cf Annexe 9*)

En résumé, la création de la base de données pour notre site web a nécessité la réalisation d'un modèle conceptuel de donnée sur Looping, la création d'une base de données sur PHPMyAdmin et l'importation du schéma dans la BDD. Nous avons ensuite récupéré les informations utiles depuis le site Polytech et avons rempli la BDD avec ces informations.

# 2

## Site statique

### 2.1 – Création de la maquette

La deuxième partie du projet consiste à créer une maquette. De nombreux logiciels en ligne, tel que Figma, sont disponibles, mais nous avons décidé d'exploiter le logiciel Paint car celui-ci est plus simple d'utilisation.

La création de la maquette est une étape importante dans le développement de l'application. Elle représente l'organisation et l'interface graphique d'un site web sous forme de schéma. Ce travail est indispensable pour tous projets de création ou de refonte de sites. Cela nous donne une idée précise de l'arborescence et de l'ergonomie du site. La maquette fonctionnelle permet de définir les zones et composants que doit contenir une interface. Il est possible de traduire ce rendu visuel en HTML et CSS. Pour créer la maquette, il est nécessaire de prendre en compte les besoins des utilisateurs et de concevoir une interface qui répond à ces besoins.

Nous avons décidé de placer au centre de la page (*Annexe 2*) une carte de France, sur laquelle chaque école du réseau Polytech est représentée par un petit logo. Chaque petit logo sera positionné en fonction de l'emplacement géographique de l'école, que nous pourrions récupérer depuis la base de données (*cf Annexe 11*). Au chargement de la page, chaque logo sera grisé, mais lorsque l'on cliquera dessus pour sélectionner une école, il deviendra bleu. Le style de cette carte a été inspirée de celle affichée par le site du réseau Polytech.

En haut de la page, nous avons placé deux listes déroulantes, affichant respectivement la liste des domaines et la liste des types de formations. Ces informations seront aussi récupérées à partir de la base de données. Lorsque l'utilisateur passera son curseur sur une liste (*cf Annexe 4*), celle-ci se déroulera et l'utilisateur aura alors

la possibilité de cliquer sur les options de son choix.

Enfin, nous avons décidé de placer, à gauche (*cf Annexe 1*), une liste de tableaux affichant toutes les spécialités pour chaque école, en fonction des paramètres choisis par l'utilisateur. Le style des tableaux a été copié à partir de celui du site de Polytech. Chaque ligne de ce tableau affichera respectivement le nom, le type et les domaines de la formation. De plus, chaque ligne redirigera l'utilisateur vers le lien de la formation lorsqu'il cliquera dessus. En outre, si aucune spécialité ne correspond à la recherche, alors cette partie deviendra invisible, afin de ne pas créer un vide à gauche.

Pour le style de notre page, nous avons décidé de mettre un fond blanc, c'est un design simple mais efficace et qui ne pique pas les yeux de l'utilisateur. Pour la partie de gauche, en revanche, le fond est de couleur grise. Cette couleur permet à l'utilisateur de différencier rapidement les deux parties principales de notre page.

Notre site web n'est constitué que d'une seule page, ainsi l'utilisateur peut accéder directement et de manière dynamique aux informations utiles sans devoir passer par plusieurs pages.

## **2.2 – Écriture du code HTML / CSS de manière statique**

Une fois la maquette créée, il est temps de passer à l'écriture du code HTML/CSS, de manière statique. Cette étape permettra de créer une version de l'application qui ne sera pas encore connectée à la base de données. C'est-à-dire que notre page sera pour le moment indépendante. Nous allons donc écrire manuellement le formulaire et les résultats, mais ils seront par la suite écrits directement par un code PHP.

Pour afficher la carte de France, ainsi que chaque écoles du réseau Polytech, nous avons choisi de les placer de manière absolue (« Position : absolute ; » en CSS). C'est-à-dire que nous les avons placées par rapport à la carte de France, c'est pour cela que nous avons ajouté les lignes « top » et « left » dans notre MCD. Cependant, cette étape nous a posé un problème car les écoles et leurs noms se superposaient si elles étaient trop proches l'une de l'autre. Pour répondre à cela, nous avons choisi de réduire la taille des logos et des titres. Nous avons donc contourné le problème pour le moment. Mais si une nouvelle école décide d'ouvrir juste à côté d'une autre, ce problème ressortira. Pour résoudre ce problème, il faudra ajouter une ligne correspondante à la position du titre de l'école dans le MCD en ajustant correctement les positions pour éviter les problèmes de superposition.

Pour l'affichage des résultats dans la partie de gauche, nous avons décidé d'utiliser des tableaux (« <table> » en HTML). Cette méthode permet de positionner correctement chaque ligne afin qu'elles fassent toutes la même taille et restent alignées.

En résumé, la deuxième étape importante à notre projet était de créer une page web de manière statique afin de créer une première version, de manière indépendante à la base de données, qui ne nécessite plus que du code PHP. Pour ce faire, nous avons d'abord créé une maquette depuis le logiciel Paint afin de le rendu final de notre page Web. Ensuite, nous avons converti cette maquette en un code HTML / CSS.

# 3

## Formulaire et résultats

### 3.1 – Écriture du formulaire depuis le code PHP

La troisième partie du projet consiste à écrire le code PHP afin de lier le formulaire à la base de données.

Tout d'abord, il faut connecter le code PHP à la base de données. Pour cela il est nécessaire de définir les paramètres de connexion à la base de données (cf *Annexe 6*). Ces paramètres incluent le nom d'utilisateur, le mot de passe et le nom de la base de données. Dans notre cas, le nom d'utilisateur est « root » et il n'y a pas de mot de passe. Une fois que les paramètres de connexion sont définis, il est possible d'exécuter des requêtes SQL pour récupérer les données directement à partir du code PHP.

Ensuite, nous récupérons les requêtes MySQL que nous avons écrites plus tôt au brouillon et les écrivons dans notre code PHP. Les requêtes SQL doivent être écrites de manière à récupérer les données de façon efficace et à minimiser le temps de réponse.

Pour pouvoir afficher les informations nécessaires dans le formulaire, le code PHP va envoyer des requêtes SQL à la base puis récupérer les informations utiles. Avec, il va dresser une liste contenant toutes les écoles du réseau ainsi que leurs informations, une liste contenant chaque domaine, puis une liste contenant chaque type de formation. Ensuite, à l'aide de boucle « foreach », il va compléter le formulaire en écrivant ligne par ligne chaque élément input du formulaire à partir des listes qu'il aura créé (cf *Annexe 7*). Ainsi, si un nouveau type de formation ou une nouvelle école est ajoutée / supprimée de la base de données, le formulaire l'affichera pour que l'utilisateur puisse y avoir accès lors de sa recherche.

## 3.2 – Affichage des résultats

Maintenant que le formulaire est créé et lié à la base de données depuis le PHP, il faut récupérer les filtres sélectionnés par l'utilisateur depuis ce formulaire.

Pour cela, nous allons utiliser le langage JavaScript, plus particulièrement la bibliothèque jQuery.

jQuery est une bibliothèque JavaScript populaire qui simplifie la manipulation des documents HTML, la gestion des événements, les animations et les requêtes AJAX. Il permet aux développeurs de créer des applications web interactives et dynamiques avec moins de code JavaScript. jQuery fournit également une grande quantité de plugins et d'extensions pour faciliter la création de fonctionnalités avancées.

Nous devons donc créer un fichier JavaScript et le lier au fichier HTML en utilisant une balise « `<script>` ». Dedans, nous écrivons d'abord un script qui récupère tous les filtres cochés par l'utilisateur en accédant au formulaire depuis le DOM. Ensuite, ces filtres seront envoyés, en utilisant la technologie Ajax inclus dans JQuery, à un nouveau fichier PHP.

Ce nouveau fichier va récupérer ces filtres et s'en servir pour personnaliser des requêtes MySQL (*cf Annexe 10*) qu'il enverra à la base de données. Une fois envoyé, le code PHP réceptionnera tous les résultats correspondant à la recherche de l'utilisateur puis les organisera par ville en dressant une liste correspondant à chacune.

Si l'utilisateur ne choisit aucune école, alors elles seront toutes sélectionnées, de même pour les types de formations et pour les domaines. Ainsi, il y aura toujours un résultat pertinent qui s'affichera, même si l'utilisateur ne choisit aucun type de formation par exemple.

A partir de ces listes, des tables HTML vont être créées (une pour chaque école) et seront envoyées au code JavaScript (*cf Annexe 12*) comme réponse. Le code JavaScript va récupérer toutes ces tables, puis en utilisant la bibliothèque JQuery, il va accéder à la partie gauche de la page et y insérer le nouveau code HTML. (*cf annexe 8*)

Nous avons choisi d'utiliser le langage JavaScript afin de pouvoir afficher les

résultats dès que l'utilisateur choisi de cliquer sur un filtre. Ainsi, l'affichage devient dynamique et ne nécessite pas d'ouvrir une nouvelle page ou d'actualiser celle déjà ouverte.

A partir du moment où l'utilisateur cliquera sur un élément input du formulaire, des résultats s'afficheront et l'utilisateur pourra alors choisir une spécialité qui lui convient.





# Conclusion

En conclusion, le projet de création d'un formulaire en PHP est un excellent moyen d'apprendre les bases de la programmation web. Le projet est divisé en quatre parties principales, chacune d'entre elles étant importante pour la création du formulaire. L'utilisation des différents langages et technologies s'est naturellement imposée à nous. Nous en connaissions déjà la plupart.

La première partie du projet consiste à créer la base de données. C'est l'endroit où les données saisies par les utilisateurs seront stockées. Il est important qu'elle soit bien structurée pour faciliter la récupération des données. Une des difficultés majeures fut d'aller sur chaque site Polytech de chaque ville car certains liens n'étaient pas toujours fonctionnels. Grâce à notre encadrante, nous avons également découvert le moteur de stockage transactionnel InnoDB.

La deuxième partie du projet consiste à créer le formulaire. Le formulaire est l'interface utilisateur qui leur permet de saisir les données. Il est important de créer un formulaire facile à utiliser et à comprendre pour les utilisateurs.

La troisième partie du projet consiste à connecter le PHP à la base de données. Pour cela, il est nécessaire d'écrire des requêtes SQL utiles. Ces requêtes permettront de récupérer les données à partir desquelles le formulaire sera créé.

La quatrième partie du projet consiste à écrire des requêtes qui seront personnalisées par le PHP. Pour cela, il est nécessaire de créer un deuxième fichier PHP qui communiquera avec le JavaScript. Le code JavaScript récupérera les saisies du formulaire et les enverra au nouveau fichier PHP. Les requêtes SQL seront personnalisées à partir de ces saisies. Les données seront récupérées de la base de

données à partir de ces requêtes. Le PHP renverra au JavaScript un tableau HTML qui contiendra ces données. Le code JavaScript récupérera ce tableau et l'affichera sur la page. Nous avons alors découvert JQuery, bibliothèque que nous avons trouvée très simple et utile. C'est une méthode qui affiche les résultats de manière asynchrone et plus rapide, en une seule ligne.

En suivant ces quatre parties, il est possible de créer un formulaire en PHP qui permettra aux utilisateurs de sélectionner des données et d'obtenir un affichage de toutes les spécialités correspondantes aux besoins de l'utilisateur. Le projet peut être utilisé comme point de départ pour des projets plus avancés et pourrait être ajouté au site du réseau Polytech afin de permettre à chaque étudiant de trouver la formation qui lui convient le mieux.



# Bibliographie

Pour télécharger et apprendre à utiliser le logiciel Wamp :

<https://www.wampserver.com/>

La page du site du réseau de Polytech qui nous a permis récupérer toutes les spécialités :

<https://www.polytech-reseau.org/rechercher-une-formation-ingenieur-polytech/>

La page Polytech pour récupérer toutes les écoles :

<https://www.polytech-reseau.org/nos-ecoles-ingenieurs/>

La page Polytech pour récupérer tous les domaines de formations :

<https://www.polytech-reseau.org/12-domaines-de-formation/>

Pour apprendre les bases du langage HTML et CSS :

<https://openclassrooms.com/fr/courses/1603881-creez-votre-site-web-avec-html5-et-css3>

Pour apprendre les bases en PHP et SQL (et du logiciel PHPMyAdmin) :

<https://openclassrooms.com/fr/courses/918836-concevez-votre-site-web-avec-php-et-mysql>

Pour apprendre les bases du JavaScript :

<https://openclassrooms.com/fr/courses/5543061-ecrivez-du-javascript-pour-le-web>

Pour apprendre à se servir du Framework JQuery :

<https://openclassrooms.com/fr/courses/3504441-introduction-a-jquery>



# Comptes rendus hebdomadaires

## Compte rendu n°1

Nous avons commencé en nous familiarisant avec les outils dont nous aurons besoin lors de notre projet (WAMP, Looping, PHPMyAdmin), et appris les bases du langage de programmation PHP et du langage SQL.

## Compte rendu n°2

Nous avons commencé la base de données avec Looping et nous nous sommes mis d'accord sur ce à quoi ressemblera le site web une fois créé. Dans la semaine numéro 3 nous comptons commencer le code HTML et CSS.

## Compte rendu n°3

La partie HTML/CSS a été complétée. Le site affiche maintenant la carte de France avec le réseau Polytech, ainsi qu'une liste déroulante contenant les types et domaines de formation.

Les informations affichées ne proviennent pas d'une base de données pour le moment, mais sont écrites manuellement depuis le code HTML

Il est possible de sélectionner une ou plusieurs écoles en particulier.

#### Compte rendu n°4

Cette semaine, nous avons continué de créer le MCD (modèle conceptuel de donnée) et commencer à écrire le code PHP.

#### Compte rendu n°5

Cette semaine, nous avons fini le schéma sur Looping, créer et rempli la base de données sur l'outil PHPMyAdmin.

Ensuite, nous avons lié le code PHP et la base de données pour que le site puisse récupérer les informations utiles et les afficher dans le formulaire.

*(Annexe 5)*

#### Compte rendu n°6

Comme vous pouvez voir dans l'image, nous avons bien réussi à remplacer MyISAM par InnoDB. On a aussi listé les requêtes SQL au brouillon qui pourraient être utiles pour le projet.

*(Annexe 3)*

#### Compte rendu n°7

Cette semaine, nous avons écrit les requêtes SQL dans le code PHP pour qu'il puisse accéder aux différentes spécialités et n'afficher que celles qui nous intéressent.

Comme vous pouvez le voir sur l'image, en choisissant Polytech Tours ou Paris, en formation étudiante, dans le domaine de l'informatique ou de la mécanique, on obtient ce résultat.

En choisissant un ou plusieurs domaines, les requêtes SQL vont sélectionner toutes les spécialités qui contiennent au moins un de ces domaines.

Pareil pour le type de formation.

#### Compte rendu n°8

Nous avons modifié les requêtes SQL pour que toutes les spécialités d'une ville s'affichent lorsque l'on clique dessus.

Le projet est maintenant terminé.

#### Compte rendu n°9

Cette semaine, nous avons commencé le compte rendu.

(Le projet avait été complété et validé par notre encadrante)

# Introductions aux bases de données avec développement d'une application web en PHP

## Résumé

Nous avons créé une page Web permettant à l'utilisateur de rechercher une formation Polytech qui correspondra le plus à ses envies.

Nous avons ainsi approfondi nos connaissances en programmation Web et au développement aux base de données.

## Mots-clés

PHP, base de données, SQL, InnoDB, Looping, HTML, CSS, JavaScript, JQuery, Utilisateur, Polytech, requêtes

## Abstract

We created a web page allowing the user to search a Polytech speciality that will suit his wishes.

So, we have improve our knowledge of web programming and database development.

## Keywords

PHP, database, SQL, InnoDB, Looping, HTML, CSS, JavaScript, JQuery, User, Polytech, request



Encadrant académique <i>Claudine Tacquard</i>	Étudiants <i>Siméon Joseph-Schenfeigel</i> <i>Youssef Hafidi</i>
--	--

## Annexe 1



Type de formation ▼

Domaine de formation ▼




### Paris-Saclay

Spécialités	Formation	Domaine
Informatique et Ingénierie Mathématique - FISE	FISE	 
Matériaux : mécanique et énergie - FISE	FISE	  

### Tours


Spécialités	Formation	Domaine
Informatique	FISE	 
Mécanique et conception des système	FISE	 

## Annexe 2

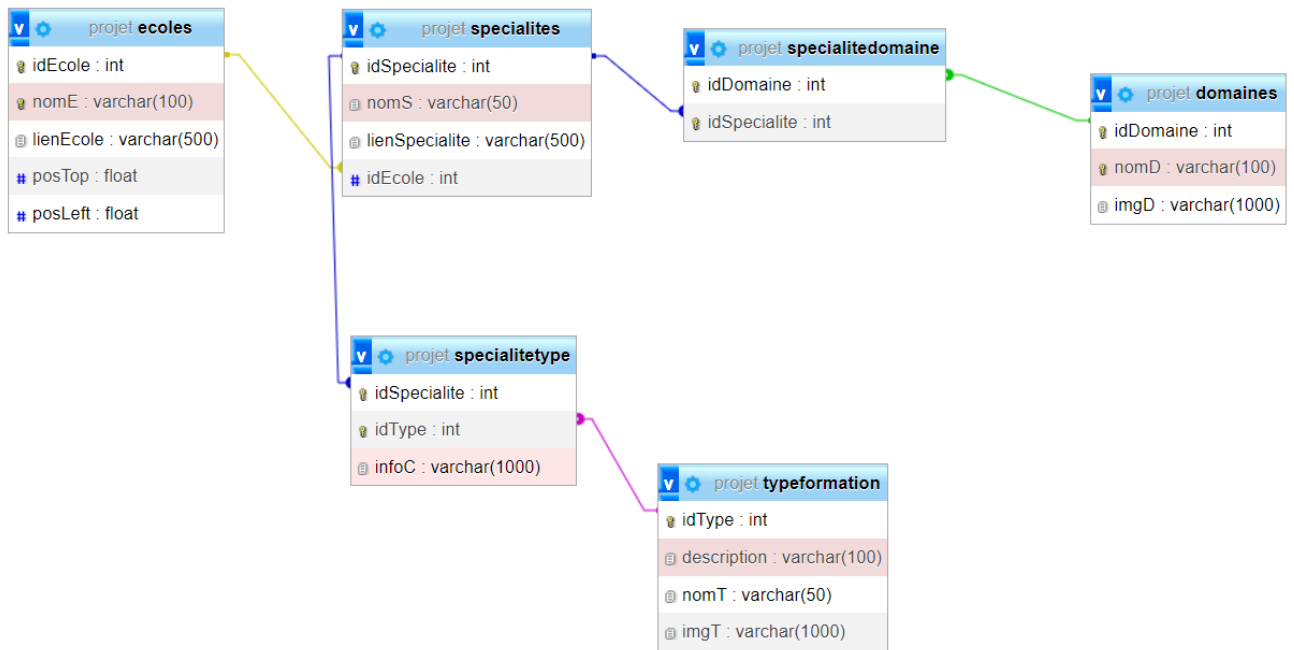


Type de formation ▼

Domaine de formation ▼



## Annexe 3



## Annexe 4

Type de formation ▼

Domaine de formation ▼

- ☐ Tout sélectionner
- ☐ Eau, environnement, aménagement
- ☐ Électronique et systèmes numériques
- ☐ Énergétique, génie des procédés
- ☐ Génie biologique et alimentaire
- ☐ Génie biomédical, instrumentation
- ☐ Génie civil
- ☐ Génie industriel
- ☒ Informatique
- ☐ Matériaux
- ☐ Mathématiques appliquées et modélisation
- ☐ Mécanique
- ☐ Systèmes électroniques

### Lyon

Spécialités	Formation	Domaine
Informatique - FISA	FISA	Informatique
Informatique - FISE	FISE	Informatique

### Nancy

Spécialités	Formation	Domaine
Informatique, Automatique, Robotique, Réseaux	FISE	Informatique, Automatique
Informatique, Automatique, Robotique, Réseaux - FISA	FISA	Informatique, Automatique

### Tours

Spécialités	Formation	Domaine
Informatique	FISE	Informatique
Informatique et systèmes intelligents embarqués	FISA	Informatique, Automatique

## Annexe 5

idSpecialite	nomS	lienSpecialite	idEcole
1	Bâtiment : exploitation-maintenance et sécurité	<a href="http://www.polytech-angers.fr/fr/trouver-une-forma...">http://www.polytech-angers.fr/fr/trouver-une-forma...</a>	1
2	Génie biologique et santé	<a href="http://www.polytech-angers.fr/fr/trouver-une-forma...">http://www.polytech-angers.fr/fr/trouver-une-forma...</a>	1
3	Qualité, innovation, fiabilité - FISA	<a href="http://www.polytech-angers.fr/fr/trouver-une-forma...">http://www.polytech-angers.fr/fr/trouver-une-forma...</a>	1
4	Systèmes automatisés et génie informatique	<a href="http://www.polytech-angers.fr/fr/trouver-une-forma...">http://www.polytech-angers.fr/fr/trouver-une-forma...</a>	1
5	Bâtiment écoconstruction énergie - FISA	<a href="https://www.polytech.univ-smb.fr/formation/batimen...">https://www.polytech.univ-smb.fr/formation/batimen...</a>	2
6	Écologie industrielle et territoriale	<a href="https://www.polytech.univ-smb.fr/formation/ecologi...">https://www.polytech.univ-smb.fr/formation/ecologi...</a>	2
7	Informatique, Données, Usages	<a href="https://www.polytech.univ-smb.fr/formation/ingenie...">https://www.polytech.univ-smb.fr/formation/ingenie...</a>	2
8	Mécanique Mécatronique Matériaux composites - FISA	<a href="https://www.polytech.univ-smb.fr/formation/mecaniq...">https://www.polytech.univ-smb.fr/formation/mecaniq...</a>	2
9	Systèmes numériques - Instrumentation	<a href="https://www.polytech.univ-smb.fr/formation/systeme...">https://www.polytech.univ-smb.fr/formation/systeme...</a>	2
10	Génie biologique	<a href="https://www.uca.fr/formation/nos-formations/par-uf...">https://www.uca.fr/formation/nos-formations/par-uf...</a>	3

## Annexe 6

```
function executeC($ligne) {
    $db = new PDO(
        'mysql:host=localhost;dbname=projet;charset=utf8',
        'root', '');
    $commande = $db->prepare($ligne);
    $commande->execute();
    return $commande->fetchAll();
}

$lDom = executeC('Select * from domaines');
$lEco = executeC('Select * from ecoles Order By nomE');
$lType = executeC('Select * from typeformation');
```

## Annexe 7

```
<div id="zone">
  

  <?php foreach ($lEco as $i) : ?>
    <input type="checkbox" id="ecole<?php echo $i[0] ?>" name="ecole<?php echo $i[0] ?>" <?php testCheck("ecole".$i[0]) ?>>
    <p class="positionVille" id="lienE<?php echo $i[0] ?>" style="Left: <?php echo $i[4] ?>%; top: <?php echo $i[3] ?>%;">
    <label for="ecole<?php echo $i[0] ?>"></label></p>

    <h2 class="nomVille" style="top:<?php echo $i[3] ?>%; Left:<?php echo $i[4] ?>%;"><?php echo $i[1] ?></h2>
  <?php endforeach ?>
</div>
```

## Annexe 8

```
function affResult() {
    $.post('resultats.php', $('form').serialize(), function(data) {
        $('#resultats').html(data).css('width', (data=== "") ? '0%' : '35%');
        $('form').css('width', (data=== "") ? '100%' : '65%');
    })
    .fail(function(xhr, status, error) {
        alert('Une erreur s\'est produite');
    });
};
```

## Annexe 9

```
DROP TABLE IF EXISTS `domaines`;
CREATE TABLE IF NOT EXISTS `domaines` (
  `idDomaine` int NOT NULL AUTO_INCREMENT,
  `nomD` varchar(100) DEFAULT NULL,
  `imgD` varchar(1000) NOT NULL,
  PRIMARY KEY (`idDomaine`),
  UNIQUE KEY `nomD` (`nomD`)
) ENGINE=InnoDB AUTO_INCREMENT=14 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

## Annexe 10

```
foreach($lVilleS as $ville) {  
    $listeSpe = executeC(  
        "Select DISTINCT s.idSpecialite, nomS, lienSpecialite from specialites as s  
  
        JOIN Specialitetype as st ON st.idSpecialite = s.idSpecialite  
        JOIN typeformation as tf ON tf.idtype = st.idtype and (".substr($ligneType,0,-3)."  
  
        JOIN specialitedomaine as sd ON sd.idspecialite = s.idspecialite  
        JOIN domaines as d ON d.idDomaine = sd.idDomaine and (".substr($ligneDomaine,0,-3)  
  
        Where s.idEcole = ".$ville[0]."  
        Order By nomS
```

## Annexe 11

```
<p class="positionVille" id="lienE1" style="left: 28%; top: 32.8%;">  
<label for="ecole1"></label></p>
```

## Annexe 12

```
echo "  
<table>  
  <thead><tr>  
    <th>Spécialités</th><th>Formation</th><th>Domaine</th>  
  </tr></thead>  
  <tbody>";  
  
foreach($listeSpe as $spe) {  
  $listeDomSpe = executeC(  
    "Select d.idDomaine, d.nomD, d.imgD  
    from domaines as d  
    Join specialiteDomaine as sd On sd.idDomaine = d.idDomaine  
    Where sd.idSpecialite = ".$spe[0]);  
  
  $listeTypeSpe = executeC(  
    "Select t.idType, t.nomT, t.imgT, st.infoC, description  
    from typeformation as t  
    Join specialitetype as st On st.idType = t.idType  
    Where st.idSpecialite = ".$spe[0]);  
  
  echo "<tr class='resultat'>  
    <td>  
      <a target='blank' href='\"'. $spe[2].\"' class='nomSpe'>\". $spe[1].\"</a>  
    </td>  
    <td>";
```