

# **Développement d'un site web de gestion de fiches académiques des étudiants de la licence PRO DASI**

## Sommaire

1.	Phase de présentation.....	3
1.1.	Présentation de l'IUT et de sa formation .....	3
1.2.	Contexte .....	3
1.3.	Le sujet et ses contraintes .....	3
2.	Phase de conception .....	4
2.1.	Répartition des tâches.....	4
2.2.	Développement.....	5
2.2.1.	Base de données.....	5
2.2.2.	Code.....	5
2.2.3.	Tests.....	6
2.3.	Outils utilisés .....	11
3.	Fonctionnement de l'application .....	12
3.1.	Connexion et inscription .....	12
3.2.	Profil .....	12
3.3.	Création, modification et suppression de fiche .....	12
3.4.	Barre de navigation .....	13
4.	Conclusion .....	13
4.1.	Difficultés rencontrées .....	13
4.2.	Ce que le projet nous a apporté.....	14

# 1. Phase de présentation

## 1.1.Présentation de l'IUT et de sa formation

L'IUT ou l'Institut Universitaire de Technologie du Havre est un établissement d'enseignement supérieur qui propose à aux étudiants poursuivant leurs études dans celui-ci d'obtenir un DUT (Diplôme Universitaire Technologique). Lors de l'obtention de ce diplôme, chaque étudiant peut choisir de continuer vers une troisième année de licence informatique, d'entrer dans le monde du travail ou de s'orienter vers une licence professionnelle. L'IUT propose divers départements dont le département informatique dans lequel se trouve la formation licence professionnelle en Développement et Administration de Sites Internet (DASI).

## 1.2.Contexte

Dans le cadre de la licence professionnelle DASI, nous avons été amenés à réaliser un projet en php proposé par M. Mongetro Goint. Il a pour but de nous évaluer et de voir si nous avons acquis les compétences requises dans le développement en php.

Nous avons choisi un nom de groupe pour ce projet. Nous avons choisi Altima car le nom regroupe la première consonance des prénoms des membres du groupe.

## 1.3.Le sujet et ses contraintes

Le sujet proposé est divisé en deux grandes parties. Une partie travaux pratiques et une partie rédaction d'un rapport.

### 1.3.1. Le TP

Dans ce TP, nous souhaitons mettre en place un site web permettant aux étudiants de la licence professionnelle DASI, de gérer des fiches personnelles concernant leur parcours académique. Toutes les données doivent être stockées dans une base de données MySql.

Le site doit comporter plusieurs critères :

- Une partie inscription : les étudiants doivent pouvoir s'inscrire sur le site avec un nom, un prénom, une adresse mail, un mot de passe, un sexe, une date de naissance et une photo de profil.

Contrainte : deux étudiants ne peuvent pas avoir la même adresse mail.

- Une partie connexion : si les étudiants sont inscrits, ils doivent pouvoir se connecter via un formulaire de connexion leur demandant leur adresse mail et leur mot de passe. Quand les étudiants sont connectés, ils doivent voir leur photo de profil apparaître sur leur page de profil.
- Une partie fiches académiques : un étudiant pourra lister et enregistrer toutes ses fiches académiques dans un tableau. Pour chaque ligne du tableau, nous aurons le nom, le prénom, l'identifiant de la fiche, le nom de l'établissement, le pays, le nom de la formation, le niveau qui doit être sous forme de nombre entier, la date de début et la date de fin. Toutes les fiches devront être modifiables et supprimables. L'utilisateur doit pouvoir effectuer des recherches dans la liste en fonction de critères.

### **1.3.2. Le rapport**

Nous devons rédiger un rapport présentant notre travail :

- La conception de l'application
- Le fonctionnement de l'application
- Des résultats de tests (avec des captures d'écrans)

## **2. Phase de conception**

### **2.1.Répartition des tâches**

La base de données a été réalisée à trois. Thibault a réalisé le schéma en fonction de ce qu'indiquaient Alexi et Mattéo et Alexi s'est occupé de concevoir la base de données sur phpmyadmin grâce aux données du schéma réalisé par Thibault. Alexi a généré le fichier sql permettant la création des tables qui a été transmis à ses collègues.

Chacun avait des tâches précises à réaliser. Mattéo s'est occupé de la gestion des membres contenant les pages d'inscription et de profil tandis qu'Alexi et Thibault se sont occupés de la gestion des fiches allant de la création à la suppression, en passant par le listing et la modification.

Alexi a aussi réalisé le fichier css et la page d'accueil du site. Thibault et Alexi se sont occupés d'optimiser le code en incluant des bout de codes étant rangés dans des fichiers différents et en les appelant avec la fonction include().

## 2.2.Développement

### 2.2.1. Base de données

Notre base de données se compose de deux tables. Une table appelée “membre” et une table appelée “fiche”.

La table membre est composée des attributs email\_membre, nom\_membre, prénom\_membre, mdp\_membre, sexe, date\_naissance\_membre, photo et role\_membre.

La table fiche est quant à elle composée des attributs id\_fiche, email\_fiche, etabli\_fiche, pays\_fiche, libelle\_fiche, niveau, date\_deb\_fiche et date\_fin\_fiche.

### 2.2.2. Code

Nous avons d’abord fait une page d’accueil et une page de connexion permettant la connexion d’utilisateurs dont les identifiants n’étaient pas enregistrés dans une base de données. Nous avons ensuite conçu la base de données et avons testé la connexion entre le site et phpmyadmin, puis la connexion d’utilisateurs à notre site. Enfin nous avons récupéré l’email de l’utilisateur courant avec la variable \$\_SESSION.

Ensuite, nous avons créé des fiches directement sur phpmyadmin et nous avons cherché à les lister. Nous récupérions toutes les fiches et les affichions dans un tableau. Puis nous avons décidé de rajouter deux colonnes au tableau pour ajouter le bouton de modification et celui de suppression que nous n’affichions seulement lorsque l’email du propriétaire de la fiche était le même que celui de l’utilisateur courant. Mais, ne sachant trop pourquoi, les fiches ne s’affichaient pas dans l’ordre, alors nous avons demandé à la requête de retourner les fiches triées par ordre croissant de leur identifiant avec la commande ORDER BY.

Une fois que l’affichage fonctionnait, nous avons réalisé la page de suppression. Comme dit précédemment, la suppression s’est faite par l’ajout d’un bouton de suppression. Ce bouton, en cliquant dessus, envoie dans une page “suppression\_fiche.php” l’identifiant de la fiche à supprimer.

Nous appelons une requête qui supprime donc la fiche identifiée par le numéro fournie par le bouton de suppression.

Après cela, nous avons ajouté les pages de création et de modification en partant du principe que le formulaire serait le même. Pour la création, nous avons laissé tous les champs de saisie vides. Nous avons réalisés plusieurs tests afin de peaufiner les vérifications lors de

l'envoi des données dans la table des fiches. Par exemple, nous avons ajouter une condition vérifiant que la date de début de la formation n'était pas fixée à une date postérieure à celle de fin. Quant à la modification, comme pour la suppression, nous avons fait passer l'identifiant de la fiche via un bouton de modification. En obtenant l'identifiant de la fiche, nous exécutions une requête retournant toutes les informations concernant la fiche portant l'identifiant renvoyé par le bouton puis nous avons affiché les valeurs de la fiche dans les champs de saisie de la page.

Pendant qu'Alexi et Thibault s'occupaient de la gestion des fiches, Mattéo réalisait les pages permettant la gestion des membres. Sur la page d'inscription, nous récupérons toutes les informations que l'utilisateur saisit et il n'y a pas de vérification, si ce n'est de vérifier que le texte saisit pour l'email correspond bien à une adresse mail valide avec la fonction `filter_var()` prenant en premier paramètre le texte saisit par l'utilisateur et en deuxième paramètre `FILTER_VALIDATE_EMAIL` retournant un booléen à la valeur "true" (vrai) lorsque la chaîne de caractère correspond à une adresse mail valide. L'utilisateur ne peut modifier sa photo de profil qu'une fois qu'il est connecté et qu'il modifie sa page de profil.

Pour la page de modification du profil, nous récupérons l'email de l'utilisateur et nous affichons toutes les informations le concernant. L'utilisateur peut modifier à sa guise ses informations personnelles et lorsqu'il modifie par exemple son nom ou son prénom, alors ceux-ci sont mis à jour dans la base de données et les fiches qu'il possède sont aussi mises à jour. Cela fonctionne pour chaque donnée personnelle que ce soit le sexe, la date de naissance ou encore le mot de passe. Mais lorsqu'il change son email, alors en plus d'être mis à jour dans la base de données et de mettre à jour les fiches enregistrées, cela change aussi l'adresse mail de la variable `$_SESSION`. Enfin, lorsque le mot de passe est modifié, il est mis à jour et crypté une nouvelle fois dans la base.

### 2.2.3. Tests

Lors de la réalisation du site, nous avons dû réaliser des tests afin de vérifier que l'exécution du code se déroulait de façon souhaitée. Par exemple, pour créer une fiche, nous avons essayé d'ajouter une fiche lorsque qu'un champ ou que tous les champs restaient vides.

# Création d'une fiche


Nom de l'établissement :

Pays :


Nom de la formation :

Niveau :

Date de début :

Date de fin :

Annuler

Enregistrer la fiche

Et nous avons remarqué que l'ajout de champs vides ne posaient aucun problème.

Donc nous vérifions que les champs ne sont pas null avant de créer une fiche. C'est pourquoi une redirection sur la page de création des fiches et une remise à zéro des champs est faite si le formulaire n'est pas saisi de façon correcte. Nous avons regardé ce qu'il se passait lorsque nous essayions de fixer une date de début après la date de fin et pour le niveau, nous avons essayé de mettre un nombre égal à 0, supérieur à 0, une valeur alphabétique, une chaîne de caractères ou des caractères spéciaux ou encore de laisser le champ de saisie vide. Avant de mettre la vérification du champ vide, il était possible de mettre une valeur nulle pour le niveau.

☐ Éditer ☐ Copier ☐ Supprimer 41 admin.admin@admin.fr 0 0000-00-00 0000-00-00

# Création d'une fiche

Nom de l'établissement :

Pays :

Nom de la formation :

Niveau :

Date de début :



Date de fin :



L'ajout ne posait encore une fois aucun soucis. Mais avec les tests, nous avons vu qu'une réponse vide correspondait à la même réponse qu'un caractère ou une chaîne, étant une valeur égale à 0.

☐ Éditer ☐ Copier ☐ Supprimer 42 admin.admin@admin.fr test test test 0 2021-01-23 2021-01-11

Nous avons donc décidé d'attribuer un niveau fixé à 1 dans le cas où le niveau était vide, qu'il ne s'agissait pas d'un nombre ou qu'il était égal à 0 et nous avons donc retiré la condition vérifiant que le champ était vide. Nous vérifions aussi que les dates sont fixées dans le bon ordre.

Nous avons essayé de voir si en supprimant ou modifiant une fiche, toutes les autres fiches créées par l'utilisateur courant n'étaient pas affectées par l'exécution. Sur les captures d'écran des tests, nous



allons voir ce qu'il se passe lorsque nous modifions la fiche 48 en remplaçant le niveau fixé à 1 par 7.

Nom de l'établissement :

Pays :

Nom de la formation :

Niveau :

Date de début :

Date de fin :

Nom de l'établissement :

Pays :

Nom de la formation :

Niveau :

Date de début :

Date de fin :

Identifiant de la fiche	Email de l'étudiant	Nom de l'établissement	Pays	Libellé de la formation	Niveau	Date de début	Date de fin	Modification	Suppression
44	admin.admin@admin.fr	IUT LH	France	IUT Info	1	2021-01-04	2021-01-22		
47	fouchet.thibault76@gmail.com	IUT LH	France	Licence Pro DASI	3	2021-01-18	2021-02-28		
48	ft182806@univ-lehavre.fr	test	test	test	1	2021-01-19	2021-01-30	<input type="button" value="Modifier"/>	<input type="button" value="Supprimer"/>
49	ft182806@univ-lehavre.fr	test2	test2	test2	1	2021-01-21	2021-01-29	<input type="button" value="Modifier"/>	<input type="button" value="Supprimer"/>
50	ft182806@univ-lehavre.fr	test3	test3	test3	3	2021-01-20	2021-01-21	<input type="button" value="Modifier"/>	<input type="button" value="Supprimer"/>

Identifiant de la fiche	Email de l'étudiant	Nom de l'établissement	Pays	Libellé de la formation	Niveau	Date de début	Date de fin	Modification	Suppression
44	admin.admin@admin.fr	IUT LH	France	IUT Info	1	2021-01-04	2021-01-22		
47	fouchet.thibault76@gmail.com	IUT LH	France	Licence Pro DASI	3	2021-01-18	2021-02-28		
48	ft182806@univ-lehavre.fr	test	test	test	7	2021-01-19	2021-01-30	<input type="button" value="Modifier"/>	<input type="button" value="Supprimer"/>
49	ft182806@univ-lehavre.fr	test2	test2	test2	1	2021-01-21	2021-01-29	<input type="button" value="Modifier"/>	<input type="button" value="Supprimer"/>
50	ft182806@univ-lehavre.fr	test3	test3	test3	3	2021-01-20	2021-01-21	<input type="button" value="Modifier"/>	<input type="button" value="Supprimer"/>

Maintenant que nous avons vu ce qu'il se passait lorsque nous modifiions la fiche 48, nous allons voir si en la supprimant, les autres fiches sont supprimées.

Identifiant de la fiche	Email de l'étudiant	Nom de l'établissement	Pays	Libellé de la formation	Niveau	Date de début	Date de fin	Modification	Suppression
44	admin.admin@admin.fr	IUT LH	France	IUT Info	1	2021-01-04	2021-01-22		
47	fouchet.thibault76@gmail.com	IUT LH	France	Licence Pro DASI	3	2021-01-18	2021-02-28		
49	ft182806@univ-lehavre.fr	test2	test2	test2	1	2021-01-21	2021-01-29	Modifier	Supprimer
50	ft182806@univ-lehavre.fr	test3	test3	test3	3	2021-01-20	2021-01-21	Modifier	Supprimer

Aussi, nous avons voulu voir si en modifiant une adresse mail ou des données telles que le nom ou le prénom sur la page de profil, les fiches se mettaient bien à jour. Nous allons voir si les données des fiches sont mises à jours lorsque nous échangeons le nom et le prénom d'un utilisateur et que nous modifions son adresse mail et son sexe.

ACCUEIL
PROFIL
LISTE
CREATION DE FICHE
DECONNEXION

Afficher uniquement vos fiches

Nom: 
Prénom: 
Sexe: 
Envoyer

Nom	Prénom	Identifiant de la fiche	Nom de l'établissement	Pays	Libellé de la formation	Niveau	Date de début	Date de fin	Modification	Suppression
Hugo	Victor	1	test1	test1	test1	1	2021-01-04	2021-01-29	Modifier	Supprimer
Hugo	Victor	2	test2	test2	test2	2	2020-12-28	2021-02-07	Modifier	Supprimer
Hugo	Victor	3	test3	test3	test3	3	2021-01-12	2021-01-16	Modifier	Supprimer

© Altima 2021  
Alexi Debonne - Thibault Fouchet - Mattéo Daguenet

Modifier votre profil

Email : hugo@victor.fr

Nom : Victor

Prénom : Hugo

Sexe : Femme

Date de Naissance : 2021-01-31



Modifier votre profil

Email : victor@hugo.fr

Nom : Hugo

Prénom : Victor

Sexe : Homme

Date de Naissance : 2021-01-31



Afficher uniquement vos fiches

Nom :  Prénom :  Sexe :  Envoyer

Nom	Prénom	Identifiant de la fiche	Nom de l'établissement	Pays	Libellé de la formation	Niveau	Date de début	Date de fin	Modification	Suppression
Victor	Hugo	1	test1	test1	test1	1	2021-01-04	2021-01-29	Modifier	Supprimer
Victor	Hugo	2	test2	test2	test2	2	2020-12-28	2021-02-07	Modifier	Supprimer
Victor	Hugo	3	test3	test3	test3	3	2021-01-12	2021-01-16	Modifier	Supprimer

Nous pouvons remarquer que les fiches sont mises à jour.

Enfin, nous avons vérifié que lorsque le mot de passe est modifié, il est bien reencrypté dans la base, ce qui se passe bien.

## 2.3. Outils utilisés

Concernant la réalisation du code, chacun était libre d'utiliser l'outil qu'il souhaitait et sur lequel il se sentait le plus à l'aise car nous partons du

principe qu'il est handicapant d'imposer un outil aux autres développeurs car s'il s'agit d'un outil qu'ils n'ont pas utilisé, cela prendrait du temps pour former chaque développeur. C'est pourquoi Mattéo a utilisé Visual Studio Code et Alexi et Thibault ont développé sur l'éditeur de code SublimeText.

Pour communiquer, nous avons utilisé l'outil Discord qui permet de créer des serveurs de discussion et des salons dédiés à chaque tâches. Par exemple, nous avons un salon regroupant les captures d'écran des tests, un salon pour la base de données ou encore un salon regroupant les codes du projet.

Enfin, pour la rédaction du rapport, nous avons utilisé Google Drive permettant de travailler à plusieurs sur des documents. Chacun a donc pu rédiger sa partie en ne pénalisant pas l'avancée des autres membres du groupe. De plus, en utilisant Discord, nous pouvions demander de l'aide concernant des idées à ajouter dans la rédaction ou de l'aide pour de la relecture en temps réel, ce qui était une fois de plus un gain de temps. Discord dispose aussi d'une fonction de partage d'écran qu'il est possible de détacher de l'outil Discord et d'épingler au premier plan. Il est donc possible de rédiger le rapport tout en suivant le partage d'écran d'un des membres du groupe en lecteur réduit.

## **3. Fonctionnement de l'application**

### **3.1. Connexion et inscription**

Le site internet est constitué d'une page de connexion en tant que première page. L'utilisateur devra se connecter avec son email et son mot de passe ou s'inscrire s'il n'a pas de compte. Les informations à rentrer lors de l'inscription seront stockées dans la base de données et serviront pour le profil et la connexion en vérifiant que l'email et le mot de passe correspondent bien.

### **3.2. Profil**

La page de profil montre toutes les informations de l'utilisateur ainsi que sa photo de profil. Le profil est généré selon les informations entrées lors de l'inscription de l'utilisateur qui sont créées et stockées dans la base de données. Les données seront donc affichées sur la page profil selon la personne connectée.

### **3.3. Création, modification et suppression de fiche**

Pour la création de fiche nous utilisons un formulaire. Les informations que l'utilisateur va entrer seront créées et stockées dans la base de données. Nous récupérerons ces données pour ensuite les afficher dans la page de liste

des fiches académiques.

Pour modifier sa fiche académique, l'utilisateur devra cliquer sur le bouton modifier, il devra ensuite modifier les données du formulaire comme il le souhaite, en cliquant sur le bouton "Enregistrer la fiche", les données dans la base de données seront donc modifiées et la fiche sera mise à jour sur le site et dans la base.

Pour supprimer une fiche l'utilisateur devra tout simplement cliquer sur le bouton supprimer, ce bouton activera une requête sql qui supprimera la fiche en question dans la table.

### **3.4.Barre de navigation**

Sur le site internet, une barre de navigation est présente à l'arrivée sur la page de connexion et propose à l'utilisateur de se connecter en cliquant sur la rubrique "connexion" et de s'inscrire en cliquant sur la rubrique "inscription".

Une fois connecté, une barre de navigation différente apparaît. Elle comporte une rubrique "profil" qui dirige l'utilisateur sur sa page de profil personnel, une rubrique liste qui dirige l'utilisateur vers la liste des fiches créées par les utilisateurs et ses fiches qu'il pourra éditer et supprimer ainsi que faire des recherches par prénom ou nom ou sexe. Une rubrique "création de fiche" permettra à l'utilisateur de créer ses propres fiches. Et enfin une rubrique déconnexion est aussi présente, permettant à l'utilisateur de se déconnecter de son compte.

## **4. Conclusion**

### **4.1.Difficultés rencontrées**

Nous avons rencontré plusieurs difficultés lors de la réalisation de ce projet. Tout d'abord, nous avons eu des soucis avec des requêtes qui retournaient un booléen et non un objet de type PDOStatement sur laquelle la fonction fetch() ne pouvait donc pas fonctionner. Pour régler ce problème, nous avons décidé de faire des requêtes préparées au lieu de requêtes query, ce qui a aussi l'avantage de protéger notre site des risques liés aux injections SQL.

Ensuite, nous avons réussi facilement à crypter les mots de passe, mais lors de la vérification des identifiants et des mots de passe au moment de la

connexion, nous avons eu du mal à vérifier la concordance entre le mot de passe saisi et le mot de passe dans la base de donnée, qui lui, était crypté.

Aussi, nous avons rencontré un problème sur la gestion de la photo de profil de l'utilisateur. En effet, nous pensions utiliser une chaîne de caractères contenant le lien. Mais nous avons tout comptes faits décidé d'utiliser un longblob pour contenir l'image directement dans la base de données.

## **4.2.Ce que le projet nous a apporté**

Le projet nous à apporter de l'expérience supplémentaire dans le travail en équipe et dans la répartition des tâches à effectuer. Nous avons pu travailler notre organisation ainsi que nos compétences en php. Nous avons pu retravailler la conception et réalisation d'une base de données et ainsi que les diverses requêtes SQL. Nous avons aussi rencontré des problèmes auxquels nous n'avions pas encore fait face et avons appris à trouver des solutions à ceux-ci. Enfin, nous avons vu une nouvelle fois que les tests font partie intégrante d'un projet et qu'ils servent à peaufiner le code, à trouver des failles dans l'interprétation de messages et à savoir si l'exécution se déroule comme nous l'espérons.