

Rapport de SAE – Format' IUT

Sommaire

- Analyse
 - Liste des User stories implémentées
 - Diagramme des cas d'utilisations
- BD
 - Modèle Entité-Association
 - Liste des techniques provenant des cours de BD
- Développement WEB
 - Comment utiliser notre code
 - Connexions
 - Base de Données
 - Hébergement
 - Fichiers d'importation
 - Les fonctionnalités hors cours de PHP
 - Langages de Programmation utilisés
 - Commenter les nouvelles classes
 - Les consignes de Qualité de Dev
- Design et UI/UX

I. Analyse

Liste des User Stories implémentées

Sprint 1 :

Site facile à prendre en main

Rechercher une offre

Gestion de compte

Consultation des offres

Gestion des offres

Sprint 2 :

Gérer les membres de l'administration

Se connecter en tant qu'administrateurs

Créer sa convention

Rechercher par mots-clefs

Se connecter en tant qu'entreprise

Créer son compte d'entreprise

Importer/exporter des fichiers CSV

Se connecter en tant qu'étudiant via LDAP

Joindre un CV et une lettre de motivation

Naviguer facilement sur téléphone

Gérer une offre

Sprint 3 :

Refonte graphique

Se connecter en tant qu'enseignant

Gérer les grades d'administrateurs

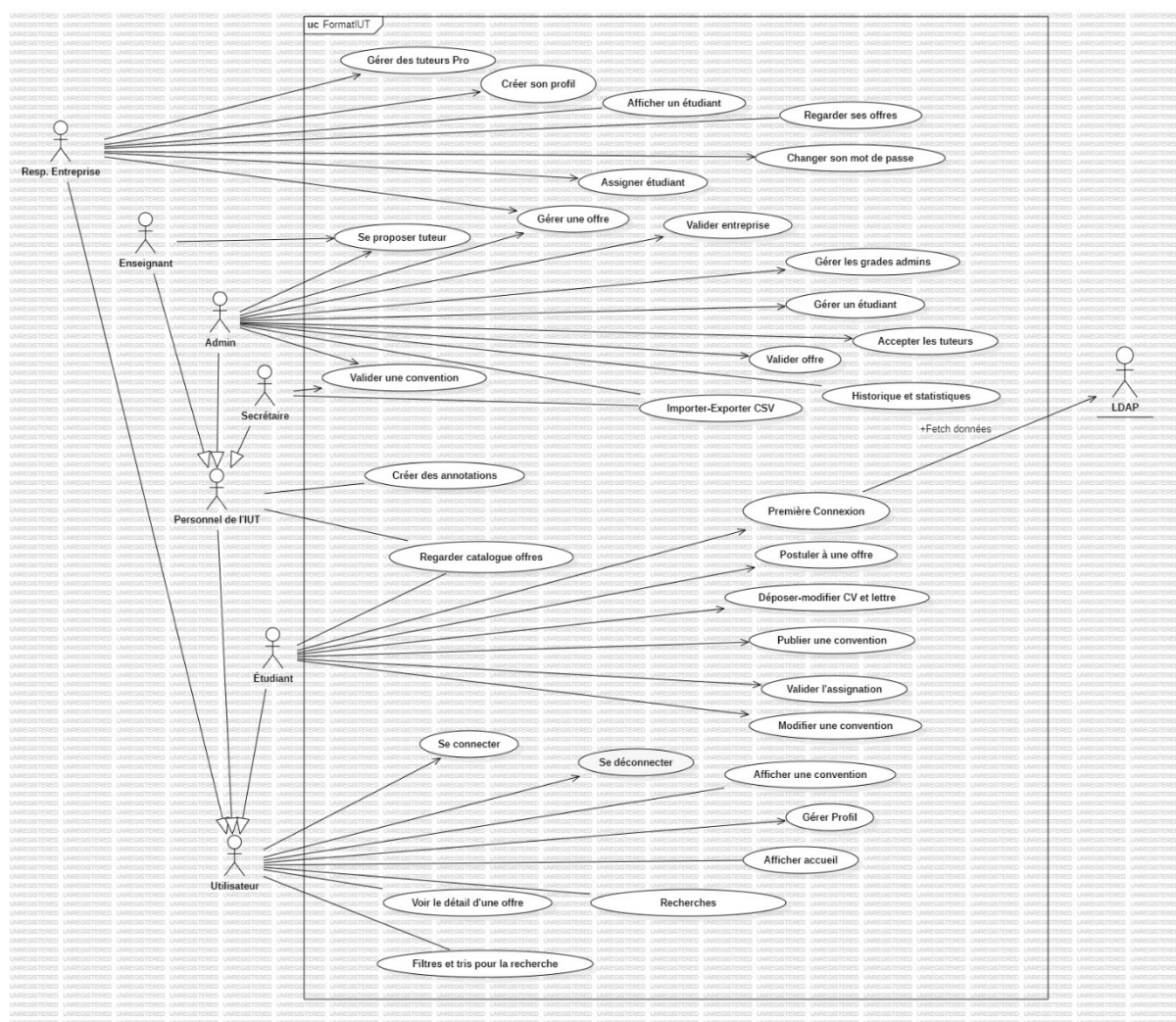
Améliorer la recherche

Ajouter un étudiant et une offre en tant qu'administrateur

Sprint 4 :

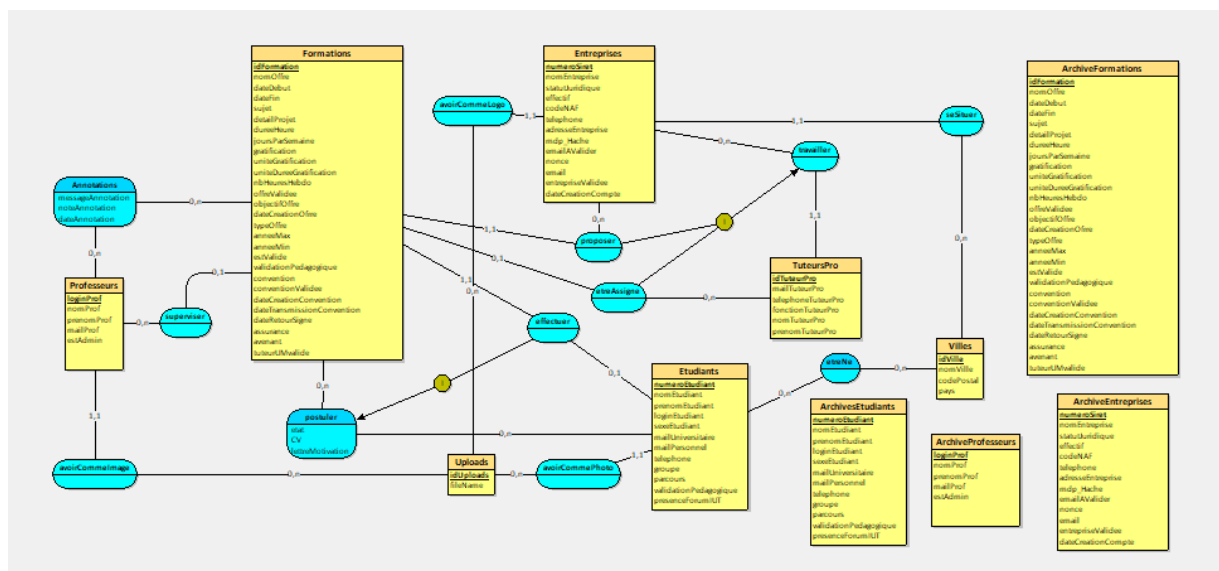
Se connecter en tant que secrétariat
Valider une convention
Tutorer un étudiant
Ecrire des annotations sur les entreprises (enseignant) et les consulter (étudiant)
Accéder à des statistiques et à un historique sur le site
Améliorer la visibilité sur téléphone
Améliorer la recherche (filtres, tris, permissions)
Créer une convention pour une offre n'étant pas sur le site
Gérer sa convention
Visualiser les informations d'un étudiant
Gérer une offre, une entreprise ou un étudiant en tant qu'administrateur
Envoyer des mails
Respecter le RGPD

Diagramme des cas d'utilisations



II. Base de données

Modèle Entités-Associations



Liste des techniques provenant des cours

Une des techniques provenant des cours que nous avons utilisé est le Trigger. En effet, nous avons utilisé un Trigger pour chacune des classes archives de notre base de données. Il y a donc quatre triggers before delete, un sur chaque table liée aux tables archives, c'est-à-dire Etudiants, Professeurs, Formations et Entreprises.

Il existe également un autre Trigger sur la table Formations, un before insert. Il vérifie qu'une entreprise soit valide avant d'ajouter une formation associée à celle-ci.

La seconde technique que nous avons utilisée est la fonction. Nous avons utilisé huit fonctions pour récupérer des données et les afficher dans la vue des historiques des administrateurs. Ce sont de simples fonctions sans aucun paramètre d'appel et renvoyant chacune un Int.

III. Développement Web

Comment utiliser notre code

- Les Connexions :

Pour se connecter en tant qu'entreprise, vous pouvez utiliser les identifiants suivants :

Identifiant= credit@yopmail.com

Mot de passe = argentargent

Pour une approche plus conventionnelle, il faudrait d'abord créer son entreprise via le formulaire dans l'accueil Entreprise, accessible depuis le menu Déroulant. Après avoir rempli les informations, accepté les CGU et appuyer sur le bouton NOUS REJOINDRE, il faudra valider son compte. Pour ce faire, une entreprise devra se rendre sur le mail qui lui a été envoyé puis suivre les instructions présentes dans celui-ci. Après cela, vous pourrez vous connecter en tant qu'entreprise.

Pour se connecter en tant qu'Étudiant ou en tant que membre du personnel, vous pouvez utiliser les connexions LDAP directement via la page de connexion. Sinon vous pouvez utiliser les identifiants suivants :

Pour un Étudiant : identifiant = EtuTest

Pour un Professeur : Identifiant = ProfTest

Pour un Administrateur : Identifiant = AdminTest

Pour un Secrétariat : Identifiant = SecretariatTest

Le mot de passe est le même pour ces quatre comptes : lLovelUT

Les connexions tests présentes ci-dessus sont désactivables dans la classe src/Service/ServiceConnexion.php, en modifiant l'attribut \$avecConnectionTest, situé en haut du fichier.

- Base de Données :

La base de données que nous utilisons est hébergée sur le webinfo de l'IUT à l'adresse suivante :

<https://webinfo.iutmontp.univ-montp2.fr/my/index.php?route=/database/structure&server=1&db=loyet>

Pour y accéder, vous pouvez utiliser les identifiants suivants :

Identifiant = Loyet

Mot de passe = gfsGnT!!hSSfE88.

Vous pouvez également retrouver ces informations dans la classe src/Configuration/Configuration.php dans la variable \$configWebInfo

- Hébergement :

Notre site est hébergé sur le webinfo de l'iut dans la session de Thomas Loye disponible à l'adresse suivante :

<https://webinfo.iutmontp.univ-montp2.fr/~loyet/2S5t5RAAd2frMP6/web/controleurFrontal.php>

En situation de secours, une backup du site est disponible à l'adresse :

https://thomasloye.fr/SAE_DEV/web/controleurFrontal.php?controleur=Main&action=afficherIndex

- Fichiers d'importation :

Notre site vous permettra d'importer certains fichiers selon les contextes :

En tant qu'admin, vous pouvez importer un fichier csv pour mettre à jour la base de données en vous rendant sur la page Mes CSV, accessible depuis le menu déroulant. Pour que l'importation se passe sans encombre, il faut utiliser des fichiers de type csv.

En tant qu'étudiant, qu'entreprise ou personnel, vous pouvez changer l'image de profil de votre compte afin de mieux se faire identifier par les autres utilisateurs. Vous aurez la possibilité de la changer lors de votre première connexion ou à tout autre moment en vous sur rendant votre profil. Après avoir choisi un fichier de type jpeg ou png pour votre image, vous pourrez enregistrer ces changements en bas du formulaire de modification de profil.

Fonctionnalités

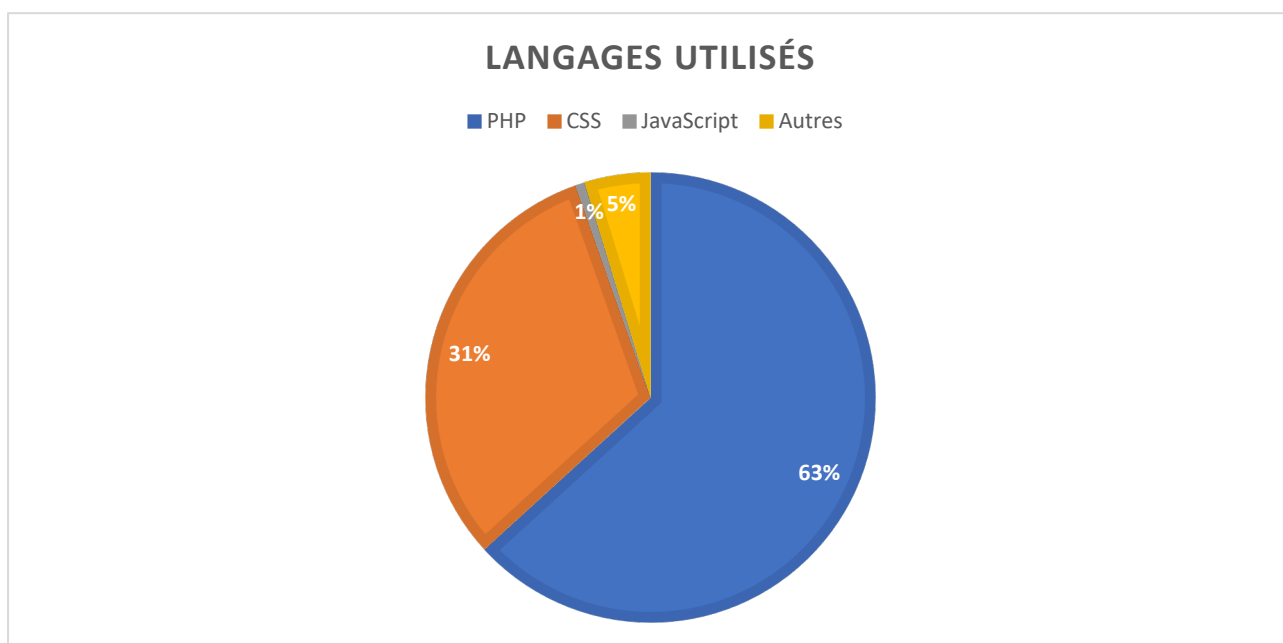
Le site Format' IUT abrite plusieurs fonctionnalités que nous n'avons pas vues en cours de PHP, en voici quelques-unes :

Nous avons mis au point un système de recherche accessible pour tout utilisateur. Nous avons essayé de rendre sa programmation la plus générique que nous pouvions le faire. Dans un premier temps, les éléments recherchables sont définis dans les classes utilisateurs : `src/Lib/Users/`, dans la fonction `getFiltresRecherche()`. Sont également définis dans cette fonction les différents filtres que l'utilisateur pourra utiliser pour trier les éléments recherchés. Pour pouvoir rechercher un élément, il faut que le repository de cet objet étende la `rechercheRepository` : `src/Modele/Repository/`. Un affichage par défaut sera mis en place pour les éléments recherchés, pour plus de personnalisation, il faudra alors créer une classe dans `src/Lib/Recherche/AffichagesRecherche` qui héritera de `AbstractAffichage`.

Il est important de préciser également que du JavaScript est utilisé sur le site, de façon purement visuelle et non fonctionnelle. Il sert de ce fait à afficher des éléments spécifiques, comme des <div> dans la quasi-totalité des cas. Nous avons pris soin de n'utiliser que très peu de JavaScript, puisque ce langage de programmation ne représente que 0,7% des langages utilisés dans ce projet.

Le JavaScript est regroupé en un seul fichier. Il se trouve dans ressources/JavaScript/mesFonctions.js. Etant donné le peu de JavaScript utilisé, il n'était pas pertinent de répartir ses fonctions dans plusieurs fichiers.

Langages de Programmation Utilisés



Une majeure partie du code est très logiquement du PHP. Beaucoup de CSS est également utilisé. La « partie » autre représente notamment les nombreuses images utilisées sur le site, tant sous la forme d'illustrations que sous la forme de fichiers appartenant aux utilisateurs, comme les photos de profil, les CV, lettres de motivations...

Les consignes de Qualité de Développement

- Respecter le S de SOLID :

Le principe de Single Responsibility est respecté pour toutes les classes du projet. Ceci est possible grâce au modèle MVCS : on va séparer les requêtes BD, la configuration, les objets métier, les classes spéciales (Lib) ayant un rôle spécifique, les contrôleurs qui font la communication entre front et back, et les services qui contiennent uniquement du code de back-end. Chaque classe a ainsi son rôle, et nous avons fait attention à qu'elles ne le débordent pas.

- Connexion LDAP :
Nous avons séparé la classe de configuration pour LDAP de la classe de connexion. Ainsi, comme pour l'accès à la base de données, nous avons deux classes bien distinctes qui gèrent les différentes responsabilités
- Couche Service :
Au début, nous avions la communication front/back et les méthodes de back-end réunies dans les classes de type Contrôleur. Mais pour alléger ces fichiers et mieux respecter le principe de Single Responsibility, nous avons déplacé les méthodes en question dans des classes Service, spécialisées là-dedans.
- Rôle du contrôleur :
Les fonctions présentes dans le contrôleur sont les fonctions d'affichage, qui permettent d'afficher des vues, ainsi que les fonctions qui appellent les services.
- Rôle du Service :
Les services contiennent les fonctions permettant le fonctionnement du site tels que les fonctions qui permettent d'ajouter, de modifier, de supprimer des offres, des étudiants, des entreprises, de postuler, de valider une convention etc.

IV. Design et UI/UX

Toute bonne application web doit respecter certains critères de développement, dont on peut citer, notamment, l'UI, l'UX et la charte graphique.

L'Université de Montpellier disposant d'une charte graphique récente et très complète, nous avons utilisé cette dernière pour réaliser l'intégralité du site. Cette dernière a été dument respectée, que ce soit au niveau des couleurs, de la mise en page, des polices, et de l'utilisations des logos.

On peut donc retrouver la police d'écriture spécifique à l'Université de Montpellier et ses couleurs sur tout le site, mais également aux documents qui y sont associés, comme les Conditions Générales d'Utilisation du site, ou encore ce présent document.

Le rouge et le bleu mythique de l'Université, sont, de ce fait, utilisés sur toutes les pages. Ils sont cependant recolorisés, de façon à les rendre plus « pastels » à quelques endroits spécifiques de façon à rendre le site le plus accessible possible.

L'UI et l'UX ont été travaillé, de façon à garantir une navigation fluide et innée de la part de l'utilisateur sur tout le site. Le bandeau vertical sur la gauche est réalisé de façon à être intuitif, et permet de retrouver son chemin dans un site

aussi vaste de Format'IUT. De plus, le site est intégralement accessible sur téléphone portable, ce qui semble être logique pour un site dédié notamment à des étudiants.

Le logo de Format'IUT rappelle les lettres F, U, T du nom du site. Il est réalisé en trois versions s'adaptant à son environnement : bleu, rouge et niveaux de gris sont utilisés en fonction de l'univers graphique de son environnement.