

Rapport de Stage

**Barthoux Sauze
Thomas**



Date : 22/05 au 23/06

—
Développement d'un outil de recherche des
assistants sociaux

—
Tutrice : Tierno-Guillaumin Karen

Sommaire

Sommaire.....	2
Remerciements.....	3
L'organisation.....	4
Le Conseil Départemental :	4
La Direction des Systèmes d'Information (DSI) :	4
Environnement de travail :	5
Description du Projet.....	7
Contexte :	7
Mission confiée / Objectif :	7
Etude de l'existant et mise en place d'un fichier CSV :	7
Réalisation et mise en place de la solution.....	9
La Base de Données :	9
Préparation :	9
Création :	10
Développement de la page :	12
Préparation :	12
Création :	13
Le Back Office :	14
Le Contrôleur liant entre Back et Front Office :	15
Le Front Office :	16
Intégration de JavaScript, Ajax, JSON :	18
Mise en sécurité de l'outil :	18
Logiciel utilisé :	19
Problèmes rencontrés :	19
Conclusion	20
Conclusion Personnel	21
Apport :	21
Perspective Professionnel :	21
Annexe 1.....	22
Annexe 2	23
Annexe 3	24
Annexe 4	31



Remerciements

Je tiens à remercier Patrice Lamy, responsable de la mission mais aussi Karen Tierno-Guillaumin, mon maître de stage et Benoit Marius, chef de projet pour leur accueil et leur accompagnement, tout au long de mon stage au sein de la mission Mission Stratégie Numérique, Nouveaux Usages et Mutualisation pour le Territoire.

Je remercie Karen et Benoit également pour toutes les réponses à mes questions et les connaissances qu'ils ont pu m'apporter. Ils m'ont permis de réaliser au mieux le travail demandé ainsi que mon rapport de stage.

Je remercie également tous les membres de la direction des systèmes d'information qui ont été bienveillants avec moi et qui auront parfois su m'aider dans la réalisation de mon travail, mais avant tout qui auront pu contribuer à mon intégration au sein du service et à la bonne ambiance durant le stage.

Enfin, je remercie M. Jean Laurent, élu du Conseil Départemental, sans qui je n'aurai pu réaliser ce stage qui était essentiel dans ma poursuite d'étude.



L'organisation

Le Conseil Départemental :

Le Conseil départemental de l'Allier est la collectivité territoriale qui gère la vie du département dans différents domaines : solidarité, tourisme, culture, les collèges et l'aménagement numérique.

Cette collectivité est représentée par des élus et un président qui sont élus lors des élections départementales. Les personnes élues le sont durant une période de 6 ans, par binôme dans chaque canton du département, ils en sont leurs représentants. Grâce aux élus, le Conseil départemental travaille sur des missions dans le but d'améliorer les conditions de vie dans le département et de valoriser son attractivité.

Le Conseil départemental traite de multiples missions, celles-ci sont affectées à différentes directions qui vont se charger de répondre aux problèmes et de mettre en œuvre les solutions dans le département. Ces directions sont représentées par l'organigramme présenté en annexe 1.

La Direction des Systèmes d'Information (DSI) :

La Direction du Systèmes d'Information est divisée en plusieurs branches présentées dans l'organigramme en annexe 2.

La DSI possède deux missions :

- 1- Mission stratégie numérique de L'éducation.
- 2- Mission Stratégie Numérique, Nouveaux Usages et Mutualisation pour le Territoire.

La DSI possède également trois services distincts :

- ➔ Le service Production en charge des projets d'infrastructure pour le conseil mais aussi pour les partenaires du Conseil comme les collèges.
- ➔ Le service Utilisateur et Centre de Services qui sont les techniciens venant régler les soucis informatiques pour les agents du Conseil départemental.
- ➔ Le service Méthodes et Projets avec la gestion des applications métiers et des logiciels internes à la collectivité.

Environnement de travail :

Lors de mon stage de cinq semaines au Conseil départemental de l'Allier à Moulins.

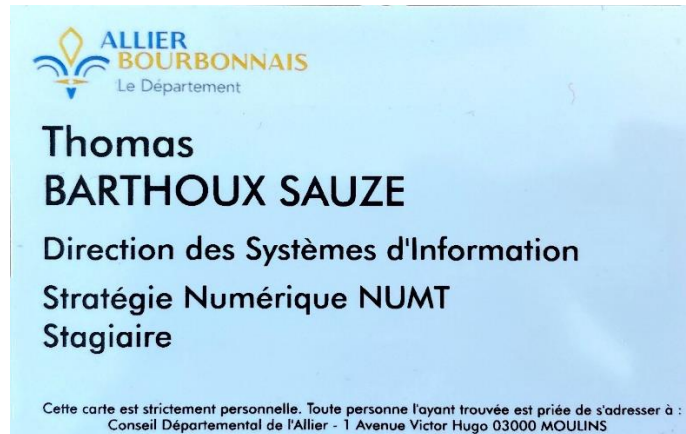
J'ai eu un environnement de travail optimal, avec pour matériel :

- ➔ Un ordinateur fixe.
- ➔ Double écran.
- ➔ Bureau commun avec ma tutrice de stage.
- ➔ Environnement calme.
- ➔ Grands locaux.

En raison de mon moyen de déplacement restreint, j'ai eu accès à un badge afin de rentrer plus tôt les matins me permettant ainsi d'être dans les locaux dès mon arrivée.



Photo du badge :



Afin de travailler au maximum en autonomie, j'ai eu accès à la messagerie professionnelle me permettant de communiquer avec ma tutrice plus facilement lors de ses jours de télétravail ou encore de communiquer avec d'autres agents si besoin.

De plus pour mener à bien le projet attribué, j'ai eu diverses autorisations. Cela m'a permis d'accéder au serveur pour y déposer mes fichiers et de tester mes programmes, j'ai aussi eu accès au service de base de données afin de créer entièrement celle-ci.



Description du Projet

Contexte :

Ce projet a vu le jour avec la demande des différentes Maisons de solidarités du département.

Ces Maisons de solidarité ont demandé au conseil d'avoir leur propre page pour un usage dédié aux professionnels sur le site du département. L'outil développé devra faciliter la recherche du bon interlocuteur.

Mission confiée / Objectif :

Pour ce stage, le projet qui m'a été confié est la création d'une page web afin de rechercher des assistants sociaux dans le département pour les professionnels.

Etude de l'existant et mise en place d'un fichier CSV :

Pour réaliser ce projet, j'ai regardé le site du département afin de voir diverses pages et comment elles étaient réalisées.

Après avoir regardé le site <https://www.allier.fr> dans sa globalité j'ai pu identifier différents facteurs qui seront à conserver et qui me seront utiles lors de la réalisation de la solution :

- ➔ Les différentes couleurs de la charte graphique.
- ➔ Le style général d'une page déjà créée pour une recherche d'élus dans le département.
Lien page des élus : <https://www.allier.fr/805-vos-elus.htm>



Pour continuer, j'ai étudié les différents fichiers fournis par les différents cantons. Ces documents ont pour but de renseigner les informations des différentes maisons de solidarité.

Cette étude a mené à la préparation d'un fichier CSV (séparateur point-virgule) qui sera utile pour la réalisation de la base de données. Ce fichier va, après modification des différentes colonnes permettre d'insérer les lignes directement sans avoir à les saisir manuellement, ce qui aurait été infaisable aux vues du nombre de ligne importants.

Image avant après modification du fichier :

Avant :

Après :

RUES ET LIEUX-DITS	Assistants Sociaux	Maison des Solidarités	Secteur	Cheffe(s) de Service Prévention Familiales
COMMUNE D'AUVERNES				
CHAMBONNAGE	LARRIERE Marisa	Auvernes	Bocage	Laurence CHAPON
RUE GABY MORLAY	LARRIERE Marisa	Auvernes	Bocage	Laurence CHAPON
AVENUE JEAN RENOU	LARRIERE Marisa	Auvernes	Bocage	Laurence CHAPON
RUE LOUIS JOUVET	LARRIERE Marisa	Auvernes	Bocage	Laurence CHAPON
PLACE SARAH BERNARD	LARRIERE Marisa	Auvernes	Bocage	Laurence CHAPON
RUE GERARD PHILIPPE	LARRIERE Marisa	Auvernes	Bocage	Laurence CHAPON
PLACE PIERRE LARQUEY	LARRIERE Marisa	Auvernes	Bocage	Laurence CHAPON
AUTRES RUES ET LIEUX DITS	DUFRESNE Emilie	Auvernes	Bocage	Laurence CHAPON
COMMUNE DE DOMPIERRE SUR BESBRE (rues)				
ACACIAS (lot des)	LAMOTTE Cécile	Dompiere-sur-Besbre	Sologne	Dominique BRU
AUBEPINE	LAMOTTE Cécile	Dompiere-sur-Besbre	Sologne	Dominique BRU
ARTICHAUT (rue de l')	MOLINARI Claire	Dompiere-sur-Besbre	Sologne	Dominique BRU
BANVILLE (place Théodore de)	MOLINARI Claire	Dompiere-sur-Besbre	Sologne	Dominique BRU
BASCULE (place de la)	MOLINARI Claire	Dompiere-sur-Besbre	Sologne	Dominique BRU
BEL AIR (chemin de)	LAMOTTE Cécile	Dompiere-sur-Besbre	Sologne	Dominique BRU
BELLEVUE (chemin de)	MOLINARI Claire	Dompiere-sur-Besbre	Sologne	Dominique BRU
BELLEVUE (HLM de)	MOLINARI Claire	Dompiere-sur-Besbre	Sologne	Dominique BRU
BERGERIE (chemin de la)	LAMOTTE Cécile	Dompiere-sur-Besbre	Sologne	Dominique BRU
BOURBONNAIS (rue du)	MOLINARI Claire	Dompiere-sur-Besbre	Sologne	Dominique BRU

1;CHAMBONNAGE;Avermes ;1;1
2;RUE GABY MORLAY;Avermes ;1;1
3;AVENUE JEAN RENOU;Avermes ;1;1
4;RUE LOUIS JOUVET;Avermes ;1;1
5;PLACE SARAH BERNARD;Avermes ;1;1
6;RUE GERARD PHILIPPE;Avermes ;1;1
7;PLACE PIERRE LARQUEY;Avermes ;1;1
8;AUTRES RUES ET LIEUX DITS;Avermes ;2;1
9;ACACIAS (lot des);Dompiere-sur-Besbre;3;2
10;AUBEPINE;Dompiere-sur-Besbre;3;2
11;ARTICHAUT (rue de l');Dompiere-sur-Besbre;4;2
12;BANVILLE (place Théodore de);Dompiere-sur-Besbre;4;2
13;BASCULE (place de la);Dompiere-sur-Besbre;4;2
14;BEL AIR (chemin de);Dompiere-sur-Besbre;3;2
15;BELLEVUE (chemin de);Dompiere-sur-Besbre;4;2
16;BELLEVUE (HLM de);Dompiere-sur-Besbre;4;2
17;BERGERIE (chemin de la);Dompiere-sur-Besbre;3;2
18;BOURBONNAIS (rue du);Dompiere-sur-Besbre;4;2



Réalisation et mise en place de la solution.

La Base de Données :

Préparation :

Pour commencer avant de réaliser la base de données, il a fallu faire une étude de la demande et comprendre les réels objets à mettre en place. Tout cela fut créé et réalisé sur un document technique présent en Annexe 3.

Celui-ci regroupe toutes les informations nécessaires comme les différents modèles de données qui ont été conçus avant le création de la base, afin de s'assurer du bon fonctionnement de celle-ci.

Dans cette documentation on y retrouve d'autre éléments qui ont été étudié avant de réaliser le projet, par exemple :

- ➔ Tableau comparatif avec deux solutions.
- ➔ Les différents modèles relationnels.
- ➔ Une maquette de la page.
- ➔ Les logiciels utilisés afin de réaliser les premiers tests de la base de données et des premières parties de code.

Tous ces éléments sont dans la documentation se trouvant en annexe 3.



Création :

Une fois la documentation rédigée, on passe à la réalisation de cette base.

Dans un premier temps la base a été créée sur un support local afin de tester si toutes les données, jointures et autres demandes fonctionnent ensemble.

Par la suite, nous pouvons créer la base de données sur le serveur où sont regroupées toutes les autres bases en lien avec le site.

Pour cette base de données il a été choisi de créer plusieurs tables qui seront reliées par des clés étrangères.

Liste des tables créées :

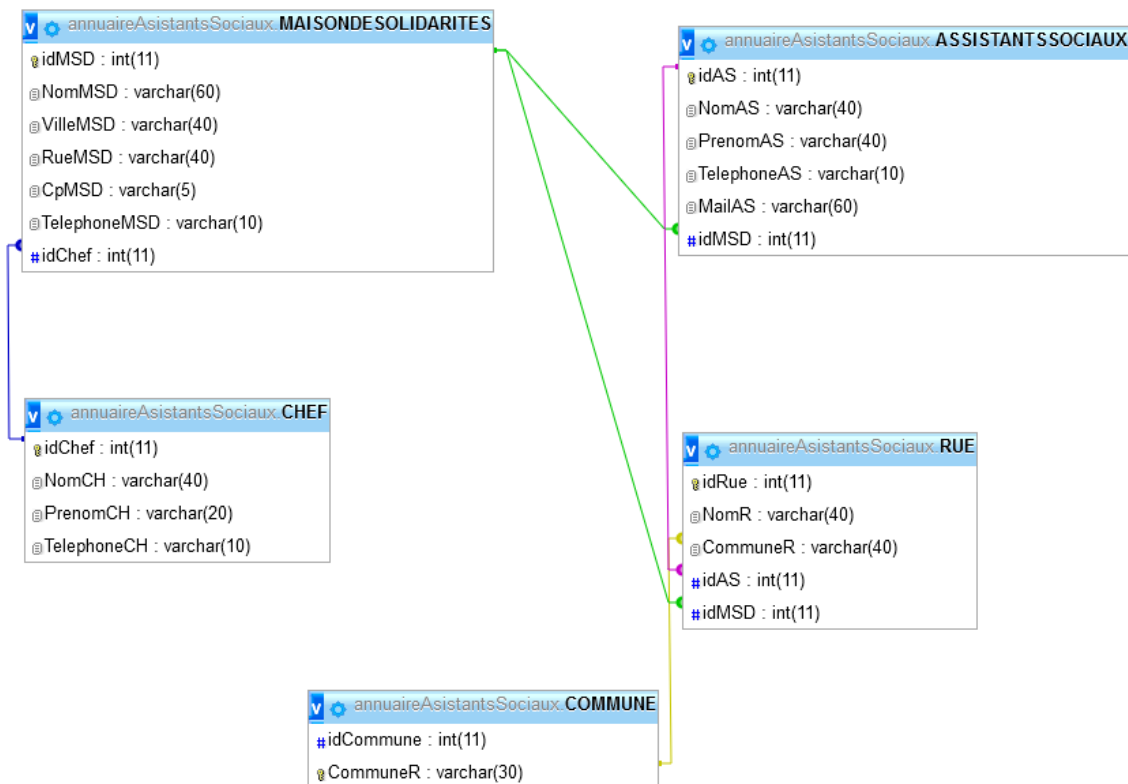
- ➔ **ASSISTANTSSOCIAUX** : Regroupe les noms des différents assistants sociaux et leurs coordonnées.
- ➔ **CHEF** : Celle-ci possède le nom et prénom des chefs des maisons de solidarités.
- ➔ **COMMUNE** : Stocke le nom des villes où les maisons de solidarités se trouvent. Celle-ci a été créée après la réalisation du document technique afin de faciliter le développement de certaines fonctions.
- ➔ **MAISONDESSOLIDARITES** : Possède toutes les informations sur chaque maison des solidarités.
- ➔ **RUE** : Cette table possède le plus d'information avec le nom de la rue, la commune à laquelle elle est rattachée mais aussi avec l'identifiant de l'assistant social et de la maison de solidarité qui lui est associé.



Pour la plupart des tables, les données ont pu être insérées à la main au vu du nombre de lignes à rentrer qui reste faible. A comparer de la table RUE qui elle a dû avoir une insertion différente des autres, ceci avait été anticipée grâce à l'étude de l'existant réalisé au préalable, cela a permis d'avoir un gain de temps avec l'insertion de toutes les données en même temps.

Une fois les tables créées et les données insérées, il faut créer les différentes liaisons.

Afin de réaliser celles-ci, j'ai utilisé le concepteur de données que propose phpMyAdmin. Le modèle est le suivant :



Pour finir avec la création de la base de données, il a fallu faire de multiples tests afin de vérifier si toutes les jointures, interclassements et valeurs sont corrects.

Une fois cela fait la partie du développement peut commencer.

Développement de la page :

Préparation :

L'objectif est de développer un outil afin de rechercher et trouver un(e) assistant(e) social(e).

Si l'on se place du côté du Conseil départemental, l'outil développé sera un module à implémenter dans le CMS (outil d'administration) du site Allier.fr dans la page dédiée. Cette page sera uniquement réservée aux entreprises.

Avant de commencer le développement, il a fallu établir une liste des besoins qui contient :

- ➔ Les différentes fonctions en liens avec la base de données.
- ➔ Les différents affichages.
- ➔ Les différents contrôles à réaliser pour coordonner le tout.

Afin de répondre à la demande de la liste des besoins, il a fallu déterminer une architecture de développement en relation avec la liste des besoins du projet.

Le choix fut l'architecture MVC. Elle permet de classer et d'organiser un projet de la meilleure des façons, avec pour chaque type de fichier son répertoire approprié.

Cela permet aussi de créer une arborescence du projet plus détaillée, disponible dans l'annexe 4.

En parallèle de cette phase de recherche d'architecture, le document technique de l'annexe 4 a commencé à être réaliser. Son but est de permettre à une tierce personne de comprendre la logique du projet, de son architecture et ce qui la compose.

Création :

Pour la création de cet outil, la préparation faite au préalable m'a permis de savoir quels langages seront utilisés :

- PHP 
- HTML 
- CSS 
- SQL 
- JavaScript 
- Ajax 
- JSON { j s o n }

Ce sont différents langages m'ont servi et de permis de voir les deux aspects de la programmation web :

- **Front Office** : La partie visible que l'utilisateur va voir est faite grâce aux langage HTML pour le contenu, CSS pour la mise en page, l'affichage le javascript pour l'ergonomie qu'il apporte. Il y aussi le JSON complémentaire du JavaScript, il renvoie à l'utilisateur des données envoyées par l'Ajax.
- **Le Back Office** : La partie non visible de la page. C'est dans cette partie que les informations saisies par l'utilisateur vont être utilisées pour de réaliser diverses tâches. Ce back office est programmé avec du PHP qui va organiser, contrôler, gérer et renvoyer les informations à l'utilisateur ou dans le programme. Mais aussi avec du SQL qui servira avec les informations reçues de communiquer avec la base et renvoyer les valeurs au PHP. L'Ajax qui lui va s'occuper de récupérer les valeurs enregistrées par le javascript et envoyer cela au serveur afin de réaliser un requête SQL et le renvoyer au programme.



Le Back Office :

Pour créer les fonctionnalités qu'utilise l'outil, le développement a nécessité la création du fichier :

→ class.pdoAS.php

Ce fichier servira à créer toutes les fonctions pour l'outil. Toutes les informations entre l'utilisateur et le serveur passeront par ce fichier.

Son rôle est d'envoyer à la base de données des requêtes préparées dans la fonction, avec les informations choisies par l'utilisateur.

Pour montrer son fonctionnement, voici une des fonctions importantes pour l'utilisation de l'outil :

```
/// Fonction qui vas renvoyer les informations des AS en rapport avec les infromations de la rue et de la commune
public function getAssitanteSocialRatacher($NomRue, $CommuneRue){
    $req = "select ASSISTANTSSOCIAUX.NomAS, ASSISTANTSSOCIAUX.PrenomAS, ASSISTANTSSOCIAUX.idAS, ASSISTANTSSOCIAUX.TelephoneAS, ASSISTANTSSOCIAUX.MailAS
    from ASSISTANTSSOCIAUX
    inner join RUE on RUE.idAS = ASSISTANTSSOCIAUX.idAS
    where RUE.NomR= :NomRue";
    $idJeuRes = PdoAS::$monPdo->prepare($req);
    $idJeuRes->execute(array(':NomRue' => $NomRue));
    $lesLignes = $idJeuRes->fetchAll();
    return $lesLignes;
```

Cette fonction s'occupe de récupérer les informations de l'assistant social en rapport avec la commune et rue sélectionnées.

Ces données seront utilisées lors de l'affichage du résultat de la recherche.

Elle va récupérer les données de l'assistant social :

→ Nom

→ Prénom

→ Id

→ Téléphone

→ Mail



Cette fonction est construite de façon à avoir enregistré la requête pour pouvoir s'exécuter dès son appel avec les valeurs sélectionnées.

Dans l'annexe 4, ce fichier est simplement expliqué avec les commentaires intégrés dans le code afin d'expliquer le fonctionnement de chaque fonction.

Le Contrôleur liant entre Back et Front Office :

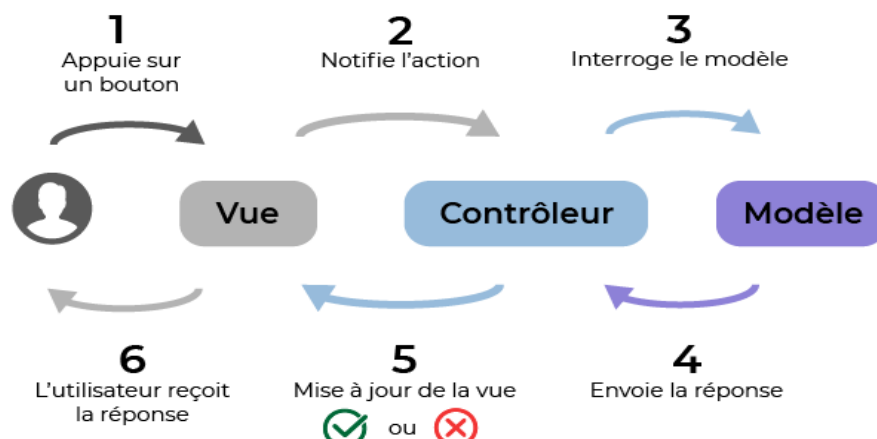
Pour effectuer une liaison entre le front et le back office il a fallu créer un fichier pour coordonner, lier les deux parties :

→ `c_assistantsociaux.php`

Ce fichier va permettre de récupérer les informations que l'utilisateur aura sélectionné sur le front office puis vas les envoyer, grâce à des variables, à la fonction du back office.

Cet envoi de données passe par ce contrôleur pour renvoyer une information.

Voici un schéma représentant le rôle du contrôleur dans le projet :



Contrairement au fichier `class.pdoAS.php`, le contrôleur ne possède pas de commentaire c'est pour cela qu'il est entièrement expliqué dans le document technique de l'annexe 4.

On verra plus tard qu'avec l'Ajax il a fallu créer un contrôleur annexe dédié pour la partie du développement avec ce langage.

Le Front Office :

Le front office est la partie visible de l'outil, c'est aussi sur cette partie que les différentes sélections seront réalisées.

Ces sélections seront et mises en forme grâce aux fichiers :

→ v_listeCommunes.php

→ v_rechercheAS.php

→ v_erreur.php

Chacun de ces fichiers seront utilisés lors de l'affichage et appelés en fonction du résultat.

Le premier affichage est réalisé par la vue v_listeCommunes.php :

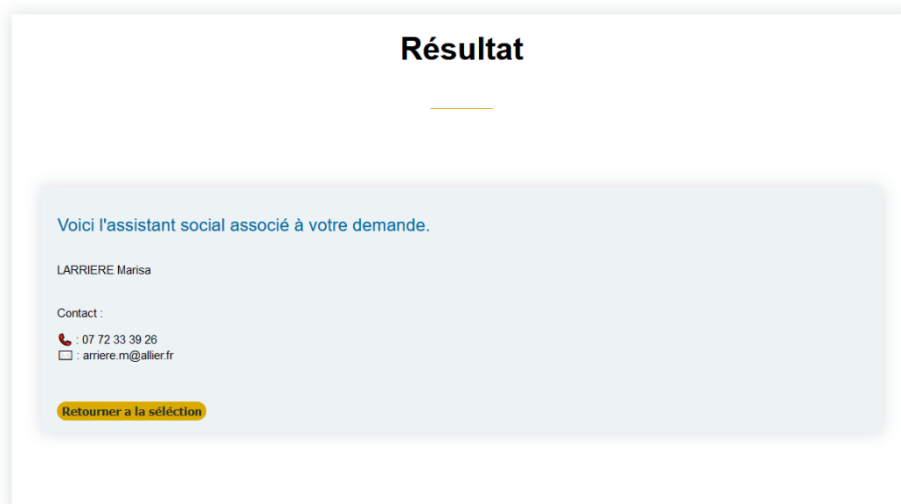


The screenshot shows a web form titled "Trouver votre assitant social". Below the title is a horizontal line. The form itself is a light blue box containing the instruction "Tout les champs proposer sont à sélectionner." (Note: the original text has a typo "les" instead of "tous"). There are two sections: "Choisir la Commune" with five radio button options (Avermes, Dompierre-sur-Besbre, Moulins, Varennes-sur-Allier, Yzeure) and "Choisir la Rue" with a dropdown menu showing "-- Sélectionner une ville proposer --". At the bottom center of the form is a yellow button labeled "RECHERCHER".

Cet affichage est expliqué dans le document technique de l'annexe 4 page 5.

En fonction de la sélection de l'utilisateur deux affichages sont possibles :

→ v_rechercheAS.php :



Résultat

Voici l'assistant social associé à votre demande.

LARRIERE Marisa

Contact :

☎ : 07 72 33 39 26
✉ : arriere.m@allier.fr

[Retourner à la sélection](#)

Cet affichage est expliqué dans le document technique de l'annexe 4 page 7.

Où

→ v_erreur.php + v_listeCommunes.php :



Trouver votre assistant social

Aucun assistant social n'est attribué à votre demande.

Tout les champs proposer sont à sélectionner.

Choisir la Commune

☐ Avermes
☐ Dompierre-sur-Besbre
☐ Moulins
☐ Varennes-sur-Allier
☐ Yzeure

Choisir la Rue

-- Sélectionner une ville proposer --

[RECHERCHER](#)

Cet affichage est expliqué dans le document technique de l'annexe 4 page 8.

Intégration de JavaScript, Ajax, JSON :

Cette intégration de langage a été mise en place en fin de développement pour un but d'ergonomie de l'outil, une meilleure fluidité dans le navigateur et une expérience plus agréable de l'outil.

Cela a apporté le principe de client-serveur :

- ➔ Dans notre cas on récupère des données ici grâce au JavaScript, cette donnée sera envoyée au serveur. Une fois les données renvoyées par le serveur, le JSON qui est format d'échange vas récupérer la donnée et la retranscrire au JavaScript qui rafraichira le champ ou cette donnée se trouve sur la page.

L'ajax (Asynchronous JavaScript and XML) est un outil qui permet une communication asynchrone entre la page et le serveur.

Dans le projet, tout cela a été mis en place afin que le nom des rues soit affiché automatiquement.

Cet affichage se fait une fois que l'utilisateur cliqué sur le bouton de la ville sans que l'utilisateur n'ait besoin de rafraîchir la page.

Mise en sécurité de l'outil :

Afin de sécuriser l'utilisation et le code de l'outil, des mesures de sécurité on été misent en place :

- ➔ Utilisation du databind dans le fichier class.pdoAS.php, cela permet de limiter le risque d'injection SQL
- ➔ Des contrôles sur les paramètres d'entrées qui seront rentre grâce à des expressions régulières alphanumériques
- ➔ L'outil sera utilisé sur une page hébergée sur le site du Département ou le service HTTPS est mis en place avec un certificat existant.



Logiciel utilisé :

Durant le stage j'ai pu utiliser diverses et multiples logiciels afin de mener à bien mon projet :

➔ Uwamp : Logiciel de serveur local ma permet de réaliser mes différents tests avant de passer sur le serveur du Conseil Départemental.



➔ Virtual Studio code : Logiciel de développement qui m'a permis de réaliser le code.



➔ PhpMyAdmin : Logiciel avec lequel j'ai réalisé la base de données.



➔ Suite office : Word pour les documentations techniques mais aussi pour le traitement des informations pour la base de données.



➔ WinSCP : Logiciel permettant de se connecter au serveur du Conseil Départemental afin de déposer mes fichiers du projet.



Problèmes rencontrés :

Durant le stage j'ai rencontré divers types d'erreurs et de problèmes qui m'ont bloqués et mis en difficulté :

➔ L'organisation au tout début du développement : ne sachant pas par quel fichier commencer j'ai réalisé des fichiers qui nécessitaient des éléments non créés et m'ont obligé à modifier une bonne partie du code réalisé pour être en raccord avec le reste du projet.

➔ Erreur de syntaxe par dizaine qui a mis un long moment à être trouvé puis corrigé.

➔ Le plus gros problème fut la compréhension et l'utilisation du JavaScript, Ajax et JSON car se sont des langages que je n'avais jamais vu.



Conclusion

En conclusion, ce projet m'aura pris la toute la durée de mon stage. En passant de la réalisation des deux documents techniques mais aussi l'étude de l'existant qui mon permis d'organiser au mieux le projet et de partir sur de bonnes bases.

Ce projet a nécessité une base de données qui a pour but de d'avoir les informations sur l'assistant social, les maisons des solidarités, le chef qui la dirige, les rues sur lesquelles sont affectés les assistants et les communes.

Cette base de données a été préparé avec un document Excel afin de tout insérer en même temps, les différentes liaisons ont été réalisé grâce au concepteur de phpMyAdmin.

Les différentes conditions que devait contenir la base de données ont été réalisé grâce à l'ajout d'id pour chaque table ce qui facilite la création de clé étrangère et permettre de réaliser des jointures qui serviront pour les différentes fonctions du développement.

Pour le développement de ce projet, une liste des attentes a permis de savoir quelle architecture le projet allait devoir utiliser. L'architecture MVC était la plus ergonomique en vue de son ergonomie et de sa facilité de prise en main. Une fois cela choisi, le début du développement commença, avec la création des divers documents permettant de relier les différents fichiers du projet mais aussi la liaison entre la base de données et le programme. Il s'en est suivi de la création du fichier permettant de réaliser toutes les fonctions que le programme aller avoir besoin. Par la suite il fut créé les différentes vues qui permettent l'affichage. Puis il y a eu la création du fichier qui allait s'occuper de coordonner tout le programme en faisant appel aux fonctions qui allaient communiquer avec les vues afin de réaliser un affichage.

Par soucis d'ergonomie une option JavaScript a été mise en place. Cette fonction permet d'afficher automatiquement les rues en fonction du choix de la commune.

Ce projet pourrait par le futur avoir une évolution sur un plan d'agrandissement de la base de données, avec l'ajout d'autre partie du Département.

Cela entrainerait la modification de la sélection de la ville en transformant les boutons radio utilisés actuellement par une liste similaire a celle de la sélection des rues.



Conclusion Personnel

Apport :

Ce stage m'a apporté des compétences comme le développement en PHP, avec un peu d'aide pour me lancer, j'ai réussi à réaliser et mener un projet à bien. Il m'a aussi montré l'importance de prendre le temps d'étudier l'existant avant de commencer à développer mais aussi l'importance de réaliser une étude avant de commencer. Cette étude m'a montré que cela rend le développement plus simple est plus rapide et de travailler sereinement en autonomie.

Grace à ma tutrice de stage et à Benoit Marius j'ai acquis du bagage technologique mais aussi de nouvelles connaissances dans le développement avec la découverte de langage comme JavaScript et d'autres éléments de développement comme Ajax ou encore JSON. Ils m'ont accompagné à comprendre la logique du développement. Ils m'ont aussi apporté des connaissances dans l'infrastructure que le web peut avoir avec le côté client et serveur.

Ce stage ma permis de me rendre compte à quoi ressemble réellement le métier de développeur web en ayant réalisé l'outil dans son ensemble. Je me suis rendu compte de la différence entre les cours et la réalité du métier.

Perspective Professionnel :

Ce stage m'a rassuré sur mes doutes dans mon avenir professionnel qui étaient apparus dans le courant de l'année scolaire pour devenir développeur web.

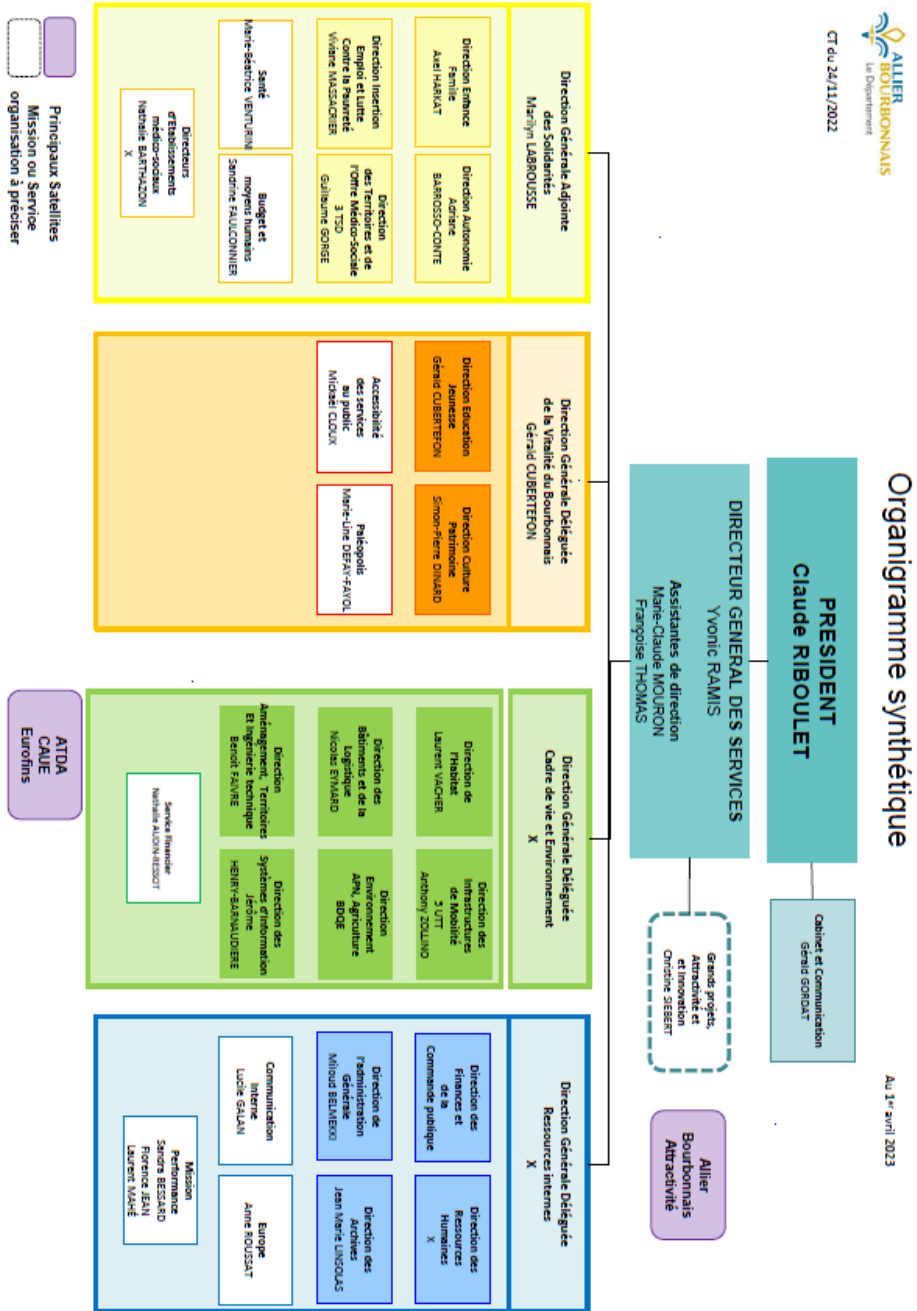
Deux perspectives se dessinent dans ma poursuite d'étude afin de devenir développeur front ou back office dans le web :

- ➔ BUT : Informatique
 - ➔ BUT : Métier du multimédia et de l'internet
 - ➔ Bachelor : Développeur d'application Web
-



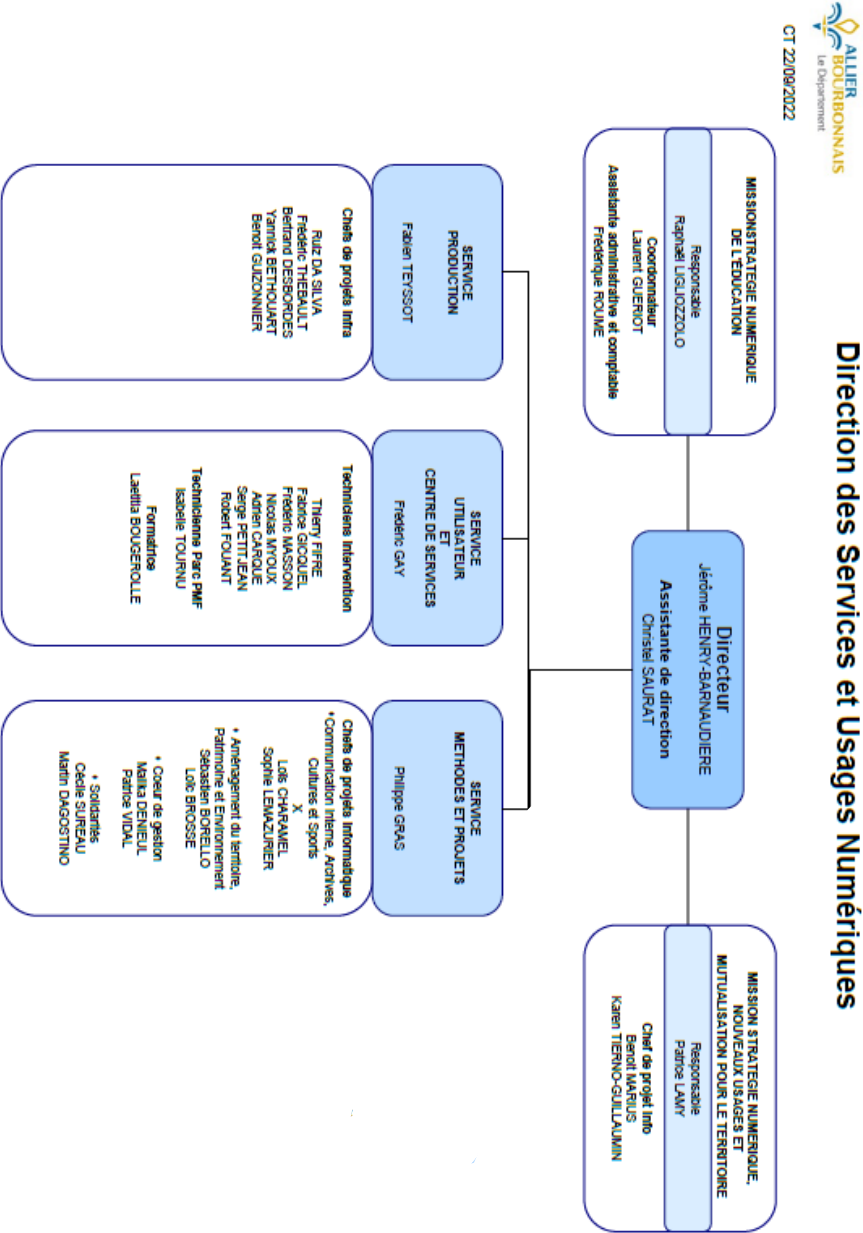
Annexe 1

Organigramme du conseil



Annexe 2

Organigramme de la DSI



Annexe 3

Document technique base de données

Projet Assistant social Moulins Nord Allier
Partie base de données



Sommaire :

La base de données : 2

Deux solutions sont envisagées : 2

 Solution 1 : 2

 Solution 2 : 2

Répartition des données et modèle relationnel : 3

Modèle relationnel : 3

 Table Assistants Sociaux : 3

 Table Maison des Solidarités : 3

 Table Chef : 3

 Table Rue et Lieux-Dits : 4

Modèle Entité/Association : 4

La page : 5

Technologies utilisées : 6



La base de données :

Deux solutions sont envisagées :

Solution 1 :

Créer une base de données comportant une seule table regroupant toutes les informations.

Point positif	Point négatif
Simple et rapide à réaliser et à mettre en place.	Peu lisible si besoin de modification. Taille importante. Peu de possibilités d'update. Peut efficace à long terme. Duplication des données.

Solution 2 :

Créer une base de données comportant plusieurs tables classées par catégories. Chaque table sera regroupée par des jointures qui vont permettre de faire les liens entre chacune.

Point positif	Point négatif
Facile à reprendre en main par une tierce personne. Données rangées et triées par catégorie distincte. Facilité lors de mise à jour d'information. Base de données optimisée pour chaque catégorie (id). Usage plus professionnel.	Long à réaliser avec un grand nombre de jointure.

La solution gardée est la deuxième car répondant bien plus à la demande et propose des avantages qui vont dans le sens du projet.



Répartition des données et modèle relationnel :

Toutes les données qui sont renseignées par le document Excel seront réparties dans différentes tables. Elles auront chacune leur propre utilité.

Par exemple pour les données concernant les Assistants Sociaux, leur table se nommera Assistants Sociaux, possèdera plusieurs colonnes : son id, son nom prénom, son Téléphone et sa Maison des Solidarités.

On fait de même pour les autres données dans les autres tables.

Modèle relationnel :

Table Assistants Sociaux :

ASSISTANTSSOCIAUX (idAS, NomAS, PrenomAS, TelephoneAS, MailAS, #idMSD, #idRue)

idAS clé primaire

idMSD clé étrangère en référence à idMSD de MAISONDESOLIDARITES

idRue clé étrangère en référence à idRue de RUE

Table Maison des Solidarités :

MAISONDESOLIDARITES (idMSD, NomMSD, VilleMSD, RueMSD, CoMSD, TelephoneMSD, #idChef)

idMSD clé primaire

idChef clé étrangère en référence à idChef de CHEF

Table Chef :

CHEF (idChef, NomCH, PrenomCH, TelephoneCH)

idChef clé primaire



Table Rue et Lieux-Dits :

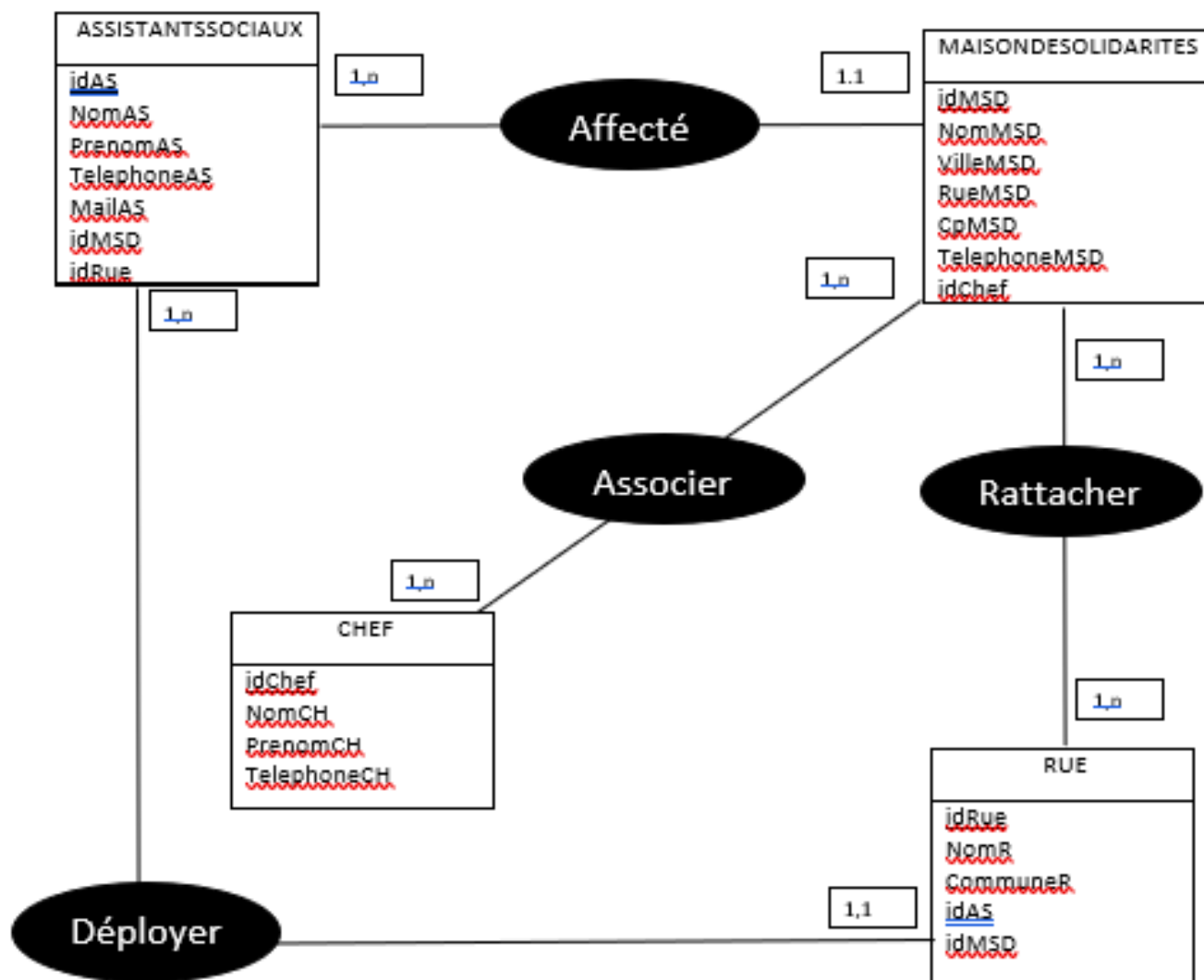
RUE (idRue, NomR, CommuneR, #idAS, #idMSD)

idRue clé primaire

idAS clé étrangère en référence à idAS de ASSISTANTSOCIAUX

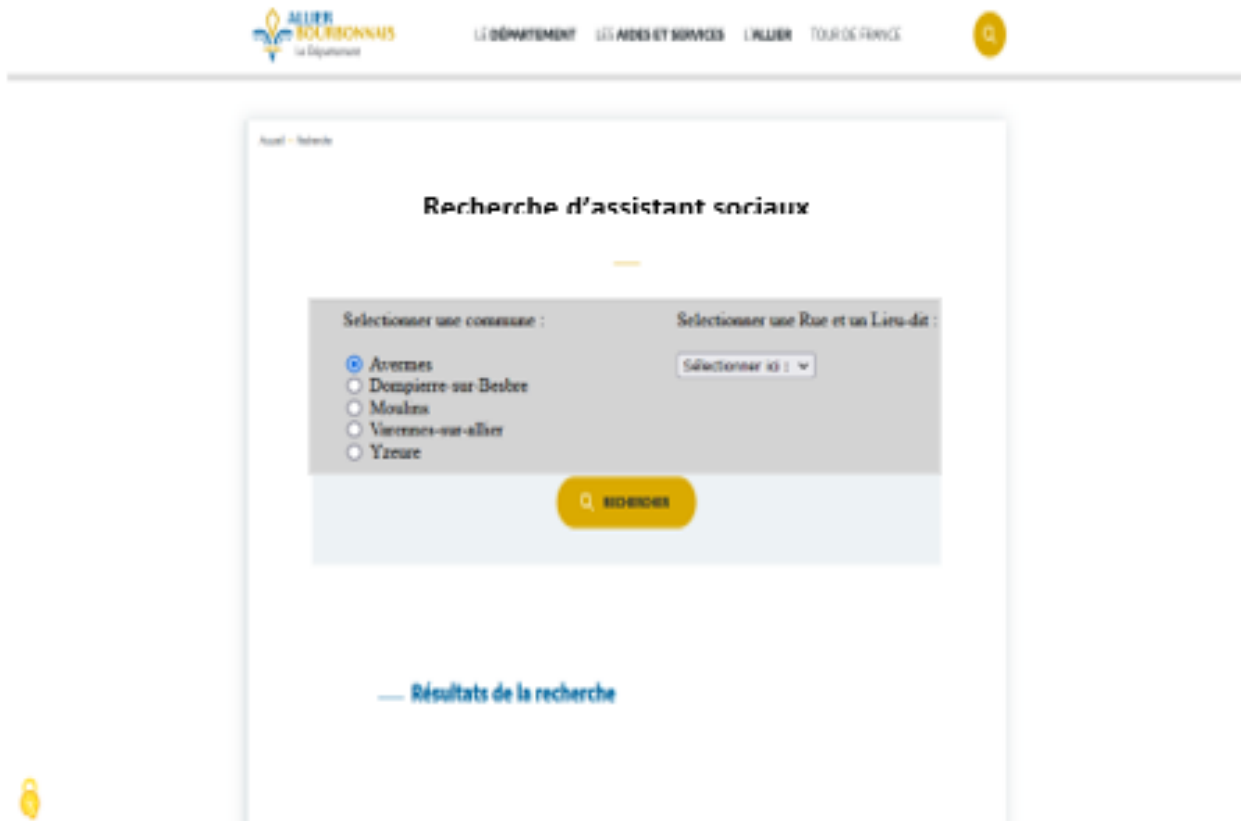
#idMSD clé étrangère en référence à idMSD de MAISONDESOLIDARITE

Modèle Entité/Association :



La page :

Pour cette page, on va utiliser le même style graphique que le reste du site allier.fr pour rester cohérent.



La page aura deux éléments comme ci-dessus. L'utilisateur pourra sélectionner une commune grâce à un bouton radio. Cela va débloquent le menu déroulant de Rue et Lieu-dit qui affichera toutes les rues en relation avec la commune sélectionnée.

Lors de la recherche, toutes les données sélectionnées de l'utilisateur vont être pris en compte.

Il n'y aura pas la possibilité de ne pas sélectionner un champ. Si un champ n'est pas sélectionné il n'y aura pas d'affichage tant qu'une commune ne sera pas sélectionnée.

Technologies utilisées :

Pour le développement de cette page on va utiliser un outil de développement nommé Visual studio code un logiciel de développement qui nous servira pour faire les parties en HTML, CSS et PHP.



Afin de tester la page et ses fonctionnalités on doit réaliser un teste sur un Uwamp portable avec les différents fichiers et une base de donné simple avec les informations nécessaires au fonctionnement.



Uwamp est un service de serveur portable qui peut s'occuper d'outils et services comme Apache, MySQL ou encore des éléments de code en langage comme le PHP et bien d'autres.

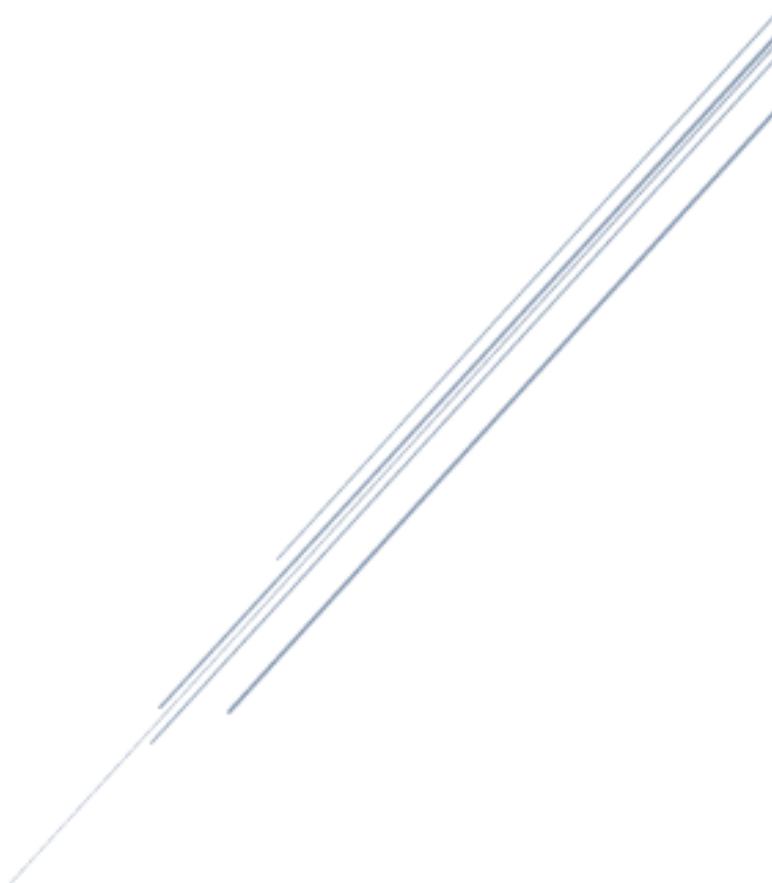
Si les différents tests reviennent concluants, les différents fichiers de code seront transférés vers le serveur qui héberge le site en rajoutant les nouveaux fichiers. Pour la base de donné on pourra la compléter en rajoutant des lignes dans les tableaux et par la suite finir de compléter les informations et les transférer vers la base du site.



Annexe 4

Document technique du développement.

Projet Assistant social Moulins Nord Allier
Partie développement



Sommaire

Structure du projet 2

 L’arborescence : 2

 Dossier Vue : 2

 Dossier Modele : 2

 Dossier Controleur : 2

 Dossier Constant : 3

Description c_assistantsociaux.php..... 4

 Action Recherche : 4

 Action Résultats : 6

 Le fichier class.pdoAS.php : 9



Structure du projet

Ce projet a été réalisé avec une architecture MVC. Elle permet de diviser le projet en différents types de fichiers en fonction de leur utilité et surtout de limiter le nombre de ligne de code en un seul fichier.

L'arborescence :

Ce projet ressemble à l'architecture employée dans le serveur où tous les fichiers sont stockés :

Annuaireas

- ↳ vue
 - ↳ v_rechercheAS.php
 - ↳ v_listeCommunes.php
 - ↳ v_erreurs.php
- ↳ modele
 - ↳ class.pdoAS.php
- ↳ controleur
 - ↳ c_assistantsociaux.php
- ↳ constant
 - ↳ constantAS.php
- ↳ index.php

Les fichiers présents ont une utilité différente les uns des autres.

Dossier Vue :

Dans ce dossier, il y aura tous les fichiers qui vont servir pour l'affichage des informations que l'utilisateur va voir. Dans notre cas les parties CSS et JavaScript des pages sont dans le fichier de chaque page.

Dossier Modele :

C'est dans ce dossier que l'on retrouve le fichier comprenant toutes les fonctions qui seront utilisées entre le code et la base de données.

Dossier Controleur :

Ici on retrouvera le fichier qui va coordonner toutes les actions du projet. Il sert à appeler les différentes fonctions et vue créées pour les faire fonctionner en même temps.



Dossier Constant :

Ce dossier va contenir le fichier qui s'occupe de relier la base de données au code. On y établit aussi les connexions avec les différents lieux où se trouvent les fichiers.

Explication de l'utilité des différents fichiers existants dans le projet :

v_rechercheAS.php	Cette vue va afficher le résultat de la recherche à l'utilisateur avec le nom de l'assistant social.
v_listeCommunes.php	Cette vue affiche les différentes sélections avec pour chaque ville un bouton radio et pour les rues une liste qui sera différente en fonction de la ville sélectionnée et renverra à v_rechercheAS.php s'il trouve un assistant social sinon il renvoie à v_erreurs.php
v_erreurs.php	Cette vue affiche un message d'erreur si aucun assistant social est trouvé par la fonction créée dans class.pdoAS.php
class.pdoAS.php	Vas regrouper toutes les fonctions entre le projet et la base de données. Ces fonctions seront réutilisables dans le contrôleur c_assistantsociaux.php
c_assistantsociaux.php	C'est dans ce fichier que les différentes fonctions et vues vont être appelées et utilisées pour l'affichage. Chaque affichage sera appelé 'action' dans le code.
constantAS.php	C'est ici que la liaison est effectuée entre base de données et arborescence des différents fichiers.
index.php	C'est le fichier central qui fait appel aux différents fichiers.



Description c_assistant sociaux.php

Le contrôleur va s'occuper d'appeler et relier les fonctions nécessaires à la vue.

Action Recherche :

Code :

```
case 'Recherche':{  
    $LesCommunes = $pdo->getNomCommunes();  
    include(VIEWSPATH."v_listeCommunes.php");  
    break;  
}
```

Case 'Rechercher' est le nom de l'action dans la barre de recherche du programme.

`$LesCommunes = $pdo->getNomCommunes() ;`

↪ La variable \$LesCommunes va exécuter la fonction getNomCommunes() qui est définie dans le pdo. Ici, la fonction renvoie toutes les communes enregistrées dans la base de données.

`Include(VIEWSPATH.'v_listeCommunes.php') ;`

↪ Faits appel à la vue v_listeCommunes.php depuis son chemin d'accès définie par VIEWSPATH dans le fichier constant.



Exemple de la vue lors de l'affichage :

Trouver votre assitant(e) social(e)

Tous les champs proposés sont à sélectionner.

Choisir la Commune

- ☐ Avermes
- ☐ Dompierre-sur-Besbre
- ☐ Moulins
- ☐ Varennes-sur-Allier
- ☐ Yzeure

Choisir la Rue

-- Sélectionner une ville proposer --

RECHERCHER

Break ;

↪ Cela signifie que l'action est finie.

Une fois les champs sélectionnés, l'utilisateur devra cliquer sur le bouton rechercher.

Cette action de cliquer sur le bouton va déclencher une seconde action qui va mener à un résultat ou à une erreur. Cette action donne le résultat de la recherche.



Action Résultats :

Code :

```
case 'Résultats':{
    $LesnomRue = $pdo->getNomRue();
    $NomRue = htmlentities($_REQUEST['1stRue']);
    $LesCommunes = $pdo->getNomCommunes();
    $CommuneRue= htmlentities($_REQUEST['1stCommunes']);
    if ($pdo->ExisteASRatacher($NomRue, $CommuneRue))
    {
        $LesnomRue = $pdo->getNomRue();
        $NomRue = htmlentities($_REQUEST['1stRue']);
        $LesCommunes = $pdo->getNomCommunes();
        $CommuneRue= htmlentities($_REQUEST['1stCommunes']);
        $LesAsRatacher = $pdo->getAssitanteSocialRatacher($NomRue, $CommuneRue);

        /*echo 'commune : '.$CommuneRue.' rue : '.$NomRue.' controleur: '.count($LesAsRatacher).' ';
        print_r($LesAsRatacher);*/
        include(VIEWSPATH."v_rechercheAS.php");
    }
    else {
        $erreur="Aucun assitant social n'est attribué a votre demande.";
        include(VIEWSPATH."v_erreurs.php");
        include(VIEWSPATH."v_listeCommunes.php");
    }
    break;
}
```

Comme on peut le voir, cette action est divisée en deux parties avec le if mais avant, nous récupérerons les informations qui ont été saisies à la page précédente une fois que l'utilisateur aura cliqué sur le bouton rechercher.

On récupère les informations saisies grâce aux deux variables :

`$NomRue = htmlentities($_REQUEST['1stRue']) ;`

↪ Récupère la valeur de rue qui a été sélectionnée.

`$CommuneRue = htmlentities($_REQUEST['1stCommunes']) ;`

↪ Récupère la valeur de commune qui a été sélectionnée.

Une fois ces valeurs récupérées, on recherche si un assistant social existe avec la fonction suivante.



↪ `ExisteASRatacher($NomRue, $CommuneRue))`

Si le résultat ressort positif, la condition va renvoyer les informations de l'assistant social.

↪ `$LesAsRatacher = $pdo->getAssistanteSocialRatacher($NomRue, $CommuneRue);`

Pour finir sur cette partie, on ajoute la vue qui affichera le résultat avec :

↪ `include(VIEWSPATH."v_rechercheAS.php");`

L'affichage sera celui-ci :

Résultat

Voici l'assistant(e) social(e) associé(e) à votre demande.

LAMOTTE Cécile

Contact :

☎ : 06 73 60 82 30
✉ : lamotte.c@allier.fr

[Retourner à la sélection](#)

Par contre, si la fonction `ExisteASRatacher($NomRue,$CommuneRue)` renvoie une valeur égale à 0, on sera redirigé vers la première vue mais cette fois-ci avec un message d'erreur qui est exprimé par :

↪ `$erreur="Aucun assistant social n'est attribué à votre demande.";`

Cette erreur est rattachée à une vue qui est appelée par la suite :

↪ `include(VIEWSPATH."v_erreurs.php");`



En plus de ce message d'erreur on inclut la première vue pour avoir le résultat annoncé avec :

↪ `include(VIEWSPATH."v_listeCommunes.php");`

Le rendu de l'affichage :

Trouver votre assitant(e) social(e)

Aucun assitant social n'est attribué a votre demande.

Tous les champs proposés sont à sélectionner.

Choisir la Commune

- ☐ Avernès
- ☐ Dompierre-sur-Besbre
- ☐ Moulins
- ☐ Varennes-sur-Allier
- ☐ Yzeure

Choisir la Rue

-- Sélectionner une ville proposer -- ▾

RECHERCHER



Le fichier class.pdoAS.php :

Ce fichier est commenté.

Ce commentaire indique au développeur le résultat que renvoie la fonction.

Exemple avec la fonction `getNomCommunes` :

```
/// Fonction pour récupérer les communes dans la base de données
public function getNomCommunes(){
    $req = "Select DISTINCT CommuneR, idCommune from COMMUNE";
    $idJeuRes = PdoAS::$monPdo->prepare($req);
    $idJeuRes->execute();
    $Leslignes = $idJeuRes->fetchAll();
    return($Leslignes);
}
```

Dans le pdo, le ou les commentaire(s) commence toujours avec `///`.

Tous les commentaires se trouvent au-dessus du nom de la fonction pour permettre de savoir directement le résultat sans avoir à faire la commande SQL sur la base de données et voir le résultat retourné.

