

Rapport de stage

BTS - Services informatiques aux organisations 2e année

Option **SLAM** : solutions logicielles et applications métiers

Stage effectué du 18/01/2021 au 19/02/2021

au sein de l'entreprise WCoding



<https://wcoding.com/>

11 Nonhyeon-ro 151-gil,
Gangnam-gu, Seoul 06037
02-501-6064

Maître de stage : Mr Oh Alex - CTO co-fondateur

Tuteur de stage CNED : Mr Roche Benoit

Sommaire

- I. Remerciements
- II. Introduction
- III. L'entreprise
 - Présentation
 - Les membres
 - Mon espace de travail
- IV. Préparation du projet
 - L'idée de projet
 - Le choix des technologies
 - Création d'un cahier des charges
- V. Développement Front-End
 - Créations des vues
 - Création de tableau
 - Le Framework Bootstrap 4
- VI. Développement Back-End
 - La base de données
 - Modèle conceptuel
 - L'ajout de donnée
 - La cryptographie
 - Les E-mails
 - Les rôles
 - Les vérifications
- VII. Conclusion
- VIII. Cahier des charges

I. Remerciements

Je tiens à remercier dans un premier temps, toute l'équipe pédagogique du CNED et les intervenants professionnels responsables de la formation BTS SIO, pour avoir assuré la partie théorique de celle-ci.

Je tiens également à remercier tout particulièrement et à témoigner toute ma reconnaissance aux personnes suivantes, pour l'expérience enrichissante et pleine d'intérêt qu'elles m'ont fait vivre durant ces 5 semaines au sein de l'entreprise Wcoding :

Monsieur Kyle Kim, CEO co-fondateur de l'entreprise pour son accueil et la confiance qu'il m'a accordé dès mon arrivée.

Monsieur Alex Oh, CTO co-fondateur, mon maître de stage, pour m'avoir intégré rapidement au sein de l'entreprise et m'avoir accordé toute sa confiance ; pour le temps qu'il m'a consacré tout au long de cette période, sachant répondre à toutes mes interrogations ; sans oublier sa participation au cheminement de ce rapport.

Ainsi que l'ensemble du personnel de Wcoding que ce soit l'équipe Marketing ou des développeurs pour leur accueil sympathique et leur coopération professionnelle tout au long de ces 5 semaines.

II. Introduction

C'est avec enthousiasme que j'effectue ma deuxième année de BTS – SIO option SLAM au sein du CNED.

Dans le cadre de ma formation j'ai été amené à effectuer un stage au sein de l'entreprise Wcoding se situant à Séoul en Corée du Sud au date du 18/01/2021 au 19/02/2021.

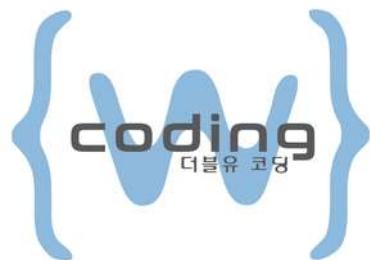
La mission qui m'a été confiée durant ce stage était la création d'un site internet pour l'administration d'une agence de voyage française également située en Corée.

Dans un premier temps je vous expliquerai la mise en place du projet puis le déroulement de celui-ci en commençant par le choix des technologies utilisées lors du stage mais également avec des bouts de code et les explications nécessaires à la compréhension de celui-ci que ce soit pour la partie Front et Back-end.

Nous terminerons sur une conclusion de stage très positive.

III. L'entreprise

- Présentation



L'entreprise Wcoding a été créée en 2017 au cœur de la capitale Sud-Coréenne qui est Séoul par deux personnes, Mr Kyle Kim co-fondateur spécialisé dans la partie commerciale de l'entreprise suite à son parcours universitaire ainsi que Mr Alex Oh également co-fondateur s'occupant de la partie développement avec plus de 30ans d'expérience dans le domaine de la programmation, et ayant enseigné pour les plus grandes entreprises comme Fujitsu, Helwett-Packard mais également Orange Belgique ou Mr Kim et Oh se rencontrèrent.

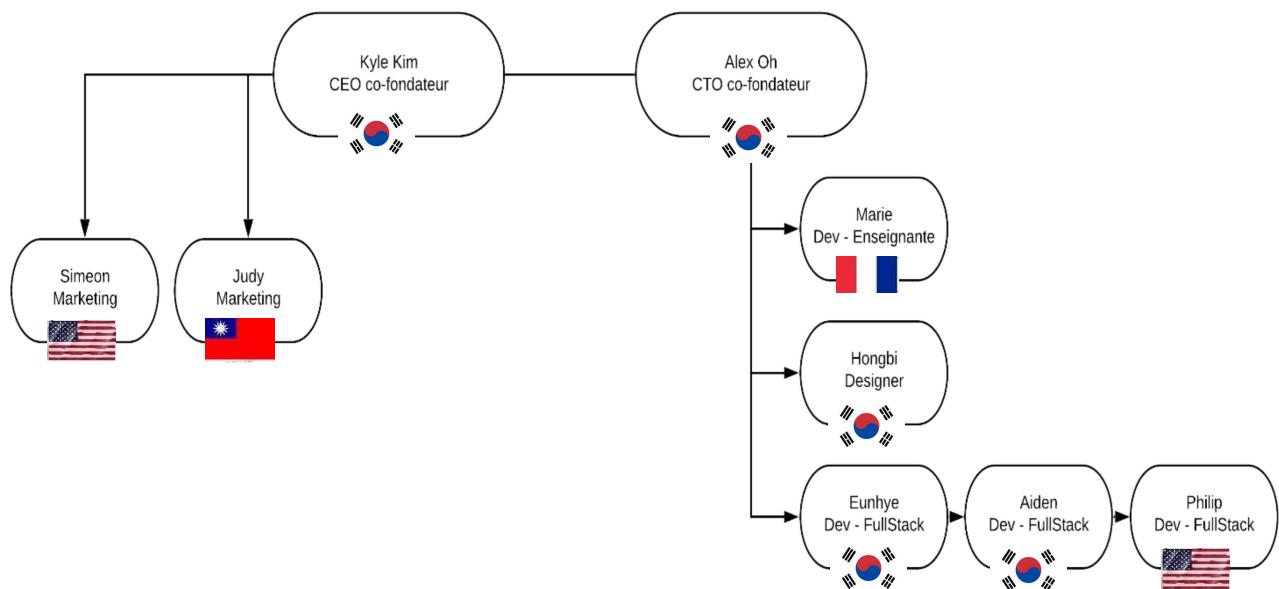


L'entreprise a deux casquettes, l'une étant la formation de la programmation à des gens souhaitant s'initier ou se perfectionner dans un langage en particulier avec des classes dispensées quotidiennement (par une enseignante française) en langue anglaise qui de ce fait rend le lieu très cosmopolite.

Leur deuxième casquette étant le développement pur avec une équipe de 4 développeurs qui sont en charge de créer des applications Mobiles/ Site Web que des entreprises leurs demandent, c'est donc avec eux que j'ai effectué mon stage.

- Les membres

Voici un organigramme pour représenter les membres au sein de l'entreprise de Wcoding au total de 9 employés et 4 nationalités différentes, la langue principale étant l'anglais pour la communication interne, meeting, cahier des charges etc...

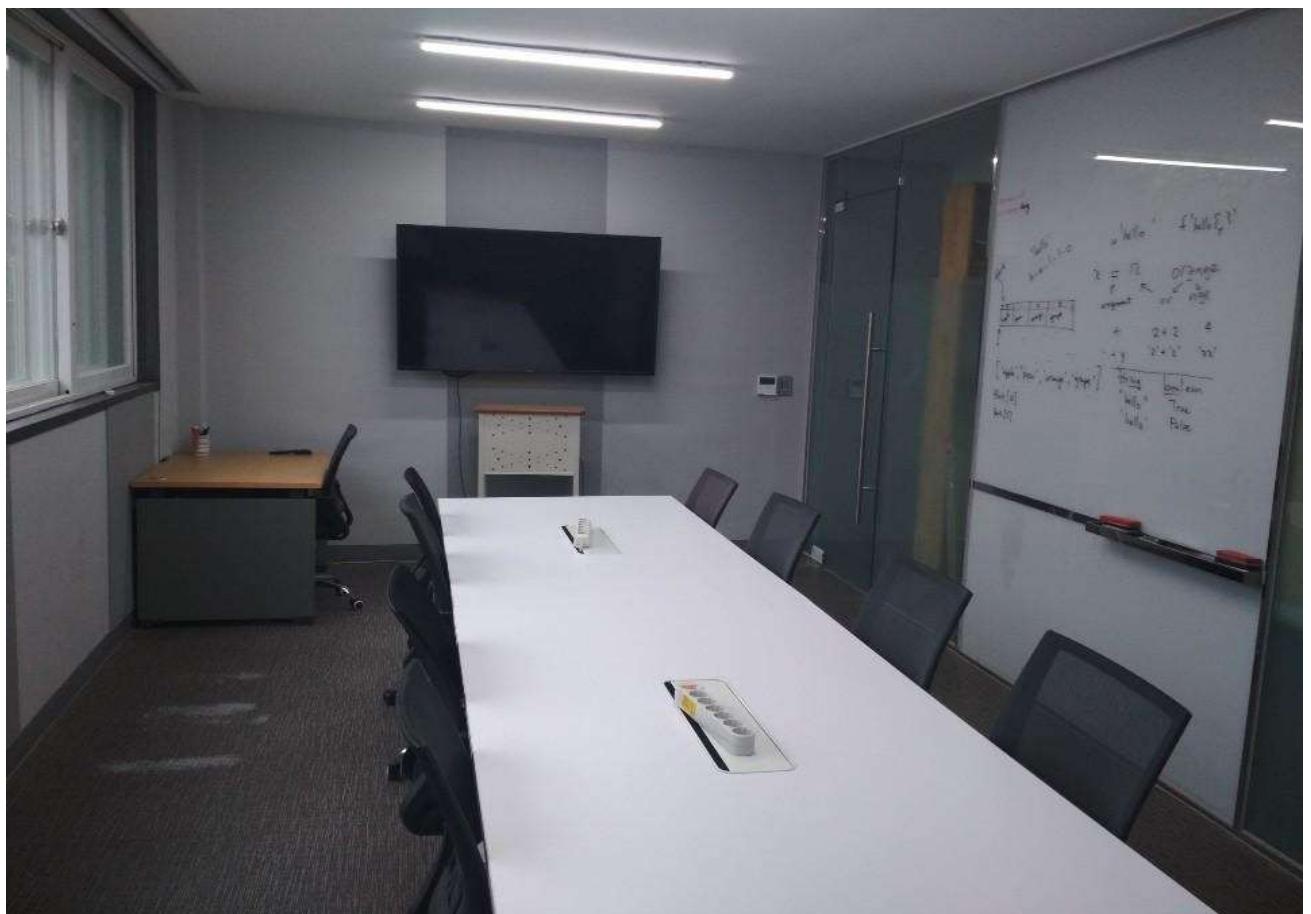
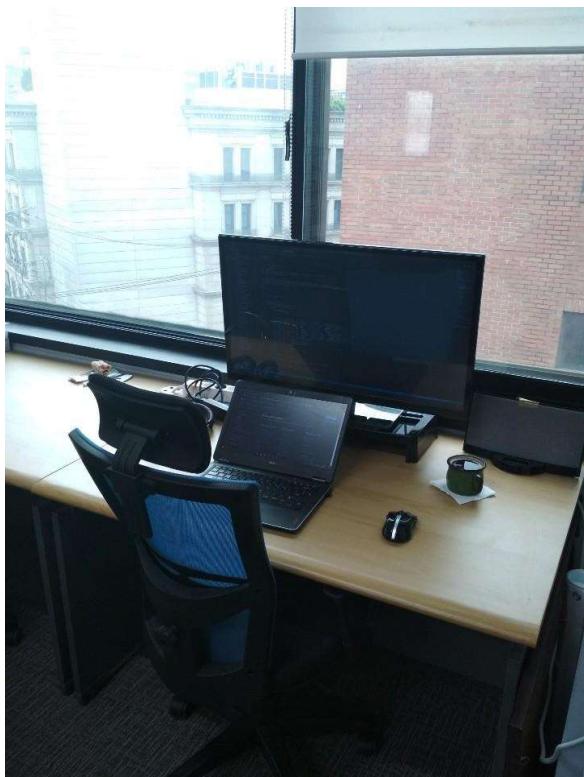


- Mon espace de travail

Tous les employés de l'entreprise travaillent avec un ordinateur portable pour des raisons de mobilité, ne seraient ce que pour les réunions, et un écran de plus grande taille leur est fourni ce qui permet un certain confort et une meilleure productivité.

Il en a été de même pour moi, je me suis vu remettre un poste de travail avec un écran supplémentaire juste à côté de l'équipe de développeur ce qui a été très pratique pour mes questions durant le stage, ainsi que d'écouter leur façon d'échanger sur leurs projets.

Voici deux photos des locaux avec en premier les postes des développeurs, ainsi que la salle de classe pour les formations



IV. Préparation du projet

- L'idée de projet

Durant ma première année de BTS j'avais effectué mon stage dans la même entreprise avec les mêmes personnes. Et ce durant ce stage j'avais eu la tâche d'effectuer une application Bureau pour la plateforme Windows en C# avec le logiciel Visual Studio.

L'application devait permettre à une agence de voyage de gérer ses employés ainsi que ses voyageurs

L'idée était donc de repartir sur ce même projet mais cette fois, de développer un site internet et non une application bureau. Ceci m'a permis de mettre à profit ce que j'avais appris durant ma deuxième année scolaire.

Donc pour résumer la tâche qui m'a été demandée était :

La création d'un site internet pour la gestion des employés d'une agence de voyage et celle de leurs voyageurs, seuls les employés auront accès à ce site.

Je précise également que **ceci est un projet fictif** en aucun cas ce site ne sera utilisé par cette fameuse agence de voyage.

- Le choix des technologies

Nous connaissons à présent l'idée du projet, le pose maintenant la question des technologies à utiliser. Quand je parle de technologie, je parle des différents langages de programmation que j'ai eu besoin pour mener à bien mon projet.

J'englobe également dans cette partie les outils qui m'ont servi.

Pour créer un site, il y a ce qu'on appelle le Front-end (la partie visible des utilisateurs) et le Back-end (la partie cachée de l'iceberg), ces deux parties distinctes ne s'écrivent pas avec les mêmes langages de programmation.

Nous ne faisons donc pas du tout la même chose avec des langages dit "Front" et "Back" end. L'un étant pour afficher à l'utilisateur et l'autre pour faire des requêtes aux serveurs pour récupérer des données sur une base de données par exemple.

Pour la partie **Front-End**, j'ai donc utilisé naturellement le **HTML version 5** qui est un langage de description et le langage de balisage **CSS version 3**.



Logo Html5/Css3/Javascript

J'ai également utilisé du Javascript qui m'a permis de créer des animations sur le site, ou alors pour simplifier certaines tâches sur les formulaires par exemple, que je détaillerais plus loin dans le rapport.

Pour la partie responsive, je me suis servi du framework **Bootstrap version 4**, qui permet de développer rapidement avec une multitude de classe déjà prêtées à l'emploi pour les sites web, je cite Mr Giraud Pierre :

« Bootstrap est un ensemble de fichiers CSS et JavaScript fonctionnant ensemble et qu'on va pouvoir utiliser pour créer des designs complexes de manière relativement simple. »

Ces avantages sont :

- Un gain de temps de développement qui peut être conséquent ;
- Une certaine robustesse dans l'architecture globale du code ;
- Un framework appartenant à une grande société (Twitter) mais open-source



Logo Bootstrap 4

En ce qui concerne le **Back-end** le **PHP version 7.2** a été utilisé pour faire des requêtes au serveur, et le langage **SQL** m'a permis d'écrire les requêtes pour communiquer avec ma base de données.



Logo Sql et Php

Maintenant que les langages de programmation ont été sélectionnés, j'ai dû réfléchir aux différents outils pour ce projet.

Ayant mes habitudes avec l'IDE Visual Studio Code (propriété de Microsoft), je me suis naturellement dirigé vers celui-ci, ayant déjà quelques extensions que j'ai pu utiliser sur d'autres projets. De plus, cet éditeur de texte (comme beaucoup d'autres) propose une fonction pour le versionning avec Github.

Ne pas repartir sur un nouvel éditeur a été un vrai gain de temps.

Concernant le serveur, travaillant en local, donc sur mon ordinateur j'ai opté pour WampServer avec son serveur Apache, gratuit et facile à mettre en place sur un poste de travail Windows. De plus, il inclut PhpMyAdmin qui permet la gestion d'une base de données en Sql également facile à utiliser.

Travaillant seul sur ce projet, mais sur deux postes différents et pour des raisons évidentes de sécurité, j'ai utilisé git, en créant un répertoire sur la plateforme github et en prenant soin d'envoyer lors de nouvelles fonctions.



WampServer/MySQL/Git/GitHub

Pour les gens ne connaissant pas le versionning (gestion de versions en Français) je cite Wikipedia :

« Le versionning consiste à gérer l'ensemble des versions d'un ou plusieurs fichiers. Essentiellement utilisée dans le domaine de la création de logiciels, elle concerne surtout la gestion des codes

source. »

Il permet aux développeurs de créer des branches, ou ils travaillent chacun sur la leur, et par la suite regroupe le code sur un tronc commun. Mais il permet de revenir en arrière si quelque chose ne fonctionnait pas sur une version antérieure par exemple.

Autres logiciels nécessaires au bon déroulement :

Fini les outils pour la programmation, voici donc quelques outils dont je me suis servi, et dont je me sers pour d'autres projets en général.

- La suite Office et tout particulièrement le logiciel **Word** qui n'est plus à présenter et qui m'a servi dans la rédaction du cahier des charges du projet mais également pour le rapport de stage.
- Le logiciel **Discord** pour échanger avec les membres de l'équipe, quand ils n'étaient pas présents.
- Le site **Trello**, excellent pour savoir où nous en sommes dans notre travail, un MUST HAVE pour la gestion de projet et de plus gratuit (avec certaines limitations, mais largement suffisante pour ce genre de projet)

Concernant le design, ayant à le faire moi-même j'ai dans un premier temps utilisé le logiciel **Balsamiq** (version gratuite) pour créer les maquettes des différentes pages du site, ce qui m'a permis de voir où allaient se situer les informations, images, etc...



Logo Word / Discord



Logo Trello / Balsamiq

- **Création d'un cahier des charges**

Avant même de commencer à coder, j'ai dû créer un cahier des charges pour me faciliter la suite. Ceci n'a pas été une mince affaire, car il a fallu essayer de penser à un maximum de choses pour ne pas avoir de chose manquante par la suite.

Dans mon cas, le projet étant fictif, mais l'agence de voyage existe belle et bien. Connaissant le gérant de cette petite agence également situé à Séoul, je les ai rencontrés pour discuter de ce petit projet qu'était le mien.

Nous avons échangé et essayé de voir s'il était possible de faire quelque chose ensemble, j'ai vite compris qu'ils cherchaient plutôt un logiciel/site sales force, qui vous le comprenez bien n'est pas réalisable en seulement 5 semaines.

J'ai donc durant notre entretien noté le plus d'informations possible, sur ce qui pourrait les intéresser, la façon dont ils aimeraient l'application et ce dont ils souhaiteraient voir apparaître dedans. Ce qui m'a permis de créer ma base de données, qui vous l'imaginez a évolué par la suite au fur et à mesure de l'avancement du projet.

Le détail de ce cahier des charges est en fin de rapport, vous y retrouverez les questions que je me suis posées pour le créer, les maquettes de toutes les vues du site qui m'ont aidées par la suite pour coder plus rapidement avec un design en tête.

L'écriture du cahier des charges m'a pris environ 3 jours, entre les questions, les maquettes mais également la confection de la base de données. **J'ai réalisé ce cahier des charges en amont du stage pour ne pas perdre de temps.**

Je me suis servi de Word, et du logiciel Balsamiq pour la création des maquettes.

Balsamiq est un logiciel(Freemium) qui permet donc de créer des maquettes en Glisser/Déposer très simple d'utilisation avec un large choix d'objets.

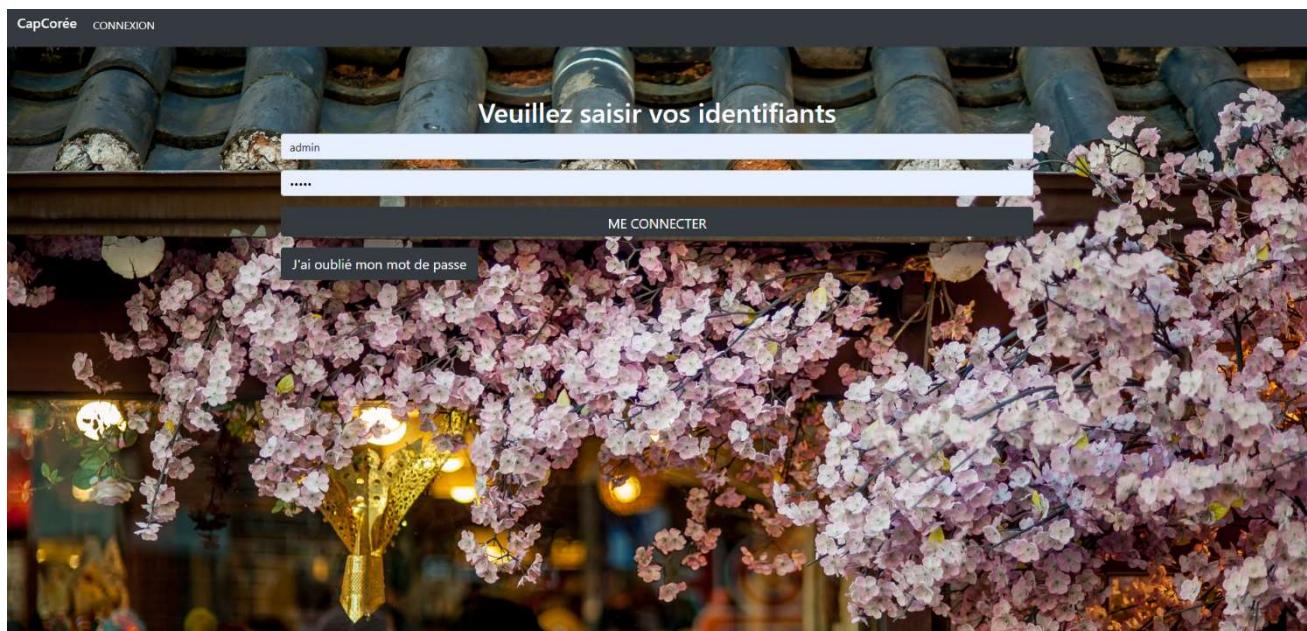
Le jour J mon poste de travail était prêt et je pouvais désormais commencer à coder. Étant familier avec cette entreprise et ses employés pour avoir effectué mon stage de première année chez eux, je n'ai pas vraiment eu besoin de temps d'adaptation.

V. Développement Front-End

- Créations des vues

Me voilà donc rendu au développement, j'ai donc commencé par écrire mes pages, la mise en forme. Le fait d'avoir écrit un cahier des charges, et surtout la confection des maquettes, m'a grandement facilité la tâche.

J'ai commencé par créer une page de connexion qui une fois connecté nous redirigera vers notre compte et tous les menus auxquels nous aurons accès en fonction de notre rôle dans l'agence.



Page Connexion

Il sera également possible de recevoir un nouveau mot de passe si vous avez oublié le vôtre. Au lieu d'ouvrir une nouvelle page, j'ai décidé d'utiliser Javascript et de créer une fenêtre "Modale" (nouveau nom pour pop-up).

```
159 |         </form> <!-- En cliquant sur le bouton on ouvrira le modal -->
160 |         <button id="button_modal" class="btn btn-lg btn-dark" onclick="openModal()">J'ai oublié mon mot de passe</button>
161 |     </section>
162 | 
```

```
163 | 
```

```
164 |     <div id="modal"> <!-- Creation du modal pour reinitialiser le MDP -->
165 |         <button id="close" class="btn btn-sm btn-dark" onclick="closeModal()">X</button>
166 |         <h2>Mot De Passe Oublié ?</h2>
167 |         <h6>Veuillez saisir les champs pour le réinitialiser</h6>
168 | 
169 |         <form method="POST" class="text-center w-auto">
170 |             <div div class="form-group">
171 |                 <input required type="text" name="identifiant" placeholder="Identifiant *" value="" class="form-control-sm">
172 |             </div>
173 |             <div class="form-group">
174 |                 <input required type="text" name="email" value="" placeholder="E-mail *" class="form-control-sm">
175 |             </div>
176 |             <div class="form-group">
177 |                 <button id="resetButton" style="display:inline" type="submit" name="submit" value="reset" class="form-control-sm btn btn-sm btn-dark">Réinitialiser</button>
178 |             </div>
179 |             <div id="waiting" style="display:none; font-weight:bold">
180 |                 <span>veuillez patienter...</span>
181 |             </div>
182 |         </form>
183 |     </div>
184 | 
```

```
185 | 
```

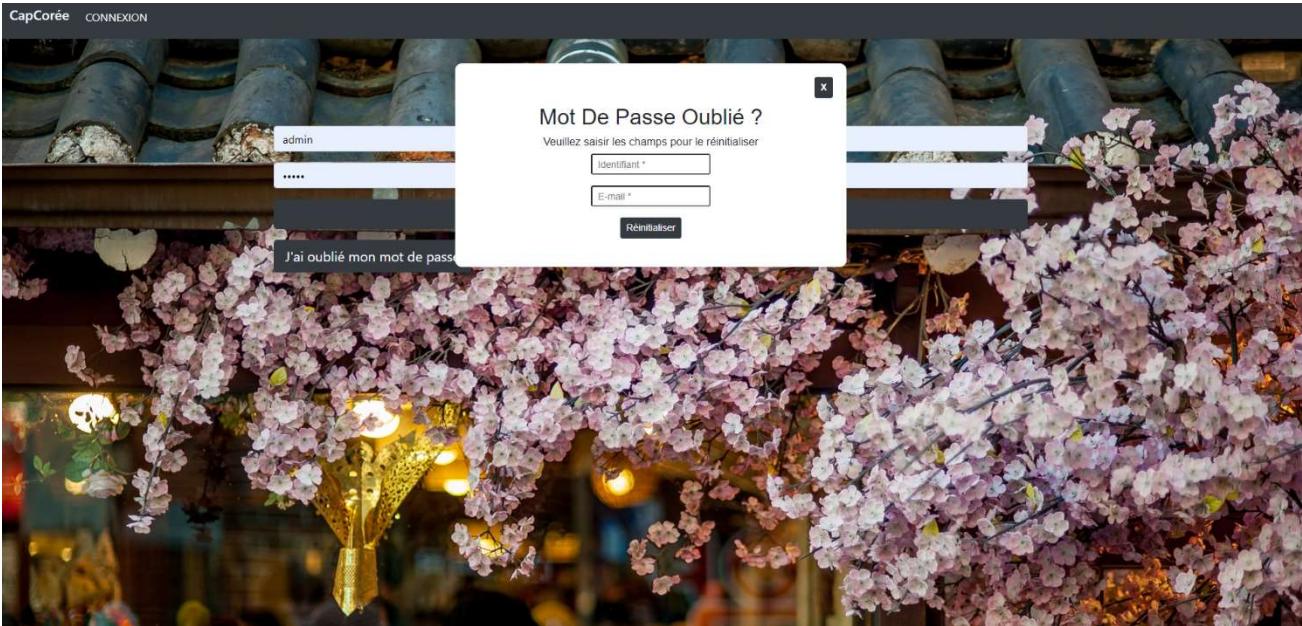
```
186 | 
```

Après avoir écrit le Modale de la ligne 165 jusqu'à 186 j'ai dû faire un peu de Css pour le design.

```
/* Voici le modal du MDP oublié sur la page LOGIN */
#modal{
    height: 300px;
    width: 30%;
    background-color: #fff;
    font-family: Helvetica, sans-serif;
    text-align: center;
    position: absolute;
    top: -400px;
    left: 35%;
    border-radius: 10px;
    transition: 0.5s;
}

#modal h2{
    padding-top: 60px;
}

#close { /* fermer le modale */
    margin-top: 20px;
    margin-right: 20px;
    font-weight: bolder;
    float: right;
}
```



Mot de passe oublié

Un identifiant de type id= modal a été donné à cette div qui permet d’interagir avec. En css j’ai donc donné une taille, une couleur ainsi qu’une certaine typo d’écriture.

On remarque que top est à -400, ce qui veut dire qu’il n’est pas visible (hors de notre navigateur web) et que transition est de 0.5seconde.

Effectivement le modal n’est pas visible dans un premier temps, c’est lorsque l’on cliquera sur le bouton mot de passe oublié que celui-ci apparaîtra. Il descendra du top vers le haut de la page comme sur la photo ci-dessus. Avec une transition qui prendra donc une demie seconde.

Et pour cela nous avons besoin de **Javascript**.

Remontons un peu plus haut pour voir le code et nous remarquerons qu’à la ligne 161 on peut voir qu’il y a un paramètre “onclick” qui fait appel à la fonction openModal() et également à la ligne 166 qui lui fait appel à closeModal() qui se trouve dans un fichier javascript.

Pour faire appel à ce fichier, il faut placer le chemin de celui-ci en bas de page.

```

189  <!-- Optional JavaScript -->
190  <script src="JavaScript/modalLogin.js" type="text/javascript"></script>
191  <!-- jQuery first, then Popper.js, then Bootstrap JS -->
192  <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965O:0rT7abK41J5tQIAqYgRVzbzo5smXKp4YFRvH+8abTE1Pi6jizo" crossorigin="anonymous"></script>
193  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-UO2eT0CpLqdS70hJ1ty5KVphtPhwVj9hO1c1HTMa3J0ZwrrQq4sf86dIHMDz0W1" crossorigin="anonymous"></script>
194  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-Jj5MVGdppSPX81rkibZUAYolly6oRQbVrJIEaFt/nJGZX-Ust4x0x1M+B0/jI04" crossorigin="anonymous"></script>
195  </body>
196  </html>
```

Ligne 190 source du fichier (le chemin) Dossier Javascript, et le fichier modalLogin.js

On remarque également 3 autres lignes : (une extension rajoute ces lignes à chaque page)

- 192 : appel de Jquery (librairie Javascript, pas utilisée dans mon projet)
- 193 : appel de Popper (librairie Javascript, pas utilisée dans mon projet)
- 194 : appel du framework Bootstrap

Nous voulons donc la le fichier javascript qui a été appelé quand nous avons cliqué sur le bouton "mot de passe oublié"

Voici donc les très courtes fonctions, qui permettent d'afficher ou de cacher l'affichage du modale.

```
1 // Permet d'afficher le modal MDP oublié
2 // Display the modal Forgot Password
3
4 function openModal(){
5     document.getElementById("modal").style.top = "100px";
6 }
7
8 // Permet de cacher le modal MDP oublié
9 // Hide the modal Forgot Password
10
11 function closeModal() {
12     document.getElementById("modal").style.top = "-400px";
13 }
```

▪ **Création de tableau**

Sur plusieurs pages je devais créer des tableaux pour afficher les informations de la base de données, j'ai donc utilisé le même design pour chaque page utilisant un tableau.

- Employé
- Hôtel
- Activité
- Historique
- Voyageur

Voici un exemple de tableau pour la page Employé :

Liste des Employés

+ Ajouter

Role	Nom	Prenom	Sexe	Action
Admin	super	admin	Homme	VOIR / MODIFIER / SUPPRIMER
Employé	Bernier	Brice	Homme	VOIR / MODIFIER / SUPPRIMER
Employé	Perillard	Laurette	Femme	VOIR / MODIFIER / SUPPRIMER
Employé	Choe	Jung	Homme	VOIR / MODIFIER / SUPPRIMER
Finance	Seo	Min-kyung	Femme	VOIR / MODIFIER / SUPPRIMER
Stagiaire	Dubost	Héloïse	Femme	VOIR / MODIFIER / SUPPRIMER
Guide	Giraud	Mégane	Femme	VOIR / MODIFIER / SUPPRIMER
Guide	Charpentier	Danièle	Femme	VOIR / MODIFIER / SUPPRIMER
Guide	Bechard	Élodie	Femme	VOIR / MODIFIER / SUPPRIMER
Guide	Grandis	Thibaud	Homme	VOIR / MODIFIER / SUPPRIMER
Guide	Deschanel	Matthieu	Homme	VOIR / MODIFIER / SUPPRIMER

Tableau liste employés avec données

On remarque que le tableau possède 5 colonnes, 4 servants à afficher quelques informations de la table EMPLOYES, et la 5ième pour afficher des boutons qui permettront de réaliser des actions comme Voir (Plus d'informations) Modifier et Supprimer.

Je montrerais dans la partie Back-end le code du tableau étant donné que ceux-ci sont dynamiques nous utilisons du PHP pour retourner les informations.

Il y a un module de recherche également en haut à gauche du tableau, selon les tableaux les recherches seront différentes (Nom,Ville, Email etc...)

La recherche a été créé avec un formulaire :

```

42 <form action="" method="POST">
43   <label for="search">Recherche</label>
44   <select class="curseur" name="search" id="search">
45     <option value="Role">Role</option>;
46     <option value="Nom">Nom</option>;
47     <option value="Prenom">Prenom</option>;
48     <option value="Sexe">Genre</option>;
49   </select>
50   <input name="search1" id="search" type="text" value=<?php echo $search1 ?>>
51   <input class="btn-sm btn-secondary rounded" type="submit" name="submit">
52 </form>
```

Fonction recherche employé

L42 – un formulaire avec aucune action, c'est-à-dire que nous restons sur la même page et la **Méthode Post** que nous verrons un peu plus tard.

L43 – un titre en l'occurrence ici “Recherche”

L44 - Nous créons une liste avec des options, comme dans ce cas, Role,Nom,Prenoms etc...

L50 – Étant le champs de recherche appelé INPUT en HTML (il existe plusieurs type d'input, comme text,password,submit)

L51 – Encore un INPUT mais cette fois de type Submit ce qui permettra de soumettre le formulaire et donc d'effectuer la recherche.

Le SELECT et l'INPUT de type text, on des valeurs, celles-ci nous serviront pour effectuer notre requête SQL pour demander au serveur les informations.

La page profil employé, comme son nom l'indique, permet d'avoir plus d'informations à propos de celui-ci avec sa photo par exemple.

CapCoree EMPLOYES COMPTA HISTORIQUE ALERTE HOTELS ACTIVITES VOYAGEURS PARAMETRES MON COMPTE DECONNEXION

Profil de l'employé



Role : Employe

Nom : Bernier

Prenom : Brice

Sexe : Homme

Identifiant : Bernier.Brice

Email : bernier-brice@capcoree.fr

Telephone : +82-3-079-3197

KakaoTalk : bernier32

Adresse : 1249, 대구, 수성구, 가천동, 96-6

AlienCard : HH7831822

Banque : Citibank Korea

NumeroCompte : 851-26756-030-61

[MODIFIER](#)

[SUPPRIMER](#)

[RETOUR](#)

Profil employé

En voila une autre mais cette fois avec les informations d'un voyageur. Dans un premier temps nous avions la liste de tous nos voyageurs dans un tableau et en cliquant sur le bouton VOIR nous avions accès à sa fiche comme celle-ci pour le voyageur **Bernard Laurent** :

CapCoree Employés Compta Historique Alertes Hôtels Activités Voyageurs Paramètres Mon Compte Déconnexion

Information Voyageur

[Voir Construction](#) [Créer Réservation](#)

Nom : Bernard
Prenom : Laurent
Sexe : Homme
Age : 44
Email : laurent.bj@gmail.com
WhatsApp : bernard34
Telephone : 04363256130
Passeport : ouewnfz2323
Banque : Societe General
Numero Compte : Fr177085231004410
Commentaire :

Liste des Membres

+ Ajouter

Nom	Prenom	Age	Passeport	Actions
Bernard	Lucie	43	Fr4234224345433	MODIFIER / SUPPRIMER
Bernard	sophie	34	Fr4234244353453	MODIFIER / SUPPRIMER

[RETOUR](#)

En plus de fournir plus d'informations sur le voyageur, nous avons également un tableau qui cette fois représente les personnes qui l'accompagnent. Mais également 2 boutons vert qui permettent de Créer/Voir une construction/réservation (ceci étant des informations supplémentaires sur le voyage)

Là encore ces 2 boutons sont dynamiques, au chargement de la page il y a une requête en PHP qui est fait au serveur, si celui-ci nous retourne quelque chose alors le bouton affichera "Voir" sinon celui-ci sera "Créer".

```

73 |         <a class="btn btn-success btn-lg" href="construction.php?id=<?php echo $id ?>">
74 |             <h1 id="btnConst">Construction</h1>
75 |         </a>
76 |         <a class="btn btn-success btn-lg" href="reservation.php?id=<?php echo $id ?>">
77 |             <h1 id="btnReser">Réservation</h1>
78 |         </a><br>
79 |         <?php
80 |             if(!empty($_GET['id'])){
81 |                 $id = checkInput($_GET['id']); // id de voyageurs
82 |             }
83 |             // On récupère les infos du voyageurs
84 |             $db = Database::connect();
85 |
86 |             // On vérifie si il a déjà une fiche construction
87 |             $sqlConst = $db->prepare("SELECT COUNT(*) FROM construction WHERE IdVoyageurs = ? ");
88 |             $sqlConst ->execute(array($id));
89 |
90 |             $tailleConst = $sqlConst->fetchColumn();
91 |
92 |             if($tailleConst > 0){ // SI OUI alors on modifie le bouton
93 |             ?>
94 |                 <script type="text/javascript">
95 |                     document.getElementById("btnConst").innerHTML = "Voir Construction";
96 |                 </script>
97 |             }
98 |             else{ // SI NON alors on modifie le bouton
99 |             ?>
100 |                 <script type="text/javascript">
101 |                     document.getElementById("btnConst").innerHTML = "Créer Construction";
102 |
103 |                 </script>
104 |             };
105 |
106 |             $sqlReser = $db->prepare("SELECT COUNT(*) FROM reservation WHERE IdVoyageurs = ? ");
107 |             $sqlReser ->execute(array($id));
108 |
109 |             $tailleReser = $sqlReser->fetchColumn();
110 |
111 |             if($tailleReser > 0){ // SI OUI alors on modifie le bouton
112 |             ?>
113 |                 <script type="text/javascript">
114 |                     document.getElementById("btnReser").innerHTML = "Voir Réservation";
115 |                 </script>
116 |             }
117 |             else{ // SI NON alors on modifie le bouton
118 |             ?>
119 |                 <script type="text/javascript">
120 |                     document.getElementById("btnReser").innerHTML = "Créer Réservation";
121 |
122 |                 </script>
123 |             };
124 |
125 |             Database::disconnect();
126 |
127 |
128 |
129 |
130 |
131 |
132 |
133 |
134 |
135 |
136 |
137 |
138 |
139 |
140 |
141 |
142 |
143 |
144 |
145 |
146 |
147 |
148 |
149 |
150 |
151 |
152 |
153 |
154 |
155 |
156 |
157 |
158 |
159 |
160 |
161 |
162 |
163 |
164 |
165 |
166 |
167 |
168 |
169 |
170 |
171 |
172 |
173 |
174 |
175 |
176 |
177 |
178 |
179 |
180 |
181 |
182 |
183 |
184 |
185 |
186 |
187 |
188 |
189 |
190 |
191 |
192 |
193 |
194 |
195 |
196 |
197 |
198 |
199 |
200 |
201 |
202 |
203 |
204 |
205 |
206 |
207 |
208 |
209 |
210 |
211 |
212 |
213 |
214 |
215 |
216 |
217 |
218 |
219 |
220 |
221 |
222 |
223 |
224 |
225 |
226 |
227 |
228 |
229 |
230 |
231 |
232 |
233 |
234 |
235 |
236 |
237 |
238 |
239 |
240 |
241 |
242 |
243 |
244 |
245 |
246 |
247 |
248 |
249 |
250 |
251 |
252 |
253 |
254 |
255 |
256 |
257 |
258 |
259 |
260 |
261 |
262 |
263 |
264 |
265 |
266 |
267 |
268 |
269 |
270 |
271 |
272 |
273 |
274 |
275 |
276 |
277 |
278 |
279 |
280 |
281 |
282 |
283 |
284 |
285 |
286 |
287 |
288 |
289 |
290 |
291 |
292 |
293 |
294 |
295 |
296 |
297 |
298 |
299 |
300 |
301 |
302 |
303 |
304 |
305 |
306 |
307 |
308 |
309 |
310 |
311 |
312 |
313 |
314 |
315 |
316 |
317 |
318 |
319 |
320 |
321 |
322 |
323 |
324 |
325 |
326 |
327 |
328 |
329 |
330 |
331 |
332 |
333 |
334 |
335 |
336 |
337 |
338 |
339 |
340 |
341 |
342 |
343 |
344 |
345 |
346 |
347 |
348 |
349 |
350 |
351 |
352 |
353 |
354 |
355 |
356 |
357 |
358 |
359 |
360 |
361 |
362 |
363 |
364 |
365 |
366 |
367 |
368 |
369 |
370 |
371 |
372 |
373 |
374 |
375 |
376 |
377 |
378 |
379 |
380 |
381 |
382 |
383 |
384 |
385 |
386 |
387 |
388 |
389 |
390 |
391 |
392 |
393 |
394 |
395 |
396 |
397 |
398 |
399 |
400 |
401 |
402 |
403 |
404 |
405 |
406 |
407 |
408 |
409 |
410 |
411 |
412 |
413 |
414 |
415 |
416 |
417 |
418 |
419 |
420 |
421 |
422 |
423 |
424 |
425 |
426 |
427 |
428 |
429 |
430 |
431 |
432 |
433 |
434 |
435 |
436 |
437 |
438 |
439 |
440 |
441 |
442 |
443 |
444 |
445 |
446 |
447 |
448 |
449 |
450 |
451 |
452 |
453 |
454 |
455 |
456 |
457 |
458 |
459 |
460 |
461 |
462 |
463 |
464 |
465 |
466 |
467 |
468 |
469 |
470 |
471 |
472 |
473 |
474 |
475 |
476 |
477 |
478 |
479 |
480 |
481 |
482 |
483 |
484 |
485 |
486 |
487 |
488 |
489 |
490 |
491 |
492 |
493 |
494 |
495 |
496 |
497 |
498 |
499 |
500 |
501 |
502 |
503 |
504 |
505 |
506 |
507 |
508 |
509 |
510 |
511 |
512 |
513 |
514 |
515 |
516 |
517 |
518 |
519 |
520 |
521 |
522 |
523 |
524 |
525 |
526 |
527 |
528 |
529 |
530 |
531 |
532 |
533 |
534 |
535 |
536 |
537 |
538 |
539 |
540 |
541 |
542 |
543 |
544 |
545 |
546 |
547 |
548 |
549 |
550 |
551 |
552 |
553 |
554 |
555 |
556 |
557 |
558 |
559 |
560 |
561 |
562 |
563 |
564 |
565 |
566 |
567 |
568 |
569 |
570 |
571 |
572 |
573 |
574 |
575 |
576 |
577 |
578 |
579 |
580 |
581 |
582 |
583 |
584 |
585 |
586 |
587 |
588 |
589 |
589 |
590 |
591 |
592 |
593 |
594 |
595 |
596 |
597 |
598 |
599 |
599 |
600 |
601 |
602 |
603 |
604 |
605 |
606 |
607 |
608 |
609 |
609 |
610 |
611 |
612 |
613 |
614 |
615 |
616 |
617 |
618 |
619 |
619 |
620 |
621 |
622 |
623 |
624 |
625 |
626 |
627 |
628 |
629 |
629 |
630 |
631 |
632 |
633 |
634 |
635 |
636 |
637 |
638 |
639 |
639 |
640 |
641 |
642 |
643 |
644 |
645 |
646 |
647 |
648 |
649 |
649 |
650 |
651 |
652 |
653 |
654 |
655 |
656 |
657 |
658 |
659 |
659 |
660 |
661 |
662 |
663 |
664 |
665 |
666 |
667 |
668 |
669 |
669 |
670 |
671 |
672 |
673 |
674 |
675 |
676 |
677 |
678 |
679 |
679 |
680 |
681 |
682 |
683 |
684 |
685 |
686 |
687 |
688 |
689 |
689 |
690 |
691 |
692 |
693 |
694 |
695 |
696 |
697 |
698 |
699 |
699 |
700 |
701 |
702 |
703 |
704 |
705 |
706 |
707 |
708 |
709 |
709 |
710 |
711 |
712 |
713 |
714 |
715 |
716 |
717 |
718 |
719 |
719 |
720 |
721 |
722 |
723 |
724 |
725 |
726 |
727 |
728 |
729 |
729 |
730 |
731 |
732 |
733 |
734 |
735 |
736 |
737 |
738 |
738 |
739 |
740 |
741 |
742 |
743 |
744 |
745 |
746 |
747 |
748 |
749 |
749 |
750 |
751 |
752 |
753 |
754 |
755 |
756 |
757 |
758 |
759 |
759 |
760 |
761 |
762 |
763 |
764 |
765 |
766 |
767 |
768 |
769 |
769 |
770 |
771 |
772 |
773 |
774 |
775 |
776 |
777 |
778 |
779 |
779 |
780 |
781 |
782 |
783 |
784 |
785 |
786 |
787 |
788 |
789 |
789 |
790 |
791 |
792 |
793 |
794 |
795 |
796 |
797 |
798 |
799 |
799 |
800 |
801 |
802 |
803 |
804 |
805 |
806 |
807 |
808 |
809 |
809 |
810 |
811 |
812 |
813 |
814 |
815 |
816 |
817 |
818 |
819 |
819 |
820 |
821 |
822 |
823 |
824 |
825 |
826 |
827 |
828 |
829 |
829 |
830 |
831 |
832 |
833 |
834 |
835 |
836 |
837 |
838 |
839 |
839 |
840 |
841 |
842 |
843 |
844 |
845 |
846 |
847 |
848 |
849 |
849 |
850 |
851 |
852 |
853 |
854 |
855 |
856 |
857 |
858 |
859 |
859 |
860 |
861 |
862 |
863 |
864 |
865 |
866 |
867 |
868 |
869 |
869 |
870 |
871 |
872 |
873 |
874 |
875 |
876 |
877 |
878 |
879 |
879 |
880 |
881 |
882 |
883 |
884 |
885 |
886 |
887 |
888 |
889 |
889 |
890 |
891 |
892 |
893 |
894 |
895 |
896 |
897 |
898 |
899 |
899 |
900 |
901 |
902 |
903 |
904 |
905 |
906 |
907 |
908 |
909 |
909 |
910 |
911 |
912 |
913 |
914 |
915 |
916 |
917 |
918 |
919 |
919 |
920 |
921 |
922 |
923 |
924 |
925 |
926 |
927 |
928 |
929 |
929 |
930 |
931 |
932 |
933 |
934 |
935 |
936 |
937 |
938 |
939 |
939 |
940 |
941 |
942 |
943 |
944 |
945 |
946 |
947 |
948 |
949 |
949 |
950 |
951 |
952 |
953 |
954 |
955 |
956 |
957 |
958 |
959 |
959 |
960 |
961 |
962 |
963 |
964 |
965 |
966 |
967 |
968 |
969 |
969 |
970 |
971 |
972 |
973 |
974 |
975 |
976 |
977 |
978 |
979 |
979 |
980 |
981 |
982 |
983 |
984 |
985 |
986 |
987 |
988 |
989 |
989 |
990 |
991 |
992 |
993 |
994 |
995 |
996 |
997 |
998 |
999 |
999 |
1000 |
1001 |
1002 |
1003 |
1004 |
1005 |
1006 |
1007 |
1008 |
1009 |
1009 |
1010 |
1011 |
1012 |
1013 |
1014 |
1015 |
1016 |
1017 |
1018 |
1019 |
1019 |
1020 |
1021 |
1022 |
1023 |
1024 |
1025 |
1026 |
1027 |
1028 |
1029 |
1029 |
1030 |
1031 |
1032 |
1033 |
1034 |
1035 |
1036 |
1037 |
1038 |
1039 |
1039 |
1040 |
1041 |
1042 |
1043 |
1044 |
1045 |
1046 |
1047 |
1048 |
1049 |
1049 |
1050 |
1051 |
1052 |
1053 |
1054 |
1055 |
1056 |
1057 |
1058 |
1059 |
1059 |
1060 |
1061 |
1062 |
1063 |
1064 |
1065 |
1066 |
1067 |
1068 |
1069 |
1069 |
1070 |
1071 |
1072 |
1073 |
1074 |
1075 |
1076 |
1077 |
1078 |
1078 |
1079 |
1080 |
1081 |
1082 |
1083 |
1084 |
1085 |
1086 |
1087 |
1087 |
1088 |
1089 |
1089 |
1090 |
1091 |
1092 |
1093 |
1094 |
1095 |
1095 |
1096 |
1097 |
1098 |
1099 |
1099 |
1100 |
1101 |
1102 |
1103 |
1104 |
1105 |
1106 |
1107 |
1108 |
1109 |
1109 |
1110 |
1111 |
1112 |
1113 |
1114 |
1115 |
1116 |
1117 |
1118 |
1119 |
1119 |
1120 |
1121 |
1122 |
1123 |
1124 |
1125 |
1126 |
1127 |
1128 |
1129 |
1129 |
1130 |
1131 |
1132 |
1133 |
1134 |
1135 |
1136 |
1137 |
1138 |
1139 |
1139 |
1140 |
1141 |
1142 |
1143 |
1144 |
1145 |
1146 |
1147 |
1148 |
1149 |
1149 |
1150 |
1151 |
1152 |
1153 |
1154 |
1155 |
1156 |
1157 |
1158 |
1159 |
1159 |
1160 |
1161 |
1162 |
1163 |
1164 |
1165 |
1166 |
1167 |
1168 |
1169 |
1169 |
1170 |
1171 |
1172 |
1173 |
1174 |
1175 |
1176 |
1177 |
1178 |
1178 |
1179 |
1180 |
1181 |
1182 |
1183 |
1184 |
1185 |
1186 |
1187 |
1187 |
1188 |
1189 |
1189 |
1190 |
1191 |
1192 |
1193 |
1194 |
1195 |
1196 |
1197 |
1198 |
1198 |
1199 |
1200 |
1201 |
1202 |
1203 |
1204 |
1205 |
1206 |
1207 |
1208 |
1209 |
1209 |
1210 |
1211 |
1212 |
1213 |
1214 |
1215 |
1216 |
1217 |
1218 |
1219 |
1219 |
1220 |
1221 |
1222 |
1223 |
1224 |
1225 |
1226 |
1227 |
1228 |
1229 |
1229 |
1230 |
1231 |
1232 |
1233 |
1234 |
1235 |
1236 |
1237 |
1238 |
1239 |
1239 |
1240 |
1241 |
1242 |
1243 |
1244 |
1245 |
1246 |
1247 |
1248 |
1249 |
1249 |
1250 |
1251 |
1252 |
1253 |
1254 |
1255 |
1256 |
1257 |
1258 |
1259 |
1259 |
1260 |
1261 |
1262 |
1263 |
1264 |
1265 |
1266 |
1267 |
1268 |
1269 |
1269 |
1270 |
1271 |
1272 |
1273 |
1274 |
1275 |
1276 |
1277 |
1278 |
1278 |
1279 |
1280 |
1281 |
1282 |
1283 |
1284 |
1285 |
1286 |
1287 |
1288 |
1288 |
1289 |
1290 |
1291 |
1292 |
1293 |
1294 |
1295 |
1296 |
1297 |
1298 |
1298 |
1299 |
1300 |
1301 |
1302 |
1303 |
1304 |
1305 |
1306 |
1307 |
1308 |
1309 |
1309 |
1310 |
1311 |
1312 |
1313 |
1314 |
1315 |
1316 |
1317 |
1318 |
1319 |
1319 |
1320 |
1321 |
1322 |
1323 |
1324 |
1325 |
1326 |
1327 |
1328 |
1329 |
1329 |
1330 |
1331 |
1332 |
1333 |
1334 |
1335 |
1336 |
1337 |
1338 |
1339 |
1339 |
1340 |
1341 |
1342 |
1343 |
1344 |
1345 |
1346 |
1347 |
1348 |
1349 |
1349 |
1350 |
1351 |
1352 |
1353 |
1354 |
1355 |
1356 |
1357 |
1358 |
1359 |
1359 |
1360 |
1361 |
1362 |
1363 |
1364 |
1365 |
1366 |
1367 |
1368 |
1369 |
1369 |
1370 |
1371 |
1372 |
1373 |
1374 |
1375 |
1376 |
1377 |
1378 |
1378 |
1379 |
1380 |
1381 |
1382 |
1383 |
1384 |
1385 |
1386 |
1387 |
1388 |
1388 |
1389 |
1390 |
1391 |
1392 |
1393 |
1394 |
1395 |
1396 |
1397 |
1398 |
1398 |
1399 |
1400 |
1401 |
1402 |
1403 |
1404 |
1405 |
1406 |
1407 |
1408 |
1409 |
1409 |
1410 |
1411 |
1412 |
1413 |
1414 |
1415 |
1416 |
1417 |
1418 |
1419 |
1419 |
1420 |
1421 |
1422 |
1423 |
1424 |
1425 |
1426 |
1427 |
1428 |
1429 |
1429 |
1430 |
1431 |
1432 |
1433 |
1434 |
1435 |
1436 |
1437 |
1438 |
1439 |
1439 |
1440 |
1441 |
1442 |
1443 |
1444 |
1445 |
1446 |
1447 |
1448 |
1449 |
1449 |
1450 |
1451 |
1452 |
1453 |
1454 |
1455 |
1456 |
1457 |
1458 |
1459 |
1459 |
1460 |
1461 |
1462 |
1463 |
1464 |
1465 |
1466 |
1467 |
1468 |
1469 |
1469 |
1470 |
1471 |
1472 |
1473 |
1474 |
1475 |
1476 |
1477 |
1478 |
1478 |
1479 |
1480 |
1481 |
1482 |
1483 |
1484 |
1485 |
1486 |
1487 |
1488 |
1488 |
1489 |
1490 |
1491 |
1492 |
1493 |
1494 |
1495 |
1496 |
1497 |
1498 |
1498 |
1499 |
1500 |
1501 |
1502 |
1503 |
1504 |
1505 |
1506 |
1507 |
1508 |
1509 |
1509 |
1510 |
1511 |
1512 |
1513 |
1514 |
1515 |
1516 |
1517 |
1518 |
1519 |
1519 |
1520 |
1521 |
1522 |
1523 |
1524 |
1525 |
1526 |
1527 |
1528 |
1529 |
1529 |
1530 |
1531 |
1532 |
1533 |
1534 |
1535 |
1536 |
1537 |
1538 |
1539 |
1539 |
1540 |
1541 |
1542 |
1543 |
1544 |
1545 |
1546 |
1547 |
1548 |
1549 |
1549 |
1550 |
1551 |
1552 |
1553 |
1554 |
1555 |
1556 |
1557 |
1558 |
1559 |
1559 |
1560 |
1561 |
1562 |
1563 |
1564 |
1565 |
1566 |
1567 |
1568 |
1569 |
1569 |
1570 |
1571 |
1572 |
1573 |
1574 |
1575 |
1576 |
1577 |
1578 |
1578 |
1579 |
1580 |
1581 |
1582 |
1583 |
1584 |
1585 |
1586 |
1587 |
1588 |
1588 |
1589 |
1590 |
1591 |
1592 |
1593 |
1594 |
1595 |
1596 |
1597 |
1598 |
1598 |
1599 |
1600 |
1601 |
1602 |
1603 |
1604 |
1605 |
1606 |
1607 |
1608 |
1609 |
1609 |
1610 |
1611 |
1612 |
1613 |
1614 |
1615 |
1616 |
1617 |
1618 |
1619 |
1619 |
1620 |
1621 |
1622 |
1623 |
1624 |
1625 |
1626 |
1627 |
1628 |
1629 |
1629 |
1630 |
1631 |
1632 |
1633 |
1634 |
1635 |
1636 |
1637 |
1638 |
1639 |
1639 |
1640 |
1641 |
1642 |
1643 |
1644 |
1645 |
1646 |
1647 |
1648 |
1649 |
1649 |
1650 |
1651 |
1652 |
1653 |
1654 |
1655 |
1656 |
1657 |
1658 |
1659 |
1659 |
1660 |
1661 |
1662 |
1663 |
1664 |
1665 |
1666 |
1667 |
1668 |
1669 |
1669 |
1670 |
1671 |
1672 |
1673 |
1674 |
1675 |
1676 |
1677 |
1678 |
1678 |
1679 |
1680 |
1681 |
1682 |
1683 |
1684 |
1685 |
1686 |
1687 |
1688 |
1688 |
1689 |
1690 |
1691 |
1692 |
1693 |
1694 |
1695 |
1696 |
1697 |
1698 |
1698 |
1699 |
1700 |
1701 |
1702 |
1703 |
1704 |
1705 |
1706 |
1707 |
1708 |
1709 |
1709 |
1710 |
1711 |
1712 |
1713 |
1714 |
1715 |
1716 |
1717 |
1718 |
1719 |
1719 |
1720 |
1721 |
1722 |
1723 |
1724 |
1725 |
1726 |
1727 |
1728 |
1729 |
1729 |
1730 |
1731 |
1732 |
1733 |
1734 |
1735 |
1736 |
1737 |
1738 |
1739 |
1739 |
1740 |
1741 |
1742 |
1743 |
1744 |
1745 |
1746 |
1747 |
1748 |
1749 |
1749 |
1750 |
1751 |
1752 |
1753 |
1754 |
1755 |
1756 |
1757 |
1758 |
1759 |
1759 |
1760 |
1761 |
1762 |
1763 |
1764 |
1765 |
1766 |
1767 |
1768 |
1769 |
1769 |
1770 |
1771 |
1772 |
1773 |
1774 |
1775 |
1776 |
1777 |
1778 |
1778 |
1779 |
1780 |
1781 |
1782 |
1783 |
1784 |
1785 |
1786 |
1787 |
1788 |
1788 |
1789 |
1790 |
1791 |
1792 |
1793 |
1794 |
1795 |
1796 |
1797 |
1798 |
1798 |
1799 |
1800 |
1801 |
1802 |
1803 |
1804 |
1805 |
1806 |
1807 |
1808 |
1809 |
1809 |
1810 |
1811 |
1812 |
1813 |
1814 |
1815 |
1816 |
1817 |
1818 |
1819 |
1819 |
1820 |
1821 |
1822 |
1823 |
1824 |
1825 |
1826 |
1827 |
1828 |
1829 |
1829 |
1830 |
1831 |
1832 |
1833 |
1834 |
1835 |
1836 |
1837 |
1838 |
1839 |
1839 |
1840 |
1841 |
1842 |
1843 |
1844 |
1845 |
1846 |
1847 |
1848 |
1849 |
1849 |
1850 |
1851 |
1852 |
1853 |
1854 |
1855 |
1856 |
1857 |
1858 |
1859 |
1859 |
1860 |
1861 |
1862 |
1863 |
1864 |
1865 |
1866 |
1867 |
1868 |
1869 |
1869 |
1870 |
1871 |
1872 |
1873 |
1874 |
1875 |
1876 |
1877 |
1878 |
1878 |
1879 |
1880 |
1881 |
1882 |
1883 |
1884 |
1885 |
1886 |
1887 |
1888 |
1888 |
1889 |
1890 |
1891 |
1892 |
1893 |
1894 |
1895 |
1896 |
1897 |
1898 |
1898 |
1899 |
1900 |
1901 |
1902 |
1903 |
1904 |
1905 |
1906 |
1907 |
1908 |
1909 |
1909 |
1910 |
1911 |
1912 |
1913 |
1914 |
1915 |
1916 |
1917 |
1918 |
1919 |
1919 |
1920 |
1921 |
1922 |
1923 |
1924 |
1925 |
1926 |
1927 |
1928 |
1929 |
1929 |
1930 |
1931 |
1932 |
1933 |
1934 |
1935 |
1936 |
1937 |
1938 |
1939 |
1939 |
1940 |
1941 |
1942 |
1943 |
1944 |
1945 |
1946 |
1947 |
1948 |
1949 |
1949 |
1950 |
1951 |
1952 |
1953 |
1954 |
1955 |
1956 |
1957 |
1958 |
1959 |
1959 |
1960 |
1961 |
1962 |
1963 |
1964 |
1965 |
1966 |
1967 |
1968 |
1969 |
1969 |
1970 |
1971 |
1972 |
1973 |
1974 |
1975 |
1976 |
1977 |
1978 |
1978 |
1979 |
1980 |
1981 |
1982 |
1983 |
1984 |
1985 |
1986 |
1987 |
1988 |
1988 |
1989 |
1990 |
1991 |
1992 |
1993 |
1994 |
1995 |
1996 |
1997 |
1998 |
1998 |
1999 |
2000 |
2001 |
2002 |
2003 |
2004 |
2005 |
2006 |
2007 |
2008 |
2009 |
2009 |
2010 |
2011 |
2012 |
2013 |
2014 |
2015 |
2016 |
2017 |
2018 |
2019 |
2019 |
2020 |
2021 |
2022 |
2023 |
2024 |
2025 |
2026 |
2027 |
2028 |
2029 |
2029 |
2030 |
2031 |
2032 |
2033 |
2034 |
2035 |
2036 |
2037 |
2038 |
2039 |
2039 |
2040 |
2041 |
2042 |
2043 |
2044 |
2045 |
2046 |
2047 |
2048 |
2049 |
2049 |
2050 |
2051 |
2052 |
2053 |
2054 |
2055 |
2056 |
2057 |
2058 |
2059 |
2059 |
2060 |
2061 |
2062 |
2063 |
2064 |
2065 |
2066 |
2067 |
2068 |
2069 |
2069 |
2070 |
2071 |
2072 |
2073 |
2074 |
2075 |
2076 |
2077 |
2078 |
2078 |
2079 |
2080 |
2081 |
2082 |
2083 |
2084 |
2085 |
2086 |
2087 |
2088 |
2088 |
2089 |
2090 |
2091 |
2092 |
2093 |
2094 |
2095 |
2096 |
2097 |
2098 |
2098 |
2099 |
2100 |
2101 |
2102 |
2103 |
2104 |
2105 |
2106 |
2107 |
2108 |
2109 |
2109 |
2110 |
2111 |
2112 |
2113 |
2114 |
2115 |
2116 |
2117 |
2118 |
2119 |
2119 |
2120 |
2121 |
2122 |
2123 |
2124 |
2125 |
2126 |
2127 |
2128 |
2129 |
2129 |
2130 |
2131 |
2132 |
2133 |
2134 |
2135 |
2136 |
2137 |
2138 |
2139 |
2139 |
2140 |
2141 |
2142 |
2143 |
2144 |
2145 |
2146 |
2147 |
2148 |
2149 |
2149 |
2150 |
2151 |
2152 |
2153 |
2154 |
2155 |
2156 |
2157 |
2158 |
2159 |
2159 |
2160 |
2161 |
2162 |
2163 |
2164 |
2165 |
2166 |
2167 |
2168 |
2169 |
2169 |
2170 |
2171 |
2172 |
2173 |
2174 |
2175 |
2176 |
2177 |
2178 |
2178 |
2179 |
2180 |
2181 |
2182 |
2183 |
2184 |
2185 |
2186 |
2187 |
2188 |
2188 |
2189
```

J'ai décidé de mettre mon code JAVASCRIPT au milieu de mon code HTML, on remarque même qu'il y'a également du PHP.

A la ligne 73 et 76 j'ai créé un lien, et à l'intérieur de ce lien j'ai créé un titre H1 avec un identifiant unique. C'est ce H1 que nous modifions en fonction du résultat de la requête.

L88/89 on fait la requête en PHP,

SI retourne valeurs alors L96 on affichera "VOIR"

SINON L103 on affichera "Créer "

- **Le Framework Bootstrap 4**

Je cite Wikipedia :

«Bootstrap est une collection d'outils utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc.) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option. C'est l'un des projets les plus populaires sur la plate-forme de gestion de développement GitHub. »

Concrètement, Bootstrap nous permet de gagner beaucoup de temps car nous n'avons pas à créer les classes pour le design en Css.

Prenons par exemple les boutons dans un tableau :



Si je n'utilisais que du HTML pour créer ma page Web j'écrirai ce code :

```
<a id="btnModifier" href="#">MODIFIER</a>
```

En donnant un identifiant unique qui me permettrais de lui donner du style avec du css dans un fichier CSS. (Un fichier css étant placé dans le head d'un fichier html)

Puis j'aurai donc dû faire le design en changer la taille, la couleur de fond etc...

Et donc Bootstrap nous permet de faire ça plus rapidement en utilisant directement les classes pré-faites.

Il faut dans un premier temps ajouter cette ligne dans le Head :

```
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-gg0yR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZ
w1T" crossorigin="anonymous">
```

Et ensuite au lieu de donner un id pour le modifier dans le fichier css, nous utiliserons les classes comme ceci :

```
<a class="btn btn-info btn-lg"
href="employeesupdate.php?id=<?php echo $id ?>">MODIFIER</a>
```

On voit class avec 3 paramètres :

Btn = on utilise la classe Bouton donc celui-ci récit un design de bouton

Btn-info = ceci affectera la couleur en l'occurrence bleu, si au lieu du mot info nous avions écrit "danger" le bouton serait rouge.

Btn-lg = ceci affectera la taille, lg voulant dire large, si nous avions mis sm (small) le bouton serait de plus petite taille.

Nous pouvons bien sûr modifier ce lien avec une des classes que nous aurions pu écrire.

Bootstrap permet également de faire du responsive facilement. Un site responsive signifie que celui-ci est capable de s'adapter à la taille d'écran, pensons que l'on affiche pas de la même façon un site sur un écran 27 pouces par exemple et sur un écran de smartphone.

Dans mon cas, le site n'a pas pour vocation d'être utilisé sur téléphone, mais seulement sur pc donc ceci ne m'a pas vraiment servi. Sauf pour le positionnement par exemple de 2 colonnes.

```

44   <div class="container-fluid">
45     <div class="row text-center center-block">
46       <section class="col-sm-12 col-md-6">
47         <br>
48         <h1>Information Voyageur</h1>
49
50         <br><br>
51
52         <label class="font-weight-bold" for="">Nom : </label><?php echo " " . $item['Nom']; ?><br>
53         <label class="font-weight-bold" for="">Prenom : </label><?php echo " " . $item['Prenom']; ?><br>
54         <label class="font-weight-bold" for="">Sexe : </label><?php echo " " . $item['Sexe']; ?><br>
55         <label class="font-weight-bold" for="">Age : </label><?php echo " " . $item['Age']; ?><br>
56         <label class="font-weight-bold" for="">Email : </label><?php echo " " . $item['Email']; ?><br>
57         <label class="font-weight-bold" for="">WhatsApp : </label><?php echo " " . $item['WhatsApp']; ?><br>
58         <label class="font-weight-bold" for="">Telephone : </label><?php echo " " . $item['Telephone']; ?><br>
59         <label class="font-weight-bold" for="">Passeport : </label><?php echo " " . $item['Passeport']; ?><br>
60         <label class="font-weight-bold" for="">Banque : </label><?php echo " " . $item['Banque']; ?><br>
61         <label class="font-weight-bold" for="">Numero Compte :
62           </label><?php echo " " . $item['NumeroCompte']; ?><br>
63         <label class="font-weight-bold" for="">Commentaire :
64           </label><?php echo " " . $item['Commentaire']; ?><br>
65
66
67         <br><br>
68       </section>
69
70     <section class="col-sm-12 col-md-6">
71

```

Pour rendre son site responsive, il faut placer ses éléments Div,Section Article etc... à l'intérieur d'une boîte dirons-nous.

Pour créer cette boîte, nous allons ajouter une div avec comme classe “container-fluid” pour avoir toute la largeur de notre page. Imaginons que notre page va être découpée en 12 colonnes de taille égale. (Ceci va nous servir juste après)

Ensuite il faut créer une section ou un article en utilisant encore des classes de Bootstrap.

Col-sm-12 : Signifie que SI l'écran est petit(sm) alors on affichera la SECTION sur 12 colonnes, donc sur toute la largeur de l'écran. Les sections seront donc les unes sous les autres.

Col-md-6 : Signifie que SI l'écran est de taille moyenne(md) alors on affichera la SECTION sur 6 colonnes, donc la moitié de l'écran. On pourra alors afficher une deuxième section juste à côté l'une de l'autre.

On pourrait également changer les chiffres, par exemple écrire **col-md-4**, et donc sur un écran de taille moyenne, la section prendrait 4 colonnes, nous pourrions donc en afficher 3 par largeur, tant que le chiffre ne dépasse pas 12.

Voilà ce qu'il en est de la partie Front-End, je ne détaillerais pas plus de page étant donné que beaucoup d'entre elles se ressemblent, les informations de la base changent mais le design reste le même.

VI. Développement Back-End

- La base de données

J'ai développé ce projet sur mon ordinateur, et non sur un serveur distant. De ce fait, j'ai choisi le programme WAMPSEVER qui nous permet de simuler un serveur et de travailler sur un poste en LOCAL (sur notre machine).

L'avantage de travailler en local étant que je n'ai pas à transférer mes fichiers à chaque modification, que ce logiciel WampServer possède également un gestionnaire de base de données PhpMyAdmin.

The screenshot shows the PhpMyAdmin interface for a MySQL server. On the left, a sidebar lists databases: Nouvelle base de données, blog, burgercode, cap, capsule, faker, information_schema, modèle, mysql, performance_schema, store, sys. The 'capsule' database is selected. The main area displays a table of tables from the 'capsule' database, including: activités, chambre, construction, historique_pale, hotel, listeemploye, mailcontact, newpale, palevalide, prixprestation, reservation, voyageurs, voyageursfamille. The table has columns: Table, Action, Lignes, Type, Interclassement, Taille, Perte. At the bottom, there's a 'Nouvelle table' form with 'Nom:' and 'Nombre de colonnes: 4' fields, and a 'Exécuter' button.

Vue d'ensemble des différentes BDD et Tables

Comme indiqué plus haut j'utiliserais le langage SQL pour communiquer avec la base de données.

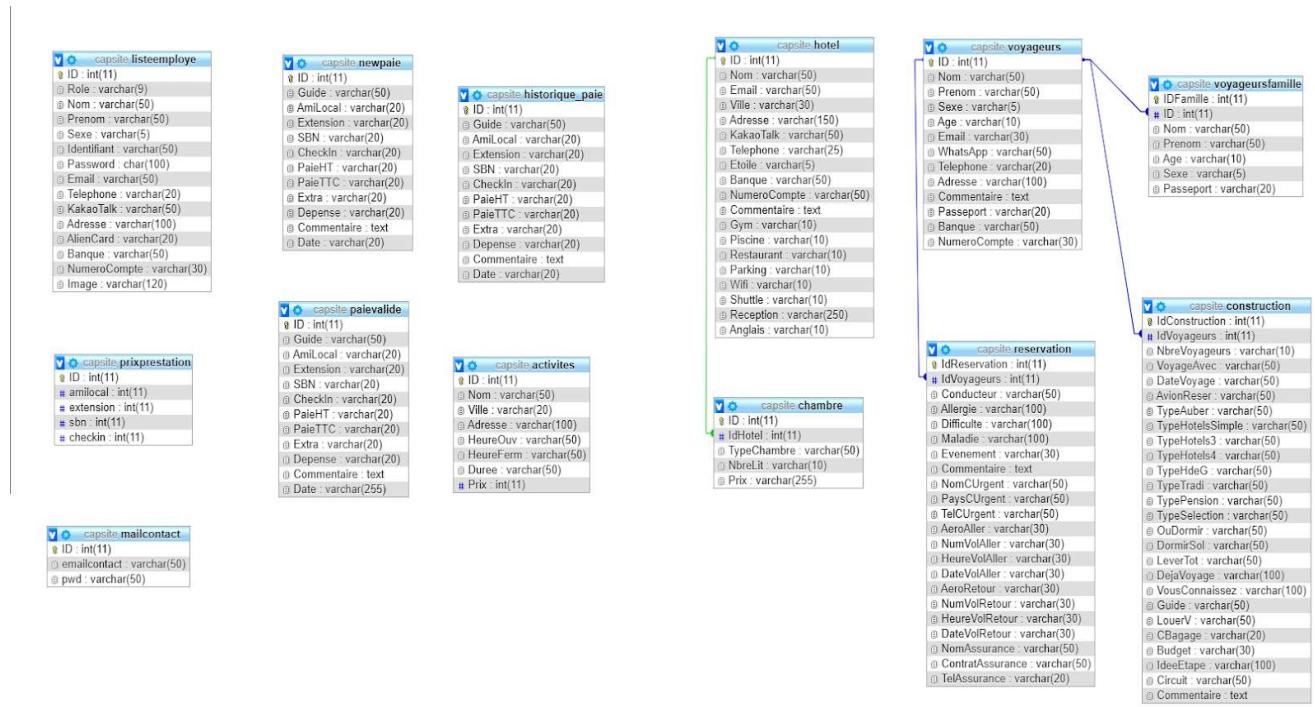
Lors de l'entretien avec la personne de l'agence de voyage, j'avais pris des notes au sujet des différentes informations que l'on souhaitait voir apparaître sur le site. Ceci m'a donc permis de créer les tables de ma base ainsi que les champs et leurs types.

On remarque que certaines tables sont MyISAM et d'autres InnoDB pourquoi avoir fait le choix de ne pas utiliser les mêmes types de table ?

Tout simplement car MyISAM serait plus rapide en Lecture et en Ecriture contrairement à InnoDB. Ce pourquoi certaines tables sont en MyISAM, mais j'ai tout de même utilisé InnoDB car ce type autorise les relations entre les tables, donc l'utilisation de clé étrangère.

Définition Wikipedia de Clé Etrangère : « Une clé étrangère, dans une base de données relationnelle, est une contrainte qui garantit l'intégrité référentielle entre deux tables. Une clé étrangère identifie une colonne ou un ensemble de colonnes d'une table comme référençant une colonne ou un ensemble de colonnes d'une autre table. »

▪ Modèle conceptuel



Concepteur de vue PhpMyAdmin

Les 7 tables de gauche sont indépendantes. Mais on remarque sur la droite 6 autres tables qui sont reliées par des traits verts et bleus.

Prenons la table avec le trait vert, la table "hotel" cette table possède 19 champs. Le premier étant ID. Chaque nouvelle entrée dans la table "hotel" se verra attribuer un identifiant unique.

Prenons à présent la table "chambre" juste en dessous (reliée par le trait vert), celle-ci possède 5 champs, avec également un ID unique, puis le deuxième champ "idHotel" qui lui est la relation avec la table "hotel".

Je n'ai pas fait la méthode Merise en écrivant les contraintes 1,1/0/1 etc..., mais nous pouvons comprendre dans ce cas qu'un hôtel peut ne pas avoir de chambre, mais qu'une chambre a forcément un hôtel et seulement un hôtel.

Il en est de même pour les tables "voyageurs", "réservation", "construction" et "voyageursfamille" fonctionnent de la même manière :

Un voyageur peut voyager seul, mais "voyageursfamille" ne peut être créé sans voyageurs.

Et de même pour Réservation/Construction qui ne peuvent exister sans un voyageur.

- L'ajout de donnée

Comment peut on ajouter des données dans nos tables ?

- Première chose, il faut faire la liaison entre la table et site/l'application
- Créer un formulaire pour pouvoir récupérer les informations de l'utilisateur
- Insérer les informations dans les tables de la base de données.

Pour faire la liaison entre les deux, j'ai donc créé un fichier PHP et une class DATABASE avec les informations de connexions et deux fonctions que je pourrais utiliser partout dans mon site quand j'en aurais besoin et sans avoir à réécrire plusieurs fois la même chose (principe d'une fonction)

```

1 <?php
2
3 class Database {
4     private static $dbHost = "localhost";
5     private static $dbName = "capsite";
6     private static $dbUser = "root";
7     private static $dbPassword = "";
8
9     private static $connection = Null;
10
11    public static function connect(){
12        try{
13            $options = array(
14                PDO::MYSQL_ATTR_INIT_COMMAND => 'SET NAMES utf8'
15            );
16            self::$connection = new PDO("mysql:host=" . self::$dbHost . ";dbname=" . self::$dbName, self::$dbUser, self::$dbPassword, $options );
17        }catch(PDOException $e){
18            die($e->getMessage());
19        };
20        return self::$connection;
21    }
22
23    public static function disconnect(){
24        self::$connection = null;
25    }
26 };
27
28 Database::connect();
29
30 ?>

```

L 4/5/6/7 sont les informations concernant la BDD, étant l'adresse du serveur, le nom de la BDD, l'accès à celle-ci avec son Identifiant et son Mot de Passe, qui par défaut est root et MDP vide.

L'avantage de créer un fichier séparé avec des variables comme ceci, est que le jour où les infos de la BDD changent, il ne suffira que de modifier un seul fichier et non pas chaque page où on fait appel au serveur.

Ensuite deux fonctions de connexion et déconnexion.

Durant mon stage je me suis aidé énormément de mes cours, mais également du site [PHP.net](#) qui permet de trouver des classes, des fonctions avec des explications, mais aussi des exemples détaillés. De plus, on peut savoir que certaines fonctions sont utilisables sous telles ou telles versions de PHP.

A présent nous pouvons nous connecter il faut donc avoir un formulaire pour recueillir les informations de l'utilisateur, prenons l'exemple de l'ajout d'un employé. Voici donc la page du site.

Ajouter Un Employé

Role : <input type="text" value="Employé"/>	KakaoTalk : <input type="text" value="KakaoTalk"/>
Nom : <input type="text" value="Nom *"/>	Adresse : <input type="text" value="Adresse"/>
Prenom : <input type="text" value="Prenom *"/>	AlienCard : <input type="text" value="AlienCard"/>
Sexe : <input type="text" value="Femme"/>	Banque : <input type="text" value="Banque"/>
Email : <input type="text" value="Email *"/>	Numero De Compte : <input type="text" value="Numero De Compte"/>
Telephone : <input type="text" value="Telephone"/>	Sélectionner une image: <input type="button" value="Choisir un fichier"/> Aucun fichier choisi
<input type="button" value="Ajouter"/> <input type="button" value="Retour"/>	

Vue Ajout Employé

12 champs sont à remplir/sélectionner :

- Role, Nom,Prenom etc...

Et voici une partie du formulaire de celui-ci

```

186      <form action="employeesinsert.php" method="POST" enctype="multipart/form-data" class="text-center w-auto">
187          <div class="form-group">
188              <label class="font-weight-bold" for="role">Role : </label>
189              <select id="search" name="role" id="role" >
190                  <option value="Employe">Employé</option>;
191                  <option value="Finance">Finance</option>;
192                  <option value="Guide">Guide</option>;
193                  <option value="Stagiaire">Stagiaire</option>;
194                  <option value="Admin">Admin</option>;
195              </select><br>
196              <span class="erreur"><?php echo $roleError; ?></span>
197          </div>
198          <div class="form-group">
199              <label class="font-weight-bold" for="name">Nom : </label>
200              <input id="search" type="text" id="name" name="name" placeholder="Nom *" value="<?php echo $name ?>"><br>
201              <span class="erreur"><?php echo $nameError; ?></span>
202          </div>
203          <div class="form-group">
204              <label class="font-weight-bold" for="surname">Prenom : </label>
205              <input id="search" type="text" id="surname" name="surname" placeholder="Prenom *" value="<?php echo $surname ?>"><br>
206              <span class="erreur"><?php echo $surnameError; ?></span>
207          </div>

```

Une partie d'un formulaire

On voit L186, la balise form qui indique que nous sommes dans un formulaire, mais aussi ‘action’ qui permet de voir où se traitera le formulaire. Method, qui nous indique comment les informations seront transmises :

- POST : permettant de faire passer les informations de façons plus sécurisées, et invisibles à l’utilisateur.
- GET : les informations passent par l'url donc visible de tous.

Toujours L186, il y a enctype, car nous allons envoyer des informations texte, mais aussi des fichiers (photo), puis enfin des classes venant de BootStrap.

J'avais décrit plus haut déjà les labels, et inputs du formulaire, ici ce qui est important c'est l'attribut value des inputs.

```
value=""
```

Le ‘value’ est en fait la valeur que l’utilisateur va rentrer/sélectionner.

```
<label class="font-weight-bold" for="surname">Prenom : </label>
<input id="search" type="text" id="surname" name="surname" placeholder="Prenom *"
value="php echo $surname ?&gt;"&gt;</pre
```

Le formulaire est en HTML mais la valeur sera récupérée dans une variable PHP, et ce pour chaque inputs du formulaire.

Une fois le formulaire rempli et envoyé, la page va se rafraîchir, et celle-ci va lire notre fichier de haut en bas encore une fois. Mais alors que la première fois avant le remplissage du formulaire elle n’était pas passée dans une vérification du formulaire, car celui-ci était vide, mais à présent que nous en avons soumis un, il faut le traiter.

```
if(!empty($_POST)){
```

Quand la page se recharge après l’envoie du formulaire, celui-ci passe donc dans cette condition :

Traduction du code ci-dessus : **SI (différent de vide (formulaire POST)) ALORS**

Donc comme nous avons envoyé un formulaire, celui-ci n’est pas vide donc il passe dans cette condition.

Nous allons à présent vérifier tous les champs du formulaire pour être certains que ceux-ci n’ont pas de code ou de choses malveillantes. Pour éviter toutes sortes d’injections SQL par exemple.

Nos professeurs nous ont toujours dit de ne pas faire confiance aux saisies des utilisateurs

```
$role = checkInput($_POST['role']) ;
$name = checkInput($_POST['name']);
$surname = checkInput($_POST['surname']);
$sexe = checkInput($_POST['sexe']);
```

```

// l'identifiant sera le nom et surnom de la personne
// the Id user would be name and surname
$identifiant = $_POST['name'] . " " . $_POST['surname'];
$email = checkInput($_POST['email']);
$telephone = checkInput($_POST['telephone']);
$kakaotalk = checkInput($_POST['kakaotalk']);
$adresse = checkInput($_POST['adresse']);
$aliencard = checkInput($_POST['aliencard']);
$banque = checkInput($_POST['banque']);
$numerocompte = checkInput($_POST['numerocompte']);
$image = checkInput($_FILES["image"]["name"]);
$imagePath = '../images/employes/' . basename($image);
$imageExtension = pathinfo($imagePath, PATHINFO_EXTENSION);

```

Après vérification des champs saisis, tous ceux-ci sont placés dans une variable pour que ce soit plus clair.

Ensuite, nous vérifions si certains champs sont vides ou non. Car le formulaire peut être considéré comme non vide si un seul des champs est rempli sur l'ensemble de celui-ci.

Et nous voudrions obligatoirement que certains champs de notre futur employé ne soient pas vides comme :

- Rôle – Nom – Prénom - Email

```

40     $isSuccess = true;
41     $isUploadSuccess = false;
42
43
44     if(empty($role)){
45         $roleError = "Ce champs ne peut etre vide";
46         $isSuccess = false;
47     };
48     if(empty($name)){
49         $nameError = "Ce champs ne peut etre vide";
50         $isSuccess = false;
51     };
52     if(empty($surname)){
53         $surnameError = "Ce champs ne peut etre vide";
54         $isSuccess = false;
55     };
56     if(empty($email)){
57         $emailError = "Ce champs ne peut etre vide";
58         $isSuccess = false;
59     };

```

Si un de ces champs venait à être vide, alors une erreur s'afficherait à l'écran de l'utilisateur.

```

60     if(empty($image)){ // Si l'utilisateur ne renseigne pas de photo, alors en fonction de son Sexe on lui attribue une Image par Default
61         if($sexe == "Femme"){$image = "female.png";}
62         else{$image = "male.png";}
63         $isUploadSuccess = true;
64     }
65     else // Sinon on verifie les extensions, la taille ainsi que le nom du fichier
66     {
67         $isUploadSuccess = true; // Permet de verifier si on a bien envoye le fichier sur le serveur
68         $newImagePath = ""; // Permet de savoir si on a un fichier avec le même nom et donc pour redefinir le chemin
69
70         if($imageExtension != "jpg" && $imageExtension != "png" && $imageExtension != "jpeg" && $imageExtension != "gif" ) {
71             $imageError = "Les fichiers autorises sont: .jpg, .jpeg, .png, .gif";
72             $isUploadSuccess = false;
73         };
74         if($_FILES["image"]["size"] > 500000) [
75             $imageError = "Le fichier ne doit pas depasser les 500KB";
76             $isUploadSuccess = false;
77         ];
78
79         if($isUploadSuccess && file_exists($imagePath)) {
80             $image = date("dmYHis") . basename($image); // On rajoute la date et l'heure au nom de l'image pour etre sur que celle-ci n'a
81             $newImagePath = '../images/employes/' . $image; // On cree le nouveau chemin de sauvegarde
82         };
83
84         if($newImagePath != "") { // Si different de vide alors on utilise le nouveau chemin
85             if(!move_uploaded_file($_FILES["image"]["tmp_name"], $newImagePath)){
86                 $imageError = "Il y a eu une erreur lors de l'upload";
87                 $isUploadSuccess = false;
88             };
89         }
90         else{
91             if(!move_uploaded_file($_FILES["image"]["tmp_name"], $imagePath)){ // Sinon on utilise l'ancien.
92                 $imageError = "Il y a eu une erreur lors de l'upload";
93                 $isUploadSuccess = false;
94             };
95         };
96     };
97 }

```

Dans ce cas, nous contrôlons si le champ d' image est vide. SI celui-ci l'est alors en fonction du genre de la personne nous attribuons une image par défaut, SINON nous contrôlons :

- L'extension de l'image
- La taille de celle-ci
- L'existence d'une image avec le même nom

Et pour finir si aucunes erreurs nous insérons les données dans la BDD.

```

// Si tout ce passe bien alors on INSERT
    if($isSuccess && $isUploadSuccess){
        $db = Database::connect();
        $hash = password_hash("12345",PASSWORD_DEFAULT); // creation d un
mot de passe pour l utilisateur avec par default avec la securite la plus perf
ormante du moment

        try{
            $statement = $db->prepare
("INSERT INTO listeemploye (Role,Nom,Prenom,Sexe,Identifiant,Password,Email,Te
lephone,KakaoTalk,Adresse,AlienCard,Banque,NumeroCompte,Image)
            VALUES(?,?,?,?,?,?,?,?,?,?,?,?,?,?)");
            $statement->execute
(array($role,$name,$surname,$sexe,$identifiant,$hash,$email,$telephone,$kakaot
alk,$adresse,$aliencard,$banque,$numerocompte,$image));

```

Nous récupérons la fonction connect() de la classe DATABASE.

Nous passons dans le try/catch, qui permet d'essayer de faire quelque chose, mais si ceci ne fonctionne pas alors au lieu de planter l'application nous afficherons par exemple un message à l'utilisateur lui disant les raisons pour lesquelles ceci ne fonctionne pas.

Nous passons donc à la ligne avec le \$statement. On peut voir le ‘prepare’ ceci veut dire que nous allons insérer dans une base de données via des variables non connues. Si nous connaissions toutes les valeurs, nous aurions pu utiliser le mot ‘query’

Puis donc l’Insertion :

Nous insérons dans la table ‘listeemploye’ (tous les champs de la table) Et les valeurs, ici des ? pour chaque champ.

Et nous exécutons cette variable avec les valeurs dans un tableau.

- La cryptographie

Parlons à présent de la cryptographie, il n'est pas conseillé de sauvegarder les mots de passe des sessions en clair dans la BDD. Ce pourquoi nous devons utiliser la cryptographie.

En Php, il est très simple de prendre un Mot de passe puis de le hacher. Le code ci- dessous nous le montre.

```
$hash = password_hash("12345",PASSWORD_DEFAULT);
```

Pour remettre dans le contexte, je souhaiterais que lorsque je crée un nouvel employé, je lui attribue un MDP par défaut qui serait ‘12345’. Pour cela nous allons créer une variable \$hash, dans cette variable nous utiliserons la fonction PHP password_hash() qui demande au minimum 2 paramètres. Le premier, le mot de passe à hacher, et le deuxième la technologie de hachage, en mettant Password_default, nous choisissons la technologie la plus avancée du moment.

Ce hash sera alors envoyé dans la BDD avec le INSERT TO

Voici ce que ressemble le MDP ‘12345’ dans la BDD (chaque MDP ‘12345’ a un hash différent)

‘\$2y\$10\$nnpvcD0ayAqQk4Luu3fK4uCnA2yGUbHzrx2GaxBYDb/SfMR9SgyNa’

Pour se connecter, il faudra utiliser une autre fonction PHP qui est : password_verify() également avec 2 paramètres, le premier étant le MDP renseigné par l'utilisateur dans le formulaire de connexion et le deuxième le MDP Haché venant de la BDD.

- Les E-mails

Une fonctionnalité qui peut être intéressante et souvent très utile dans les sites c'est le fait de pouvoir envoyer des e-mails, par exemple pour remercier une personne d'avoir créé un compte, d'avoir acheté quelque chose sur notre site et j'en passe.

J'ai opté pour la librairie PhpMailer() (<https://github.com/PHPMailer/PHPMailer>) qui est très simple à mettre en place.

```
$mail = new PHPMailer(true);
try {

    $sqlContact = $db->query("SELECT * FROM mailcontact");
    $sqlContact ->execute(array());
    $item= $sqlContact->fetch();

    //Server settings
    $mail->SMTPDebug = 2;                                // Enable verbose debug output
    $mail->isSMTP(true);                               // Send using SMTP
    $mail-> CharSet = "UTF-8";
    $mail->Host      = 'smtp.gmail.com';                // Set the SMTP server to send through
    $mail->SMTPAuth = true;                             // Enable SMTP authentication
    $mail->Username  = $item['emailcontact'];           // SMTP username
    $mail->Password  = $item['pwd'];                   // SMTP password
    $mail->SMTPSecure = 'tls';                         // Enable TLS encryption; 'PHPMailer::ENCRYPTION_SMTPS' encouraged
    $mail->Port      = 587;                            // TCP port to connect to, use 465 for 'PHPMailer::ENCRYPTION_SMTPS' above

    //Recipients
    $mail->setFrom($item['emailcontact'], 'CapCoree');
    $mail->addAddress($email);

    // Content
    $mail->isHTML(true);                                // Set email format to HTML
    $mail->Subject = 'Identifiant Compte CapCoree';
    $mail->Body   = "Bonjour, <br>Nous sommes heureux de vous compter parmi l'équipe CapCorée. Pour nous rejoindre  

                    Identifiant : " . $identifiant . "<br>Mot De Passe : 12345 <br> <br>
                    Pour des raisons évidentes de sécurité il vous est recommandé de modifier celui-ci au plus vite.  

                    A bientôt sur <a href=https://www.capcoree.fr>CapCorée</a>";
```

La librairie phpmailer() est orientée objet, donc on voit à la première ligne que nous créons un objet appelé mail.

Puis par la suite tout le code correspond aux réglages du mail, comme le charset utf8 qui autorise les accents dans le mail, le serveur SMTP en l'occurrence ici j'ai choisi les serveurs google. Puis bien sûr l'adresse de l'envoyeur, son MDP (on remarque que ce sont des variables car nous les récupérons de la BDD)

Il faudra également l'email du futur employé, dans notre cas nous l'avons récupéré dans le formulaire. Et pour finir le sujet et le corps du mail.

J'ai écrit le MDP en dur dans le mail, mais j'aurai très bien pu créer une variable comme je l'ai fait pour \$identifiant, et le mail serait encore plus dynamique.

- Les rôles

Il était important de créer des rôles en fonction du poste exercé au sein de l'agence.

Un Admin par exemple était censé pouvoir tout faire/voir sur le site, mais un guide lui ne devait avoir accès seulement à quelques pages, en aucun cas un guide devrait pouvoir modifier les informations de voyageur ou d'hôtel.

Pour cela un employé lors de sa création se voyait attribuer un rôle.

Voici les 5 rôles possibles de l'agence :

Admin – Employé – Finance – Stagiaire - Guide

Comme chaque employé (comprends tous les rôles confondu), avait un identifiant et un mot de passe, il pouvait se connecter à un compte, et donc nous pouvions créer ce qu'on appelle des sessions.

```
if(password_verify($_POST["password"], $item["Password"])) {  
    $_SESSION["membre"] = $identifiant;  
    $_SESSION["role"] = $item["Role"];  
    $_SESSION["nom"] = $item["Nom"];  
    $_SESSION["prenom"] = $item["Prenom"];  
    header("Location: admin/compte.php");  
} else{  
    $erreur = "Les identifiants sont erronés";  
};
```

Sur cette image on voit que si le MDP correspond à celui de la BDD alors on ouvre 4 sessions, la première comprenant l'identifiant, ensuite le rôle, le nom, et le prénom et que nous redirigeons la personne sur la page compte.php.

Une fois ces sessions créées l'utilisateur pourra avoir accès ou non à certaines pages. Et pour cela en début de chaque page il faudra définir qui peut ou non y accéder.

Prenons exemple sur la page employes.php, voici ce qui est placé en haut de la page :

```
session_start();  
if(!isset($_SESSION["membre"]) || $_SESSION["role"] == "Guide")  
    header("Location: ../login.php");
```

On active les sessions, et SI la session membre n'est pas paramètre, OU Si la session rôle est un utilisateur guide alors on les redirige vers la page login (page de connexion).

- Les vérifications

A FAIRE, mais le but est de vérifier les entrées des inputs compta !!!!!!!!

VII. Conclusion

Pour conclure sur ce stage, je dirai que ceci a été une expérience très enrichissante professionnellement.

Le fait d'approfondir des technologies vues principalement durant mes cours, le fait de pouvoir l'appliquer sur un projet qui fut des plus intéressant. Découvrir également des concepts plus avancés.

La gestion du temps qui était un paramètre à prendre en compte, ayant seulement 5 semaines durant ce deuxième stage, il fallait essayer de terminer le projet dans le temps imparti, ce qui fut le cas.

L'échange avec mon maître de stage et toute l'équipe était vraiment au rendez-vous, ce qui fut très appréciable. Toujours prêt à répondre à mes questions, et prendre cinq minutes à m'expliquer pourquoi cette technologie plutôt que celle-ci. En m'apportant toutes leurs expériences.

Je tiens à les remercier encore une fois pour leur temps, leurs explications et toutes les choses qu'ils m'ont apportées durant ce stage.

VIII. Cahier des charges

Cahier des charges Site Web CapCorée

Présentation de l'entreprise :

CapCorée petite agence de voyage francophone situé en Corée du Sud à Séoul.

Date du début et fin de projet :

Au date du/2021 au/2021

Présentation brève du projet :

Réalisation d'un site internet permettant la gestion des informations des employés ainsi que celles des voyageurs.

Accès au Site Web pour qui ?

Seuls les employés de l'agence auront accès à ce site.

Ce site doit être multilingue ?

Non seul le français sera utilisé dans ce projet.

Ce site doit-il être responsive pour pouvoir être utilisé sur différents supports ?

Ce site aura pour vocation de n'être utilisé seulement à l'agence sur des grands écrans, donc le responsive n'est pas obligatoire.

Un design doit être respecté ? si oui, qui le fait ?

Non, libre à mon imagination.

Quelles sont les fonctionnalités que le site doit avoir ? (Veuillez créer pour chaque fonctionnalité un détail de chacune d'entres elles)

1. Pouvoir créer compte par employé avec des autorisations d'accès à certaines sections du site.
2. Pouvoir faire la comptabilité des prestations des guides.
3. Posséder un historique des paies des guides.
4. Posséder un espace dédié aux nouvelles paies.
5. Pouvoir ajouter des informations à propos des hôtels/chambres que l'agence utilise.
6. Pouvoir ajouter des informations sur les activités que l'agence propose.
7. Pouvoir ajouter des informations sur les voyageurs.
8. Ajouter une section paramètre permettant de faire du contenu dynamique.

1. Pouvoir créer compte par employé avec des autorisations d'accès à certaines sections du site.

Il y a 5 rôles différents dans l'entreprise, et en fonction du droit, ils auront accès ou non à certaines sections du site :

- Admin (tous les droits)
- Finance (verra seulement la partie alerte en ce qui concerne les paies)
- Employé (accès à tout, mais restriction pour supprimer/modifier)
- Stagiaire (limité en termes d'accès et restriction supprimer/modifier)
- Guide (accès très limité, seulement pour indiquer ses prestations et voir historique)

Seul un **Admin** peut créer/modifier/supprimer un compte.

Tous les rôles ont accès à une page « Mon Compte » et peuvent modifier leurs propres informations

Lors de la création d'un compte, celui-ci se voit attribuer un rôle, un identifiant ainsi qu'un mot de passe provisoire. Le nouveau compte se verra envoyer un e-mail avec ses identifiants pour pouvoir se connecter et modifier son profil.

2. Pouvoir faire la comptabilité des prestations des guides.

Les guides auront un accès très limité, pouvant simplement avoir accès à leur compte, mais également la page de comptabilité ainsi qu'à leur historique de travail.

Le reste des rôles pourront quant à eux faire la comptabilité pour un guide qui n'aurait pas accès à internet par exemple.

3. Posséder un historique des paies des guides.

Ceci rejoint le point précédent, un accès permettant de regarder son historique de paie. Bien sur un guide se connectant n'aura accès qu'à son historique quant aux autres, ils auront accès à tous les guides.

4. Posséder un espace dédié aux nouvelles paies.

Lorsqu'une comptabilité aura été soumise, dans la section « Alertes » une ligne s'affichera et demandera à un Admin de vérifier les informations puis si tout est valide, elles iront à l'équipe de finance qui une fois le salaire versé, le validera et cette paie ira donc dans la section « Historique ».

5. Pouvoir ajouter des informations à propos des hôtels/chambres que l'agence utilise.

Diverses informations aux sujets des hôtels seront possibles d'ajouter/modifier/supprimer avec en plus les chambres et les informations que les hôtels proposent comme piscine, salle de sport etc...

6. Pouvoir ajouter des informations sur les activités que l'agence propose.

Possibilité d'ajouter/modifier/supprimer diverses activités que l'agence de voyage propose à ses voyageurs, ex : visite d'un palais etc...

7. Pouvoir ajouter des informations sur les voyageurs.

Diverses informations aux sujets des voyageurs seront possibles d'ajouter/modifier/supprimer avec en plus les différents membres qui participeront aux voyages.

8. Ajouter une section paramètre permettant de faire du contenu dynamique.

Cette section sera réservée à l'**Admin**, il pourra ajouter des informations du type l'email & mot de passe de connexion pour envoyer des mails, mais également le prix des prestations pour la comptabilité des guides.

Technologies utilisées et outils pour le projet :

Maquette : Balsamiq

Ide : Visual Studio Code

Gestion Base de données : MySql

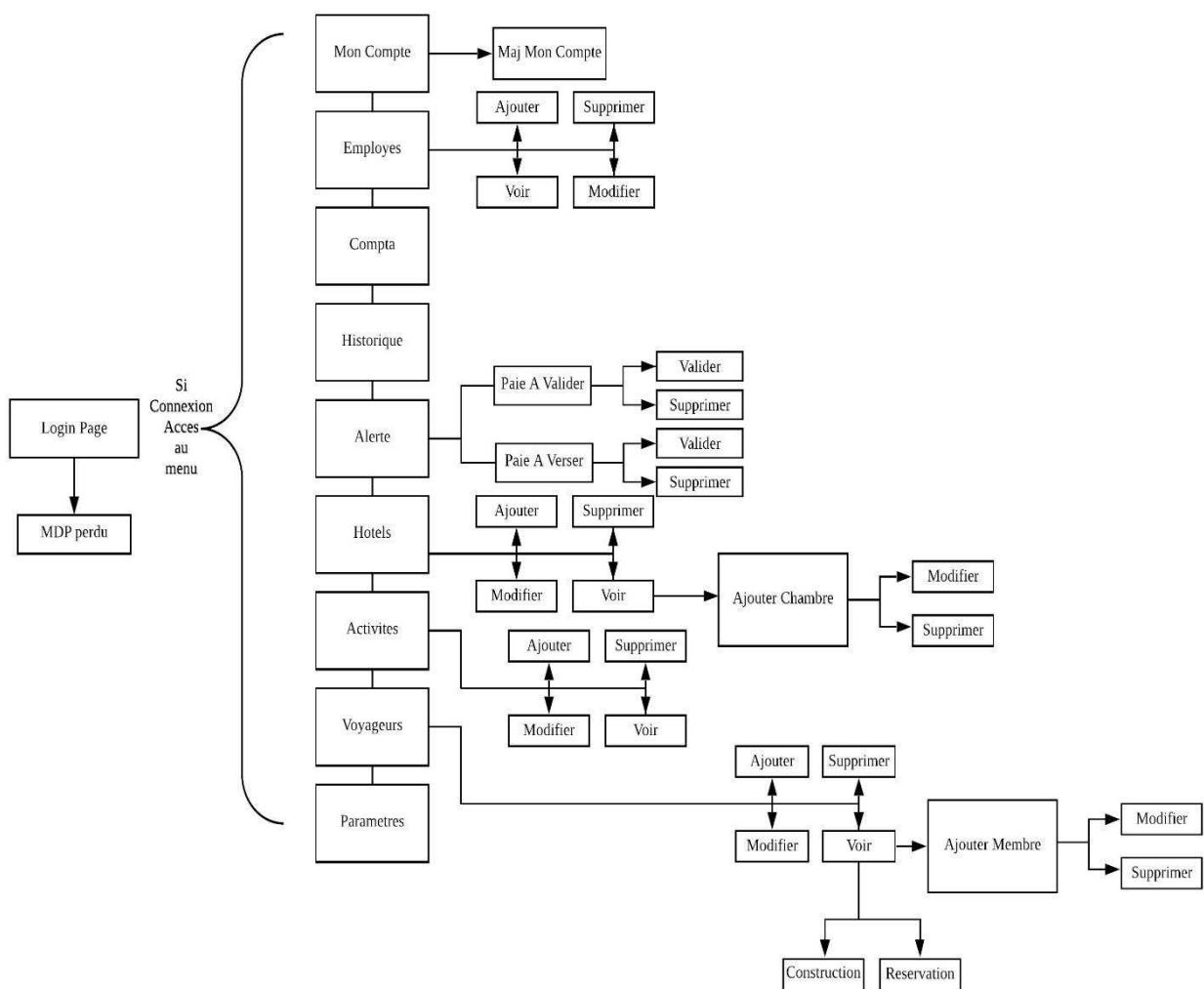
Partie Front-End :

- Html 5
- Css 3
- Javascript
- Framework Bootstrap 4

Partie Back-End :

- Php 7.2
- Sql

Organigramme de toutes les vues du site :



Maquettes de toutes les vues du site :

Page Connexion

The screenshot shows a web browser window titled "Login". The URL bar contains "http://". The header includes the logo "NI VOYAGE DIFFERENT" and a "Connexion" button. The main content area has a heading "Veuillez saisir vos identifiants". Below it, a red error message "Les identifiants sont erronés" is displayed next to a left-pointing arrow. There are two input fields: "Identifiant*" and "Mot de passe*". To the right of these fields is a yellow callout box containing the text "En cas d'erreur un label s'affiche pour vous indiquer une erreur". At the bottom right is a "Me Connecter" button. A right-pointing arrow is positioned between the "Mot de passe" field and the "Me Connecter" button. A yellow callout box on the left contains the text "Si MDP oublié alors un modal JS s'affiche pour rentrer son mail et son identifiant".

Page Mon Compte et Page Voir Employé (seront identiques)

The screenshot shows a web browser window titled "Mon Compte Update". The URL bar contains "http://". The header includes the logo "NI VOYAGE DIFFERENT" and a search icon. The navigation menu lists: Employes, Compta, Historique, Alertes, Hotels, Activites, Voyageurs, Parametres, Mon Compte, and Deconnexion. The main content area features a large user profile icon on the left. To its right are two rows of text fields labeled "Some text". A yellow callout box at the bottom left contains the text "La page mon compte et mon compte Update auront la même configuration que Voir/Update Employes" and "Les pages comme Voir Hotels, voir Activite seront sur le même modèle sans la photo".

Page Mon compte Mise à Jour

Mon Compte Update

NIVAGEOIS
Capcorse.fr

Employes Compta Historique Alertes Hotels Activites Voyageurs Parametres Mon Compte Deconnexion



Some text
Some text
Some text

Some text
Some text
Some text

La page mon compte et mon compte Update auront la même configuration que Voir/Update Employees
Les pages comme Voir Hotels, voir Activite seront sur le même modèle sans la photo.

Page Paramètres

Parametres

NIVAGEOIS
Capcorse.fr

Employes Compta Historique Alertes Hotels Activites Voyageurs Parametres Mon Compte Deconnexion

Parametres

Some text
Some text
Some text

Some text
Some text

Pas sur encore des infos que je mettrai ici, mais en cliquant sur un des 2 boutons alors les inputs apparaîtront et le bouton "Modifier" deviendra "Envoyer"
Tout ceci en JS

Page Comptabilité

Employees

http://

N V oyage
Corporation

Employees Compta Historique Alertes Hotels Activites Voyageurs Parametres Mon Compte Deconnexion

Some text Nom Guide ↗
 Some text ComboBox ↗
 Some text ComboBox ↗
 Some text
 Some text
 Some text
 Some text / / ↗
 Some text ↗

Si connecte avec un compte guide, alors celui-ci ne voit que son nom, sinon la liste de tous les guides

Apres avoir repondu au formulaire, il faudra bien choisir une date pour l'historique et savoir de quel mois le travail correspond

envoyer retour

Page Alerta

Alertes

http://

N V oyage
Corporation

Employees Compta Historique Alertes Hotels Activites Voyageurs Parametres Mon Compte Deconnexion

Paie A Valider

Une fois que la comptabilité a été validée, elle apparaît dans cette page ALERTA qui permet à l'admin de vérifier si les jours travaillés du guide sont correct, si oui il valide, la paie ira donc dans le deuxième tableau et donc la personne de la finance elle se chargera juste de verser son salaire. Une fois validé par la finance alors la paie ira dans historique.

Nom	extension	Amilocal	paie	date	action
Alexis	2	3	300.000	11/2020	VALIDER/SUPPRIMER
Pierre	6	8	634.000	04/2021	VALIDER/SUPPRIMER
Jean	4	6	367.000	03/2021	VALIDER/SUPPRIMER

Paie A Verser

Nom	extension	Amilocal	paie	date	action
Alexis	2	3	300.000	11/2020	VALIDER/SUPPRIMER
Pierre	6	8	634.000	04/2021	VALIDER/SUPPRIMER
Jean	4	6	367.000	03/2021	VALIDER/SUPPRIMER

Page Historique

The screenshot shows a web browser window with the title "Historique". The navigation bar includes links for Employes, Compta, Historique, Alertes, Hotels, Activites, Voyageurs, Parametres, Mon Compte, and Deconnexion. The main content area contains a table with columns: Nom, extension, Amilocal, poie, and date. The data rows are:

Nom	extension	Amilocal	poie	date
Alexis	2	3	300.000	11/2020
Pierre	6	8	634.000	04/2021
Jean	4	6	367.000	03/2021
Aline	1	1	122.000	03/2021
Fanny	q	3	784.000	02/2021

A yellow callout box at the bottom left states: "Un guide ne verra que son historique mais les autres employes verront l historique de tous les guides".

Page Employés

The screenshot shows a web browser window with the title "Employes". The navigation bar includes links for Employes, Compta, Historique, Alertes, Hotels, Activites, Voyageurs, Parametres, Mon Compte, and Deconnexion. The main content area displays a list titled "Liste des Employes" with a "Ajouter" button. A yellow callout box points to the "Ajouter" button with the text: "Possibilité d'ajouter un profil mais également de Voir / Modifier / Supprimer sont des boutons Ceci sera utilisé dans la plupart des pages". Below the list is a table with columns: Role, Nom, Prenom, Sexe, and Action.

Role	Nom	Prenom	Sexe	Action
Admin	Super	admin	Homme	VOIR/MODIFIER/SUPPRIMER
Employe	Jeanne	Beal	Femme	VOIR/MODIFIER/SUPPRIMER
Employe	Bernard	Roger	Homme	VOIR/MODIFIER/SUPPRIMER
Employe	Alexis	jean	Homme	VOIR/MODIFIER/SUPPRIMER
Employe	Sylvie	Dumont	Femme	VOIR/MODIFIER/SUPPRIMER
Employe	Jeremie	Jean	Homme	VOIR/MODIFIER/SUPPRIMER

Page Activités

Activités

Employees Compta Historique Alertes Hotels Activites Voyageurs Parametres Mon Compte Deconnexion

Liste des Activites

Ajouter

Nom	Ville	email	Action
Cuisine	Seoul	Cuisine@hotmail.fr	VOIR/MODIFIER/SUPPRIMER
Lotte	Busan	Lotte@hotmail.fr	VOIR/MODIFIER/SUPPRIMER
Palais	Seoul	Palais@hotmail.fr	VOIR/MODIFIER/SUPPRIMER
Garden	Busan	Garden@hotmail.fr	VOIR/MODIFIER/SUPPRIMER
Marche trad	Seoul	Marche-trad@hotmail.fr	VOIR/MODIFIER/SUPPRIMER
Spectacle trad	Jeonju	Spectacle-trad@hotmail.fr	VOIR/MODIFIER/SUPPRIMER

Page Hotels

hotels

Employees Compta Historique Alertes Hotels Activites Voyageurs Parametres Mon Compte Deconnexion

Liste des Hôtels

Ajouter

Nom	Ville	email	Action
Hilton	Seoul	hilton@hotmail.fr	VOIR/MODIFIER/SUPPRIMER
Mariott	Busan	mariott@hotmail.fr	VOIR/MODIFIER/SUPPRIMER
BestHotel	Seoul	Best@hotmail.fr	VOIR/MODIFIER/SUPPRIMER
Accord	Busan	Accordhotel@hotmail.fr	VOIR/MODIFIER/SUPPRIMER
Sheraton	Seoul	sheraton@hotmail.fr	VOIR/MODIFIER/SUPPRIMER
Euro Hotel	Jeonju	eurohotel@hotmail.fr	VOIR/MODIFIER/SUPPRIMER

Page Voir Hotels

Page Voyageurs

Voyageurs

Employees Compta Historique Alertes Hotels Activites Voyageurs Parametres Mon Compte Deconnexion

Liste des Voyageurs

Ajouter

Nom	Prenom	Email	Action
Jeanne	Beal	jeanne@gmail.com	VOIR/MODIFIER/SUPPRIMER
Bernard	Roger	@gmail.com	VOIR/MODIFIER/SUPPRIMER
Alexis	jean	alexis-jean@gmail.com	VOIR/MODIFIER/SUPPRIMER
Sylvie	Dumont	dumont54@gmail.com	VOIR/MODIFIER/SUPPRIMER
Jeremie	Jean	jeje@gmail.com	VOIR/MODIFIER/SUPPRIMER

Page Voir Voyageurs

The screenshot shows a web application interface titled "Voir Voyageurs". At the top, there is a navigation bar with links: Employes, Compta, Historique, Alertes, Hotels, Activites, Voyageurs, Parametres, Mon Compte, and Deconnexion. Below the navigation bar, there is a section titled "Information" containing a grid of travel details. A yellow callout box points to two buttons: "Voir Construction" and "Voir Reservation". To the right of the grid, there is a section titled "Liste des Membres" with a table showing member details: Nom, Prenom, Age, Passeport, and Action. An "Ajouter" button is located above the table.

Nom	Prenom	Age	Passeport	Action
Dubois	julie	19	fr783234	MODIFIER/SUPPRIMER
Dubois	mathieu	19	fr7834332	MODIFIER/SUPPRIMER
Dubois	thierry	42	fr783t434	MODIFIER/SUPPRIMER

Page Construction / Réservation (identiques)

The screenshot shows a web application interface titled "Construction / Reservation". At the top, there is a navigation bar with links: Employes, Compta, Historique, Alertes, Hotels, Activites, Voyageurs, Parametres, Mon Compte, and Deconnexion. Below the navigation bar, there is a section containing a grid of travel details. At the bottom of the page, there is a note in a yellow box explaining the purpose of the page and how it relates to Google Forms.

Note:

La page construction et Reservation a pour but de saisir les informations que les voyageurs auront préalablement remplis dans un document google form. En cliquant sur modifier, les réponses deviendront des input box, et le bouton modifier deviendra VALIDE! Tout ceci sera en JS

Page Supprimer

Mon Compte Update

Employees Compta Historique Alertes Hotels Activites Voyageurs Parametres Mon Compte Deconnexion

Est vous sur de vouloir supprimer ?

Valider Retour

Les pages pour supprimer auront toutes la même configuration.

Vue de la base de données depuis MySQL

