

Maxime  
Blanc

---

# RAPPORT DE STAGE

---

18/04/2022 - 10/06/2022



|  |              |
|--|--------------|
| <b>Page de garde.....</b>                        | <b>1</b>     |
| <b>Sommaire .....</b>                            | <b>2</b>     |
| <b>Remerciements .....</b>                       | <b>3</b>     |
| <b>Introduction .....</b>                        | <b>.....</b> |
| <b>Ma démarche .....</b>                         | <b>4</b>     |
| <b>Présentation de la mission .....</b>          | <b>5</b>     |
| <b>L'entreprise .....</b>                        | <b>6</b>     |
| <b>Les missions .....</b>                        | <b>.....</b> |
| <b>Objectifs .....</b>                           | <b>7</b>     |
| <b>La création du site web .....</b>             | <b>8-9</b>   |
| <b>La conception de la base de données .....</b> | <b>10-11</b> |
| <b>Démarches suivies .....</b>                   | <b>.....</b> |
| <b>Organisation des missions .....</b>           | <b>12</b>    |
| <b>Processus et méthodes suivis .....</b>        | <b>13</b>    |
| <b>Réalisation des missions .....</b>            | <b>.....</b> |
| <b>Logiciels utilisés .....</b>                  | <b>14-16</b> |
| <b>Grandes étapes des missions .....</b>         | <b>.....</b> |
| <b>Création du site .....</b>                    | <b>17-26</b> |
| <b>Conception de la base de données .....</b>    | <b>27-29</b> |
| <b>Conclusion .....</b>                          | <b>30-31</b> |
| <b>Bibliographie .....</b>                       | <b>32</b>    |

Je voudrais tout d'abord adresser mes remerciements à Franck MAISONNEUVE, mon maître de stage et président de l'association, qui m'a pris en stage et m'a fait confiance pour la refonte du site web.

A ma mère, Florence BLANC, qui m'a proposé à Franck pour le stage et qui m'a aidé lorsque j'avais des questions sur l'association.

A Eric GUDIN qui a pris le temps de m'aider tout au long du stage pour le développement du site et qui m'a permis, de facto, d'améliorer mes compétences générales.

## **Ma démarche**

Mes recherches ont débuté fin décembre, pendant les vacances de Noël. J'ai tout d'abord demandé dans des entreprises du secteur informatique, plus principalement celles spécialisées dans le développement et situées dans la région lyonnaise mais aussi dans d'autres départements tels que l'Ain ou la Saône et Loire.

Malheureusement, je ne peux pas dire que j'ai essuyé des refus puisqu'aucune entreprise ne m'a répondu. Je me suis donc tourné vers de plus grosses entreprises dans plusieurs secteurs différents et qui seraient susceptibles d'avoir un département informatique. Mais je n'ai reçu aucune réponse positive encore une fois.

C'est alors que ma mère, qui a suivi l'évolution de mes recherches, m'a proposé de demander à refaire le site internet de l'association dont elle fait partie. Je connais bien l'association et le site était obsolète alors j'ai accepté.

## **Présentation de la mission**

L'association avait besoin de moi afin d'avoir une refonte complète de leur site web.

Il fallait que le site puisse mieux représenter leurs activités, gérer leur système de mise à disposition de matériel tout en offrant un visage plus moderne.

Je devais donc développer une base de données pour gérer la mise à disposition du matériel, les événements organisés par l'association ainsi qu'un système d'actualités afin que les membres du bureau puissent partager des photos, vidéos ou des documents. Il fallait aussi penser à un nouveau visuel qui rendrait le site plus moderne et plus attirant.

Le poste qui m'était proposé était donc webmaster.

## L'entreprise



### Ce qu'il faut savoir sur l'association

L'association du Comité des Fêtes est une organisation à but non lucratif créée en 1994 et son siège se situe sur la Place de l'Hôtel de Ville à Châtillon sur Chalaronne. Elle permet d'organiser, dans cette même ville, plusieurs événements tout au long de l'année ainsi que de mettre à disposition du matériel pour les événements de particuliers.

L'association compte environ 40 bénévoles dont 5 qui sont membres du bureau.

Le bureau est composé, entre autres, de :

Franck MAISONNEUVE, qui est le président de l'association. Il était mon maître de stage.

Florence BLANC qui est la vice-présidente.

Christine MAISONNEUVE qui est la secrétaire chargée de la communication.

**Objectifs :**

- Continuer d'approfondir mes connaissances en langages de développement (HTML/CSS/PHP/SQL).
- Confirmer la voie dans laquelle je veux me lancer (développement ou réseau). Grâce à ce stage, je sais que je veux aller en développement.
- Apprendre à travailler en autonomie.
- Apprendre à avoir des responsabilités.
- Apprendre à mettre en ligne un site web grâce à un hébergeur.

## **1ère mission : La création du site web**

### Contexte professionnel

Avant mon arrivée, l'association avait un site internet depuis quelques années déjà mais il était plus source de problèmes que bienfaiteur. Le site étant très vieux, la navigation n'était pas facile et les utilisateurs devaient appeler l'association dans le but d'avoir des réponses à leurs questions. De plus, des modifications faites par des gens externes à l'association ont entraîné des bugs et plusieurs pages étaient devenues inaccessibles.

### Enjeu

Comme dit précédemment, le site devait mieux représenter l'association mais aussi offrir un meilleur système de mise à disposition.

La partie vitrine du site devait informer les visiteurs des activités futures de l'organisation avec des affiches par exemple mais aussi les activités passées, cette fois avec des photos ou des vidéos.

### Présentation

Le site internet devait être composé de 5 pages qui présenteraient :

- une page actualités où les membres du bureau pourraient poster de photos ou vidéos par exemple
- une page pour présenter les membres du bureau
- une page pour les événements organisés par l'association
- une page pour répertorier les produits mis à disposition
- une page "panier" de façon à ce que l'utilisateur puisse voir ses produits demandés



Sachant que le site demandé était un gros morceau, mon maître de stage a fait appel à Eric Gudin (auto-entrepreneur de E2G Informatique) qui m'a aidé et conseillé durant toute la durée du stage.

Pour mener à bien cette mission, j'aurai besoin d'utiliser les langages HTML, CSS, PHP, SQL et Javascript.

### Problèmes à résoudre

Pour réaliser les fonctionnalités demandées par l'association, il fallait utiliser beaucoup de php avec des méthodes que je n'avais jamais vues, il a fallu un petit temps pour que je me renseigne sur ces méthodes.

## **2ème mission : La création de la base de données**

### Contexte professionnel

Avant le début du stage, la gestion du matériel disponible avec les quantités, le prix, ou encore les dates de mise à disposition se faisait à la main sur excel. De plus, lorsqu'ils voulaient ajouter ou changer des informations sur le site, ils étaient forcés de changer directement le code source, ce qui limitait les modifications.

### Enjeu

Le principal enjeu était de faciliter la tâche des bénévoles qui s'en occupent en automatisant la gestion des produits ainsi que le système de factures et de cautions, le paiement restant en physique et non par internet. Le second enjeu était de simplifier le changement de contenu des pages afin que les administrateurs du site puissent modifier et rajouter ce que bon leur semble.

### Présentation de la mission

Cette mission consiste à concevoir une base de données qui rassemblerait donc, comme mentionné plus haut, les produits mis à disposition ainsi que le panier des produits choisis mais aussi les éléments de la partie vitrine comme les événements ou le système de post d'actualités. Cette méthode de stockage permettra au site d'être le plus dynamique possible. La base de données devra contenir aussi une partie membres dans le but de recenser ceux qui auraient les droits d'administrateurs.

Il fallait que la base soit complète mais aussi compréhensible pour que les membres de l'association puissent la modifier plus facilement que précédemment.

Pour cette mission, nous nous sommes partagé la tâche avec Eric. Nous avons donc utilisé les langages PHP et SQL.

### Problèmes rencontrés

Le système de mise à disposition et de gestion du panier étant très complexe, relier les tables entre elles n'était pas une chose aisée car notre site devait aussi créer les factures correspondantes. Il fallait donc tout stocker correctement et logiquement afin de réutiliser les données facilement.

## Organisation des missions

Eric étant en entreprise, je travaillais naturellement toute la journée et lui venait m'aider dès qu'il le pouvait. Afin de produire un travail correct, les horaires étaient de 8h30-12h et de 13h30-18h et 5 jours par semaine. Sachant que nous avons beaucoup de travail à faire, il n'était pas rare que nous allongions les horaires pour finaliser le travail de la journée ou de la semaine.

Pour faire un point sur l'avancée des objectifs quotidiens, nous nous appelions tous les jours avec Eric de façon à confirmer le travail effectué, suggérer des modifications et programmer le travail du lendemain.

Toutes les deux semaines, nous avons aussi des réunions en présentiel avec l'association du Comité des Fêtes afin qu'ils puissent voir l'avancée du site, poser des questions et qu'ils puissent, eux aussi, proposer des modifications et expliquer leurs attentes précises sur certaines parties.

Lors de nos appels, pour qu'Eric puisse vérifier le travail que j'avais effectué durant la journée et y ajouter des modifications, nous utilisons le logiciel TeamViewer pour pouvoir contrôler l'écran de l'autre.

En ce qui concerne la mission de la conception de la base de données, nous nous sommes partagés la tâche, Eric a fait la partie panier, qui nécessitait, comme je l'ai dit plus haut, un grand nombre d'informations et j'ai fait la partie événements, actualités et membres.

## Processus et méthodes suivis

Pour mieux nous retrouver dans nos fichiers php, nous avons créé différents fichiers qui contenaient chacun des classes qui, elles-mêmes, regroupaient toutes les fonctions d'une seule et même page web. Ainsi, les pages php faisaient appel aux fichiers "class" qui ne contenaient que les lignes back-end afin d'avoir seulement les lignes front-end dans celles-ci.

Toujours dans le but de simplifier la façon de travailler mais aussi la relecture, un travail de renommage des classes et des identifiants a été effectué. Cette action rend le code plus limpide et plus facilement lisible. Elle nous permet aussi de nous faciliter la tâche lors de l'ajout du CSS.

```
<div class="text_even">
  <div class="title_even">
    <h1>Nos événements</h1>
  </div>
  <div class="desc_events">
    <p>Le comité des fêtes organise chaque année près de 10 évén. Retrouvez les juste en dessous.</p>
  </div>
</div>
<div class="all_events">
  <div class="affiche_events">
    <!-- on affiche tous les événements sélectionnés -->
    <?php foreach($mesevenements->Lignes as $value): ?><a id="myBtn" onClick=MDM_openwindow(<?php echo($value->UrlPopup()); ?>>
      <div class="even" style="background-image: url(../SiteComite/images/evenements/<?= $value->LibEvenement; ?>.jpg);">
        <h1><?php echo($value->NomEvenement) ; ?></h1>
      </div>
    </a>
  </div>
</div>
```

Comme vous pouvez le voir, l'ajout de commentaires a été un élément essentiel pour comprendre le code plus facilement. Nous en avons ajouté dans les fichiers "vus" qui contiennent de l'html mais aussi dans les fichiers "class", comme vous pouvez le voir ci-dessous, qui ne contiennent que des fonctions en PHP.

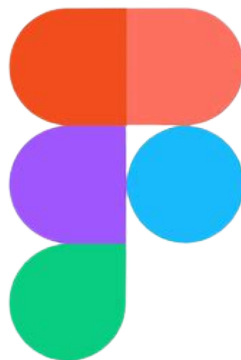
```
/* verification des variables session */
if (isset($_SESSION['Idpanier'])) {
  $this->Idpanier=0;
} else {
  $this->Idpanier=$_SESSION['Idpanier'];
}
$this->DB = $DB;

/*Chargement des lignes */
$stmt = $this->DB->getDbd()->prepare("SELECT IdLigne, IdArticle, Reference,Designation ,Qte, Prix,JourDebut, JourFin,InformationLocation
$stmt->bindParam(':idpanier',$this->Idpanier);
$stmt->execute();
/* on parcourt toutes les lignes */
$i=0;
while ($row = $stmt->fetch(PDO::FETCH_NUM, PDO::FETCH_ORI_NEXT)) {
```

## Les logiciels utilisés

Pour mener à bien ces deux missions, nous avons utilisé plusieurs logiciels qui nous ont servi chacun pour une durée plus ou moins longue.

### Figma :



Afin de créer les maquettes dans les meilleures conditions, nous avons utilisé le logiciel Figma. Figma est une application web d'édition graphique et de conception d'interface qui permet à tous les utilisateurs présent sur la toile de modifier les schémas simultanément.

De plus, ce logiciel est capable de relier entre eux les schémas de page web ce qui permet à l'utilisateur d'être en condition réelle et de vérifier la structure complète et reliée du site.

Cette application étant l'application que nous utilisons dans la plupart de nos projets de cours, la prise en main était déjà effectuée et j'ai pu créer mes maquettes sans avoir besoin de temps d'adaptation.

Looping :

Pour la création de la base de données, nous nous sommes servis de l'application Looping qui est gratuite et facile à prendre en main. Comme pour Figma, j'avais déjà utilisé cette application lors de mes précédents projets en cours, j'avais donc déjà les bases pour m'en servir et n'avais pas besoin de temps d'adaptation.

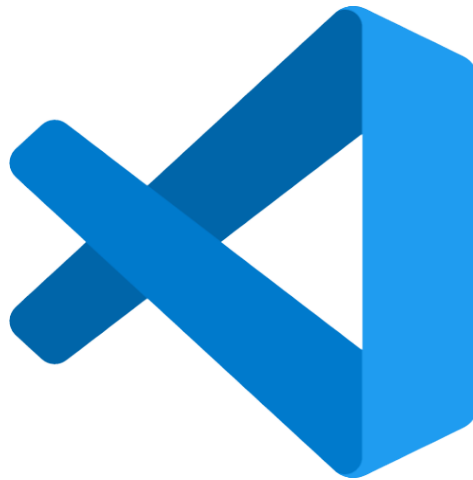
L'un des avantages que propose Looping est la fonctionnalité de retranscrire le modèle de conceptualisation des données en un script SQL que nous n'avons plus qu'à importer dans notre base de données sur Xampp.

Xampp :

Comme nous l'avons évoqué juste avant, nous utilisons l'application Xampp pour gérer et utiliser nos bases de données. XAMPP est un serveur web multiplateforme qui permet aux programmeurs d'écrire et de tester leur code sur un serveur web local. Grâce aux modules MySQL et Apache, ce logiciel nous permet de tester notre site comme sur un hébergeur mais de façon locale. Il nous donne aussi la possibilité de s'occuper de nos bases de données grâce à phpmyadmin.

J'ai choisi d'utiliser ce serveur local car c'est celui que nous utilisons habituellement en cours mais aussi le seul que je connaisse pour l'instant.

### Visual Studio Code :



Afin de développer notre site dans les meilleures conditions, j'ai choisi de le faire sur Visual Studio Code. C'est un éditeur de code open source principalement utilisé pour corriger et réparer les erreurs de codage des applications cloud et Web.

Contrairement à Xampp où je ne connaissais pas d'alternative, utiliser VS code est un choix par rapport à Sublime Text par exemple car VS code propose un grand nombre d'extensions et j'en avais déjà d'installées ici. De plus, comme pour les 3 autres applications avant celle-ci, nous l'utilisons



couramment lors des projets en cours, ce qui ne m'a pas amené à réfléchir longtemps lors du choix d'éditeur de code.

## Les grandes étapes des missions

Dans cette partie, nous allons détailler le cheminement parcouru afin de réaliser chacune des missions. Celles-ci seront partagées en plusieurs sous-parties qui retraceront le déroulé et les moments importants de chaque mission

### **La création du site**

Lors de la création du site, nous avons choisi de segmenter le travail en ne s'occupant que d'une seule page tant qu'elle n'était pas finalisée.

Pour cette tâche, les sous-parties seront :

- Recherche de la structure du site
- Réalisation de la partie panier
- Création de la page produit et de la page article
- Conception de la page actualités
- Reste du site

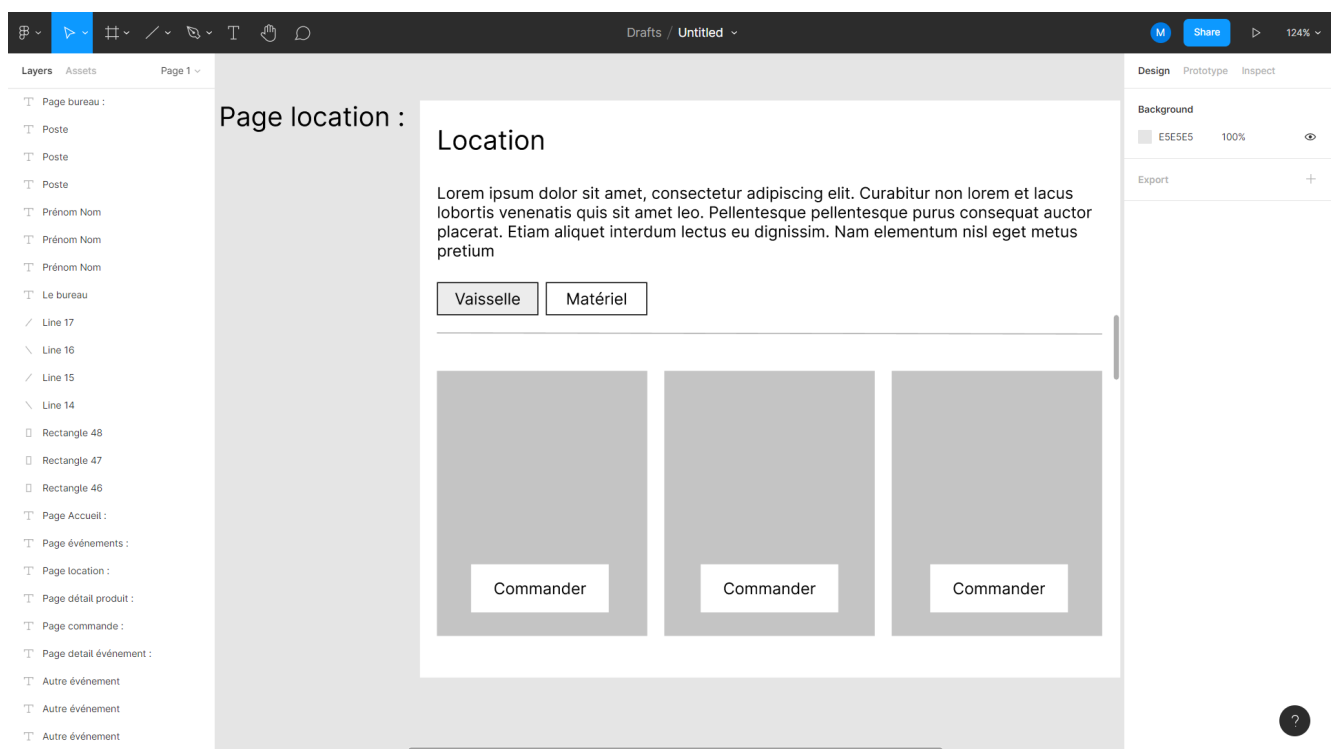
### Recherche de la structure du site

Avant de commencer à coder, il fallait tout d'abord déterminer à quoi ressemblerait le site dans les grandes lignes. Pour ce faire, j'ai effectué une veille prospective dans le but de regrouper dans un Word plus d'une quinzaine de pages d'accueil de site de Comité des Fêtes afin que mon maître de stage puisse choisir quelle "forme" de site il préférerait. Ce dernier a choisi une structure en 3 colonnes (voir



photo modèle ci-dessus), une forme qui permet au site d'être responsive automatiquement.

Une fois la structure choisie, il ne restait plus qu'à faire les maquettes sur Figma. Ici, nous devons trouver un modèle moderne mais aussi navigable facilement puisque l'association touche principalement des personnes âgées qui ne savent pas forcément se diriger sur internet.



Les maquettes produites n'étaient pas colorées puisque la couleur du site n'était pas définie. En effet, le site devant être vivant et accueillant, il y avait le projet de changer les couleurs du site en fonction de la période (Noël, Pâques, etc...).

Cette étape nous a pris une semaine et nous avons en tout 8 maquettes. Ces maquettes n'étant pas définitives, il y en a certaines qui n'ont pas été reproduites.

## Réalisation de la partie panier

La partie panier a été la première partie sur laquelle nous nous sommes penchés puisque c'était certainement la plus dure.

N'ayant jamais fait de système de panier avant, je me suis beaucoup aidé grâce aux recherches sur des sites tels que stackoverflow et w3schools mais aussi et surtout sur des chaînes youtube telles que Grafikart par exemple (voir liens en bibliographie).

Toujours dans la même logique, nous avons créé un fichier "panier.php" et un fichier "panier.class.php" où nous avons créé la classe "panier" afin d'avoir les lignes back-end dans un seul fichier et ne pas avoir de SQL dans le fichier avec les lignes front-end par exemple.

Nous avons donc commencé par créer la fonction "\_\_\_construct" ici à droite, elle nous permet de faire ce qui est inscrit à l'intérieur automatiquement dès que la classe est appelée. A l'intérieur de cette fonction, nous initialisons la session puis nous appelons tous les éléments de la table "TPanierLignes" que nous rangeons dans un tableau de façon à ce que tout soit affiché facilement.

```
class LignePanier{
    public int $idLigne;
    public int $idProduit;
    public string $Reference;
    public string $Designation;
    public int $Qte;
    public float $Prix;
    public int $Jours;
    public string $InformationLocation;
```

```
public function __construct($DB){
    if(!isset($_SESSION)){
        session_start();
    }
    /* verification des variables session */
    if (!isset($_SESSION['Idpanier'])) {
        $this->Idpanier=0;
    } else {
        $this->Idpanier=$_SESSION['Idpanier'];
    }
    $this->DB = $DB;

    /*Chargement des lignes */
    $stmt = $this->DB->getDbd()->prepare("SELECT IdLigne, IdArticle, Reference, Desi
    $stmt->bindParam(':idpanier',$this->Idpanier);
    $stmt->execute();
    /* on parcourt toutes les lignes */
    $i=0;
    while ($row = $stmt->fetch(PDO::FETCH_NUM, PDO::FETCH_ORI_NEXT)) {
        /* $data = $row[0] . "\t" . $row[1] . "\t" . $row[2] . "\n"; */
        $this->Lignes[$i] = new LignePanier();
        $this->Lignes[$i]->idLigne = $row[0];
        $this->Lignes[$i]->idProduit = $row[1];
        $this->Lignes[$i]->Reference = $row[2];
        $this->Lignes[$i]->Designation = $row[3];
        $this->Lignes[$i]->Qte = $row[4];
        $this->Lignes[$i]->Prix = $row[5];
        $this->Lignes[$i]->JourDebut = $row[6];
        $this->Lignes[$i]->JourFin = $row[7];
        $this->Lignes[$i]->InformationLocation = $row[8];

        /* calcul du prix du panier */
        $this->CumulPanier= $this->CumulPanier + $this->Lignes[$i]->GetPrixLigne()
        $i=$i+1;
    }
}
```

Nous avons ensuite créé la classe “LignePanier” que vous pouvez voir dans le tableau.

Cette classe nous permet d’initialiser les variables que nous utilisons ensuite dans le tableau. Cette classe nous sert aussi à créer des fonctions qui nous permettent d’avoir des informations propres à chaque ligne comme le prix de la ligne par exemple.

Une fois les produits choisis stockés dans le tableau, nous avons créé 3 fonctions qui nous permettent d’agencer le panier. Nous avons d’abord :

- La fonction “AjouterLignePanier”. Son nom est éloquent, nous vérifions d’abord si le panier est créé, si non, nous le créons en ajoutant la date de création et son adresse IP. Une fois que le panier est récupéré ou créé, nous sélectionnons le bon panier et mémorisons ses identifiants.

```
if($this->Idpanier == 0){
    $vIdpanier=0;
    $vDateCreation = date('Y-m-d h:m:s');
    $vAdresseIp = $_SERVER ['REMOTE_ADDR'];
    $sql = 'INSERT INTO TPanier (AdresseIP, DateCreation) VALUES (:AdresseIp, :DateCreation)';
    $stmt=$this->DB->getBdd()->prepare($sql);
    $stmt->bindParam(':AdresseIp',$vAdresseIp);
    $stmt->bindParam(':DateCreation', $vDateCreation);
    $stmt->execute();
    /* lecture de l'Idpanier ajouté */
    $stmt = $this->DB->getBdd()->prepare("SELECT IdPanier FROM TPanier WHERE AdresseIP = :AdresseIp AND DateCreation = :DateCreation");
    $stmt->bindParam(':AdresseIp',$vAdresseIp);
    $stmt->bindParam(':DateCreation', $vDateCreation);
    $stmt->execute();
    $result = $stmt->fetch();
    $vIdpanier=$result[0];
    /* on mémorise le N° de panier */
    $this->Idpanier=$vIdpanier;
    /* on memorise la variable session */
    $_SESSION['Idpanier']=$vIdpanier;
}
```

Une fois la question de panier existant ou non résolu, nous n’avons plus qu’à ajouter les éléments de l’article choisi dans une ligne de la table.

```
/* On ajoute la ligne */
$sql = "INSERT INTO TPanierLignes (IdPanier,IdA
$stmt = $this->DB->getBdd()->prepare($sql);
$stmt->bindParam(':idpanier',$this->Idpanier);
$stmt->bindParam(':Qte', $vQte);
$stmt->bindParam(':JourDebut', $vJourDebut);
$stmt->bindParam(':JourFin', $vJourFin);
$stmt->bindParam(':IdArticle', $vArticle);
$stmt->execute();
}
```

- La fonction “SupprimerLignePanier”. Celle-ci sert à supprimer un article de son panier en fonction de ses identifiants de ligne et d'article.

```
public function SupprimerLignePanier(){
    $VIdligne=$_GET['IDLIGNE'];
    $sql = "DELETE FROM TPanierLignes WHERE IdLigne = :Idligne AND IdPanier = :Idpanier";
    $stmt = $this->DB->getBdd()->prepare($sql);
    $stmt->bindParam(':Idligne', $VIdligne);
    $stmt->bindParam(':Idpanier', $this->Idpanier);
    $stmt->execute();
}
```

- La fonction “ModifierQteLignePanier”. Celle-ci permet de modifier la quantité choisie d'un seul article grâce à la requête SQL “DELETE”.

```
public function ModifierQteLignePanier(){
    $VIdligne=$_GET['IDLIGNE'];
    $VQte=$_GET['QTE'];
    $sql = 'UPDATE TPanierLignes SET Quantite=:Qte WHERE IdLigne = :Idligne AND IdPanier = :Idpanier';
    $stmt = $this->DB->getBdd()->prepare($sql);
    $stmt->bindParam(':Idpanier', $this->Idpanier);
    $stmt->bindParam(':Idligne', $VIdligne);
    $stmt->bindParam(':Qte', $VQte);
    $stmt->execute();
}
```

Cette étape nous a pris environ 3 semaines car ces techniques étaient nouvelles pour moi, nous n'allions donc pas à notre vitesse maximale.

## Création de la page produit et de la page article

```
INSERT INTO `tarticles` (`IdArticle`, `Reference`, `Designation`, `Prix`, `IdPanier`)
VALUES
(1, 'VAI001', 'Verres ballon 19 cl en lot de 30\r\n', '2.00', 1, '', 0, '', 0),
(2, 'VAI002', 'Verres ballon 19 cl en lot de 30\r\n', '1.00', 1, '', 0, '', 0),
(3, 'VAI003', 'Verres ballon 12 cl en lot de 12\r\n', '1.00', 1, '', 0, '', 0),
(4, 'VAI004', 'Verres à eau 30 cl en lot de 6\r\n', '0.00', 1, '', 0, '', 0),
(5, 'VAI005', 'Coupes à champagne en lot de 20\r\n', '1.00', 1, '', 0, '', 0),
(6, 'VAI006', 'Flûtes en lot de 6\r\n', '0.00', 1, '', 0, '', 0),
(7, 'VAI007', 'Petit verres bistrot\r\n', '0.00', 1, '', 0, '', 0),
(8, 'VAI008', 'Seaux à champagne\r\n', '0.00', 1, '', 0, '', 0),
(9, 'VAI009', 'Pichets verre en lot de 12\r\n', '4.00', 1, '', 0, '', 0),
(10, 'VAI010', 'Pichets plastique\r\n', '0.00', 1, '', 0, '', 0),
(11, 'VAI011', '10 fourch-10 couteaux-10 p.cuill\r\n', '3.00', 1, '', 0, '', 0),
(12, 'VAI012', 'Fourchettes en lot de 10\r\n', '1.00', 1, '', 0, '', 0),
(13, 'VAI013', 'Couteaux en lot de 10\r\n', '1.00', 1, '', 0, '', 0),
(14, 'VAI014', 'Petites cuillères en lot de 10\r\n', '1.00', 1, '', 0, '', 0),
(15, 'VAI015', 'Grosses cuillères en lot de 10\r\n', '1.00', 1, '', 0, '', 0),
(16, 'VAI016', 'Corbeille à pain inox\r\n', '0.00', 1, '', 0, '', 0),
(17, 'VAI017', 'Tasses à café en lot de 32\r\n', '2.00', 1, '', 0, '', 0),
(18, 'VAI018', 'Sous tasse\r\n', '0.00', 1, '', 0, '', 0),
(19, 'VAI019', 'Assiettes plates \r\n\r\n' en lot de 20\r\n', '2.00', 1, '', 0, '', 0),
(20, 'VAI020', 'Assiettes à dessert \r\n\r\n' en lot de 18\r\n', '1.00', 1, '', 0, '', 0),
(21, 'VAI021', 'Assiettes plates \r\n\r\n' en lot de 20\r\n', '0.00', 0, '', 0),
(22, 'VAI022', 'Assiettes à dessert \r\n\r\n' en lot de 18\r\n', '1.00', 1, '', 0, '', 0),
(23, 'VAI023', 'Assiettes creuses\r\n', '0.00', 1, '', 0, '', 0),
(24, 'VAI024', 'Plats inox ovales 60 cm\r\n', '0.00', 1, '', 0, '', 0),
(25, 'VAI025', 'Plats inox ovales 45 cm\r\n', '0.00', 1, '', 0, '', 0),
```

Après avoir développé la page panier, nous sommes passés à l'affichage des produits et de la page description de chaque produit. Avant de commencer à coder le PHP de la partie produits, il a fallu remplir la base de données avec les produits mis à disposition par l'association.

De façon à ce que les produits ne soient pas tous affichés sur la même page, nous les avons séparés en 3 familles, la vaisselle, le matériel et les barnums. Une fois ces produits classés par famille, nous avons pu commencer à organiser la page.

Nous appelons la classe “Articles” et encore ici, nous commençons avec la fonction “\_\_construct” qui nous permet d’appeler d’abord les familles de produits puis les produits eux-mêmes que nous rangeons dans un tableau. Nous réutilisons la même technique que pour le panier en créant une autre classe, “ArticleVisu”, afin d’initialiser les variables.

```
public function __construct($DB,$IdFamille){
    $this->DB = $DB;
    /* Chargement du nom de la famille */
    $stmt = $this->DB->getBdd()->prepare("SELECT Famille FROM Tarticle");
    $stmt->bindParam(':IdFamille',$IdFamille);
    $stmt->execute();
    $row = $stmt->fetch();
    $this->Famille = $row[0] ;

    /*Chargement des lignes */
    $stmt = $this->DB->getBdd()->prepare("SELECT IdArticle, Reference,");
    $stmt->bindParam(':idfamille',$IdFamille);
    $stmt->execute();
    /* on parcourt toutes les lignes */
    $i=0;
    while ($row = $stmt->fetch(PDO::FETCH_NUM, PDO::FETCH_ORI_NEXT)) {
        /* $data = $row[0] . "\t" . $row[1] . "\t" . $row[2] . "\n"; */
        $this->Lignes[$i] = new ArticleVisu();
        $this->Lignes[$i]->IdArticle = $row[0] ;
        $this->Lignes[$i]->Reference = $row[1];
        $this->Lignes[$i]->Designation = $row[2];
        $this->Lignes[$i]->Prix = $row[3];
        // $this->Lignes[$i]->DesignationLongue = $row[4];
        // $this->Lignes[$i]->InformationLocation = $row[5];
        $i=$i+1;
    }
}
```

Maintenant que nous avons sélectionné les articles selon leur famille et que nous les avons affichés, il nous reste la page article, celle qui présenterait le produit choisi, à développer.

Nous étions partis initialement sur un haut de page qui présenterait le produit avec sa photo, son nom, son prix, etc... En dessous se trouverait un calendrier qui permettrait de réserver directement l’article sans avoir besoin d’appeler un membre du bureau pour réserver. Pour réaliser ce calendrier, nous avons créé une classe “Month” qui contient 8 fonctions :

```
public function __construct(?int $month = null, ?int $year = null){
    if ($month === null || $month < 1 || $month > 12){
        $month = intval(date('m'));
    }
    if ($year === null){
        $year = intval(date('Y'));
    }
    $this->MoisEncours = $month;
    $this->AnneeEncours = $year;
}

/**
 * Renvoie le premier jour du mois
 */
public function GetPremierJours(): \DateTime {
    return new \DateTime("{$this->AnneeEncours}-{$this->MoisEncours}-01");
}

/**
 * Retourne le mois en toutes lettres (ex: Mai 2022)
 */
public function MoisString(){
    return $this->TableauMois[$this->MoisEncours - 1];
}
```

```
/**
 * Renvoie le nombre de semaine dans le mois
 */
public function GetNombreSemaine(){
    $start = $this->GetPremierJours();
    $end = (clone $start)->modify('+1 month -1 day');
    $weeks = intval($end->format('W')) - intval($start->format('W')) + 1;
    if ($weeks < 0){
        $weeks = intval($end->format('W'));
    }
    return $weeks;
}

public function DansleMois(\DateTime $date){
    return $this->GetPremierJours()->format('Y-m') === $date->format('Y-m');
}

/**
 * Est ce que le jour est dans le mois en cours ?
 */
public function EstAvantAujourd'hui(\DateTime $date){
    return ($date->format('Y-m-d') > date('Y-m-d')) ;
}

public function MoisProchain(){
    $MoisEncours = $this->MoisEncours + 1;
    $AnneeEncours = $this->AnneeEncours;
    if ($MoisEncours > 12){
        $MoisEncours = 1;
        $AnneeEncours += 1;
    }
}
```

Malheureusement, les dates de dépôt et de retrait de matériel n'étant pas fixes, ce système était trop compliqué. Après de longues discussions et plusieurs propositions, nous avons donc choisi de n'afficher que les détails de l'article.

Concernant la réservation, nous avons mis un formulaire dans cette fiche article. Le visiteur choisirait la date de début et la date de fin et l'enverrait au président de pour qu'il valide ou non la mise à disposition. Cette étape nous a pris environ deux semaines.

```
<form class="choix_date" method="POST" name="datechoisie" action="article.p
  <div class="redirection">
    <p>Choisissez votre quantité :</p>
    <select name="quantite">
      <?php $i = 1;
      while ($i <= $article->NombreDisponibilites()) : ?>
        <option value="<?php echo($i); ?>"><?php echo($i); ?></opti
      <?php $i = $i + 1;
      endwhile; ?>
    </select>
  </div>
  <div class="redirection">
    <p>Choisissez la date qui vous convient : </p>
  </div>
  <div class="input_date">
    <p>Du </p>
    <input type="date" name="startdate">
    <p> au </p>
    <input type="date" name="enddate">
  </div>
  <div class="input_confirm">
    <a href="panier2.php?addpanier&Article=<?php echo($_GET['id']); ?>8
  </div>
</form>
```

### Conception de la page actualités

Cette page est la page d'accueil du site, lorsque les visiteurs cliqueront pour aller sur le site, ils atterriront sur cette page, il faut donc qu'elle soit visuelle, accueillante et démonstratrice de l'activité du Comité des Fêtes.

Comme sur le modèle choisi (voir photo page 13), nous avons choisi de séparer la colonne centrale en 2 colonnes. La colonne de droite serait, comme sur notre modèle, une colonne où seraient présentés les futurs événements. Celle de gauche serait consacrée aux actualités, des posts que les membres du bureau pourraient eux-mêmes publier.

Afin d'afficher ces données, nous utilisons toujours le même principe que pour les autres pages avec la classe "Actualités".

La demande avait été faite de pouvoir avoir une ou plusieurs photos par actualités, il fallait donc développer un moyen d'adapter dynamiquement le carrousel selon s'il y avait une ou plusieurs photos.



Pour cela nous avons utilisé deux fonctions. La première, “ChargeImage”, sert à stocker sous forme de tableau toutes les images dont le nom commencerait par un même texte (ici “Actu\_1”). La deuxième fonction sert à compter les lignes du tableau de la fonction précédente afin de savoir le nombre de photos sélectionnées.

```
public function ChargeImage(){
    $dir = './images/actualite/';
    $this->mesImages = glob($dir.'Actu_1*.jpg');
}
public function NombreImages(){
    return (count($this->mesImages));
}
```

Une fois ces deux fonctions créées, nous n'avons plus qu'à coder le carrousel en utilisant ces dernières pour qu'il s'adapte automatiquement en fonction du nombre d'images.

```
$i = 1;
$f = 1;
$ev->ChargeImage();
$nbimages = $ev->NombreImages();
<section class="carousel" aria-label="Gallery">
    <ol class="carousel__viewport">
        <?php foreach ($ev->mesImages as $img) : ?>
            <li id="<?php echo($value->IdActualite); ?>carousel__slide<?php echo($i); ?>" tabindex="0" class="carousel__slide">
                <div class="carousel__snapper">
                    <a href="#"<?php echo($value->IdActualite); ?>carousel__slide<?php if($i == 1){
                        echo($nbimages);
                    }else{
                        echo($i - 1);
                    } ?>"
                    class="carousel__prev">Go to last slide</a>
                    <a href="#"<?php echo($value->IdActualite); ?>carousel__slide<?php if ($i == $nbimages){
                        $i = 1;
                        echo($i);
                    }else{
                        echo($i + 1);
                    } ?>"
                    class="carousel__next">Go to next slide</a>
                </div>
            </li>
            <?php $i = $i + 1;
        <?php endforeach; ?>
    </ol>
```



Nous nous sommes ensuite concentrés sur la colonne des événements pour que les visiteurs puissent tout de suite voir les prochains événements. A la place de rediriger vers une nouvelle page web nous avons choisi d'ouvrir un pop-up.

```
<h3>Nos événements</h3>
<?php foreach($even->Lignes as $value): ?>
<a onClick=MDM_openWindow(<?php echo($value->UrlPopup()); ?>)><div class="evenement">
  <div class="date_even">
    <?php echo ($value->DateEvenement);?>
  </div>
  <div class="nom_even">
    <?php echo ($value->NomEvenement); ?>
  </div>
```

La fonction que vous voyez qui se nomme "UrlPopup" est une fonction qui retourne le lien de la page que l'on veut ouvrir.

Afin d'ouvrir cette pop-up, nous avons utilisé une fonction Javascript :

```
function MDM_openWindow(theURL,winName,features)
{
var _W=window.open(theURL,'Interface1','width=1200,height=800,
_W.focus();
_W.moveTo(175,150);
}
```

- "window.open" est la fonction javascript qui permet d'ouvrir une fenêtre de navigateur secondaire vers l'URL renseignée et aux dimensions inscrites dans les parenthèses.
- "\_W.focus" sert à viser principalement cette nouvelle page.
- "\_W.moveTo" est la fonction qui permet de délimiter les marges sur les côtés et en haut.

Cette étape s'est déroulée sans anicroche et nous a pris un peu moins d'une semaine.

## La création de la base de données

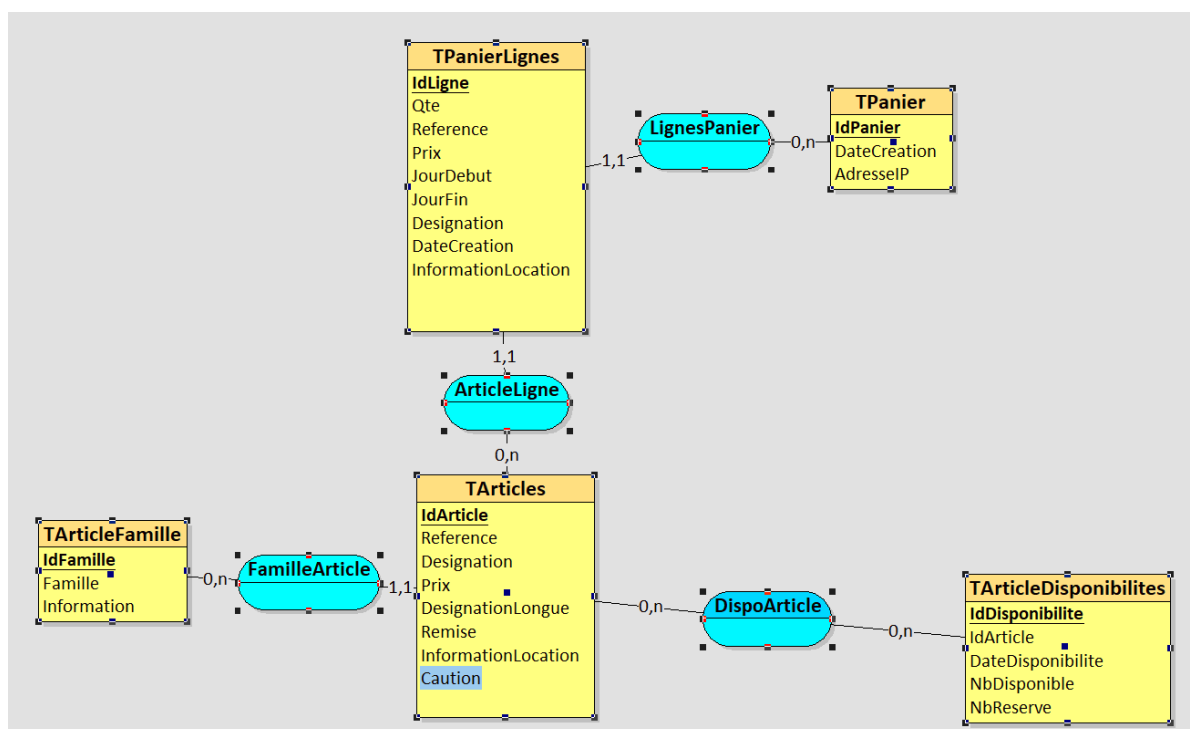
Pour que le site soit le plus dynamique possible, il faut utiliser une base de données.

Cette mission sera divisée en plusieurs sous-parties :

- La modélisation avec le modèle conceptuel des données (MCD)
- La liaison de la base de données au site

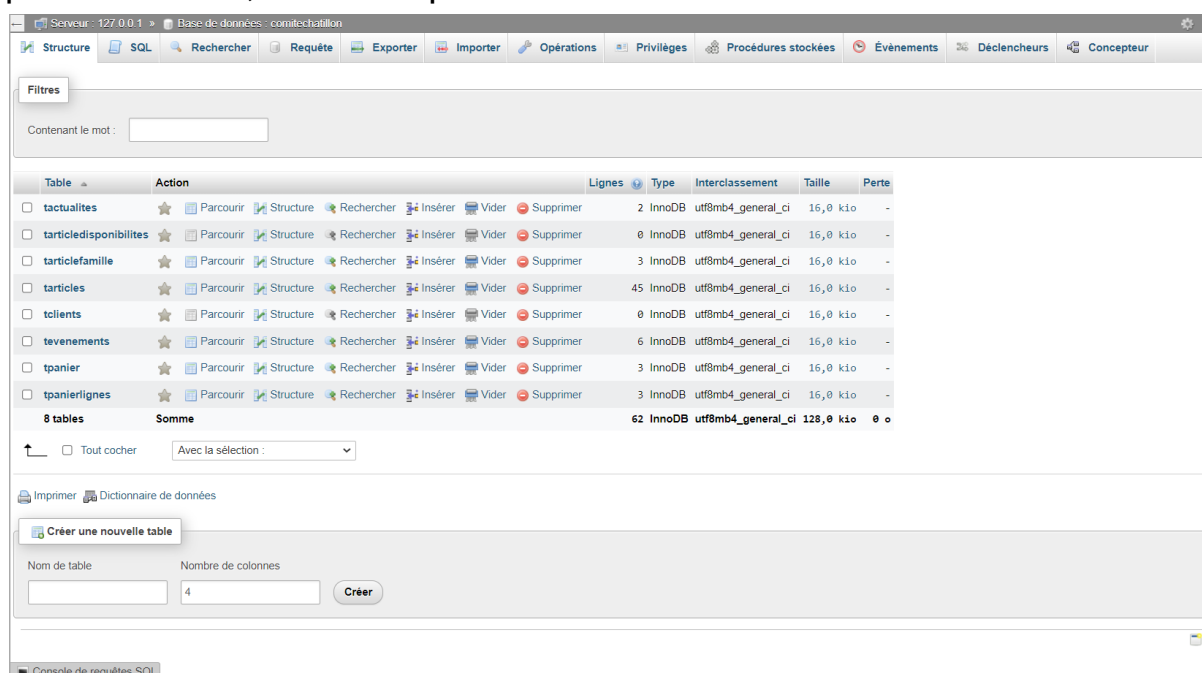
### La modélisation de la base de données

Comme dit précédemment, nous avons modélisé notre MCD sur Looping. Mais nous n'avons fait que la partie du panier et des articles car c'était celle qui nécessitait le plus de liens et qui avait besoin d'être la plus claire possible.



Ceci était une partie de la base de données qui était temporaire puisque nous avons fait des modifications par la suite en supprimant les associations et déplaçant les bons identifiants aux bons endroits par exemple.

Nous avons transformé ce schéma en script SQL que nous avons importé sur Xampp puis nous avons ensuite ajouté, à la main, les tables nécessaires pour les autres pages qui n'avaient aucun lien avec cette partie. Au total, nous comptons 8 tables dans notre base de données.



Une fois que toutes les tables ont été importées, il a fallu remplir les tables comme celle des articles par exemple, avec tous les éléments qu'il fallait.

### Liaison de la base de données jusqu'au site

La base de données étant l'endroit où l'association regroupe toutes ses informations, c'est naturellement un des éléments qui nécessitent le plus de sécurité. Pour ce faire, nous avons utilisé le PHP et la fonction PDO. Cette dernière étant capable d'exécuter des requêtes SQL de nombreuses façons, elle nous protège donc des injections SQL.

Nous avons donc commencé par créer une classe qui se nomme “DB” puis nous avons ensuite renseigné nos informations de connexion.

```
class DB{  
    private $host = 'localhost';  
    private $username = 'root';  
    private $password = '';  
    private $database = 'comitechatillon';  
    private $db;
```

Dans la fonction “\_\_construct” ci-dessous, nous avons stocké nos informations de connexion dans des variables que nous ajoutons ensuite dans notre fonction PDO. Puis, nous gardons dans un tableau le mode d’erreur si cette dernière intervient.

```
public function __construct($host = null, $username = null, $password = null, $database = null){  
    if($host != null){  
        $this->host = $host;  
        $this->username = $username;  
        $this->password = $password;  
        $this->database = $database;  
    }  
    try {  
        $this->db = new PDO('mysql:host='.$this->host.';dbname='.$this->database, $this->username, $this->password, array(  
            PDO::MYSQL_ATTR_INIT_COMMAND => 'SET NAMES UTF8',  
            PDO::ATTR_ERRMODE => PDO::ERRMODE_WARNING  
        ));  
        /*$this->db = new PDO('mysql:host='.$this->host.';dbname='.$this->database, $this->username, $this->password ); */  
    }catch(PDOException $e){  
        die('<h1>Impossible de se connecter à la base de données</h1><br>'. $e );  
    }  
}
```

Si le site ne réussit pas à se connecter à la base de données, alors nous récupérerons l’erreur que nous affichons dans un message d’erreur.

Maintenant que nous sommes connectés, nous créons 2 fonctions, une qui nous aidera pour les requêtes futures qui se nomme query et une autre pour retourner la variable DB :

```
public function query($sql, $data = array()){  
    $req = $this->db->prepare($sql);  
    $req->execute($data);  
    return $req->fetchAll(PDO::FETCH_OBJ);  
}  
public function getBdd(){  
    return $this->db;  
}
```

A présent, pour que nous puissions utiliser cette connexion, nous n'avons plus qu'à effectuer cette démarche sur chaque page où nous voulons l'utiliser.

```
require 'classes/db.class.php';  
$DB =new DB();
```

### Analyse des conditions de travail

Ce stage a été une très bonne opportunité pour moi de découvrir ce que c'est de prendre part à un projet comme celui-là. De plus, cela m'a permis d'apprendre à avoir des responsabilités dans le milieu professionnel.

Durant tout le stage, j'ai été accompagné par Eric qui, dès que je coinçais, prenait le temps de m'aider et de me guider sur certaines choses, ce qui m'a permis de progresser plus rapidement que si j'avais été tout seul pour effectuer le travail. Mon maître de stage n'hésitait pas non plus à répondre aux questions que je me posais.

Le fait d'être en télétravail était parfois un désavantage comme le fait d'être tout le temps tout seul et donc de pouvoir communiquer plus facilement mais ce mode de travail avait aussi des bons côtés. J'ai pu apprendre à être autonome ou en tout cas m'améliorer dans la façon de l'être. J'ai pu aussi avancer à mon rythme même si j'avais quelques fois des dates butoirs, ce qui m'a permis d'approfondir les domaines dans lesquels j'étais en retard par rapport au travail demandé.

Par ailleurs, ce stage s'est toujours déroulé dans une bonne ambiance étant donné que je connaissais déjà l'association et ses membres.

### Apport des missions pour le stagiaire

Bon nombre d'objectifs inscrits plus hauts (voir page 6) ont été validés. Tout d'abord, les objectifs d'apprendre à travailler en autonomie et d'apprendre à avoir des responsabilités ont été réussis comme je l'ai dit quelques lignes plus haut.

J'ai pu approfondir mes connaissances des langages de programmation. Si mon niveau n'a pas eu une évolution exponentielle en HTML et CSS puisque nous nous en sommes très peu servis, je peux désormais dire que je possède une bonne maîtrise du PHP puisque nous avons utilisé ce langage durant une bonne majorité du stage et que je connais les

bases du Javascript. J'ai aussi exploité et administré une base de données dans un environnement client-serveur.

Ce stage m'a permis aussi de confirmer mon envie de me lancer dans la voie développement plutôt que dans la voie réseau. Néanmoins, après bientôt un an de développement front-end, j'aspire dorénavant à me concentrer majoritairement sur le développement back-end qui me donne plus de plaisir.

Malheureusement, certains objectifs n'ont pas été remplis. En effet, celui d'apprendre à mettre en ligne un site sur un hébergeur n'a pas été effectué puisque malgré ces deux mois, nous n'avons pas encore fini le site et que sa diffusion est prévue pour début juillet.

### Prochain stage

Après avoir fait un stage en télétravail comme celui-ci, j'ambitionne d'être en entreprise lors de mon prochain stage car cela me permettra de découvrir l'environnement professionnel dans son ensemble, des horaires jusqu'aux exigences. Sachant que je veux désormais me lancer à fond dans la voie du développement, j'aimerais grandement que ma prochaine entreprise soit spécialisée dans ce même domaine afin que je sois dans une atmosphère qui me permette de progresser et d'acquérir des connaissances le plus possible et le plus rapidement possible.

**Merci d'avoir lu mon rapport de stage !**

<https://stackoverflow.com>

<https://www.w3schools.com>

<https://codepen.io>

<https://github.com>

<https://www.youtube.com/watch?v=OX3FxG>

<https://www.youtube.com/watch?v=t0s7ycR1lb8>