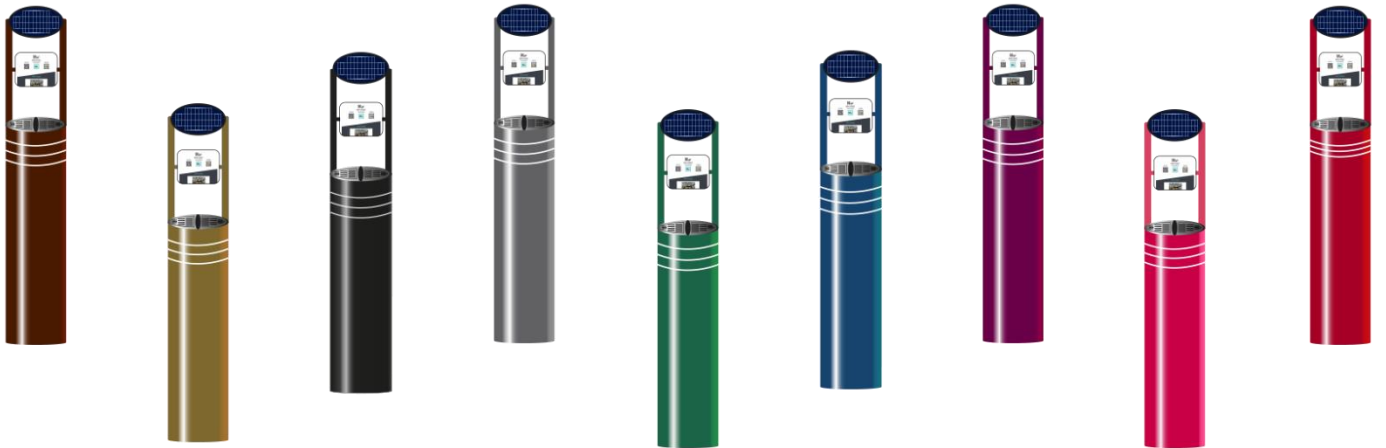


# DOSSIER PROJET

PROJET H'UP : création d'un site vitrine

Réalisé par Célia Lachenal



# SOMMAIRE

## 1<sup>ère</sup> PARTIE : DEVELOPEMENT DE LA PARTIE FRONT-END DU SITE D'H'UP (page 3)

- I- Compétences visées pour le titre couvertes par ce projet : -----p.4
- II- Introduction du projet : -----p.4
  - 1) Le contexte :
  - 2) Le concept :
  - 3) Les recommandations :
  - 4) Ma mission : -----p.5
- III- Maquetter une application web : -----p.5
  - 1) L'identité visuelle :
  - 2) L'arborescence du site : ----- p.6
  - 3) Les wireframes et maquettes : ----- p.7
- IV- Intégration des maquettes : ----- p.10
  - 1) Création du menu burger :
  - 2) Les médias queries : ----- p.12

## 2<sup>ème</sup> PARTIE : DEVELOPEMENT DE LA PARTIE BACK- END DU SITE D'H'UP

- I- Création de la base de données : ----- p.14
- II- Traitement PHP des formulaires de connexion et d'inscription :
- III- Documentation : ----- p.16
- Conclusion : ----- p.18

# 1<sup>ère</sup> PARTIE : DEVELOPPEMENT DE LA PARTIE FRONT-END DU SITE D'H'UP



## **I- Compétences visées pour le titre couvertes par ce projet :**

Au travers de ce projet, j'ai eu l'occasion d'exploiter les compétences suivantes :

- Maquetter une application (CP1)
- Réaliser une interface utilisateur web statique et adaptable (CP2)
- Développer une interface utilisateur web dynamique (CP3)
- Créer une base de données (CP5)
- Développer les composants d'accès aux données (CP6)
- Développer la partie back-end d'une application web ou web mobile

## **II- Introduction du projet :**

Afin mettre en pratique mes connaissances, j'ai souhaité réaliser un site from scratch.

J'ai donc imaginé la société H'up et son cendrier connecté destiné aux zones fumeurs des entreprises.

J'ai conçu puis réalisé un site vitrine pour qu'H'up puisse promouvoir son cendrier le H'up One tout en respectant le cahier des charges.

### **1) Le contexte :**

En France, 30 Milliards de mégots sont jetés au sol chaque année alors que chacun d'entre eux contiennent plus de 4000 substances nocives. Ils sont donc une source importante de pollution.

Dans un souci de préservation de l'environnement, Julie Arnoud veut inciter les fumeurs à adopter un comportement plus responsable et écologique.

Elle imagine alors un cendrier connecté, ludique et design, le H'UP One.

Julie fonde alors l'entreprise H'UP qui a pour mission de concevoir et de distribuer ce cendrier innovant.

Les ventes des H'up One ayant grimpées en flèche, Julie souhaite refondre son site WordPress pour un site vitrine from scratch plus haute-gamme à l'image de son produit.

### **2) Le concept :**

Le H'UP One s'il est connecté à internet en wifi, il offre également, via une application mobile, de nombreuses fonctionnalités permettant aux utilisateurs d'allier au geste écologique et civique un environnement ludique et informatif.

En effet, en installant un ou plusieurs cendriers devant leurs locaux ou dans leurs zones fumeurs, les entreprises peuvent proposer aux utilisateurs des concours, ou jeux les incitant à utiliser le H'UP One plutôt que le sol pour jeter leurs mégots.

En outre, Julie ARNOUD souhaite confier à une entreprise spécialisée la collecte, la dépollution et le recyclage des mégots des H'UP.

### **3) Les recommandations :**

À la suite des études marketing effectuées, nous préconisons :

- Un site vitrine ergonomique à la navigation fluide et intuitive intégrant :
  - Un formulaire de contact permettant aux entreprises intéressées de demander des informations sur le produit

- La possibilité de créer un compte utilisateur et de s'y connecter
- Design mobile first : important pour le référencement naturel du site. De plus, une seule URL aide les utilisateurs à partager plus facilement les contenus du site sur les réseaux sociaux.
- Développement d'une communauté sur les réseaux sociaux afin d'augmenter la notoriété de la marque

#### 4) Ma mission :

Développeuse freelance, j'ai pour mission d'accompagner H'UP dans le cadre du développement de son site Internet. Son produit doit être présenté dans un site vitrine afin de conquérir les marchés déterminés.

Dans ce cadre, je dois effectuer les missions suivantes :

- Mise en avant du concept
- Réalisation de l'identité visuelle d'H'UP : planche de tendance, création du logo, charte graphique
- Conception du site vitrine du cendrier connecté : arborescence du site, wireframes, maquettes pour la version desktop et la version mobile
- Développer le site : réalisation d'une interface utilisateur conforme aux maquettes

### III- Maquetter une application web :

#### 1) L'identité visuelle :

- Création du logo et de la favicon d'H'up :

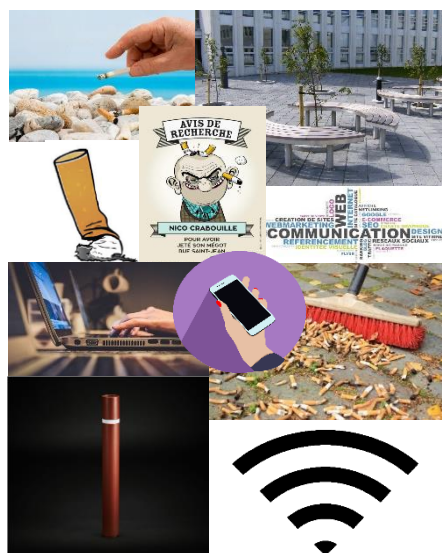


Le logo



La favicon

- Création de La planche tendance :



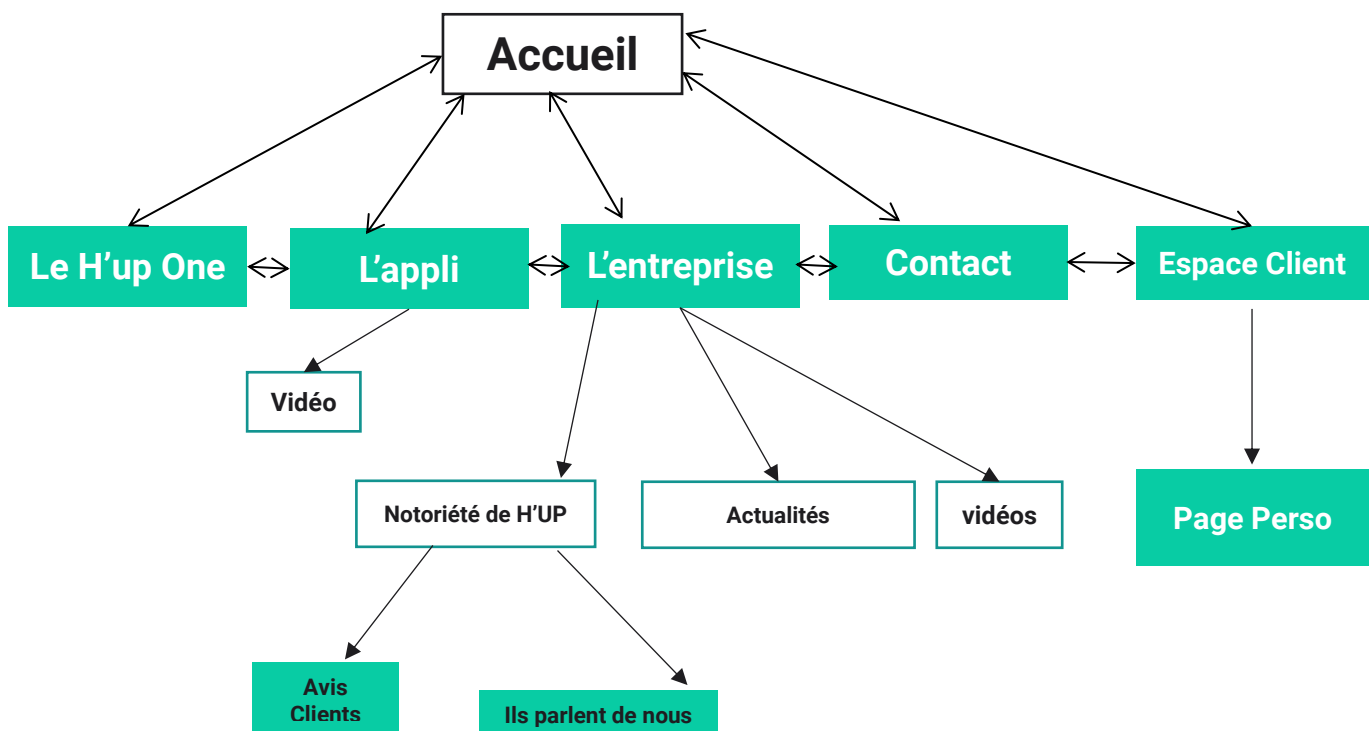
- Les couleurs :



- Les polices :

<b>Poppins regular :</b> ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz 0123456789	<b>Roboto regular :</b> ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz 0123456789
<b>Oswald regular :</b> ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz 0123456789	

2) L'arborescence du site :



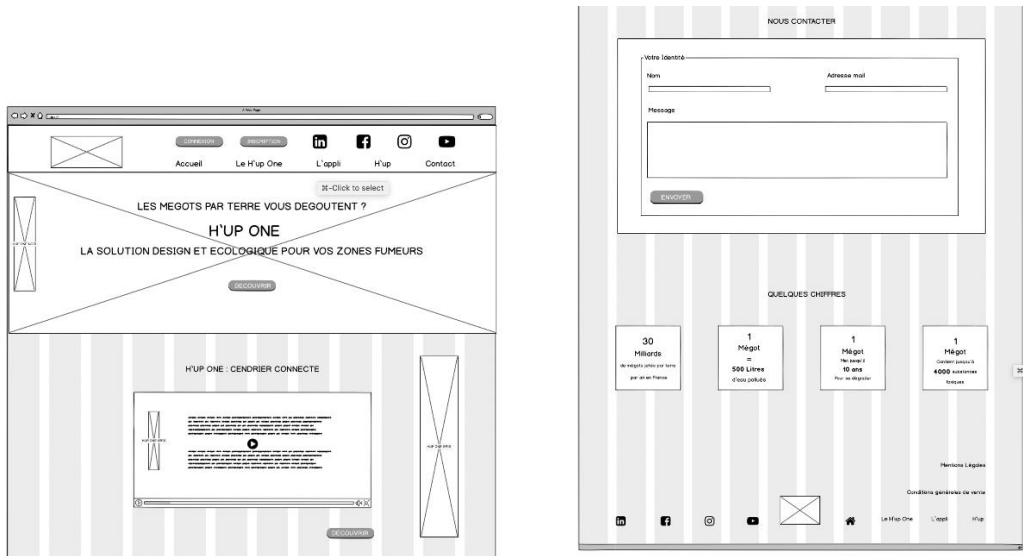
### 3) Wireframes et maquettes :

#### - Les wireframes :

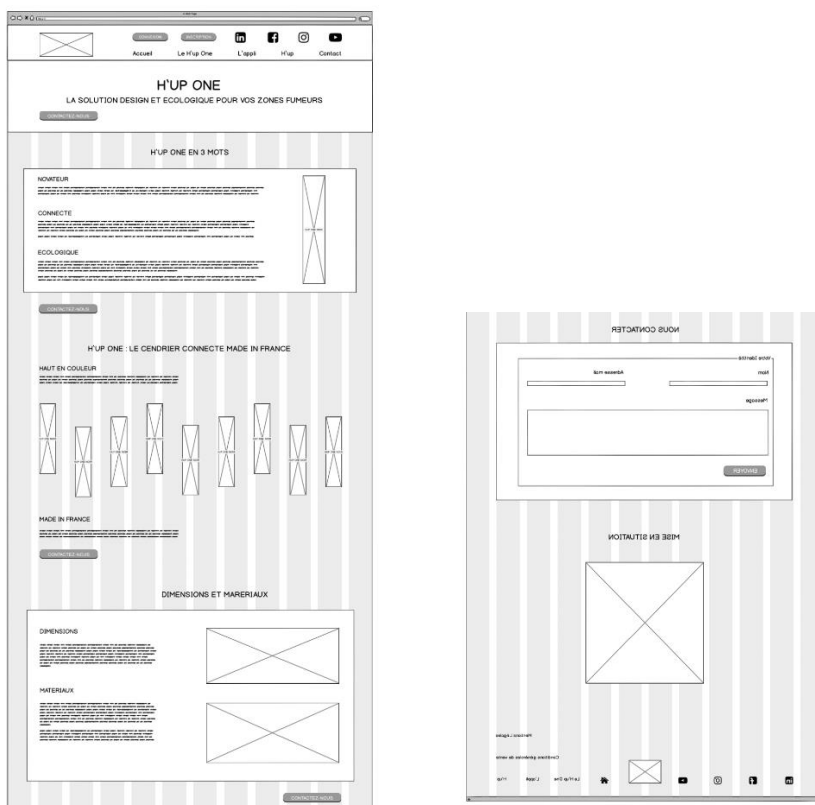
Dans un premier temps j'ai réalisé des wireframes afin d'élaborer une première esquisse de ce que serait le site d'H'up. J'ai pour cela utilisé le logiciel en ligne Balsamiq.

#### Version desktop :

##### Page d'accueil



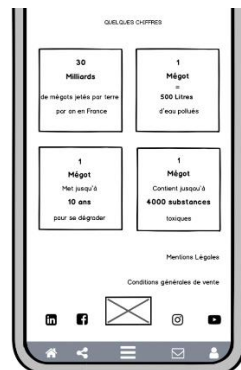
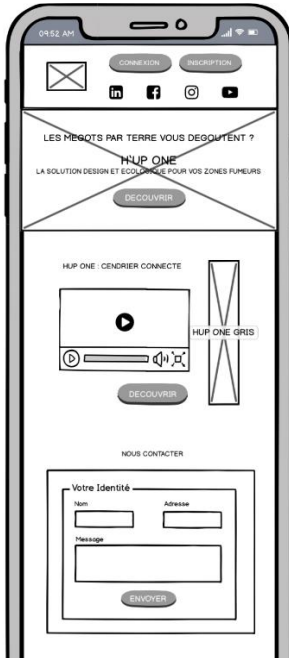
##### Page produit :



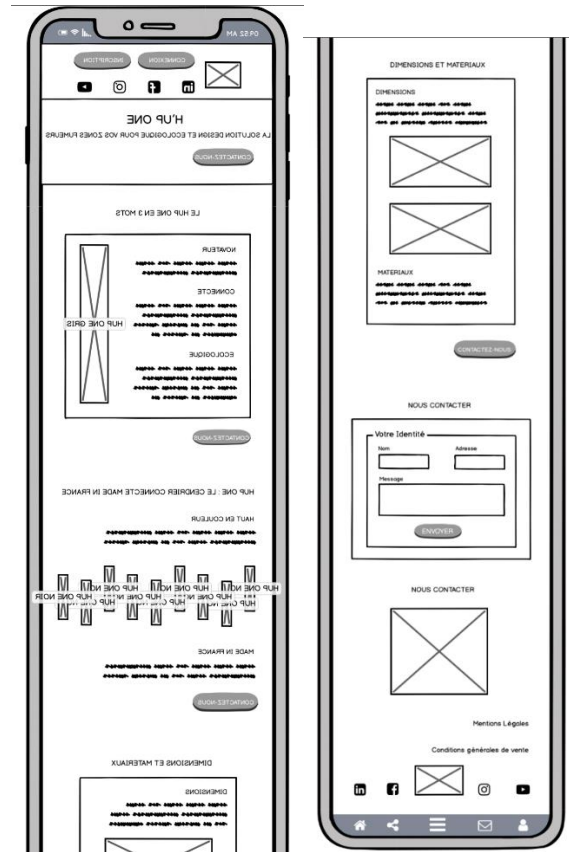


## Version mobile :

### 1) Page d'accueil :



### 2) Page produit :



## - Les maquettes :

J'ai réalisé les maquettes avec le logiciel Photoshop.

Les maquettes ont été pensées mobile first c'est pourquoi j'ai choisi un menu burger également pour la version desktop.



**H'UP**

LES MEGOTS PAR TERRE VOUS DEGOUTENT ?

**H'UP ONE**

LA SOLUTION DESIGN ET ECOLOGIQUE POUR VOS ZONES FUMEURS

[DECOUVRIR](#)

**H'UP ONE: CENDRIER CONNECTE**

Le Hup One est LA solution pour vos zones fumeurs !

Le cendrier connecté au design élégant, adapte les fumées à vos besoins et évite les odeurs.

Choisissez la couleur de votre cendrier (à l'achat) et collectez vos mégots.

[DECOUVRIR](#)

LE H'UP ONE VOUS INTERESSE ? CONTACTEZ-NOUS !

Nom

Prénom

Société

Adresse mail

Téléphone

Message

[ENVOYER](#)

**QUELQUES CHIFFRES**

30 Milliards de mégots par an en France

1 Mégot = 500 Litres d'eau polluée

1 Mégot = 10 ans pour se dégrader

1 Mégot = 4000 microplastiques

Mentions Légales  
FAQ  
Site fictif réalisé par Cello Lachetel

SAS H'UP  
20 rue XXX  
69003 Lyon  
06 13 21 30 15

**H'UP**

La page d'accueil desktop

**H'UP**

**H'UP ONE**

LA SOLUTION DESIGN ET ECOLOGIQUE POUR VOS ZONES FUMEURS

[CONTACTEZ-NOUS](#)

LE H'UP ONE EN 3 MOTS

**NOUVEAU**

Le cendrier connecté H'UP ONE, créé par Julie Arnaud, la fondatrice de la société H'UP est LA solution innovante et ludique pour toutes zones fumeurs.

**CONNECTE**

Le cendrier peut se connecter au réseau wifi de votre entreprise. Vous pouvez ainsi, grâce à l'application H'UP, mettre en place des jeux ou concours.

Les utilisateurs peuvent ainsi voir leurs résultats sur l'application mobile et interagir directement avec le H'UP One directement via la tablette intégrée.

**ECOLOGIQUE**

Le H'UP ONE permet de protéger l'environnement.

En effet, son interaction ludique avec les utilisateurs via l'écran tactile et l'application permet de réduire les déchets. Les fumées ne sont plus jetées dans la poubelle mais sont collectées et recyclées.

[CONTACTEZ-NOUS](#)

**H'UP ONE: LE CENDRIER CONNECTE MADE IN FRANCE**

**HAUT EN COULEUR**

Avec sa large gamme de couleurs métalliques, le H'UP ONE s'adapte parfaitement à l'environnement de votre entreprise.

**MADE IN FRANCE**

Le H'UP ONE est le premier cendrier connecté 100% FRANÇAIS !

Si son design épuré et élégant a été imaginé à Lyon, il est fabriqué en France.

[CONTACTEZ-NOUS](#)

**DIMENSIONS ET MATERIAUX**

**DIMENSIONS**

Cylindre : Diamètre 8,2 cm - Volume 75L  
Écran tactile LCD : 9,5 pouces - inclinable  
Le Panneau solaire : position à 1,80 m  
Cave : Diamètre 0,15 m - Hauteur 0,15 m - Volume 65 L

**MATERIAUX**

Cylindre et cave de l'unité : acier inoxydable et inextinguible  
Panneau solaire : résine haute résistance aux UV  
et aux intempéries  
Cave : aluminium anodisé  
Écran tactile : à faible consommation grâce à la technologie LCD  
son indicateur de niveau HDPI permet de visualiser les niveaux  
Panneau solaire : il fournit l'énergie nécessaire à l'écran tactile

[CONTACTEZ-NOUS](#)

Mentions Légales  
FAQ  
Site fictif réalisé par Cello Lachetel

SAS H'UP  
20 rue XXX  
69003 Lyon  
06 13 21 30 15

**H'UP**

La page produit desktop

**MADE IN FRANCE**

Le H'UP ONE est le premier cendrier connecté 100% FRANÇAIS !

Si son design épuré et élégant a été imaginé à Lyon, il est fabriqué en France.

[CONTACTEZ-NOUS](#)

**DIMENSIONS ET MATERIAUX**

**DIMENSIONS**

Cylindre : Diamètre 8,2 cm - Volume 75L  
Écran tactile LCD : 9,5 pouces - inclinable  
Le Panneau solaire : position à 1,80 m  
Cave : Diamètre 0,15 m - Hauteur 0,15 m - Volume 65 L

**MATERIAUX**

Cylindre et cave de l'unité : acier inoxydable et inextinguible  
Panneau solaire : résine haute résistance aux UV  
et aux intempéries  
Cave : aluminium anodisé  
Écran tactile : à faible consommation grâce à la technologie LCD  
son indicateur de niveau HDPI permet de visualiser les niveaux  
Panneau solaire : il fournit l'énergie nécessaire à l'écran tactile

[CONTACTEZ-NOUS](#)

**MISE EN SITUATION**

[CONTACTEZ-NOUS](#)

Mentions Légales  
FAQ  
Site fictif réalisé par Cello Lachetel

SAS H'UP  
20 rue XXX  
69003 Lyon  
06 13 21 30 15

**H'UP**

## IV- Intégration des maquettes :

### 1) Création du menu burger :

J'ai décidé de créer l'icône burger en CSS plutôt que d'intégrer une image car cela va me permettre de l'animer.

Je vais créer moi-même le 3 petites barres de l'icône du coup au clic je vais pouvoir faire une transformation dessus et donc transformer le burger en croix. En ajoutant une transition j'aurai une animation.

#### - Au niveau du HTML :

Je vais simplement créer une `<div>` et on va lui donner la class `.toggle_btn`

Cette div sera le conteneur de l'icône. Pour créer les 3 barres on va se servir d'une `<span>` vide.

La `<span>` sera la barre du milieu et je vais me servir des pseudo-éléments `::before` pour créer la barre du haut et `::after` pour celle du bas.

#### - Au niveau du CSS :

Comme ce site a été pensé mobile first, j'ai choisi de créer menu burger de 60px de haut par 60px de large car pour que la version mobile d'un site soit ergonomique, les icônes doivent avoir une hauteur minimum de 60px.

Ainsi, je stylise d'abord la `<div>` en lui donnant une largeur de 60px et une hauteur de 60px.

Ensuite je cible la `<span>` et lui donne une largeur de 100% (donc de 60px) ainsi qu'une hauteur de 6px. Or comme la `<span>` est un élément de type inline, elle ne comprend pas les hauteurs et les largeurs, il faut mettre un `display : bloc ;`

Afin de positionner la `<span>` au milieu de la `<div>` j'utilise une position `absolute` avec un top à 50%. Mais pour que ce soit parfaitement centré il faut également que je décale ma `<span>` de la moitié de sa hauteur vers le haut. Pour cela j'applique un `transform` avec un `translateY` de -50%.

Je mets en suite en place le before et l'after.

Je leur applique tout d'abord la propriété content que je laisse vide afin de générer leur contenu. Puis je leur attribut la même hauteur, largeur et la même couleur que la `<span>`.

Puis je les positionne grâce à la propriété CSS `position : absolute ;` à laquelle j'ajoute un `transform` avec un `translateY` de -27% au before pour le décaler vers le haut de 27% et un `transform` avec un `translateY` de 27% à l'after pour le décaler de 27% vers le bas.

Je leur rajoute la propriété `cursor : pointer ;` pour faire apparaitre la main au survol.

```
78
79 .toggle_btn span::before {
80     content: '';
81     height: 6px;
82     width: 100%;
83     background-color: #08CCA4;
84     position: absolute;
85     /* top: -26px; */
86     right: 0;
87     transform: translateY(-27px);
88     cursor: pointer;
89 }
90
91 .toggle_btn span::after {
92     content: '';
93     height: 6px;
94     width: 100%;
95     background-color: #08CCA4;
96     position: absolute;
97     /* top: 26px; */
98     right: 0;
99     transform: translateY(27px);
100     cursor: pointer;
101 }
```

```
# styleHeaderHup2020.css > ...
57  /* icon menu burger */
58  .toggle_btn {
59      width: 60px;
60      height: 60px;
61      position: relative;
62      cursor: pointer;
63      margin-top: 2em;
64      margin-bottom: 2em;
65  }
66
67  .toggle_btn span {
68      width: 100%;
69      height: 6px;
70      display: block;
71      background-color: #08CCA4;
72      position: absolute;
73      top: 50%;
74      right: 0;
75      transform: translateY(-50%);
76      transition: background-color 0.2s ease-in-out;
77  }
78
```

## - Au niveau du JavaScript :

Pour transformer mon burger en croix, il faut qu'au clic, je fasse disparaître la barre du milieu puis que la barre du haut et celle du bas effectuent une rotation.

Je commence par créer une constante que j'appelle *btn\_partage* qui va récupérer la <div> qui a la class *.bnt\_partage* en utilisant *document.querySelector(btn\_partage)* ;

```
1  // conserve la div qui contient la nav du menu
2  let btn = document.querySelector('.toggle_btn');
3  let nav = document.querySelector('.nav');
4
5  btn.onclick = function() {
6      nav.classList.toggle('nav_open');
7  }
8
9  // conserve le menu burger
10 const toggle_btn = document.querySelector('.toggle_btn');
11
12 toggle_btn.addEventListener('click', () => { //on écoute le cli
13     toggle_btn.classList.toggle('active') //on ajoute la class
14 });
```

J'utilise un toggle pour que lors du click on ajoute une class à la <div> et que, si cette class est présente, on modifie les propriétés CSS. Puis lorsque que l'on click une seconde fois, on fait disparaître cette class et on revient aux propriétés initiales. Ensuite je crée un toggle qui permet d'écouter le click et qui déclenche la fonction qui ajoute une class.

Ainsi, la fonction, lors du premier click ajoute la class *.active* à la <div> qui a la class *.toggle\_bnt* puis la retire lors du second click.

J'utilise ce même principe pour faire apparaître ou disparaître la <div> contenant la <nav> du menu et celle des réseaux sociaux.

## 2) Les média queries :

Afin de rendre le site d'H'up responsive, j'ai ajouté ds média queries.

J'ai par exemple, au niveau de la <section> qui contient l'image sous le header, changé le display de la <div> contenant l'image du cendrier afin de laisser plus de place aux titres.

J'ai pour cela mis un display : none ;

J'ai également modifié la taille et la position des titres sous le header.

```
# styleIndexHup2020.css > main
318
319 /***** MEDIA QUERIES *****/
320
321 @media screen and (max-width: 500px) {
322     body {
323         width: 100%;
324     }
325     #fond_hupNoir {
326         display: none;
327     }
328     h1,
329     h3 {
330         font-size: 22px;
331     }
332     h2 {
333         font-size: 44px;
334     }
335     h4 {
336         font-size: 25px;
337     }
338     h5 {
339         font-size: 20px;
340     }
341     #titresSousHeader {
342         width: 80%;
343         position: absolute;
344         top: 200px;
345         left: 50px;
346         padding-left: 10px;
347         padding-right: 10px;
348     }
349     #hup0ne {
350         display: none;
351     }
352     #contenuDecoupe1 {
353         width: 100%;
354     }
355     #concept {
356         width: 100%;
357         margin-top: 1em;
358     }
359     #Hup0neForm {
360         display: none;
361     }
```

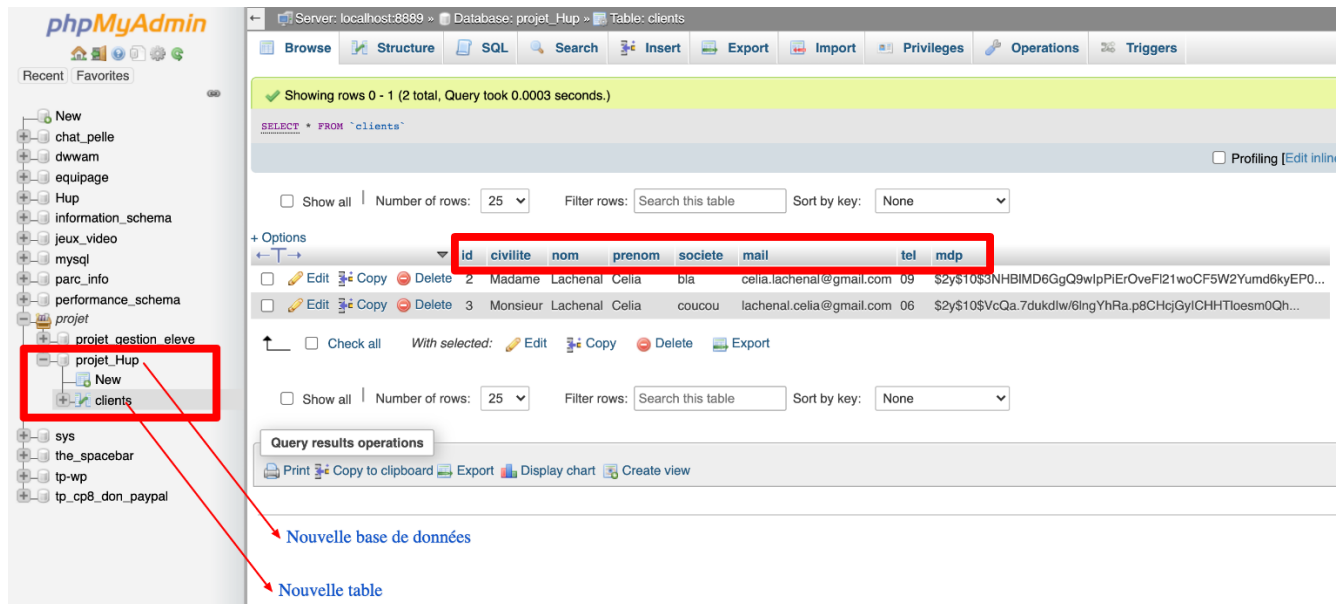
## 2<sup>ème</sup> PARTIE : DEVELOPPEMENT DE LA PARTIE BACK-END



## I- Création de la base de données :

L'entreprise H'up souhaite que ses visiteurs puissent s'inscrire via un premier formulaire puis se connecter à leur espace personnel qui leur permet de modifier à tout moment leurs informations personnelles.

J'ai tout d'abord créé, sur PhpMy Admin, une nouvelle base de données que j'ai intitulée projet\_Hup. Elle contient la table clients qui pour chaque client répertorie : l'id, leur civilité, leur nom et prénom, le nom de leur société, leur adresse mail, leur numéro de téléphone et leur mot de passe.



Puis, j'ai créé les pages espaceClient.php et pagePerso.php.

La page espaceClient.php contient un formulaire permettant aux usagers déjà membre de se connecter à leur page personnelle ainsi qu'un second pour pouvoir s'inscrire et devenir membre. C'est dans cette même page que j'effectue le traitement PHP lié à ses deux formulaires.

La page pagePerso.php contient le formulaire d'inscription avec les informations du client déjà remplies lui permettant de les mettre rapidement à jour.

## II- Traitement PHP des formulaires de connexion et d'inscription :

Il faut tout d'abord établir la connexion avec la base de données. Pour cela je crée la page connexion.php. J'utilise un Try/catch afin de pouvoir gérer les erreurs de connexion.

```
connexion.php > ...
1  <?php
2      try {
3          $base= new PDO("mysql:host=localhost;dbname=projet_Hup", "root", "root");
4          $base->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
5      }
6      catch(PDOException $e) {
7          echo "Erreur : " . $e->getMessage();
8      }
9  ?>
```

Ensuite, sur la page espaceClient.php j'ouvre une session avec session\_start();  
 Puis avec la condition if(isset(\$\_SESSION['idClient'])) je vérifie si la session que je viens d'ouvrir est rattachée à un id client ce qui signifierait que le visiteur est connecté à sa session et qu'il doit être redirigé vers sa page perso.  
 J'indique que le traitement PHP requière la page connexion.php.

```
espaceClient.php > ...
1  <?php
2  session_start();
3
4  if(isset($_SESSION['idClient'])){
5      header('Location:pagePerso.php');
6  }
7
8  require("connexion.php");
9  $message='';
10
11  $title = "Connexion";
12  include("header.php");
13
```

Comme je traite les deux formulaires sur la même page, je test le \$\_POST pour savoir si je reçois les informations liées au bouton du formulaire d'inscription ou celui du formulaire de connexion. Je peux facilement les distinguer car lors de la création des formulaires en HTML j'ai donné un name différent pour ces deux input type submit.

Dans le cas où le \$\_POST vient du bouton inscription\_envoyer :

- Je récupère les informations contenues dans les différents inputs et je les sécurise grâce à htmlspecialchars afin d'éviter toute faille XSS.

```
13
14  //inscription
15  if (isset($_POST["inscription_envoyer"])){
16      $civilite = htmlspecialchars($_POST["civilite"]);
17      $nom = htmlspecialchars($_POST["lastname"]);
18      $prenom = htmlspecialchars($_POST["firstname"]);
19      $societe = htmlspecialchars($_POST["company"]);
20      $mail = htmlspecialchars($_POST["mail"]);
21      $telephone = htmlspecialchars($_POST["phone"]);
22      $mdp = password_hash(htmlspecialchars($_POST["mdp"]), PASSWORD_DEFAULT);
23
```

- Je dois ensuite vérifier si le mail existe déjà en base de données grâce à un SELECT.
- Si le mail n'existe pas, je mets en place ma requête préparée pour insérer le nouveau client en base de données. Si le mail existe déjà j'en informe l'utilisateur par un message. J'utilise une requête préparée pour mon insert en base de données afin d'empêcher les injections SQL.

```
//je test si le mail est déjà en bdd
$sql_test= 'SELECT * FROM clients WHERE mail = :mail';
$stmt = $base -> prepare($sql_test);
$stmt -> execute(array(':mail'=> $mail));
$count = $stmt -> rowCount();

if ($count === 0){
    // si le mail n'est pas présent en bdd j'ajoute la personne
    //je mets en place ma requête préparée pour insérer un nouveau client

    $sql= 'INSERT INTO clients (civilite, nom, prenom, societe, mail, tel, mdp)
VALUES (:civilite, :nom, :prenom, :societe, :mail, :tel, :mdp)';
```



```

$insert= $base -> prepare($sql);
$insert->execute(array(':civilite' => $civilite, ':nom' => $nom, ':prenom'
=> $prenom, ':societe' => $societe, ':mail' => $mail, ':tel' => $telephone,
':mdp' => $mdp));

}
else{
    $message = 'une personne est déjà enregistrée sous ce mail';
}

```

Dans le cas où le \$\_POST vient du bouton connexion\_envoyer :

- Grâce à un SELECT je récupère les informations contenues dans la table clients dont le mail correspond au mail récupéré par le \$\_POST.
- Je vérifie que le mot de passe récupéré par \$\_POST correspond bien avec celui enregistré dans la base de données avec password\_verify. Si ce n'est pas le cas j'affiche le message « connexion échec ». Si c'est le bon mot de passe je récupère l'id du client et je le rattache à la session et je le redirige vers sa page personnelle.

espaceClient.php > ...

```

46 //connexion
47
48 if (isset($_POST["connexion_envoyer"])){
49     $mail = htmlspecialchars($_POST["mail"]);
50     $mdp = htmlspecialchars($_POST["mdp"]);
51     $sql = 'SELECT id,mail, mdp FROM clients WHERE mail = :mail';
52     $connect = $base -> prepare($sql);
53     $connect -> execute(array(':mail'=> $mail));
54     $resultat = $connect->fetch();
55
56     if(!$resultat OR !password_verify($mdp, $resultat['mdp'])){
57         $message = 'connexion echec';
58     }
59     else{
60         //je recupere l'id du client pour le rattacher à sa session
61         $_SESSION["idClient"] = $resultat['id'];
62         header('Location:pagePerso.php');
63         //$message = 'connexion ok';
64     }
65
66 }
67 ?>

```

### III- La documentation utilisée :

Afin de mieux comprendre la différence entre htmlentities() et htmlspecialchars() j'ai fait des recherches et j'ai trouvé cette documentation sur le site :

<https://www.geeksforgeeks.org/htmlentities-vs-htmlspecialchars-function-in-php/>

# htmlspecialchars() vs htmlspecialchars() Function in PHP

Last Updated: 21-08-2019

## htmlspecialchars() Function

The htmlspecialchars() function is an inbuilt function in PHP which is used to transform all characters which are applicable to HTML entities. This function converts all characters that are applicable to HTML entity.

### Syntax

```
string htmlspecialchars( $string, $flags, $encoding, $double_encode )
```

**Parameters:** This function accepts four parameters as mentioned above and described below:

- **\$string:** This parameter is used to hold the input string.
- **\$flags:** This parameter is used to hold the flags. It is combination of one or two flags, which tells how to handle quotes.
- **\$encoding:** It is an optional argument which specifies the encoding which is used when characters are converted. If encoding is not given then it is converted according to PHP default version.
- **\$double\_encode:** If double\_encode is turned off then PHP will not encode existing HTML entities. The default is to convert everything.

**Return Values:** This function returns the string which has been encoded.

## htmlspecialchars() Function

The htmlspecialchars() function is an inbuilt function in PHP which is used to convert all predefined characters to HTML entities.

### Syntax:

```
string htmlspecialchars( $string, $flags, $encoding, $double_encode )
```

- **\$string:** This parameter is used to hold the input string.
- **\$flags:** This parameter is used to hold the flags. It is combination of one or two flags, which tells how to handle quotes.
- **\$encoding:** It is an optional argument which specifies the encoding which is used when characters are converted. If encoding is not given then it is converted according to PHP default version.
- **\$double\_encode:** If double\_encode is turned off then PHP will not encode existing HTML entities. The default is to convert everything.

**Return Values:** This function returns the converted string. If there is invalid input string then empty string will returned.

**Difference between htmlentities() and htmlspecialchars() function:** The only difference between these function is:

- htmlspecialchars() function convert the special characters to HTML entities.
- htmlentities() function convert all applicable characters to HTML entities.

D'après cette documentation, les différences entre ces deux fonctions php sont que :

- htmlspecialchars() convertit que les caractères spéciaux ( ça ne comprend pas les accents) en entités html alors qu'htmlentities () convertit tous les caractères applicables c'est-à-dire les accents et tous les autres caractères spéciaux.
- La réponse renvoyée par ses deux fonctions est également différente : htmlspecialchars comme htmlentities renvoie la chaine de caractère convertie mais si la chaine qui a été entrée dans l'input n'est pas valide alors la fonction htmlspecialchars renvoie une chaine de caractère vide.

## **Conclusion :**

Cela m'a beaucoup plu de réaliser ce projet de bout en bout car cela m'a permit d'avoir une vision globale sur les différentes étapes de la création d'un site vitre. Cela m'a fait réaliser l'importance de de réfléchir au front et au back en parallèle et ce dès la phase de conception du site.