

CONCEPTION DE SITE WEB

CSS



HTML



SOMMAIRE

1. Expression des besoins	3
1.1 Contexte, domaine, processus métier	3
1.2 Demandeur, acteurs, utilisateurs	3
1.3 Étude de l'existant, diagnostic	3
1.4 Description de la demande, objectifs, bénéfices attendus	3
1.5 Spécifications fonctionnelles	3
1.6 Contraintes ou exigences	4
2. Conception, Spécifications Techniques	4
2.1 Description de la solution	4
2.2 Outils logiciels de la solution	5
2.3 Architecture matérielle et logicielle de la solution (schémas)	5
2.4 Besoins techniques, ressources (humaines, matérielles, logicielles et budgétaires, coûts),	6
2.5 IHM (interfaces homme-machine)	6
2.6 Conduite de projet : décomposition en tâches, durée	7
3. Développement	7
3.1 Réalisation des interfaces et programmes conformes aux spécifications fonctionnelles attendues	7
3.2 Dossier de programmation codes sources documentés et commentés	18
3.3 Difficultés rencontrées	19
4. Exploitation, Mise en production	20
4.1 Tests (unitaires, fonctionnels, intégration)	20
4.2 Installation, déploiement, fonctionnement éventuel en double avec l'ancienne procédure	20
4.3 Rédaction notice utilisateur	20
4.4 Formation des utilisateurs	20
5. Annexe	21
6. Bilan	22
6.1 Semaine 1	22
6.2 Semaine 2	22
6.3 Semaine 3	22
6.4 Semaine 4	22
6.5 Semaine 5	22

1. Expression des besoins

1.1 Contexte, domaine, processus métier

Dans le cadre de mon BTS SIO, j'ai réalisé mon stage à Label Informatique, une société spécialisée dans la récupération de données, la conception de site web et la réparation de matériels informatiques. Mon projet principal consiste à créer un site internet pour un camping fictif, basé sur une version préexistante que j'avais développée lors d'ateliers professionnels en cours.

1.2 Demandeur, acteurs, utilisateurs

Le demandeur principal de ce projet est la clientèle. Les acteurs impliqués dans ce projet incluent :

- Moi-même, en tant que développeur stagiaire.
- Mon maître de stage, qui supervise et valide mon travail.
- Les futurs utilisateurs : potentiels clients du camping fictif, qui utiliseront le site pour obtenir des informations et effectuer des réservations.

1.3 Étude de l'existant, diagnostic

Avant mon arrivée, une version initiale du site web existait, développée dans le cadre de mes cours. Cependant, cette version nécessitait des améliorations significatives en termes de design, de fonctionnalités et d'optimisation pour répondre aux standards professionnels de mon maître de stage.

1.4 Description de la demande, objectifs, bénéfices attendus

L'objectif principal de ce projet est d'améliorer le site vitrine d'un camping fictif. Cela inclut l'amélioration du design et de la structure, l'ajout de nouvelles fonctionnalités telles que la réservation en ligne, la connexion à un compte client, et la présentation détaillée de l'entreprise fictive de ce camping. Ceci va donc améliorer l'expérience utilisateur grâce à une interface attrayante et une fonctionnalité robuste qui peut servir de modèle pour mes futurs projets.

1.5 Spécifications fonctionnelles

Au cours de mon stage, je vais ajouter de nouvelles fonctionnalités et améliorer le graphisme de mon site vitrine :

- Barre de navigation améliorée : Modification des liens affichés et ajout d'une bannière pour améliorer le design. Permet une navigation claire et intuitive pour les utilisateurs.

- Ajout d'un diaporama sur la page d'accueil : Affichage d'images du camping avec des flèches de navigation pour changer d'image pour améliorer la présentation visuelle et attirer l'attention des visiteurs.
- Création d'une page de contact : Affichage des informations de contact.
- Barre de contact : Ajout d'une barre de contact pour un accès rapide au formulaire de demande de renseignement et aux conditions générales.
- Ajout d'un formulaire de demande de renseignement : Formulaire à remplir avec les coordonnées de l'utilisateur dans le but d'envoyer une demande d'information au camping.
- Système de compte client : Utilisation de variables de session pour gérer les comptes clients, avec la possibilité de se connecter, de se déconnecter et de créer un compte.
- Amélioration de la page de réservation : Ajout de fonctionnalités pour choisir la date de début et de fin du séjour, ainsi que le type de mobil-home souhaité. Récapitulatif de la réservation avant le paiement.
- **Promotions** : Ajout de bannières publicitaires sur la page d'accueil pour promouvoir les offres spéciales et les formules, incitant les visiteurs à réserver.

1.6 Contraintes ou exigences

Les contraintes matérielles comprennent l'utilisation de mon propre ordinateur et de l'ordinateur de l'entreprise. Sur le plan technique, je dois maîtriser les langages et outils de développement web requis pour le projet (HTML, CSS, PHP, JS). Les délais du stage imposent une progression significative du projet chaque semaine. Le budget est limité, donc je dois utiliser des outils et technologies gratuits (WampServer64 et Sublime Text) ou déjà disponibles dans l'entreprise comme PhotoFiltre.

2. Conception, Spécifications Techniques

2.1 Description de la solution

La solution développée offre une plateforme conviviale et fonctionnelle pour les utilisateurs du camping fictif. Elle intègre plusieurs fonctionnalités essentielles pour le bon fonctionnement et l'attractivité du site web :

- Amélioration de la structure et du design du site en utilisant HTML, CSS, PHP et JavaScript : Assure une interface utilisateur attrayante et une navigation fluide.
- SEO (référencement naturel) : Utilisation des balises ALT pour les images afin d'améliorer l'accessibilité et l'indexation par les moteurs de recherche. Choix de titres de page descriptifs pour indiquer clairement le contenu de chaque page. Intégration de mots-clés pertinents dans le contenu pour améliorer la visibilité du site dans les résultats de recherche.
- **Intégration de Google Analytics** : Ajout d'un script Google Analytics pour suivre les visiteurs du site et obtenir des données sur leur comportement afin d'améliorer le référencement et l'expérience utilisateur. Ci-dessous, voici le script Google Analytics

ajouté sur toutes mes pages afin d'améliorer le référencement et de permettre un suivi des actions des utilisateurs sur mon site :

```
<!-- Google tag (gtag.js) -->
<script async src="https://www.googletagmanager.com/gtag/js?id=G-DW3N1F4ZNN"></script>
<script>
  window.dataLayer = window.dataLayer || [];
  function gtag(){dataLayer.push(arguments);}
  gtag('js', new Date());

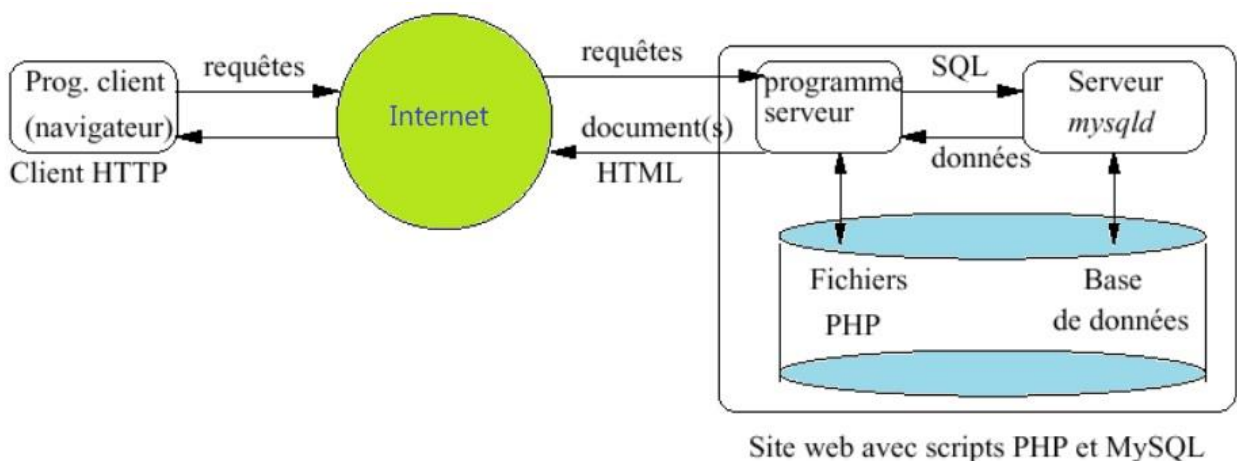
  gtag('config', 'G-DW3N1F4ZNN');
</script>
```

2.2 Outils logiciels de la solution

Pour ce projet, j'ai donc utilisé :

- *HTML*, un langage de balisage qui permet de structurer les éléments visuels qui composent une page web.
- *CSS*, qui permet de faire une mise en forme des pages web (HTML ou XML).
- *PHP*, un langage de script côté serveur, spécialement conçu pour le développement d'applications web. Il peut être facilement intégré au HTML.
- *JavaScript*, qui est un langage de programmation permettant de créer et de contrôler des éléments dynamiques, pour rendre une page web interactive.
- *SEO* (référencement naturel) : utilisation des balises ALT pour les images, des titres de page descriptifs, et des mots-clés pertinents pour améliorer la visibilité du site dans les moteurs de recherche.
- *MySQL* pour la gestion de base de données SQL.
- Utilisation de *PhotoFiltre* pour la création et la modification d'éléments graphiques.

2.3 Architecture matérielle et logicielle de la solution (schémas)



2.4 Besoins techniques, ressources (humaines, matérielles, logicielles et budgétaires, coûts), ...

Sur le plan technique, j'ai besoin d'un ordinateur et de logiciels pour coder (Sublime Text et WampServer64) ainsi que d'un logiciel de gestion de base de données (MySQL). Les ressources humaines comprennent moi-même, développeur stagiaire, et l'assistance de mon maître de stage. Il n'y a pas de coût budgétaire.

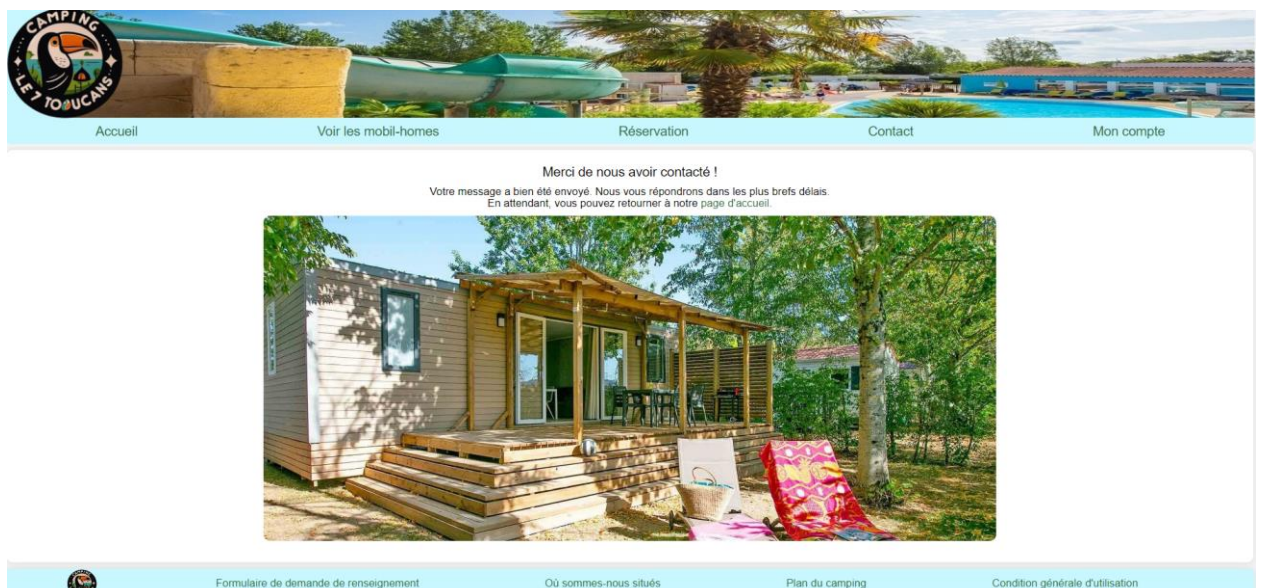
2.5 IHM (interfaces homme-machine)

IHM fait référence à un tableau de bord qui permet à un utilisateur de communiquer avec une machine, un programme informatique ou un système. Les trois principaux IHM sont :

- le toucher (commande par boutons, claviers, souris)
- La vision (contrôle sur écran, palette de couleur, images)
- l'ouïe (mon site ne possède pas de son)

L'IHM (Interface Homme-Machine) de mon site web englobe tous les éléments visuels et interactifs que les utilisateurs voient et avec lesquels ils interagissent lors de leur navigation. C'est un aspect crucial car elle influence directement l'expérience de utilisateur. L'IHM de mon site doit être intuitive, conviviale et esthétiquement plaisante. Elle comprend plusieurs éléments essentiels :

1. Navigation : Les menus de ma barre de navigation, les liens et les boutons qui permettent aux utilisateurs de se déplacer entre les différentes pages du site.
2. Disposition et design : La manière dont les éléments visuels sont disposés sur chaque page pour assurer une présentation claire et attrayante.
3. Interactivité : Les éléments interactifs comme les formulaires de demande d'informations, les boutons, le slider d'images de la page d'accueil qui permettent aux utilisateurs d'effectuer des actions spécifiques sur le site.
4. Feedback utilisateur : Les réponses visuelles ou auditives du site qui indiquent à l'utilisateur que son action a été effectuée avec succès (comme la page après avoir envoyé le formulaire de renseignement ci-dessous) ou qui l'orientent en cas d'erreur.



En résumé, l'IHM de mon site web vise à optimiser l'expérience de utilisateur, en répondant aux besoins et aux attentes des visiteurs tout en soutenant les objectifs du site, que ce soit pour fournir des informations, promouvoir des produits ou proposer des services.

2.6 Conduite de projet : décomposition en tâches, durée

La conduite du projet implique une planification rigoureuse des différentes étapes de développement en collaboration avec mon maître de stage. Voici comment le projet a été structuré :

Analyse des besoins :

- Révision des exigences fonctionnelles et techniques du site web du camping avec mon maître de stage.
- Établissement des objectifs principaux du projet en termes de fonctionnalités et d'expérience utilisateur.

Conception de l'architecture et maquettage :

- Définition de l'architecture logicielle et des interactions entre les différents composants du site web.

Développement front-end et back-end:

- Implémentation des fonctionnalités (décrites dans la partie 1.5).
- Utilisation de PHP, JavaScript, HTML, CSS pour le développement des interfaces utilisateur et des fonctionnalités back-end (comme l'authentification des utilisateurs et l'enregistrement des réservations).

Tests unitaires et fonctionnels :

- Validation de chaque composant du site pour assurer son bon fonctionnement et sa conformité aux spécifications.
- Correction des bugs et optimisation des performances en fonction des retours obtenus.

Déploiement :

- Déploiement sur l'hébergeur 000WebHost afin de garantir que le site fonctionne totalement même après publication.

3. Développement

3.1 Réalisation des interfaces et programmes conformes aux spécifications fonctionnelles attendues

Conformément aux spécifications fonctionnelles, j'ai finalisé les dernières fonctionnalités du site web du camping fictif, en veillant à améliorer l'expérience de utilisateur et à optimiser les processus de réservation et de gestion de compte.

1- Intégration de la réservation de mobil-home : Les utilisateurs peuvent sélectionner le type de mobil-home souhaité et les dates de leur séjour. Après validation, une confirmation est envoyée par e-mail aux utilisateurs connectés, tandis que les utilisateurs non connectés sont redirigés vers la page de connexion.

Voici la première page pour procéder à une réservation que l'utilisateur devra remplir avec sa date et son type de mobil-home :

Ci-dessous le code de cette page :

```
<div class="container">
  <div class="left">
    <form method="post" action="resultat-reservation.php">
      <h2 style="font-weight: normal; font-size: 1.22em;">Sélectionnez votre date de séjour</h2>
      <label>Choisissez la date de début :</label>
      <input type="date" name="date_debut" id="date_debut" onchange="updateDateFin()" required><br><br>
      <label>Choisissez la date de fin :</label>
      <input type="date" name="date_fin" id="date_fin" readonly><br><br>
      <label>Choisissez un type de mobil-home :</label>
      <select name="cbotype" required>
        <?php
          $bdd = new PDO("mysql:host=localhost;dbname=campingredolfi-dilena;charset=utf8", "root", "");
          $req = $bdd->prepare("SELECT idtyp, libtyp FROM typemobil");
          $req->execute();
          $leslignes = $req->fetchAll();
          foreach ($leslignes as $uneligne) {
            echo("<option value='{$uneligne['idtyp']}'>{$uneligne['libtyp']}</option>");
          }
          $req->closeCursor();
        >
      </select><br><br>
      <input type="submit" class="btnsubmit" name="valider" value="Réserver">
    </form>
  </div>
  <div class="right">
    
    
    
    
  </div>
</div>

<div id="monModele" class="modele">
  <span class="modele-close" onclick="fermerModele()">&times;</span>
  <img class="modele-content" id="modeleImage">
</div>
```


L'utilisateur est ensuite dirigé vers la page « resultat-reservation.php », où il peut consulter un récapitulatif de sa réservation ainsi que les mobil-homes disponibles pour son séjour et sélectionner celui qu'il préfère.

Ci-dessous, le code HTML/PHP de cette page :

```
<div class="container">
<?php
if (isset($_POST['cbotype']) && isset($_POST['date_debut']) && !empty($_POST['date_debut'])) {
    $bdd = new PDO("mysql:host=localhost;dbname=campingredolfi-dilena;charset=utf8", "root", "");

    $id_type_mobilhome = $_POST['cbotype'];
    $date_debut = $_POST['date_debut'];
    $date_fin = date('Y-m-d', strtotime($date_debut . ' + 7 days'));

    $req_type_mobilhome = $bdd->prepare("SELECT libtyp FROM typemobil WHERE idtyp = :idtyp");
    $req_type_mobilhome->bindParam(':idtyp', $id_type_mobilhome, PDO::PARAM_INT);
    $req_type_mobilhome->execute();
    $libelle_type_mobilhome = $req_type_mobilhome->fetchColumn();

    $sql = "SELECT * FROM mobilhome WHERE idtyp = :idtyp AND idmob NOT IN (
        SELECT idmob FROM reservation
        WHERE datedebut = :date_debut)";
    $req = $bdd->prepare($sql);
    $req->bindParam(':idtyp', $id_type_mobilhome, PDO::PARAM_INT);
    $req->bindParam(':date_debut', $date_debut, PDO::PARAM_STR);
    $req->execute();
    $mobilhomes_disponibles = $req->fetchAll();

    if ($mobilhomes_disponibles) {
        echo "<h2 style='margin-bottom: 16px; font-weight: normal; font-size: 1.22em;'>Récapitulatif de la réservation :
        </h2>";
        echo "Vous avez choisi : <br><strong>$libelle_type_mobilhome </strong><br><br>Pour la période du : <br><strong>$
        date_debut au $date_fin</strong>.<br><br><br>";
        echo "<form method='post' action='reservation-connexion.php'>";
        echo "<ul>";
        foreach ($mobilhomes_disponibles as $mobilhome) {
            echo "<li class='liste'><input type='radio' name='mobilhome' value='{ $mobilhome['idmob'] }'> { $mobilhome['nom'] }
            <br>";

            $req_photos = $bdd->prepare("SELECT nomfichier FROM photo WHERE idtyp = :idtyp");
            $req_photos->bindParam(':idtyp', $id_type_mobilhome, PDO::PARAM_INT);
            $req_photos->execute();

            $photos = $req_photos->fetchAll();
            foreach ($photos as $photo) {
                $nomfichier = $photo['nomfichier'];
                $imagePath = "photo/{ $id_type_mobilhome }/res/$nomfichier";

                if (file_exists($imagePath)) {
                    echo "<br><img src='$imagePath' alt='Mobil-home Camping Les 7 Toucans Mâcon' class='img'
                    onclick='ouvrirModele(this.src)'><br><br>";
                }
            }

            echo "</li>";
        }
        echo "</ul>";
        echo "<input type='hidden' name='date_debut' value='$date_debut'>";
        echo "<input type='submit' name='valider_reservation' value='Valider la réservation' class='btnsubmit'>";
        echo "</form>";
    } else {
        echo "Aucun mobil-home disponible pour la date $date_debut, veuillez choisir une autre date.";
    }

    $bdd = null;
} else {
    echo "Les données du formulaire de réservation n'ont pas été correctement soumises.";
}
?>
</div>
```

Voici le rendu de cette page:

Récapitulatif de la réservation :

Vous avez choisi :
5 Mobil-Home Confort 4 personnes

Pour la période du :
2024-07-30 au 2024-08-06.

☐ De Vinci

☐ Rembrandt

☐ Duchamp

☐ Bacon

☐ Cassat



Valider la réservation

Par la suite, si l'utilisateur n'est pas connecté, il sera envoyé sur la page de connexion.

The screenshot shows the website's navigation bar with links: Accueil, Voir les mobil-homes, Réservation, Contact, and Mon compte. Below the navigation bar, there are two main sections for user authentication. The left section, titled 'Vous avez déjà un compte ?', contains fields for Email and Mot de passe, with a 'Se connecter' button. The right section, titled 'Vous n'avez pas de compte ?', contains fields for Informations (Nom, Adresse, Code postal, Ville, Téléphone) and Identifiants (Email, Mot de passe), with a 'Valider' button. The background of the forms features a scenic view of a lake and a forest. At the bottom of the page, there is a footer with links: Formulaire de demande de renseignement, Où sommes-nous situés, Plan du camping, and Condition générale d'utilisation.

Après connexion, ou s'il était déjà connecté, il directement envoyé vers la page ci-dessous.

Sous réserve de disponibilité et du paiement, nous vous adresserons un mail de confirmation.

Retour à la page d'accueil



2- Page de connexion, création de compte et gestion du compte utilisateur: J'ai revu et combiné la page de connexion et de création de compte pour une meilleure fluidité et une expérience utilisateur améliorée. Cela permet aux nouveaux utilisateurs de s'inscrire facilement et aux utilisateurs existants de se connecter sans effort.

Voici le code de ma page de connexion :

```
<body>
<?php
session_start();
include('navbar.php');

// Traitement de la déconnexion
if (isset($_POST['deconnexion'])) {
    unset($_SESSION['id_client']);
    session_destroy(); // Détruire la session pour plus de sécurité
}

// Connexion à la base de données
$dbdd = new PDO("mysql:host=localhost;dbname=mondbdd");

// Vérification de la connexion
if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST["email"]) && isset($_POST["motdepasse"])) {
    $email = $_POST["email"];
    $motdepasse = $_POST["motdepasse"];

    // Requête pour vérifier les informations de connexion
    $req = $dbdd->prepare("SELECT idcli FROM client WHERE mail = :email AND motpasse = :motdepasse");
    $req->bindParam(':email', $email, PDO::PARAM_STR);
    $req->bindParam(':motdepasse', $motdepasse, PDO::PARAM_STR);
    $req->execute();

    // Vérifier s'il y a des résultats
    if ($req->rowCount() > 0) {
        $client = $req->fetch(PDO::FETCH_ASSOC);
        $_SESSION['id_client'] = $client['idcli'];
        echo "<script>
        window.location.href = 'mobilhomes-camping-7-toucans.php';
        </script>";
    } else {
        echo "<p>Identifiants incorrects. Veuillez réessayer.</p>";
    }
}

// Fermeture de la connexion à la base de données
$req->closeCursor();
}
```

Comme ce qui est commenté, « if (isset(\$_POST['deconnexion'])) » permet de se déconnecter une fois que le bouton avec le nom « deconnexion » est cliqué (il sera affiché par la suite sur le profil utilisateur après leur connexion).

De plus, « if (\$_SERVER["REQUEST_METHOD"] == "POST" && isset(\$_POST["email"]) && isset(\$_POST["motdepasse"])) » permet de vérifier si l'Email et le mot de passe que l'utilisateur a écrit sont égaux à ceux de la base de donnée lors de l'inscription de l'utilisateur, sinon un message d'erreur est affiché.

```
// Création de compte
if (isset($_POST['creation'])) {
    if (!empty($_POST['nom']) && !empty($_POST['adresse']) && !empty($_POST['cp']) && !empty($_POST['ville']) && !empty($_POST['telephone']) && !empty($_POST['email']) && !empty($_POST['mot_de_passe'])) {
        // Préparation de la requête d'insertion
        $req = $dbdd->prepare("INSERT INTO client (nomcli, adresse, cp, ville, telephone, mail, motpasse) VALUES (:nomcli, :adresse, :cp, :ville, :telephone, :mail, :motpasse)");

        // Liaison des paramètres
        $req->bindParam(':nomcli', $_POST['nom'], PDO::PARAM_STR);
        $req->bindParam(':adresse', $_POST['adresse'], PDO::PARAM_STR);
        $req->bindParam(':cp', $_POST['cp'], PDO::PARAM_STR);
        $req->bindParam(':ville', $_POST['ville'], PDO::PARAM_STR);
        $req->bindParam(':telephone', $_POST['telephone'], PDO::PARAM_STR);
        $req->bindParam(':mail', $_POST['email'], PDO::PARAM_STR);
        $req->bindParam(':motpasse', $_POST['mot_de_passe'], PDO::PARAM_STR);

        // Exécution de la requête
        $reponse = $req->execute();

        // Vérification de l'insertion
        if ($reponse) {
            $id_client = $dbdd->lastInsertId();
            $_SESSION['id_client'] = $id_client;
            echo "<script>
            window.location.href = 'mobilhomes-camping-7-toucans.php';
            </script>";
        } else {
            echo "<p>Erreur lors de l'enregistrement.</p>";
        }
    }

    // Fermeture de la requête
    $req->closeCursor();
} else {
    echo "<p>Veuillez remplir tous les champs du formulaire.</p>";
}
}
```

Ensuite, lorsque le bouton « creation » est cliqué, mon programme vérifie que l'utilisateur a bien inscrit toutes les données requises pour la création de compte. Si l'utilisateur n'a pas tout rempli, un message d'erreur sera affiché. Après vérification que toutes les données du formulaire sont remplies, la requête permettant d'enregistrer le nouveau compte client dans la base de donnée sera vérifiée (if (\$reponse)) Si la requête fonctionne, le compte est créé et l'utilisateur est redirigé vers la page où l'on peut visualiser tous les mobil-homes. Si non, un message d'erreur survient.

```
// Affichage du compte client
if (isset($_SESSION['id_client'])) {
    $id_client = $_SESSION['id_client'];

    // Récupérer les informations du client depuis la base de données
    $req = $bdd->prepare("SELECT nomcli, adresse, cp, ville, telephone, mail FROM client WHERE idcli = :idcli");
    $req->bindParam(':idcli', $id_client, PDO::PARAM_INT);
    $req->execute();
    $client_info = $req->fetch(PDO::FETCH_ASSOC);

    if ($client_info) {
        echo "<div class='container'>";
        echo "<div class='container-info'>";
        echo "<h2>Bienvenue, {$client_info['nomcli']} !</h2>";
        echo "<p style='font-size:1.1em'>Vos informations personnelles : </p>";
        echo "<p>Adresse : {$client_info['adresse']}, {$client_info['cp']} {$client_info['ville']}</p>";
        echo "<p>Téléphone : {$client_info['telephone']}</p>";
        echo "<p>Email : {$client_info['mail']}</p>";
        echo "<form method='post'>";
        echo "<input type='submit' name='deconnexion' value='Se déconnecter' class='btn'>";
        echo "</form>";
        echo "</div>";
        echo "</div>";
    }

    // Fermeture de la connexion à la base de données
    $req->closeCursor();
} else {
    // Si l'utilisateur n'est pas connecté, afficher les formulaires de connexion et de création de compte
    echo "<div class='container'>";
    echo "<div class='container-flex'>";

    echo "<div class='container-connexion'>";
    echo "<h2>Vous avez déjà un compte ?</h2><br>";
    echo "<form method='post' action=''>";
    echo "<label for='email'>Email :</label><br>";
    echo "<input class='inputtxt' type='email' id='email' name='email' required><br><br>";
    echo "<label for='motdepasse'>Mot de passe :</label><br>";
    echo "<input class='inputtxt' type='password' id='motdepasse' name='motdepasse' required><br><br>";
    echo "<input type='submit' value='Se connecter' class='btn'>";
    echo "</form>";
    echo "</div>";

    echo "<div class='container-creation'>";
    echo "<h2>Vous n'avez pas de compte ?</h2><br>";
    echo "<form method='post' action=''>";
    echo "<h3>Informations :</h3>";
    echo "<label for='nom'>Nom :</label><br>";
    echo "<input class='inputtxt' type='text' id='nom' name='nom' required><br><br>";
    echo "<label for='adresse'>Adresse :</label><br>";
    echo "<input class='inputtxt' type='text' id='adresse' name='adresse' required><br><br>";
    echo "<label for='cp'>Code postal :</label><br>";
    echo "<input class='inputtxt' type='text' id='cp' name='cp' required><br><br>";
    echo "<label for='ville'>Ville :</label><br>";
    echo "<input class='inputtxt' type='text' id='ville' name='ville' required><br><br>";
    echo "<label for='telephone'>Téléphone :</label><br>";
    echo "<input class='inputtxt' type='text' id='telephone' name='telephone' placeholder='00 00 00 00' required><br><br>";
    echo "<h3>Identifiants :</h3>";
    echo "<label for='email'>Email :</label><br>";
    echo "<input class='inputtxt' type='email' id='email' name='email' required><br><br>";
    echo "<label for='mot_de_passe'>Mot de passe :</label><br>";
    echo "<input class='inputtxt' type='password' id='mot_de_passe' name='mot_de_passe' required><br><br>";
    echo "<input type='submit' name='creation' value='Valider' class='btn'>";
    echo "</form>";
    echo "</div>";
    echo "</div>";
}
```

Après leur connexion, les utilisateurs peuvent accéder à leur profil personnel où ils peuvent voir leurs informations de compte et se déconnecter.

Mais si l'utilisateur n'est pas connecté, il sera envoyé vers les formulaires de connexion et de création de compte qu'il devra remplir afin que les requête de connexion ou de création de compte précédentes fonctionnent (il devra remplir uniquement le formulaire de connexion s'il est déjà inscrit et il devra créer son compte s'il ne l'a jamais fait).

Voici le rendu de la page de connexion lorsque l'utilisateur n'est pas connecté :

Enfin, voici le rendu lorsqu'il est déjà connecté :

3- Barre de navigation : J'ai amélioré le design et la structure et les liens de la barre de navigation pour une navigation intuitive et efficace à travers le site. Voici le code HTML de ma barre de navigation :

```
<nav>
  <ul>
    <li><a href="index.php">
      <div></div>
    </a></li>
  </ul>
  <ul>
    <li><a href="index.php" class="nav-link">Accueil</a></li>
    <li><a href="projetles7toucans.php" class="nav-link">Voir les mobil-homes</a></li>
    <li><a href="reservation.php" class="nav-link">Réservation</a></li>
    <li><a href="contact.php" class="nav-link">Contact</a></li>
    <li><a href="page_de_connexion.php" class="nav-link">Mon compte</a></li>
  </ul>
</nav>
```

Ci-dessous, le rendu de ma barre de navigation :



4- Page d'accueil : J'ai intégré un diaporama de plusieurs images, des publicités et des éléments visuels pour présenter le camping de manière attrayante dès l'arrivée sur le site.

Ci-dessous, voici le code en HTML de la page d'accueil :

```
<body>
  <!-- Navbar -->
  <?php include('navbar.php');?>

  <!-- Structure principale avec trois conteneurs alignés -->
  <div class="main-container">
    <!-- Conteneur gauche -->
    <div class="left-container ad-container">
      <!-- Contenu du conteneur gauche -->
      
      
      <p>Offre spéciale : profitez de 20% de réduction sur le mobilhome SweetFlower sur Pilotis pour 5
        personnes pour seulement 375,20 € au lieu de 469 €<br><br>
      <a href="reservation.php" class="promo">Réservez maintenant</a> pour la période du 15 au 22 août.</p>
    </div>

    <!-- Contenu principal -->
    <div class="containerautr">
      <p class="slogan">Détendez-vous et créez des souvenirs inoubliables au cœur du camping Les 7 Toucans
      </p>

      <p class="titre">Le camping Les 7 Toucans</p>

      <!-- Diaporama -->
      <div class="slideshow-container">

        <div class="mySlides">
          
        </div>

        <div class="mySlides">
          
        </div>

        <div class="mySlides">
          
        </div>

        <div class="mySlides">
          
        </div>

        <a class="prev" onclick="plusSlides(-1)">&#10094;</a>
        <a class="next" onclick="plusSlides(1)">&#10095;</a>
      </div>
    </div>
  </div>
```



```

<!-- Conteneur droit -->
<div class="right-container event-container">
  <!-- Contenu du conteneur droit -->
  
  
  <p>Profitez d'un incroyable séjour avec notre<span class="pension">formule pension complète</span>
  réservée aux familles de plus de 3 personnes.<br><br>
  Vos 3 repas quotidiens vous sont fournis au restaurant du camping tous les jours.</p>
</div>
</div>

<!-- modele pour afficher l'image en grand -->
<div id="monModele" class="modele">
  <span class="close" onclick="fermerModele()">&times;</span>
  <img class="modele-content" id="modeleImage">
</div>

<!-- Script JavaScript -->
<script>
  var slideIndex = 1;
  showSlides(slideIndex);

  function ouvrirModele(imgPath) {
    var modele = document.getElementById("monModele");
    var modeleImg = document.getElementById("modeleImage");
    modele.style.display = "block";
    modeleImg.src = imgPath;
  }

  function fermerModele() {
    var modele = document.getElementById("myModele");
    modele.style.display = "none";
  }

  function plusSlides(n) {
    showSlides(slideIndex += n);
  }

  function showSlides(n) {
    var i;
    var slides = document.getElementsByClassName("mySlides");
    if (n > slides.length) { slideIndex = 1; }
    if (n < 1) { slideIndex = slides.length; }
    for (i = 0; i < slides.length; i++) {
      slides[i].style.display = "none";
    }
    slides[slideIndex - 1].style.display = "block";
  }
</script>

<!-- Inclure le footer -->
<?php include('footer.php'); ?>
</body>

```

Ci-dessus, nous pouvons voir que afin d'afficher des promotions sur les cotés d'un diaporama, j'ai créer 3 conteneurs à l'intérieur d'un plus grand conteneur (main-container). Le premier conteneur est celui du milieu (containerautr), contenant le diaporama des images du camping et le slogan du camping. Le deuxième et le troisième (left-container et right-container) sont construit de la même façon et contiennent les promotions du camping.

À la fin du code HTML, j'ai intégré un script JavaScript qui permet de faire fonctionner le diaporama d'image de la page d'accueil et qui permet d'affiché les images en grand lorsque les utilisateurs cliquent dessus.

Avec le HTML, j'ai rajouté du CSS afin d'améliorer le diaporama de la page (capture d'écran de gauche) et j'ai aussi rajouté la possibilité de voir une image en plus grand lorsque l'on clique dessus (capture d'écran de droite) :

```
.slideshow-container {
  max-width: 73.02%;
  position: relative;
  margin: auto;
}

.mySlides img {
  width: 100%;
  height: auto;
  cursor: pointer;
  border-radius: 10px;
}

.prev, .next {
  position: absolute;
  top: 50%;
  transform: translateY(-50%);
  padding: 10px;
  font-size: 20px;
  background-color: rgba(0, 0, 0, 0.5);
  color: white;
  border: none;
  cursor: pointer;
}

.prev:hover, .next:hover {
  color: #bbb;
  background-color: rgba(0, 0, 0, 0.7);
}

.prev {
  left: 0;
}

.next {
  right: 0;
}
```

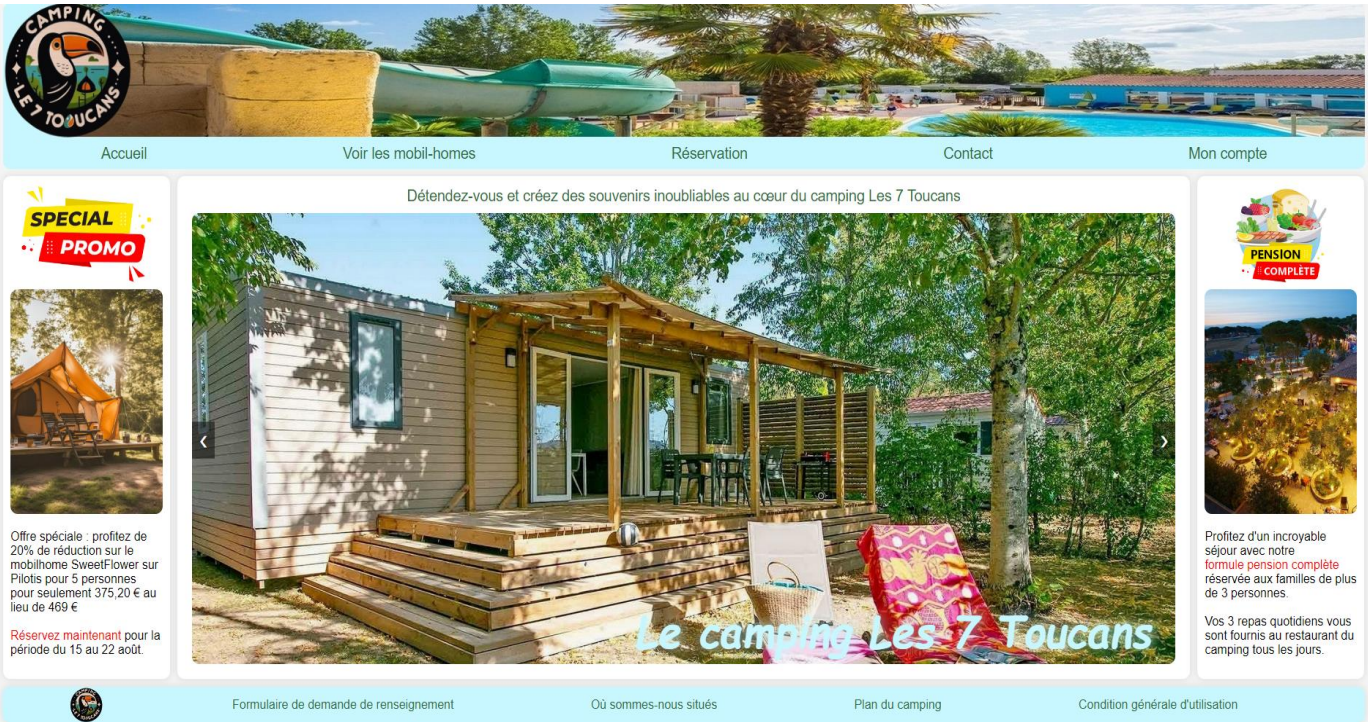
```
.modal {
  display: none;
  position: fixed;
  z-index: 9999;
  left: 0;
  top: 0;
  max-width: 100%;
  max-height: 100%;
  overflow: auto;
  background-color: rgba(0, 0, 0, 0.9);
}

.modal-content {
  margin: auto;
  display: block;
  max-width: 80%;
  max-height: 80%;
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
}

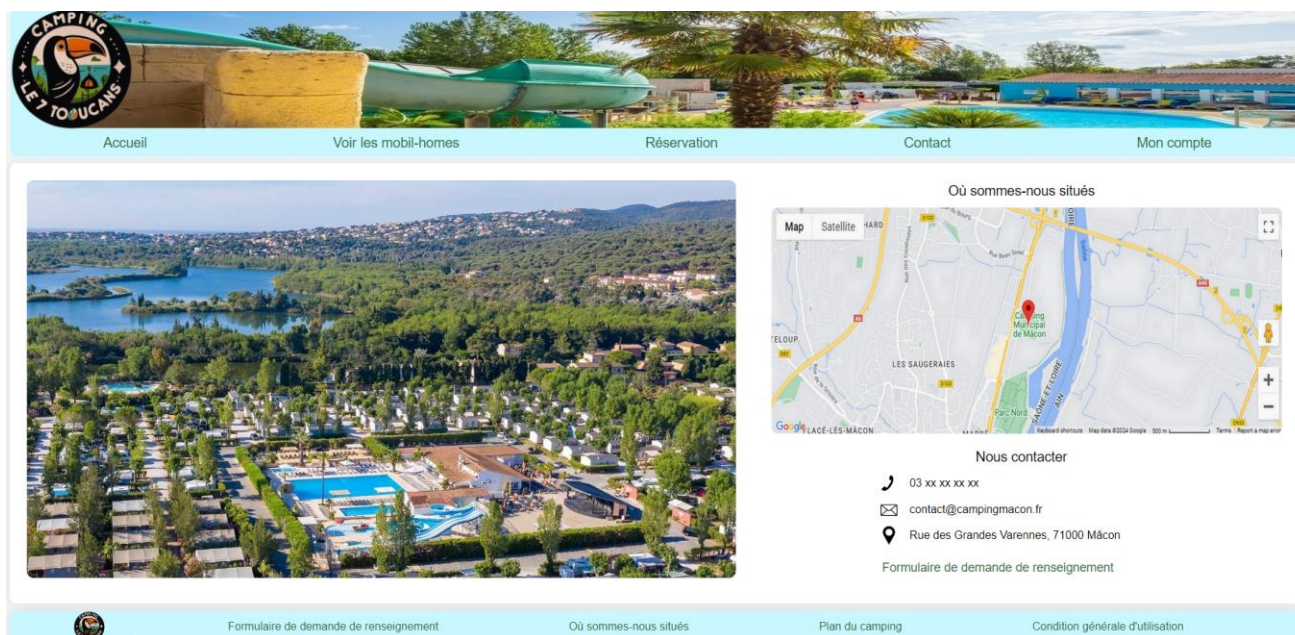
.close {
  position: absolute;
  top: 15px;
  right: 35px;
  color: #fff;
  font-size: 30px;
  font-weight: bold;
  transition: 0.3s;
  cursor: pointer;
}

.close:hover,
.close:focus {
  color: red;
  text-decoration: none;
  cursor: pointer;
}
```

Voici le rendu page d'accueil avec son diaporama fonctionnel et ses promotions :



4- Page de contact : J'ai créé une page de contact, incluant des informations essentielles, un lien vers le formulaire de demande d'information et une image montrant l'emplacement du camping (lorsque l'utilisateur clique sur la carte, un nouvel onglet avec Google Maps est ouvert dans son navigateur). Voici le rendu de cette page :



5- Barre de contact : J'ai ensuite créé une barre de contact afin d'accéder à la page de contact, le formulaire de demande d'information et les conditions générales plus facilement par les utilisateurs, comme-ci dessous :

```
<footer>
  <div class="footer-container">
    <div class="footer-content">
      <ul>
        <li></li>
        <li><a href="formulaire-de-contact.php">Formulaire de demande de renseignement</a></li>
        <li><a href="contact.php">Où sommes-nous situés</a></li>
        <li><a href="#" onclick="openModal('image/plan-camping.jpg')">Plan du camping</a></li>
        <li><a href="cgu.php" target="_blank">Condition générale d'utilisation</a></li>
      </ul>
    </div>
  </div>
</footer>
```

Voici le rendu :



6- De plus, j'ai concentré mes efforts sur l'amélioration du graphisme du site en :

- Palette de couleurs : J'ai ajusté la palette de couleurs pour refléter l'ambiance naturelle et accueillante d'un camping, améliorant ainsi l'attraction visuelle du site.
- Dimensions des images : J'ai optimisé les dimensions des images pour une présentation harmonieuse, améliorant significativement l'expérience visuelle des utilisateurs.
- Réduction des espaces vides : J'ai revisité la mise en page pour minimiser les espaces vides et offrir une expérience plus compacte et agréable lors de la navigation.

Ces ajustements ont été réalisés en réponse aux exigences du projet, visant à rendre le site convivial et attractif pour ses visiteurs.

3.2 Dossier de programmation codes sources documentés et commentés

Les codes sources des nouvelles fonctionnalités ont été commentés pour garantir la clarté du code du site. Par exemple, une page a été créée pour présenter tous les types de mobil-homes du camping. Cette page est structurée de manière à afficher à gauche la description et les tarifs des mobil-homes (utilisation de la classe CSS « mobilhome-texte »), et à droite les images correspondantes (grâce à la classe « mobilhome-image »). L'image principale en grande taille (grâce à la classe « image-principale ») et les autres en taille réduite (avec « photos-mini »).

Ci-dessous, voici le rendu visuel d'un des types de mobil-homes de la page de présentation de ceux-ci.

4 Mobil-Home Luxe 6 personnes

1 grande chambre avec un lit de 140x190, 2 chambre avec 2 lits de 80x190, un grand séjour avec TV, cuisine toute équipée (frigo-congélateur, plaque 4 feux, four, lave-vaisselle, cafetière électrique, micro-onde, kit vaisselle), salle de bain avec douche, 1 salon de jardin avec 2 chaises.

Tarif semaine : 553 €

Réserver maintenant



Voici le code HTML/PHP de cette page :

```
<div class="content">
<?php
$dbdd = new PDO(
$req = $dbdd->prepare("SELECT idtyp, libtyp, descriptlongue, tarifsemaine FROM typemobil");
$req->execute();
$types = $req->fetchAll();

foreach ($types as $type)
{
    echo "<div class='mobilhome'>";
    echo "<div class='mobilhome-texte'>";
    echo "<h3 style='font-weight: normal; font-size: 1.4em;'>$type[libtyp]</h3>";
    echo "<p>$type[descriptlongue]</p>";
    echo "<p>Tarif semaine : $type[tarifsemaine] €</p>"; // Afficher le tarif semaine

    // Ajout du lien vers la page de réservation avec le bouton "Réserver maintenant"
    echo "<br><a href='reservation.php' class='btnsubmit'>Réserver maintenant</a>";

    echo "</div>"; // Fermeture du div mobilhome-texte

    echo "<div class='mobilhome-image'>";
    $req_photos = $dbdd->prepare("SELECT nomfichier FROM photo WHERE idtyp = :idtyp");
    $req_photos->bindParam(':idtyp', $type[idtyp], PDO::PARAM_INT);
    $req_photos->execute();
    $photos = $req_photos->fetchAll();

    echo "<div class='image-principale'>";
    // Afficher la première photo comme image principale
    if (!empty($photos)) {
        $nomfichier = $photos[0][nomfichier];
        $image = "photo/{$type[idtyp]}/$nomfichier";
        if (file_exists($image)) {
            echo "<a href='#' onclick='openModal(\"$image\")'>";
            echo "<img src='$image' class='main-image'>";
            echo "</a>";
        }
    }
    echo "</div>"; // Fermer le div image-principale

    echo "<div class='photos-mini'>";
    // Afficher les autres photos en ligne
    if (count($photos) > 1) {
        for ($i = 1; $i < count($photos); $i++) {
            $nomfichier = $photos[$i][nomfichier];
            $imagePath = "photo/{$type[idtyp]}/$nomfichier";
            if (file_exists($imagePath)) {
                echo "<img src='$imagePath' class='thumbnail' onclick='openModal(\"$imagePath\")'>";
            }
        }
    }
    echo "</div>"; // Fermer le div photos-mini

    $req_photos->closeCursor();
    echo "</div>"; // Fermer le div mobilhome image

    echo "</div><br>"; // Fermer le div mobilhome
}
$dbdd = null;
?>
</div>
```

Ci-dessous, voici le code CSS des classes du code présent :

```
.mobilhome {
  display: flex;
  justify-content: space-between;
  align-items: flex-start;
  margin-bottom: 10px;
}

.mobilhome-texte {
  flex: 1;
  margin: 85px 0 0 100px;
}

.mobilhome-image {
  flex: 1.7;
}

.content {
  display: flex;
  flex-direction: column;
  align-items: center;
}

/* Styles pour les images principales */
.main-image {
  display: block;
  margin: 0 auto; /* Centrer l'image principale */
  border-radius: 10px;
  margin-bottom: -10px;
}

/* Styles pour les miniatures */
.thumbnail {
  max-width: 230px;
  margin: 5px;
  cursor: pointer;
  display: inline-block;
  border-radius: 5px;
  margin-bottom: -10px;
}

.photos-mini {
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
}
```

3.3 Difficultés rencontrées

Au cours de cette semaine, j'ai rencontré quelques défis techniques :

En voulant publier mon site sur un hébergeur, j'ai remarqué que la page affichant tous les mobil-homes s'affichait mal alors qu'en local, la page fonctionnait très bien. Sur l'hébergeur, elle était coupée en deux (même au niveau de la bannière et de ma barre de contact), et les types de mobil-homes s'affichaient tous en double. J'ai passé beaucoup de temps sur ce problème en croyant que c'était un problème dû au CSS, mais en cherchant plus loin et en analysant le code source de la page, j'ai compris que le problème venait de mon code HTML. J'ai donc restructuré mon code HTML et le problème n'a pas persisté. Le problème était en fait une balise div mal placée (dans un if).

J'ai eu beaucoup de conflits de classes CSS, j'ai donc dû reprendre tout le CSS de mon code à plusieurs reprises.

4. Exploitation, Mise en production

4.1 Tests (unitaires, fonctionnels, intégration)

Tout au long du processus de développement, j'ai effectué des tests fréquents sur le site après chaque modification de code. Cette pratique m'a permis de détecter rapidement les erreurs et les bugs potentiels et de les corriger avant qu'ils ne deviennent des problèmes plus importants.

Après avoir apporté des modifications au code source pour intégrer de nouvelles fonctionnalités ou améliorer le graphisme, je procédais immédiatement à des tests sur les pages concernées par ces changements pour m'assurer que les changements étaient bien pris en compte et que le site fonctionnait correctement.

4.2 Installation, déploiement, fonctionnement éventuel en double avec l'ancienne procédure.

Le site est déployé sur l'hébergeur 000webhost pour les vérifications finales et pour voir si il y a des problèmes lors de la publication sur hébergeur.

4.3 Rédaction notice utilisateur

Création de compte et connexion :

- Instructions sur la procédure de création de compte pour les nouveaux utilisateurs.
- Étapes à suivre pour remplir les informations nécessaires et valider le compte et étapes simples pour la connexion des utilisateurs existants.

Réservation d'un séjour :

- Guide étape par étape pour effectuer une réservation en ligne, y compris la sélection des dates de séjour et du type de logement.
- Explication des fonctionnalités du formulaire de réservation.

Possibilité de s'informer sur tous les types de mobil-homes du camping.

- Mise en avant des informations sur les types de mobil-homes ainsi que leur tarif.

Support et assistance :

- Coordonnées et formulaire de demande de renseignement pour contacter le support client en cas de besoin d'assistance ou de questions supplémentaires.

4.4 Formation des utilisateurs

Une formation est dispensée aux administrateurs et utilisateurs finaux sur l'utilisation et la gestion du site, car seulement moi et mon maître de stage auront accès a ce site.

5. Annexe

Lors de ma première semaine de stage, j'ai effectué des recherches sur l'origine de l'expression du code web et j'ai fait un mémoire sur le résultat de mes recherches, ce qui m'a pris toute la première semaine de stage. En plus de mes recherches, j'ai beaucoup échangé avec mon maître de stage sur les étapes du développement d'un site web. J'ai ainsi montré mon mémoire à mon maître de stage qui l'a validé.

J'en ai conclu qu'un site web devait être développé selon un plan de travail :

Le développeur doit d'abord créer le contenu, et le travailler pour le faire correspondre à la structure et au design du site par la suite. Il travaille ensuite l'architecture, puis finit par mettre en place une stratégie commerciale afin d'attirer les visiteurs du site.

Mémoire sur l'Histoire du Développement Web

Introduction

Le développement web a transformé notre accès à l'information et à la communication, depuis les débuts de l'ARPANET dans les années 1960 jusqu'à l'émergence du World Wide Web en 1989 par Tim Berners-Lee.

Le développement web a été marqué par une évolution majeure de la communication, passant d'un réseau militaire à un outil mondial de partage d'informations.

Dans les années 1990, l'accessibilité du web au grand public a été rendue possible grâce à des navigateurs graphiques tels que Mosaic, posant ainsi les bases technologiques du web moderne avec les langages HTML, CSS et JavaScript.

Dans les années 2000, AJAX a permis des applications web interactives sans rechargement de page, et les CMS comme WordPress ont simplifié la gestion de contenu.

Les Framework JavaScript ont facilité le développement Front End, tandis que Node.js a permis un développement Full Stack en JavaScript.

De plus l'avènement de navigateurs concurrents comme Firefox, Safari et Google Chrome a stimulé l'innovation et amélioré les standards web, marquant la fin de la domination d'Internet Explorer.

Ainsi, la conception d'un site internet efficace exige aujourd'hui une intégration harmonieuse du développement, du graphisme et de la stratégie commerciale afin d'attirer et de fidéliser les visiteurs renforçant ainsi sa présence en ligne et son impact commercial.

Le développeur a donc un plan de travail à respecter :

Tout d'abord, le développeur commence par créer le contenu textuel et visuel, incluant la bannière, les images de haute qualité, le logo et la palette de couleurs.

Ensuite, il conçoit l'architecture du site, en assurant une navigation fluide entre les pages et une disposition cohérente du contenu.

Enfin, il met en place une stratégie commerciale claire et ciblée, qui inclut la présentation de l'entreprise pour établir une relation de confiance avec les visiteurs, en mentionnant l'adresse, la disponibilité, la date de création, les contacts, les conditions générales et les compétences.

6. Bilan

6.1 Semaine 1

Cette première semaine m'a permis d'acquérir des bases solides pour le projet de création de site internet. J'ai pu approfondir mes connaissances en développement web, graphisme et mieux comprendre les stratégies marketing utilisées par un site web, tout en bénéficiant des conseils de mon maître de stage. Je suis impatiente de continuer à développer ce projet et d'acquérir de nouvelles compétences.

6.2 Semaine 2

La deuxième semaine de stage a été très productive. Les améliorations apportées au site internet ont considérablement renforcé son attractivité et sa fonctionnalité. Je suis satisfaite du travail que j'ai accompli et je suis motivée pour poursuivre le développement du projet dans les semaines à venir.

6.3 Semaine 3

Durant cette semaine, j'ai concentré mes efforts sur l'amélioration du graphisme et l'ajout de nouvelles pages de mon site vitrine. Ces ajouts ont enrichi l'expérience de utilisateur et renforcé l'identité visuelle du camping. Grâce à des tests réguliers, j'ai assuré la stabilité et la qualité du site. Les défis rencontrés ont été surmontés en collaboration avec mon maître de stage.

6.4 Semaine 4

Pendant la semaine 4, j'ai complété les dernières fonctionnalités du site web du camping fictif, en améliorant l'expérience de utilisateur et optimisant les processus de réservation et de gestion de compte. J'ai implémenté la fonctionnalité de réservation, intégré la gestion des comptes utilisateurs, et effectué des tests rigoureux pour assurer le bon fonctionnement du site. J'ai également apporté des améliorations graphiques et résolu les problèmes d'affichage, finalisant ainsi les fonctionnalités de ce site vitrine conformément aux spécifications définies.

6.5 Semaine 5

La cinquième semaine a été consacrée à la finalisation et à l'optimisation du site web. Les ajustements finaux ont été réalisés suite aux retours des tests utilisateurs effectués par mon maître de stage et moi-même, garantissant ainsi une optimisation des performances et la correction des dernières erreurs. Ce projet m'a permis de consolider mes compétences en développement web et de fournir un produit fonctionnel et attrayant.