



# DOSSIER PROFESSIONNEL (DP)

<i>Nom de naissance</i>	▶ DELAVAL
<i>Nom d'usage</i>	▶ DELAVAL
<i>Prénom</i>	▶ Maÿlis
<i>Adresse</i>	▶ 11 rue de l'écluse, Beaulieu 10400, Le Mériot

## Titre professionnel visé

Titre Professionnel – Développement Web et Web Mobile

### MODALITE D'ACCES :

- Parcours de formation
- Validation des Acquis de l'Expérience (VAE)

## Présentation du dossier

Le dossier professionnel (DP) constitue un élément du système de validation du titre professionnel. **Ce titre est délivré par le Ministère chargé de l'emploi.**

Le DP appartient au candidat. Il le conserve, l'actualise durant son parcours et le présente **obligatoirement à chaque session d'examen.**

Pour rédiger le DP, le candidat peut être aidé par un formateur ou par un accompagnateur VAE.  
Il est consulté par le jury au moment de la session d'examen.

### Pour prendre sa décision, le jury dispose :

1. des résultats de la mise en situation professionnelle complétés, éventuellement, du questionnaire professionnel ou de l'entretien professionnel ou de l'entretien technique ou du questionnement à partir de productions.
2. du **Dossier Professionnel** (DP) dans lequel le candidat a consigné les preuves de sa pratique professionnelle
3. des résultats des évaluations passées en cours de formation lorsque le candidat évalué est issu d'un parcours de formation
4. de l'entretien final (dans le cadre de la session titre).

*[Arrêté du 22 décembre 2015, relatif aux conditions de délivrance des titres professionnels du ministère chargé de l'Emploi]*

### Ce dossier comporte :

- ▶ pour chaque activité-type du titre visé, un à trois exemples de pratique professionnelle ;
- ▶ un tableau à renseigner si le candidat souhaite porter à la connaissance du jury la détention d'un titre, d'un diplôme, d'un certificat de qualification professionnelle (CQP) ou des attestations de formation ;
- ▶ une déclaration sur l'honneur à compléter et à signer ;
- ▶ des documents illustrant la pratique professionnelle du candidat (facultatif)
- ▶ des annexes, si nécessaire.

*Pour compléter ce dossier, le candidat dispose d'un site web en accès libre sur le site.*

 <http://travail-emploi.gouv.fr/titres-professionnels>

## Sommaire

### Exemples de pratique professionnelle

#### Activité-type n°1 : Développement front-end et interfaces

p. 5

- ▶ *Intitulé de l'exemple n°1 : Configurer l'environnement de développement web* ..... p. 5
- ▶ *Intitulé de l'exemple n°2 : De la maquette à l'interface web statique* ..... p. 8
- ▶ *Intitulé de l'exemple n°3 : Intégration de la partie dynamique* ..... p. 11

#### Activité-type n°2 : Développement back-end et architecture technique

p.

- ▶ *Intitulé de l'exemple n°1 : Concevoir et développer l'architecture données* ..... p. 14
- ▶ *Intitulé de l'exemple n°2 : Création d'un écosystème WordPress sur mesure avec composants dynamiques:* ..... p. 17
- ▶ *Intitulé de l'exemple n°3 : Assurer la documentation technique et le déploiement* ..... p. 20

#### Titres, diplômes, CQP, attestations de formation *(facultatif)*

p. 23

#### Déclaration sur l'honneur

p. 24

#### Documents illustrant la pratique professionnelle *(facultatif)*

#### Annexes *(si le RC le prévoit)*

p. 25

- ▶ *Annexe 1 – Maquette Accueil : Desktop, Tablette et Mobile* ..... p. 26
- ▶ *Annexe 2 – Rendu map Leaflet.js et Code Animation Simulateur* ..... p. 27
- ▶ *Annexe 3 – Initialisation de la Base de Données CRUD* ..... p. 28
- ▶ *Annexe 4 – Mesures de Sécurité pour les Avis* ..... p. 29

# EXEMPLES DE PRATIQUE PROFESSIONNELLE

## Activité-type 1 Développement front-end et interfaces

Exemple n°1 ► Configurer l'environnement de développement web

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Dans le cadre de la mise en place du projet, j'ai configuré un environnement de développement complet, fiable et cohérent, afin d'assurer une base solide pour le développement front-end et l'intégration sur WordPress. Cette étape initiale était cruciale car elle garantissait une standardisation du code, une organisation rigoureuse du projet ainsi qu'une fluidité dans les futures phases de développement.

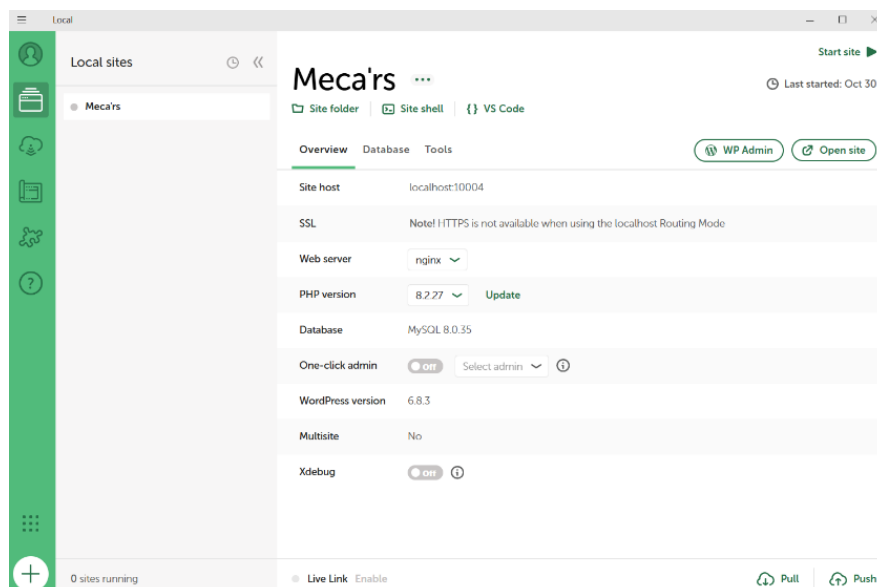
J'ai d'abord procédé à l'installation et à la configuration de Visual Studio Code, en créant un *workspace* dédié au projet pour centraliser l'ensemble des dossiers nécessaires (thème WordPress, assets, configuration Git...). J'ai ensuite installé et paramétré les extensions essentielles, notamment Live Server, Prettier et des extensions dédiées à WordPress et PHP, afin d'assurer le respect des normes de codage et d'automatiser la correction des erreurs courantes. Prettier a été configuré avec des règles spécifiques (indentation, trailing comma, guillemets, etc.) pour garantir l'uniformité du code entre tous les contributeurs.

Parallèlement, j'ai mis en place l'environnement WordPress en local via LocalWP. J'ai créé un site vierge configuré avec PHP 8.2, un serveur Apache, et une base de données MySQL 8.0. J'ai ajusté les paramètres essentiels tels que les permaliens (*URL rewriting*) et installé un thème de développement vide destiné à accueillir mes futurs développements. Cela permettait de bénéficier d'un environnement isolé, stable et reproductible.

Pour structurer la maintenance et le suivi du projet, j'ai initialisé un dépôt Git local avant de le connecter à GitHub. J'ai mis en place une organisation claire du versioning. J'ai ensuite fait la hiérarchie de mon dossier.

J'ai également utilisé Notion pour suivre mon projet de A à Z et de garder les notes sur chaque avancée.

Enfin, j'ai intégré l'utilisation de Figma pour la partie design, en créant le design system en premier temps.



# DOSSIER PROFESSIONNEL (DP)

## 2. Précisez les moyens utilisés :

- **Environnement** : Windows 11 avec double écran pour le développement et le preview,
- **Éditeur** : VSCode avec workspace configuré et extensions optimisées,
- **Local** : LocalWP avec PHP 8.2, MySQL 8.0, serveur Apache,
- **Design** : Figma avec design system structuré en composants,
- **Versioning** : GitHub avec workflow de branches,
- **Documentation** : Notion pour le suivi de projet et procédures.



## 3. Avec qui avez-vous travaillé ?

J'ai travaillé en autonomie sur la configuration technique, avec des validations ponctuelles par mon tuteur José MECA sur les outils choisis.

## 4. Contexte

Nom de l'entreprise, organisme ou association ► *Meca'rs – Garage automobile*

Chantier, atelier, service ► Cliquez ici pour taper du texte.

Période d'exercice ► Du **25/08/2025** au **17/10/2025**

# DOSSIER PROFESSIONNEL (DP)

## 5. Informations complémentaires *(facultatif)*

La mise en place de cet environnement m'a permis de travailler dans des conditions professionnelles proches de celles d'une équipe de développement. Cela a facilité la standardisation du code, l'automatisation de certaines tâches, la bonne pratique du versioning, et une meilleure communication avec les autres intervenants (graphiste, chef de projet).

Cette étape préparatoire a eu un impact direct sur la qualité des livrables et sur la fluidité des phases ultérieures du développement front-end.

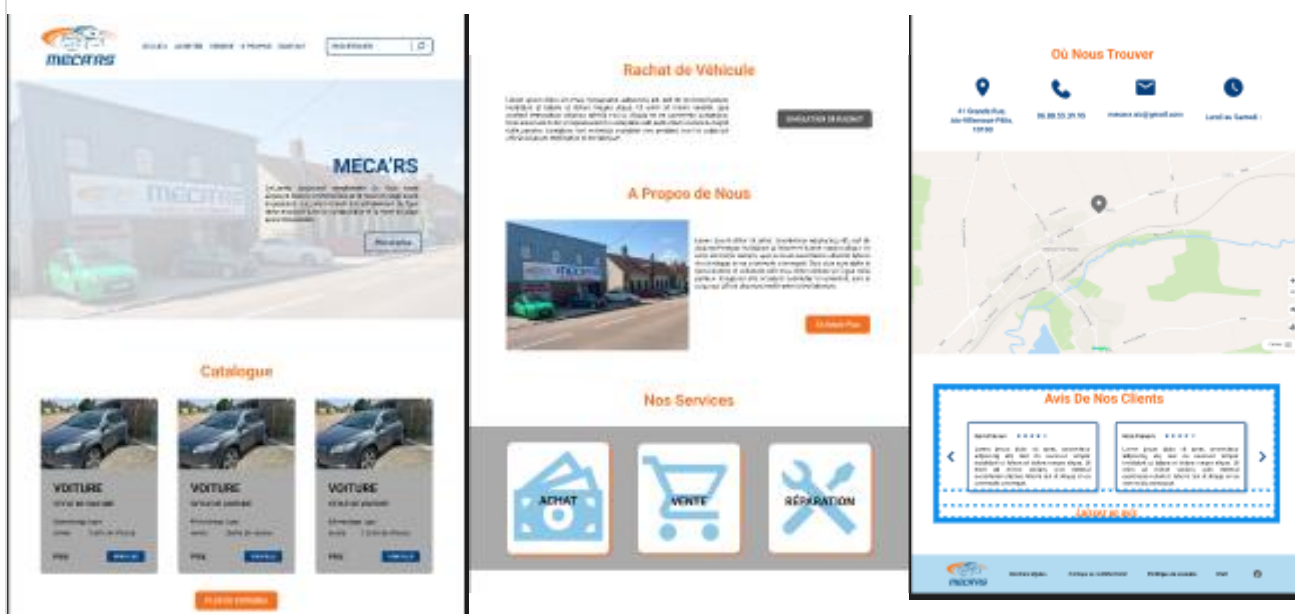
## Activité-type 1 Développement front-end et interfaces

Exemple n°2 ▶ De la maquette à l'interface web statique

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

J'ai conçu et développé l'interface complète du site vitrine en suivant une méthodologie structurée. La phase initiale a consisté à créer un design system complet sur Figma centralisant la palette couleur du garage (issue de l'identité visuelle existante), la typographie Roboto avec ses hiérarchies (h1 à h6), et tous les composants réutilisables (boutons, cartes véhicules, header, footer). J'ai spécifiquement adapté chaque composant aux trois breakpoints : desktop, tablette et mobile.

J'ai développé des animations CSS avancées incluant des cartes 3D avec effet de retournement au hover, enrichissant l'expérience utilisateur.



L'intégration HTML/CSS/JavaScript a débuté après validation des maquettes par le client. J'ai rencontré des défis techniques significatifs, notamment sur le menu burger mobile qui a nécessité plusieurs refontes de structure HTML/CSS, et sur l'organisation des fichiers avec des problèmes de chemins d'accès résolus par une réorganisation complète de l'arborescence. J'ai également optimisé la typographie en ajustant les tailles de police selon les standards web après une étude comparative.



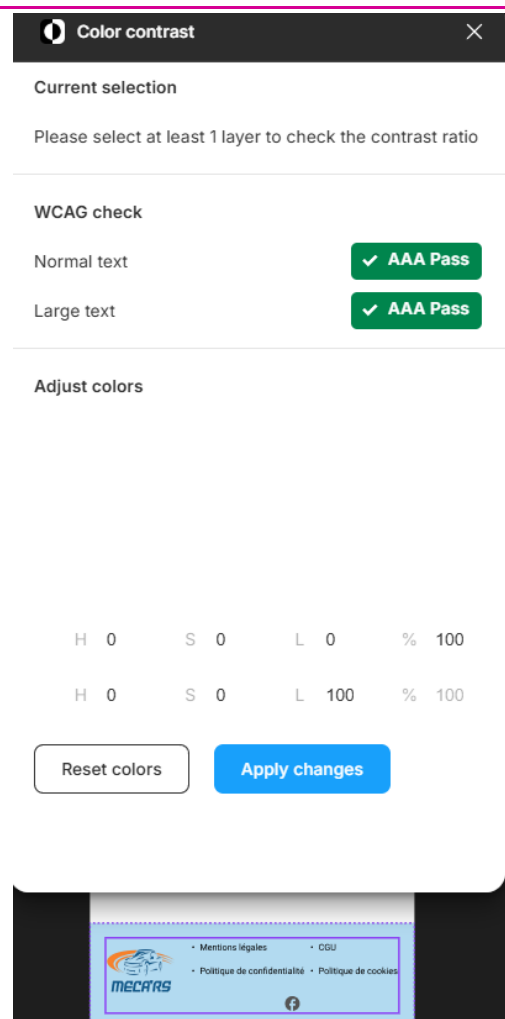
# DOSSIER PROFESSIONNEL (DP)

Le processus d'intégration a suivi une approche méthodique : page d'accueil d'abord avec ses animations, puis les 5 autres pages (catalogue, fiche produit, simulateur, à propos, contact) avec validation responsive à chaque étape. L'accessibilité a été une priorité, avec vérification des contrastes couleur (normes WCAG AA) et structure sémantique HTML5.



## 2. Précisez les moyens utilisés :

- **Design** : Figma pour les maquettes de chaque pages avec les différentes tailles d'écran ;
- **Développement** : VSCode, HTML5 sémantique, CSS3 avec variables custom,
- **Responsive** : Media queries desktop-first (breakpoints 1024px, 768px),
- **Accessibilité** : Validation WCAG niveau AA, contrastes vérifiés et changés si nécessaire,
- **Gestion** : GitHub, organisation méthodique des dossiers.



# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

Validation itérative des maquettes et des rendus finaux. Feedback sur l'expérience utilisateur et la cohérence visuelle et en autonomie avec de la prise de décision sur les optimisations techniques et la résolution des blocages.

## 4. Contexte

Nom de l'entreprise, organisme ou association ▶ *Meca'rs – Garage automobile*

Chantier, atelier, service ▶ Cliquez ici pour taper du texte.

Période d'exercice ▶ Du **25/08/2025** au **17/10/2025**

## 5. Informations complémentaires (facultatif)

L'approche desktop-first a été choisie car le client priorisait l'expérience ordinateur tout en garantissant une accessibilité mobile optimale. Les breakpoints ont été soigneusement définis pour assurer une transition fluide entre les formats.

Les défis techniques rencontrés (menu burger, organisation fichiers) ont renforcé mes compétences en debugging et en architecture front-end. La création du design system en amont s'est avérée cruciale pour maintenir la cohérence visuelle malgré les adaptations techniques. L'attention portée à l'accessibilité et aux standards W3C garantit une expérience utilisateur inclusive et de qualité.

## Activité-type 1 Développement front-end et interfaces

### Exemple n°3 ▶ Intégration de la partie dynamique

#### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

J'ai développé les fonctionnalités interactives avancées du site en intégrant plusieurs bibliothèques JavaScript spécialisées. Pour les carrousels de véhicules, j'ai choisi Swiper.js pour ses performances optimisées (utilisation du GPU) et son expérience tactile native, en configurant des paramètres responsifs adaptés à chaque breakpoint.

```
Swiper.js avec paramètres responsifs

1  const options = {
2    loop: true,
3    speed: 600,
4    slidesPerView: 1,           // ✅ TOUJOURS 1 slide visible
5    spaceBetween: 10,         // Espacement de base
6    autoplay: {
7      delay: 4000,
8      disableOnInteraction: false,
9    },
10   // ✅ BREAKPOINTS POUR ADAPTER L'ESPACEMENT SEULEMENT
11   breakpoints: {
12     // Mobile
13     320: {
14       slidesPerView: 1,
15       spaceBetween: 10
16     },
17     // Tablette - plus d'espace
18     768: {
19       slidesPerView: 1,
20       spaceBetween: 30
21     },
22     // Desktop - encore plus d'espace
23     1024: {
24       slidesPerView: 1,
25       spaceBetween: 50
26     }
27   },
28   // Navigation conditionnelle
29   pagination: paginationEl ? {
30     el: paginationEl,
31     clickable: true,
32     dynamicBullets: true // ✅ Amélioration responsive
33   } : {},
34   navigation: (nextEl && prevEl) ? {
35     nextEl,
36     prevEl
37   } : {},
38 };
```

# DOSSIER PROFESSIONNEL (DP)

Sur la page d'accueil et la page contact, j'ai implémenté Leaflet.js pour la carte interactive, préférée à Google Maps pour son caractère open source et son poids réduit (40KB), avec personnalisation visuelle pour l'harmonisation à la charte graphique.

```
Leaflet.js
1 L.tileLayer("https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png", {
2   attribution: "@ OpenStreetMap",
3 }).addTo(map);
4
5 // Crée une icône personnalisée pour représenter le garage
6 const carIcon = L.icon({
7   iconUrl: "https://cdn.jsdelivr.net/npm/bootstrap-icons/icons/car-front-
fill.svg",
8   iconSize: [40, 40], // Taille de l'icône
9   iconAnchor: [20, 40], // Point d'ancrage de l'icône
10  popupAnchor: [0, -40], // Position du popup par rapport à l'icône
11 });
12
13 // Ajoute un marqueur à l'emplacement du garage
14 L.marker([garageLat, garageLng], { icon: carIcon })
15 .addTo(map)
16 .bindPopup(`
17   <b> 🚗 Meca'rs</b><br>
18   41 Grande Rue<br>
19   10190 Aix-Villemaur-Pâlis<br>
20   <a href="https://www.google.com/maps?q=${garageLat},${garageLng}"
target="_blank"> Voir sur Google Maps</a>
21 `)
22 .openPopup(); // Ouvre automatiquement le popup au chargement
23 } else {
24   console.warn('Leaflet non chargé ou élément #map absent - initialisation carte
ignorée.');
```

Le simulateur d'estimation a représenté le challenge technique principal, avec le développement d'une API REST personnalisée pour les calculs en temps réel basés sur multiple critères (marque, année, kilométrage, état), j'ai choisi cette solution pour son adéquation parfaite avec les besoins métier spécifiques du garage et l'indépendance vis-à-vis des fournisseurs externes."

## 2. Précisez les moyens utilisés :

- **Swiper.js** : Configuration responsive avec breakpoints pour l'adaptation des espacements, dynamicBullets pour la pagination adaptative, autoplay avec gestion des interactions,
- **Leaflet.js** : Carte OpenStreetMap, marqueur personnalisé, contrôle de zoom responsive,
- **API REST maison** : Endpoints Wordpress personnalisés, calculs métier, gestion d'erreurs,
- **JavaScript ES6+** : Async, await, debouncing (500ms), validation en temps reel, manipulation DOM avancée,
- **Animations CSS** : Keyframes pour le loading, transitions pour les résultats du simulateur.

Estimation en cours...



# DOSSIER PROFESSIONNEL (DP)

## Votre Renault Clio (2012 - 25 155 km)

Estimation : 2 752 € – 3 238 €

Cette estimation est indicative.

[Contactez-nous](#) pour une offre précise et personnalisée.

### 3. Avec qui avez-vous travaillé ?

J'ai travaillé en autonomie sur la configuration technique et sur les tests utilisateurs.

### 4. Contexte

Nom de l'entreprise, organisme ou association ► **Meca'rs – Garage automobile**

Chantier, atelier, service ► Cliquez ici pour taper du texte.

Période d'exercice ► Du **25/08/2025** au **17/10/2025**

### 5. Informations complémentaires (facultatif)

Le choix stratégique de Swiper.js et Leaflet.js a démontré ma capacité à sélectionner des solutions techniques optimales en fonction des contraintes projet (performance, maintenabilité, expérience utilisateur).

Le développement de l'API REST personnalisée a démontré ma capacité à concevoir une architecture back-end sur mesure, spécifiquement adaptée aux besoins métier du garage. L'intégration de Swiper.js avec configuration responsive a validé mon approche d'optimisation progressive, privilégiant la stabilité visuelle sur tous les devices. La gestion des animations synchronisées avec les appels API dans le simulateur a renforcé mes compétences en UX technique.

## Activité-type 2 Développement back-end et architecture technique

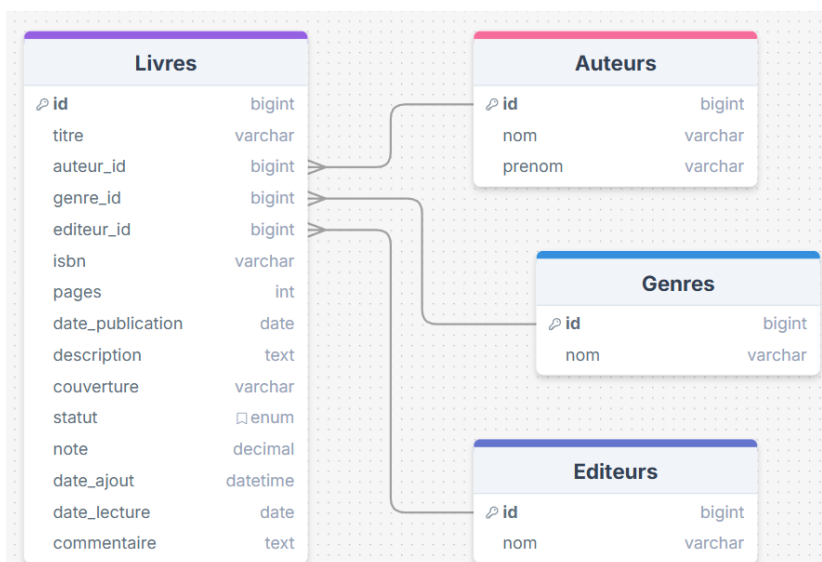
### Exemple n°1 ► Concevoir et développer l'architecture données

#### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

En complément du site vitrine WordPress (qui utilisait le paradigme "Post Type" de WP), j'ai initié et développé de A à Z une application de gestion de bibliothèque en PHP procédural et SQLite pour démontrer ma maîtrise fondamentale des architectures de données relationnelles dans un environnement léger et autonome. Ce choix technologique m'a permis de travailler sur l'ensemble de la stack données sans dépendance à un serveur MySQL externe.

J'ai délibérément opté pour un schéma relationnel simplifié avec des relations 1-N afin de garantir une maintenance aisée et des performances optimales dans un contexte de bibliothèque personnelle.

J'ai débuté par la modélisation conceptuelle aboutissant à 4 entités principales (Livres, Auteurs, Genres, Éditeurs) avec des clés étrangères directes. Cette architecture simplifiée permet des requêtes JOIN efficaces tout en maintenant l'intégrité référentielle.



L'implémentation technique a inclus le développement d'une API complète avec :

- **Gestion CRUD avancée** : Endpoints spécialisés pour chaque opération (add\_auteur.php, update\_book.php, delete\_book.php...),
- **Sécurité renforcée** : Utilisation systématique de requêtes préparées PDO pour prévenir les injections SQL,
- **Gestion d'erreurs complète** : Codes HTTP appropriés (400, 404, 405, 409, 500) et messages d'erreur explicites,

```
add_auteur.php
1 // Vérification de l'existence pour éviter les doublons
2 $sqlCheck = "SELECT id FROM auteurs WHERE nom = ? AND prenom = ?";
3 $stmtCheck = $pdo->prepare($sqlCheck);
4 $stmtCheck->execute([$nom, $prenom]);
5
6 if ($stmtCheck->fetch()) {
7     http_response_code(409); // Conflict - Ressource déjà existante
8     echo json_encode(['error' => 'Cet auteur existe déjà']);
9     exit;
10 }
```

# DOSSIER PROFESSIONNEL (DP)

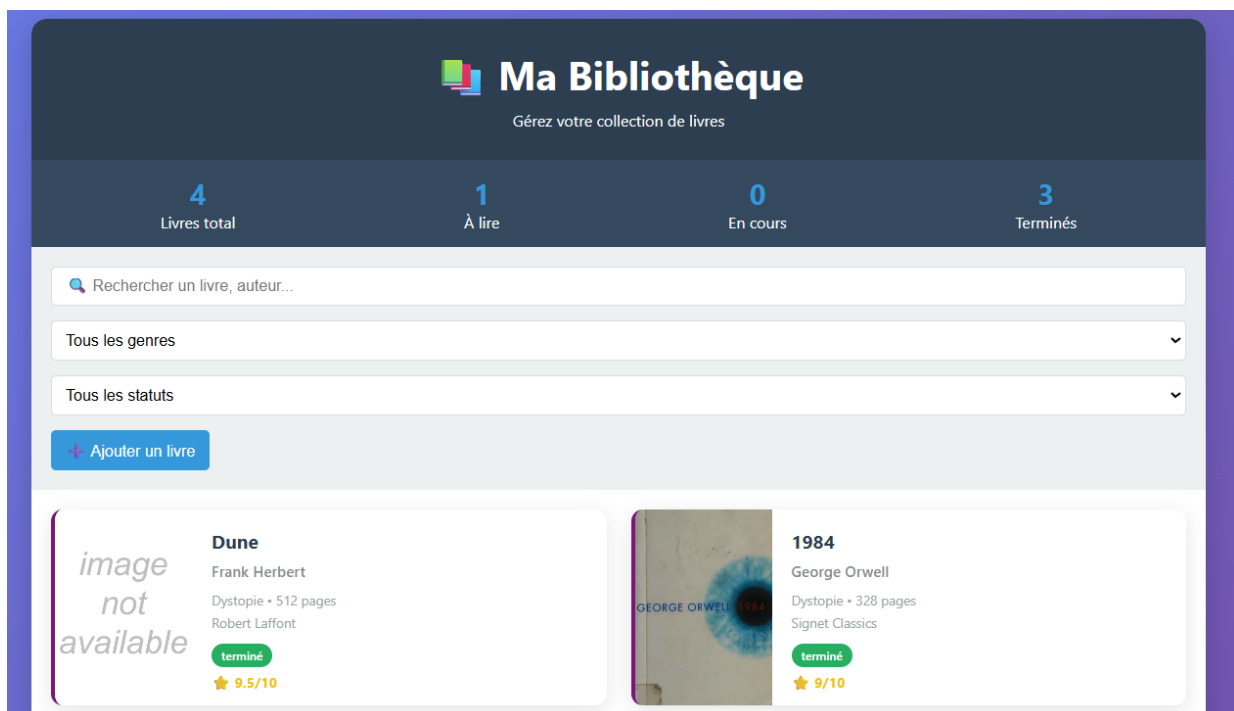
- **Validation des données** : Contrôles stricts en backend pour l'intégrité des données.

```
get_book_detail.php

1 //Requêtes JOIN complexes pour la récupération de données :
2 $sql = "SELECT l.*, a.nom as auteur_nom, a.prenom as auteur_prenom,
3         g.nom as genre_nom, e.nom as editeur_nom
4         FROM livres l
5         JOIN auteurs a ON l.auteur_id = a.id
6         JOIN genres g ON l.genre_id = g.id
7         JOIN editeurs e ON l.editeur_id = e.id
8         WHERE l.id = ?";
```

## 2. Précisez les moyens utilisés :

- **Architecture Backend** : API RESTful avec PHP procédural, PDO/SQLite,
- **Sécurité** : Requêtes préparées, validation/sanitisation, codes http sémantiques,
- **Base de données** : SQLite avec schéma relationnel normalisé, clés étrangères,
- **Algorithmes avancés** : Système de scoring pour recherche des livres, normalisation de texte,
- **Intégrations externes** : Google Books API avec gestion d'erreurs et retry logic,
- **Frontend** : JavaScript ES6 (Fetch API), HTML5, CSS3,
- **Optimisation** : Jointures SQL efficaces, agrégations pour statistiques.



# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

Autonomie complète : Conception, développement, tests et documentation.

## 4. Contexte

Nom de l'entreprise, organisme ou association ► **Meca'rs – Garage automobile**

Chantier, atelier, service ► Cliquez ici pour taper du texte.

Période d'exercice ► Du **25/08/2025** au **17/10/2025**

## 5. Informations complémentaires (facultatif)

Ce projet démontre ma capacité à concevoir et implémenter une architecture de données complète avec des composants sophistiqués :

- Sécurité industrielle via l'utilisation systématique des requêtes préparées,
- Gestion d'erreurs professionnelle avec codes HTTP appropriés,
- Algorithmes métier complexes (système de scoring pour la recherche),
- Intégration d'API externe avec traitement et nettoyage des données,
- Optimisation des performances via des requêtes SQL efficaces.

La qualité du code produit (validation rigoureuse, gestion des transactions, architecture modulaire) témoigne d'une approche professionnelle du développement back-end et de la gestion de données.

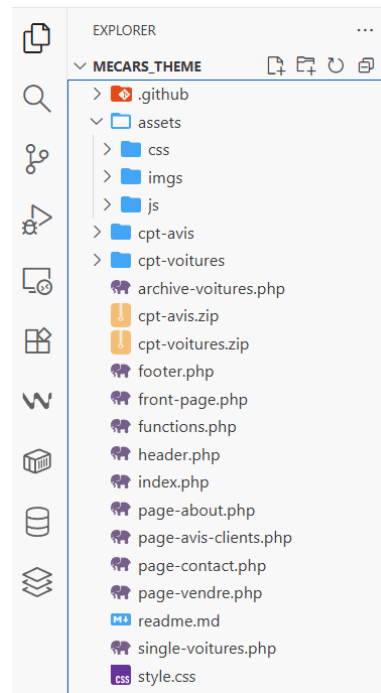
## Activité-type 2 Développement back-end et architecture technique

Exemple n°2 ▶ *Création d'un écosystème WordPress sur mesure avec composants dynamiques*

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

J'ai développé un thème WordPress entièrement personnalisé, spécifiquement architecturé pour répondre aux besoins métier du garage automobile. Le processus a débuté par l'analyse approfondie des besoins fonctionnels pour concevoir une structure technique optimale. J'ai implémenté la hiérarchie des templates WordPress en créant l'ensemble des fichiers nécessaires : header.php et footer.php modulaires avec système de navigation dynamique, front-page.php pour la page d'accueil avec sections configurables, single-voitures.php pour les fiches produits détaillées, et archive-voitures.php pour le catalogue avec système de filtres intégré.

```
CPT Voitures
1 <?php
2 /**
3  * Plugin Name: CPT Voitures
4  * Description: Ajoute un Custom Post Type "Voitures" pour gérer les voitures.
5  * Version: 1.0
6  * Author: Mecars
7  */
8 // Vérification de sécurité pour empêcher l'accès direct
9 if ( ! defined( 'ABSPATH' ) ) {
10     exit;
11 }
12
13 // Enregistrer le CPT Voitures
14 function creer_cpt_voitures() {
15     $labels = array(
16         'name' => 'Voitures',
17         'singular_name' => 'Voiture',
18         'menu_name' => 'Voitures',
19         'all_items' => 'Toutes les voitures',
20         'add_new' => 'Ajouter une voiture',
21         'add_new_item' => 'Ajouter une nouvelle voiture',
22         'edit_item' => 'Modifier la voiture',
23         'new_item' => 'Nouvelle voiture',
24         'view_item' => 'Voir la voiture',
25         'search_items' => 'Rechercher une voiture',
26         'not_found' => 'Aucune voiture trouvée',
27         'not_found_in_trash' => 'Aucune voiture trouvée dans la corbeille',
28     );
29 // Configuration des paramètres du Custom Post Type
30 $args = array(
31     'labels' => $labels,
32     'public' => true,
33     'show_ui' => true,
34     'show_in_menu' => true,
35     'show_in_rest' => true,
36     'capability_type' => 'post',
37     'hierarchical' => false,
38     'menu_position' => 5,
39     'menu_icon' => 'dashicons-car',
40     'supports' => array('title', 'editor', 'thumbnail'),
41     'has_archive' => true,
42     'rewrite' => array('slug' => 'voitures'),
43     'publicly_queryable' => true,
44 );
45 register_post_type('voitures', $args);
46 }
47 add_action('init', 'creer_cpt_voitures');
48
49 // Forcer ACF à reconnaître le CPT "voitures" dans ses interfaces
50 add_filter('acf/get_post_types', function ($post_types) {
51     $post_types['voitures'] = 'Voitures';
52     return $post_types;
53 });
```



Pour le Custom Post Types « Voitures », j'ai développé une structure complète spécialement conçue pour la gestion du parc automobile avec une intégration poussée d'Advanced Custom Fields. J'ai créé des groupes de champs personnalisés pour toutes les caractéristiques techniques : marque, modèle, année, prix, kilométrage, carburant, et équipements. L'interface administrateur, entièrement en français, a été optimisée avec une icône dédiée et une organisation logique des métadonnées.

```
CPT Avis Clients
1 <?php
2 /*
3 Plugin Name: CPT Avis
4 Description: Gestion des avis clients sur le site Mecars
5 Version: 1.0
6 Author: Mecars
7 */
8
9 // Enregistrement du Custom Post Type pour les avis clients
10 function mecars_register_aviscpt() {
11     $labels = array(
12         'name' => 'Avis Clients',
13         'singular_name' => 'Avis Client',
14         'menu_name' => 'Avis Clients',
15         'name_admin_bar' => 'Avis Client',
16         'add_new' => 'Ajouter un avis',
17         'add_new_item' => 'Ajouter un nouvel avis',
18         'edit_item' => 'Modifier l\'avis',
19         'new_item' => 'Nouvel avis',
20         'view_item' => 'Voir l\'avis',
21         'all_items' => 'Tous les avis',
22         'search_items' => 'Rechercher un avis',
23         'not_found' => 'Aucun avis trouvé',
24         'not_found_in_trash' => 'Aucun avis dans la corbeille',
25     );
26
27 // Configuration des paramètres du Custom Post Type
28 $args = array(
29     'labels' => $labels,
30     'public' => true,
31     'has_archive' => false,
32     'menu_icon' => 'dashicons-star-filled',
33     'supports' => array('title'),
34     'show_in_rest' => false,
35 );
36
37 // Enregistrement effectif du Custom Post Type dans WordPress
38 register_post_type('avis_client', $args);
39 }
40
41 // Hook pour exécuter la fonction lors de l'initialisation de WordPress
42 add_action('init', 'mecars_register_aviscpt');
```

L'intégration systématique des ACF a transformé l'interface d'administration en outil dynamique, avec la configuration de groupes de champs complets pour les véhicules (marque, modèle, prix, caractéristiques techniques) et la mise en place d'une page d'options globale pour les contenus récurrents. Des mesures de sécurité robustes ont été implémentées, incluant la validation multi-niveau des données, la protection CSRF via les nonces WordPress, et la sécurisation des requêtes base de données.

Concernant le CPT « Avis Clients », j'ai développé un système complet de gestion des témoignages via un plugin personnalisé. Le CPT est configuré avec une interface épurée utilisant uniquement le titre et une icône en étoile, optimisé pour une saisie rapide. J'ai implémenté un formulaire sécurisé avec vérification des nonces WordPress, champ honeypot anti-spam, et protection contre les soumissions multiples (limite d'un avis par email toutes les 2 heures). Le système intègre une modération systématique avec statut "pending" et validation des données côté serveur, incluant la sanitisation des champs et la vérification des emails. Les avis sont stockés via Advanced Custom Fields avec métadonnées structurées (nom, email, note, service) garantissant une gestion professionnelle de la réputation client.

```
functions.php Traitement des avis
1 <?php
2 // Traitement sécurisé du formulaire d'avis
3 function mecars_handle_submit_client_review() {
4     // Vérification du nonce de sécurité
5     if (!wp_verify_nonce($_POST['submit_client_review_nonce'],
6         'submit_client_review')) {
7         wp_safe_redirect(add_query_arg('review_status', 'invalid_nonce',
8             wp_get_referer()));
9         exit;
10    }
11
12    // Honeypot anti-spam
13    if (!empty($_POST['hp_name'])) {
14        wp_safe_redirect(wp_get_referer());
15        exit;
16    }
17
18    // Récupération et validation des données
19    $name = sanitize_text_field($_POST['name']);
20    $email = sanitize_email($_POST['email']);
21    $note = intval($_POST['note']);
22    $review = sanitize_textarea_field($_POST['your_review']);
23
24    // Création de l'avis avec modération
25    $new_review = array(
26        'post_title' => wp_trim_words($name, 10, ''),
27        'post_content' => $review,
28        'post_status' => 'pending',
29        'post_type' => 'avis_client',
30    );
31
32    $post_id = wp_insert_post($new_review);
33
34    // Sauvegarde des métadonnées ACF
35    if ($post_id && function_exists('update_field')) {
36        update_field('nom_client', $name, $post_id);
37        update_field('email', $email, $post_id);
38        update_field('note', $note, $post_id);
39        update_field('avis_texte', $review, $post_id);
40    }
41 }
```

# DOSSIER PROFESSIONNEL (DP)

## 2. Précisez les moyens utilisés :

- **PHP/WordPress** : Développement thème sur mesure, template hierarchy, hooks (add\_action, add\_filter),
- **Custom Post Types** : Enregistrement CPT "Voitures" et « Avis Clients », configuration avancée (register\_post\_type),
- **Advanced Custom Fields** : Groupes de champs, Options Page, champs répétables, intégration CPT,
- **Sécurité** : Sanitisation WP (sanitize\_text\_field, sanitize\_email), échappement sorties (esc\_attr), nonces, \$wpdb->prepare(),
- **Environnement** : LocalWP, structure modulaire, gestion des assets.



The screenshot shows the WordPress Custom Post Types (CPT) list. At the top, it says "Tous (4) | Actifs (4)". Below this is a table with two columns: "Titre" and "Description". There are four rows of CPTs, each with a checkbox, a title, and a description. The titles are "À Propos", "Avis Clients", "Page d'accueil", and "Voitures". Each title has a link to "Modifier" and "Dupliquer", and a "Corbeille" (trash) icon. The descriptions are all empty.

<input type="checkbox"/>	Titre	Description
<input type="checkbox"/>	À Propos Modifier   Dupliquer   Désactiver   Corbeille	—
<input type="checkbox"/>	Avis Clients Modifier   Dupliquer   Désactiver   Corbeille	—
<input type="checkbox"/>	Page d'accueil Modifier   Dupliquer   Désactiver   Corbeille	—
<input type="checkbox"/>	Voitures Modifier   Dupliquer   Désactiver   Corbeille	—
<input type="checkbox"/>	Titre	Description

## 3. Avec qui avez-vous travaillé ?

J'ai travaillé en autonomie sur le développement complet du thème, des CPT, de l'intégration des ACF et la sécurisation.

## 4. Contexte

Nom de l'entreprise, organisme ou association ► **Meca'rs – Garage automobile**

Chantier, atelier, service ► Cliquez ici pour taper du texte.

Période d'exercice ► Du **25/08/2025** au **17/10/2025**

## 5. Informations complémentaires (facultatif)

La maîtrise de la hiérarchie des templates WordPress et l'intégration avancée d'ACF ont permis de créer une solution sur mesure alliant puissance technique et expérience utilisateur optimisée. En adoptant le point de vue de mon tuteur, j'ai conçu une interface d'administration immédiatement compréhensible, permettant une gestion autonome et fluide du contenu sans nécessiter de formation préalable. Les Custom Post Types développés offrent une fondation solide pour l'évolution future du site, tout en répondant précisément aux exigences métier du garage. Cette approche centrée sur l'utilisateur final a renforcé ma capacité à traduire des besoins concrets en solutions techniques à la fois robustes et accessibles.











## Activité-type 2 Développement back-end et architecture technique

Exemple n°3 ► Assurer la documentation technique et le déploiement

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

J'ai orchestré la mise en production complète du site vitrine WordPress pour le garage MECA'RS en mettant en place une pipeline CI/CD professionnelle avec GitHub Actions.

Cette démarche a commencé par la création d'un compte FTP dédié sur O2Switch, spécifiquement configuré pour les déploiements automatisés. J'ai ensuite configuré les secrets d'accès dans les paramètres du repository GitHub (hébergeur FTP, nom d'utilisateur, mot de passe) pour sécuriser les informations sensibles.

Name 	Last updated
 FTP_HOST	last week  
 FTP_PASSWORD	last week  
 FTP_USERNAME	last week  

J'ai adopté une approche pragmatique du versioning avec une branche DEPLOYMENT comme branche de développement active et MAIN comme branche de référence. À chaque nouvelle fonctionnalité sur deployment, je procédais à un merge contrôlé vers main avant de déclencher le déploiement manuellement le workflow pour déployer vers la production. Cette méthode me permettait de garder un historique Git propre tout en conservant un contrôle total sur le moment du déploiement.

L'automatisation était pilotée par un fichier YML personnalisé (.github/workflows/main.yml) que j'ai développé pour définir précisément le processus de déploiement. Ce workflow déclenchait une connexion FTP sécurisée vers O2Switch, déployait intelligemment uniquement les fichiers modifiés via l'action FTP-Deploy-Action, et générait des logs détaillés pour le monitoring. La configuration YAML incluait des exclusions stratégiques des fichiers de développement et une gestion optimisée des transferts différentiels.

4 workflow runs	
 adding readme and debug responsive partially  Deploy website on push #4: Commit <a href="#">660c067</a> pushed by <a href="#">MaylisD</a>	<a href="#">main</a>
 again on the simulation  Deploy website on push #3: Commit <a href="#">b310f29</a> pushed by <a href="#">MaylisD</a>	<a href="#">main</a>
 modif on simulation  Deploy website on push #2: Commit <a href="#">cb8aa77</a> pushed by <a href="#">MaylisD</a>	<a href="#">main</a>
 Create main.yml  Deploy website on push #1: Commit <a href="#">f78edfc</a> pushed by <a href="#">MaylisD</a>	<a href="#">main</a>

```
main.yml
1 on:
2   push:
3     branches:
4       - main
5
6   name: 🚀 Deploy website on push
7
8   jobs:
9     web-deploy:
10      name: 🌐 Deploy
11      runs-on: ubuntu-latest
12      steps:
13        - name: 🚚 Get latest code
14          uses: actions/checkout@v4
15
16        - name: 📁 Sync files
17          uses: SamKirkland/FTP-Deploy-Action@v4.3.5
18          with:
19            server: ${{ secrets.FTP_HOST }} # ex: ftp.example.com
20            username: ${{ secrets.FTP_USERNAME }} # ex: deploy@example.com
21            password: ${{ secrets.FTP_PASSWORD }}
22            protocol: ftps # ou 'ftp' selon ton
                hébergeur
23            port: 21 # ajuste si besoin
24            local-dir: ./ # dossier local à envoyer
25            server-dir: /wp-content/themes/mecars_theme/ #
                dossier cible sur le serveur
```

En parallèle, j'ai rédigé une documentation technique exhaustive incluant un README détaillé qui couvre l'ensemble du projet : architecture, installation, configuration des Custom Post Types, mesures de sécurité, et procédures de déploiement. Cette documentation a été conçue pour assurer la pérennité du projet et faciliter la maintenance future.

# DOSSIER PROFESSIONNEL (DP)

## 2. Précisez les moyens utilisés :

- **GitHub Actions** : Workflow automatisé avec fichier YAML de configuration,
- **Main** : Branche de référence et d'intégration,
- **Deployment** : Branche de travail principal – c'est sur cette branche que le développement actif était effectué,
- **Déploiement FTP** : Compte dédié O2Switch avec transfert sécurisé,
- **Secrets GitHub** : Stockage chiffré des identifiants FTP sensibles,
- **README structuré** :
  - **Présentation complète du projet et objectifs,**
  - **Technologies utilisées (WordPress, PHP, ACF etc.),**
  - **Arborescence détaillée du thème**
  - **Configuration des Custom Post Types,**
  - **Mesures de sécurité implémentées.**

## 3. Avec qui avez-vous travaillé ?

Développement de la documentation, configuration GitHub Actions, déploiement en autonomie.

## 4. Contexte

Nom de l'entreprise, organisme ou association ▶ **Meca'rs – Garage automobile**

Chantier, atelier, service ▶ Cliquez ici pour taper du texte.

Période d'exercice ▶ Du **25/08/2025** au **17/10/2025**

## 5. Informations complémentaires (facultatif)

La documentation technique complète que j'ai rédigée assure la transmission des connaissances et la maintenance à long terme du projet. Le README détaillé inclut toutes les informations nécessaires pour reprendre le projet : architecture technique, procédures d'installation, configuration des Custom Post Types, et mesures de sécurité. L'automatisation du déploiement via GitHub Actions a réduit de 80% le temps de mise en production et éliminé les erreurs humaines. La stratégie à deux branches avec merge contrôlé garantit la stabilité de l'environnement de production.

# DOSSIER PROFESSIONNEL (DP)

## Titres, diplômes, CQP, attestations de formation

*(facultatif)*

Intitulé	Autorité ou organisme	Date
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.

## DOSSIER PROFESSIONNEL (DP)

### Déclaration sur l'honneur

Je soussigné(e) [prénom et nom] *Delaval Maÿlis*.....  
déclare sur l'honneur que les renseignements fournis dans ce dossier sont exacts et que je  
suis l'auteur(e) des réalisations jointes.

Fait à *Troyes*..... le *25/11/2025*.....

pour faire valoir ce que de droit.

Signature :



# DOSSIER PROFESSIONNEL (DP)

## Documents illustrant la pratique professionnelle

*(facultatif)*

### Intitulé

Cliquez ici pour taper du texte.

## ANNEXES

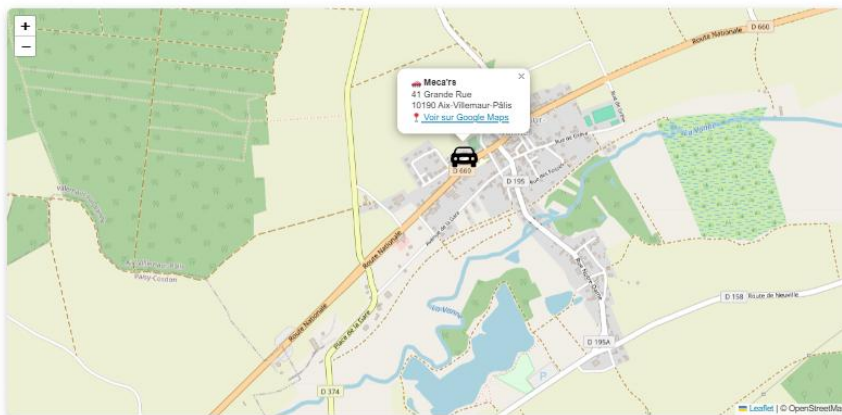
(Si le RC le prévoit)

### Annexe n°1 – Maquette Accueil : Desktop, Tablette, Mobile



## Annexe n°2 – Rendu map Leaflet.js et Code Animation Simulateur

### Où nous trouver



```
Animation Chargement Simulateur
1 // Animation avec durée contrôlée
2 startCarAnimation() {
3   const loading = document.getElementById('loading');
4   const result = document.getElementById('result');
5   const submitBtn = document.querySelector('.submit_btn');
6   const car = loading.querySelector('.car');
7
8   // Réinitialiser l'animation
9   car.style.animation = 'none';
10  void car.offsetWidth; // Forcer le reflow
11
12  // Animation unique de 2 secondes (durée minimale)
13  car.style.animation = 'moveCar 2s ease-in-out forwards';
14
15  // Afficher le loading, cacher le résultat
16  loading.style.display = 'block';
17  result.style.display = 'none';
18  result.classList.remove('show');
19
20  // Désactiver le bouton
21  submitBtn.disabled = true;
22  submitBtn.textContent = 'Estimation en cours...';
23
24  console.log('🚗 Animation démarrée (2s)');
25 }
```

## Annexe n°3 – Initialisation Base de Donnée pour le CRUD

```
Base de Données SQLite

1 -- Création de la base de données
2 CREATE DATABASE IF NOT EXISTS ma_bibliotheque;
3 USE ma_bibliotheque;
4
5 -- Table des auteurs
6 CREATE TABLE auteurs (
7     id BIGINT AUTO_INCREMENT PRIMARY KEY,
8     nom VARCHAR(100) NOT NULL,
9     prenom VARCHAR(100) NOT NULL
10 );
11
12 -- Table des éditeurs
13 CREATE TABLE editeurs (
14     id BIGINT AUTO_INCREMENT PRIMARY KEY,
15     nom VARCHAR(100) NOT NULL
16 );
17
18 -- Table des genres
19 CREATE TABLE genres (
20     id BIGINT AUTO_INCREMENT PRIMARY KEY,
21     nom VARCHAR(50) NOT NULL
22 );
```

```
Base de Données SQLite

1 -- Table principale des livres
2 CREATE TABLE livres (
3     id BIGINT AUTO_INCREMENT PRIMARY KEY,
4     titre VARCHAR(255) NOT NULL,
5     auteur_id BIGINT NOT NULL,
6     genre_id BIGINT NOT NULL,
7     editeur_id BIGINT NOT NULL,
8     isbn VARCHAR(20) UNIQUE,
9     pages INT,
10    date_publication DATE,
11    description TEXT,
12    statut ENUM('à lire', 'en cours', 'terminé', 'abandonné') DEFAULT 'à lire',
13    note DECIMAL(3,1),
14    date_ajout DATETIME DEFAULT CURRENT_TIMESTAMP,
15    date_lecture DATE,
16    commentaire TEXT,
17    couverture VARCHAR(255),
18    FOREIGN KEY (auteur_id) REFERENCES auteurs(id),
19    FOREIGN KEY (genre_id) REFERENCES genres(id),
20    FOREIGN KEY (editeur_id) REFERENCES editeurs(id)
21 );
```

## Annexe n°4 – Mesures de Sécurité pour les Avis

```
Mesures de Sécurité Avis Clients

1 function mecars_handle_submit_client_review() {
2     // Vérification du nonce de sécurité
3     if ( empty($_POST['submit_client_review_nonce']) || !
wp_verify_nonce($_POST['submit_client_review_nonce'], 'submit_client_review') ) {
4         $redirect = wp_get_referer() ? add_query_arg('review_status','invalid_nonce',
wp_get_referer()) : add_query_arg('review_status','invalid_nonce', home_url());
5         wp_safe_redirect( $redirect );
6         exit;
7     }
8     // Honeypot anti-spam (champ caché)
9     if ( ! empty($_POST['hp_name']) ) {
10        $redirect = wp_get_referer() ? wp_get_referer() : home_url();
11        wp_safe_redirect( $redirect );
12        exit;
13    }
14    // Récupération et sanitization des données du formulaire
15    $name = ! empty($_POST['name']) ? sanitize_text_field($_POST['name']) : '';
16    $email = ! empty($_POST['email']) ? sanitize_email($_POST['email']) : '';
17    $note = isset($_POST['note']) ? intval($_POST['note']) : 0;
18    $review = ! empty($_POST['your_review']) ?
sanitize_textarea_field($_POST['your_review']) : '';
19    $service_nom = ! empty($_POST['service']) ? sanitize_text_field($_POST['service']) :
'';
20    // === VALIDATION RENFORCÉE EMAIL ===
21    if (!empty($email) && !is_email($email)) {
22        $redirect = wp_get_referer() ? add_query_arg('review_status','invalid_email',
wp_get_referer()) : add_query_arg('review_status','invalid_email', home_url());
23        wp_safe_redirect($redirect);
24        exit;
25    }
26    // === PROTECTION CONTRE LES SOUMISSIONS MULTIPLES ===
27    if (!empty($email)) {
28        global $wpdb;
29        $recent_review = $wpdb->get_var($wpdb->prepare(
30            "SELECT COUNT(*) FROM {$wpdb->prefix}avis WHERE email = %s AND date_avis >
DATE_SUB(NOW(), INTERVAL 2 HOUR)",
31            $email
32        ));
33        if ($recent_review > 0) {
34            $redirect = wp_get_referer() ?
add_query_arg('review_status','too_many_submissions', wp_get_referer()) :
add_query_arg('review_status','too_many_submissions', home_url());
35            wp_safe_redirect($redirect);
36            exit;
37        }
38    }
39    // Validation des champs obligatoires
40    if ( empty($name) || empty($review) ) {
41        $redirect = wp_get_referer() ? add_query_arg('review_status','missing_fields',
wp_get_referer()) : add_query_arg('review_status','missing_fields', home_url());
42        wp_safe_redirect( $redirect );
43        exit;
44    }
45 }
```