

Déploiement de la plateforme AUTO-RENT

1. Pré-requis

Avant de déployer la plateforme AUTO-RENT, vous devez disposer de :

- Un **hébergeur web** (comme OVH, Hostinger, ou un VPS).
 - Un **nom de domaine** (ex : `auto-rent.com`).
 - **Node.js** et **npm** installés sur votre serveur (pour le backend Express).
 - Une **base de données MySQL** (accessible depuis l'hébergeur ou installée sur le serveur).
 - Un client FTP comme **FileZilla** (si vous utilisez un hébergement mutualisé).
-

2. Déploiement du Backend (Express.js)

1. Transférez les fichiers du backend sur le serveur (via **SSH** ou **FTP**).
2. Modifiez le fichier `.env` avec les informations de connexion à la base de données :

```
ini
CopierModifier
DB_HOST=...
DB_USER=...
DB_PASSWORD=...
DB_NAME=auto_rent
```

3. Connectez-vous à votre serveur via SSH et installez les dépendances :

```
bash
CopierModifier
npm install
```

4. Lancez le serveur avec :

```
bash
CopierModifier
node server.js
```

Ou, pour un déploiement stable :

```
bash
CopierModifier
npm install pm2 -g
pm2 start server.js
pm2 startup
pm2 save
```

5. Vérifiez que le **port utilisé** (ex : **3000**) est bien ouvert sur le pare-feu du serveur.
-

3. Déploiement du Frontend (HTML/CSS/JS)

1. Transférez les fichiers HTML/CSS/JS dans le dossier `public/` (ou le dossier racine selon votre configuration).
2. Si vous utilisez Express pour servir le frontend, ajoutez dans `server.js` :

```
js
CopierModifier
app.use(express.static('public'));
```

3. Vérifiez que les appels **fetch()** ou **AJAX** utilisent le bon **nom de domaine ou IP** de l'API backend.
-

4. Connexion à la Base de Données

1. Créez une base de données nommée `auto_rent` via **phpMyAdmin** ou en ligne de commande.
 2. Importez le fichier SQL (ex: `auto_rent.sql`) contenant les tables nécessaires.
 3. Vérifiez que les identifiants dans le `.env` correspondent bien à la base de données active.
-

5. Sécurité et Configuration Finale

- **Activez HTTPS via Let's Encrypt** ou un certificat SSL fourni par votre hébergeur.
 - Cachez les fichiers sensibles : `.env`, `config.js`, etc.
 - Ne donnez pas de permissions en écriture aux fichiers critiques.
 - Assurez-vous que tous les liens et formulaires fonctionnent correctement en production.
-

6. Sauvegarde et Maintenance

- Configurez des **sauvegardes automatiques** de la base de données.
- Mettez régulièrement à jour vos dépendances via :

```
bash
CopierModifier
npm update
```

- Utilisez un outil de monitoring comme **UptimeRobot** pour surveiller la disponibilité du site.
- Gardez une copie locale à jour du projet.