

Implémentation sur le site Patron creator

Introduction :

I : Téléchargement et Premier Démarrage du Serveur patron-creator

II : Tutoriel d'Implémentation

III : Points à retenir

Prérequis : Installer Node.js pour Faire Tourner le Serveur

Pour implémenter un type de patron dans le site, il est essentiel de bien comprendre son fonctionnement. J'ai préparé un exemple de test pour illustrer ce processus. Un patron est constitué de trois fichiers principaux :

1. **editor.json** : Ce fichier initialise les valeurs nécessaires lors de la création d'un rendez-vous. Il sert à configurer les paramètres de base qui seront utilisés pour le patron.
2. **element.json** : Ce fichier contient les éléments HTML pour les champs de saisie (variables en temps réel) ainsi que les éléments graphiques en SVG, comme les lignes et les courbes.
3. **[Nom du patron].js** : Ce fichier principal, nommé selon le type de patronage, contient toute la logique du patron. Il inclut une fonction clé qui est appelée à chaque modification des valeurs des champs de saisie HTML. Cette fonction permet de mettre à jour les éléments en temps réel, assurant une interaction dynamique avec le patron.

En plus de ces trois fichiers, il y a également un fichier `data.json`, qui fait office de base de données du site. Ce fichier centralise les données de tous les clients et joue un rôle crucial dans le bon fonctionnement de l'application

I : Téléchargement et Premier Démarrage du Serveur patron-creator

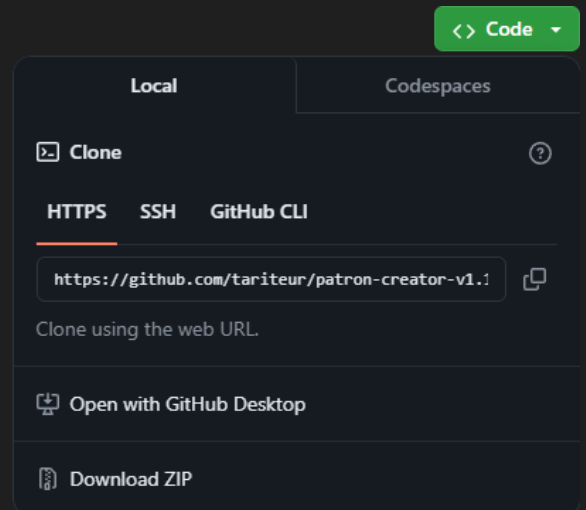
1. Télécharger le Projet

1. **Accéder au dépôt** : Ouvrez votre navigateur web et allez à l'adresse suivante :

<https://github.com/tariteur/patron-creator-v1.2>.

2. Télécharger le ZIP :

- Cliquez sur le bouton vert **Code** (généralement situé en haut à droite de la page).
- Sélectionnez **Download ZIP** dans le menu déroulant.
- Le fichier ZIP contenant le projet sera téléchargé sur votre ordinateur.

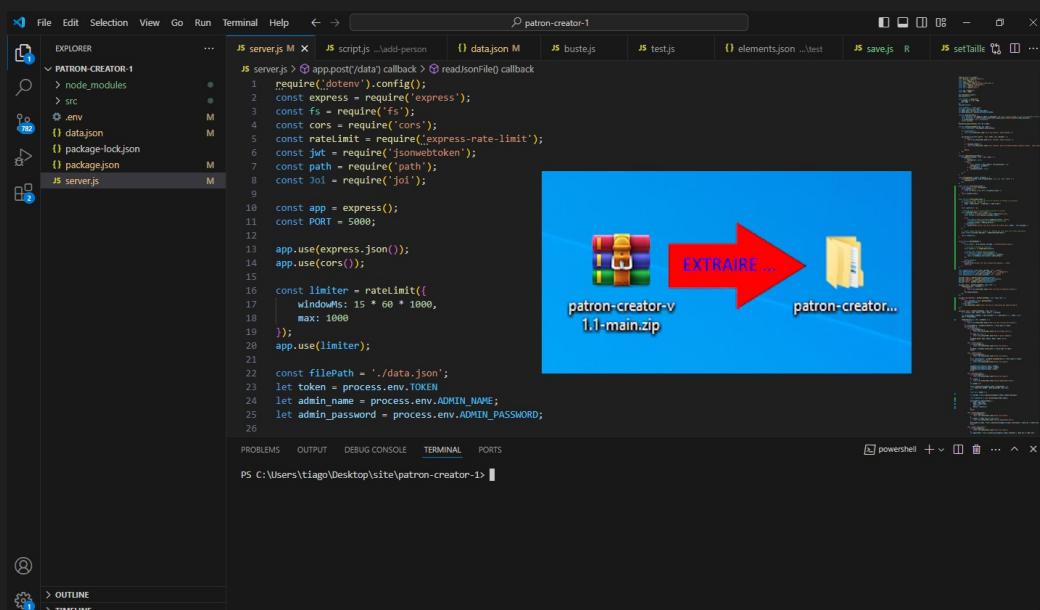


3. Placer le ZIP :

- Déplacez le fichier ZIP téléchargé dans le dossier où vous souhaitez travailler, ou sur le bureau.

4. Extraire le ZIP :

- Faites un clic droit sur le fichier ZIP et choisissez **Extraire tout** (ou utilisez un outil de décompression comme WinRAR ou 7-Zip).
- Extrayez le contenu du fichier ZIP dans un dossier.



2. Configurer le Projet

1. Ouvrir le Projet dans Visual Studio Code :

- Lancez **Visual Studio Code**.
- Cliquez sur **File > Open Folder** (ou utilisez le raccourci **Ctrl+K Ctrl+O**).
- Naviguez jusqu'au dossier où vous avez extrait le contenu du ZIP, puis sélectionnez-le et cliquez sur **Ouvrir**.

2. Ouvrir le Terminal :

- Dans Visual Studio Code, ouvrez un terminal intégré en allant dans **Terminal > New Terminal**.

3. Installer les Dépendances :

- Dans le terminal intégré, assurez-vous que vous êtes dans le dossier du projet (vérifiez que le chemin affiché est celui du dossier extrait).
- Exécutez la commande suivante pour installer toutes les dépendances nécessaires :

```
npm install
```

4. Démarrer le Serveur :

- Une fois les dépendances installées, démarrez le serveur en exécutant :

```
npm start
```

- Cette commande lancera le serveur **patron-creator** en local. Vous devriez voir des messages dans le terminal indiquant que le serveur est en cours d'exécution et à quel port il écoute.

3. Accéder au Serveur Local :

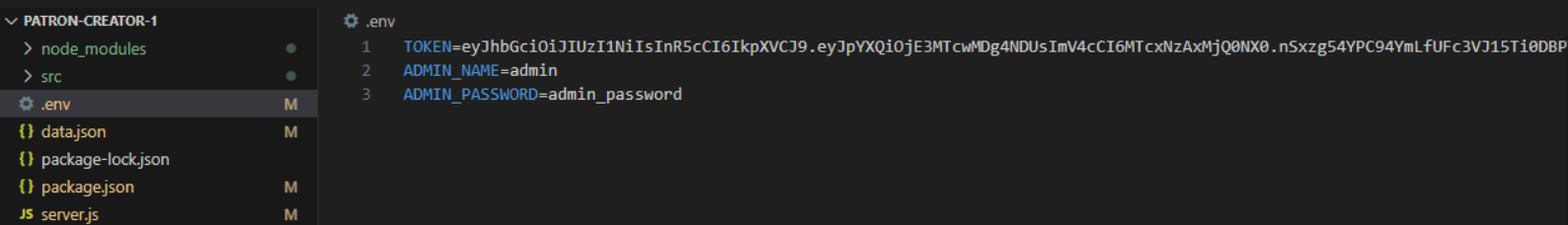
- Ouvrez un navigateur web.
- Accédez à l'adresse suivante : <http://localhost:5000>.
- À l'écran de connexion, utilisez les identifiants par défaut pour vous connecter :

Nom d'utilisateur : admin

Mot de passe : admin_password

4. Modifier les Identifiants par Défaut:

- Les identifiants par défaut peuvent être modifiés dans le fichier `.env` situé dans le répertoire racine du projet.
- Ouvrez le fichier `.env` et recherchez les lignes correspondantes aux identifiants.
- Modifiez les valeurs pour mettre à jour le nom d'utilisateur et le mot de passe comme vous le souhaitez :
- Enregistrez les modifications et redémarrez le serveur en exécutant à nouveau `npm start` pour appliquer les changements.



II : Tutoriel d'Implémentation

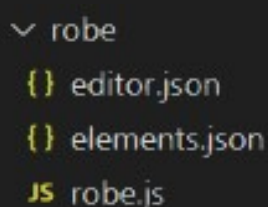
Pour ajouter un nouveau patron à votre application, suivez ces étapes :

1. Créer un Dossier pour le Patron :

- Accédez au répertoire racine de votre projet : `/src/editor/patron-logic/`.
- Créez un nouveau dossier avec le nom du patron que vous souhaitez ajouter. Par exemple, si vous ajoutez un patron pour une robe, nommez le dossier `/src/editor/patron-logic/robe`.

2. Créer les Fichiers Principaux :

- Dans le dossier nouvellement créé, ajoutez les trois fichiers suivants :
 - **editor.json** : Ce fichier contiendra toutes les valeurs par défaut nécessaires lors de la création d'un rendez-vous pour ce patron.
 - **elements.json** : Ce fichier définira les éléments HTML et SVG associés au patron.
 - **[Nom du patron].js** : Ce fichier principal, nommé en fonction du type de patron (par exemple `robe.js`), contiendra toute la logique spécifique au patron et une seule fonction nommée `.`



3. Ajouter les Valeurs dans `editor.json`:

- Ouvrez le fichier `editor.json`.
- Ajoutez-y toutes les valeurs par défaut nécessaires pour le patron. Ces valeurs seront utilisées lors de la création d'un rendez-vous et serviront de base à la configuration du patron.

```
{  
  "longueurRobe": 40,  
  "largeurRobe": 40  
}
```

4. Gestion Automatique des Éléments HTML et SVG dans `elements.json`:

- **Aucun besoin d'ajouter manuellement les éléments HTML et SVG.** Les éléments définis dans `elements.json` seront automatiquement placés sur la page lors du chargement. Cela inclut les champs de saisie pour les mesures, ainsi que les formes et courbes nécessaires à l'affichage du patron.

```
{  
  "html": [  
    {"label": "Test", "type": "number", "id": "test", "name": "test", "min": 0, "max": 100, "step": 0.1, "value": 40, "class": "EventListeners form-control"},  
  ],  
  "svg": {  
    "lines": [  
      {"id": "ligne1", "x1": "20", "y1": "0", "x2": "20", "y2": "10", "style": "stroke: black; stroke-width: 10"},  
    ]  
  }  
}
```

5. Copier le Modèle depuis `test` :

- Pour faciliter la création de ces fichiers, copiez le modèle de base depuis le fichier `/src/editor/patron-logic/test`.
- Adaptez le contenu de chaque fichier en fonction des besoins spécifiques de votre nouveau patron.

6. Démarrer le serveur et tester :

- démarrez le serveur en exécutant :
`npm start`
- Accédez à l'adresse suivante pour voir l'application en cours d'exécution :
<http://localhost:5000>.

III : Points à retenir

- Assurez-vous que les noms des fichiers sont corrects et respectent la convention de nommage.
- Node.js doit être installé pour faire fonctionner le serveur.
- Les fichiers principaux sont `editor.json`, `elements.json`, et `[Nom du patron].js`.
- Le fichier `data.json` est crucial pour la gestion des données.

Note : Ne pas prêter attention aux autres fichiers présents dans le projet ; ils sont nécessaires au bon fonctionnement du site mais ne doivent pas vous perturber.