

# SuperCalendrier - Instructions de démarrage

Ce document contient les instructions pour initialiser et démarrer le projet SuperCalendrier, une application de calendrier desktop basée sur Electron.

## Structure du projet

Nous avons créé la structure de base suivante :

```
/super-calendrier/
├─ package.json          # Configuration du projet
├─ main.js               # Point d'entrée principal Electron
├─ preload.js            # Script de préchargement
├─ app/                  # Application web
│   ├─ index.html        # Point d'entrée HTML
│   ├─ css/              # Styles CSS
│   │   └─ style.css      # Feuille de style principale
│   ├─ js/               # JavaScript
│   │   ├─ app.js         # Point d'entrée JavaScript
│   │   ├─ calendar-manager.js # Gestion du calendrier
│   │   ├─ event-manager.js # Gestion des événements
│   │   ├─ category-manager.js # Gestion des catégories
│   │   ├─ data-manager.js # Gestion des données
│   │   ├─ ui-manager.js  # Gestion de l'interface utilisateur
│   │   ├─ theme-manager.js # Gestion du thème
│   │   ├─ notification-manager.js # Gestion des notifications
│   │   ├─ print-manager.js # Gestion de l'impression
│   │   └─ utils/         # Utilitaires
│   │       ├─ date-utils.js # Manipulation des dates
│   │       └─ uuid.js     # Génération d'identifiants uniques
│   └─ assets/           # Images, icônes, etc.
├─ build/                # Ressources pour le build
├─ icons/                # Icônes de l'application
├─ installers/           # Scripts d'installation
└─ dist/                 # Dossier de distribution
```

## Initialisation du projet

1. Créez un nouveau dossier pour le projet :

```
bash
```

```
mkdir super-calendrier
```

```
cd super-calendrier
```

2. Initialisez un nouveau projet Node.js :

```
bash  
npm init -y
```

3. Installez les dépendances nécessaires :

```
bash  
npm install electron electron-store electron-updater electron-log ical-generator dayjs uuid  
npm install --save-dev electron-builder nodemon
```

4. Copiez les fichiers créés dans la structure appropriée :

- `package.json` à la racine du projet
- `main.js` à la racine du projet
- `preload.js` à la racine du projet
- Créez le dossier `app` et ses sous-dossiers
- Copiez `index.html` dans le dossier `app`
- Copiez `style.css` dans le dossier `app/css`
- Copiez tous les fichiers JavaScript dans les dossiers appropriés

5. Créez le dossier pour les icônes :

```
bash  
mkdir -p build/icons
```

## Exécution du projet

1. Pour démarrer l'application en mode développement :

```
bash  
npm run dev
```

2. Pour surveiller les modifications et redémarrer automatiquement :

```
bash  
npm run watch
```

3. Pour construire l'application pour la distribution :

bash

*# Pour toutes Les plateformes*

`npm run build`

*# Pour Windows uniquement*

`npm run build:win`

*# Pour macOS uniquement*

`npm run build:mac`

*# Pour Linux uniquement*

`npm run build:linux`

## Fonctionnalités implémentées

- **Visualisation du calendrier** : Vues annuelle, mensuelle, hebdomadaire et quotidienne
- **Gestion des événements** : Création, modification, suppression d'événements avec récurrence
- **Système de catégories** : Catégories personnalisables avec émojis et couleurs
- **Thème clair/sombre** : Basculement entre thèmes clair et sombre
- **Importation/Exportation** : Sauvegarde et restauration des données au format JSON
- **Impression** : Options d'impression adaptées à différents formats de papier

## Prochaines étapes

1. **Synchronisation avec des services externes** : Intégration avec Google Calendar, Outlook, etc.
2. **Format iCal** : Implémentation de l'importation/exportation au format iCal (.ics)
3. **Optimisations de performance** : Amélioration du rendu pour les calendriers avec beaucoup d'événements
4. **Notifications système** : Rappels pour les événements à venir
5. **Statistiques et analyses** : Visualisation de la répartition des événements par catégorie

## Notes de développement

- L'application utilise le stockage local d'Electron pour sauvegarder les données
- Les mises à jour automatiques sont configurées via electron-updater
- Le système de préférences permet de personnaliser l'expérience utilisateur
- L'architecture modulaire facilite l'extension des fonctionnalités

---

Pour toute question ou problème, n'hésitez pas à me contacter. Bonne utilisation de SuperCalendrier !