

Installation d'Electron et configuration de l'application

Étape 1 : Prérequis

Assurez-vous d'avoir Node.js installé sur votre machine. Vous pouvez vérifier cela en ouvrant une fenêtre de terminal et en exécutant la commande suivante :

```
node -v
```

Si Node.js n'est pas installé, rendez-vous sur le site officiel de Node.js (<https://nodejs.org>) et suivez les instructions d'installation pour votre système d'exploitation.

Étape 2 : Initialisation du projet

Dans le répertoire de votre projet, ouvrez une fenêtre de terminal et exécutez la commande suivante pour initialiser un nouveau projet Node.js :

```
npm init -y
```

Cela créera un fichier package.json qui contiendra les dépendances de votre projet.

Étape 3 : Installation d'Electron

Dans le répertoire de votre projet, exécutez la commande suivante pour installer Electron :

```
npm install electron
```

Cela téléchargera et installera Electron dans le dossier node_modules de votre projet.

Étape 4 : Configuration du fichier package.json

Ouvrez le fichier package.json de votre projet et ajoutez la propriété main avec le nom du fichier javascript qui contient le code js de la fenêtre Electron, dans notre cas, "main.js". Cela indique à Electron quel fichier JavaScript doit être exécuté en tant que point d'entrée de votre application.

```
{
  "name": "esp32-electron",
  "version": "1.0.0",
  "description": "Work with esp32 & electron & Websocket",
  "main": "main.js",
  "scripts": {
    "start": "electron . --disable-gpu"
  },
  "repository": "https://github.com/atmoner/esp32-electron",
  "keywords": [
    "esp32",
    "Websocket",
    "electron",
    "tutorial",
    "demo"
  ],
  "author": "atmoner",
  "license": "CC0-1.0",
  "devDependencies": {
    "electron": "^7.1.7"
  },
  "dependencies": {
    "chart.js": "^2.9.3",
    "electron-reload": "^1.5.0",
    "jquery": "^3.4.1",
    "ws": "^7.2.3"
  }
}
```

Étape 5 : Création du fichier main.js

Dans le répertoire de votre projet, créez un fichier `main.js` qui servira de point d'entrée de votre application Electron. Vous pouvez le créer avec le contenu minimal suivant pour démarrer :

```
const { app, BrowserWindow } = require('electron')

function createWindow() {
  const win = new BrowserWindow({
    width: 800,
    height: 600,
    webPreferences: {
      nodeIntegration: true
    }
  })

  win.loadFile('index.html')
}

app.whenReady().then(createWindow)
```

Ce code crée une fenêtre de navigateur Electron avec une taille de 800x600 pixels et charge le fichier `index.html`.

Étape 6 : Création du fichier index.html

Dans le répertoire de votre projet, créez un fichier `index.html` qui servira de page principale de votre application. Vous pouvez y ajouter votre contenu HTML, CSS et JavaScript personnalisé.

Lancement de l'application

Étape 1 : Lancer l'application en mode développement

Pour lancer votre application Electron en mode développement, ouvrez une fenêtre de terminal dans le répertoire de votre projet et exécutez la commande suivante :

```
npm start
```

Cela exécutera le fichier `main.js` et ouvrira votre application Electron.

Étape 2 : Création d'un exécutable pour votre application

Une fois que vous êtes satisfait de votre application et que vous souhaitez créer un exécutable pour la distribution, vous pouvez utiliser des outils comme `electron-packager` ou `electron-builder` pour empaqueter votre application en un exécutable adapté à votre système d'exploitation.