

ELECTRON



Sommaire

- Présentation d'Electron
- Avantages
- Inconvénients
- Initialisation d'un projet

Présentation d'Electron



- Framework pour simplifier la réalisation de l'interface graphique d'un logiciel en utilisant des langages du web
- Dérivé d'Atom, développé par Github
- Backend en node.js
- Frontend basé sur les outils de Chromium
- Utilisé par Skype, Discord, Visual Studio Code, ...
- Compatible Mac, Windows et Linux

Avantages

- Notifications et menus natifs
- Mises à jour automatiques
- Rapport de plantage
- Débogage et analyse de performance

Inconvénients

- Chromium consomme beaucoup
- Electron est lourd
- Déconseillé si les fonctions back end sont lourdes
- Déconseillé de charger la page avec trop de JavaScript ou de framework comme jquery, angular, vue

Initialisation d'un projet

`npm init`

Modifier le package.json

```
"scripts": { "start": "electron ."
```

```
"main": "main.js"
```

`npm install --save-dev electron`

structure d'un application electron de base :

```
your-app/  
├── package.json  
├── main.js  
└── index.html
```

main.js

Importer le module electron et définir des raccourcis vers des objets :

```
const electron = require('electron');
```

```
const app = electron.app;
```

```
const BrowserWindow = electron.BrowserWindow;
```

```
const url = require('url');
```

Créer une fenêtre

```
app.on('ready', function(){  
  
    mainWindow = new BrowserWindow({  
  
        mainWindow.loadUrl(url.format({  
  
            pathname: path.join(__dirname, 'index.html'),  
  
            protocol: 'file:',  
  
            slashes: true  
  
        }));  
  
});
```


Démonstration



ELECTRON

