

NODEJS – ELECTRON

YOANN DUC

DEV BACK - KEYRUS

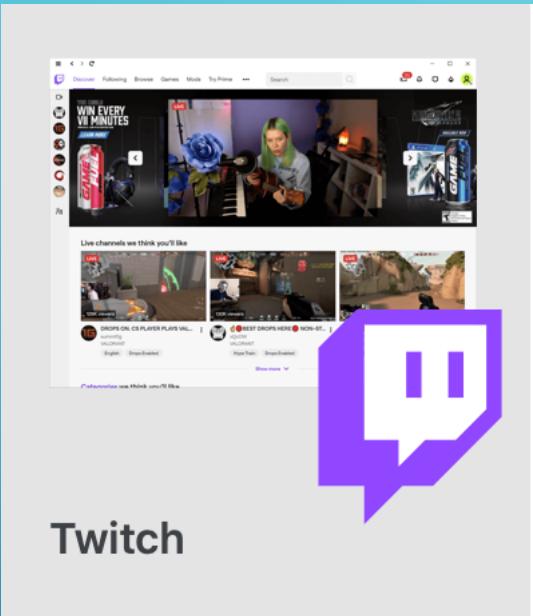
ELECTRON – INTRODUCTION

- Crée par Github
- Première version stable sortie le 15 July 2013 (Wikipédia)
- Développement d'applications bureaux multiplateformes
- Basé sur Chromium et Nodejs

ELECTRON - UTILISATIONS



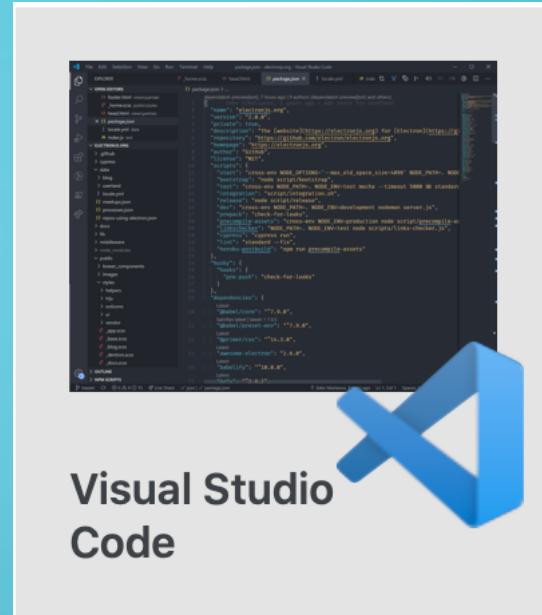
WhatsApp



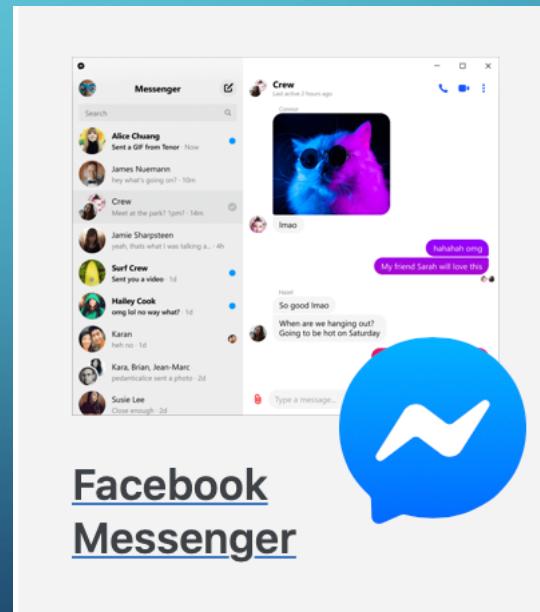
Twitch

Mais aussi (<https://www.electronjs.org/apps>):

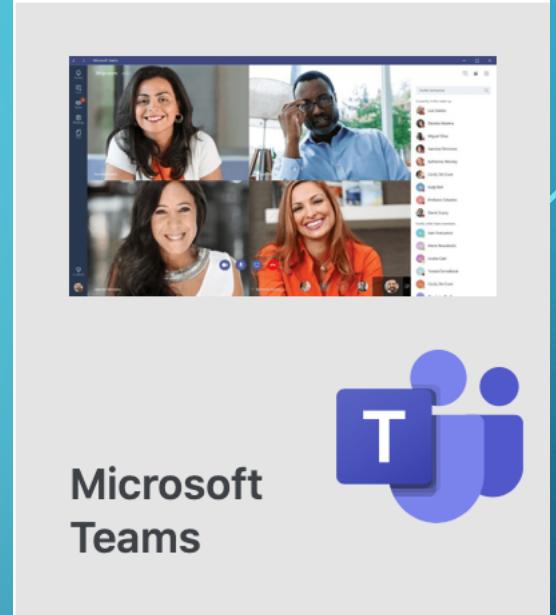
- Discord
- Atom
- Station
- MongoDB Compass
- Insomnia...



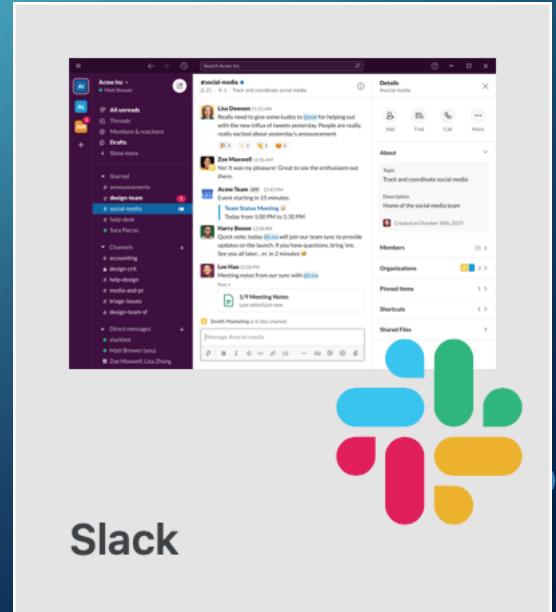
Visual Studio
Code



Facebook
Messenger



Microsoft
Teams



Slack

ELECTRON – POURQUOI

- Initialement créé pour Atom (son premier nom était d'ailleurs Atom Shell)
- Possibilité de rendre des app web en tant qu'application desktop
- Pas de nouvelles compétences pour intégrer pages (Javascript)
- Possibilité de transformer application web en application bureau
- Accès API Nodejs directement dans le Brower (accès OS, au filesystem...)

ELECTRON - FONCTIONNEMENT

- Application Nodejs classique (`package.json`, `node_modules...`)
- Multi-process:
 - **Main process:** 1 seul main process par instance d'electron. Process qui gère la fenêtre et le fonctionnement de l'app en elle même. Process à lancer pour démarrer l'app.
 - **Rendered process(es):** multiple process (au moins 1). Responsables de la partie rendu (affichage dans la fenêtre gérée par le main process), de la partie navigateur. C'est dans ces process que l'on a accès au DOM.

ELECTRON – AVANTAGES

- Courbe d'apprentissage peu élevée (si on connaît déjà Nodejs/Javascript)
- Équipe web « classique » peut intervenir, donc un intégrateur sans compétences en développement Nodejs peut changer l'UI
- Permet un gain de temps considérable dans le cadre d'une app cross plateforme (une seule base de code)
- Support en natif (Nodejs) de l'asynchrone et des événements

ELECTRON - LIMITES

- Littéralement page web dans browser; fragile face attaques web classiques
- Partage failles Nodejs
- (Très) gourmand en RAM et en ressources en général (Chromium)
- Peut être beaucoup plus lourd qu'une application native équivalente (dépendances, application web, server Nodejs intégrés au build)

ELECTRON – DÉMARRER

- <https://github.com/electron/electron>
- <https://github.com/electron/electron-api-demos-building>
- <https://github.com/electron/simple-samples>
- <https://github.com/hokein/electron-sample-apps>

ELECTRON - PRATIQUE

- Application de gestion de TODO listes et items
- Interfaçage avec une API REST existante
- Sauvegarde en local pour reprise au retour du réseau
- Croix minimise l'application mais ne la quitte pas (discord, spotify...)
- Une seule instance de l'application
- Ressources disponibles sur <https://github.com/yoannduc/todo-ressources>

ELECTRON – HELP & TIPS

Vous pouvez utiliser VueCLI (`yarn global @vue/cli`) pour générer une APP Vue et ensuite créer une app electron avec la CLI de Vue (voir slide suivante)

ELECTRON – HELP & TIPS

```
yoann@MBP-Yoann ~/Dev/YOANN/playground/tmp
$ ./node_modules/.bin/vue create projectname
```

```
Vue CLI v4.5.6
? Please pick a preset:
  Default ([Vue 2] babel, eslint)
  Default (Vue 3 Preview) ([Vue 3] babel, eslint)
❯ Manually select features
```

Vue CLI v4.5.6

- ? Please pick a preset: Manually select features
- ? Check the features needed for your project:
 - Choose Vue version
 - Babel
 - TypeScript
 - Progressive Web App (PWA) Support
 - Router
 - Vuex
 - CSS Pre-processors
 - Linter / Formatter
 - Unit Testing
 - E2E Testing

```
yoann@MBP-Yoann ~/Dev/YOANN/playground/tmp/projectname <master>
$ ./node_modules/.bin/vue add electron-builder
```

ELECTRON – HELP & TIPS



Don't use **arrow functions** on an options property or callback, such as

```
created: () => console.log(this.a) or
```

```
vm.$watch('a', newValue => this.myMethod())
```

. Since an arrow function doesn't have
a **this**, **this** will be treated as any other variable and lexically looked up through parent
scopes until found, often resulting in errors such as

```
Uncaught TypeError: Cannot read property of undefined or
```

```
Uncaught TypeError: this.myMethod is not a function .
```