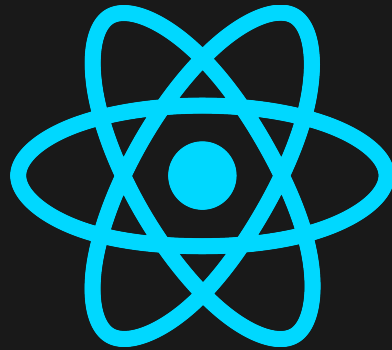


# DÉVELOPPEMENT WEB AVANCÉ



ReactJS

Année 2025-2026

# L'ENSEIGNANT

Clément Dandrieux

[clement.dandrieux@proton.me](mailto:clement.dandrieux@proton.me)

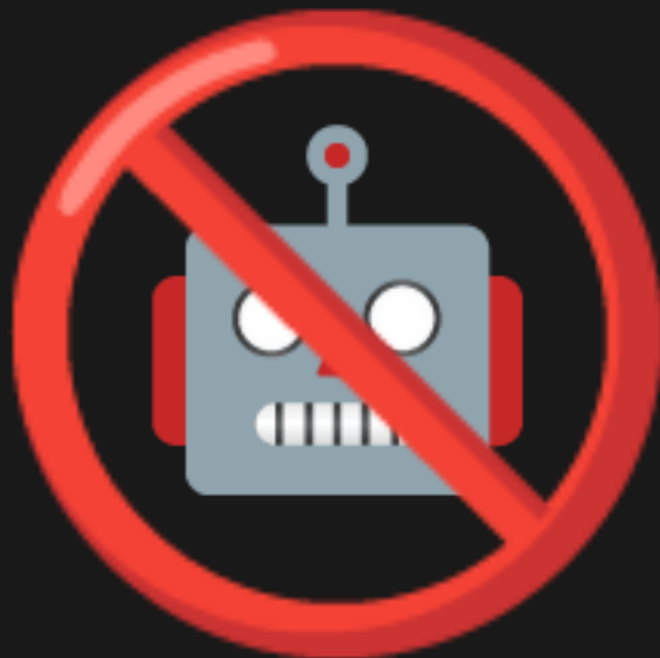


# QUESTIONS PÉDAGOGIQUES

- Sur le canal Discord dédié
- Par Discord en privé
- Par mail

# ORGANISATION ET EVALUATION

- Évaluation continue
- Une évaluation théorique
- Hackathon



# LLM INTERDITS

- Pendant les TPs
- Pendant les évaluations
- Sur les projets de TPs React

# POURQUOI ?

- Cours pour comprendre les concepts de base
- Très forts avec React sur des projets basiques
- Beaucoup moins forts sur des projets de taille réelle

# INTRODUCTION

- Prérequis et ressources
- React, qu'est-ce que c'est ?



# PRÉREQUIS

- Les langages HTML, CSS et Javascript
- Fonctionnalités d'ES6 (arrow functions, classes, let, const...)

# RESSOURCES

- Le cours
- Le MDN, la bible du Javascript
- La doc officielle de React (en anglais de préférence...)

# REACT, QU'EST-CE QUE C'EST ?

React est une *bibliothèque* Javascript **déclarative**,  
**efficace** et **flexible** conçue pour faire *des interfaces*  
*utilisateur*

- Développé par Facebook
- Depuis 2013
- Open-source (MIT)
- Utilisé par Facebook, Netflix, Slack...

# LES COMPOSANTS

React permet de créer des interfaces utilisateur complexes depuis de petits morceaux de code isolés appelés **composants**.

```
const Hello = () => {  
  return <section>  
    <h1>Salut</h1>  
  </section>  
}
```

# LA COMPILATION

La syntaxe `<div></div>` est transformée au moment de la compilation en `React.createElement('div')`

L'exemple précédent est équivalent à ceci :

```
const Hello = () => {  
  return React.createElement("section", null,  
    React.createElement("h1", null, "Salut")  
  )  
}
```

# PASSER DES DONNÉES AVEC LES *PROPS*

```
const Hello = (props) => {  
  return <section>  
    <h1>Salut {props.name} !</h1>  
  </section>  
}  
  
const HelloAll = () => {  
  return <main>  
    <Hello name="tout le monde" />  
  </main>  
}
```

# MAINTENIR UN ÉTAT AVEC LE *STATE*

```
const Counter = () => {  
  const [count, setCount] = useState(0)  
  return <main>  
    <p>Count: {count}</p>  
    <button onClick={() => setCount(count+1)}>  
      +1  
    </button>  
  </main>  
}
```



# A VOUS D'ESSAYER

Faut bien mettre un peu les mains dedans...

[Aller vers les exercices](#)

# EXERCICE 1 : MODIFIEZ HELLOALL

Modifiez le composant *HelloAll* pour utiliser plusieurs fois le composant *Hello* avec des *prop* "name" différents.

## EXERCICE 2 :

Faites en sorte que :

- le bouton ajoute 2 au compteur
- le compteur n'affiche que les nombres impairs
- une alerte soit affichée au clic

# OBJECTIF DU TP1

Faire une application de gestion de tâches

[Voir le résultat](#)