

# TIW 8

## Technologies Web synchrones et multi-dispositifs

---

Introduction au cours

<https://aurelient.github.io/tiw8/2019/>

# Plan

---

- ▶ **Introduction au cours**
- ▶ L'informatique Ubiquitaire
- ▶ Rappels Stack Javascript

# Technologies Web synchrones et multi-dispositifs

---

## Suite de MIF13

### Objectifs :

- ▶ Aller au delà de *l'état de l'art* (applis Web desktop ou mobiles).
- ▶ Concevoir et réaliser des applications multi-dispositifs
  - ▶ Qui fonctionnent sur plusieurs dispositifs à la fois (tablette, télé, ordi, mobile, etc.)
  - ▶ Qui sont collaboratives temps réel (visio, éditeur de documents type google doc)
  - ▶ Qui sont embarquées : IoT, utilisation de capteurs.
- ▶ Réfléchir aux enjeux de l'informatique ambiante : temps réel, synchronisation, consommation, performance, vie privée, etc.

# Contenu de l'UE

---

Outils de programmation en JS

Programmation réactive (React x Redux)

Multi-dispositif

Nouvelles modalités de présentation et d'interaction

Temps réel et WebRTC (web p2p)

Web of Things

Performance

<https://aurelient.github.io/tiw8/2020/>

# Organisation de l'UE

---

7 CM de 1h30

10 séances de TP de 1h30 ou 3h

# Évaluation

---

3 grands TPs	(66%)	
▶ Mise en place Stack	(1 TP)	10%
▶ React et interaction multi-dispositif	(4 TP)	40%
▶ Techno synchrones et p2p	(2 TP)	20%
▶ Web of Things	(3 TP)	30%
Un examen (février)	(33%)	
▶ Réactivité		
▶ Algos de synchronisation, de reconnaissance de gestes...		
▶ Web of Things		
▶ Histoire, enjeux et impacts de l'informatique ubiquitaire...		

# Modalités d'évaluation

---

## Rendu des TPs

- ▶ Vous poussez votre TP sur la forge
- ▶ Vous ajoutez vos encadrants comme reporters du projet
- ▶ Vous déployez sur Heroku
- ▶ Vous remplissez les champs Tomuss

Les rendus de TP identiques seront notés 0 pour TOUS les groupes incriminés

# TP : Problèmes de rendu à éviter

---

- ▶ Impossible de cloner le projet
  - ▶ Erreur dans l'identifiant du projet
  - ▶ Nous n'avons pas les droits d'accès au projet
- ▶ Manque de doc
  - ▶ Placez tous les fichiers dans le repo du projet
  - ▶ README.md obligatoire et clair sur les étapes à suivre pour faire tourner votre code
- ▶ Notre environnement est différent du votre
  - ▶ URLs absolues dans le code
  - ▶ Ne mettez pas le contexte en dur dans votre code
  - ▶ Déployez votre projet sur une autre machine, et testez toutes les fonctionnalités.
  - ▶ Testez sur plusieurs navigateurs
  - ▶ "Ça marche chez moi" n'est pas un bon argument



# A l'extérieur

---

## LyonTechHub

- ▶ Calendrier d'évènements
- ▶ Slack
- ▶ ...

## Lyon JS

- ▶ Pour améliorer vos connaissances
- ▶ Pour étendre votre réseau