

TIW 8

Technologies Web synchrones et multi-dispositifs

Introduction au cours

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

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)
 - ▶ Centralisées ou distribuées
- ▶ Réfléchir aux enjeux de l'informatique ambiante : temps réel, synchronisation, consommation, performance, vie privée, etc.

Contenu de l'UE

Outils de programmation en JS / TypeScript

Programmation réactive (React / Redux)

Multi-dispositif

Nouvelles modalités de présentation et d'interaction

Temps réel et WebRTC (web p2p)

Algorithmes et données distribuées

<https://aurelient.github.io/tiw8/2025/>

Organisation de l'UE

6 CM de 1h30

9 séances de TP de 1h30 ou 3h

Évaluation

2 grands TPs	(66%)
1. Mise en place Stack	(aujourd'hui) 10%
2. React et interaction multi-dispositif (1 bloc)	40%
3. Techno synchrones et p2p	(projet) 50%
Un examen	(33%)

Modalités d'évaluation

Rendu des TPs

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

L'entraide oui, le clone de projets non.

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** - avec **4 sections**: Etat actuel de l'app, Dépendances (version de node, packages, ...), Setup (commandes), Utilisation (commandes à lancer).
- ▶ 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
 - ▶ "Ca marche chez moi" n'est pas un bon argument

LLM

Ne pas utiliser sauf là où indiqué dans les TP 1 et 2.

Libre usage à partir du projet.

Comprenez chaque ligne que vous intégrez à votre code.

- ▶ Pour l'informatique distribuée, les bugs sont subtils, une compréhension précise de ce que vous codez est nécessaire

Je ne suis pas le SAV de Claude ou Gemini.

A l'extérieur

LyonTechHub

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

Lyon JS

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