



# PROJET FINAL W2

Docker, PHP Natif, POO, MVC, API Rest



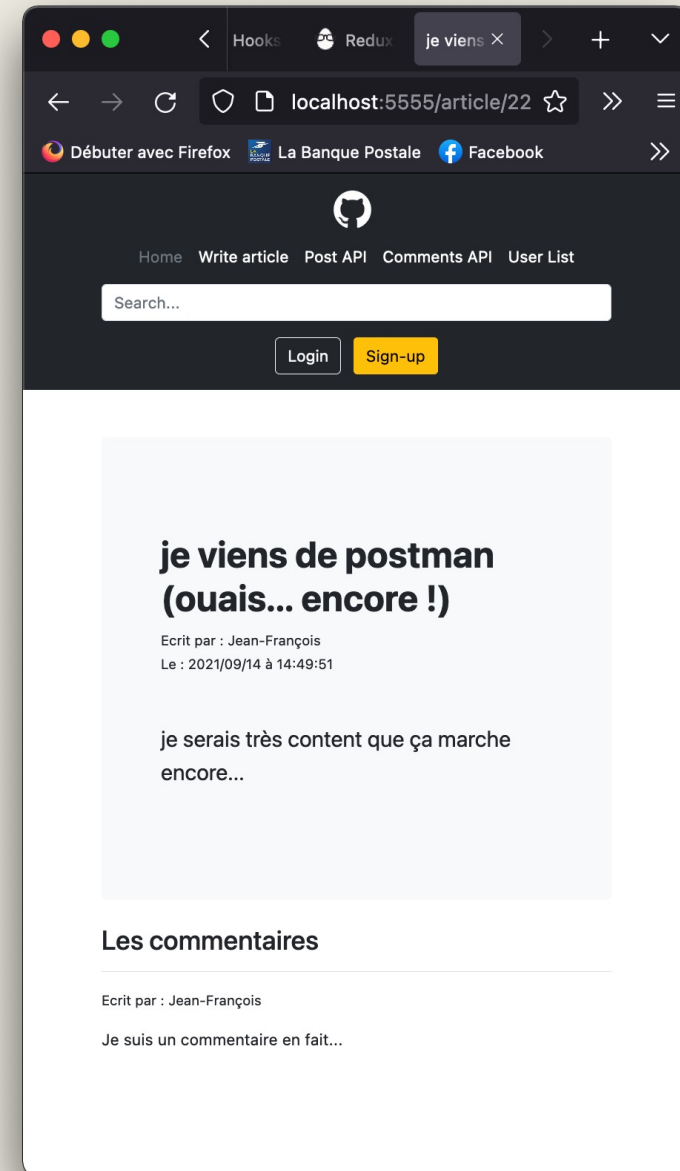
# Consignes

- Pour ce projet, je vous propose de faire un petit CMS (Content Management System) basique
- Ce CMS devra gérer :
  - *Des posts avec :*
    - Titre
    - Contenu
    - Auteur
    - Date de publication
    - Une image d'illustration de l'article
  - *Des commentaires avec :*
    - Un auteur
    - Une date de publication
    - Des réponses au commentaire



# Consignes

- Voici une capture d'écran d'un post quand vous n'êtes pas connecté

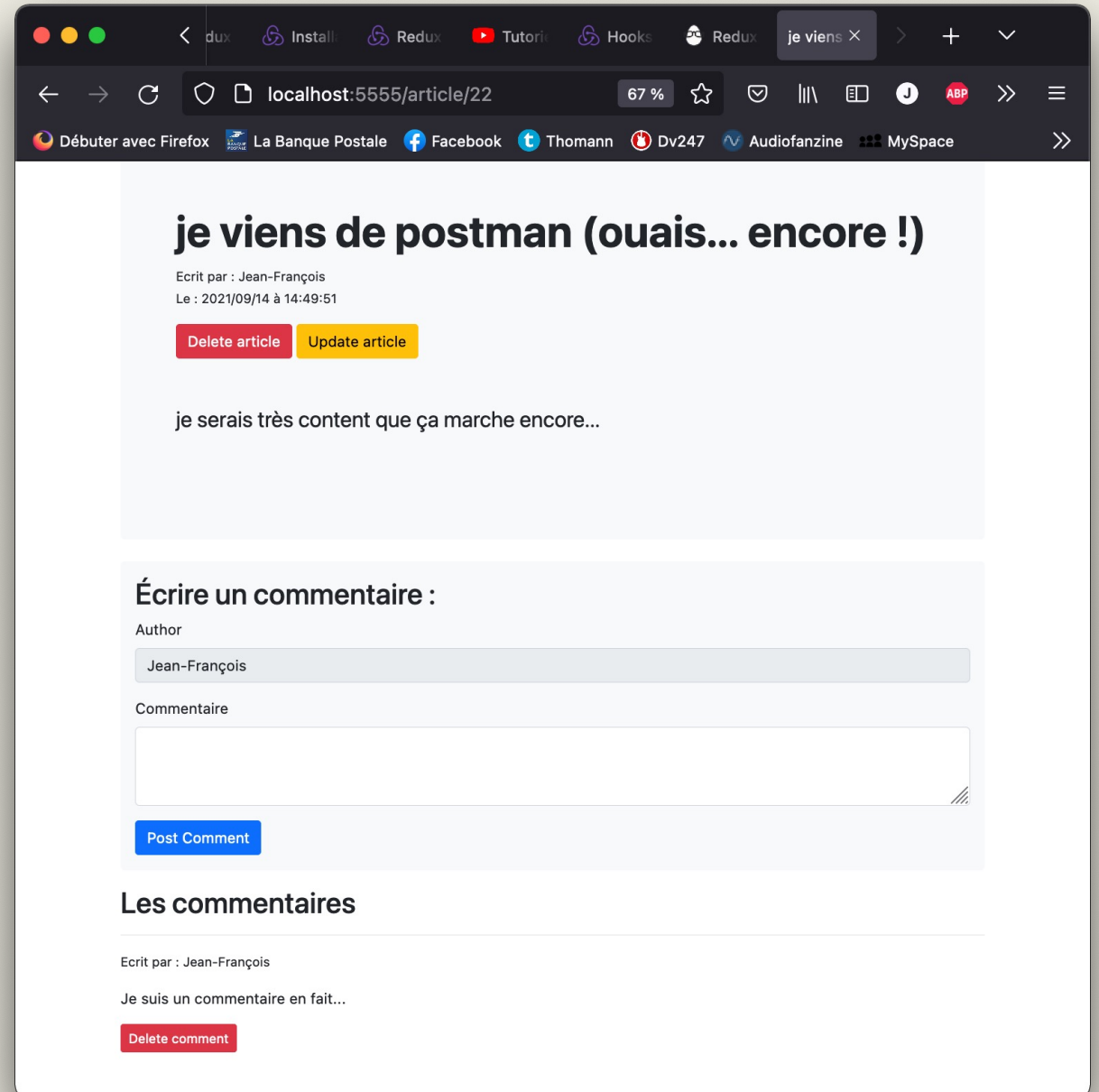


# Consignes

- Votre CMS devra aussi être capable de gérer deux types d'utilisateurs :
  - *Les utilisateurs « standards »*
    - Ils peuvent écrire des posts et des commentaires
    - Ils peuvent également modifier ou effacer les posts et commentaires dont ils sont les auteurs
  - *Les Admins*
    - Ils peuvent écrire, modifier ou supprimer n'importe quel commentaire ou post
    - Ils peuvent également supprimer un autre utilisateur

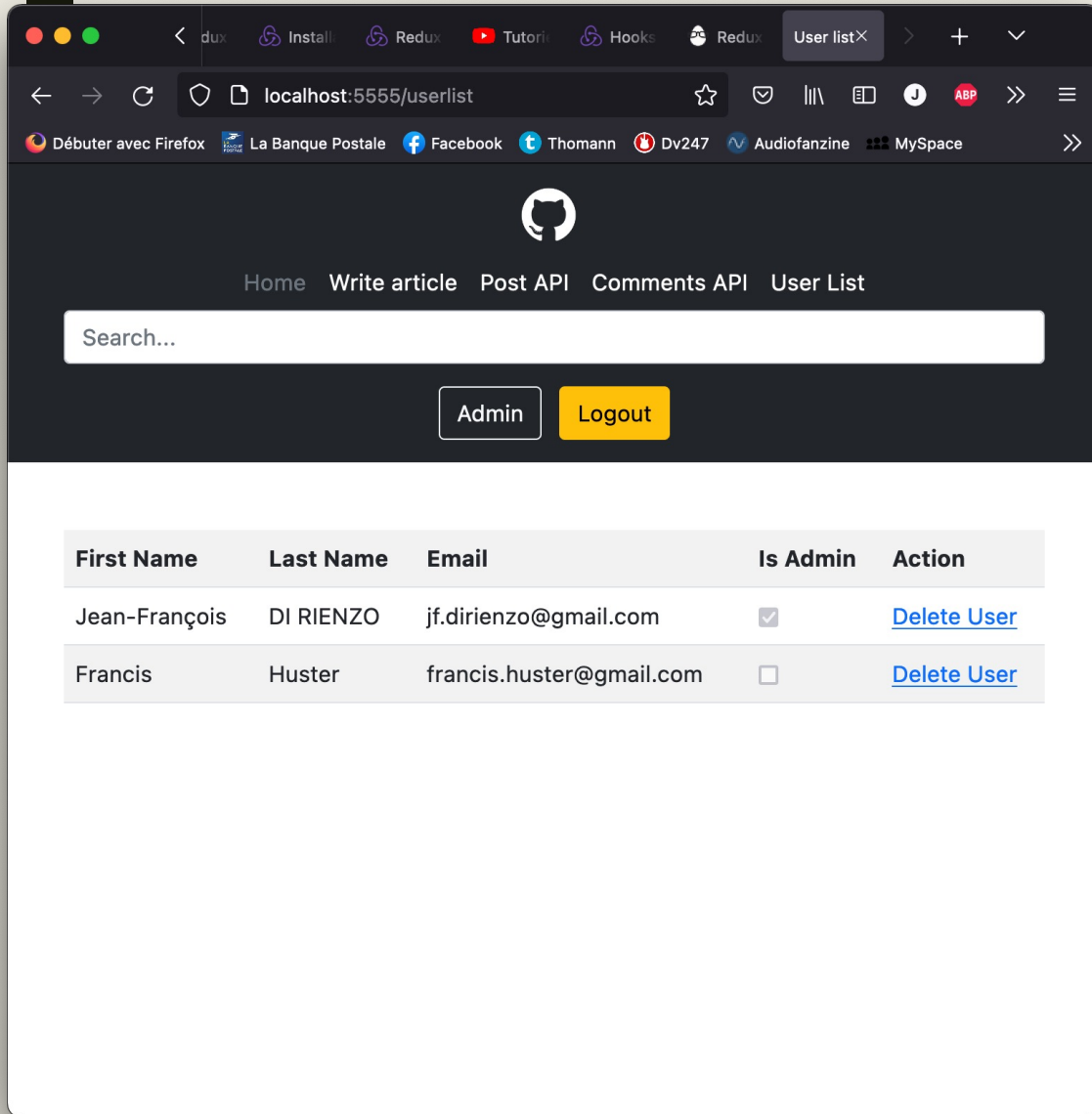
# Consignes

- Voilà ce que voit un Admin



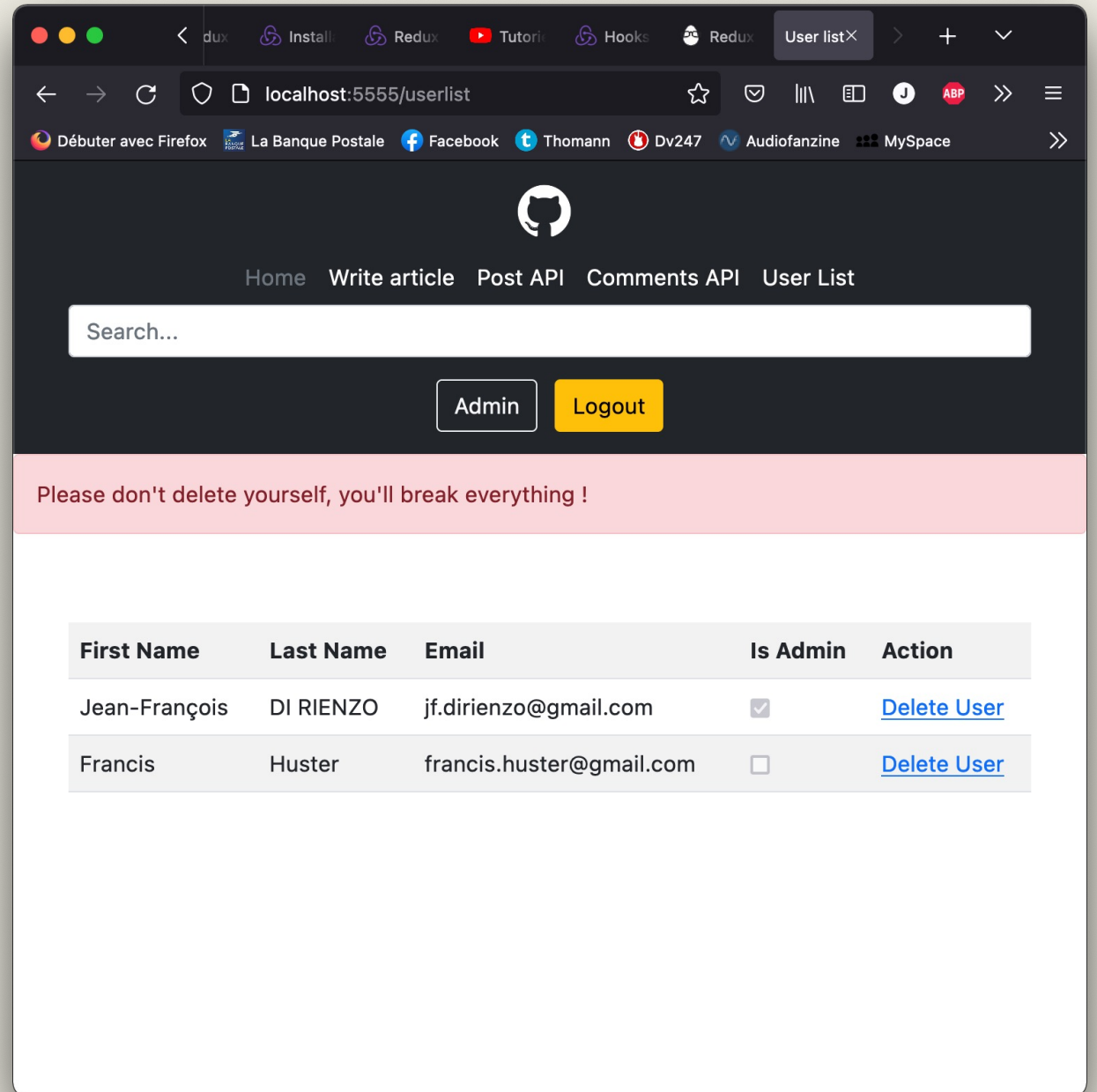
# Consignes

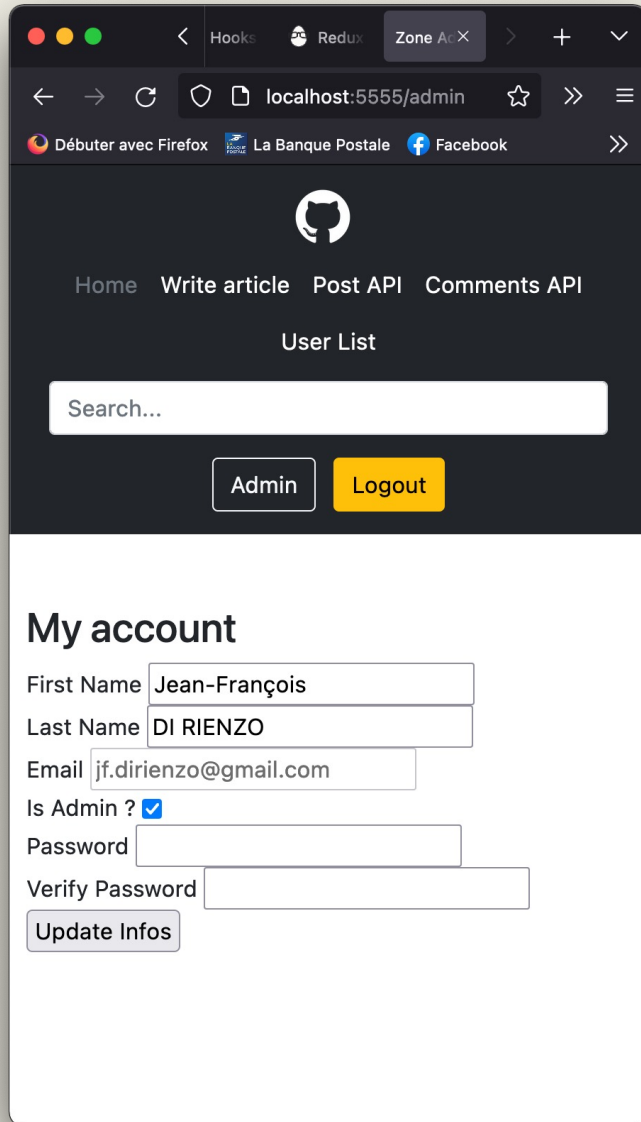
- Cette page n'est accessible qu'aux utilisateurs « admin »



# Consigne

- Un utilisateur ne peut évidemment pas se supprimer lui-même





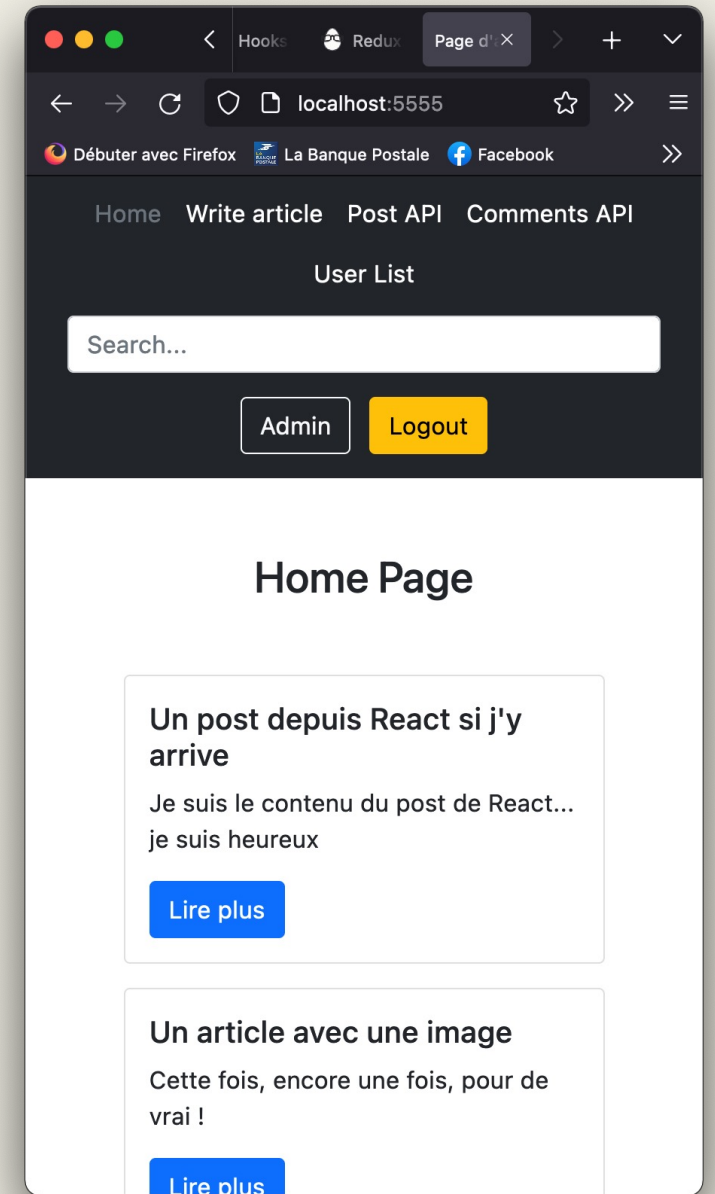
# Consignes

- Dans la zone admin, tous les utilisateurs pourront modifier leurs données personnelles mais aussi s'octroyer des droits administrateurs (même si ce n'est pas forcément super réaliste, ça sera plus simple pour vos tests)
- Votre CMS devra aussi comporter une API REST (promis, on voit ça en cours)



# Frontend

- C'est un projet Backend, **ne perdez pas de temps sur le frontend**, il ne sera pas noté !
- Utilisez Bootstrap, Tailwind ou n'importe quel framework CSS pour gagner du temps, le but est que le tout soit lisible, rien de plus



# Architecture et technologies

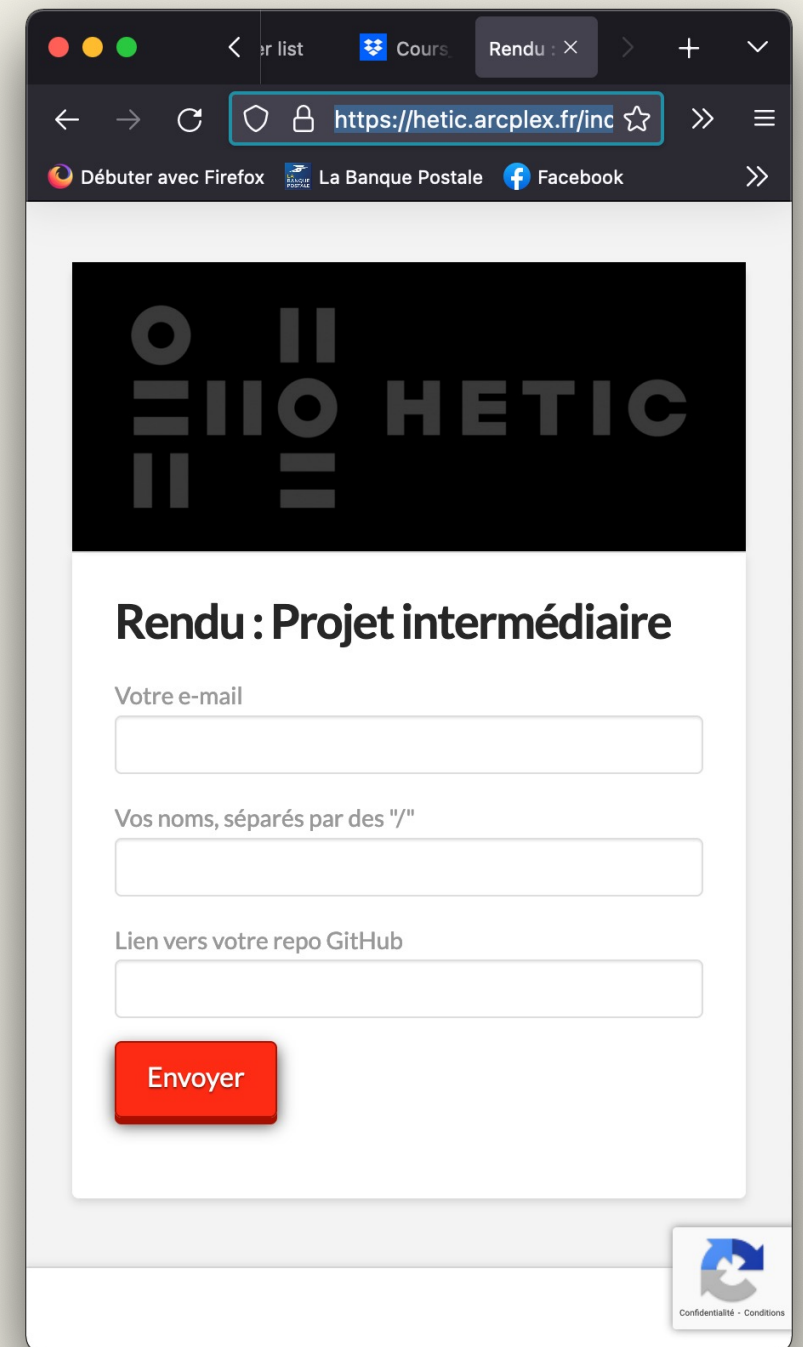
- Votre projet doit être fait sur une stack Docker
- Pérennisez vos données de base de données pour pouvoir plus facilement passer vos projets (et pour que je n'ai pas d'autres manip à faire que de lancer votre stack pour la corriger)
- Le projet doit impérativement être fait en PHP natif, aucun framework n'est autorisé
- Utilisez l'auto-load PSR-4 de Composer, mais n'utilisez aucune librairie
- Votre projet doit être monté avec une architecture MVC
- Votre projet doit utiliser une réécriture d'URLs

# Groupes de travail

- Ce projet est à réaliser en groupes de 3 (les groupes de 2 personnes peuvent être exceptionnellement acceptés)

# Rendu

- Le rendu doit être fait [via ce formulaire](#)
- Le rendu du projet se fait via un repo GitHub qui, une fois téléchargé, pourra être lancé directement en lançant la stack Docker
- La date limite de rendu est le **27/11/2022**



The screenshot shows a web browser window with the address bar displaying <https://hetic.arplex.fr/inc>. The page features a black header with the HETIC logo. Below the header, the title 'Rendu : Projet intermédiaire' is displayed. The form contains three input fields: 'Votre e-mail', 'Vos noms, séparés par des "/"', and 'Lien vers votre repo GitHub'. A red 'Envoyer' button is positioned below the fields. In the bottom right corner, there is a small icon of a circular arrow and the text 'Confidentialité - Conditions'.

**Rendu : Projet intermédiaire**

Votre e-mail

Vos noms, séparés par des "/"

Lien vers votre repo GitHub

**Envoyer**

Confidentialité - Conditions