# Projet Développement Web

Une image contenant Graphique, Police, logo, capture d’écran

Le contenu généré par l’IA peut être incorrect.

Nom du projet : HubUniversité

Enseignante : Mme Guidine Goncalves

Etablissement: Cy Tech

Date de rendu : 30/04/2025

## Introduction

Dans le cadre du module de développement web de première année à CY Tech, notre équipe a travaillé sur la conception d’un site d’université intelligente respectant les enjeux scolaires autour des étudiants, enseignants et du personnels de HubUniversité. Notre projet, intitulé HubUniversité, s’inscrit dans le thème d’une établissement intelligent, intégrant des objets connectés, une interface utilisateur responsive, et une architecture modulaire. Cette plateforme vise à centraliser différents services liés à la gestion, l’administration et la visualisation des données de capteurs ou objets intelligents.

Conformément au cahier des charges, le site se décompose en plusieurs modules fonctionnels : Information, Visualisation, Gestion, et Administration, chacun étant accessible selon le type d’utilisateur (visiteur, étudiant, enseignant, administrateur). Les technologies utilisées incluent React.js pour le frontend, Node.js/Express pour le backend, PostgreSQL pour la base de données, et Docker pour l’orchestration des services. Le dépôt GitHub a été utilisé pour le versionnement et la collaboration continue.

## Arborescence du projet

- HubUniversité/  
 ├── backend/ – API, routes, serveur Node  
 ├── database/ – Scripts PostgreSQL  
 ├── docs/ – Documentation projet  
 ├── frontend/  
 │ ├── public/  
 │ └── src/  
 │ ├── assets/  
 │ ├── components/  
 │ ├── context/  
 │ ├── data/  
 │ ├── hooks/  
 │ ├── styles/  
 │ └── utils/  
 ├── .gitignore  
 ├── docker-compose.yml  
 ├── package.json  
 └── package-lock.json

## Répartition des tâches au sein de l’équipe

Dans le cadre du développement de la plateforme **HubUniversité**, chaque membre du groupe s’est vu attribuer des responsabilités spécifiques, réparties entre le frontend et le backend de l’application. Voici le détail de cette répartition :

### Backend

**Paul PITIOT – Infrastructure, API externe, orchestration Docker**

Paul s’est chargé de la mise en place de l’environnement backend complet, permettant à tout le projet de fonctionner automatiquement via Docker. Ses contributions majeures incluent :

* **Configuration Docker (fichier docker-compose.yml) :**
  + Orchestration des conteneurs Node.js, PostgreSQL et React ;
  + Construction d’un environnement prêt à l’emploi : la commande docker-compose up --build -d permet de lancer l’ensemble du projet ;
* **Initialisation de la base de données (init.sql) :**
  + Création de la structure de base, avec scripts d’insertion de données ;
  + Préparation des tables essentielles pour les utilisateurs et les objets ;
* **Serveur Node.js (server.js) :**
  + Mise en place du serveur Express.js ;
  + Déploiement d’un point d’entrée unique pour les routes backend ;
* **Intégration de l’API SNCF (RER A - Cergy) :**
  + Récupération et traitement en temps réel des données de transport pour enrichir les services proposés aux utilisateurs.

Grâce à cette architecture, le projet peut être lancé sans configuration manuelle supplémentaire.

**Florian DELSUC – Sécurité, gestion des utilisateurs, logique métier**

Florian a travaillé sur la gestion des utilisateurs, les droits d’accès, et la sécurisation des échanges avec le backend. Ses réalisations comprennent :

* **Intégration de bcrypt pour le hachage des mots de passe :**
  + Sécurisation des connexions et inscriptions utilisateurs ;
  + Prévention des accès non autorisés et stockage sécurisé dans la base de données ;
* **Création de la table user dans init.sql :**
  + Définition des champs nécessaires à la gestion des profils (identité, niveau, rôle, mot de passe) ;
  + Intégration avec la logique de connexion via l’auth backend ;
* **Définition des types d’utilisateurs :**
  + **Implémentation des rôles (visiteur, simple, complexe, administrateur) côté backend ;**
  + **Coordination avec le frontend pour garantir des permissions adaptées selon le type de compte.**

Florian a aussi contribué à la structure des **routes Express** en vue de connecter les modules frontend aux ressources backend (utilisateurs, objets, données).

### Frontend

**Souleim GHOUDI – Architecture des pages et responsivité**

Souleim s’est concentré sur la construction fonctionnelle des pages clés du projet et leur adaptation responsive. Ses contributions incluent :

* **Responsive Design :** adaptation de toutes les pages aux formats mobile, tablette et desktop, selon le principe *Mobile First* ;
* **Page Profil utilisateur** :
  + Modification et affichage des informations personnelles (nom, rôle, avatar, etc.) ;
  + Accès aux données privées selon le type d’utilisateur ;
* **Page Gestion (module Gestion)** :
  + Interface permettant aux utilisateurs expérimentés de modifier les paramètres des objets connectés (température, statut, consommation, etc.) ;
  + Visualisation de statistiques et historique des interactions ;
* **Page Connexion Campus (module Visualisation)** :
  + Interface de connexion sécurisée avec vérification de l’identité (type “membre du campus” via la BDD) ;
  + Accès aux outils liés à l’université pour les utilisateurs authentifiés.

Souleim a également participé à la structuration du code dans les dossiers components/ et context/, en créant des composants réutilisables et les contextes utilisateur.

**Louaye SAGHIR – Design, pages d’accueil et administration**

Louaye a pris en charge l’ensemble de la conception graphique du site ainsi que le développement de plusieurs modules vitaux :

* **Charte graphique du projet :**
  + Choix des couleurs, typographies, marges, espacement ;
  + Conception des composants UI (boutons, menus, barres de navigation, cartes) ;
* **Page Visiteur (module Information) :**
  + Affichage des actualités, filtres de recherche, contenu informatif librement accessible ;
  + Accès sans authentification, avec “free tour” de la plateforme ;
* **Dashboard (module Visualisation) :**
  + Page d’accueil personnalisée pour chaque utilisateur connecté ;
  + Affichage de ses objets connectés, services, points d’expérience ;
* **Page Administrateur (module Administration) :**
  + Gestion des utilisateurs : création, suppression, attribution de niveaux ;
  + Statistiques globales (fréquence de connexion, consommation des objets) ;
  + Configuration des objets, catégories, alertes de maintenance.

Louaye a aussi travaillé sur la **navigation générale**, l'**esthétique cohérente** entre les modules, et a optimisé le CSS dans styles/.