

Cooloc

Votre gestionnaire de colocations
CDA 2024 - 2025

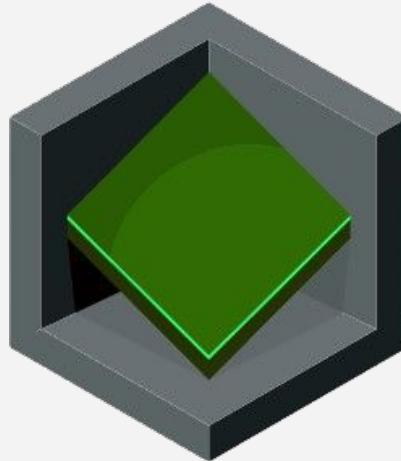


Table des matières

I. Introduction

II. Présentation de l'entreprise

- a. Qu'est ce que la CPAM
- b. Organigramme
- c. Mes missions

III. Présentation de Cooloc

- a. Contexte
- b. Problématique
- c. Objectifs
- d. Public cible

IV. Gestion de Cooloc

- a. Méthode d'organisation
- b. Gestion & planning des tâches
- c. Outils

V. Conception de Cooloc

- a. Diagramme de cas d'utilisation (Use Case)
- b. Diagramme de séquence
- c. Diagramme d'activité
- d. MCD/MLD/MPD
- e. NosQL
- f. Zoning & Wireframes
- g. Charte graphique & Maquettes
- h. Architecture du projet

VI. Démo

VII. Développement Frontend

VIII. Développement Back-end

- a. Architecture & modules pythons
- b. API RESTful
- c. SQL & NoSQL

IX. Sécurité

- a. Rôles
- b. Mots de passe sécurisé
- c. Injection XSS
- d. Injection SQL
- e. Configuration
- f. Authentification
- g. CSRF
- h. RGPD

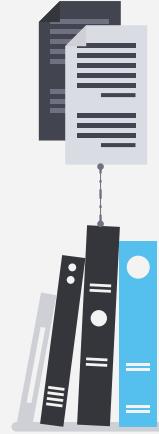
X. DevOps

- a. Conteneurisation
- b. Tests et validations
- c. SonarQube et Github Actions
- d. Documentation technique

XI. Points d'améliorations

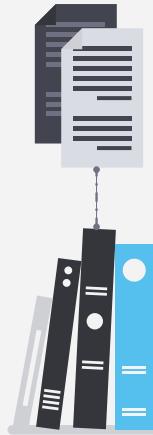
- a. Contrôle des champs
- b. Tokens

XII. Conclusion



I. Introduction

I. Introduction - Qui suis-je



Data Engineer (stage) 

LCL - Villejuif 94800

Concepteur Développeur (alternance) 

CPAM du Val-de-Marne - Créteil 94000

2023

2024

2024

2025

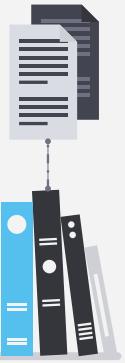
DevOps (stage) 

Société Générale - Fontenay-sous-bois
94120

Data Engineer (alternance) 

Crédit Agricole Assurances - Paris 75015 - À venir





I. Introduction - Parcours scolaire



Lycée Alexandre Dumas

2019 - 2022

Baccalauréat général
Physique-Chimie & NSI



Efrei Paris



Efrei Paris

2025 - 2027 (à venir)

Mastère
Data Engineering & IA



II. Présentation de l'entreprise

II. Présentation de l'entreprise - CPAM



Que financent 1000 € d'impôts ?



L'Assurance Maladie et la prévention, un engagement historique

1945

L'ordonnance du 4 octobre 1945 instaure la Sécurité sociale et évoque la « prévention » à 4 reprises

1985

Organisation de la première campagne de vaccination contre la grippe

2005

Lancement du programme M'T dents, qui sera étendu à de nouvelles classes d'âge en 2018

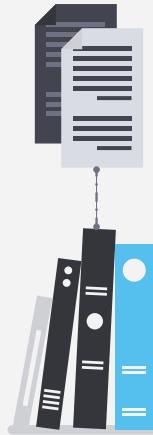
2020

Mise en place d'actions d'« aller vers » lors de la vaccination anti-Covid

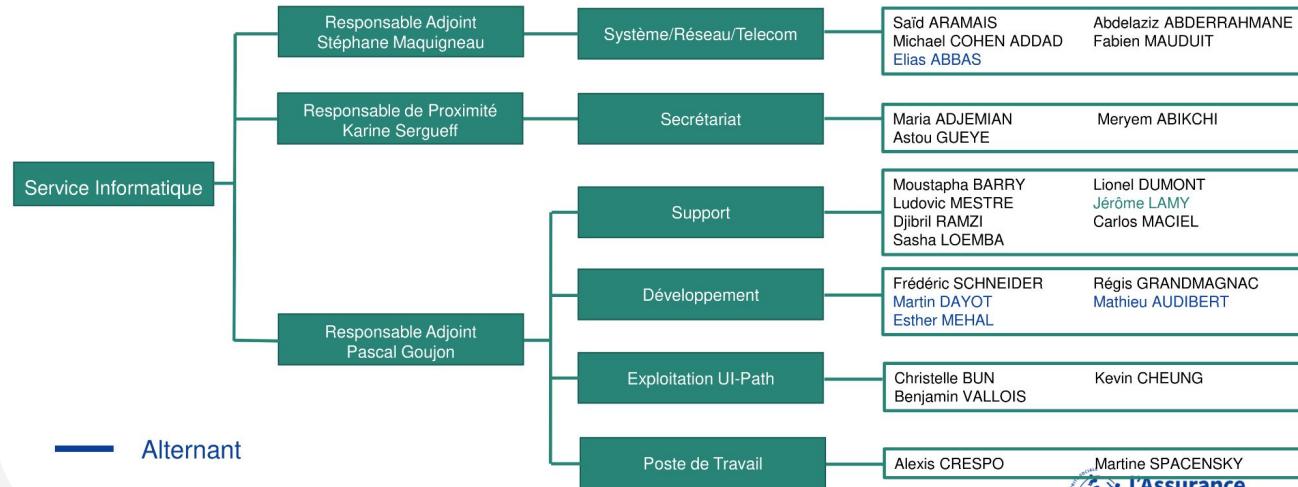
2008

Déploiement du service sophia, pour améliorer le suivi de patients diabétiques

II. Présentation de l'entreprise - Organigramme



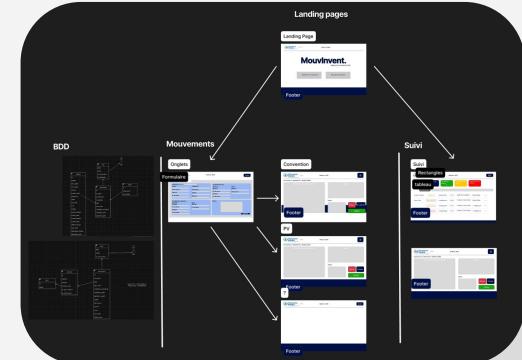
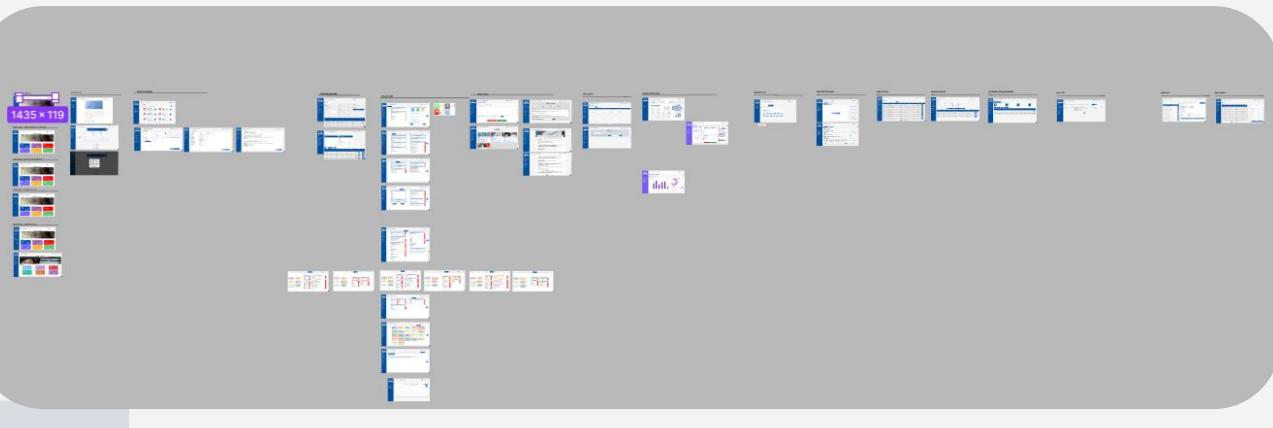
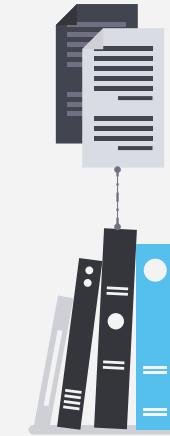
ANNEXE ETAT DES LIEUX – ORGANIGRAMME



II. Présentation de l'entreprise - Mes missions

AAVA Action Aller Vers les Assurés

The screenshot shows the AAVA application interface. At the top, there is a navigation bar with a logo and a search bar. Below the navigation bar, there is a sidebar with various menu items: Administration Locale, Accès à la plateforme, Création, Applications, Envoyer des emails, Modèles de courriers, Paramètres, and Actualisation. The main area is titled "AAVA Action Aller Vers les Assurés". It has a header with a user profile icon and the text "Nom Prenom". Below the header, there is a section titled "Campagnes" with a search bar and filters for "Trier par (a-z)", "Equipe", "Statut", and "Priorité". There are two rows of campaign cards. Each card displays a doctor icon, the campaign name (e.g., "Nom_Campagne XX fiches"), the number of faxes sent ("XX faxes"), the number of available faxes ("Disponibles : 50"), the number of faxes in queue ("En attente : 20"), and the percentage of completion ("20% effectuées"). Buttons for "Envoyer" (Send), "Annuler" (Cancel), and "..." are also present. At the bottom, there is a pagination control with buttons for 1, 2, 3, 4, and "...".



II. Présentation de l'entreprise - Mes missions

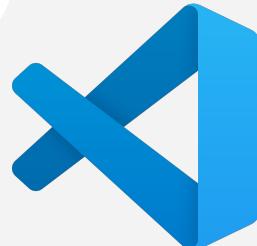


GitLab

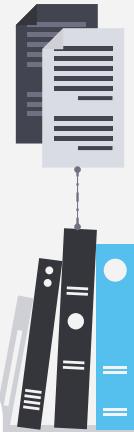
php



Symfony



MySQL®



III. Présentation de Cooloc

III. Présentation de Cooloc - Contexte



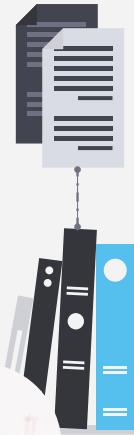
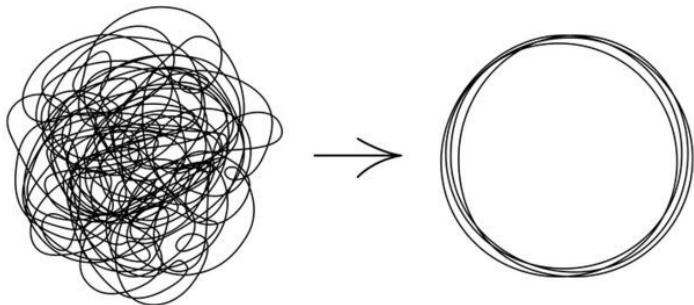
Précarité étudiante en 2023



des étudiants ne mangent pas à leur faim
soit 28% des boursiers et 16% des non-boursiers.

Source : Bouge ton Crous, la consultation étudiante de La Fage, 2024.

III. Présentation de Cooloc - Problématique



III. Présentation de Cooloc - Objectifs



+Simple

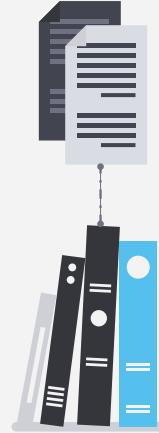
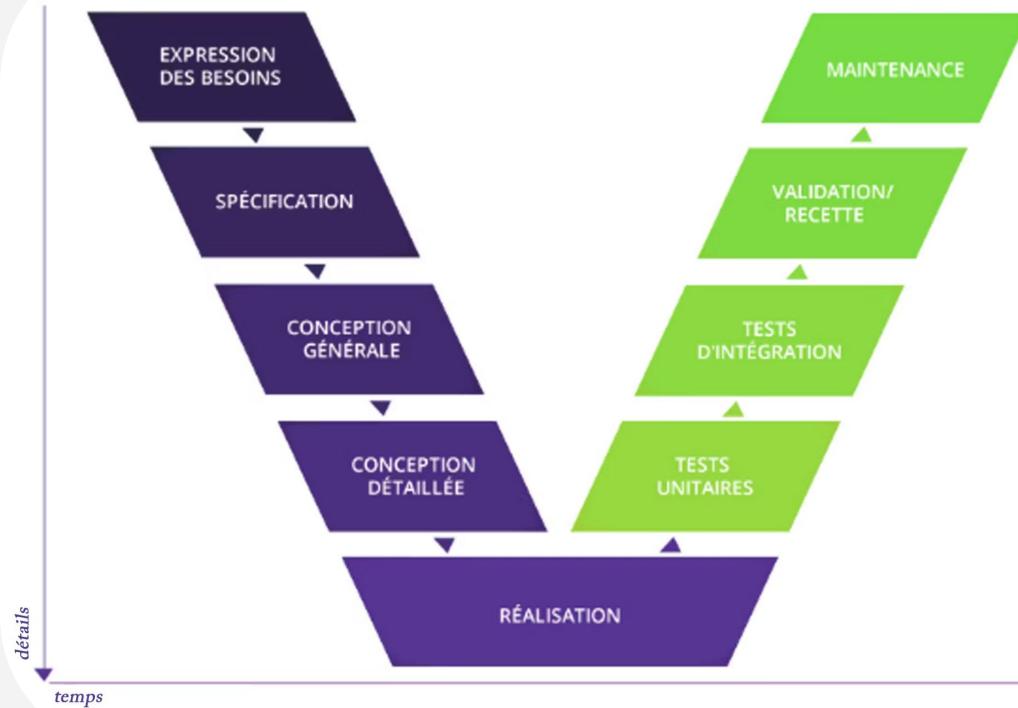


III. Présentation de Cooloc - Public cible

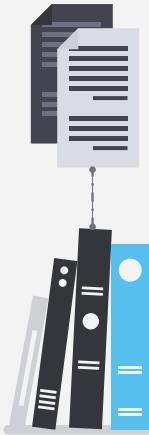


IV. Gestion du projet

IV. Gestion du projet - Méthode



IV Gestion du projet - Planning des tâches



IV. Gestion du projet - Gestion des tâches

Coolc ★ Public 0 Tableau

A FAIRE

- DIAGRAM DE GANT
- DIAGRAM DE CLASS
- PAGES TACHES 0/2

+ Ajouter une carte

EN COURS

- FIGMA
- CONCEPTION
- DOCUMENT 40 PAGES
- POWERPOINT
- DEVELOPPEMENT
- RESPONSIVE 0/2

+ Ajouter une carte

A CORRIGER/BUGS

- GITHUB ACTIONS
- SONARQUBE

+ Ajouter une carte

TERMINÉ

- DIAGRAM DE SEQUENCES
- MCD
- MLD
- MPD
- BDD 2/2
- DOCKER 2/2
- NOSQL
- JWT
- HEADER PAS CONNECTÉ 2/2
- ROLES
- LOGIN 2/2

+ Ajouter une carte

IDEES

- DOCUMENTATION

+ Ajouter une autre liste

Power-ups Automatisation Filtres Partager ...

Diagram illustrating the project management board:

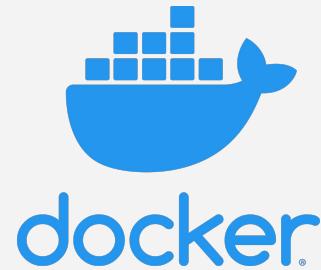
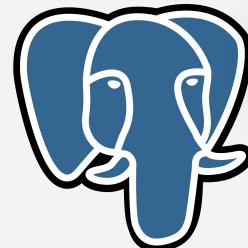
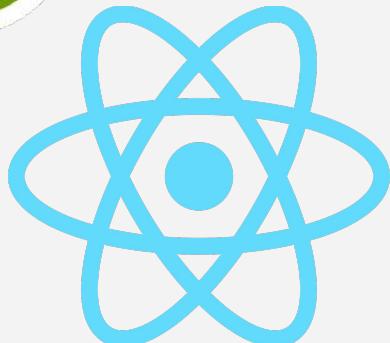
- A FAIRE:** Diagram de Gant, Diagram de Class, Pages Taches (0/2).
- EN COURS:** FIGMA, CONCEPTION, DOCUMENT 40 PAGES, POWERPOINT, DEVELOPPEMENT, RESPONSIVE (0/2).
- A CORRIGER/BUGS:** GITHUB ACTIONS, SONARQUBE.
- TERMINÉ:** Diagram de Séquences, MCD, MLD, MPD, BDD (2/2), DOCKER (2/2), NOSQL, JWT, HEADER PAS CONNECTÉ (2/2), ROLES, LOGIN (2/2).
- IDEES:** DOCUMENTATION.



IV. Gestion du projet - Outils

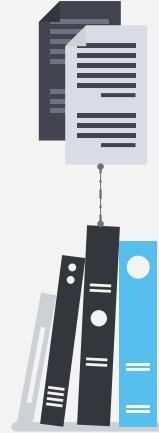
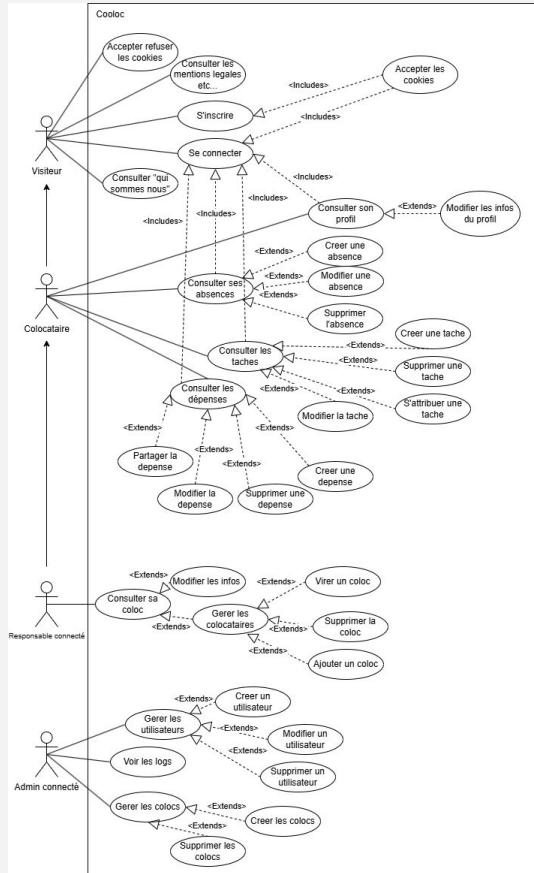


PYTHON
HTTP
MODULE

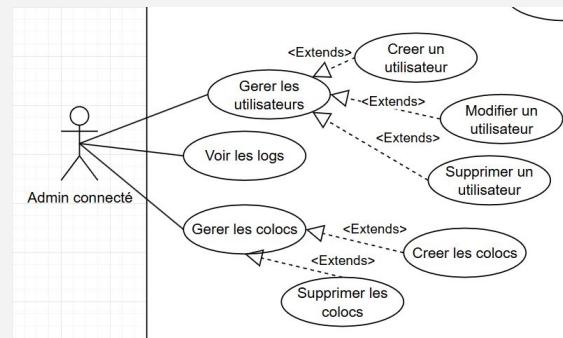
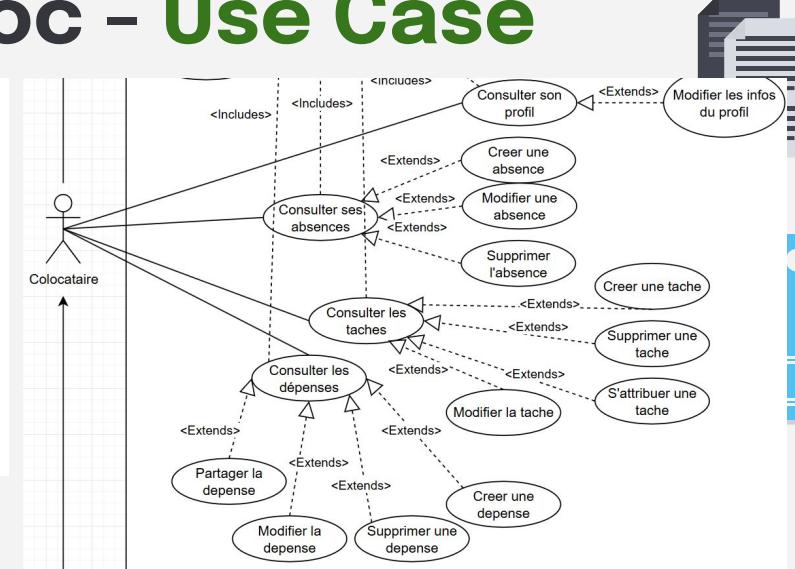
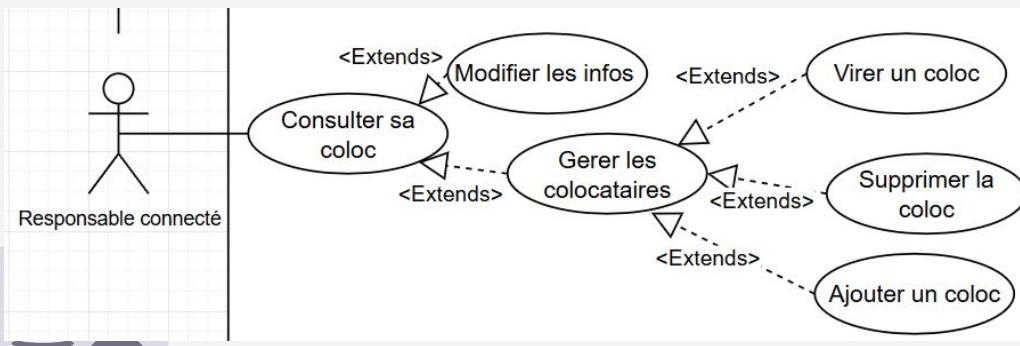
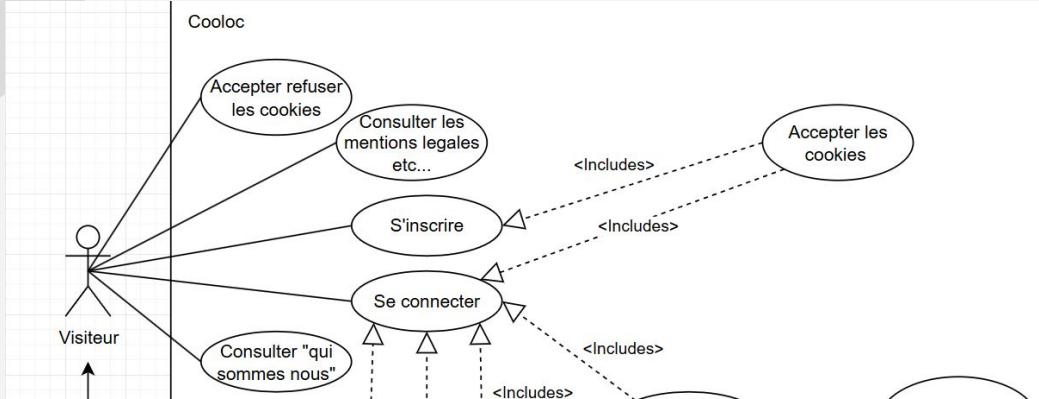


V. Conception de Cooloc

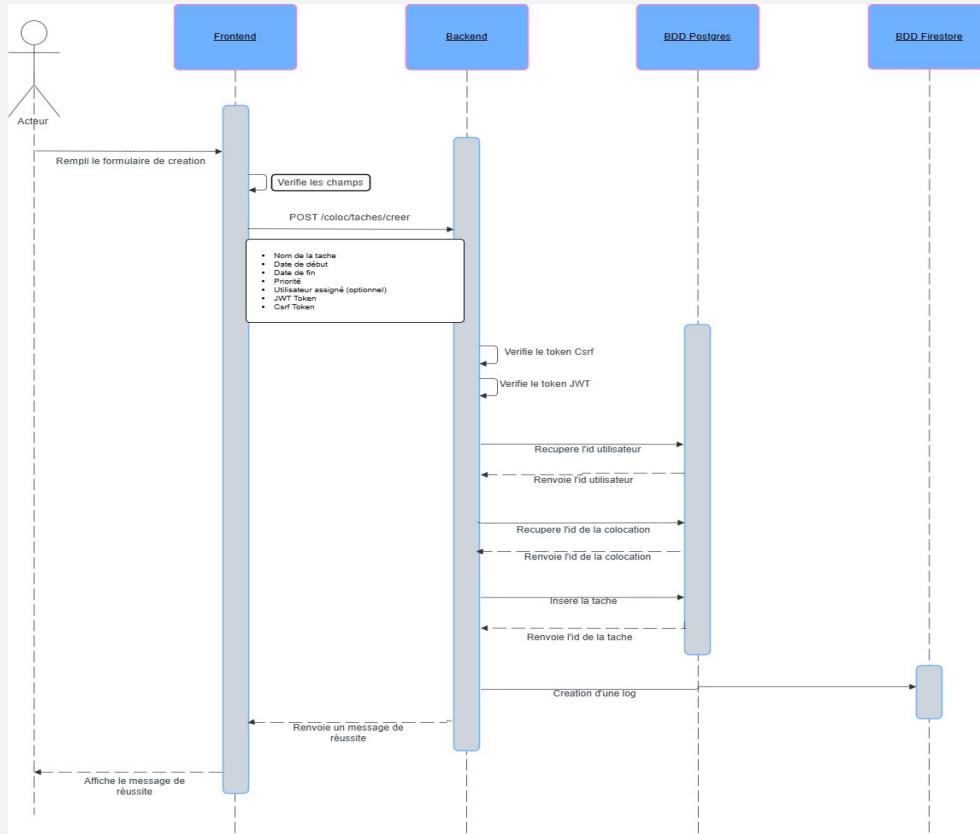
V. Conception de Cooloc - Use Case



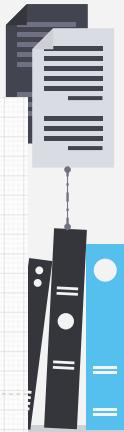
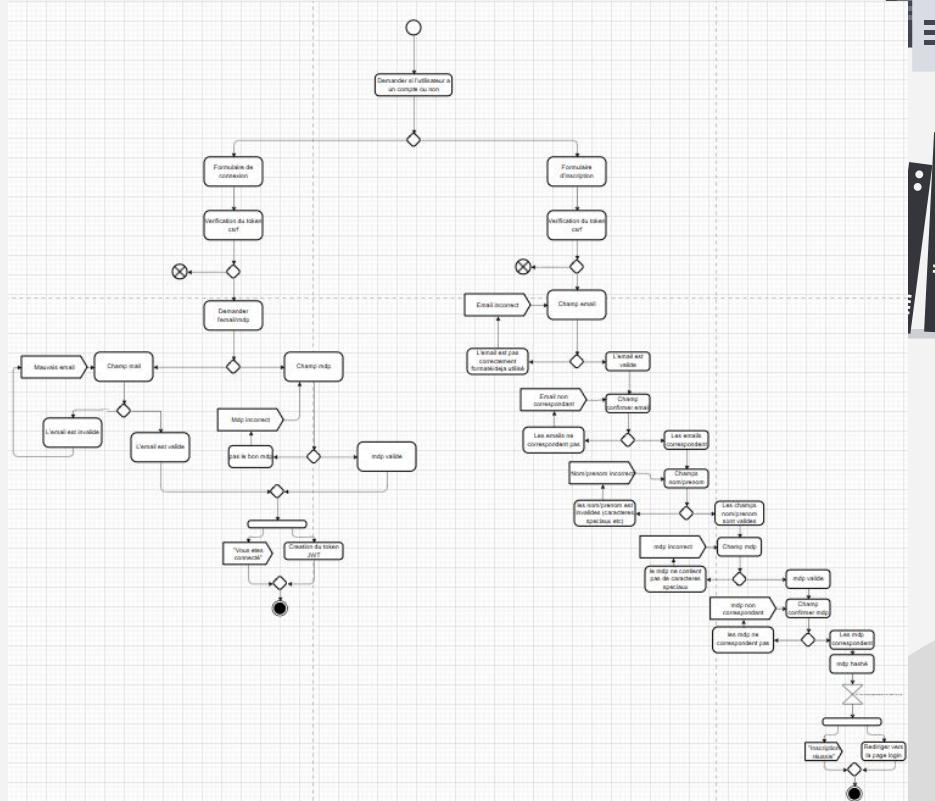
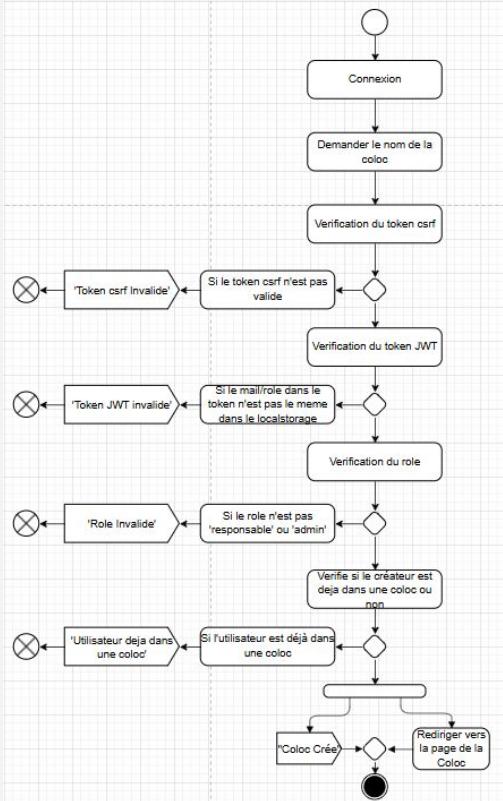
V. Conception de Cooloc - Use Case



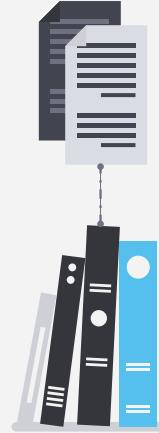
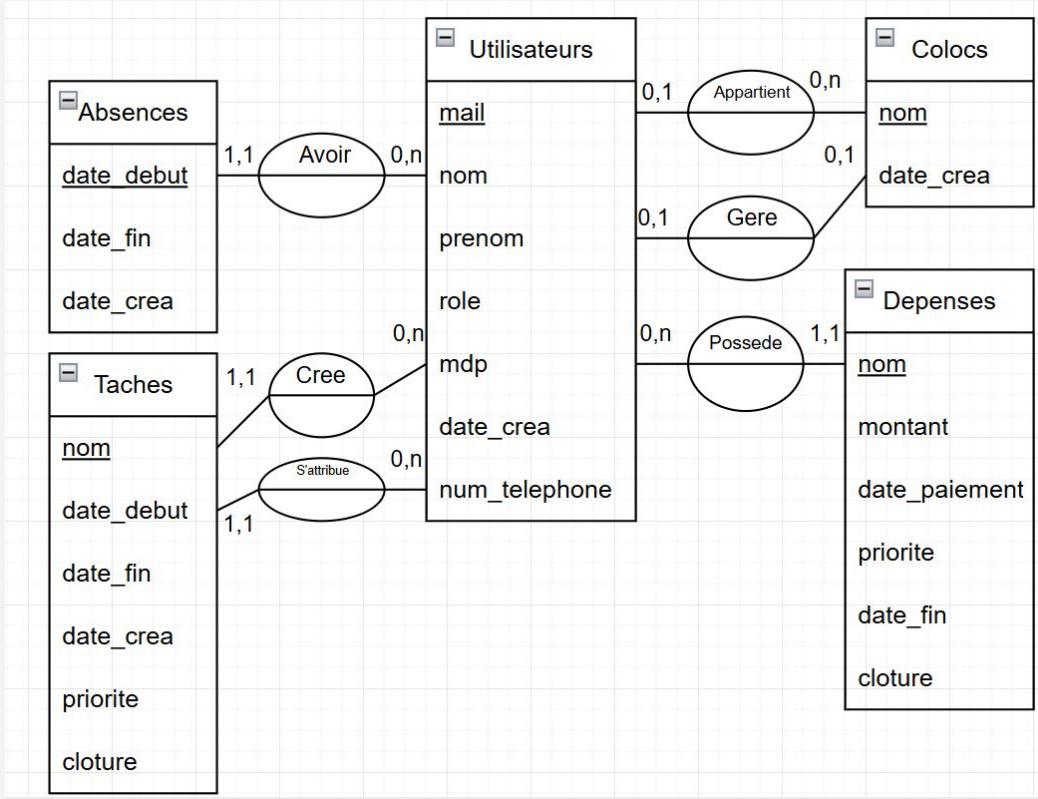
V. Conception de Cooloc - Séquence



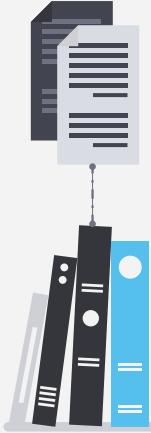
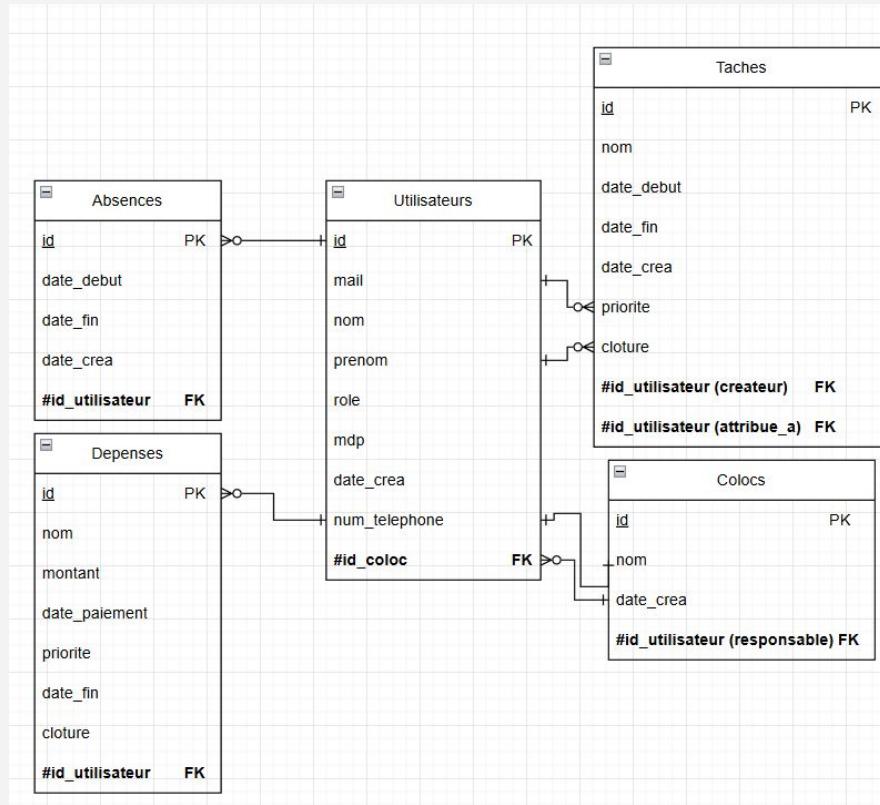
V. Conception de Cooloc - Activité



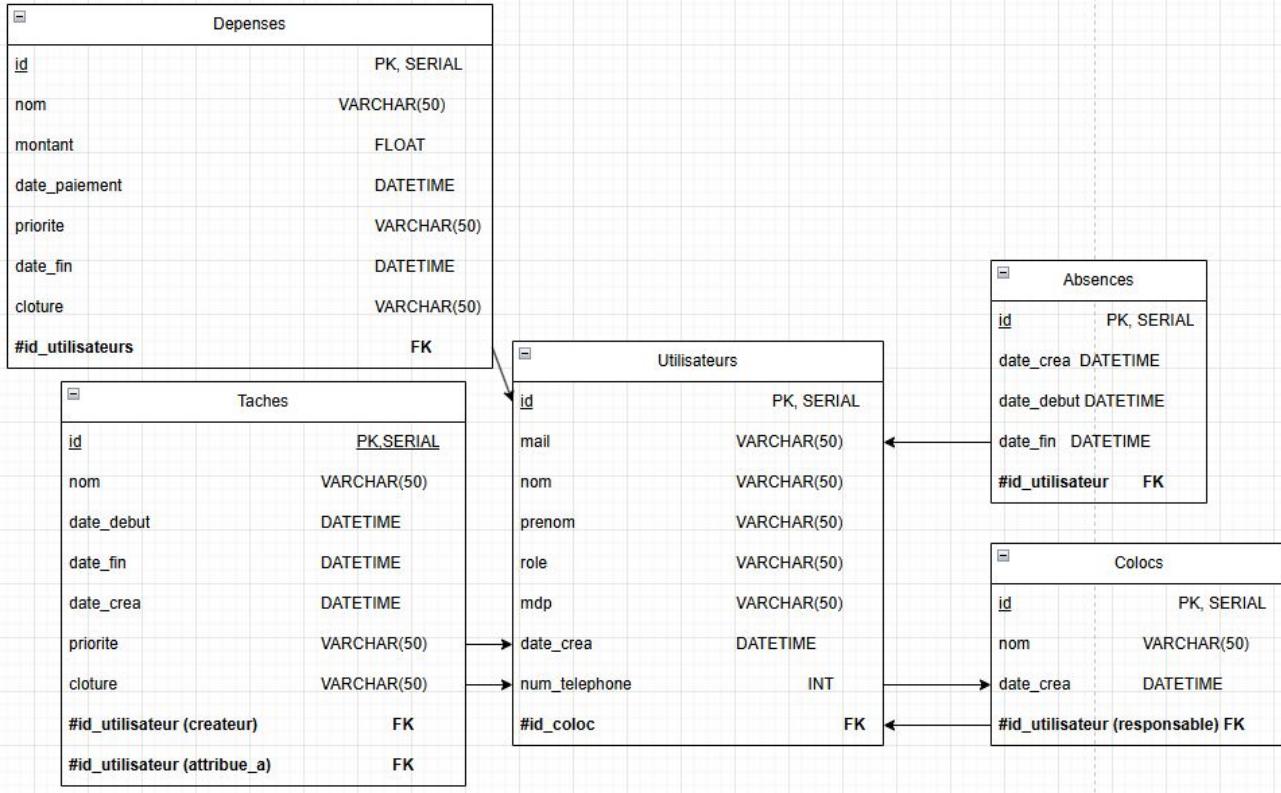
V. Conception de Cooloc - MCD



V. Conception de Cooloc - MLD



V. Conception de Cooloc - MPD



```

backups > back > bdd.sql
1 ✓ CREATE TABLE IF NOT EXISTS Colocs (
2   id SERIAL PRIMARY KEY,
3   nom VARCHAR(50),
4   date_crea TIMESTAMP,
5   responsable INT,
6 );
7
8 ✓ CREATE TABLE IF NOT EXISTS Utilisateurs (
9   id SERIAL PRIMARY KEY,
10  mail VARCHAR(50) NOT NULL UNIQUE,
11  nom VARCHAR(50) NOT NULL,
12  prenom VARCHAR(50) NOT NULL,
13  role VARCHAR(50) NOT NULL,
14  mdp VARCHAR(255) NOT NULL,
15  date_creation TIMESTAMP,
16  num_telephone VARCHAR(15),
17  id_coloc INT REFERENCES Colocs(id),
18  CONSTRAINT fk_coloc
19    FOREIGN KEY(id_coloc)
20      REFERENCES Colocs(id)
21 );
22
23 ✓ CREATE TABLE IF NOT EXISTS Absences (
24   id SERIAL PRIMARY KEY,
25   date_crea TIMESTAMP,
26   date_debut TIMESTAMP,
27   date_fin TIMESTAMP,
28   id_utilisateur INT REFERENCES Utilisateurs(id)
29 );
30
31 ✓ CREATE TABLE IF NOT EXISTS Depenses (
32   id SERIAL PRIMARY KEY,
33   nom VARCHAR(50),
34   montant FLOAT,
35   date_paiement TIMESTAMP,

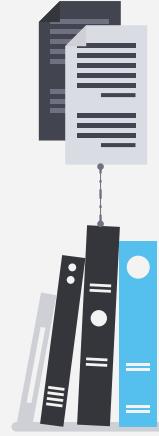
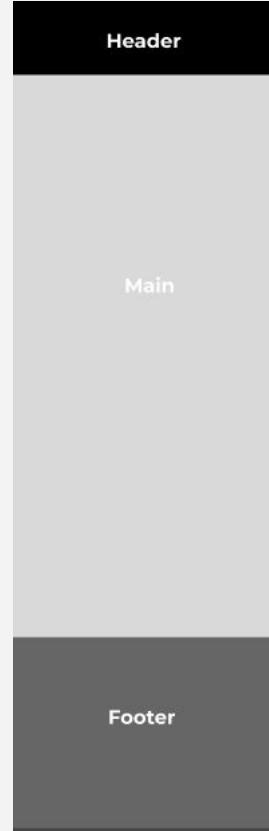
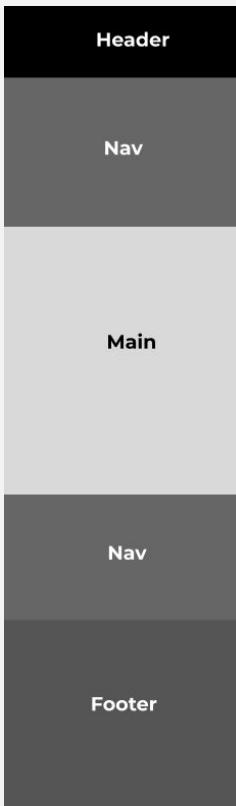
```

V. Conception de Cooloc - NoSQL

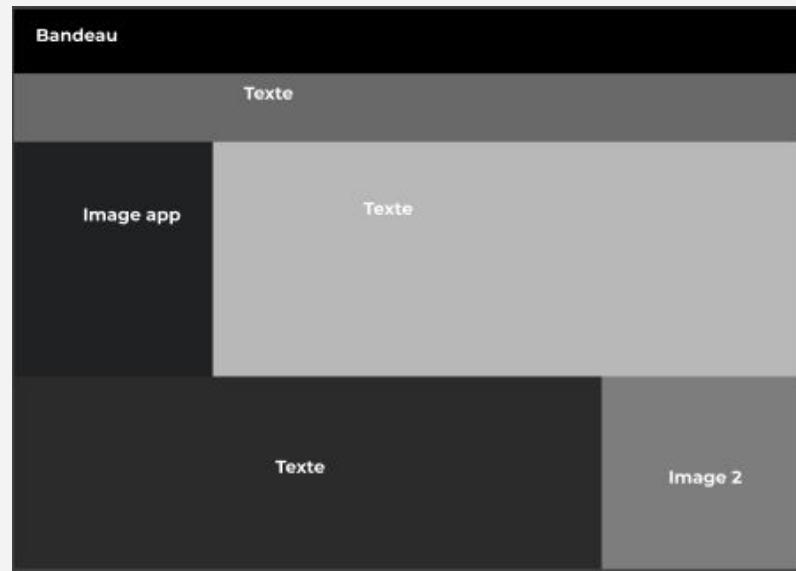
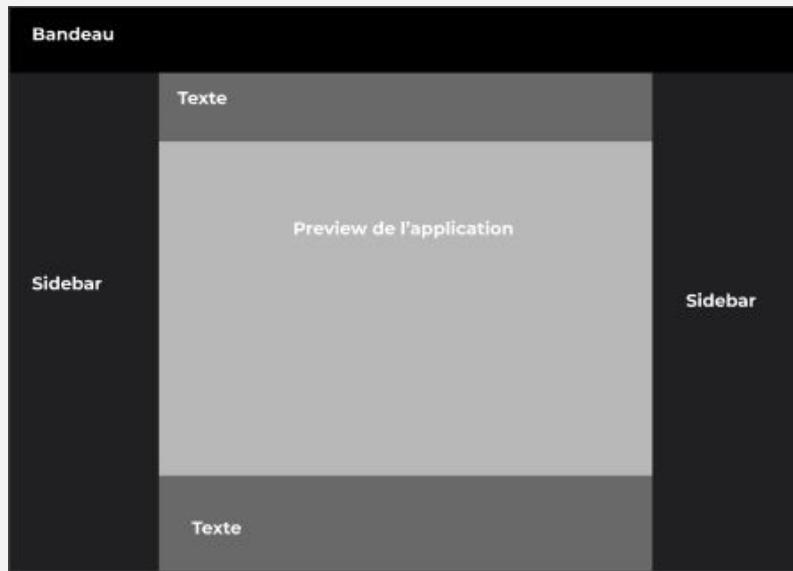
The screenshot shows the Firebase Cloud Firestore interface for the 'Logs' collection. The left sidebar includes links for Vue d'ensemble, Raccourcis de projet, Firestore Database (selected), Catégories de produits, Créer, Exécuter, Analytics, IA, and Tous les produits. The top navigation bar has tabs for Données, Règles, Index, Reprise après sinistre, Utilisation, and Extensions. A red button 'Ajouter une base de données' is visible. The main area displays a hierarchical path: Logs > 1GJ6DJa2RjwiRSZ1Czb. The collection contains several documents, with one document expanded to show its fields: action: "creation coloc", date: "23 juin 2025 à 17:09:11 UTC+", id_coloc: 21, and id_utilisateur: 10. A sidebar on the right provides links to 'Vue Panneau', 'Générateur de requêtes', and 'Plus de fonctionnalités dans Google Cloud'. A decorative graphic of books and a stack of papers is visible on the far right.

Emplacement de la base de données : eur3

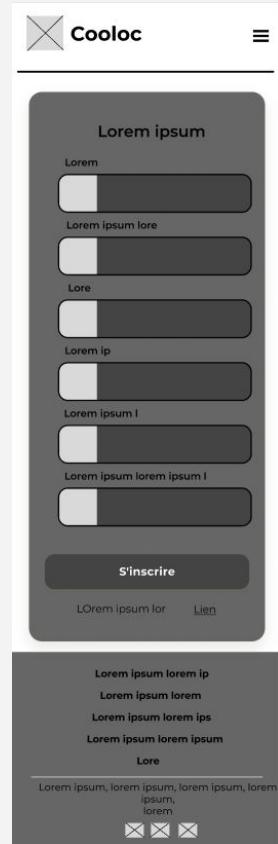
V. Conception - Zoning mobile



V. Conception - Zoning pc



V. Conception - Wireframes mobile



V. Conception - Wireframes pc



The wireframe shows the main landing page for Cooloc. At the top, there's a header with the logo (a stylized envelope icon), the word "Cooloc", and navigation links for "Accueil", "Qui sommes nous", and "Contact". Below the header is a large call-to-action section with the text "Grâce à Cooloc, rendez vos colocataires plus facilement gérables !". This section features a large gray square divided by a large 'X'. Below this, there's placeholder text: "Lorem ipsum...". Underneath this, there's a section titled "Tâches" (Tasks) containing two more gray squares with 'X' patterns. Each task has a short description below it. At the bottom of the page, there's another section titled "Lorem" with similar gray squares and placeholder text.

The wireframe shows the sign-up page for Cooloc. It features a header with the logo and navigation links for "Accueil", "Qui sommes nous", and "Contact". Below the header is a large central area with a dark gray rounded rectangle containing placeholder text ("Lorem ipsum") and several input fields. At the bottom of this area is a "S'inscrire" (Sign up) button. Below the button, there's a link labeled "Lien". The footer of the page contains a row of social media icons and the text "Lorem ipsum...".

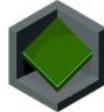


V. Conception - Charte graphique

Couleurs



Logo & Polices



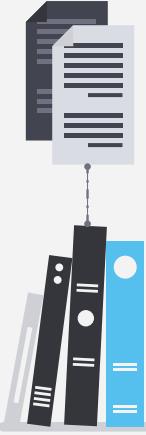
Cooloc utilise Montserrat, Lorem ipsum 123456789

Cooloc utilise Montserrat, Lorem ipsum 123456789

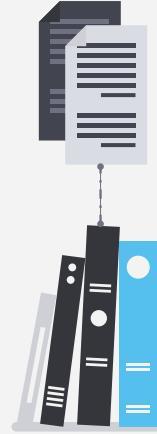
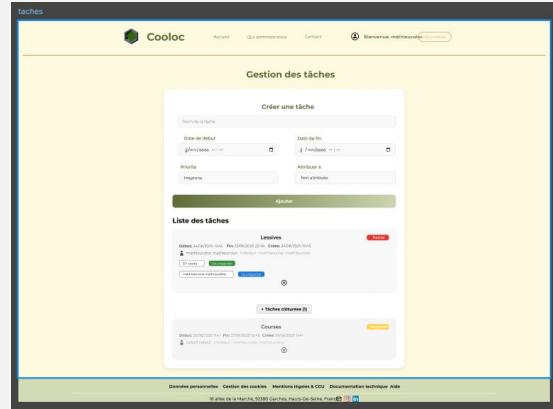
Cooloc utilise Montserrat, Lorem ipsum 123456789

Cooloc utilise Montserrat, Lorem ipsum 123456789

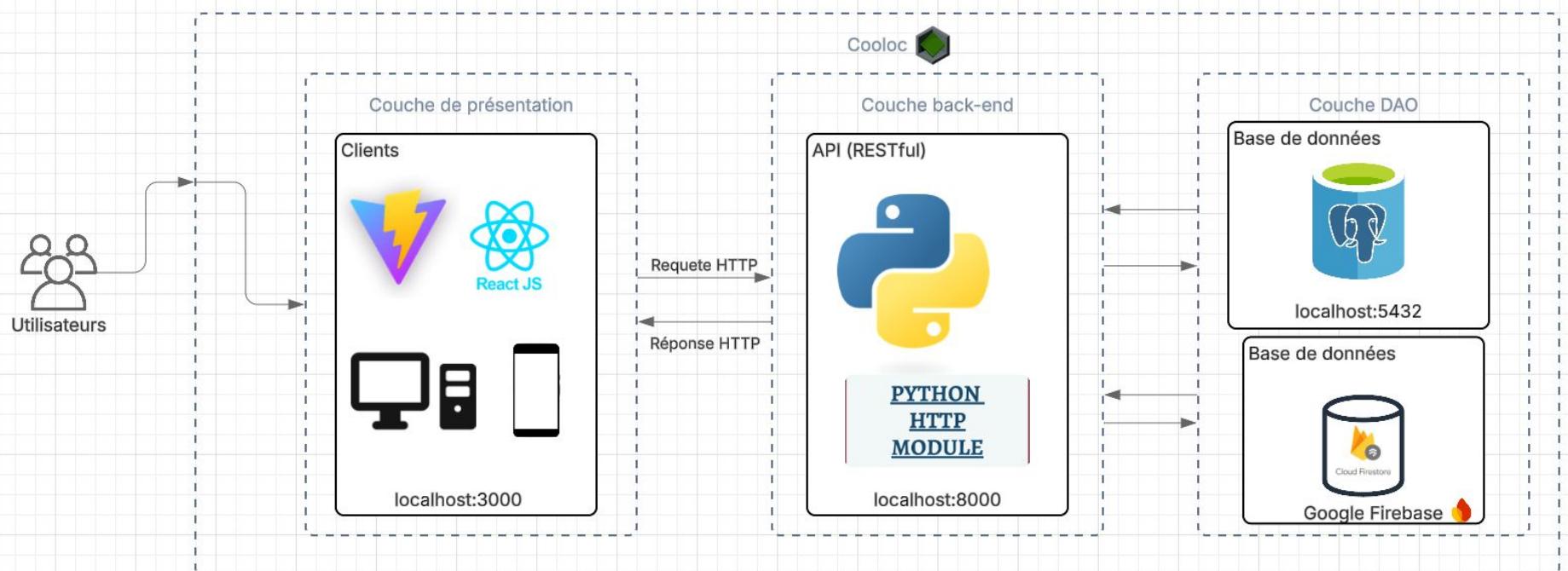
V. Conception - Maquettes mobile



V. Conception - Maquettes pc



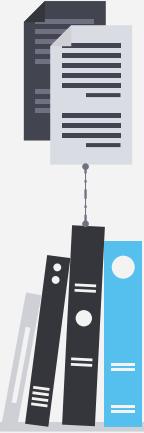
V. Conception de Cooloc - Architecture



VI. Démo

VII. Développement Frontend

VII. Frontend - Navigation conditionnelle



Login.jsx

```
...          Uploaded using RayThis Extension

const response = await fetch('http://localhost:8000/login', {
    method: 'POST',
    headers: {
        'Content-Type': 'application/json',
        'Accept': 'application/json',
    },
    body: JSON.stringify({
        mail: email,
        mdp: password,
        csrf: 'cz6hyCmAUIU7D1htACJKe2HwfE6bqAiksEOYJABM3-Y'
    }),
    mode: 'cors',
    credentials: 'include'
});

if (!response.ok) {
    throw new Error(`erreur: ${response.status}`);
}

const data = await response.json();

if (data.status === 200) {
    localStorage.setItem('user', JSON.stringify({
        email: data.data.mail,
        role: data.data.role,
        prenom: data.data.prenom,
        nom: data.data.nom,
        token: data.token,
        id_coloc: data.data.id_coloc
    }));
}

window.location.href = '/';
```



VII. Frontend - Navigation conditionnelle

App.jsx

```
...                                                 Uploaded using RayThis Extension

switch (currentPage) {
  case 'home':
    return <Home />;
  case 'login':
    return <Login onClick={() => navigate('register')} />;
  case 'register':
    return <Register onClick={() => navigate('login')} />;
  case 'profile':
    return <Profile />;
  case 'admin':
    return <AdminDashboard />;
  case 'a-propos':
    return <APropos />;
  case 'creation-colocation':
    return <CreationColocation />;
  case 'colocation':
    return <ColocationInfos />;
  case 'taches':
    return <Taches />;
  default:
    return <Home />;
}
```

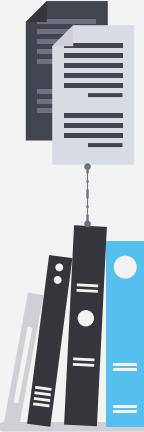


Header.jsx

```
<div className="header-right">
  {localUser ? (
    <div className="user-info">
      {localUser.role === 'admin' && (
        <button onClick={onAdminClick} className="btn btn-admin">
          Tableau de Bord Admin
        </button>
      )}
      {localUser.role !== 'admin' && (
        <button onClick={onProfileClick} className="profile-button">
          
        </button>
      )}
      <span className="welcome-text">Bienvenue, {localUser.prenom}</span>
      <button onClick={handleLogout} className="btn btn-logout">Déconnexion</button>
    </div>
  ) : (
    <>
      <button onClick={() => handleAuthClick(onLoginClick)} className="btn btn-login">Se connecter</button>
      <button onClick={() => handleAuthClick(onRegisterClick)} className="btn btn-register">S'inscrire</button>
    </>
  )
</div>
```



VII. Frontend - Interfaces

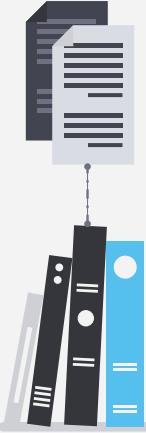


AdminDashboard.jsx

Profile.jsx

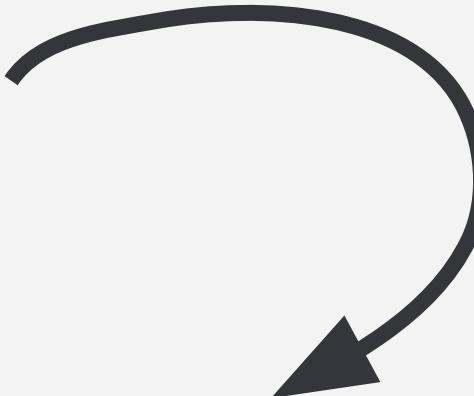
```
1 import {useState, useEffect} from 'react';
2 import './styles/profile.css';
3
4 const Profile = () => {
5   const [user, setUser] = useState(null);
6   const [isEditing, setIsEditing] = useState(false);
7   const [error, setError] = useState('');
8   const [success, setSuccess] = useState('');
9   const [formData, setFormData] = useState({
10     email: '',
11     prenom: '',
12     nom: '',
13     password: '',
14     confirmPassword: ''
15   });
16
17   useEffect(() => {
18     const storedUser = localStorage.getItem('user');
19     if (storedUser) {
20       const userData = JSON.parse(storedUser);
21       setUser(userData);
22       setFormData({
23         email: userData.email,
24         prenom: userData.prenom,
25         nom: userData.nom,
26         password: '',
27         confirmPassword: ''
28       });
29     }
30   }, []);
31
32   const handleInputChange = (e) => {
33     const { name, value } = e.target;
34     setFormData(prev => {
35       ...prev,
36       [name]: value
37     });
38   };
39
40   const handleSubmit = async (e) => {
41     e.preventDefault();
42     if (!isEditing) return;
43     setError('');
44     setSuccess('');
45
46     if (formData.password !== formData.confirmPassword) {
47       setError('Les mots de passe ne correspondent pas');
48       return;
49     }
50   };
51
52   const handleEdit = () => {
53     setIsEditing(true);
54   };
55
56   const handleSave = () => {
57     if (error) return;
58     setError('');
59     setSuccess('L\'utilisateur a été mis à jour');
60     setIsEditing(false);
61   };
62
63   const handleDelete = () => {
64     if (error) return;
65     setError('');
66     setSuccess('L\'utilisateur a été supprimé');
67     setIsEditing(false);
68   };
69
70   const handleLogout = () => {
71     localStorage.removeItem('user');
72     window.location.href = '/login';
73   };
74
75   return (
76     <div>
77       <h1>Profil</h1>
78       <div>
79         <div><img alt="User profile picture" /></div>
80         <div>
81           <div>Nom :</div>
82           <div>{user.nom}</div>
83           <div>Prénom :</div>
84           <div>{user.prenom}</div>
85           <div>Email :</div>
86           <div>{user.email}</div>
87           <div>Mot de passe :</div>
88           <div>{user.password}</div>
89           <div>Confirmer mot de passe :</div>
90           <div>{user.confirmPassword}</div>
91         </div>
92         <div>
93           <button onClick={handleEdit}>Éditer</button>
94           <button onClick={handleSave}>Sauvegarder</button>
95           <button onClick={handleDelete}>Supprimer</button>
96           <button onClick={handleLogout}>Déconnexion</button>
97         </div>
98       </div>
99     </div>
100   );
101 }
```

VII. Frontend - Interfaces



Footer.jsx

```
src > Footer.js > Footer.jsx > Footer.js
1  import React from 'react';
2  import './Footer.css';
3
4  const Footer = () => {
5    return (
6      <Footer>
7        <div>
8          <div>
9            <a href="https://www.yourwebsite.com">
10              Votre nom
11            </a>
12            <div>
13              <div>
14                <a href="https://www.yourwebsite.com">
15                  Données personnelles
16                </a>
17                <a href="https://www.yourwebsite.com">
18                  Gestion des cookies
19                </a>
20                <a href="https://www.yourwebsite.com">
21                  Mentions légales & CGU
22                </a>
23                <a href="https://www.yourwebsite.com">
24                  Documentation technique
25                </a>
26                <a href="https://www.yourwebsite.com">
27                  Aide
28                </a>
29              </div>
30            </div>
31          </div>
32        </div>
33      </Footer>
34    )
35  }
36
37  export default Footer;
```



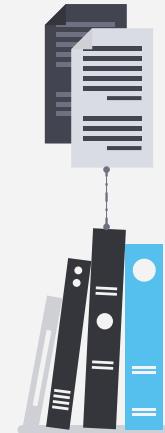
VIII. Développement Backend

VIII. Backend - Architecture & modules python

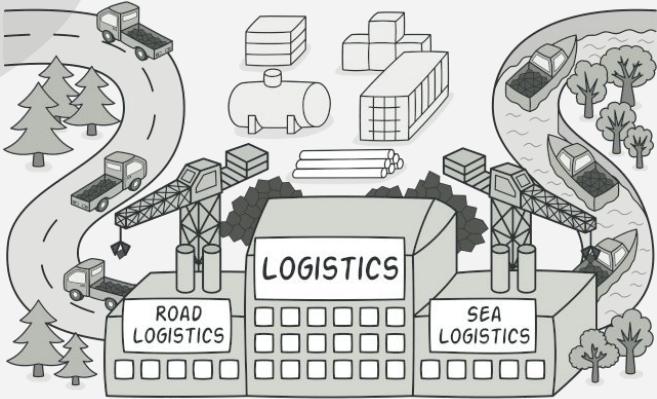
requirements.txt

Uploaded using RayThis Extension

```
psycopg2-binary  
python-dotenv  
pytest  
pytest-cov  
coverage  
bcrypt  
pyjwt  
firebase-admin
```



VIII. Backend - Architecture & modules python



Uploaded using RayThis Extension

```
class Bdd:
    """
    classe de connexion a la bdd pgsql
    """
    def __init__(self, user, mdp, nom, host, port):
        self.user = user
        self.mdp = mdp
        self.nom = nom
        self.host = host
        self.port = port

    def connexion(self):
        """
        connexion a la bdd pgsql
        :return: None
        """
        try:
            self.conn = psycopg2.connect(
                database=self.nom,
                user=self.user,
                password=self.mdp,
                host=self.host,
                port=self.port,
            )
            self.cursor = self.conn.cursor()
            print("[INFO]: Bdd OK")
        except Exception as e:
            print(f"[ERREUR]: Bdd KO : {e}")

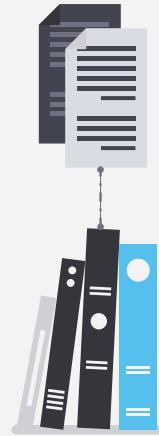
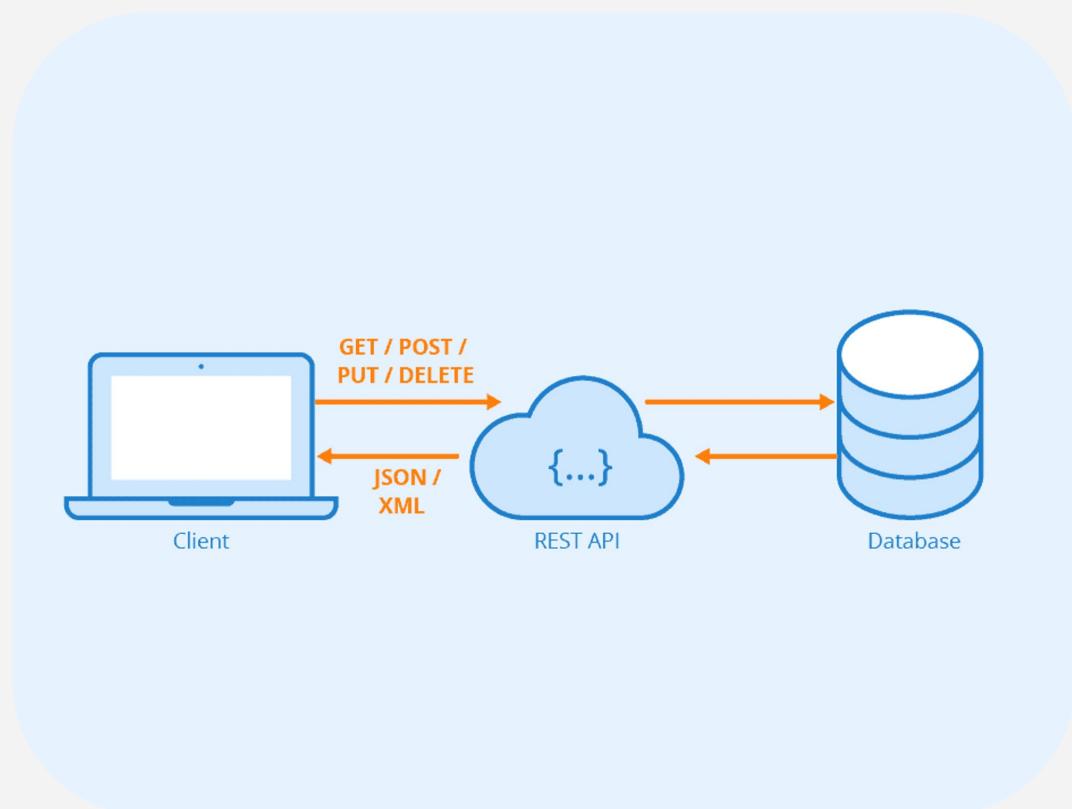
    def close(self):
        self.conn.close()
        self.cursor.close()
        print("[INFO]: Bdd close")

class Logs:
    """
    classe de connexion a la bdd firebase
    """
    def __init__(self, cle):
        self.cle = credentials.Certificate(cle)
        self.app = firebase_admin.initialize_app(self.cle)
        self.db = firestore.client()
```

bdd/connexion.py



VIII. Backend - API RESTful



VIII. Backend - API RESTful

serveur.py

```
...  
Uploaded using RayThis Extension  
  
def do_GET(self):  
    """  
    traite les req GET  
  
    :return: None  
    """  
    path, parametres = self.recuperer_parametres()  
  
    if self.path == '/':  
        self.send_response(200)  
        self.send_header('Content-type', 'text/html')  
        self.end_headers()  
        self.wfile.write('Serveur OK'.encode('utf-8'))  
  
    elif self.path == '/swagger.json':  
        self.send_response(200)  
        self.send_header('Content-type', 'application/json')  
        self.end_headers()  
        self.wfile.write(json.dumps(swagger_spec).encode('utf-8'))
```

serveur.py

```
...  
Uploaded using RayThis Extension  
  
# Threading pcq sinon il galere quand ya plusieurs reqs  
serveur = ThreadedHTTPServer((HOST, PORT), Serveur)  
print(f'le serveur ecoute sur http://:{HOST}:{PORT}')  
# lance le serveur  
serveur.serve_forever() # tourne en continu  
serveur.server_close() # ferme le serveur si ya une action de l'utilisateur  
print('stop')
```



serveur.py

```
...  
Uploaded using RayThis Extension  
  
class Serveur(BaseHTTPRequestHandler):  
    """  
    classe du serveur  
    """  
    #FIXME: elif ==> match case?  
    def recuperer_parametres(self):  
        """  
        recuper les params de la requete  
  
        :return: tuple, (path, parametres)  
        """  
        url = urllib.parse.urlparse(self.path)  
        path = url.path  
  
        parametres = {}  
        if url.query:  
            req_parametres = urllib.parse.parse_qs(url.query)  
  
            for cle, valeur in req_parametres.items():  
                parametres[cle] = valeur[0]  
        return path, parametres  
  
    def end_headers(self):  
        """  
        ajoute les headers CORS (accessible pour le front)  
  
        :return: None  
        """  
        # front a le droit de faire des reqs  
        self.send_header('Access-Control-Allow-Origin', 'http://localhost:3000')  
        # les methodes autorisee  
        self.send_header('Access-Control-Allow-Methods', 'GET, POST, PUT, OPTIONS')  
        # les headers autorises  
        self.send_header('Access-Control-Allow-Headers', 'Content-Type, Authorization')  
        # autorise les cookies  
        self.send_header('Access-Control-Allow-Credentials', 'true')  
        # duree d'1heure  
        self.send_header('Access-Control-Max-Age', '3600')  
        super().end_headers()  
  
    def do_OPTIONS(self):  
        """  
        traite les req options  
  
        :return: None  
        """  
        self.send_response(200)  
        self.end_headers()
```



VIII. Backend - API RESTful

profil/voir_son_profile.py



Uploaded using RayThis Extension

```
return {'status': 200, 'infos': utilisateurs}
```

requête postman pour voir son profil

Body Cookies Headers (8) Test Results

{ } JSON

Preview Visualize

```
1 {  
2   "status": 200,  
3   "infos": [  
4     {  
5       "id": 8,  
6       "mail": "test@test.com",  
7       "nom": "testa",  
8       "prenom": "testa",  
9       "role": "admin",  
10      "date_creation": "2025-05-08 00:58:14",  
11      "num_telephone": "0652103302",  
12      "id_coloc": null  
13    }  
14  ]  
15 }
```



VIII. Backend - SQL

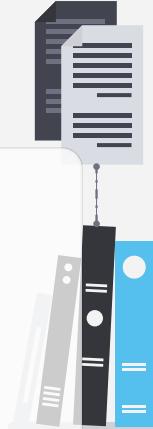
colocations/voir_utilisateurs.py

```
...  
Uploaded using RayThis Extension  
  
def recuper_infos(data):  
    """  
    recupere les informations des colocataires d'une coloc  
  
    :param data: dict, infos envoyées par le serveur  
  
    :return: list, infos des colocataires  
    """  
    coloc = data['id_colocs']  
    requete = """  
    SELECT u.id, u.nom, u.prenom, u.mail  
    FROM Utilisateurs AS u  
    JOIN Colocs AS c  
    ON c.id = u.id_coloc  
    WHERE c.id = %s"""  
    con.cursor.execute(requete, (coloc,))  
    infos = [  
        {  
            'id': row[0],  
            'nom': row[1],  
            'prenom': row[2],  
            'mail': row[3]  
        }  
        for row in con.cursor.fetchall()  
    ]  
    return infos
```



colocations/taches/voir_completes.py

```
...  
Uploaded using RayThis Extension  
  
requete = """  
SELECT t.*  
FROM Taches AS t  
JOIN Utilisateurs AS u  
ON t.createur = u.id  
JOIN Colocs AS c  
ON c.id = u.id_coloc  
WHERE c.id = %s  
AND t.cloture = 'true'  
ORDER BY t.date_crea DESC  
LIMIT 4;"""
```



VIII. Backend - NoSQL

register.py



Uploaded using RayThis Extension

```
log = {
    'date': datetime.now(),
    'action': 'inscription',
    'mail': data['mail']
}
logs.db.collection('Logs').add(log)
```



adm/voir_logs.py



Uploaded using RayThis Extension

```
logs_bdd = logs.db.collection('Logs').stream()
# recuper TOUTEs les logs
log = []
for l in logs_bdd:
    log.append({
        'id': l.id,
        'date': l.to_dict().get('date').strftime('%Y-%m-%d %H:%M:%S'),
        # .strftime mets au bon format
        'action': l.to_dict().get('action'),
        'infos': [
            'id_coloc': l.to_dict().get('id_coloc'),
            'id_utilisateur': l.to_dict().get('id_utilisateur'),
            'id_logs': l.to_dict().get('id_logs')
        ]
    })
# transforme les objets de firebase en dict
```



IX. Sécurité

IX. Sécurité - Rôles

Broken Access Control (Top 1 OWASP 2023)

The screenshot shows a REST client interface with the following details:

- URL:** `http://localhost:8000/adm/logs?token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJtYWlsjoibWF0aGlidUBjb2xvYy5mcilsInJvbGUIoJyZXNwb25zYWJsZSJ9.DwlsPnO05cxnqAlRZKixfcQj4Pnb5i4U_Irc5xWav8&role=...`
- Method:** GET
- Params:** Key, token, role, mail
- Headers:** (8)
- Body:** (Empty)
- Script:** (Empty)
- Settings:** (Empty)
- Cookies:** (Empty)
- Query Params:** (Empty)
- Response Status:** 403 Forbidden
- Response Headers:** 44 ms, 405 B
- Response Body:**

```
1 { "status": 403,
2   "message": "Role KO"
3 }
4 }
```



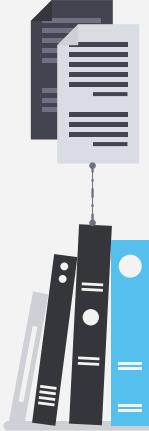
IX. Sécurité - Rôles

la majorité de mes fichiers backend



Uploaded using RayThis Extension

```
if token_decode['role'] not in ['admin']:  
    return {'status': 403, 'message': 'Role KO'}
```



App.jsx



Uploaded using RayThis Extension

```
if (parsedUser.role === 'aucun') {  
    setShowRoleModal(true);  
}  
  
if (parsedUser.role === 'admin' && currentPage !== 'admin') {  
    window.history.pushState({}, '', '/admin');  
    setCurrentPage('admin');  
}
```



IX. Sécurité - Mots de passe sécurisé

Cryptographic Failures (Top 2 OWASP 2023)

The screenshot shows a browser-based testing tool interface. At the top, it displays the URL `http://localhost:8000/register`. Below the URL, there are tabs for `POST`, `Params`, `Authorization`, `Headers (10)`, and `Body`. The `Body` tab is selected, showing the following JSON payload:

```
1 {  
2     "nom": "test",  
3     "prenom": "test",  
4     "mail": "test@test.com",  
5     "mdp": "te",  
6     "csrf": "cz6hyCmAUIU7D1htACJKe2Hw"  
7 }
```

Below the payload, there are tabs for `Params`, `Authorization`, `Headers (10)`, `Body`, `Scripts`, and `Settings`. The `Body` tab has sub-options: `none`, `form-data`, `x-www-form-urlencoded`, `raw` (selected), `binary`, `GraphQL`, and `JSON`. The `JSON` tab is also visible.

The response section shows a `400 Bad Request` status with the message `"Mdp court"`.

IX. Sécurité - Mots de passe sécurisé

Combien de temps faut-il à un pirate pour trouver votre mot de passe **2025**

12 x RTX 5090 | bcrypt (10)

Nombre de caractères	Nombres seulement	Lettres minuscules	Lettres majuscules et minuscules	Nombres, lettres majuscules et minuscules	Nombres, lettres majuscules et minuscules, symboles
4	Instantané	Instantané	Instantané	Instantané	Instantané
5	Instantané	Instantané	57 minutes	2 heures	4 heures
6	Instantané	46 minutes	2 jours	6 jours	2 semaines
7	Instantané	20 heures	4 mois	1 an	2 ans
8	Instantané	3 semaines	15 ans	62 ans	164 ans
9	2 heures	2 ans	791 ans	3k ans	11k ans
10	1 jour	40 ans	41k ans	238k ans	803k ans
11	1 semaine	1k ans	2M ans	14M ans	56M ans
12	3 mois	27k ans	111M ans	917M ans	3Md ans
13	3 ans	705k ans	5Md ans	56Md ans	275Md ans
14	28 ans	18M ans	300Md ans	3Bn ans	19Bn ans
15	284 ans	477M ans	15Bn ans	218Bn ans	1Bd ans
16	2k ans	12Md ans	812Bn ans	13Bd ans	94Bd ans
17	28k ans	322Md ans	42Bd ans	840Bd ans	6Tn ans
18	284k ans	8Bn ans	2Tn ans	52Tn ans	463Tn ans

register.py

```
...  
Uploaded using RayThis Extension  
  
def car_spe(mdp):  
    """  
    verifie que les mdp ont au moins 1 caractère spécial  
  
    :param mdp: str, mot de passe de l'utilisateur  
  
    :return: bool, True si le mot de passe contient au moins un caractère spécial, False sinon  
    """  
    caracteres = "!@#$%^&*(),.?":{}|<>"  
    for c in caracteres:  
        if c in mdp:  
            return True  
    return False
```

...
Uploaded using RayThis Extension

```
if not car_spe(data['mdp']):  
    return {'status': 400, 'message': 'Car speciaux KO'}  
  
if len(data['mdp']) < 14:  
    return {'status': 400, 'message': 'Mdp court'}
```



Hive Systems

hivesystems.com/password



IX. Sécurité - Mots de passe sécurisé

register.py

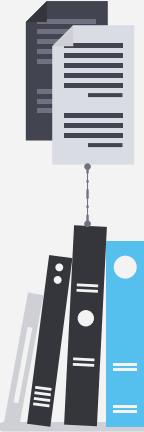


Uploaded using RayThis Extension

```
def mdp_hash(mdp):
    """
        hashe le mot de passe de l'utilisateur avec bcrypt

    :param mdp: str, mot de passe de l'utilisateur

    :return: str, mot de passe hashe
    """
    mdp_propre = mdp.encode('utf-8')
    salt = bcrypt.gensalt(12) # 12 --> eviter bruteforce ralentir l'algo
    mdp_hashe = bcrypt.hashpw(mdp_propre, salt)
    return mdp_hashe
```



IX. Sécurité - Mots de passe sécurisé

Register.jsx



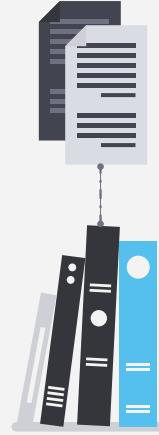
Uploaded using RayThis Extension

```
if (email !== confirmPassword) {
    setError('Les adresses email ne correspondent pas');
    return;
}

if (password !== confirmPassword) {
    setError('Les mots de passe ne correspondent pas');
    return;
}

if (password.length < 14) {
    setError('Le mot de passe doit contenir au moins 14 caractères');
    return;
}

const specialChars = /[!@#$%^&*(),.?":{}|<>]/;
if (!specialChars.test(password)) {
    setError('Le mot de passe doit contenir au moins un caractère spécial');
    return;
}
```



IX. Sécurité - Mots de passe sécurisé



	id [PK] integer	mail character varying (50)	nom character varying (50)	prenom character varying (50)	role character varying (50)	mdp character varying (255)
1	13	mathieu@coloc.com	mathieu	coloc	colocataire	\x24326224313224562e4a6f484d4c716e553963353235324f4f772e772e322f3664754977303169515678593078556b51534532792e69346a6d647...
2	11	bibou@bib.com	bibou	bibou	admin	\x24326224313224343072357650624e614b6c42a44734943446a52657556474e3644444737761394d444a6e4141303050622f3777635075356f7...
3	16	mathieu@coloc.fr	mathieu coloc	mathieu coloc	responsable	\x2432622431322454517535664e4c6b6b495442717674516444a4a6c4f445a3552326a6e5a4966426d5272444f6954765944465a5764364d71426...
4	17	mathieu@coloc2.fr	coloc2	coloc2	colocataire	\x24326224313224322f323162417079486853307073366d686a7463572e32794e7832784a66315a5048766c467950425764643775692e6d6b7579...
5	10	caca@caca.com	cacio	cacioooo	responsable	\x2432622431322439476f31787442334e59716448644d4a317342626e6565395241534a58576663735a444952707545734e685a66336859414a...
6	8	test@test.com	testa	testa	admin	\x2432622431322448494c697a6b2e616a3436385748346145394e62434f556b7950676478626a712f38595a6331615836344e3751706253634b483...



IX. Sécurité - Injection XSS

Register.jsx

Injection XSS (Top 3 OWASP 2023)



Uploaded using RayThis Extension

```
<input
```

```
    type="text"
    id="nom"
    className="form-input"
    value={nom}
    onChange={(e) => setNom(e.target.value)}
    required
    autoComplete="nom-de-famille"
/>

```



IX. Sécurité - Injection SQL

coloc/creation.py

```
... Uploaded using RayThis Extension

requete = """
INSERT INTO Colocs (nom, date_crea, responsable) VALUES (%s, %s, %s) RETURNING id
"""

# RETURNING id ==> recup l'id de la coloc cree
param = (data['nom'], datetime.now(), id_utilisateur)
con.cursor.execute(requete, param)
id_coloc = con.cursor.fetchone()

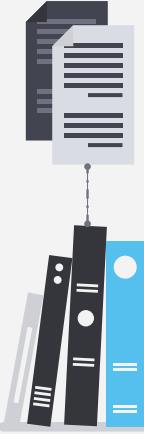
if not id_coloc:
    return {'status': 500, 'message': 'Erreur interne'}

requete2 = """UPDATE Utilisateurs SET id_coloc = %s WHERE id = %s"""
con.cursor.execute(requete2, (id_coloc, id_utilisateur))
# lorsque l'user cree une coloc, il en devient responsable

log = {
    'date': datetime.now(),
    'action': 'creation coloc',
    'id_utilisateur': id_utilisateur,
    'id_coloc': id_coloc
}
logs.db.collection('Logs').add(log)

# FIXME: verifier si les actions passent avant et en fonction commit/rollback ?
con.conn.commit()
```

Injection SQL (Top 4 OWASP 2023)



IX. Sécurité - Configuration



.env

Security Misconfiguration (Top 5 OWASP 2023)



Uploaded using RayThis Extension

BDD_NOM=Cooloc

BDD_PORT=5432

BDD_HOST=localhost

BDD_USER=postgres

BDD_MDP=postgres

JWT=a-string-secret-at-least-256-bits-long

bdd/connexion.py



Uploaded using RayThis Extension



load_dotenv()

```
con = Bdd(user=os.getenv("BDD_USER"), mdp=os.getenv("BDD_MDP"), nom=os.getenv("BDD_NOM"),
host=os.getenv("BDD_HOST"), port=os.getenv("BDD_PORT"))
```

IX. Sécurité - Authentification



Broken Authentication (Top 7 OWASP 2023)

Cooloc / Login/Register / <http://localhost:8000/login>

POST | http://localhost:8000/login

Params Authorization Headers (10) Body Scripts Settings

Body Type: raw | x-www-form-urlencoded | JSON | GraphQL | Beautify

```
1 {
2   "mail": "mathieu@coloc.fr",
3   "mdp": "mathieu coloc@1234",
4   "csrf": "gQS_QLRI1gzS9Fk-hT6ozC-kfwPxx-mQkQUEsJLunQQ"
5 }
6 }
```

Body Cookies Headers (8) Test Results

200 OK | 329 ms | 660 B | Save Response | ⋮

{ } JSON | Preview | Visualize | ⌂

```
1 {
2   "status": 200,
3   "data": {
4     "mail": "mathieu@coloc.fr",
5     "role": "responsable",
6     "prenom": "mathieu coloc",
7     "nom": "mathieu coloc",
8     "id_coloc": 27
9   },
10   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJtYWlsIjoiZW1haWwiLCJpY3QiOiJ3bGU1O1JyZXNwb25zY2hpdHJ9.DwIsPh005cxnqAlRZK1xfcQ3q4Pnb6i4U_1zc5xNav8"
11 }
```



IX. Sécurité - Authentification



login.py



Uploaded using RayThis Extension

```
def create_token(data):
    """
    encode avec l'ago HS256 un token jwt avec le mail & role de l'user

    :param data: dict, infos envoyées par le serveur

    :return: str, token JWT
    """
    return jwt.encode({'mail': data['mail'], 'role': data['role']}, jwt_secret,
                      algorithm=jwt_algo)
```



IX. Sécurité - Authentification

profil/voir_profil.py



Uploaded using RayThis Extension

```
def verifier_token(data, token):
    """
    verifie le token JWT fourni par l'user

    :param data: dict, infos infos envoyées par le serveur
    :param token: str, token JWT de l'user

    :return: dict, status et message de la verification du token
    """
    token_decode = jwt.decode(token, jwt_secret, algorithms=[jwt_algo])

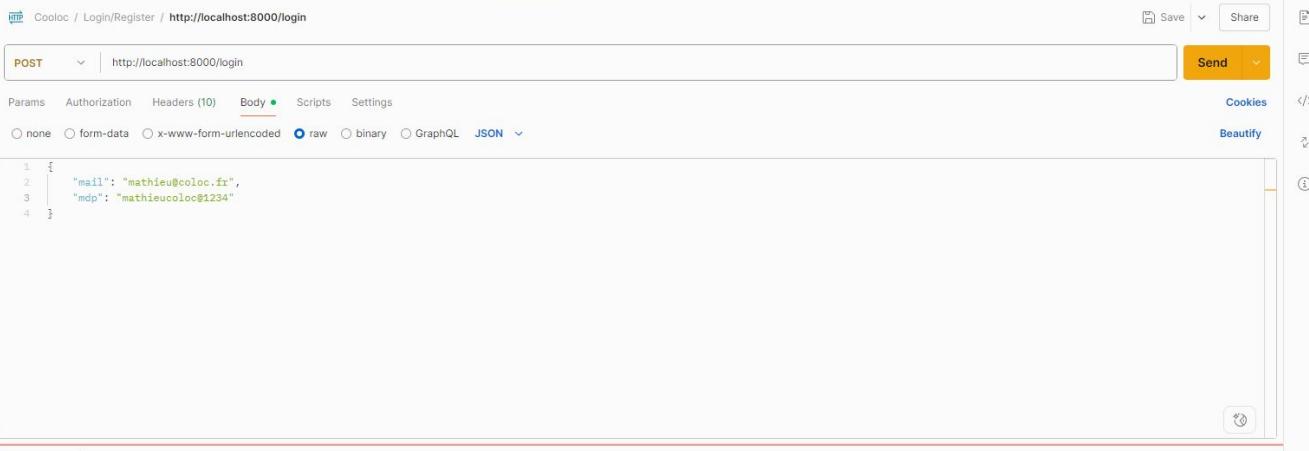
    if token_decode['mail'] != data['mail']:
        return {'status': 403, 'message': 'Token KO'}

    return {'status': 200, 'message': 'Token OK'}
```



IX. Sécurité - CSRF

Cross Site Request Forgery (Top 8 OWASP 2023)



The screenshot shows a POST request to `http://localhost:8000/login`. The request body is a JSON object with two fields: `mail` and `mdp`, both set to their respective values.

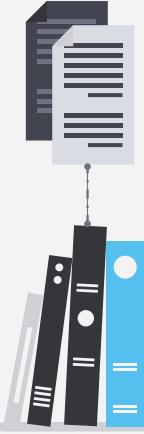
```
1 {  
2     "mail": "mathieu@coloc.fr",  
3     "mdp": "mathieu@coloc@1234"  
4 }
```

The response section indicates that the request failed, showing a rocket icon and the message "Could not get response". A troubleshooting link "View in Console" is provided.

Could not get response

Error: socket hang up | [View in Console](#) |  What's wrong?

[Learn more about troubleshooting API requests](#)



IX. Sécurité - CSRF

login.py



Uploaded using RayThis Extension

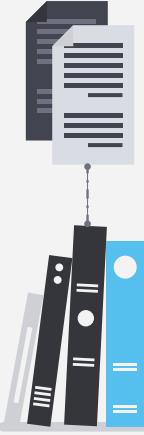
```
def verifier_csrf(data):
    """
    verifie le token CSRF

    :param data: dict, infos envoyées par le serveur

    :return: dict, status et message de la verification du token CSRF
    """
    csrf = data['csrf']

    if not csrf:
        return {'status': 403, 'message': 'CSRF KO'}

    return {'status': 200, 'message': 'CSRF OK'}
```



IX. Sécurité - CSRF

Login.jsx



Uploaded using RayThis Extension

```
body: JSON.stringify({  
    mail: email,  
    mdp: password,  
    csrf: 'cz6hyCmAUIU7D1htACJKe2HwfE6bqAiksEOYJABM3-Y'  
}),
```



IX. Sécurité - RGPD



admin/supprimer_utilisateur.py



Uploaded using RayThis Extension

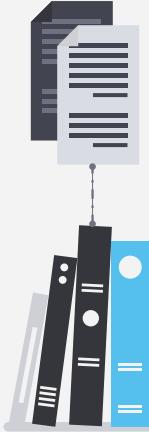
```
requete = """DELETE FROM Utilisateurs WHERE id = %s"""
con.cursor.execute(requete, (data['id_utilisateur_supprime'],))
```

profil/supprimer.py



Uploaded using RayThis Extension

```
requete = """DELETE FROM Utilisateurs WHERE id = %s"""
con.cursor.execute(requete, (id_utilisateur,))
```



X. DevOps

X. DevOps - Conteneurisation

backend/Dockerfile

```
...  
Uploaded using RayThis Extension  
  
FROM python:3.9-slim-bookworm  
WORKDIR /app  
  
RUN pip install uv  
  
COPY backend/requirements.txt backend/requirements.txt  
RUN uv pip install --system -r backend/requirements.txt  
  
COPY . .  
  
EXPOSE 8000  
CMD ["python", "server.py"]
```

docker-compose

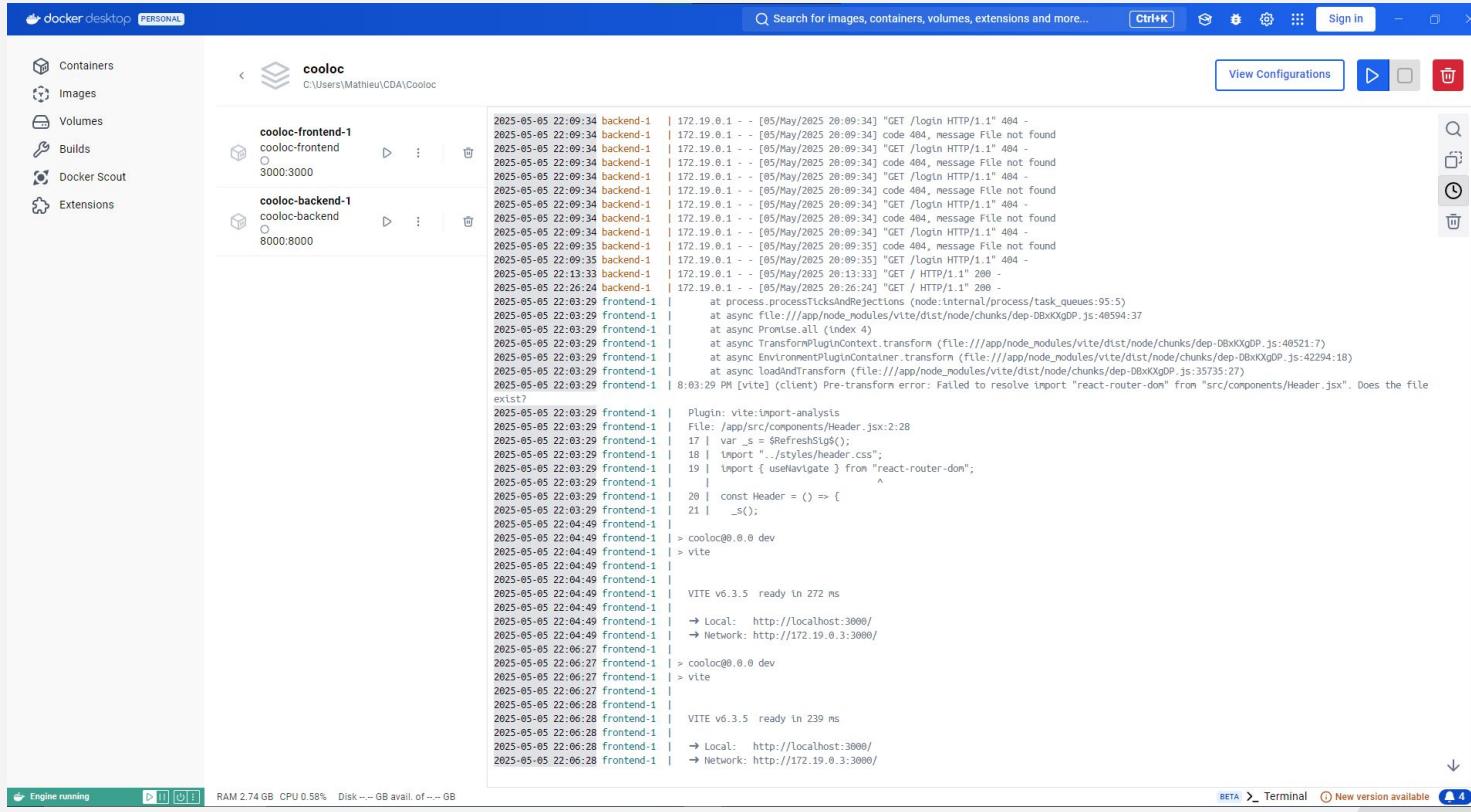
```
...  
Uploaded using RayThis Extension  
  
services:  
  backend:  
    build:  
      context: .  
      dockerfile: backend/Dockerfile  
    ports:  
      - "8000:8000"  
    volumes:  
      - ./backend:/app  
    environment:  
      - ENV=development  
  
  frontend:  
    build:  
      context: .  
      dockerfile: frontend/Dockerfile  
    ports:  
      - "3000:3000"  
    volumes:  
      - ./frontend:/app  
      - /app/node_modules  
    depends_on:  
      - backend
```

frontend/Dockerfile

```
...  
Uploaded using RayThis Extension  
  
FROM node:20-alpine  
WORKDIR /app  
  
COPY frontend/package*.json ./  
RUN rm -rf node_modules  
RUN npm install  
  
COPY frontend/ .  
RUN npm run build  
  
EXPOSE 3000  
CMD ["npm", "run", "dev"]
```



X. DevOps – Conteneurisation



X. DevOps - Tests & validations

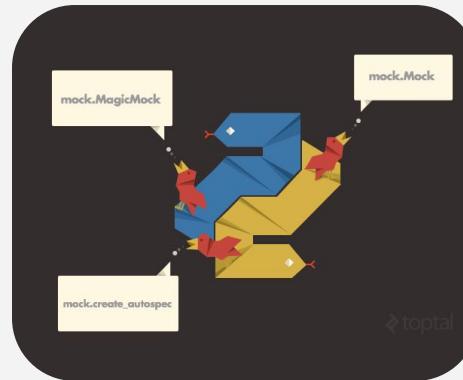


Uploaded using RayThis Extension

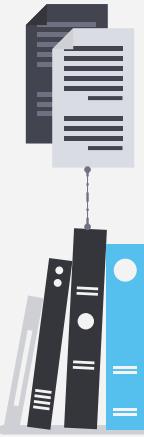
```
import pytest
from unittest.mock import MagicMock, patch

@pytest.fixture(autouse=True)
def mock_firebase():
    with patch('firebase_admin.initialize_app') as mock_init, \
        patch('firebase_admin.credentials.Certificate') as mock_cert, \
        patch('firebase_admin.firebaseio') as mock_firestore, \
        patch('psycopg2.connect') as mock_connect:

        mock_db = MagicMock()
        mock_firestore.client.return_value = mock_db
        mock_cert.return_value = MagicMock()
        mock_connect.return_value = MagicMock()
        yield {
            'init': mock_init,
            'cert': mock_cert,
            'firebase': mock_firestore,
            'db': mock_db,
            'connect': mock_connect
        }
```



▼	File	Login/Register	...
		POST http://localhost:8000/login	
		POST http://localhost:8000/register	
▼	Coloc	GET SEL - Voir les utilisateurs d'une c...	
		GET SEL - Voir les tâches dispos	
		GET SEL - Voir les tâches complètes	
		GET SEL - Voir mes tâches	
		GET SEL - Voir les infos coloc	
		POST INS - Créeation d'une coloc	
		POST INS - Créeation d'une tâche	
		POST DEL - Supprimer Coloc	
		POST DEL - Supprimer Tâches	
		PUT MAJ - Nom coloc	
		PUT MAJ - Nom tache	
		PUT MAJ - Clôturer tache	
		PUT MAJ - S'attribuer une tâche	
		PUT MAJ - Ajouter coloc	
		PUT MAJ - Supprimer un coloc	
▼	Utils	GET SEL - Profil	
		PUT MAJ - Role	
		PUT MAJ - Tel	
		PUT MAJ - Profil	
▼	Admin	...	
		File	
		Utilisateurs	
		GET SEL - Voir toutes les logs	
		POST DEL - Supprimer une logs	



X. DevOps - SonarQube & GitHub Actions

All workflows

Showing runs from all workflows

8 workflow run results

- ✓ Update notes.md Build and SonarCloud Analysis #38: Commit e40589a pushed by MathieuAudibert
- ✓ Update notes.md Build and SonarCloud Analysis #37: Commit 1475dd1 pushed by MathieuAudibert
- ✓ Update notes.md Build and SonarCloud Analysis #36: Commit 5b52de0 pushed by MathieuAudibert
- ✓ Update notes.md Build and SonarCloud Analysis #35: Commit 30de983 pushed by MathieuAudibert
- ✓ Update notes.md Build and SonarCloud Analysis #34: Commit 9129580 pushed by MathieuAudibert
- ✓ Update notes.md Build and SonarCloud Analysis #33: Commit b4662b8 pushed by MathieuAudibert
- ✓ docs: sonarqube & git actions OK Build and SonarCloud Analysis #32: Commit 67cfe4f pushed by MathieuAudibert
- ✓ ci: app.test Build and SonarCloud Analysis #31: Commit 0c13897 pushed by MathieuAudibert



GitHub Actions

SonarQube

My Projects My Issues Explore

Event Status Branch Actor

Upgrade

Cap92 > Cooloc > Overview

Cooloc

Public

Overview Main Branch Pull Requests Branches

No tags Last analysis 14 Apr 2025 221 Lines of Code

● CSS ● JavaScript ● Docker ● HTML

Main Branch Status Quality Gate Passed Enjoy your sparkling clean code!

Main Branch Evolution since 2 months ago 1 Issues = Issues Coverage Duplications

See Full Analysis See full history

Information Administration Collapse

Latest Activity NEW ANALYSIS Main Branch Passed

X. DevOps - Documentation technique



http://localhost:8000/swagger-ui/index.html

Swagger 2.0.19

Cooloc API REST de gestion de colocations

Servers

http://localhost:8000 - Serveur python local

Authentification

- POST /login Connexion
- POST /register Inscription

Colocation

- GET /coloc/utilisateurs/voir Voir les utilisateurs d'une colocat
- GET /coloc/voir Voir les informations d'une colocat
- POST /coloc/creer Crée une colocat
- POST /coloc/supprimer Supprimer une colocat
- PUT /coloc/nom/nom Modifier le nom d'une colocat
- PUT /coloc/utilisateurs/sjouter Ajouter des utilisateurs à une colocat

Utilisateur

- GET /profil/voir Voir son profil
- PUT /role/nom Mettre son rôle
- PUT /profil/nom Mettre son profil

Administration

- GET /adm/utilisateurs/voir Voir tous les utilisateurs
- PUT /adm/utilisateurs/nom Mettre un utilisateur
- POST /adm/utilisateurs/supprimer Supprimer un utilisateur
- GET /adm/logs Voir les logs de l'application
- POST /adm/logs/supprimer Supprimer un log

Tâches

- GET /coloc/taches/voir Voir toutes les tâches de la colocat (non clôturé)
- GET /coloc/taches/mesmes Voir les tâches attribuées à l'utilisateur connecté
- GET /coloc/taches/voir_complet Voir les tâches clôturées (à terminer)
- POST /coloc/taches/creation Entrer une tâche
- PUT /coloc/taches/attribuer Attribuer une tâche à un utilisateur
- PUT /coloc/taches/cloturer Clôturer une tâche

Administration

GET /adm/utilisateurs/voir Voir tous les utilisateurs

Parameters

Name	Description
token * required	string (query)
csrf * required	string (query)
role * required	Available values : admin string (query)
mail * required	string (query)

Responses

Code	Description	Links
200	Liste des utilisateurs récupérée	No links
403	Token invalide ou CSRF invalide ou rôle insuffisant	No links

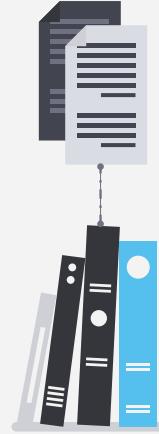
Media type: application/json
Content Accept header: Example Value: Schema

```
{ "status": 0, "data": [ { "id": 1, "mail": "string", "nom": "string", "prenom": "string", "role": "string", "taches": "string", "date_creation": "2021-07-02T14:02:55.778Z", "date_modification": "string", "id_categorie": 1 } ] }
```

XI. Points d'améliorations

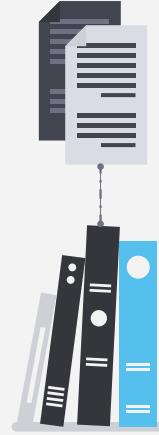
XI. Points d'améliorations - Champs

- ❖ Champs + stricts
- ❖ Contrôles sur **TOUS** les champs



XI. Points d'améliorations - Tokens

- ❖ Pas d'expiration
- ❖ Génération tokens CSRF



XII. Conclusion

Merci

Mathieu AUDIBERT - B3 Développement web & applications

CREDITS: This presentation template was created by [Slidesgo](#), and includes icons, infographics & images by [Freepik](#)