Analyse

Ce document a pour but de définir les contraintes liées à l'application et de schématiser les solutions mises en place pour réaliser l'application.

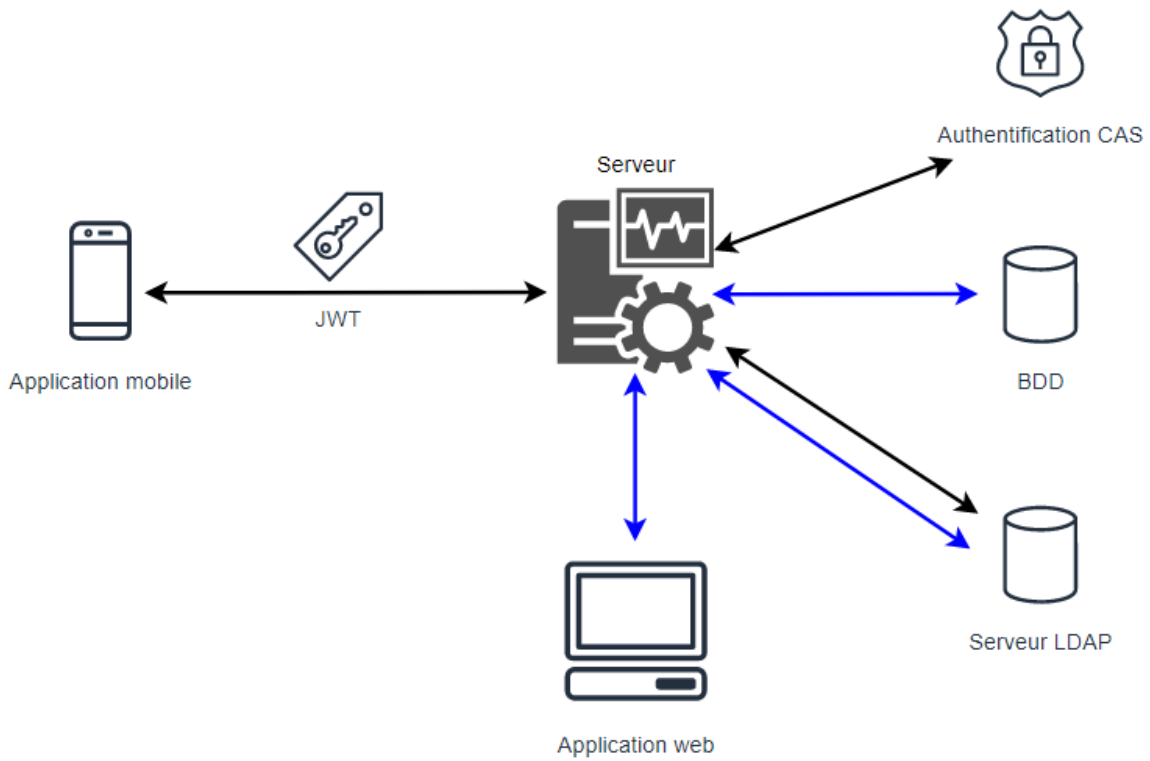
Table des matières

Description de l'application

L'application que nous allons développer a pour objectif de permettre à un personnel de l'université de contrôler l'identité des étudiants qui lors de manifestations, d'entrées à un examen, etc. La personne qui effectue le contrôle devra disposer des autorisations nécessaires et se connecter via l'application, pour vérifier si elle dispose de ces autorisations. Le contrôle d'identité des étudiants se fera grâce au scanne d'un code-barre qui contient les informations de l'étudiant et qui sera présent dans l'application Unvilorraine.

L'application développée communiquera avec un serveur qui se chargera de réaliser l'authentification CAS pour le contrôleur (la personne qui contrôle les identités), ainsi que les différentes actions liées au contrôle de l'étudiant et à la récupération de ces informations (nom, prénom, image, etc). Le serveur aura un second rôle, qui sera de vérifier si le contrôleur dispose des autorisations nécessaires au contrôle d'identité, grâce à une base de données. L'application et le serveur communiqueront de manière sécurisée avec le protocole JWT.

Par la suite, l'application pourra également s'orienter vers le contrôle de présence dans les cours. Ainsi, les professeurs pourront scanner le code-barre des étudiants à chaque entrée et donc vérifier la présence de l'étudiant.

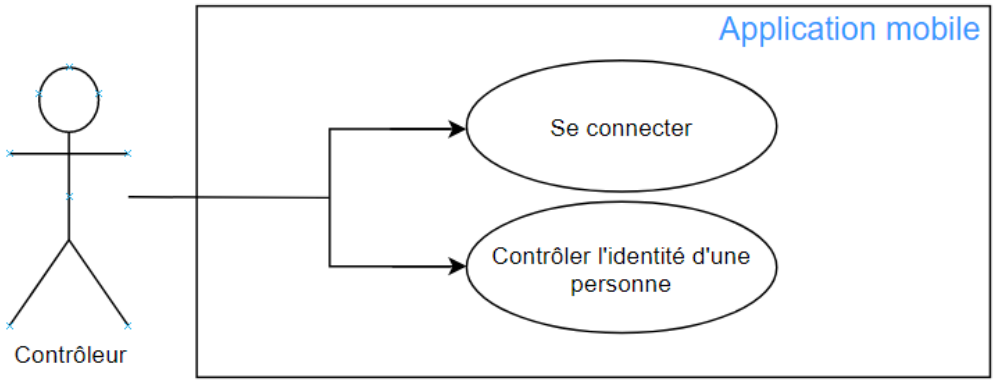


Technologies utilisées

Pour réaliser à bien l'application nous allons utiliser le Framework Flutter qui permettra de réaliser une application pour les plateformes Android et iOS avec un seul code source. Du côté serveur, nous allons utiliser le langage PHP avec le Framework Symfony 4.4, la base de données quant à elle sera en MySQL. Nous utiliserons également le site Draw.io pour réaliser les schémas utiles au développement, ainsi que le site MockFlow pour faire la maquette de l'application.

Cas d'utilisation

Dans sa première version, l'application mobile ne possède que deux cas d'utilisation le premier, étant l'authentification du contrôleur et le second, le contrôle d'identité.



SCHEMA DES CAS D'UTILISATION DE L'APPLICATION MOBILE

CU "Se connecter"

- **Nom** : Se connecter
- **Description** : Le contrôleur se connecte dans l'application pour vérifier son habilitation
- **Acteur principal** : Contrôleur
- **Préconditions** : -
- **Post-condition** : Le contrôleur à accès à l'application et peut contrôler les identités
- **Déroulement normal :**
    1. Le client se connecte
    2. Il peut utiliser la fonctionnalité de contrôle
- **Variantes :**
  1.1 Le contrôleur n'a pas l'habilitation nécessaire
  1.2 L'authentification a échoué
- **Contraintes : -**

CU "Contrôler l'identité d'une personne"

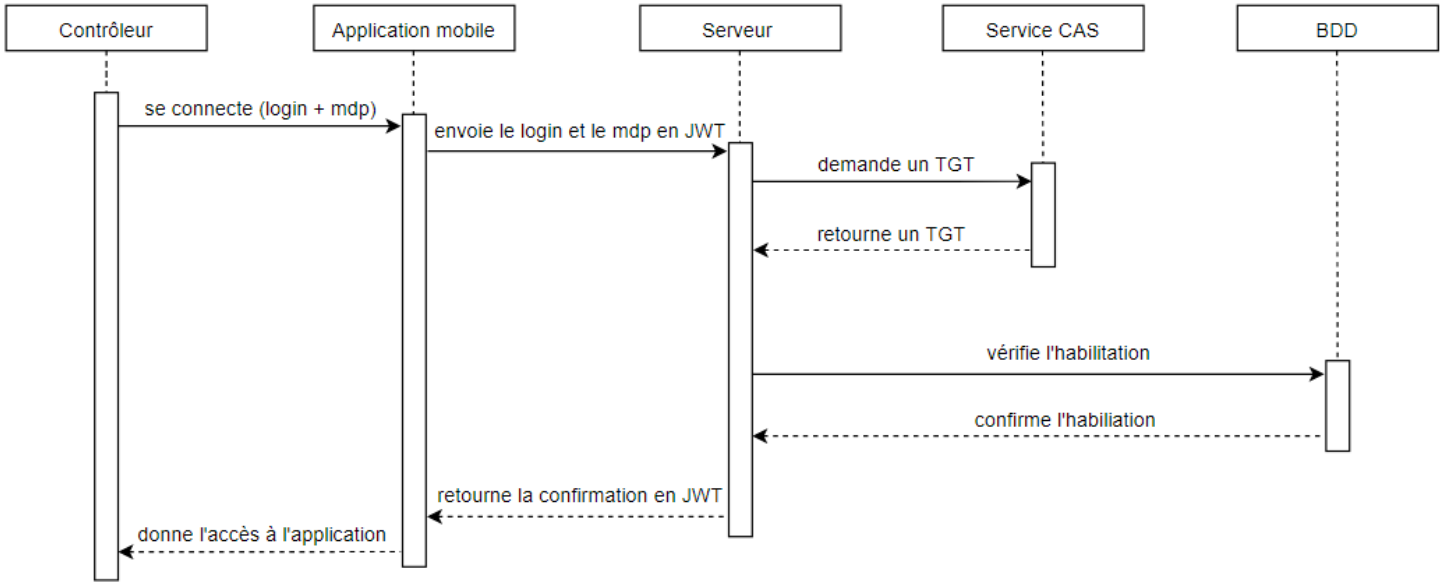
- **Nom** : *Contrôler l'identité d'une personne*
- **Description** : Le contrôleur contrôle l'identité d'une personne via l'application
- **Acteur principal** : Contrôleur
- **Préconditions** :
    - Être connecté dans l'application
- **Post-condition** : Le contrôleur a accès aux informations de la personne
- **Déroulement normal :**
    1. Le contrôleur scanne le code-barre
    2. Il reçoit les informations de la personne
- **Variantes :**
  1.1 Le scanne échoue
  2.1 Les informations ne sont pas transmises
- **Contraintes : -**

CU "Ajout d'une habilitation"

- **Nom** : *Ajout d'une habilitation*
- **Description** : Donner l'habilitation à une personne pour le contrôle des identités
- **Acteur principal** : Administrateur
- **Préconditions** :
    - L'administrateur est autorisé à se connecter à l'application
    - La personne existe dans la base de données de l'université
- **Post-condition** : La personne possède l'habilitation pour contrôler les étudiants
- **Déroulement normal :**
    1. L'administrateur se connecte à l'application web
    2. Il donne l'habilitation à la personne
- **Variantes :**
  1.1 L'administrateur ne peut pas se connecter
  2.1 La personne ne peut pas recevoir l'habilitation
- **Contraintes : -**

Diagramme de séquence

Connexion



Scanne de code-barre

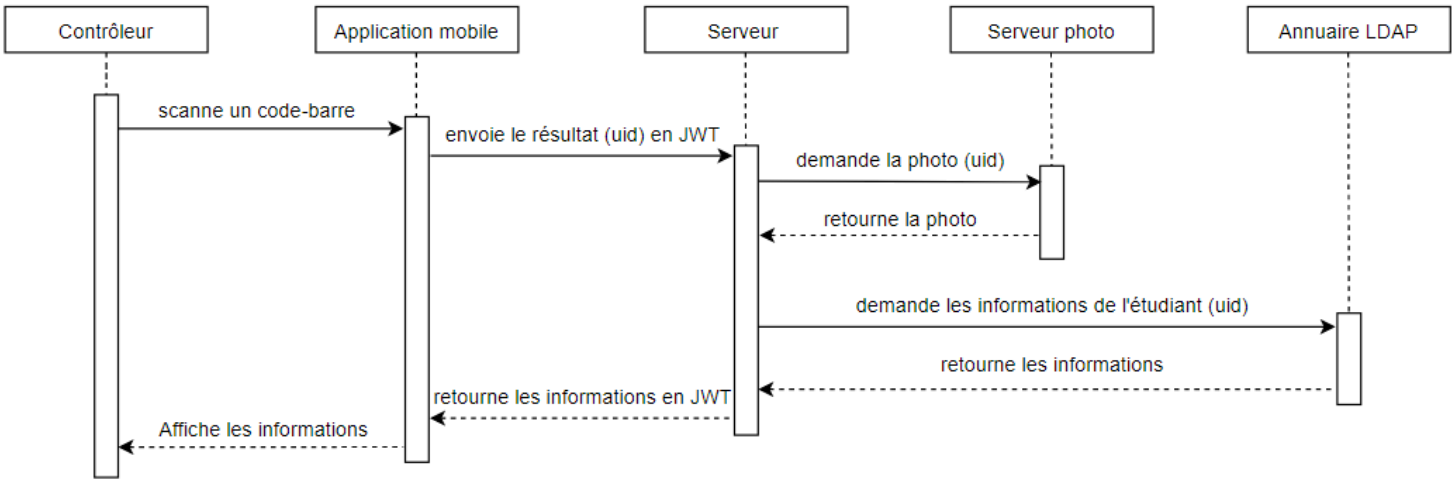
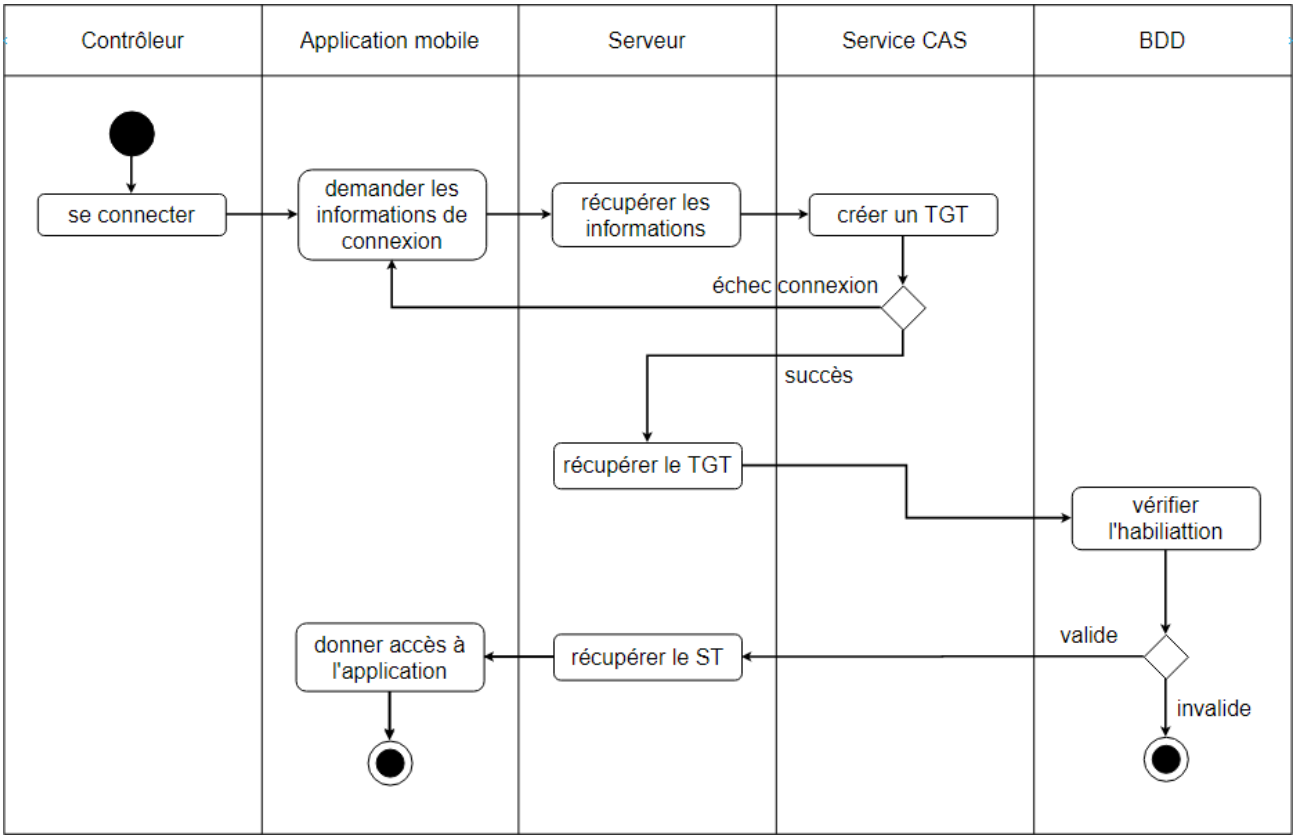
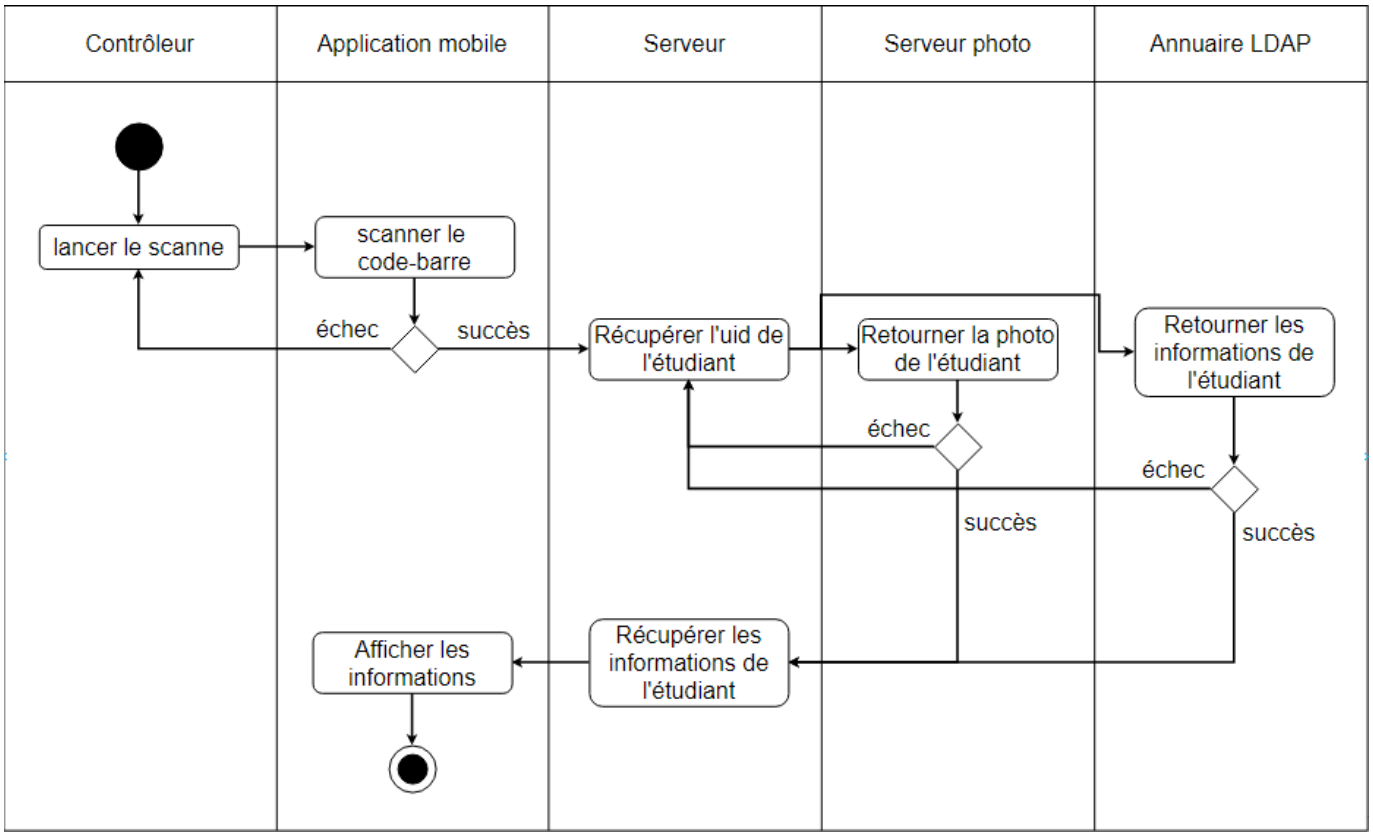


Diagramme d’activité

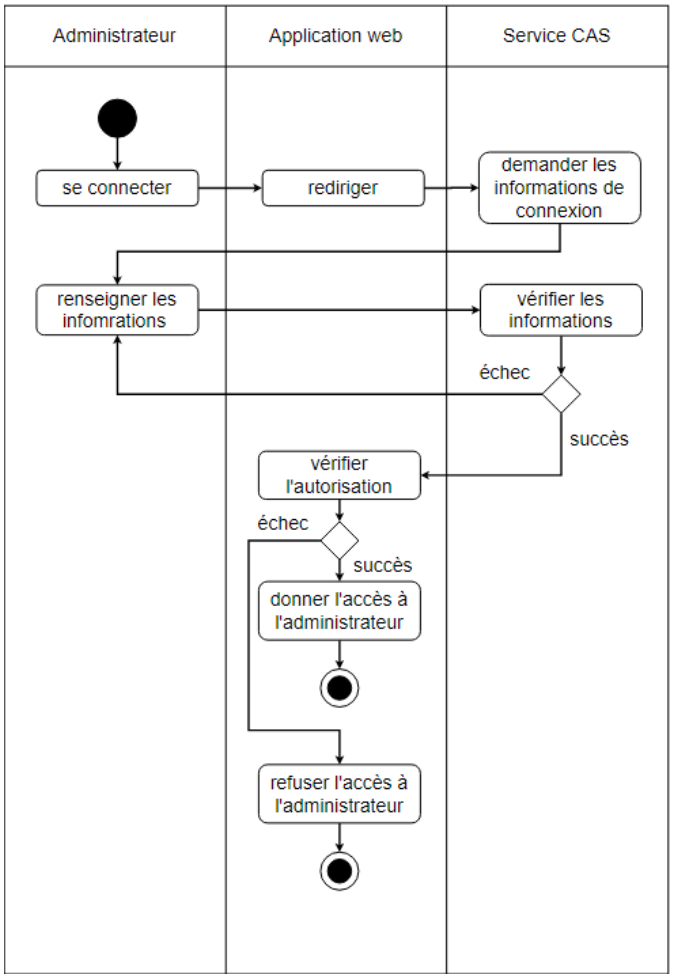
Connexion

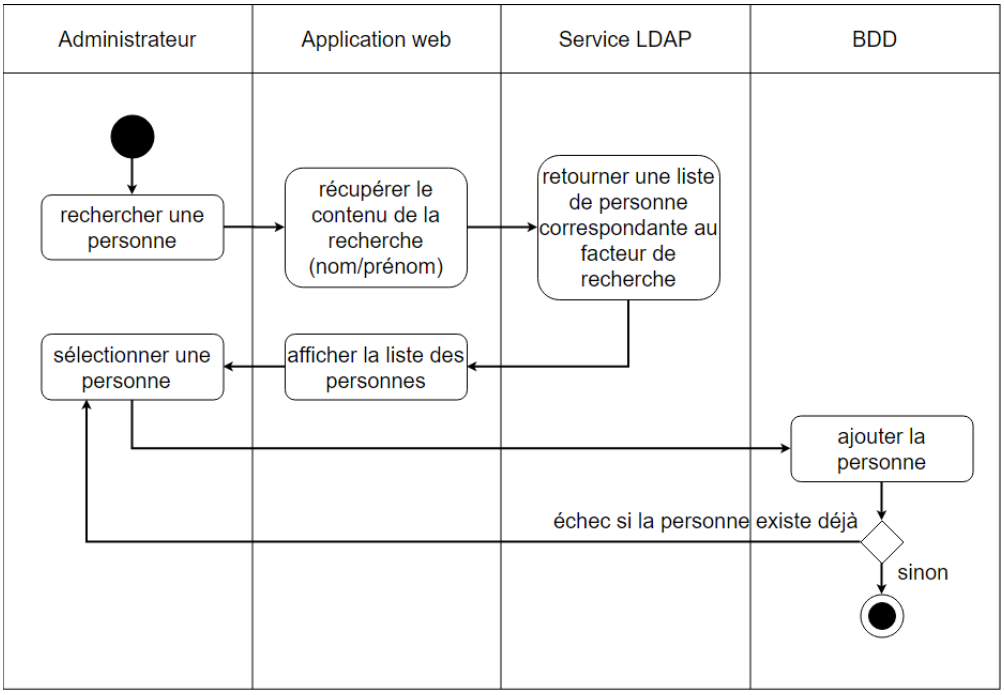


Scanne de code-barre



Connexion et ajout d'une habilitation depuis l'application web



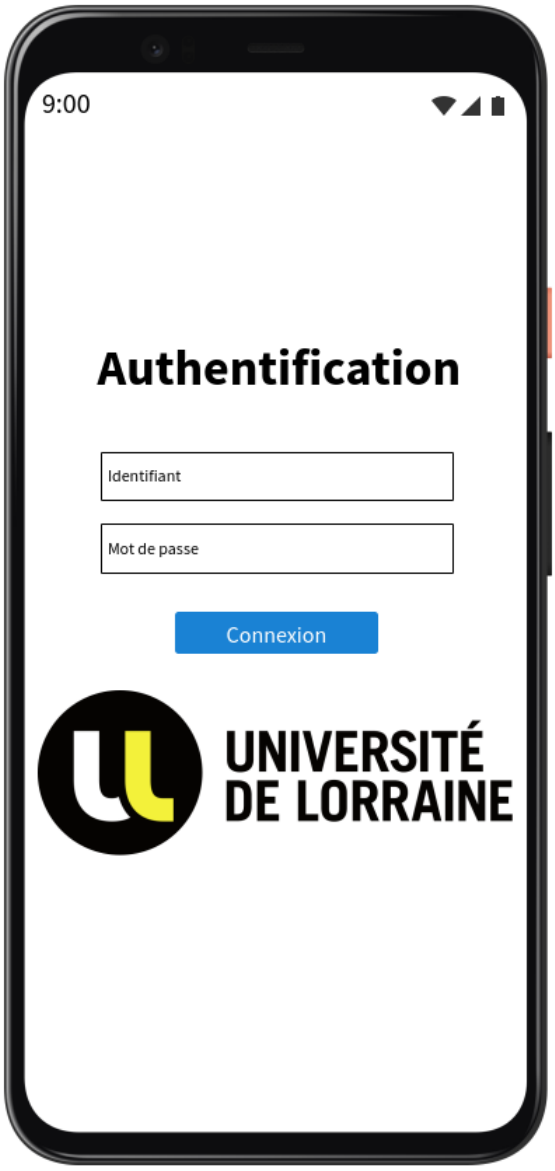


Maquette de l’application mobile

Cette maquette a pour but de donner un aperçu du rendu final de l’application et de modéliser les fonctionnalités de celle-ci.

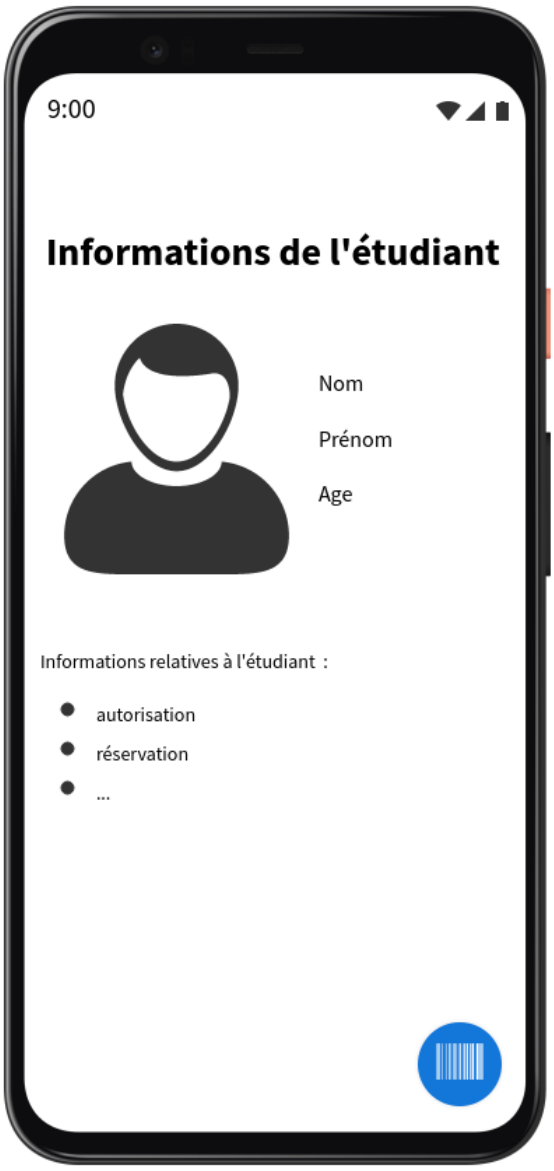
Page connexion

La page de connexion est la première page qui s’affiche quand on ouvre l’application. Vous devez rentrer votre identifiant et votre mot de passe pour passer au reste de l’application.



Page de scanne et de récupération des informations

Cette page s'affiche une fois que vous vous être connecté. Elle a deux fonctionnalités, la première qui permet de scanner le code-barre des cartes étudiantes et d'ainsi vérifier l'identité de la personne et d'avoir accès à un certain nombre d'informations. Au lancement de la page les informations de l'étudiant sons vierges et ne sont affichés qu'après le scanne du cade-barre. Il est également très facile de scanner plusieurs personnes à la suite avec ce système, car tout est fait sur la même page.



Serveur web

Comme je l'ai déjà indiqué précédemment, le serveur web sert d'API pour l'application mobile. Il permet de réaliser toutes les étapes de l'authentification CAS, ainsi que récupérer les informations des personne scannées. Pour sécuriser leur communication, le serveur et l'application utiliseront le protocole JWT.

Une application web sera également mis à disposition depuis le serveur pour ajouter des habilitations. Celles-ci seront directement stockées dans une base de données. Dans la première version de l'application web, seules les habilitations seront dans la base, mais d'autres éléments pourraient se rajouter et un diagramme UML serait intéressant pour représenter la base.