

# Réalisation d'une application web de scolarité.

Réalisé par :

- HENDEL Samy .
- AMROUCHE Wissem.
- MEBARKI Lamisse.

Groupe : 05

Encadré par :

**MR .MOSTEFEI**  
**MR BATATA**

## *Table des matières*

Introduction : .....	3
1. L'application .....	4
2. diagramme des CU après les modifications .....	6
3. l'architecture des SOA .....	7
4. Conclusion .....	8

## Introduction :

Dans ce TP on implémente une application web de scolarité ciblant les étudiants, les enseignants et l'administration.

Ce TP contient trois parties, modélisation, architecture et implémentation et qualité et documentation.

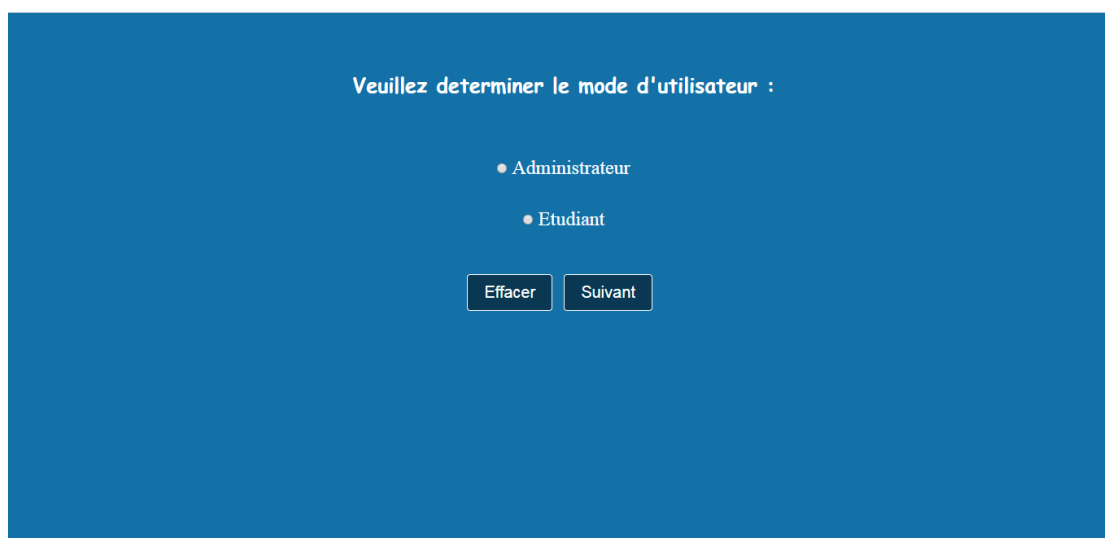
La partie modélisation contient les fonctionnalités de l'application, le diagramme de cas d'utilisation et le diagramme de classe de cette application ce qui a été déjà fait.

Dans ce rapport , on expose ce qu'on a fait en détail plus des captures de l'application web pour illustrer ce qu'on a fait.

Notre application a pour but de faciliter l'opération des permutations entre les groupes au début de l'année d'où la gestion des permutations automatiques .

## Application web : Gestion des permutation

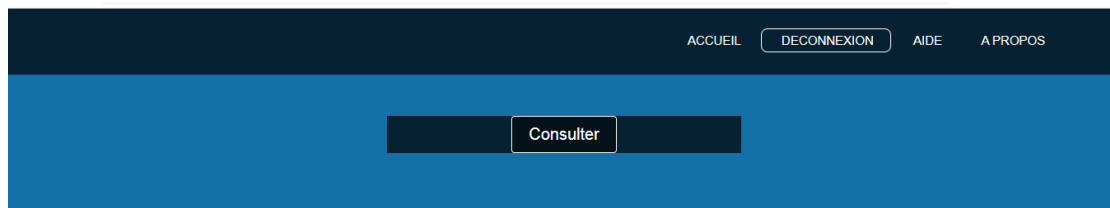
Notre application permet à l'utilisateur de choisir son profile ( étudiant /Admin ) comme illustre la photo ci-dessous :



Si l'utilisateur est un étudiant alors il aura 2 options :

- 1.consultation la liste des demandes.
2. ajouter une demande de permutation

Comme illustre les photos ci-dessous :



Veuillez introduire les informations suivantes :

Nom :  
Nom

Prenom :  
Prenom

Matricule :  
Matricule

Email :  
Email

Groupe actuel :  
Groupe actuel

Groupe voulu :  
Groupe voulu

Envoyer Effacer

Exemple d'une consultation :

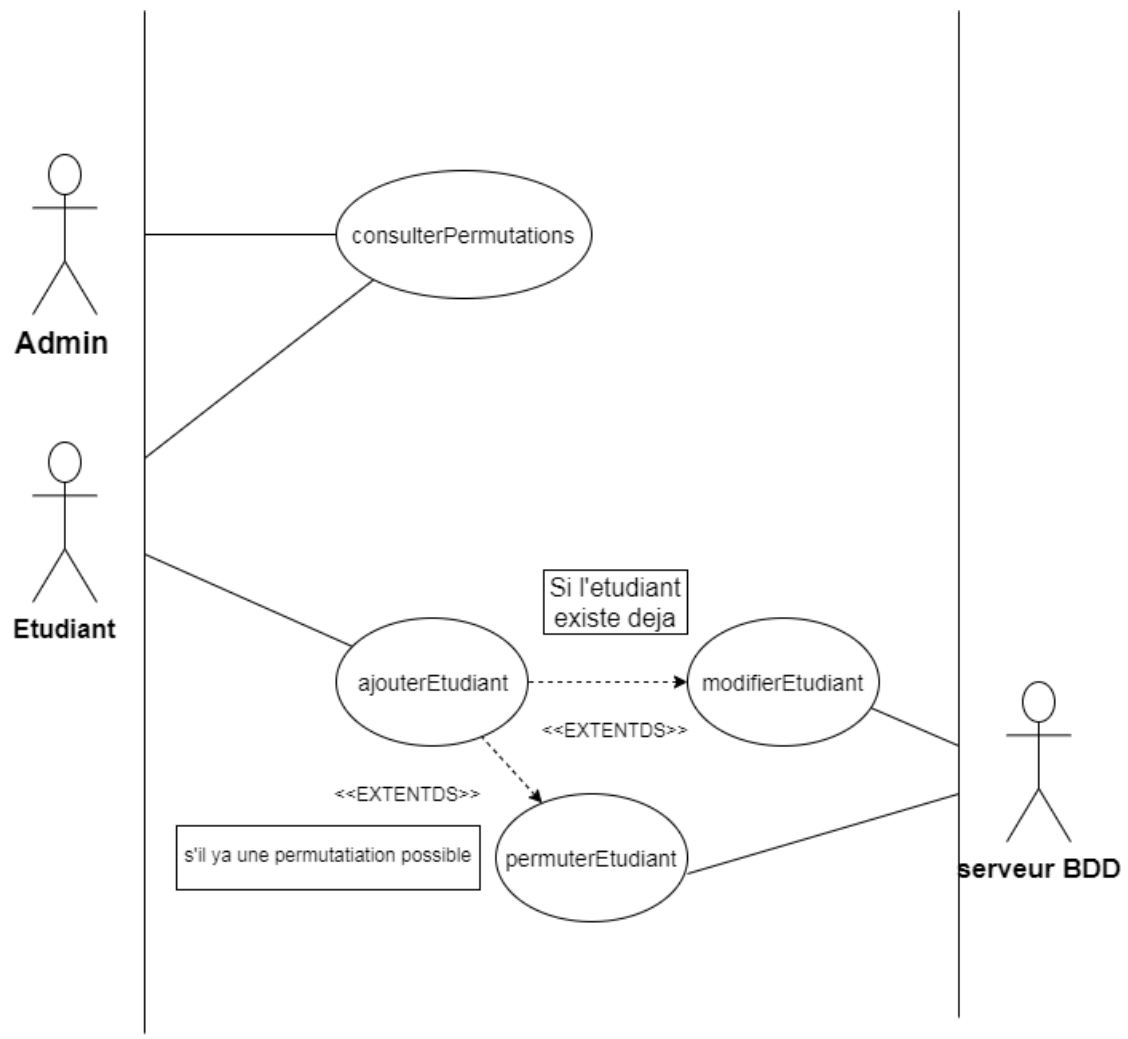
la liste des permutations

Nom	Prenom	Matricul	Email	Groupe actuel	Groupe voulu
amrouche	wissem	2017/0101	hw_amrouche@esi.dz	3	8

Si l'utilisateur est l'admin il aura ces options :



## Diagramme des cu après les modifications :



## *L'architecture de notre application :*

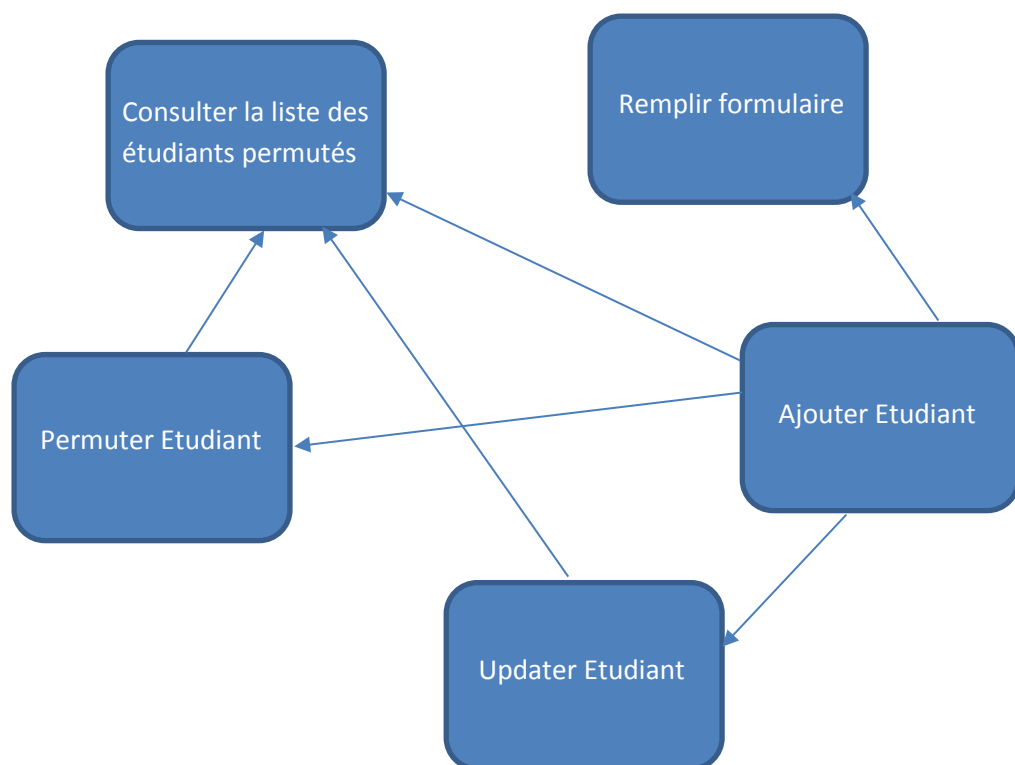
### 1. DEFINITION :

L'architecture SOA (Service Oriented Architecture) est une sorte d'architecture qui permet de modéliser les interactions et relations entre les différents services. Ces derniers, doivent être IRA :

**I : Interesting** => Les fonctionnalités de l'application doivent être intéressantes non seulement pour servir la finalité du composant que vous développez mais potentiellement intéressantes pour l'ensemble des web services du système.

**R : Reusable** => Le service doit être réutilisable pour d'autres contextes. Il ne doit pas être lié.

**A : Atomic** => Le services doit être indivisible et élémentaire. Il doit pouvoir faire une tache bien précise et correctement



## *Conclusion :*

À la fin de cette activité, nous avons appris à travailler en tant que groupe

Cela nous a également poussé à enrichir notre connaissance des architectures variantes existantes et décider laquelle on peut adapter au mieux l'application à l'ingénierie inverse. La mise en place d'un prototype était une expérience très bénéfique d'autant plus que on a été impliqués dans le développement Web, en utilisant node.js, vue.js comme programmation langages et MongoDB en tant que système de gestion de base de données relationnelle, pour la première fois à travers cette activité.