

# ***RAPPORT DE PROJET GESTION DES ETUDIANTS***

*Projet réalisé par*

*Majbri Oussema*

*Yahyaoui Khalil*

*Classe* : *2ème année Réseaux & Télécommunications*

*Année* : *2019/2020*

*Projet encadré par*

*Mr. Karoui Kamel*

*Institut National des Sciences Appliquées et de  
Technologie*

# Table des matières

---

I.	Dossier de spécification.....	3
1.	Objectif.....	3
2.	Scenario.....	4
II.	Dossier de conception.....	5
1.	Architecture .....	5
a.	Architecture Générale.....	5
b.	Fonctionnalités .....	5
2.	Le paquetage.....	6
III.	Dossier technique.....	7
1.	Technologies mises en œuvre.....	7
2.	Interface Serveur – Console.....	8
3.	Interface Client – Graphique.....	9
4.	Gestion des Sockets.....	11
5.	Gestion du Base de Données.....	12
a.	Relation entre JAVA et SQLite.....	12
b.	Création de la Base de Données.....	13
6.	Bibliothèques utilisées.....	14
IV.	Test d'exécution.....	15
V.	Conclusion.....	19

# *Dossier de spécification*

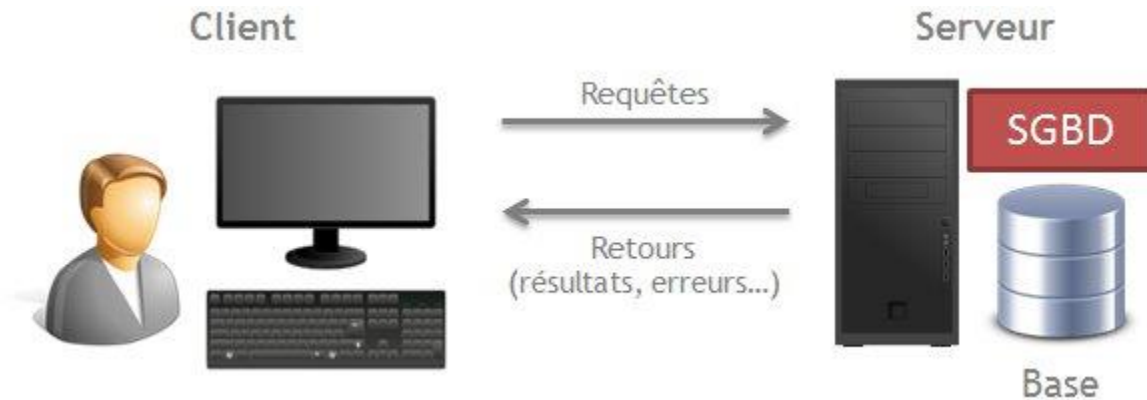
---

## *Objectif*

*Le but de ce projet est de se familiariser avec la Programmation Réseau en modélisant la résolution du problème de Gestion Des Etudiants. A cette fin, nous souhaitons développer une interface graphique qui facilite notre tâche.*

*Ce projet nous a permis de mieux maîtriser la gestion Sockets, la gestion des bases de données et la gestion des I/O.*

## Scenario



*L'application permet au client de se connecter au serveur en envoyant des requêtes contenant des informations. De son côté, le serveur interagit avec le SGBD (Système de Gestion de Base de Données) pour les traiter afin de retourner au client soit un résultat soit une erreur.*

## Dossier de Conception

---

# *I. Architecture*

- *Architecture Générale*

*Pour des raisons de sécurité, de fiabilité et de performance, l'application est répartie en deux sous programmes : une application console pour le serveur & une application graphique pour le client gérant les étudiants.*

- *Fonctionnalités*

*L'application qu'on souhaite réaliser, dispose plusieurs méthodes pour traiter les données stockées sur la base de données. On aura donc quatre fonctionnalités principales :*

***Ajouter un étudiant*** : *Le client insère les informations nécessaires afin d'ajouter un nouvel étudiant.*

***Afficher les informations d'un étudiant*** : *A travers le numéro d'inscription, le client est capable de consulter les informations d'un étudiant.*

***Supprimer un étudiant*** : *En indiquant le numéro d'inscription, le client est susceptible de supprimer un étudiant.*

***Modifier un étudiant*** : *Le client peut modifier les informations d'un étudiant en insérant des nouveaux changements.*

## II. Le paquetage

*Afin de mieux structurer l'ensemble des classes, on a rassemblé les classes dans des paquetages qui nous a permis d'optimiser l'opération d'importation :*

- *Package main : il renferme les classes contenant les méthodes mains.*
- *Package gui : il renferme les classes qui s'occupent de la gestion de Login, l'interface graphique de client et les Listeners.*
- *Package gui.ajouter : il renferme les méthodes qui gèrent l'opération d'ajout d'un étudiant.*
- *Package gui.supprimer : il renferme les méthodes qui gèrent l'opération de suppression d'un étudiant.*
- *Package gui.afficher : il renferme les méthodes qui gèrent l'opération d'affichage des informations relatives à un étudiant.*
- *Package gui.modifier : il renferme les méthodes qui gèrent l'opération de modification des informations d'un étudiant.*

# *Dossier Technique*

---

## *I. Technologies mises en œuvre*

- *Java* : On a utilisé le langage Java pour le développement de l'application et le design de l'interface graphique.
- *SQLite* : Pour la création et la gestion de la base des données, on a introduit le SGBD SQLite.

Pourquoi SQLite ? SQLite est beaucoup plus performant et facile à utiliser que d'autres SGBD, de plus SQLite est portable.

- *GitHub* : Vue les circonstances et les difficultés de coworking, on a utilisé le GitHub qui facilite le contrôle de versions et l'envoi des fichiers.

## II. Interface Serveur – Console

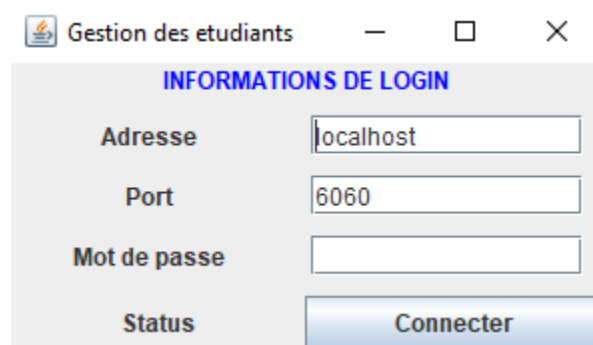
*L'interface du serveur montre les états de connexions récents au serveur, les changements dans la base de données et les erreurs s'ils existent.*

```
"C:\Program Files\Java\jdk1.8.0_181\bin\java.exe" ...  
Connecte a la base de donnees avec succes  
Bienvenue dans le serveur de l'INSAT  
Connectez-vous en utilisant ces parametres :  
adresse = localhost || port = 6060 || mot de passe = insat2020  
En attente du connexion ...  
[+] Connexion avec Succes , Port de client : 56156  
[+] oussema est ajouté avec succes  
[+] Modification avec Succes
```

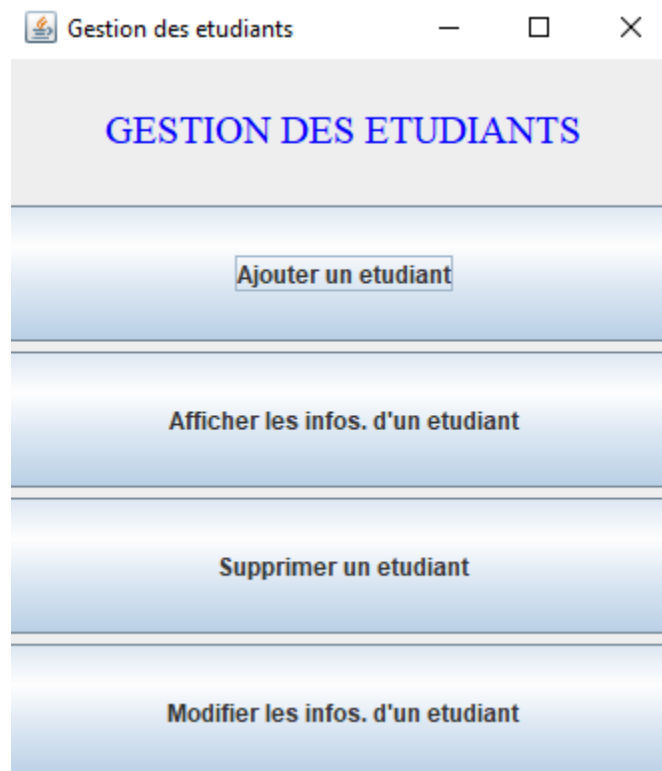


### III. L'interface Client – Graphique


*L'interface graphique permet au client de gérer facilement les étudiants. Elle est accompagnée par des buttons de retour au menu principal, de plus on a ajouté une interface de login pour spécifier l'adresse IP, le port et le mot de passe du serveur.*



The screenshot shows a window titled "Gestion des etudiants" with a standard Windows title bar (minimize, maximize, close buttons). The window content has a header "INFORMATIONS DE LOGIN" in blue. Below the header, there are four labels on the left and corresponding input fields on the right: "Adresse" with "localhost", "Port" with "6060", "Mot de passe" with an empty field, and "Status" with a "Connecter" button.



The screenshot shows a window titled "Gestion des etudiants" with a standard Windows title bar. The window content has a header "GESTION DES ETUDIANTS" in blue. Below the header, there are four large, light blue buttons stacked vertically: "Ajouter un etudiant", "Afficher les infos. d'un etudiant", "Supprimer un etudiant", and "Modifier les infos. d'un etudiant".

 Ajout d'un etudiant


Numero d'inscription

Nom

Prenom


E-mail

Choisir la classe   
Choisir la filiere


 Affichage des Infos. d'un etudiant

Numero d'Inscription

Nom	Prenom	Email	Classe	Filiere

 Suppression d'un etudiant

Numéro d'inscription

 Modification d'un etudiant

Numero d'inscription

Nom

Prenom

E-mail

Choisir la classe   
Choisir la filiere

## *IV. Gestion des Sockets*

*Le socket du Serveur :*

```
ServerSocket server = new ServerSocket(port);  
Socket connect = server.accept();
```

*Le socket du Client :*

```
Socket cnx = new Socket(log.getAdresse(), Integer.parseInt(log.getPort()));
```

*L'échange des informations :*

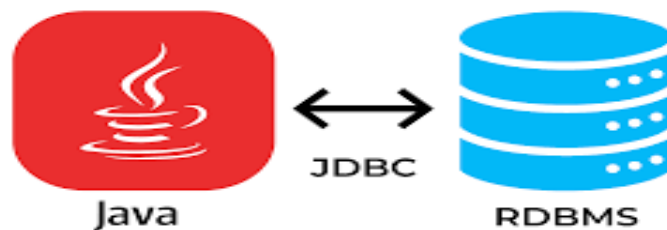
```
DataInputStream in = new DataInputStream(connect.getInputStream());  
DataOutputStream out = new DataOutputStream(connect.getOutputStream());
```

## V. Gestion de la Base de Données

### 1. Relation entre JAVA et SQLite

*Pour pouvoir accéder et gérer les bases de données, une application JAVA a besoin d'un connecteur JDBC.*

*Dans notre cas, on a utilisé la dernière version de SQLite Driver : **sqlite-jdbc-3.27.2.1.jar***



## 2. *Création de la Base de Données*

- *Création de la connexion*

```
Connection con = DriverManager.getConnection("jdbc:sqlite:src/db/GestionEtudiants.db")
```

- *Création des tables*

```
CREATE TABLE IF NOT EXISTS Filiere(  
    IdFiliere VARCHAR(5) PRIMARY KEY,  
    NomComplet VARCHAR(50) NOT NULL  
);  
  
CREATE TABLE IF NOT EXISTS Etudiant(  
    NumInscription VARCHAR(8) PRIMARY KEY,  
    Nom VARCHAR(20) NOT NULL,  
    Prenom VARCHAR(20) NOT NULL,  
    Email VARCHAR(30) NOT NULL,  
    Classe VARCHAR(6) NOT NULL,  
    FiliereEtud VARCHAR(5) NOT NULL REFERENCES Filiere(IdFiliere)  
);  
  
INSERT INTO Filiere(IdFiliere,NomComplet) VALUES("RT","Réseaux et Télécommunications");  
INSERT INTO Filiere(IdFiliere,NomComplet) VALUES("IIA","Informatique Industriel et Automatisme");  
INSERT INTO Filiere(IdFiliere,NomComplet) VALUES("IMI","Instrumentation et Maintenance Industrielle");  
INSERT INTO Filiere(IdFiliere,NomComplet) VALUES("GL","Génie Logiciel");  
INSERT INTO Filiere(IdFiliere,NomComplet) VALUES("CH","Chimie");  
INSERT INTO Filiere(IdFiliere,NomComplet) VALUES("BIO","Biologie");
```

- *Exemple d'envoi des requêtes*

```
String Suppression = "DELETE FROM Etudiant WHERE Numinscription = '"+NumInscri+"'";
```

```
String test1 = "SELECT * FROM Etudiant"+" WHERE NumInscription = '"+NumInscri+"'";
```

```
Statement stm = con.createStatement();
```

```
ResultSet rs1 = stm.executeQuery(test1);
```

```
if(rs1.next()){  
    stm.executeUpdate(Suppression);  
}
```

## VI. *Bibliothèques utilisées*

*On a utilisé les bibliothèques suivantes :*

- *Java.net : Gestion des sockets*
- *Java.sql : Gestion de la base de données*
- *Java.io : Gestion des exceptions et des I/O*
- *Java.swing : interfaces graphiques*
- *Java.awt : interfaces graphique et gestion des évènements*

# Test d'exécution

---

- *Lancement du Serveur (fichier .bat)*

```
"C:\Program Files\Java\jdk1.8.0_181\bin\java.exe" ...  
Connecte a la base de donnees avec succes  
Bienvenue dans le serveur de l'INSAT  
Connectez-vous en utilisant ces parametres :  
adresse = localhost || port = 6060 || mot de passe = insat2020  
En attente du connexion ...
```

- *Login du Client*

Gestion des etudiants

**INFORMATIONS DE LOGIN**

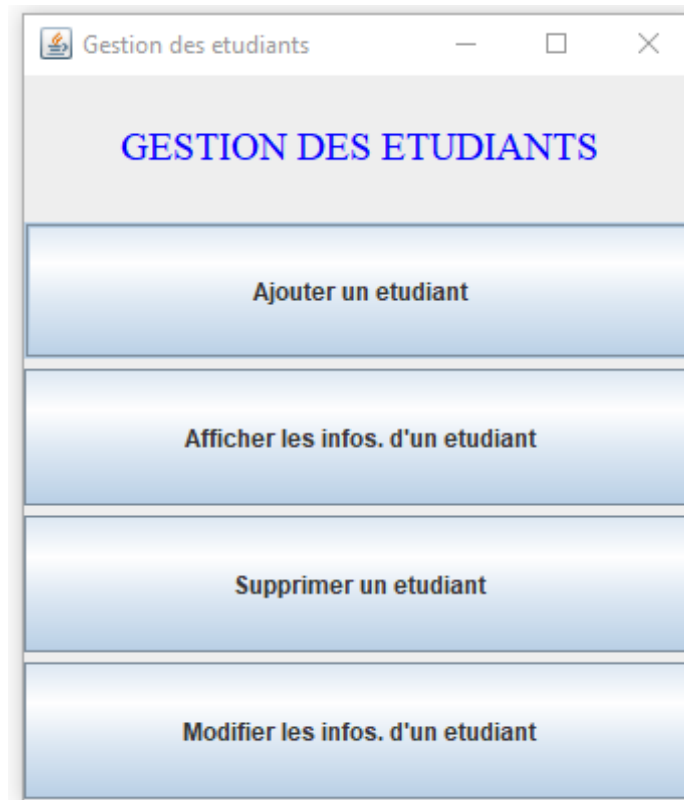
Adresse: localhost

Port: 6060

Mot de passe: .....

Status:

- *Choix d'opération*



Gestion des étudiants

**GESTION DES ETUDIANTS**

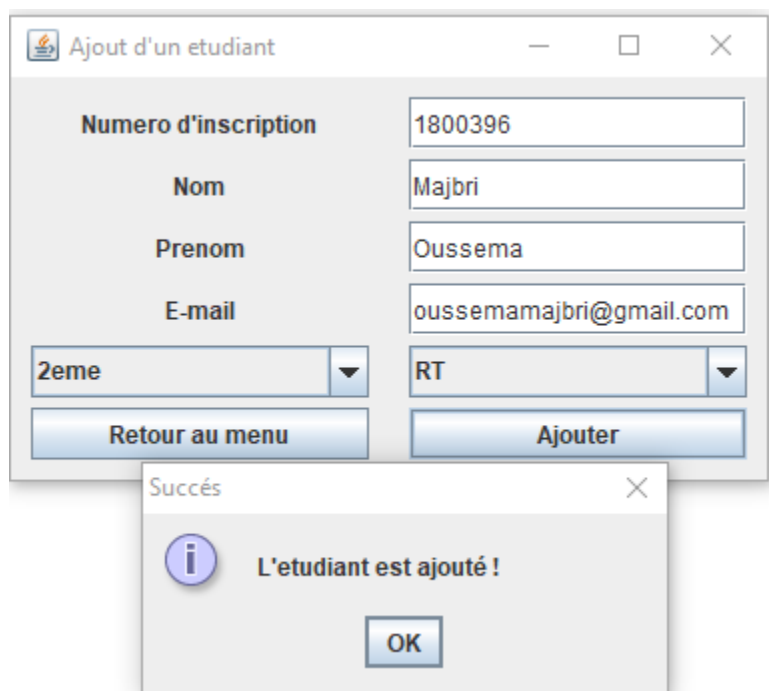
Ajouter un étudiant

Afficher les infos. d'un étudiant

Supprimer un étudiant

Modifier les infos. d'un étudiant

- *Ajout d'un étudiant*



Ajout d'un étudiant

Numero d'inscription: 1800396

Nom: Majbri

Prenom: Oussema

E-mail: oussemamajbri@gmail.com

Zeme: 2eme

RT: RT

Retour au menu

Ajouter

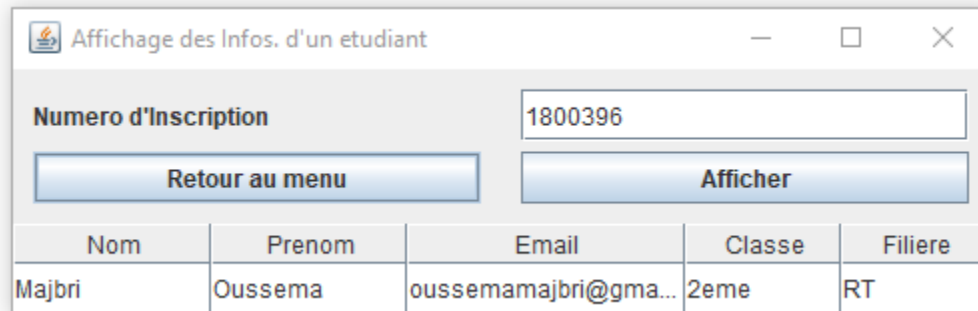
Succès

L'etudiant est ajouté !

OK

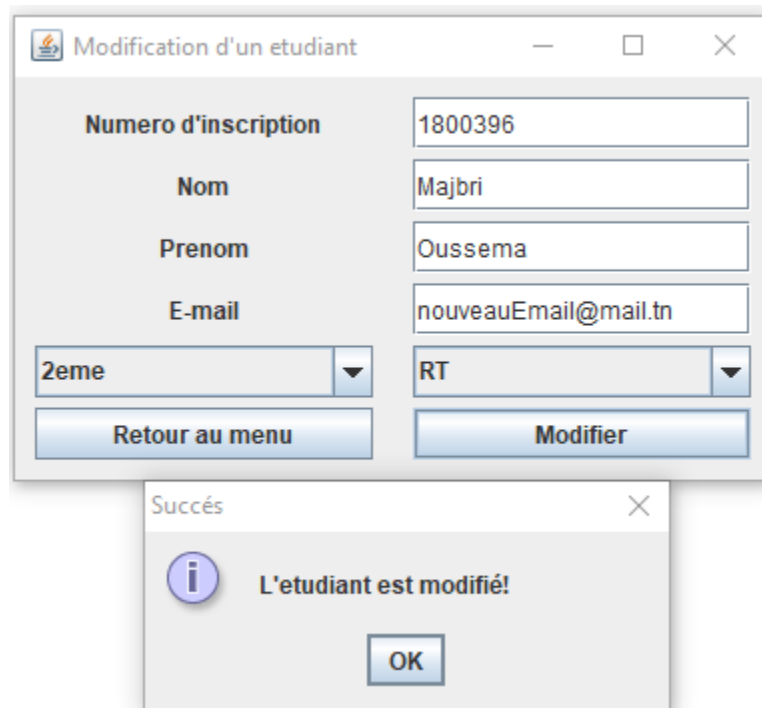


- *Affichage des informations d'un étudiant*




Nom	Prenom	Email	Classe	Filiere
Majbri	Oussema	oussemamajbri@gma...	2eme	RT

- *Modification des informations d'un étudiant*

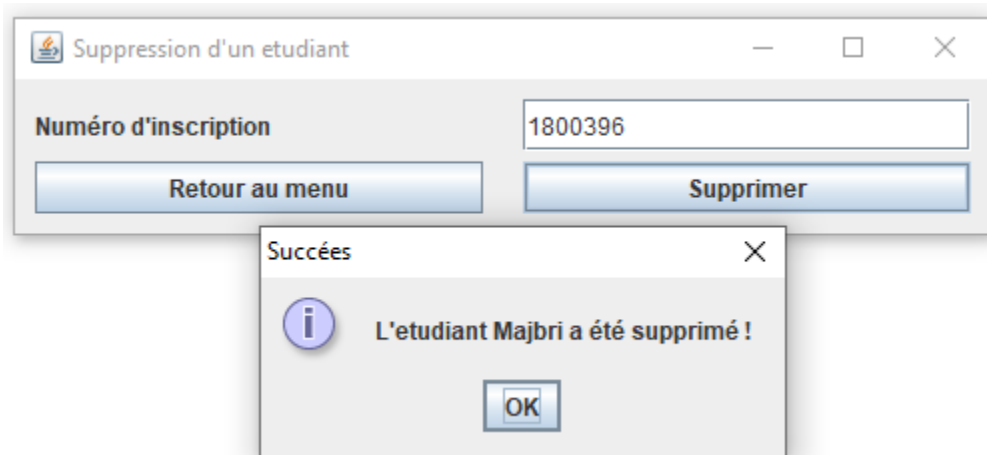


Succès

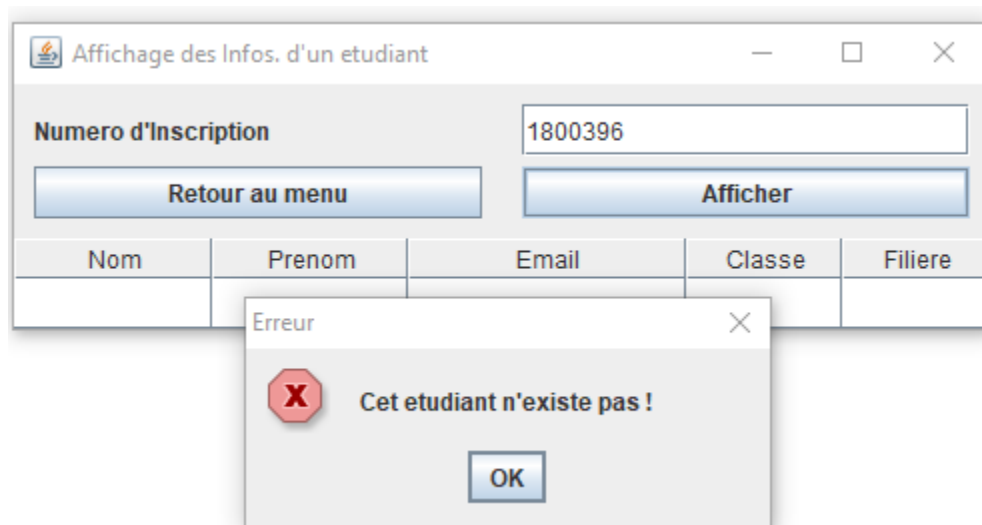
 L'etudiant est modifié!

OK

- *Suppression d'un étudiant*



- *Vérification de suppression*



# Conclusion

---

*Le projet d'application de gestion des étudiants avait pour but de créer une application qui permet de gérer l'ensemble des étudiants de l'INSAT.*

*Pour atteindre notre objectif, on a opté pour cette solution : base de données gérée par une application externe ce qui nous a permis d'appliquer et de développer nos connaissances de concept orienté objet ainsi que les langages SGBD, et plus précisément SQLite.*

*Ce projet nous a été bénéfique, d'un point de vue conception et design, puisque c'est notre première application avec interface graphique et notre premier rapport de projet, développant ainsi notre côté artistique et créatif.*

*On apprécie cette opportunité que vous nous avez donné pour consolider nos connaissances.*