

## \* MINI PROJET POO \*

# DÉVELOPPEMENT D'UNE APPLICATION QUI PERMET DE GÉRER LES ABSENCES DES ETUDIANTS

Réalisé par :  
**Bougacha Ghassen**  
**Mosbahi Wided**

**Année Scolaire:2022/2023**

# Plan

## 1/ Introduction Générale :

1.1- Quels sont les raisons pour choisir le Développement de Bureau ?

1.2-A Propos « Développement de Bureau »

- Définition

- Historique

- L'Importance de ce Développement

1.3- A propos le Projet

1.4-Conclusion

## 2/ La Présentation Du Projet :

2.1 - Introduction

2.2 – le Requis

2.3- Solution

- les Etapes Suivies

2.4-Conclusion

## 3/Conception:

### 3.1 - Introduction

### 3.2- Diagramme De Paquet

- Qu'est-ce qu'un diagramme de paquet ?
- Quel en est le but ?

### 3.3- Réalisation

- Paquet « Sign In Sign Up » & « Account »

### 3.4- Diagramme De Classe

## 4/Présentation Des Différents Outils Utilisés :

### 4.1 - Introduction

### 4.2- Outils de Recherche

### 4.3- Outils de prototypage

- figma

### 4.4- Outils de développement

- IntelliJ Idea
- MySQL

### 4.5-Conclusion

## 5/Phase De Design :

### 5.1- Introduction

### 5.2- Palette de Couleurs

- Qu'est-ce qu'une palette de couleurs ?
- Usage
- Avantages

### 5.3- UI design

- Connexion&Inscription
  - .Connexion
  - .Inscription
- Tableau de Bord (DashBoard)

### 5.4- Conclusion

## 6/Phase De Développement:

### 6.1- Introduction

### 6.2- Développement front-end

- La Table de « Account » & La forme des données
- Le Tableau de Bord (Dashboard)
- La Base De Donnée
- Préchargeur (preloader) :
- Interface de connexion(Sign IN)
- Interface d'Inscription(Sign UP)

## 6.6-Développement Back-End

- Développement en Java
- Travailler avec la base de données


## 6.7-Conclusion

# 1 / INTRODUCTION GÉNÉRALE :

## 1.1 - Quels sont les raisons pour choisir le Developpement de Bureau ?

**Voici quelques raisons pour choisir des applications de bureau dans votre développement :**

- Les applications de bureau ont une meilleure interaction avec le PC de l'utilisateur.
- Les performances des applications de bureau pour les calculs complexes sont beaucoup plus élevées que les performances des applications web.
- L'exécution d'une logique personnalisée côté client est possible mais beaucoup plus difficile avec une application web.
- L'utilisation de multithreading est plus facile et plus efficace dans une application de bureau.

- 
- La courbe d'apprentissage pour la conception d'interfaces utilisateur (IU) n'est pas en pente raide. Et pour WinForms, il est intuitif avec l'expérience glisser-déplacer du concepteur Windows Forms.
  - Il est facile de commencer à coder et tester vos algorithmes sans avoir à configurer une infrastructure de serveur ou à vous soucier des problèmes de connectivité, des pare-feux et de la compatibilité du navigateur.
  - Le débogage est puissant par rapport au débogage web.
  - L'accès aux appareils matériels, tels que l'appareil photo, le Bluetooth ou les lecteurs de cartes, est facile.
  - Comme la technologie existe depuis un certain temps, il existe de nombreux experts et une base de connaissances disponibles pour développer des applications de bureau.

## 1.2- A Propos le « Développement de Bureau »

### Définition :

Le Développement De Bureau ou on peut l'appeler aussi le développement d'application de Bureau est :



la création et l'implémentation des applications bureautiques qui sont des programmes qui s'exécutent sur un seul ordinateur, sans avoir besoin d'accéder à Internet ou à un réseau privé. Parmi les multiples points forts d'un logiciel CRM ; la rapidité, la fiabilité et la stabilité. Cela n'empêche qu'un point faible pas moins important est à connaître. Il s'agit de sa portabilité.



## Historique :

Avant l'essor d'Internet, les applications de bureau étaient l'approche principale de la création de systèmes logiciels. Les développeurs peuvent choisir n'importe quel langage de programmation, tel que COBOL, Fortran, VB6 ou C++. Mais qu'ils aient développé de petits outils ou des architectures distribuées complexes, elles étaient toutes des applications de bureau.

Ensuite, les technologies Internet ont commencé à choquer le monde du développement et à gagner plus d'ingénieurs avec des avantages tels que le déploiement facile et les processus de distribution simplifiés. Le fait qu'une fois l'application web déployée en production, tous les utilisateurs reçoivent des mises à jour automatiques a eu un impact considérable sur l'agilité du logiciel.

Toutefois, l'infrastructure Internet, les protocoles sous-jacents et les normes comme HTTP et HTML n'ont pas été conçus pour créer des applications complexes. En fait, l'effort de développement majeur vise alors seulement un objectif : donner aux applications web les mêmes fonctionnalités que celles dont disposent les applications de bureau, telles que la gestion rapide des entrées de données et de l'état.

Même si les applications web et mobiles ont augmenté à un rythme incroyable, certaines applications de bureau conservent la place de numéro un en termes d'efficacité et de performances. Cela explique pourquoi il existe des millions de développeurs qui créent leurs projets avec WPF et WinForms et la quantité de ces applications augmente constamment.

## L'Importance de ce Développement :

---

- Le développement pour le bureau est idéal pour de nombreuses raisons. La technologie est mature et testée à temps, le cycle de développement est rapide, le débogage est puissant et les applications de bureau ont moins de complexité et sont plus faciles à démarrer.

## 1.3-A PROPOS LE PROJET :

- Le projet est une application qui offre aux utilisateurs un moyen simple et rapide d'interagir avec l'administration, d'optimiser le système d'information existant de l'université, d'offrir aux gestionnaire des absences des étudiants un outil fiable et fonctionnel afin de faciliter leur travail , gagner plus de temps et d'ajouter aux anciennes fonctionnalités de nouveaux utilitaires permettant d'automatiser plus d'activités de travail de ces gestionnaires tout en offrant une ergonomie et une bonne expérience utilisateur.

## 1.4-Conclusion :

- Sur la base de ce que nous avons vu, nous pouvons conclure que le développement en général devient un point important de notre vie, il le facilite, il facilite notre relation

## 2/ LA PRÉSENTATION DU PROJET :

### 2.1-Introduction :

- Dans ce chapitre, nous décrirons le projet et quelle en est la cause.

### 2.3-Solution :

Développement d'une application afin de faciliter le travail du gestionnaire des absences dans une université

### 2.2-Le Requis:

- Nous avons aller faire dans ce projet ,une application permettant de :
- -identifier les différents utilisateurs de l'application
- -remplir les listes de présence par chaque classe à chaque cours
- -consulter les absences enregistrés pour une classe ou un étudiant donné
- -annuler une absence enregistré en cas de justification par l'étudiant
- -envoyer des mails d'alertes aux étudiants absentéistes
- -générer de graphe , statique du taux d'absentéisme par classe ,par filière ou par niveau d'étude

# Les Etapes Suivies :

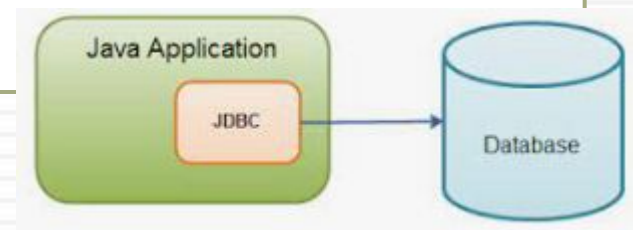
- -La Lecture du L'énoncé avec attention
- -la Bonne Compréhension du problème
- -l'Analyse et L'étude du sujet
- -la Conception et la modélisation
- -La Recherche des différents solution
- -la préparation du notre Base de donnée
- -l'Ecriture du code en Java
- -La connexion entre MySQL et IntelliJ Idea avec le « jdbc »
- -la vérification du code écrit avec la compilation et l'exécution
- -Création des tables et des requêtes SQL avec java
- -Faire le Test avec le Preloader
- -Connexion&Inscription (Tant que Utilisateur )



## Remarque :

### Définition sur le « JDBC » :

La connectivité de base de données Java™ (JDBC) est la spécification Java Soft d'une interface de programmation d'application (API) standard qui permet aux programmes Java d'accéder aux systèmes de gestion de base de données. L'API JDBC consiste en un ensemble d'interfaces et de classes écrites dans le langage de programmation Java.





## 2.4-Conclusion :

à la fin de cette partie , on a conclue qu'on doit réfléchir avec une bonne conception et bonne modélisation du problème pour permettre d'écrire un meilleur code

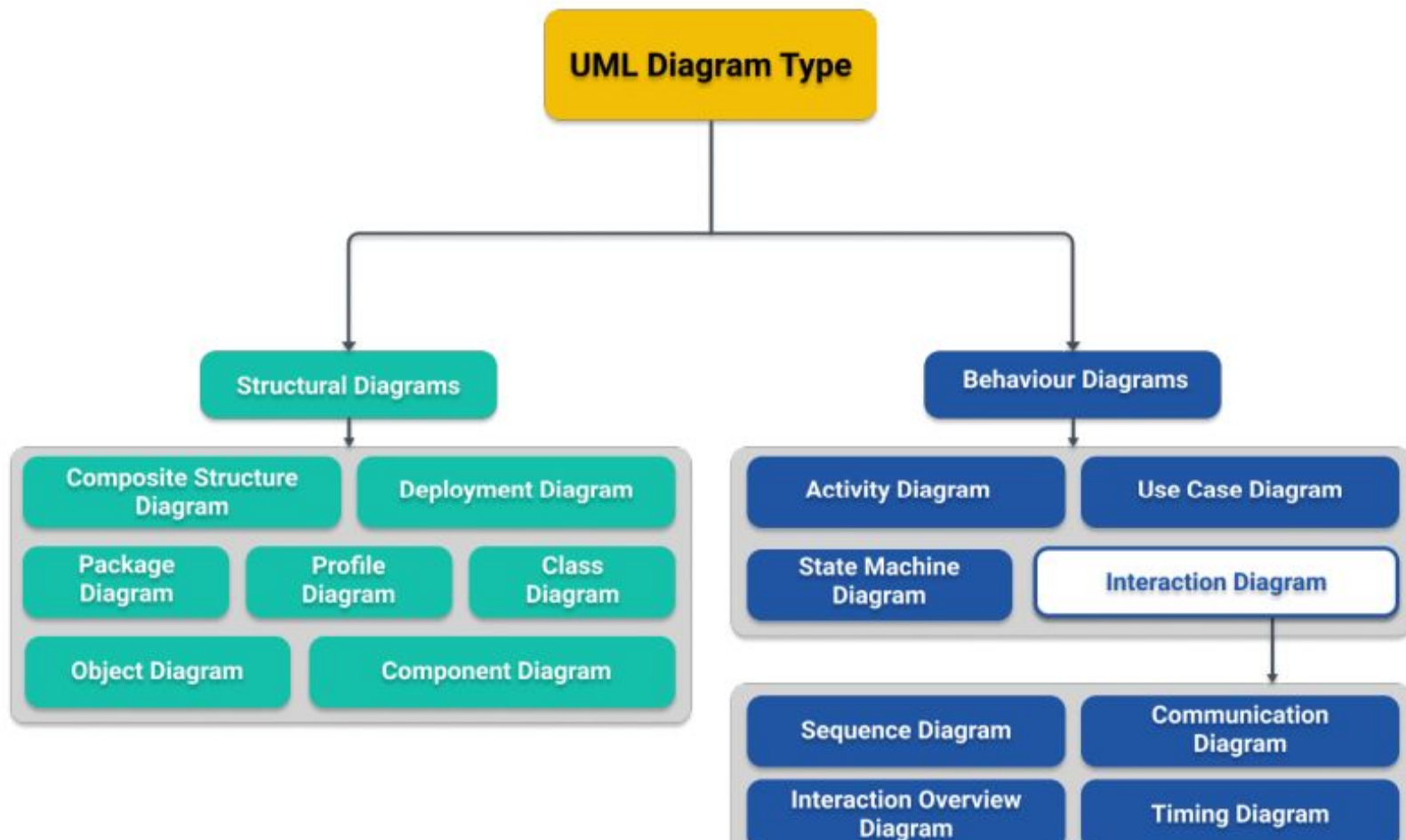
# 3/CONCEPTION :

## 3.1-Introduction :

Dans cette partie, nous allons présenter le diagramme de paquet ( package diagram)et le diagramme de classe de l'application et nous plongerons profondément dans certaines classes qui gèrent la génération principale et le système dynamique de notre application.

## 3.2-Diagramme de paquet :

-Qu'est-ce qu'un diagramme de paquet ?



Les standards UML actuels font appel à 15 types de diagrammes différents : classe, activité, objet cas d'utilisation, séquence, paquet , état, composant, communication, structure composite aperçu de l'interaction, calendrier et déploiement . Ces diagrammes sont organisés en deux groupes distincts : les diagrammes comportementaux et structurels ou diagrammes d'interaction :

- Diagramme de Paquet
- Diagramme de profil
- Diagramme de classe
- Diagramme de structure composite
- Diagramme de déploiement
- Diagramme d'objets
- Diagramme des composants

\* Les Diagrammes comportementaux

- Diagramme d'activité
- Diagramme de machine d'état
- Diagramme de cas d'utilisation
- Diagramme de séquence
- Schéma de communication
- Diagramme de présentation des interactions
- Chronogramme

### Qu'est-ce qu'un diagramme de paquet ?

- Un diagramme de paquet est une sorte de diagramme structurel qui montre l'arrangement et l'organisation des éléments du modèle dans le cadre d'un projet de moyenne à grande échelle. Le diagramme de paquet peut montrer à la fois la structure et les dépendances entre les sous-systèmes ou les modules en montrant différentes vues d'un système. Par exemple, en tant qu'application multicouche (alias multi niveau) -modèle d'application multicouche.

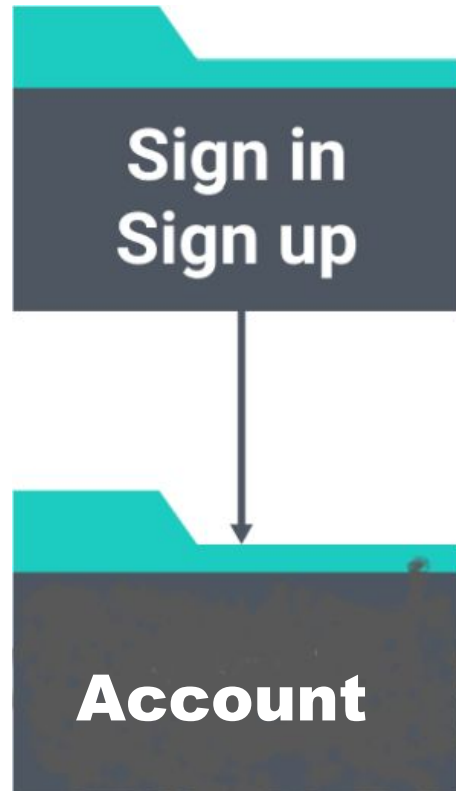
### Quel en est le but ?

- Le diagramme de paquet regroupera tous les diagrammes de classes et changera leur domaine, c'est pour quoi nous avons choisi de l'utiliser, et nous savons que les diagrammes de classes sont le principal élément de base de modélisation orientée objet. Ils sont utilisés pour montrer les différents objets d'un système, leurs attributs, leurs opérations et les relations entre eux.

### 3.3-Réalisation :

#### Diagramme de Paquet

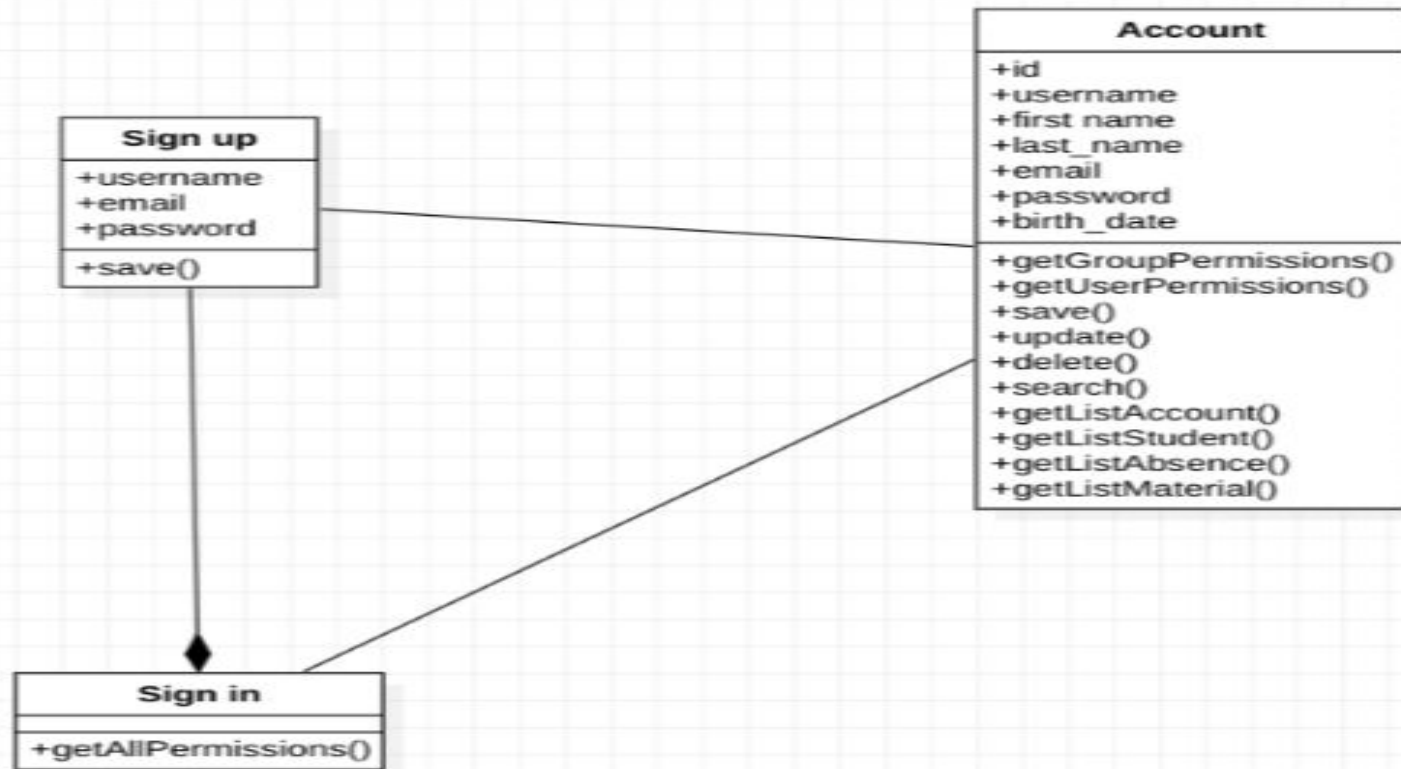
- Dans cette section, nous allons montrer notre diagramme de paquet



**Nous pouvons diffuser notre travail en Deux paquets :**

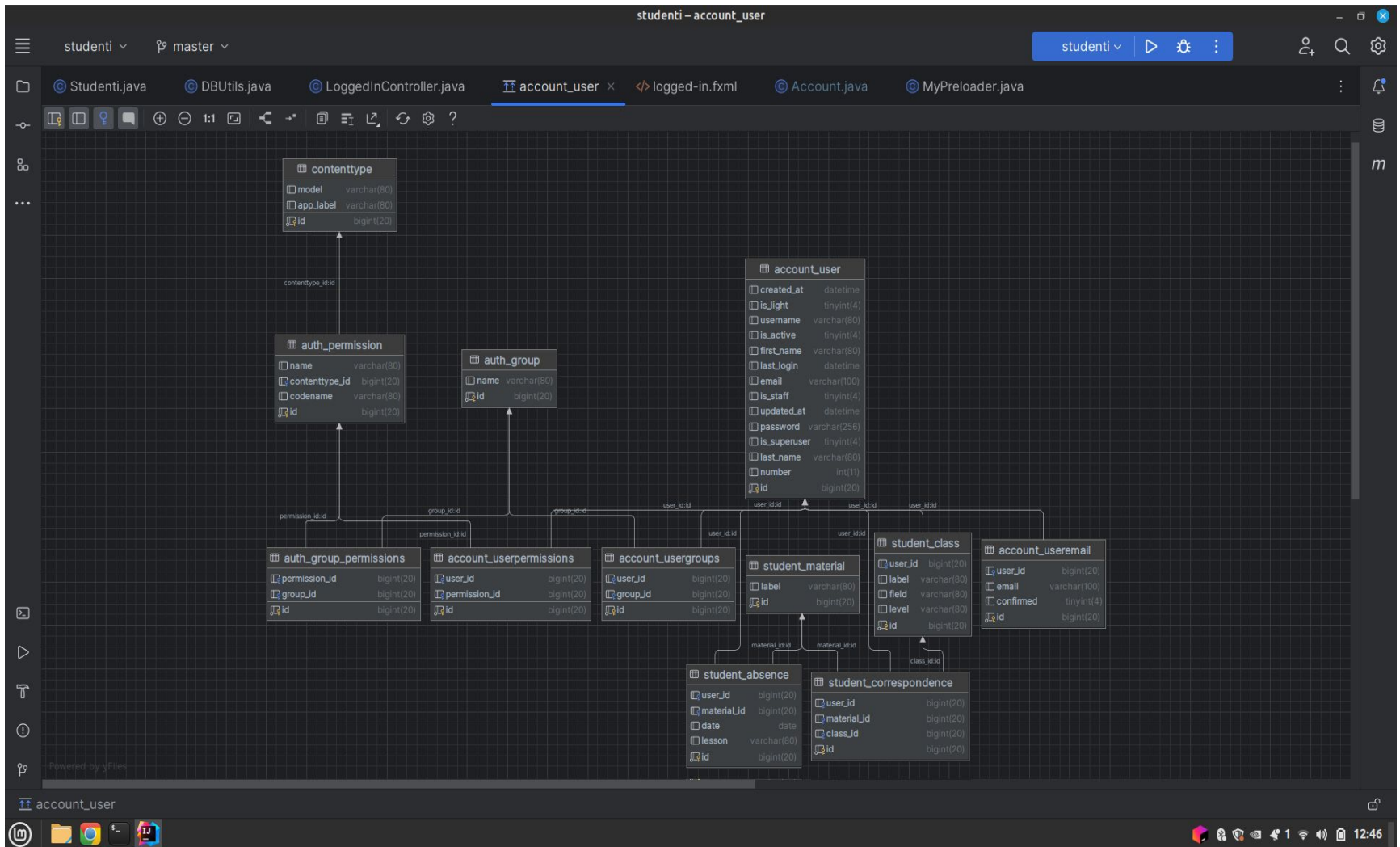
- Le paquet « Sign In Sign Up »**
- Le paquet « Account »**

## Paquet « Sign In Sign Up » & « Account »



Il s'agit du diagramme de Classe qui contient trois classes qui indiquent clairement définit que chaque utilisateur devait s'inscrire et se connecter pour avoir un compte et pour pouvoir entrer et manipuler son Les données.

## 3.4-Diagramme de Classe





# 4/PRÉSENTATION DES DIFFÉRENTS OUTILS UTILISÉS :

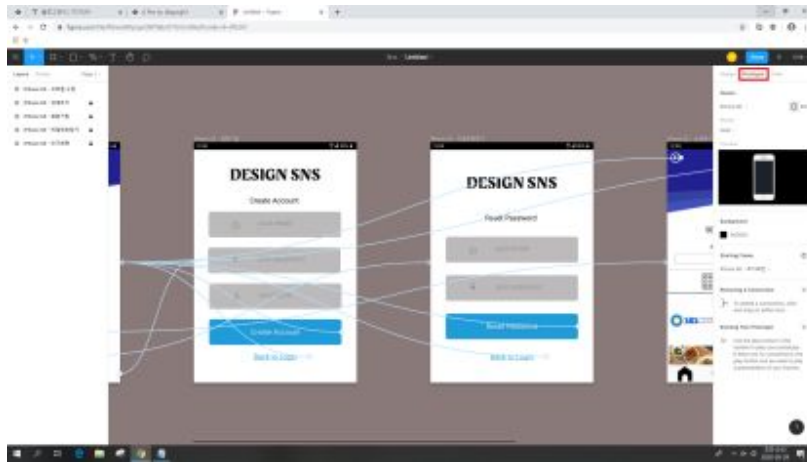
## 4.1 -Introduction :

- Faire un plan et le suivre étape par étape pour fournir une application qui satisfait tous les besoins des utilisateurs n'est pas quelque chose de facile du tout. Nous avons donc besoin d'outils qui peuvent nous aider rendre notre travail plus facile et plus rapide.

## 4.2- Les Outils De Recherche :

- Dans ce module, nous avons dû chercher plus loin pour trouver des solutions aux problèmes auxquels les gestionnaires d'absences sont confrontés tout le temps , l'administration de l'université ne peut pas trouver une solution pour. Donc, pour ce faire, certaines étapes ont été nécessaires pour rendre cela possible.

## 4.3- Les Outils de prototypage

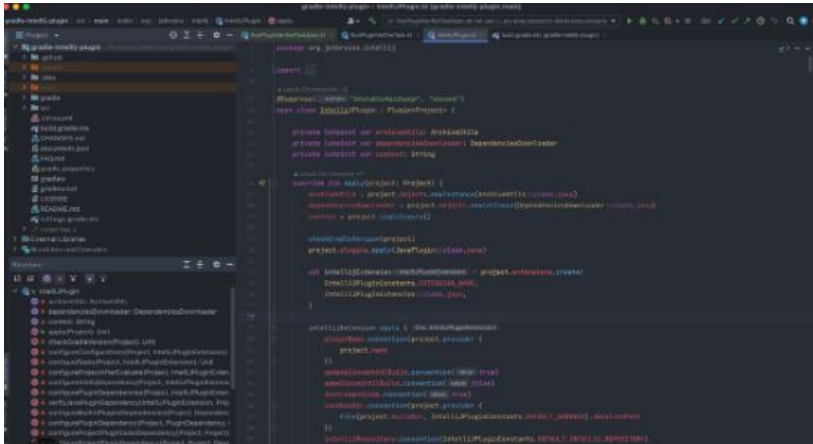


Figma interface

- Figma est l'une des applications d'édition graphique les plus révolutionnaires qui prend en charge la conception monde par la tempête. Ce qui le rend si attrayant, c'est le fait qu'il est gratuit. Il s'agit d'une application Web d'édition graphique et de conception d'interface utilisateur . Vous pouvez l'utiliser pour effectuer toutes sortes de travaux de conception graphique à partir de sites Web de cadrage filaire, en concevant interfaces d'applications mobiles, conceptions de prototypes, création de publications sur les réseaux sociaux et tout le reste.

## 4.4-Les Outils de Développement :

*IntelliJ Idea :*



IntelliJ Idea interface

- ❑ **IntelliJ IDEA** également appelé « IntelliJ », « IDEA » ou « IDJ » est un environnement de développement intégré (en anglais Integrated Development Environment - IDE) destiné au développement de logiciels informatiques reposant sur la technologie Java. Il est développé par JetBrains (anciennement « IntelliJ ») et disponible en deux versions, l'une communautaire, open source, sous licence Apache 2 et l'autre propriétaire, protégée par une licence commerciale. Tous deux supportent les langages de programmation Java, Kotlin, Groovy et Scala.

## MySQL :



- MySQL est un système de gestion de base de données relationnelle (RDBMS) open source. Son nom est une combinaison de "My", le nom de la fille du co-fondateur Michael Widenius, et "SQL", l'abréviation de Structured Query Language. Une base de données relationnelle organise les données dans une ou plusieurs tables de données dans lesquelles les types de données peuvent être liés les uns aux autres ; ces relations aident à structurer les données. SQL est un langage utilisé par les programmeurs pour créer, modifier et extraire des données de la base de données relationnelle et contrôler l'accès des utilisateurs à la base de données . En Outre les bases de données relationnelles et SQL, un SGBDR comme MySQL fonctionne avec un système d'exploitation pour implémenter une base de données relationnelle dans le système de stockage d'un ordinateur, gère utilisateurs, permet l'accès au réseau et facilite le test de l'intégrité de la base de données et la création de sauvegardes.

## 4.5-Conclusion :

ON PEUT DONC DIRE QUE LES OUTILS SONT NÉCESSAIRES ET IMPORTANTS DANS TOUT PROJET TEL QUE LE DÉVELOPPEMENT OU TOUT AUTRE DOMAINE. NOUS DEVONS BIEN CHOISIR NOS OUTILS POUR FACILITER LE TRAVAIL PAS PLUS DUR. ET EN UTILISANT MES ÉLUS J'AI PASSÉ DEUX PHASES IMPORTANTES QUI SONT LA CONCEPTION ET LE DÉVELOPPEMENT QUE NOUS VERRONS DANS LES PROCHAINS CHAPITRES.

# 5/PHASE DE DESIGN

## 5.1-Introduction :

- Dans cette partie, nous parlerons de la conception de l'interface utilisateur de notre application, du cadrage filaire et de la palette de couleurs que nous choisissons pour notre interface utilisateur. Comment que la conception est une partie difficile de notre travail. Créer une belle interface utilisateur avec une excellente UX est une chose très importante pour le site Web.

## 5.2- Palette de Couleurs

- Qu'est-ce qu'une palette de couleurs ?
  - Une palette de couleurs est une gamme de couleurs qui sont mélangées les unes aux autres pour offrir une vue parfaite à l'utilisateur et le rendre satisfait de ce qu'il voit.



## Choix de couleur :

- Comment choisir les couleurs dans l'interface utilisateur de notre site Web ? C'est la question qui a été dans l'esprit de nombreux développeurs de tous les temps. Choisir les couleurs n'est pas aussi simple qu'il y paraît car c'est la plus chose importante dans la construction de sites Web, c'est la façon dont l'utilisateur est satisfait en voyant des couleurs attrayantes qui le rendent heureux tout en naviguant.

### - Usage :

- Une palette de couleurs est utilisée dans de nombreux domaines différents comme :
  - Conception d'interface utilisateur et d'expérience utilisateur «UI and UX design »
  - Cinématographie
  - La photographie
  - Correction de couleur
  - LUT

### - Avantages :

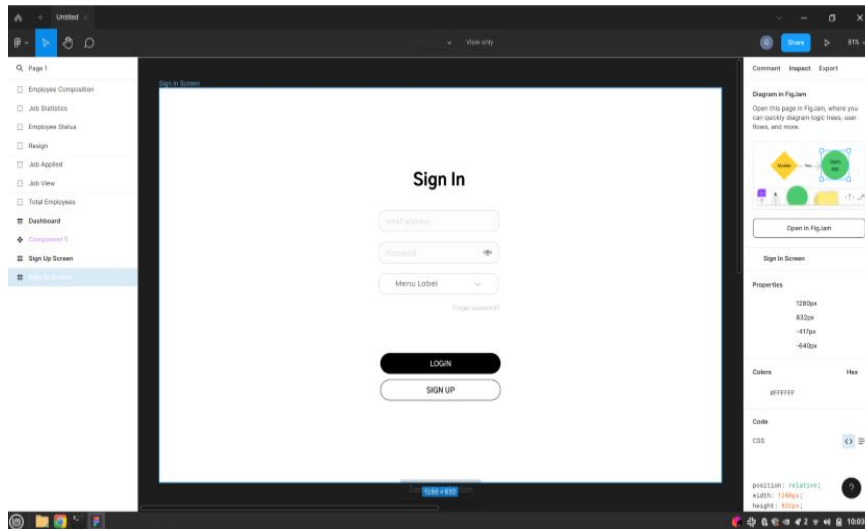
Les avantages de la palette de couleurs sont nombreux tels que :

- Simplification du mélange des couleurs.
- Mieux comprendre comment les couleurs fonctionnent ensemble.
- Atteindre l'harmonie des couleurs.
- Contrôle facile des valeurs et de la température de couleur.
- Réduction du poids des fournitures dans le pack de peinture portable.
- etc.....

## 5.3 UI design

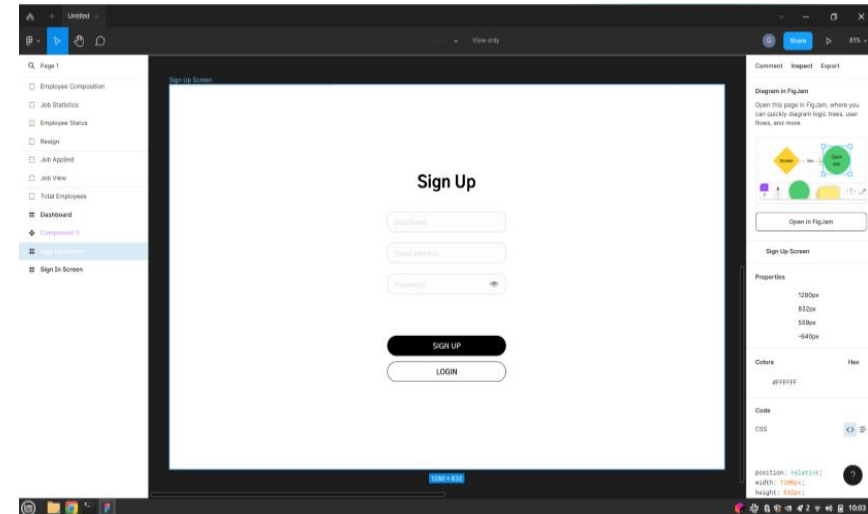
Connexion & Inscription :

Connexion :



Login UI  
(se connecter )

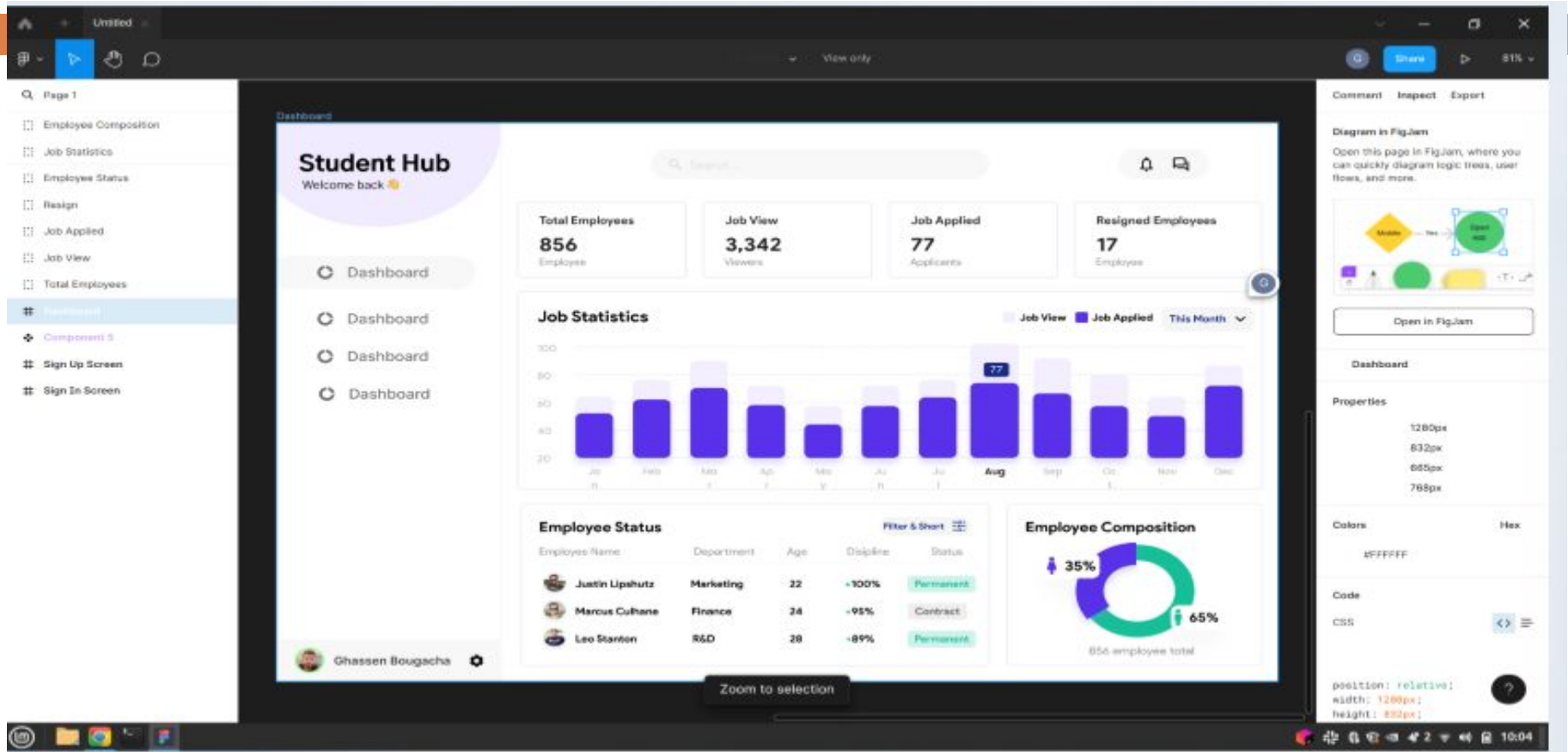
Inscription :



Register UI  
(s'inscrire)



# Tableau De Bord :



Les tableaux de bord sont un moyen unique et puissant de présenter des renseignements basés sur des données à l'aide de techniques de visualisation de données qui affichent des données pertinentes et exploitables ainsi que des statistiques de suivi et des indicateurs de performance clés (KPI).

## 5.4-Conclusion :

La fin de cette partie conclut la fin de la phase de Design et le début de la phase de développement . Donc dans la partie suivante qui est la dernière nous présenterons toutes les différentes étapes de développement que nous avons effectuées pour développer notre application

# 6/PHASE DE DÉVELOPPEMENT:

## 6.1-Introduction :

- Après la phase de design, nous arrivons à une étape très importante qui est la réalisation de notre application. Dans cette phase, nous devons décrire le travail de développement à l'aide d'outils qui a déjà été défini dans le chapitre de présentation des différents outils utilisés.

## 6.2-Développement front-end:

- Tout d'abord, nous devons concrétiser notre conception, en une chose fonctionnelle en la développant, nous avons utilisé le XML .
- (Qu'est-ce que XML ?
- **Extensible Markup Language** (XML) vous permet de définir et de stocker des données de manière à pouvoir les partager. XML prend en charge l'échange d'informations entre des systèmes informatiques tels que les sites web, les bases de données et les applications tierces.)

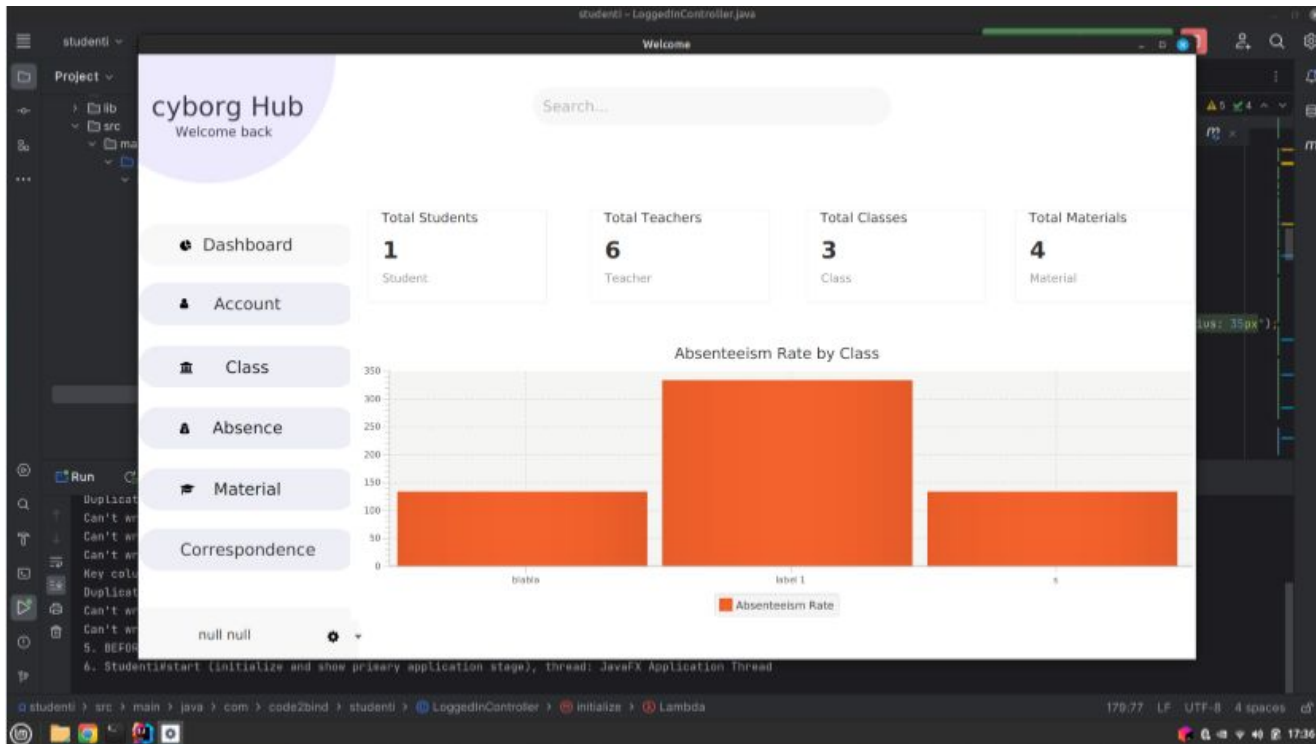
## Table de « Account » & la forme des données

The screenshot shows the 'cyborg Hub' interface. On the left is a sidebar with a 'Welcome back' message and a list of menu items: Dashboard, Account (highlighted), Class, Absence, and Material. At the bottom of the sidebar is the user's name 'Ghassen Bougacha'. The main area contains a search bar at the top. Below it is a table with columns: #, First name, Last name, Email, Phone, and Role. The table is currently empty, displaying the message 'No content in table'. To the right of the table is a form titled 'Account' with input fields for Username, Email, First name, Last name, and Phone.

Cette partie (la forme des données) est l'endroit où l'utilisateur peut intégrer les données, toutes, et en cliquant sur "Confirmer" nous avons eu accès à la demande de publication qui publie toutes les données dans la base de données afin que nous puissions les obtenir dans le tableau.

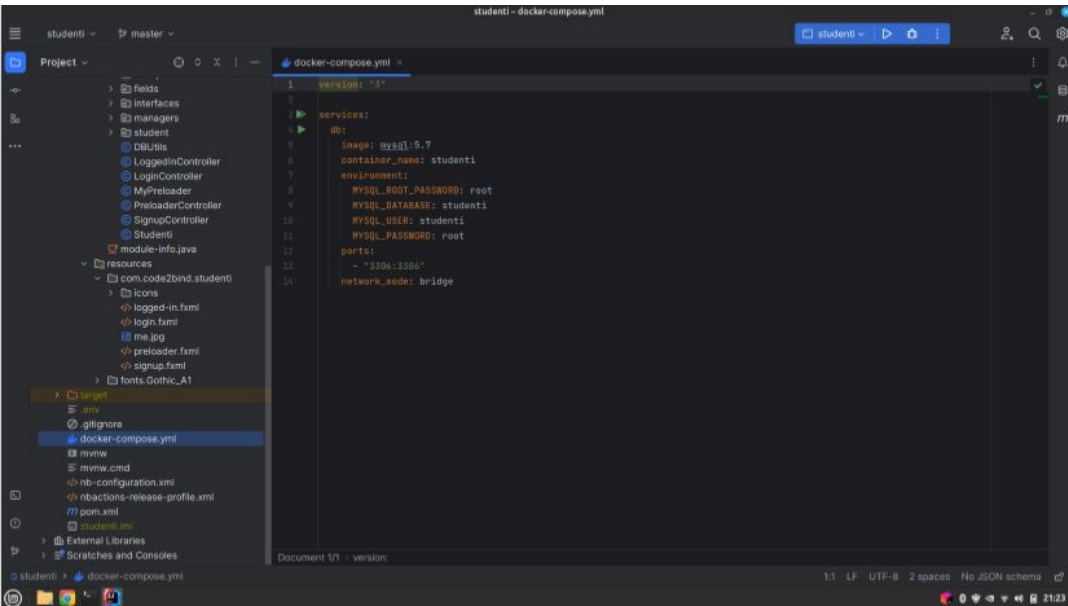
- Cette table est l'endroit où nous obtenons nos données de la base de données et il existe d'autres fonctionnalités que nous pouvons faire dans cette interface. La barre de recherche ci-dessus est l'endroit où nous pouvons rechercher notre cible, nous écrivons simplement les données que nous voulons rechercher, puis nous cliquons sur le bouton « Search ». avant même de pouvoir faire cela, nous devons ajouter nos comptes avant

## Tableau de Bord :



Ce tableau présente la génération de graphe , statistique du taux d'absentéisme par classe

# La Base de Données :



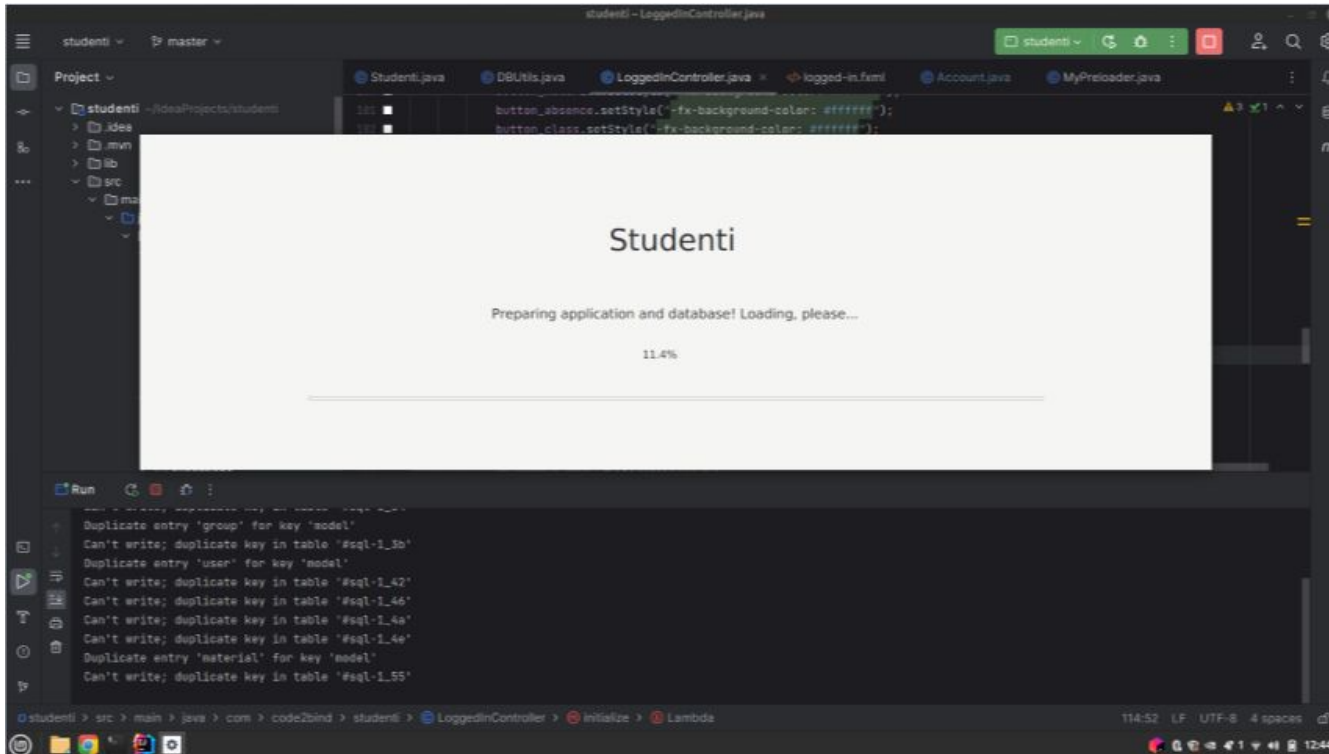
## « Cloud Computing » :

Le Cloud Computing (en français, « informatique dans les nuages ») fait référence à **l'utilisation de la mémoire et des capacités de calcul des ordinateurs et des serveurs répartis dans le monde entier et liés par un réseau.**

Création d'un Conteneur docker contenant la base de données MySQL

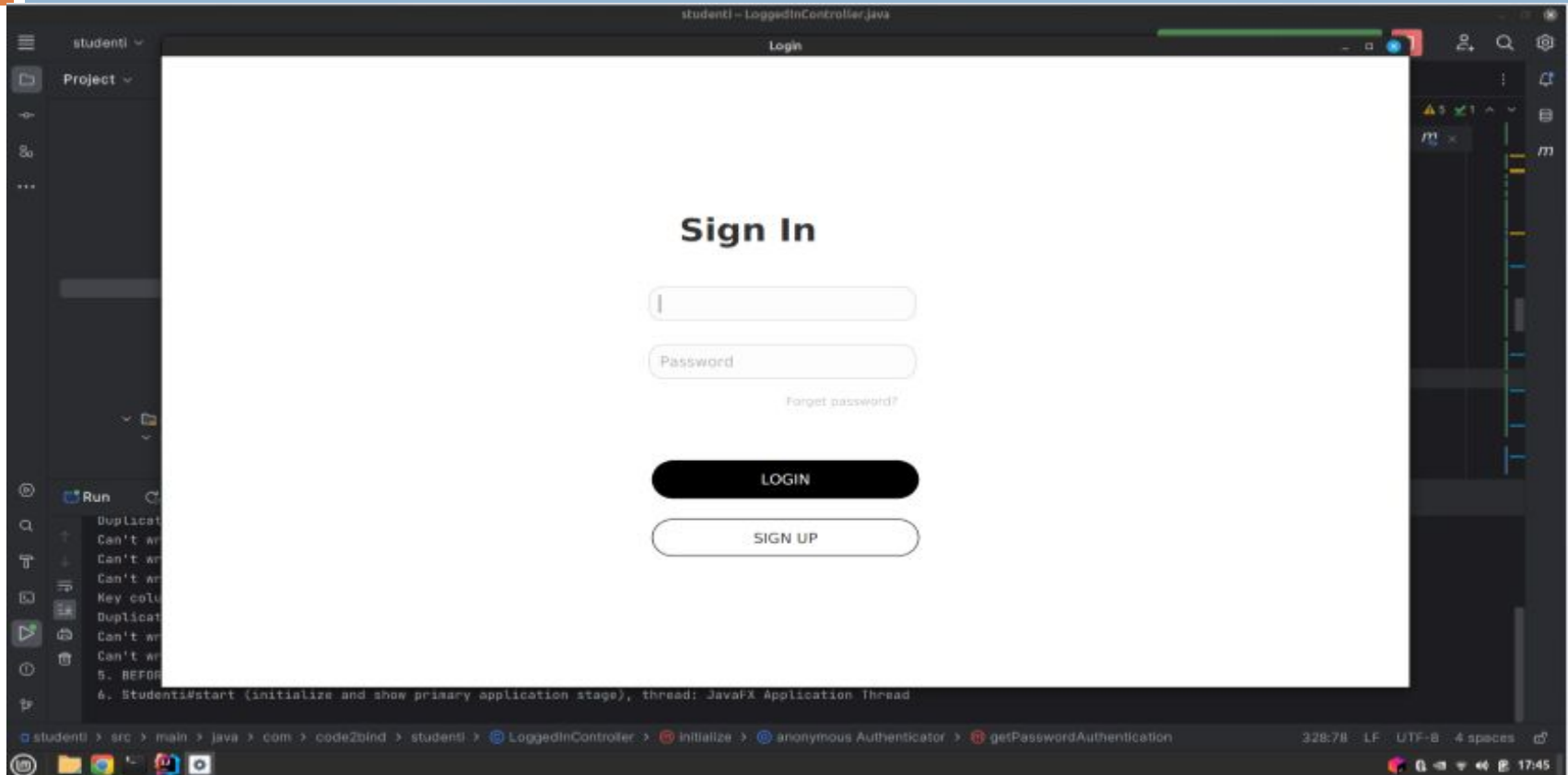
Création d'un conteneur sur docker avec la commande docker-compose qu'il exécute ce fichier et il crée notre base de données

# Préchargeur (preloader) :



C'est une étape importante pour vérifier les données et d'éviter les erreurs

## L'interface de Connexion :

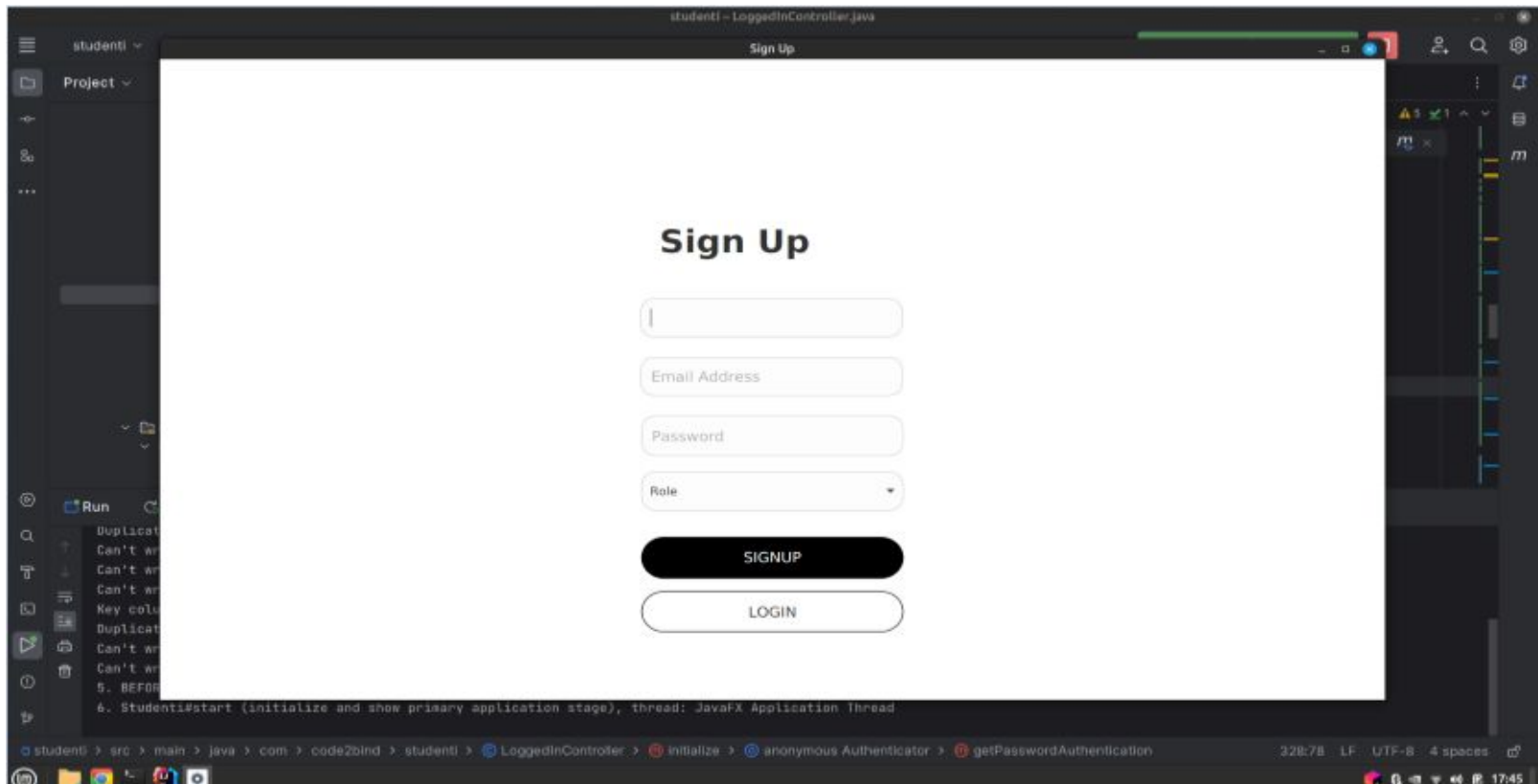


La connexion est l'endroit où l'utilisateur devait entrer ses coordonnées pour avoir accès à entrer et manipuler les données, mais avant cela, il devait s'inscrire pour pouvoir se connecter



# L'Interface d'Inscription

:



Pour s'inscrire, l'utilisateur (l'enseignant ,l'étudiant , le(s) responsable(s) administratif(s)) devait saisir ses données, puis appuyer sur le bouton « SIGNUP" pour pouvoir se connecter.

## 6.3-Développement Back-end:

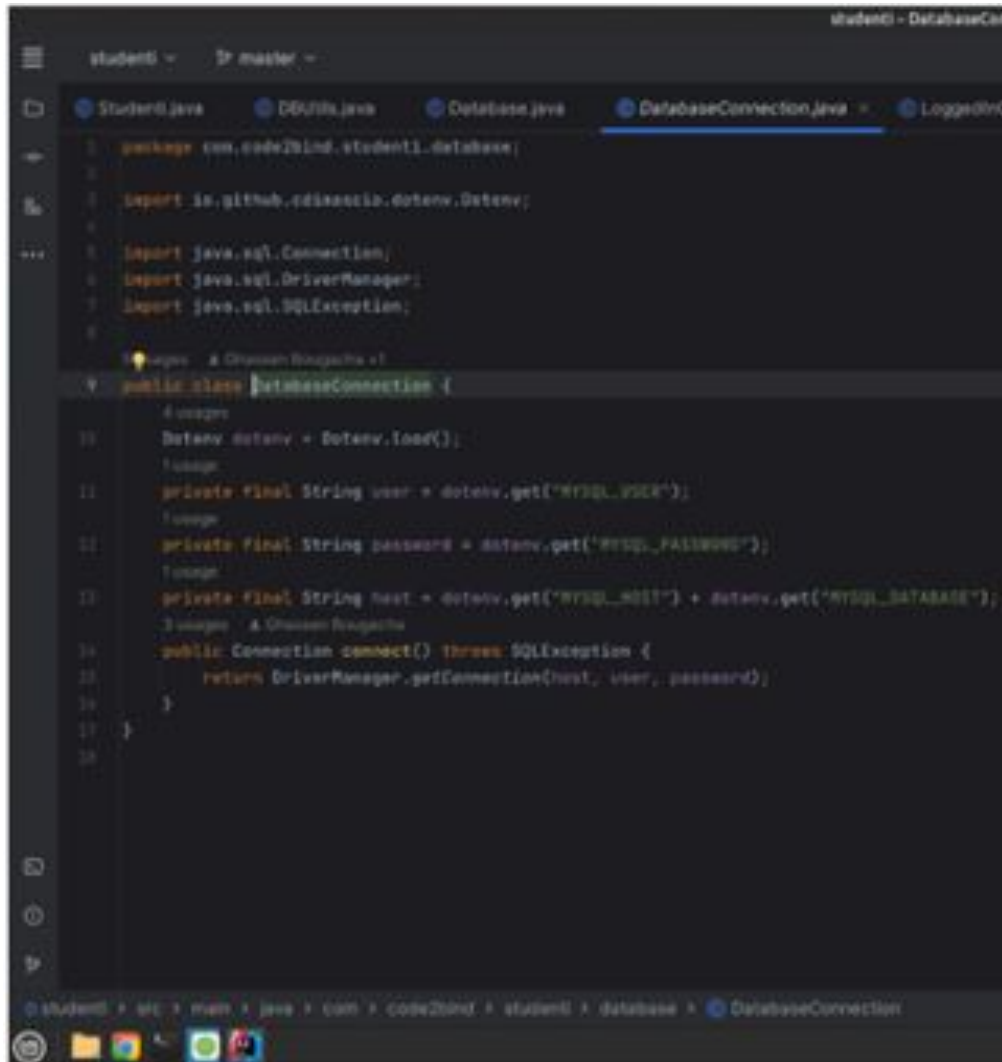
- Dans cette section, nous parlerons de la phase très importante de notre travail que tout notre travail est basé sur.

## Développement en Java :

- La technologie **Java** définit à la fois un langage de programmation orienté objet et une plateforme informatique.
  - Nous avons exactement utilisé le JavaFx qui est un framework et une bibliothèque d'interface utilisateur issue du projet OpenJFX, qui permet aux développeurs **Java** de créer une interface graphique pour des applications de bureau, des applications internet riches et des applications smartphones et tablettes tactiles.



# La Connexion entre base de donnée et Java :

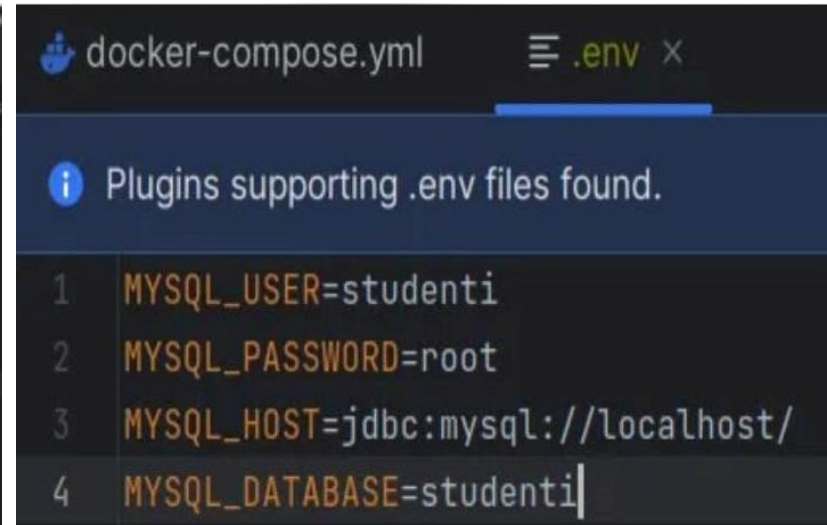


```
student1 - DatabaseConnection.java
package com.code2blind.student1.database;

import io.github.cdimascio.dotenv.Dotenv;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class DatabaseConnection {
    // ...
    private final String user = dotenv.get("MYSQL_USER");
    private final String password = dotenv.get("MYSQL_PASSWORD");
    private final String host = dotenv.get("MYSQL_HOST") + dotenv.get("MYSQL_DATABASE");
    public Connection connect() throws SQLException {
        return DriverManager.getConnection(host, user, password);
    }
}
```

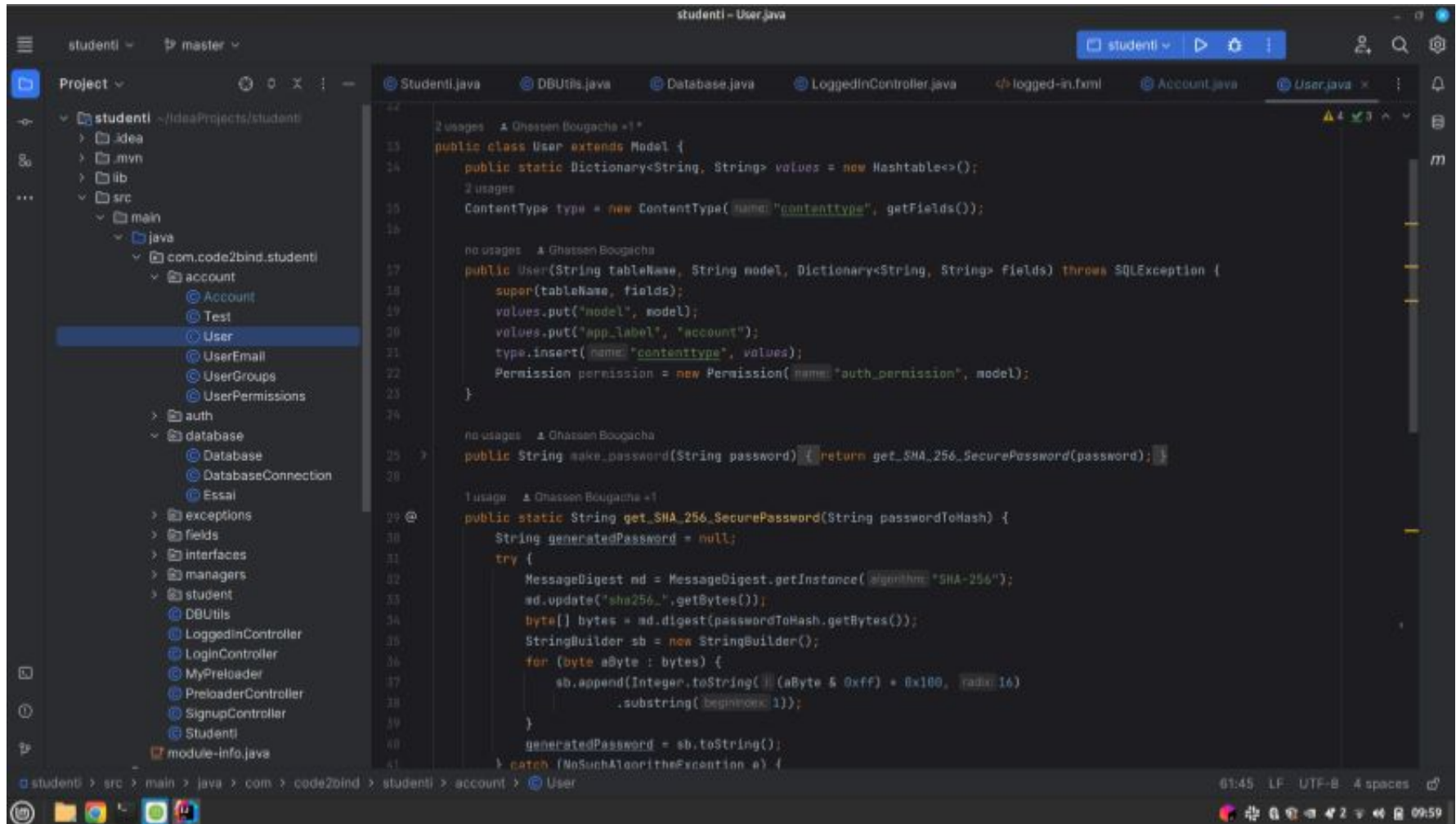


```
docker-compose.yml .env x
i Plugins supporting .env files found.
1 MYSQL_USER=studenti
2 MYSQL_PASSWORD=root
3 MYSQL_HOST=jdbc:mysql://localhost/
4 MYSQL_DATABASE=studenti
```

Notons qu'on a utilisé un fichier .env pour notre environnement, et assurons la sécurité de notre base de données coté serveur et coté développement .

Ce fichier sera caché et facilite la modification dans le serveur après le déploiement

# Class « User »:



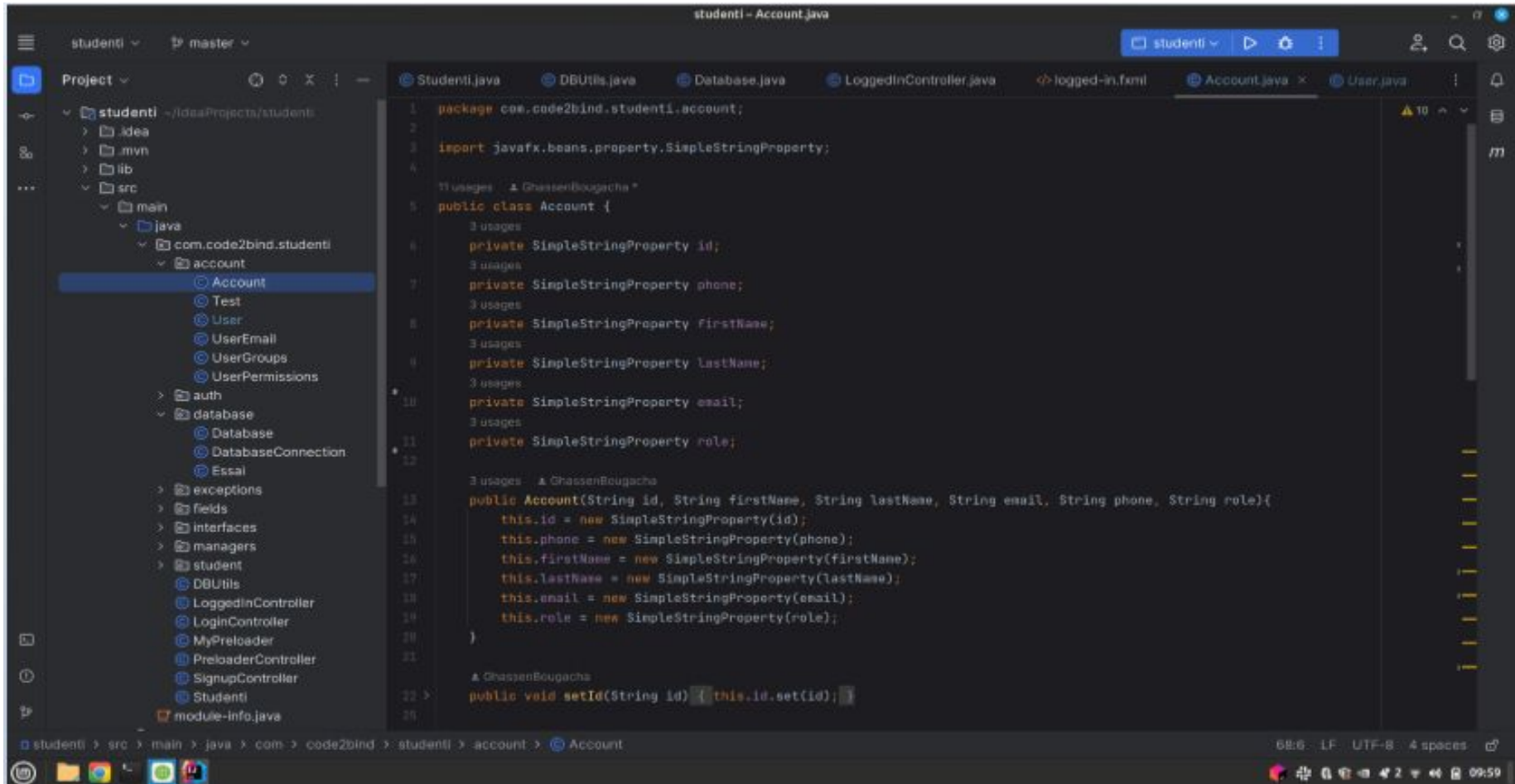
```
student1 - User.java

2 usages  ▲ Ghassen Bougacha +1 *
33 public class User extends Model {
34     public static Dictionary<String, String> values = new Hashtable<>();
35     2 usages
36     ContentType type = new ContentType( name: "contenttype", getFields());
37
38     no usages  ▲ Ghassen Bougacha
39     public User(String tableName, String model, Dictionary<String, String> fields) throws SQLException {
40         super(tableName, fields);
41         values.put("model", model);
42         values.put("app_label", "account");
43         type.insert( name: "contenttype", values);
44         Permission permission = new Permission( name: "auth_permission", model);
45     }
46
47     no usages  ▲ Ghassen Bougacha
48     public String make_password(String password) { return get_SHA_256_SecurePassword(password); }
49
50     1 usage  ▲ Ghassen Bougacha +1
51     public static String get_SHA_256_SecurePassword(String passwordToHash) {
52         String generatedPassword = null;
53         try {
54             MessageDigest md = MessageDigest.getInstance( algorithm: "SHA-256");
55             md.update("sha256_".getBytes());
56             byte[] bytes = md.digest(passwordToHash.getBytes());
57             StringBuilder sb = new StringBuilder();
58             for (byte aByte : bytes) {
59                 sb.append(Integer.toString( (aByte & 0xff) + 0x100, radix: 16)
60                     .substring( beginIndex: 1));
61             }
62             generatedPassword = sb.toString();
63         } catch (NoSuchAlgorithmException e) {
```

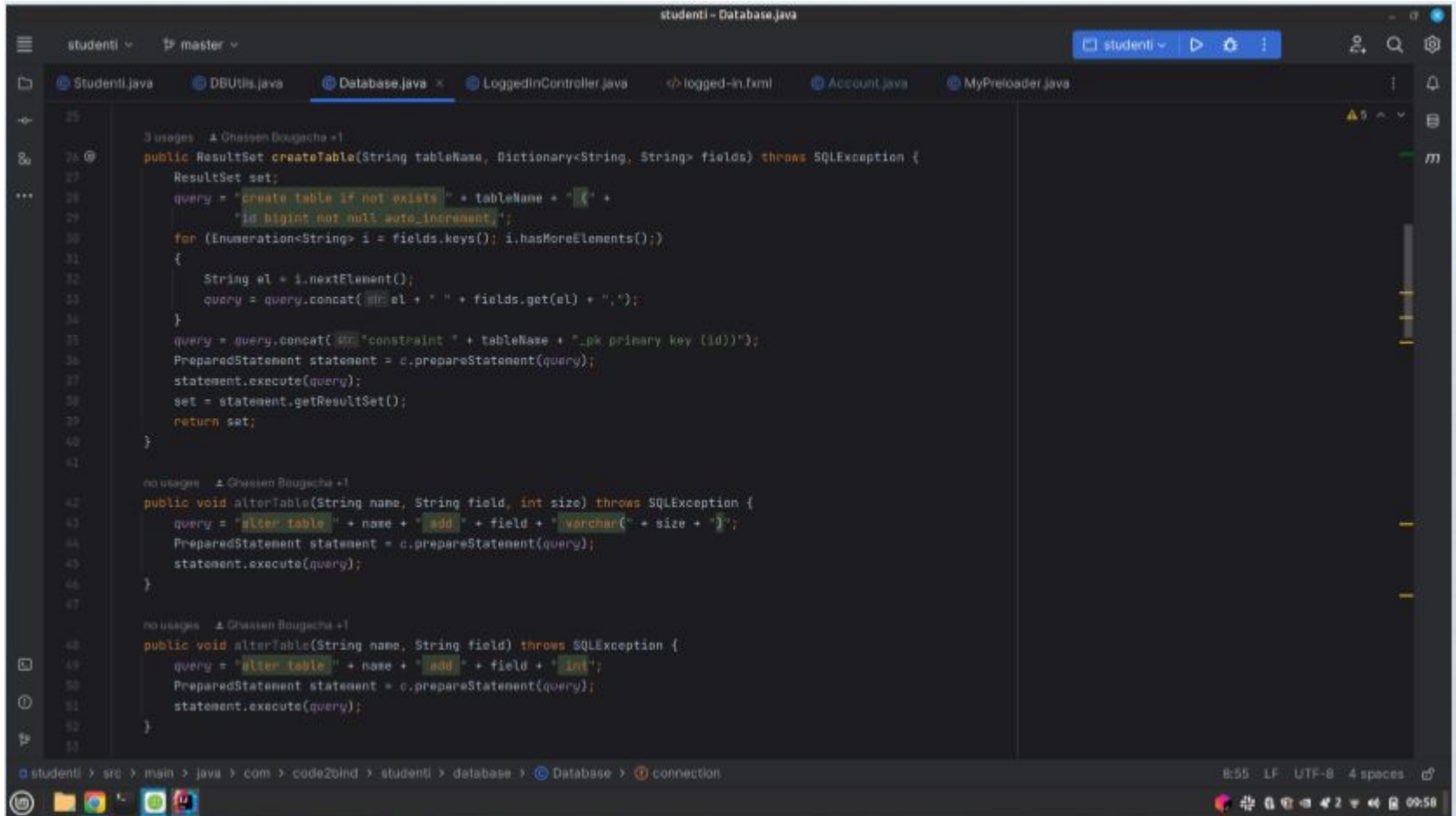
student1 > src > main > java > com > code2bind > student1 > account > User

61:45 LF UTF-8 4 spaces

# Class « Account » :



# Travailler avec la Base de données :



```
studenti - Database.java
studenti ▾ ▶ master ▾
Studenti.java DBUtils.java Database.java LoggedInController.java logged-in.fxml Account.java MyPreloader.java
25
3 usages 1 Ghassen Bougacha +1
26 @
27 public ResultSet createTable(String tableName, Dictionary<String, String> fields) throws SQLException {
28     ResultSet set;
29     query = "create table if not exists " + tableName + " (" +
30         "id bigint not null auto_increment,";
31     for (Enumeration<String> i = fields.keys(); i.hasMoreElements();)
32     {
33         String el = i.nextElement();
34         query = query.concat(" " + el + " " + fields.get(el) + "," );
35     }
36     query = query.concat("constraint " + tableName + "_pk primary key (id))");
37     PreparedStatement statement = c.prepareStatement(query);
38     statement.execute(query);
39     set = statement.getResultSet();
40     return set;
41 }
42
43 no usages 1 Ghassen Bougacha +1
44 public void alterTable(String name, String field, int size) throws SQLException {
45     query = "alter table " + name + " add " + field + " varchar(" + size + ")";
46     PreparedStatement statement = c.prepareStatement(query);
47     statement.execute(query);
48 }
49
50 no usages 1 Ghassen Bougacha +1
51 public void alterTable(String name, String field) throws SQLException {
52     query = "alter table " + name + " add " + field + " int";
53     PreparedStatement statement = c.prepareStatement(query);
54     statement.execute(query);
55 }
56 }
```

studenti > src > main > java > com > code20ind > studenti > database > Database > connection

8:55 LF UTF-8 4 spaces 00:58

```
studentl - Database.java

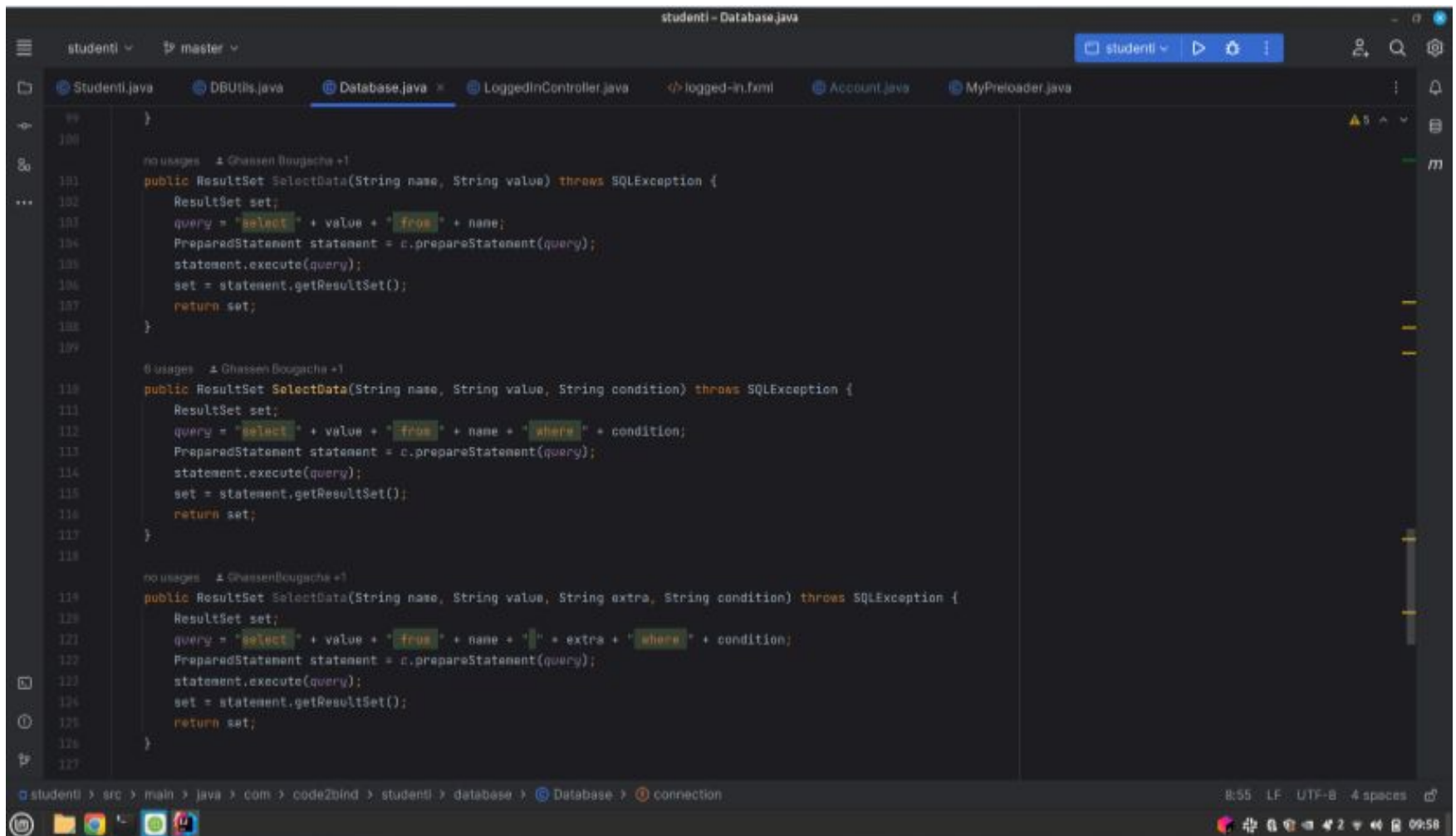
Studentl.java DBUtils.java Database.java x LoggedInController.java logged-in.xml Account.java MyPreloader.java

20 usages A Ghassen Bougacha +1
62 public void alterTable(String name, String field, String key, String reference) throws SQLException {
63     query = "alter table " + name + " add constraint " + name + "." + reference + "_fk " +
64         "foreign key (" + field + ") references " + reference + " (" + key + ")";
65     PreparedStatement statement = c.prepareStatement(query);
66     statement.execute(query);
67 }
68

2 usages A Ghassen Bougacha +1
69 public void InsertData(String name, Dictionary<String, String> values) throws SQLException {
70     query = "insert into " + name + " (";
71     for (Enumeration<String> i = values.keys(); i.hasMoreElements(); )
72     {
73         String el = i.nextElement();
74         if (i.hasMoreElements())
75             query = query.concat(aln el + ", ");
76         else
77             query = query.concat(el);
78     }
79     query = query.concat(") values (");
80     for (Enumeration<String> i = values.keys(); i.hasMoreElements(); ) {
81         String el = i.nextElement();
82         if (i.hasMoreElements())
83             query = query.concat(" " + values.get(el) + ", ");
84         else
85             query = query.concat(" " + values.get(el) + " ");
86     }
87     query = query.concat(")");
88     PreparedStatement statement = c.prepareStatement(query);
89     statement.execute(query);
90 }
91

studentl > src > main > java > com > code20ind > studentl > database > Database > connection
8:55 LF UTF-8 4 spaces
```



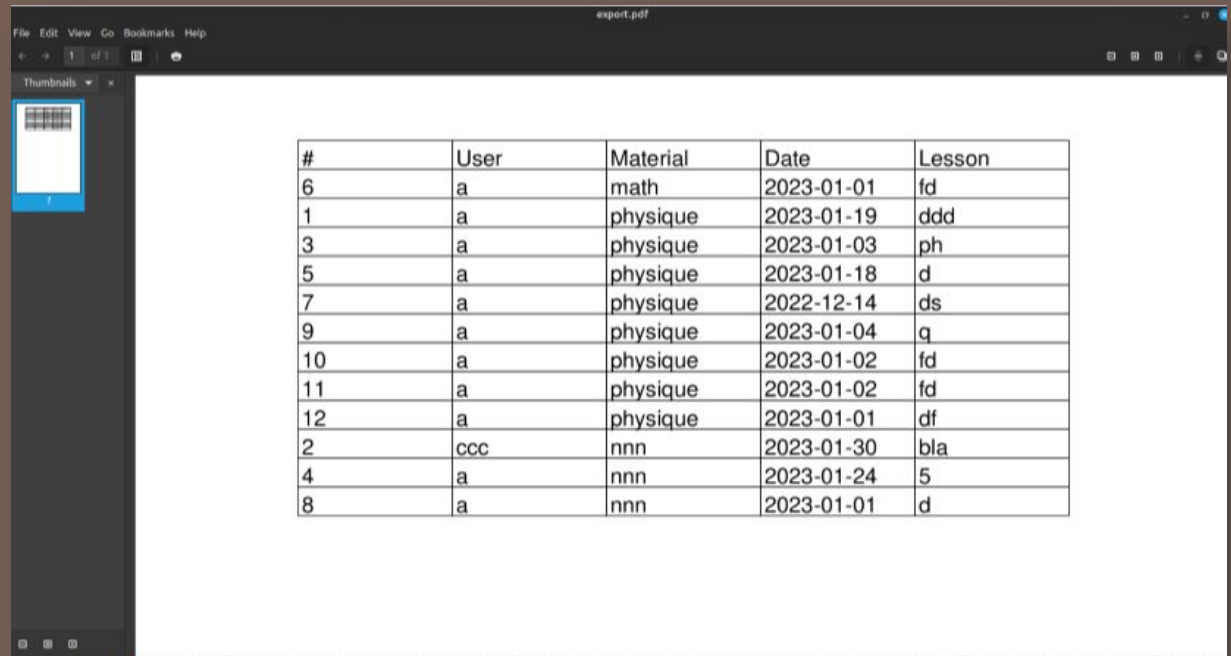




Avant de passer à la conclusion , ce sont deux autres résultats du notre travail :

1/ la consultation des listes d'absences enregistrés

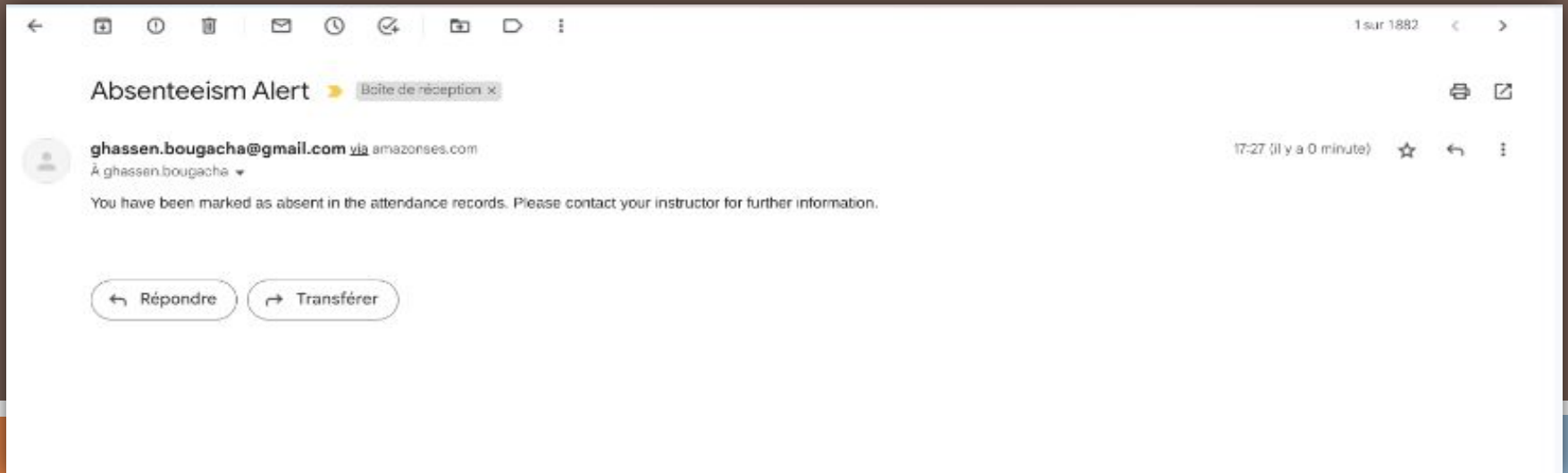
2/l'envoi des mails d'alertes aux étudiants absents



The screenshot shows a PDF viewer window titled 'export.pdf'. On the left is a sidebar with a 'Thumbnails' panel. The main content area displays a table with 5 columns: '#', 'User', 'Material', 'Date', and 'Lesson'. The table contains 15 rows of data.

#	User	Material	Date	Lesson
6	a	math	2023-01-01	fd
1	a	physique	2023-01-19	ddd
3	a	physique	2023-01-03	ph
5	a	physique	2023-01-18	d
7	a	physique	2022-12-14	ds
9	a	physique	2023-01-04	q
10	a	physique	2023-01-02	fd
11	a	physique	2023-01-02	fd
12	a	physique	2023-01-01	df
2	ccc	nnn	2023-01-30	bla
4	a	nnn	2023-01-24	5
8	a	nnn	2023-01-01	d

-1-



-2-

## 6.4-Conclusion :

Voici donc la dernière partie, elle présente toutes les différentes étapes que nous avons franchies pour développer ce module applicatif.

