



RAPPORT

Projet : Gestion d'Etablissement

L3 MIAGE – POO 2018

Année 2017/2018

BEN AISSA OUADIE

BELKHIRI LYES

AOUALI SALIM

3ème Année informatique/POO

Table des matières

1. Introduction

2. Description du projet

2.1. Objectif du projet

2.2. Exigences

3. Analyse, modélisation et conception

4. Gestion du projet

5. Conclusion

6. amélioration

7. Sources

1.Introduction

Dans le cadre de notre projet, nous avons réalisé une application de gestion d'établissement proposée par notre enseignant Mr.BALBALI au sein de notre UE POO. Dans un temps limité, nous nous sommes lancés un défi, qui consiste à réaliser une application qui serait en mesure de gérer les différentes actions classiques qui peuvent être proposées par un système de gestion d'établissement. Nous avons essayé de respecter au maximum le cahier de charge proposé. Nous avons également prévu de rajouter des fonctionnalités avancées, inspirées d'autres applications et des cours que nous avons suivi sur internet que l'on mentionnera à la fin du rapport, mais malheureusement nous n'avons pas assez de temps. Espérant que notre projet soit jugé pertinent par les membres de jurys, nous espérons avoir réussi à relever une partie de ce défi et à appliquer les règles exigées.

Pour mener à bien ce projet, nous avons premièrement commencé à réaliser le diagramme de classe que nous allons suivre, afin d'avoir une vue plus globale, sur les besoins des clients. Cette démarche nous a permis de modéliser les différents objets java, ainsi qu'une solide base de données, qui gère efficacement les différentes fonctionnalités proposées, et qui répond aux différentes attentes des utilisateurs.

2. Description du projet

Le projet réalisé par Ben Aissa Ouadie,BELKHIRI Lyes et AOUALI Salim a pour nom: << School 1.0 >>, celui-ci a pour but de gérer un établissement scolaire. Une application attractive, interactive et dynamique, autant pour son administration que pour son utilisation.

2.1. Objectif

Nous devons réaliser une application qui répond aux différentes attentes du cahier de charge.

2.2. Exigences

- le directeur, les agents et les étudiants doivent s'authentifier afin de pouvoir accéder aux différentes fonctionnalités de l'application.
- l'interface d'administration doit être intuitive et simple.

- L'application doit répondre conformément aux demandes du cahier de charge: une bonne conception des objets ainsi que la minimisation des requêtes vers la base de données et la possibilité d'enregistrer dans la BD les tâches effectuées au moment où l'utilisateur choisit cette action.
- un étudiant n'a uniquement le droit de suivre son assiduité et voir ses résultats.
- adopter un design qui permet aux utilisateurs de repérer les différentes fonctionnalités qui s'offrent à eux.

3. Analyse, modélisation et conception

Pour élaborer ce projet, il nous a fallu créer plusieurs versions afin d'arriver à une version plus ou moins finale. La première nous avons créé toutes les classes demandées car au début nous voulions impliquer toutes les classes dans l'application mais comme nous n'avions pas assez de temps nous avons réduit le nombre de classe.

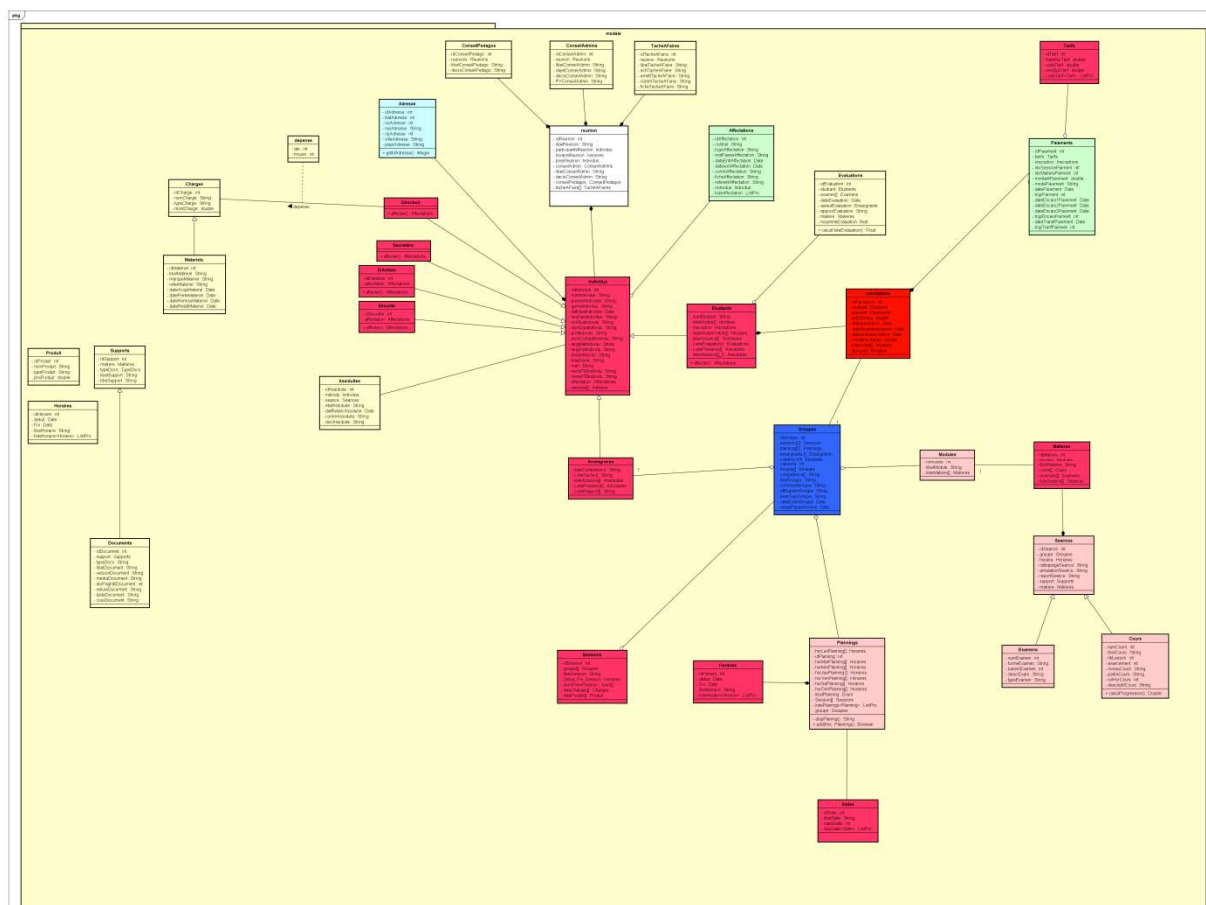


Figure 1: diagramme de classe School1.0

Et pour réaliser l'action enregistrer, nous avons pensé à créer une Classe générique qui contient 4 liste qui va nous permettre par la suite d'enregistrer les différentes actions des utilisateurs afin d'assurer la traçabilité et l'enregistrement des données

passant par les bases de données. Pour cela, nous avons créé dans la classe générique ListPro quatre variables de type HashSet liste :

- la variable liste « métier » correspond à la liste manipulée par l'utilisateur.
- la variable liste « asupprimer » qui permet d'enregistrer les données supprimées par l'utilisateur.
- la variable liste « ainserer » qui permet d'enregistrer les données ajoutées par l'utilisateur.
- la variable liste « amodifier » qui permet d'enregistrer les données modifiées par l'utilisateur.

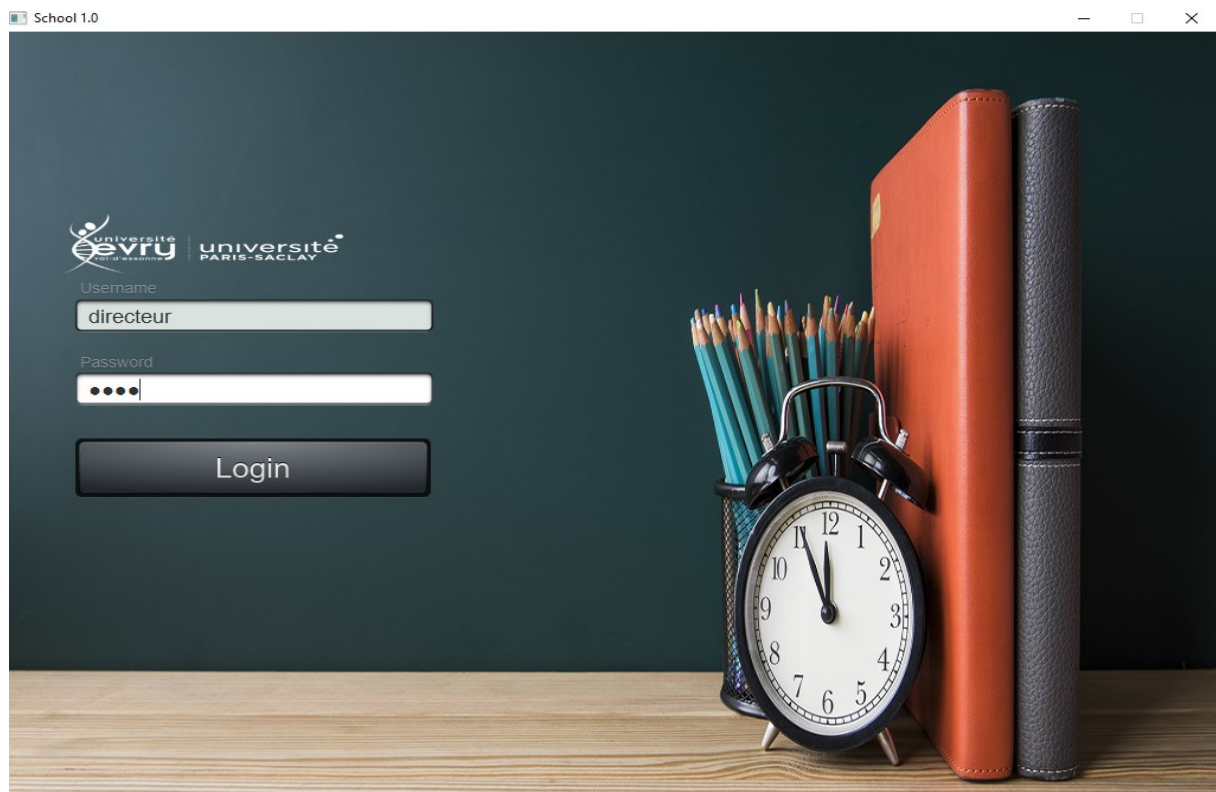
Lorsque l'utilisateur choisi d'enregistrer son travail, l'application parcourt toutes les listes commençant par celles qui contiennent les variables a supprimer, finissant ainsi par celles qui contiennent les variables a modifier, et ce, tout en minimisant l'accès à la base de données.

Nous avons également créé une interface pour le directeur ainsi qu'une seconde, différente, pour les étudiants.

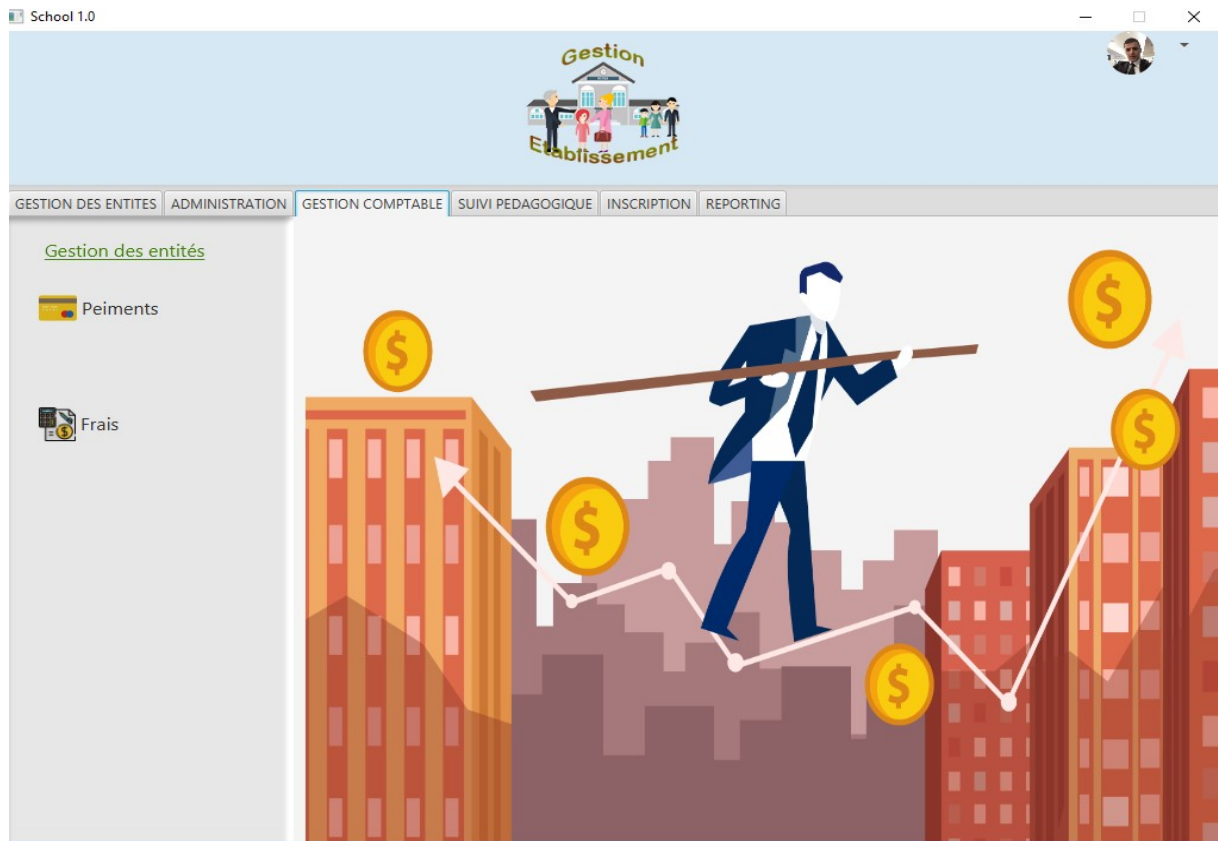
Pour tester l'application en tant que directeur ou étudiant nous avons créé pour le moment des affectations dans le main du projet vous pouvez vous connecter avec le login : « etudiant » ou « directeur » et le mot de passe « pass ». Ainsi vous aurez un Script pour la création de la base de données (nous ne l'avons pas encore connecté).

Nous n'avons pas encore relié toutes les interfaces avec leurs controleurs.

Ci-dessous les différentes interfaces de l'application.



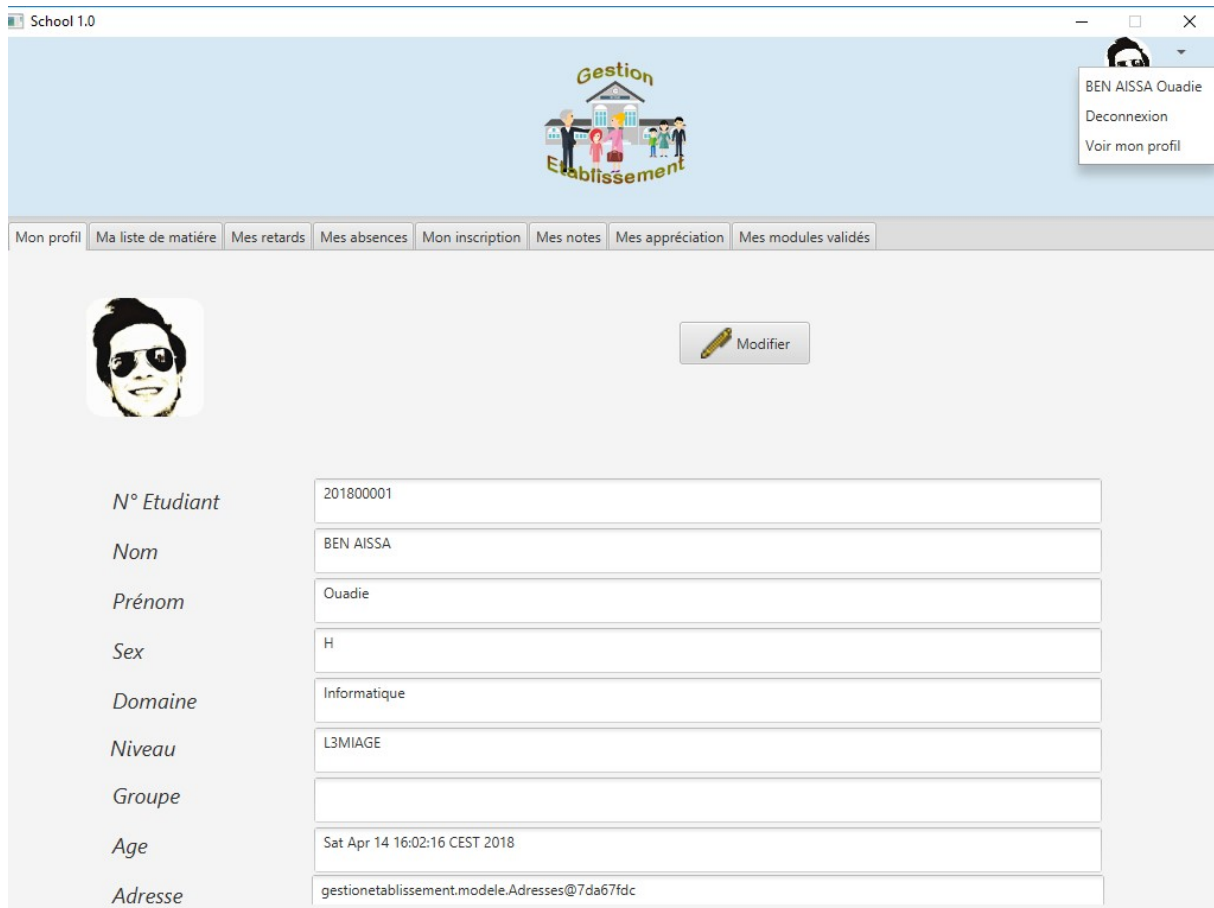






[illegible]

The image is a composite of two distinct scenes. The left scene is a digital login interface for 'School 1.0'. It features a dark teal background. At the top left, there is a small icon of a school building and the text 'School 1.0'. Below this, the logo for 'université de vevru val d'essonne' is displayed, followed by the text 'université PARIS-SACLAY'. The login form consists of a 'Username' field with the text 'etudiant' entered, a 'Password' field with four dots, and a 'Login' button. The right scene is a photograph of a desk setup. It includes a black alarm clock with a white face, a mesh pencil holder filled with teal pencils, and two books standing upright: one with an orange cover and one with a grey cover. The desk surface is light-colored wood, and the background is a dark teal wall.


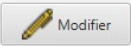


School 1.0

Gestion Etablissement

BEN AISSA Ouadie
Deconnexion
Voir mon profil

Mon profil | Ma liste de matière | Mes retards | Mes absences | Mon inscription | Mes notes | Mes appréciation | Mes modules validés

N° Etudiant	201800001
Nom	BEN AISSA
Prénom	Ouadie
Sex	H
Domaine	Informatique
Niveau	L3MIAGE
Groupe	
Age	Sat Apr 14 16:02:16 CEST 2018
Adresse	gestionetablissement.modele.Adresses@7da67fdc

4. Gestion du projet :

Au début, nous avons divisé les tâches à faire en utilisant l'outil de [gestion de projet informatique proposé par Github](#).

Afin de mener à bien notre projet, il nous a fallu déterminer les différentes tâches, les découper et les répartir équitablement entre nous. Pour cela, nous avons, pendant la première semaine, pris connaissance du sujet, établi les fonctions nécessaires pour remplir les besoins, fait des recherches sur JavaFx, GIT, GitHub, Model MVC, Astah(pour les diagramme), Scene Builder 2.0, et déterminé les tâches à accomplir. Notre projet est composé de 2 parties essentielles : la réalisation du diagramme de classe et la programmation.

4.1. Organisation de l'équipe

L'organisation de l'équipe est une chose importante pour le bon fonctionnement du projet et est faite selon nos capacités, nos connaissances et nos personnalités. Initialement, on a commencé par la réalisation du diagramme de classe, chacun de nous a donné ses idées jusqu'à ce que nous avons obtenu un schéma correcte et cohérent. Ensuite on a commencé la programmation. Pour continuer à avancer, nous avons échangé les tâches. Nous avons travaillé en relative autonomie sur ses tâches en expliquant clairement

chaque modification effectuée et l'utilisation des fonctions réalisées. Etant donné que nous n'étions que trois, nous travaillions beaucoup sur Git et GitHub en expliquant les modifications effectuées à travers les commits. Afin de maintenir la cohésion au sein de l'équipe.

4.2. Difficultés rencontrées dans la gestion du projet et solutions apportées

Malgré cette organisation, il y a eu des soucis avec GitHub. A chaque fois la résolution des conflits nous prend beaucoup de temps. Certaines fonctions sont fortement reliées entre elles mais elles sont effectuées séparément par chacun de nous. Il a fallu alors effectuer des petites réunions presque chaque jours et travailler sur une même machine. Néanmoins, en travaillant ensemble, nous avons pu trouver une méthode de travail efficace et résoudre les problèmes. Nous n'avons pas eu de problèmes d'entente.

5. Conclusion:

Au final, nous avons beaucoup travaillé pendant hors des séances. Nous avons compris qu'il est important d'évaluer les connaissances de chacun, et de faire un point sur nos tâches régulièrement. Notre organisation était plutôt bonne. Il y avait une bonne cohésion de groupe. Nous avons appris à gérer un projet et à faire face aux difficultés ensemble.

6. Amélioration proposée:

Nous proposons d'améliorer :

- ✓ Le niveau de sécurité d'application contre les scripts malveillants.
- ✓ Renforcer le contrôle des données.
- ✓ Améliorer l'interface des étudiants
- ✓ Améliorer l'interface du directeur ainsi de rajouter la possibilité d'afficher les statistiques des charges et produits en tant réelle.
- ✓ Rajouter une interface pour les agents.
- ✓ Effectuer plus de tests pour fixer les différents bugs existants.
- ✓ Mieux gérer les variables de type dates.

5. Sources:

- ✓ OpenClassrooms : <https://openclassrooms.com/courses/apprenez-a-programmer-en-java>
- ✓ <https://www.developpez.net>
- ✓ <http://www.elephorm.com/web-multimedia/formation-java/apprendre-java-les-fondamentaux.html>
- ✓ <https://www.jmdoudoux.fr/java/dej/chap-threads.htm#threads>
- ✓ <http://imss-www.upmf-grenoble.fr/prevert/Prog/Java/Conteneurs/ArrayList.html>
- ✓ <https://openclassrooms.com/courses/presentation-de-la-javadoc>

Autre informations :

Une copie du diagramme en png et un script de création de BD également fournis.