



Rapport sur la Plateforme d'Apprentissage Interactif

Version 1

29 Avril 2024

1 Introduction

La plateforme d'apprentissage interactif est conçue pour fournir une expérience éducative enrichissante, permettant aux utilisateurs d'accéder à divers cours en ligne dans des domaines comme les sciences numériques, la littérature, et les mathématiques. Cette application autonome utilise Java et JavaSwing pour offrir une interface utilisateur conviviale et engageante. Elle vise à encourager l'apprentissage autodidacte à travers des interactions et des évaluations dynamiques.

2 Architecture du Système

La structure du système repose sur plusieurs composants principaux développés en Java, illustrant les principes de l'orienté objet et de la gestion efficace des données.

2.1 Classes Principales

- **Main:** La classe principale qui lance l'application et initialise l'interface utilisateur. Elle gère les interactions de base et la navigation entre les différentes sections de l'application.
- **GestionneurData:** Responsable de la gestion des données locales. Cette classe manipule les informations de session des utilisateurs, les performances et les résultats des quiz à travers une base de données locale.
- **Student et Enseignant:** Représentent les profils utilisateurs. Ils héritent de fonctionnalités communes mais disposent également de méthodes spécifiques à leurs rôles, comme la gestion de contenu pour les enseignants et le suivi des cours pour les étudiants.
- **ConnexionEtudiant et InscriptionEtudiant:** Gèrent les processus d'authentification et de création de compte pour les étudiants.
- **ConnexionEnseignant et InscriptionEnseignant:** Similaires aux classes étudiant mais adaptées pour les enseignants, incluant des vérifications et des fonctionnalités spécifiques aux administrateurs de cours.

2.2 Relations entre les Classes

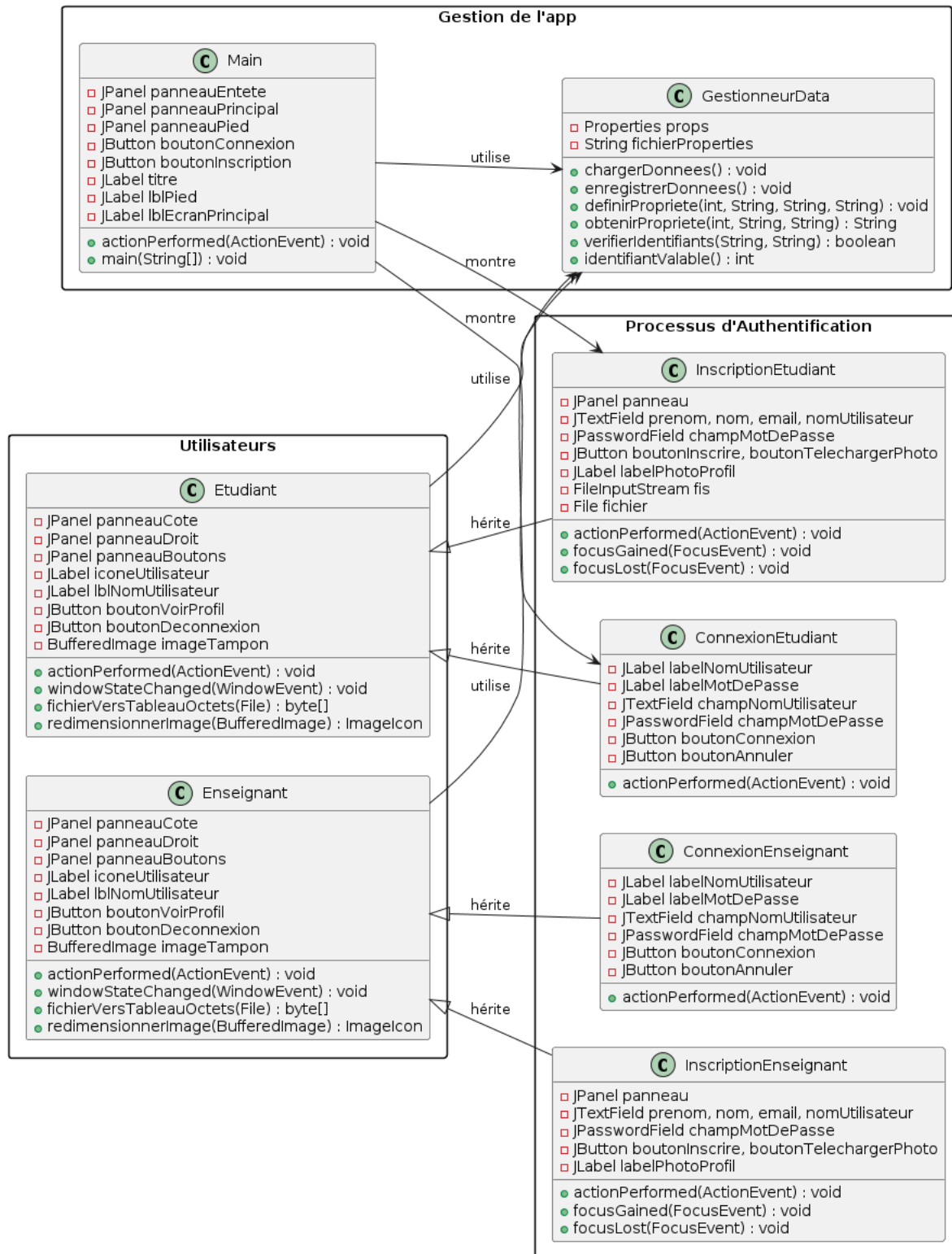
- **Héritage:** Les classes *ConnexionEtudiant*, *InscriptionEtudiant*, *ConnexionEnseignant*, et *InscriptionEnseignant* héritent des fonctionnalités des classes *Student* et *Enseignant*.

- **Utilisation:** La classe *Main* utilise *GestionneurData* pour toutes les opérations de données, et crée des instances des classes d'authentification et d'inscription selon les besoins de l'utilisateur.

3 Les User Stories

- **User Story 1 :** Création d'une interface graphique principale (Main): Une page d'accueil précédée par une page de "Loading", où l'on vérifie si la base de données est bien liée au programme. On vérifie notamment si le fichier `.properties` existe et n'est pas erroné. L'interface graphique principale contient deux boutons initiaux pour se connecter et pour s'inscrire. Cette interface a été développée en utilisant Apache NetBeans, exploitant ainsi Java Swing et Java AWT.
- **User Story 2 :** Création d'une classe `GestionneurData` qui gère la base de données, y compris la création, la sauvegarde et la lecture à partir de cette base pour vérifier si un profil existe ou non, comme illustré dans le diagramme UML.
- **User Story 3 :** Création d'une classe `Student` avec ses différentes fonctions et attributs.
- **User Story 4 :** Développement de classes (fenêtres/JFrame) où chaque classe représente une fonctionnalité spécifique telles qu'Ajouter Cours, Supprimer Cours, Les Participants, Quiz, Inbox, Mail. Ce développement est en cours.
- **User Story 5 :** Création d'une classe `Enseignant` de même utilité que `Student` avec toutes ses classes associées. Cette partie est également en cours.

4 Diagramme UML simplifié



5 Conclusion

Cette plateforme d'apprentissage interactive représente un outil de l'éducation en ligne, fusionnant technologie avancée et pédagogie pour maximiser l'engagement et l'apprentissage des étudiants. Elle évoluera continuellement pour intégrer de nouvelles technologies, assurant ainsi sa pertinence et son efficacité.