****

**Réalisé le Groupe**

**TEAM PROGRAMMEUR**

**Bangoura Mohamed**

**Kamano Tamba Etienne**

**Touré Mohamed**

**THEME : REALISATION D’UNE APPLICATION DE GESTION D’UNE ECOLE A DEUX CYCLES**

**Chargé du cours : M. Moussa Touré**

**REPUBLIQUE DE GUINEE**

Travail-Justice-Solidarité

**CENTRE INFORMAIQUE**

**RAPPORT DE PROJET EN JAVA**

**UNIVERSITE GAMAL ABDEL NASSER DE CONAKRY**

**PROGRAMME : LICENCE 3 DEVELOPPEMENT LOGICIEL**

**----------------------**

**Sommaire**

#### 1. Introduction

* 1.1 Contexte du Projet
* 1.2 Objectifs du Projet
* 1.3 Portée du Projet

#### 2. Description du Projet

* 2.1 Exigences Fonctionnelles
* 2.2 Exigences Non Fonctionnelles

#### 3. Analyse et Conception

* 3.1 Spécification des Exigences : Les Cas d'Utilisation
* 3.2 Analyse du Domaine : Le Diagramme de Classe
* 3.3 La Conception Fonctionnelle
* 3.4 La Conception Graphique

#### 4. Réalisation Technique

* 4.1 Technologies Utilisées
* 4.2 Description de la Structure du Code
* 4.3 Fonctionnalités Implémentées
* 4.4 Tests Réalisés

#### 5. Organisation et Bilan du Projet

* 5.1 Organisation du Groupe : Méthode de Travail
* 5.2 Résultat
* 5.3 Améliorations Envisagées

#### 6. Conclusion

**Introduction**

L'éducation est un pilier fondamental de toute société et la gestion efficace des écoles est essentielle pour assurer un environnement d'apprentissage optimal. Dans ce contexte, le besoin d'un système de gestion scolaire efficace se fait de plus en plus sentir. Ce projet vise à répondre à cette exigence en développant une application de gestion pour une école proposant des programmes à deux cycles, comprenant le collège et le primaire.

**Objectifs du projet**

Le projet de gestion d'une école à deux cycles vise à répondre à ces défis en développant une application logicielle complète et innovante pour l'administration scolaire. Les principaux objectifs de ce projet comprennent :

* Concevoir et développer un système de gestion scolaire complet et convivial pour répondre aux besoins spécifiques de l'école à deux cycles.
* Automatiser les processus manuels liés à la gestion des élèves, des enseignants, des classes et des notes.
* Fournir une plateforme centralisée pour l'administration de l'école afin d'améliorer l'efficacité et la transparence dans la gestion des données scolaires.
* Offrir une interface utilisateur intuitive permettant la gestion facile des informations et des opérations de l'école.

**Portée du projet**

Le projet consistera à développer une application de bureau mono-utilisateur destinée à l'administrateur de l'école. Cette application permettra à l'administrateur de gérer les processus liés aux inscriptions des élèves, aux affectations de classes, à l'enregistrement des enseignants, à la saisie et à la consultation des notes, ainsi qu'à la génération de rapports de performance des élèves. L'application sera développée en utilisant le langage de programmation Java et le framework JavaFX pour assurer la portabilité et l'ergonomie. Le système sera conçu pour être évolutif afin de prendre en charge d'éventuelles extensions futures, telles que l'ajout de fonctionnalités multi-utilisateurs ou l'intégration de modules supplémentaires.

### Analyse des Besoins

#### Description des besoins fonctionnels et non fonctionnels

**Besoins fonctionnels :**

1. Authentification

- Authentifier l'utilisateur

2. Gestion du Collège

2.1 Élèves :

- Inscrire un nouvel élève

- Réinscrire un élève existant

- Visualiser et filtrer les informations des élèves

2.2 Professeurs :

- Inscrire un nouveau professeur

- Visualiser les informations des professeurs

2.3 Notes :

- Noter les élèves par classe, trimestre et matière

- Générer les résultats avec filtre

- Générer, imprimer et enregistrer les bulletins de notes trimestriels et annuels

2.4 Matières :

- Gérer les matières (CRUD)

2.5 Dossiers et historiques :

- Visualiser les dossiers de tous les élèves du primaire

- Consulter les historiques des anciens élèves

3. Gestion du Primaire

3.1 Élèves :

- Inscrire un nouvel élève

- Réinscrire un élève existant

- Visualiser et filtrer les informations des élèves

3.2 Enseignants :

- Inscrire un nouvel enseignant

- Visualiser les informations des enseignants

3.3 Notes :

- Noter les élèves par classe, trimestre et matière

- Générer les résultats avec filtre

- Générer, imprimer et enregistrer les bulletins de notes trimestriels et annuels

3.4 Matières :

- Gérer les matières (CRUD)

3.5 Dossiers et historiques :

- Visualiser les dossiers de tous les élèves du primaire

- Consulter les historiques des anciens élèves

#### 2.2 Exigences Non Fonctionnelles

Les exigences non fonctionnelles définissent les contraintes et les qualités attendues de l'application qui ne sont pas liées à des fonctionnalités spécifiques. Elles peuvent inclure des aspects tels que la sécurité, la performance, la convivialité, etc. Les principales exigences non fonctionnelles comprennent :

#### Sécurité :

Pour garantir la sécurité des données sensibles des élèves, des enseignants et de l'école dans son ensemble, plusieurs mesures ont été mises en place :

* **Authentification sécurisée :** Le système utilise un mécanisme d'authentification robuste pour s'assurer que seuls les utilisateurs autorisés ont accès aux fonctionnalités de l'application.
* **Gestion des droits d'accès :** Les administrateurs ont la possibilité de définir des droits d'accès spécifiques pour différents utilisateurs afin de limiter l'accès aux informations sensibles.
* **Chiffrement des données :** Les données sensibles sont stockées de manière sécurisée en utilisant des algorithmes de chiffrement robustes pour protéger leur confidentialité, notamment les informations personnelles des élèves et des enseignants.

#### Performance :

Pour assurer des performances optimales de l'application, des mesures ont été prises pour minimiser les temps de réponse et maximiser l'efficacité du système :

* **Optimisation du code :** Le code de l'application est optimisé pour garantir des opérations rapides et une utilisation efficace des ressources matérielles.
* **Gestion de la charge :** Le système est conçu pour gérer efficacement les charges de travail élevées, notamment lors des périodes d'inscription et de notation où de nombreuses transactions peuvent survenir simultanément.
* **Mise en cache des données :** Les données fréquemment utilisées sont mises en cache pour réduire les temps de chargement et améliorer la réactivité de l'application.

#### Convivialité :

L'interface utilisateur de l'application est conçue pour être conviviale et intuitive, facilitant ainsi son utilisation par les administrateurs de l'école :

* **Design ergonomique :** L'interface utilisateur est conçue de manière à ce que les fonctionnalités principales soient facilement accessibles et que la navigation soit fluide et intuitive.
* **Guides d'utilisation :** Des guides d'utilisation et des instructions claires sont fournis pour aider les utilisateurs à comprendre les fonctionnalités de l'application et à effectuer des tâches spécifiques.
* **Réactivité de l'interface :** L'interface utilisateur répond de manière rapide et fluide aux interactions de l'utilisateur, offrant ainsi une expérience utilisateur agréable et sans heurts.

#### Portabilité :

L'application est développée en utilisant des technologies et des frameworks qui garantissent sa compatibilité et sa portabilité sur différentes plateformes :

* **Compatibilité multiplateforme :** L'application est développée en utilisant des technologies compatibles avec les principaux systèmes d'exploitation tels que Windows, macOS et Linux, garantissant ainsi sa portabilité sur différentes plates-formes.
* **Adaptabilité à différents périphériques :** L'interface utilisateur est conçue de manière à s'adapter à différents périphériques et résolutions d'écran, offrant ainsi une expérience cohérente sur les ordinateurs de bureau, les tablettes et les appareils mobiles.

### 3. Analyse et Conception

#### 3.1 Spécification des Exigences : Les Cas d'Utilisation

Les cas d'utilisation décrivent les interactions entre les acteurs (utilisateurs) et le système. Nous allons identifier les principaux cas d'utilisation pour chaque fonctionnalité du système, en décrivant les étapes impliquées et les actions de l'utilisateur.

**Diagramme de cas d’utilisation**

<<Include>>

<<extend>>

<<extend>>

#### 3.2 Analyse du Domaine : Le Diagramme de Classe

L'analyse du domaine consiste à modéliser les entités principales du système et leurs relations à l'aide d'un diagramme de classe. Nous allons identifier les entités principales telles que les élèves, les enseignants, les matières, etc., et spécifier leurs attributs et relations.

#### 3.3 La Conception Fonctionnelle

#### 1. Inscription d'un Nouvel Élève

**Description :** Cette fonctionnalité permet à l'administrateur d'inscrire un nouvel élève dans le système.

**Étapes :**

1. L'administrateur accède à la fonctionnalité d'inscription des élèves.
2. L'administrateur saisit les informations personnelles de l'élève telles que le nom, prénom, date de naissance, etc.
3. L'administrateur sélectionne la classe à laquelle l'élève doit être affecté.
4. L'administrateur enregistre les informations de l'élève dans le système.

**Entrées :**

* Nom de l'élève
* Prénom de l'élève
* Date de naissance de l'élève
* Classe à laquelle l'élève doit être inscrit

**Sorties :**

* Confirmation de l'inscription de l'élève avec un numéro de matricule unique généré par le système.

#### 2. Réinscription d'un Élève

**Description :** Cette fonctionnalité permet à l'administrateur de réinscrire un élève existant dans le système pour une nouvelle année scolaire.

**Étapes :**

1. L'administrateur accède à la fonctionnalité de réinscription des élèves.
2. L'administrateur sélectionne l'élève à réinscrire.
3. L'administrateur vérifie et met à jour si nécessaire les informations de l'élève telles que la classe, les coordonnées, etc.
4. L'administrateur enregistre la réinscription de l'élève dans le système.

**Entrées :**

* Sélection de l'élève à réinscrire
* Mise à jour des informations de l'élève si nécessaire

**Sorties :**

* Confirmation de la réinscription de l'élève pour la nouvelle année scolaire.

#### 3. Visualisation et Filtrage des Élèves

**Description :** Cette fonctionnalité permet à l'administrateur de visualiser et filtrer la liste des élèves en fonction de différents critères.

**Étapes :**

1. L'administrateur accède à la fonctionnalité de visualisation des élèves.
2. L'administrateur peut filtrer la liste des élèves par classe, cycle, année scolaire, etc.
3. L'administrateur peut consulter les détails individuels des élèves sélectionnés.

**Entrées :**

* Critères de filtrage tels que classe, cycle, année scolaire, etc.

**Sorties :**

* Liste des élèves filtrés selon les critères spécifiés.

#### 4. Inscription d'un Nouveau Professeur

**Description :** Cette fonctionnalité permet à l'administrateur d'inscrire un nouveau professeur dans le système.

**Étapes :**

1. L'administrateur accède à la fonctionnalité d'inscription des professeurs.
2. L'administrateur saisit les informations personnelles du professeur telles que le nom, prénom, adresse, etc.
3. L'administrateur enregistre les informations du professeur dans le système.

**Entrées :**

* Nom du professeur
* Prénom du professeur
* Adresse du professeur
* Autres informations personnelles du professeur

**Sorties :**

* Confirmation de l'inscription du professeur avec un identifiant unique généré par le système.

#### 5. Visualisation des Professeurs

**Description :** Cette fonctionnalité permet à l'administrateur de visualiser la liste des professeurs enregistrés dans le système.

**Étapes :**

1. L'administrateur accède à la fonctionnalité de visualisation des professeurs.
2. L'administrateur peut consulter la liste complète des professeurs enregistrés dans le système.

**Entrées :** Aucune entrée nécessaire.

**Sorties :**

* Liste des professeurs enregistrés dans le système avec leurs détails.

#### 6. Notation des Élèves

**Description :** Cette fonctionnalité permet aux professeurs de saisir les notes des élèves pour différentes matières, classes et trimestres.

**Étapes :**

1. Le professeur accède à la fonctionnalité de notation des élèves.
2. Le professeur sélectionne la classe, la matière et le trimestre pour lesquels les notes doivent être saisies.
3. Le professeur saisit les notes individuelles des élèves pour la matière sélectionnée.
4. Le professeur enregistre les notes dans le système.

**Entrées :**

* Sélection de la classe, de la matière et du trimestre
* Saisie des notes individuelles des élèves

**Sorties :**

* Confirmation de l'enregistrement des notes pour la classe, la matière et le trimestre sélectionnés.

### 4. Réalisation Technique

La mise en œuvre technique du système de gestion de l'école à deux cycles repose sur un ensemble de choix technologiques et de pratiques de développement rigoureuses. Cette section examine en détail les technologies utilisées, la structure du code, les fonctionnalités implémentées et les tests réalisés pour assurer la qualité et la fiabilité du système.

#### 4.1 Technologies Utilisées

Le choix des technologies est crucial pour garantir le succès du projet. Les technologies sélectionnées offrent une combinaison idéale de performances, de fiabilité et de convivialité pour le développement de notre système :

* **Langage de Programmation :** Java - Choisi pour sa robustesse, sa portabilité et sa popularité dans le développement d'applications d'entreprise.
* **Framework pour l'Interface Utilisateur :** JavaFX - Offre une solution moderne et flexible pour la création d'interfaces utilisateur graphiques attrayantes et ergonomiques.
* **Base de Données :** Oracle Database - Sélectionnée pour sa fiabilité, ses performances élevées et ses fonctionnalités avancées en matière de gestion des données.
* **Environnement de Développement Intégré (IDE) :** IntelliJ IDEA - Offre un ensemble complet d'outils de développement et une interface utilisateur intuitive pour faciliter le processus de développement.

Ces technologies ont été soigneusement évaluées et choisies pour répondre aux besoins spécifiques du projet tout en garantissant une expérience utilisateur optimale.

#### 4.2 Description de la Structure du Code

La structure du code du système est organisée de manière à favoriser la lisibilité, la maintenabilité et l'extensibilité du code. Voici un aperçu de la structure du code :

* **Modèles de Données :** Les classes du package "models" définissent les modèles de données utilisés dans le système, assurant ainsi une représentation claire et cohérente des entités du domaine.
* **Contrôleurs :** Le package "controllers" contient les contrôleurs responsables de la logique métier et de la gestion des interactions entre les vues et les modèles.
* **Vues :** Les fichiers FXML du package "views" décrivent l'interface utilisateur graphique (GUI) de chaque fonctionnalité, offrant ainsi une expérience utilisateur intuitive et conviviale.
* **Utilitaires :** Les classes du package "utils" regroupent les fonctionnalités utilitaires telles que la gestion de la base de données, la génération de rapports, etc., favorisant ainsi la réutilisabilité du code.
* **Tests :** Le package "tests" contient les classes de tests unitaires et d'intégration qui garantissent la qualité du code et la stabilité du système.

Cette structure du code assure une séparation claire des préoccupations et favorise une collaboration efficace au sein de l'équipe de développement.

#### 4.3 Fonctionnalités Implémentées

Le système de gestion de l'école à deux cycles propose un ensemble complet de fonctionnalités conçues pour répondre aux besoins spécifiques de l'administration scolaire. Parmi les fonctionnalités implémentées :

* Authentification sécurisée des utilisateurs.
* Gestion complète des élèves, y compris l'inscription, la réinscription et la visualisation avec fonctionnalité de filtrage.
* Enregistrement et gestion des professeurs, permettant l'inscription et la visualisation des professeurs.
* Système de notation des élèves avec la possibilité de saisir les notes par classe, trimestre et matière.
* Génération automatisée des résultats des élèves avec possibilité de filtrage et de visualisation.
* Fonctionnalité de génération, d'impression et d'enregistrement des bulletins de notes pour chaque trimestre.
* Gestion complète des matières avec des opérations CRUD (Création, Lecture, Mise à jour, Suppression).
* Visualisation des dossiers historiques des élèves, offrant un accès rapide aux informations passées.

Ces fonctionnalités offrent une expérience utilisateur complète et intuitive, tout en simplifiant les tâches administratives et en améliorant l'efficacité opérationnelle de l'école.

#### 4.4 Tests Réalisés

La qualité du système est garantie par une batterie complète de tests, comprenant des tests unitaires, des tests d'intégration, des tests de validation et des tests de performance.

* **Tests Unitaires :** Les tests unitaires vérifient le bon fonctionnement des composants individuels du système, assurant ainsi leur fiabilité et leur conformité aux spécifications.
* **Tests d'Intégration :** Les tests d'intégration valident les interactions entre les différents modules du système, garantissant ainsi l'intégrité de l'ensemble de l'application.
* **Tests de Validation :** Les tests de validation vérifient que le système répond aux exigences fonctionnelles spécifiées et offre une expérience utilisateur cohérente et satisfaisante.
* **Tests de Performance :** Les tests de performance évaluent les performances du système sous différentes charges de travail, garantissant ainsi sa réactivité et sa robustesse dans des conditions réelles d'utilisation.

### 5. Organisation et Bilan du Projet

Dans cette section, nous examinons l'organisation du groupe de développement, évaluons les résultats obtenus et discutons des améliorations envisagées pour le système.

#### 5.1 Organisation du Groupe : Méthode de Travail

L'équipe de développement a adopté une approche méthodique et collaborative pour mener à bien le projet. Voici un aperçu de l'organisation du groupe et de la méthode de travail utilisée :

* **Répartition des Tâches :** Les tâches ont été réparties entre les membres de l'équipe en fonction de leurs compétences et de leurs domaines d'expertise. Chaque membre était responsable d'une ou plusieurs fonctionnalités spécifiques du système.
* **Réunions Régulières :** Des réunions régulières ont été organisées pour discuter de l'avancement du projet, résoudre les problèmes rencontrés et planifier les prochaines étapes. Ces réunions ont favorisé la communication et la collaboration au sein de l'équipe.
* **Utilisation d'Outils de Gestion de Projet :** Des outils de gestion de projet tels que Trello ou Jira ont été utilisés pour suivre les progrès, attribuer les tâches et gérer les dépendances.
* **Tests et Revues de Code :** Des tests unitaires ont été écrits pour chaque fonctionnalité, et des revues de code ont été effectuées régulièrement pour garantir la qualité du code et détecter les éventuels problèmes dès que possible.

#### 5.2 Résultats

Le projet a abouti à la création d'un système de gestion d'école à deux cycles complet et fonctionnel, répondant aux besoins spécifiques de l'administration scolaire. Les principales réalisations comprennent :

* Une interface utilisateur conviviale basée sur JavaFX, offrant une expérience utilisateur intuitive et attrayante.
* Un système robuste d'authentification des utilisateurs assurant la sécurité des données.
* Des fonctionnalités complètes de gestion des élèves, des professeurs, des notes, des matières et des dossiers historiques.
* Des tests rigoureux garantissant la qualité et la fiabilité du système.

#### 5.3 Améliorations Envisagées

Malgré les résultats satisfaisants, il reste des opportunités d'amélioration pour le système :

* **Optimisation des Performances :** Améliorer les performances du système pour garantir une expérience utilisateur fluide, même sous des charges de travail élevées.
* **Ajout de Fonctionnalités Avancées :** Intégrer des fonctionnalités supplémentaires telles que la gestion des absences, la communication avec les parents, ou la génération de rapports statistiques.
* **Amélioration de l'Interface Utilisateur :** Affiner l'interface utilisateur pour la rendre encore plus intuitive et conviviale, en tenant compte des retours des utilisateurs.
* **Évolutivité et Maintenance :** Mettre en place une architecture flexible et évolutive pour faciliter la maintenance et les futures mises à jour du système.

### 6. Conclusion

La conception et la réalisation du système de gestion d'une école publique à deux cycles ont été une entreprise passionnante et enrichissante. Grâce à un effort d'équipe soutenu et à une planification minutieuse, nous avons pu aboutir à un produit fonctionnel et robuste, répondant aux besoins complexes de l'administration scolaire.

Ce projet nous a permis de mettre en pratique nos compétences en développement logiciel, en conception de bases de données et en gestion de projet. Nous avons également acquis une meilleure compréhension des défis et des exigences spécifiques du domaine de l'éducation.

Bref, ce système apporte une contribution significative à l'amélioration de la gestion des écoles publiques. Nous sommes fiers du travail accompli et sommes reconnaissants pour l'opportunité qui nous a été donnée de contribuer à un domaine aussi crucial que l'éducation. Nous restons ouverts aux retours des utilisateurs et aux possibilités d'amélioration continue afin de garantir que notre système reste pertinent et utile dans un environnement éducatif en constante évolution.