Rapport sur le système de gestion scolaire Bawabati

Bawabati est un système de gestion scolaire développé en Python, conçu pour faciliter et automatiser les tâches administratives et pédagogiques au sein d’un établissement éducatif. Ce projet a été réalisé par un étudiant en ingénierie informatique à l’EMSI Casablanca, dans le cadre de sa formation et de la mise en pratique de ses compétences techniques. Il s'agit d'une plateforme centralisée qui permet aux utilisateurs (administrateurs, enseignants, élèves) d'interagir efficacement à travers une interface web.

# 1. Aperçu du Projet

## Nom et Objectif

Bawabati est un système complet de gestion scolaire visant à digitaliser et centraliser les processus pédagogiques et administratifs. Il permet la gestion des cours, l’inscription des élèves, le suivi des notes, la communication entre utilisateurs, et bien plus.

## Technologies Utilisées

- Backend : Python 3.11, Django 5.2  
- Frontend : HTML5, CSS3, JavaScript  
- UI Framework : Bootstrap 5  
- Gestion des formulaires : Django Crispy Forms (Bootstrap5)  
- Base de données : MySQL  
- Outils de développement : Django Debug Toolbar  
- Authentification : Système intégré de Django

## Structure du Projet

Le projet est organisé selon les standards du framework Django, incluant des applications modulaires telles que bawabati\_app, students, teachers, courses, grades. Chaque application gère des composants spécifiques (modèles, vues, templates).

# 2. Rôles des Utilisateurs

Les utilisateurs sont répartis en trois rôles : Administrateur, Enseignant, et Élève. Chaque rôle possède une interface et des autorisations spécifiques adaptées à ses fonctions :  
- Administrateur : gestion globale du système, des utilisateurs, des cours.  
- Enseignant : gestion des cours assignés, des contenus pédagogiques, et suivi des élèves.  
- Élève : inscription aux cours, accès aux ressources, consultation des notes.

# 3. Fonctionnalités Principales

Le système comprend :  
- Authentification avec redirection par rôle  
- Gestion des cours, des notes et des ressources  
- Système d’inscription des élèves aux cours  
- Gestion des fichiers (notes) avec sécurité et permissions  
- Interfaces de tableaux de bord spécifiques à chaque utilisateur

# 4. Modélisation de la Base de Données

Les principaux modèles : UserProfile, Course, Note, Enrollment. Chaque modèle représente une entité centrale du système éducatif et est relié par des clés étrangères pour assurer l'intégrité relationnelle.

# 5. Routage et Vues

Les URL sont organisées autour des actions courantes : connexion, inscription, gestion des utilisateurs, visualisation et édition des cours, ajout de notes, etc. Les vues sont principalement fonctionnelles avec des décorateurs de permission.

# 6. Templates et Interface Utilisateur

L’interface utilise Bootstrap pour un design responsive. Les templates sont séparés par rôle et héritent d’un modèle de base (`base.html`) pour homogénéiser l’UI. Crispy Forms améliore la présentation des formulaires.

# 7. Sécurité et Bonnes Pratiques

Le système met en œuvre :  
- Authentification sécurisée et permissions par rôle  
- Validation des données serveur  
- Protection CSRF, XSS, SQL Injection via Django  
- Contrôle d’accès aux fichiers et journalisation possible

# 8. Améliorations Futures

Les pistes d'amélioration incluent :  
- Ajout d’un système de notation avancé  
- Suivi d’assiduité détaillé  
- Système de messagerie interne  
- Application mobile et notifications  
- Intégration d’un calendrier scolaire et des forums

# Conclusion

Ce projet, réalisé dans le cadre d’un apprentissage en ingénierie informatique à l’EMSI Casablanca, démontre l’application concrète des compétences techniques en développement web, modélisation de bases de données et structuration logicielle. Bawabati constitue une solution évolutive, sécurisée et adaptée aux besoins d’un établissement scolaire moderne.