

I. Projet : Développement d'une Plateforme "Online School" pour la Gestion et la Collaboration Éducative

Contexte :

Le projet consiste à développer une plateforme d'e-learning intuitive et interactive, "Online School", qui offre des fonctionnalités avancées pour la gestion des processus éducatifs et la collaboration entre enseignants, étudiants et administrateurs. Ce système vise à centraliser et numériser les activités éducatives, allant de l'organisation des classes à la gestion des évaluations.

Fonctionnalités Principales :

1. Gestion des accès utilisateurs

- **Administrateur :**
 - Créer et gérer les filières, niveaux, classes, étudiants, enseignants et matières.
 - Affecter les matières aux enseignants.
 - Inscrire les étudiants dans les classes correspondantes.
- **Enseignant :**
 - Se connecter
 - Accéder aux matières qui leur sont assignées.
 - Organiser les cours en chapitres et leçons.
 - Publier des contenus pédagogiques (texte, images, PDF, vidéos).
 - Créer des forums et participer à des chats de classe.
 - Ajouter des quiz et des devoirs avec :
 - Nombre de tentatives autorisées.
 - Questions et propositions de réponses.
 - Identification des bonnes réponses.
 - Ajouter des devoirs :
 - Date limite pour rendre le devoir, ...
- **Étudiant :**
 - Se connecter pour accéder à ses cours, quiz, devoirs et évaluations.
 - Participer aux forums et chats (de classe et b2b avec enseignants).
 - Passer des quiz et devoirs, et visualiser ses notes/moyennes en fin de cours.

2. Gestion des contenus éducatifs

- Les enseignants peuvent publier leurs cours sous les formats suivants :
 - Texte enrichi.
 - Images et infographies.
 - Documents PDF téléchargeables.
 - Vidéos intégrées ou téléchargeables.

3. Collaboration et interaction

- **Forums :** Chaque cours dispose d'un forum où enseignants et étudiants peuvent poser et répondre à des questions.

- **Chat de classe** : Outil de discussion en temps réel pour chaque classe.
- **Chat B2B (Teacher-Student)** : Communication directe et privée entre un enseignant et un étudiant.

4. Gestion des évaluations

- **Quiz et devoirs** :
 - Gérés par les enseignants.
 - Nombre de tentatives limité ou illimité.
 - Correction automatique ou manuelle.
 - Feedback après soumission.
- **Suivi des résultats** :
 - Les étudiants peuvent consulter leurs notes par matière.
 - Moyenne calculée en fin de cours.

II. TAF

Contexte :

Une base de code partiellement fonctionnelle pour une plateforme éducative "Online School" est fournie aux étudiants. Ce projet a pour objectif de compléter, déboguer et améliorer cette application tout en appliquant les bonnes pratiques de test et de validation logicielle. Les étudiants doivent se concentrer sur l'identification des problèmes existants, la mise en œuvre des fonctionnalités manquantes et la garantie de la qualité du code à travers des tests rigoureux.

Travail demandé :

1. Mise en route et compréhension du projet fourni :

- Installer et configurer le projet selon les instructions fournies.
- Analyser la structure du code pour comprendre les fonctionnalités déjà implémentées.
- Repérer et corriger les erreurs ou dysfonctionnements dans le code existant.

2. Développement des fonctionnalités manquantes :

Les étudiants devront ajouter les fonctionnalités suivantes :

- **Gestion des enseignants** :
 - Organisation des cours en chapitres et leçons.
 - Ajout de quiz et devoirs avec gestion des questions, propositions de réponse, et validations.
 - Participation aux forums et aux chats de classe.
- **Gestion des étudiants** :
 - Accès aux cours, quiz, devoirs, et évaluations.
 - Participation aux forums et discussions privées avec les enseignants.

- Visualisation des notes et moyennes en fin de cours.
- **Gestion par l'administrateur :**
 - Création et gestion des filières, niveaux, classes, matières, étudiants et enseignants.
 - Inscription des étudiants dans les classes.

(Voir I)

3. Débogage et amélioration :

- Identifier les bugs existants et les corriger.
- Vérifier que les fonctionnalités de base fonctionnent comme prévu (gestion des utilisateurs, navigation, authentification, etc.).
- Proposer des améliorations pour optimiser les performances et l'expérience utilisateur.

4. Mise en œuvre des tests logiciels :

Les étudiants ont le choix de rédiger leurs tests **avant** ou **après** les phases de débogage et d'ajout de fonctionnalités :

- Tests unitaires pour valider chaque composant isolé.
- Tests d'intégration pour vérifier l'interopérabilité entre les modules.
- Tests fonctionnels pour valider les cas d'usage.
- Tests de validation pour garantir que le système respecte les exigences initiales.

Livrables :

1. Une version corrigée et améliorée de la plateforme "Online School".
2. Documentation du projet incluant :
 - Rapport sur les bugs corrigés.
 - Liste des fonctionnalités ajoutées.
 - Manuel utilisateur pour les fonctionnalités développées.
3. Suite de tests automatisés et leurs résultats.
4. Rapport final de validation détaillant les scénarios de test et leurs succès/échecs.

Critères d'évaluation :

- Qualité du débogage : identification et correction efficace des problèmes.
- Implémentation correcte et complète des fonctionnalités manquantes.
- Cohérence et robustesse des tests réalisés.
- Respect des bonnes pratiques en termes de développement et validation logicielle.
- Qualité de la documentation.