

# Rapport d'évaluation - Tests Logiciels & Intégration Continue

**Projet:** Application de gestion de tâches collaboratives

**Étudiants:**

- **YAMEOGO** Baowendsomme Armel
- **SANGO** Appolinaire

**Enseignant:** KABORE Abdoul Kader

**Institut/Université:** Ecole Polytechnique de Ouagadougou (EPO)

## Introduction

Ce rapport présente le projet d'application de gestion de tâches collaboratives développé dans le cadre du cours Tests Logiciels & Intégration Continue. Le projet a été réalisé en utilisant le framework Django en Python.

## Fonctionnalités de l'application

Le menu de l'application se présente comme suit :

```
-- MENU PRINCIPAL --
1. Se connecter
2. Créer un compte utilisateur
3. Afficher toutes mes tâches
4. Créer une tâche
5. Marquer une tâche comme terminée
6. Modifier une tâche
7. Supprimer une tâche
8. Statistiques
9. Quitter
Entrez votre choix (1-9): █
```

L'application implémente les fonctionnalités clés suivantes:

- **Création de compte utilisateur et connexion**
  - Création de compte utilisateur

```
Entrez votre choix (1-9): 2
*****
Entrez votre nom d'utilisateur: bob
Entrez votre mot de passe: bob
```

- Connexion au compte utilisateur

```
Entrez votre choix (1-9): 1
*****

Entrez votre nom d'utilisateur: bob
Entrez votre mot de passe: bob
Successfully logged in as bob
```

- **Gestion des tâches:**

- Création de nouvelles tâches avec un titre et une description.

```
Entrez votre choix (1-9): 4
*****

Entrez le titre de la tâche: Terminer projet CI/CD
Entrez une description: Empty
Tâche créée: Terminer projet CI/CD
```

- Affichage de la liste des tâches.

```
Entrez votre choix (1-9): 3
*****

***** MES TACHES *****

10 - Terminer projet CI/CD : Empty
*****
```

- Marquage des tâches comme terminées.

```
Entrez votre choix (1-9): 5
*****

Entrez l'id de la tâche: 10
Tache 10 marquée comme terminée!
```

- **Statistiques sur les tâches:**

- Affichage du nombre total de tâches, du nombre de tâches terminées et du pourcentage de tâches terminées.

```
Entrez votre choix (1-9): 8
*****

Statistiques
Nombre total de tâches : 1
Nombre total de tâches terminées : 1
Pourcentage de tâches terminées : 100.0 %
```

- **Tests logiciels:**

- Tests unitaires pour chaque fonctionnalité de base de l'application.
- Couverture de test élevée.

```
python manage.py test
Found 3 test(s).
Creating test database for alias 'default'...
System check identified no issues (0 silenced).
...
-----
Ran 3 tests in 0.005s

OK
Destroying test database for alias 'default'...
```

- **Intégration continue:**
  - Configuration d'un pipeline CI/CD sur GitLab CI/CD.
  - Exécution automatique des tests unitaires à chaque push de code.



```
Run Django tests (3.11.4)
succeeded 9 hours ago in 25s

> ✔ Set up job
> ✔ Run actions/checkout@v4
> ✔ Set up Python 3.11.4
> ✔ Install Dependencies
▼ ✔ Run Tests
  1 ▶ Run cd GestionTache && python manage.py test
  7 Creating test database for alias 'default'...
  8 Found 3 test(s).
  9 System check identified no issues (0 silenced).
 10 ...
 11 -----
 12 Ran 3 tests in 0.005s
 13
 14 OK
 15 Destroying test database for alias 'default'...
> ✔ Post Set up Python 3.11.4
> ✔ Post Run actions/checkout@v4
> ✔ Complete job
```

## Technologies utilisées

- **Langage de programmation:** Python
- **Framework web:** Django
- **Base de données:** SQLite
- **Outils d'intégration continue:** GitLab CI/CD

## Implémentation

L'application a été développée en utilisant le framework Django, qui offre une structure robuste et des outils puissants pour le développement web. Django a été choisi pour sa simplicité d'utilisation, sa flexibilité et sa communauté active.

GitLab CI/CD a été utilisé pour configurer le pipeline d'intégration continue. GitLab CI/CD permet d'automatiser les tâches de construction, de test et de déploiement de l'application.

## Tests logiciels

Des tests unitaires ont été écrits pour chaque fonctionnalité de base de l'application afin de garantir son bon fonctionnement. Les tests ont été automatisés et exécutés dans le cadre du pipeline CI/CD.

## **Intégration continue**

Un pipeline CI/CD a été configuré sur GitHub Actions pour automatiser le processus de construction, de test et de déploiement de l'application. Le pipeline est déclenché à chaque push ou pull request de code vers le référentiel Git. Il exécute les tests unitaires et déploie l'application si les tests réussissent.

## **Conclusion**

Le projet d'application de gestion de tâches collaboratives a été réalisé avec succès en utilisant le framework Django et en mettant en œuvre des pratiques de tests logiciels et d'intégration continue. L'application répond aux exigences fonctionnelles et met en avant les compétences acquises dans le cadre du cours.