|  |
| --- |
| **Rapport de Test et Validation de Logiciel pour le Projet de Blog** |

**Objet :**

Réalisation des tests unitaires, fonctionnels, d'intégration et de performance pour assurer la qualité, la robustesse et l'efficacité des applications **about**, **elenizado**, **oeuvre**, et **website** du projet de blog.

**Résumé**

Ce rapport présente les résultats des tests effectués sur l'ensemble du projet de blog, qui inclut les applications **about**, **elenizado**, **oeuvre**, et **website**. Les tests visent à garantir la stabilité, la sécurité et la performance de l'ensemble du système. Ce processus a couvert les tests unitaires des modèles, les vues et les intégrations entre les différents modules du projet, ainsi que les tests de performance.

**Introduction**

Le projet de blog repose sur plusieurs applications interconnectées, qui permettent de gérer les publications, les événements, les cours, les vidéos, les commentaires, et les informations du site. Le but de ce rapport est de documenter les résultats des tests réalisés sur ces différentes applications, afin de garantir que le système fonctionne de manière fiable et performante avant sa mise en production.

**Contexte du Projet**

Le projet consiste en la création d'un blog dynamique où les utilisateurs peuvent publier des articles, consulter des événements, gérer des cours, répondre à des commentaires et interagir avec le contenu. Chaque application du projet remplit un rôle spécifique :

* **About** : Gestion des informations liées à l'entreprise ou au blog.
* **Elenizado** : Gestion des événements, publications et autres contenus.
* **Oeuvre** : Gestion des poèmes ou œuvres artistiques publiées.
* **Website** : Application principale pour l'affichage du blog, la gestion des newsletters, et l'intégration des données provenant des autres applications.

**Exigences du Projet**

**1. Fonctionnelles :**

* Gestion des publications, événements, cours, vidéos, poèmes, etc.
* Interaction des utilisateurs via les commentaires, réponses et likes.
* Inscription à la newsletter.
* Affichage des œuvres poétiques et des articles du blog.
* Pagination des articles et publications.

**2. Non fonctionnelles :**

* Application rapide et légère.
* Sécurisation des données des utilisateurs et des publications.
* Compatibilité avec les principaux navigateurs et appareils mobiles.

**Objectifs des Tests**

Les tests réalisés visent à :

* Vérifier la conformité de chaque application avec ses spécifications fonctionnelles.
* Assurer une intégration fluide entre les applications.
* Garantir la sécurité des données et la robustesse du système.
* Valider la performance et l'expérience utilisateur.

**Plan de Test**

**Étapes de Mise en Place des Tests :**

1. **Identification des besoins :**
   * Cibler les fonctionnalités critiques : affichage des publications, gestion des commentaires, gestion des poèmes, intégration des événements, et gestion de la newsletter.
2. **Priorisation des tests :**
   * Tests des fonctionnalités essentielles comme les vues, la validation des emails, et la gestion des publications.
3. **Choix des outils :**
   * pytest et Django TestCase pour les tests unitaires et d’intégration.
4. **Exécution des tests :**
   * Tests des modèles pour vérifier la création et la sauvegarde des données.
   * Tests des vues pour valider l'affichage et l'interaction avec l'utilisateur.
   * Tests de performance pour simuler une utilisation élevée.

**Résultats des Tests**

**Résumé des Résultats :**

* Nombre total de tests exécutés : **36**
* Tests réussis : **16**
* Tests échoués : **6**
* Tests en erreur : **14**
* Durée totale des tests : **0,519 secondes**

**Détail des Tests Exécutés :**

**1. Tests Unitaires :**

* **Modèles :**
  + Tests de création, validation, et méthode \_\_str\_\_ des modèles dans **about**, **elenizado**, **oeuvre**, et **website**. Tous les modèles (e.g., SiteInfo, Poesie, Publication, Evenement, Newsletter) ont été validés.
* **Résultats :**
  + Tests réussis pour tous les modèles.
  + Relations entre les modèles (ForeignKey, ManyToMany) vérifiées.

**2. Tests Fonctionnels :**

* **Vues :**
  + Les vues pour l'affichage des publications (index, detail), les poèmes (poeme), les événements, et l'intégration avec les autres modules ont été testées pour s'assurer qu'elles retournent les bonnes données et les bons statuts HTTP.
* **Newsletter :**
  + Inscription à la newsletter avec un email valide et invalide a été testée et validée.
* **Résultats :**
  + Échecs et erreurs ont été rencontrés sur les vues impliquant des données manquantes ou mal configurées, notamment pour les entrées de la newsletter.

**3. Tests d’Intégration :**

* **Interactions entre les modules :**
  + Vérification de l'intégration des modules **website**, **elenizado**, **about**, et **oeuvre** pour afficher correctement les publications, événements, poèmes, et gérer les commentaires.
* **Pagination :**
  + Tests de pagination des publications sur la page d'index et d'affichage des événements.
* **Résultats :**
  + Erreurs d'intégration liées à des dépendances de modèles mal configurées, entraînant des erreurs d'affichage.

**4. Tests de Performance :**

* **Pagination et charge utilisateur :**
  + Tests de performance pour mesurer la charge du système avec plusieurs utilisateurs simultanés.
* **Résultats :**
  + Le système a bien géré une charge modérée, mais des problèmes ont été rencontrés lors des tests avec une charge plus élevée.

**Analyse des Résultats**

Tous les tests ont été réussis dans certaines parties du code, mais plusieurs erreurs et échecs ont été rencontrés. Les échecs sont principalement dus à des problèmes d'intégration, de données manquantes ou mal formatées, et des erreurs dans les vues liées à des objets qui ne sont pas correctement instanciés.

**Problèmes Identifiés :**

* **Modèles mal instanciés** : Certaines vues échouent en raison de dépendances non satisfaites entre les modèles.
* **Données manquantes ou incorrectes** : Certaines entrées de test sont manquantes ou mal formatées, notamment dans la gestion des emails dans la newsletter.
* **Erreurs de pagination** : La pagination fonctionne correctement dans des cas simples, mais échoue sous une charge plus élevée.

**Conclusion**

Le projet de blog, comprenant les applications **about**, **elenizado**, **oeuvre**, et **website**, a passé avec succès une partie des tests. Toutefois, des erreurs et échecs ont été identifiés qui nécessitent une correction. Les problèmes rencontrés sont liés principalement aux interactions entre les modules, aux données manquantes ou mal formatées, et à des erreurs dans certaines vues.

**Recommandation :**

* Corriger les erreurs d'intégration et vérifier que toutes les dépendances entre les modèles sont correctement configurées.
* Revoir la gestion des entrées de la newsletter et la validation des emails.
* Effectuer des tests supplémentaires après les corrections.

Le logiciel est en bonne voie, mais des ajustements doivent être effectués avant la mise en production.

**Professeur :**  
Sédrick Kouagni  
**Étudiant :**  
Brou Kouamé Chrys-David