

Projet Final: Missié Moustass



L'île Maurice a récemment été le théâtre d'écoutes téléphoniques illégales, soulevant des inquiétudes sur la protection des communications sensibles.

L'entreprise **Barbichetz Ltd** opérant dans le secteur financier souhaite:

- Renforcer la confidentialité de ses échanges de message vocaux
- Mandater une équipe de deux développeurs pour concevoir l'application
- Pouvoir **enregistrer** des messages vocaux
- Assurer la **confidentialité** et **intégrité** de ces messages

Ce projet représente une opportunité d'appliquer des concepts clés en cours:

- Sécurité des données, programmation Java, gestion de base de données et DevOps.
- Le binôme devra respecter les délais et assurer la qualité du code tout au long du développement.

L'objectif

Développer une application desktop sous Windows ou Mac qui permettra aux utilisateurs :

- D'enregistrer et lire un message vocal.
- De chiffrer et déchiffrer ce message avec AES-256.
- De vérifier son intégrité avec SHA-256.
- D'assurer un contrôle d'accès sécurisé avec authentification via une base de données SQLite contenant les utilisateurs.

Déroulement du projet

- Le projet sera réalisé en 2 semaines par groupe de 2 programmeurs
- Suivi rigoureux du code via Git
- Utilisations des outils de qualité logicielle.

Mission 1: Développement

- Application desktop Windows ou Mac
- Interface graphique intuitive pour faciliter l'enregistrement vocal.

Mission 2: Chiffrement pour la confidentialité

- Mise en place d'un chiffrement AES-256 des fichiers audio
- Pour garantir leur confidentialité.

Mission 3: Hachage pour l'intégrité

- Intégration d'un hachage SHA-256 pour
- Vérifier l'intégrité des messages stockés.

Mission 4: Gestion Utilisateurs

- Gestion des utilisateurs via une base de données SQLite,
- Avec des opérations CRUD (Create, Read, Update, Delete).

Mission 5: Authentification

- Validation des identifiants et
- Des règles de sécurité du mot de passe via des tests unitaires JUnit.

Mission 6: Versionning

- Gérer les Versions du code sur GitHub
- Avec des démonstrations d'historique et
- De gestion des branches via Eclipse Git.

Mission 7: DevOps

- Mise en place d'une pipeline CI/CD avec
- GitHub Actions pour automatiser les tests et les livraisons.

Mission 8: Qualité du code

- Analyse de la qualité du code avec SonarCloud,
- Suivie de corrections en fonction des rapports générés.

Mission 9: Documentation

- Documentation complète avec JavaDoc
- pour assurer la maintenabilité et la clarté du projet.

Mission 10: Rapport Final

Chaque mission sera notée

Backend & Sécurité

- Langage : Java 17
- Base de données : SQLite (stockage des utilisateurs et des messages)
- Chiffrement : AES-256
- Intégrité des données : SHA-256
- Frameworks : JavaFX pour l'interface graphique

Frontend & Interface Utilisateur

- Technologie : Windows Builder (GUI avec JavaFX)
- Authentification utilisateur : Formulaire sécurisé avec validation des mots de passe

Développement et Versioning

- IDE : Eclipse
- Gestion de version : Git & GitHub
- Branching strategy : main, develop, feature/*, fix/*
- CI/CD : GitHub Actions

Tests et Qualité du Code

- Tests unitaires : JUnit
- Analyse de code : SonarCloud
- Documentation : JavaDoc

Enregistrement vocal :

- Capture de la voix de l'utilisateur en .wav ou mp3
- Lecture du message

Chiffrement des messages :

- Chiffrement et Déchiffrement AES-256
- Avec stockage sécurisé.

Vérification de l'intégrité :

- Génération du hash SHA-256
- Comparaison des hash pour vérification

Authentification

- Gestion des utilisateurs
- Connexion et sécurisation via base SQLite.

Gestion des messages

- Affichage, lecture,
- suppression des enregistrements.

Plan de Travail et Livrables

Le client Barbichetz, opérant dans le *secteur de la finance*, est particulièrement soucieux de la confidentialité de ses échanges.

Afin d'éviter toute *exposition aux risques d'écoute illégale* ou d'interception par des tiers malintentionnés, il a alloué un budget conséquent pour ce projet et *exige* une livraison du logiciel sous *15 jours maximum*.

L'urgence et la sensibilité des données échangées imposent un développement rapide, sécurisé et conforme aux meilleures pratiques en matière de protection de l'information.

1. Jour 1 - 2 :

- Création du repository GitHub et mise en place des branches.
- Définition des besoins et schéma de base de données.
- Développement de l'authentification utilisateur.

2. Jour 3 - 5 :

- Implémentation de l'enregistrement vocal et stockage.
- Mise en place du chiffrement AES-256 et du hashage SHA-256.
- Création des opérations CRUD pour les utilisateurs et messages

3. Jour 6 - 7 :

- Développement de l'interface graphique avec Windows Builder.
- Tests unitaires sur l'authentification et le chiffrement.
- Intégration avec SQLite.

4. Jour 8 - 9 :

- Configuration de GitHub Actions pour les tests CI/CD.
- Analyse et correction des défauts détectés par SonarCloud.

5. Jour 10 - 11 :

- Tests finaux et validation des fonctionnalités.
- Génération de la documentation JavaDoc

6. Jour 12 - 14 :

- Finalisation et correction des derniers bugs.
- Rédaction du rapport technique et démonstration du projet

7. Jour 15:

- Démonstration au client et Evaluation

- Application fonctionnelle (fichier .jar exécutable).
- Code source sur GitHub avec gestion de version via Eclipse Git.
- Base de données SQLite pré-remplie avec des utilisateurs test.
- Tests unitaires JUnit et résultats d'exécution.
- Documentation JavaDoc livrée avec le projet.
- Pipeline CI/CD GitHub Actions en place.
- Rapport technique détaillant l'architecture, les choix technologiques et les résultats des tests.

- **Fiabilité de l'application** : L'application doit être stable et sécurisée.
- **Conformité aux exigences** : Toutes les fonctionnalités demandées doivent être présentes.
- **Qualité du code** : Code lisible, bien structuré et conforme aux bonnes pratiques Java.
- **Gestion de version** : Utilisation efficace de Git avec des commits clairs.
- **Tests et validation** : Couverture des tests unitaires et correction des erreurs.
- **Documentation** : JavaDoc complète et explicative.

Bon Courage !