

## **Diapo 1 : Plan**

« Bonjour à tous,  
Je vais commencer par vous présenter le plan de notre présentation.  
Nous commencerons par le contexte du projet, puis nous découvrirons la plateforme LogMiner. Ensuite, je vous ferai un rappel des travaux réalisés lors des sprints 1 et 2. Nous passerons ensuite au sprint 3, qui correspond à la phase actuelle de développement. Pour finir, nous conclurons avec les perspectives d'évolution et une démonstration finale.  
Commençons donc tout de suite avec le contexte du projet. »

---

## **Diapo 2 : Contexte du projet**

« Les systèmes informatiques modernes génèrent chaque jour un volume considérable de fichiers logs. Ces logs sont essentiels pour surveiller la santé des systèmes, diagnostiquer des incidents et assurer la maintenance.  
Cependant, l'analyse manuelle de ces logs est très longue et chronophage, ce qui génère une forte probabilité d'erreurs humaines. Il est difficile d'identifier rapidement les anomalies critiques et la surveillance en temps réel s'avère quasiment impossible sans outils adaptés. C'est pour répondre à ces enjeux que nous avons lancé ce projet. »

---

## **Diapo 3 : Objectif principal**

« Notre objectif principal est de développer une plateforme performante qui permette une détection rapide et précise des anomalies dans les logs.  
Cette plateforme proposera une interface intuitive, facile à utiliser, et adaptée aux différents profils d'utilisateurs.  
Elle pourra gérer plusieurs projets en simultané et offrira à la fois une analyse en temps réel des logs, ainsi qu'une analyse à partir de fichiers importés, pour plus de flexibilité. »

---

## **Diapo 4 : Présentation de LogMiner**

« Voici les principales technologies que nous avons utilisées pour construire LogMiner.  
Le backend est développé avec Spring Boot, garantissant robustesse et scalabilité.  
Le frontend utilise React et Next.js pour une interface réactive et moderne.  
Nous avons choisi PostgreSQL comme base de données pour sa fiabilité.  
L'authentification est assurée via Microsoft Azure AD combiné avec des tokens JWT pour la sécurité.  
Pour l'intelligence artificielle, nous utilisons un système RAG avec DeepSeek LLM, ainsi que Chainlit pour un chatbot intelligent.  
Le projet est containerisé avec Docker pour faciliter le déploiement, et nous utilisons GitHub Actions pour l'intégration continue et le déploiement automatique.  
Enfin, la qualité du code est contrôlée grâce à SonarQube. »

---

## **Diapo 5 : Rappel : Sprints 1 et 2**

« Durant les deux premiers sprints, nous avons réalisé plusieurs avancées majeures. Le sprint 1, qui s'est déroulé du 24 juin au 8 juillet, a permis la mise en place d'une authentification sécurisée via Azure AD avec gestion des tokens JWT, la conception des premières interfaces de gestion des utilisateurs et de configuration des serveurs, la structuration et le déploiement de la base PostgreSQL, ainsi que le développement d'un tableau de bord dynamique adapté aux rôles des utilisateurs.

Nous avons également implémenté le système RAG pour l'analyse intelligente des logs avec affichage en temps réel, ainsi que des fonctionnalités avancées comme le regroupement automatique des logs similaires et la génération de recommandations personnalisées.

Le sprint 2 a continué sur cette dynamique en consolidant ces fonctionnalités et en préparant les bases pour la suite. »

---

## **Diapo 6 : Sprint 3 (première partie)**

« Le sprint 3 a pour objectif principal d'améliorer la valeur ajoutée de la plateforme en renforçant l'expérience utilisateur, la qualité technique et la maintenabilité du projet. Nous développons un chatbot intelligent avec Chainlit, offrant une interface utilisateur intuitive pour une interaction naturelle avec la plateforme.

Ce chatbot exploite le contexte et l'historique des logs pour fournir des réponses précises et personnalisées.

Par ailleurs, un module avancé de gestion de l'historique des logs a été mis en place, permettant le stockage, la consultation et l'analyse à long terme des logs importés, ainsi que la possibilité de revenir sur des analyses passées pour audit et vérification. »

---

## **Diapo 7 : Sprint 3 (deuxième partie)**

« Toujours dans le sprint 3, nous avons mis en place un pipeline CI/CD automatisé via GitHub Actions.

Cela accélère notre cycle de développement et améliore la fiabilité des déploiements.

Nous avons aussi renforcé la qualité du code avec l'intégration de SonarQube pour l'analyse statique et la détection des vulnérabilités, ainsi que la rédaction et l'exécution de tests unitaires pour assurer la stabilité des fonctionnalités clés.

Enfin, toute la plateforme est containerisée avec Docker, ce qui facilite le déploiement multiplateforme et garantit une meilleure portabilité ainsi qu'une cohérence entre les environnements de développement et de production. »

---

## **Diapo 9 : Conclusion & Perspectives**

« En conclusion, LogMiner propose une solution complète et innovante pour l'analyse automatisée des logs, combinant intelligence artificielle et ergonomie.

Cette plateforme facilite la gestion multi-projets et multi-utilisateurs tout en assurant une analyse précise et en temps réel.

Pour l'avenir, nous envisageons d'intégrer des algorithmes d'analyse prédictive, d'étendre la plateforme à d'autres types de données, et de continuer à améliorer l'interface pour toujours mieux répondre aux besoins des utilisateurs.

Je tiens à remercier chaleureusement toute l'équipe encadrante ainsi que nos collaborateurs pour leur soutien tout au long de ce projet.

Je vous invite maintenant à découvrir la démonstration finale. »