

TP : Gestion d'un Festival de Musique

Objectif du TP

L'objectif de ce TP est de simuler la conception et la mise en place d'un système de gestion pour un festival de musique, en associant développement et infrastructure.

Chaque équipe devra :

- Créer un backend minimal permettant de gérer la programmation et les réservations.
- Définir un plan d'hébergement et de sécurité pour ce backend.
- Mettre en place une méthode de travail claire entre développeurs et administrateurs.

Contexte

L'équipe d'organisation d'un festival de musique a besoin d'un système simple pour :

- Gérer la programmation des artistes.
- Permettre au public de réserver des places pour les concerts.
- Héberger et sécuriser l'application.

Les développeurs et administrateurs doivent travailler ensemble pour assurer le bon fonctionnement du système.

1. Organisation des équipes

Chaque équipe est composée de développeurs et administrateurs système.

- Rôle des développeurs : Concevoir et implémenter un backend minimal.
 - Rôle des administrateurs : Définir l'hébergement, la sécurité et les procédures de déploiement.
 - Travaux communs : Organisation des fichiers, tests et validation.
-

2. Travail des développeurs

Objectif : Concevoir une API REST simple pour gérer les artistes et les réservations.

2.1. Modélisation UML

- Identifier les entités principales (Artistes, Réservations, Scènes).
- Réaliser un diagramme de classes et un diagramme de cas d'utilisation.

2.2. Implémentation d'un backend minimal

- Développer deux endpoints API en PHP ou Node.js :
 - GET /artistes → Liste les artistes programmés.
 - POST /reservations → Enregistre une réservation pour un concert.

2.3. Documentation API

- Rédiger une courte documentation expliquant :
 - Les routes disponibles.
 - Le format des requêtes et réponses.
 - Les éventuelles contraintes techniques.

Collaboration avec les admins infra → Discuter des besoins en base de données et en hébergement.

3. Travail des administrateurs infrastructure

Objectif : Concevoir un plan d'hébergement et de sécurisation de l'application.

3.1. Choix de l'hébergement

- Sélectionner une solution d'hébergement fictive parmi :
 - Serveur physique
 - VPS

- Cloud
- Justifier le choix en fonction des besoins et contraintes.

3.2. Plan de déploiement

- Lister les étapes d'installation de l'application (serveur web, base de données).
- Proposer une organisation des fichiers sur le serveur.

3.3. Sécurité et sauvegarde

- Lister trois bonnes pratiques de sécurité (ex : pare-feu, accès restreints).
- Définir une stratégie de sauvegarde des données.

Collaboration avec les développeurs → Vérifier que l'API est compatible avec l'infrastructure choisie.

4. Travail commun

Objectif : Assurer un fonctionnement fluide entre le développement et l'infrastructure.

4.1. Organisation du workflow

- Définir une méthode de travail commune :
 - Comment les développeurs envoient leur code aux admins infra ?
 - Comment les admins infra récupèrent et testent l'API ?

4.2. Test et validation

- Tester si l'API fonctionne correctement après une installation simulée.
 - Rédiger un court compte-rendu des tests et des ajustements nécessaires.
-

5. Livrables attendus

Pour les développeurs :

- Un diagramme UML (classes et cas d'utilisation).
- Un backend minimal avec 2 endpoints API (mock ou code réel).
- Une documentation concise expliquant l'API.

Pour les administrateurs infra :

- Un plan d'hébergement avec justification.
- Une liste des étapes d'installation de l'application.
- Une check-list de sécurité et sauvegarde.

Travail commun :

- Un workflow clair pour la collaboration.
 - Un test de l'API et un compte-rendu des résultats.
-

6. Évaluation et conclusion

Chaque équipe présentera en quelques minutes :

- Son choix d'architecture et d'hébergement.
- La documentation de l'API et la démonstration du backend.
- Les tests réalisés et améliorations envisagées.

Critères d'évaluation :

- Qualité de la collaboration entre développeurs et administrateurs.
- Bonne compréhension des concepts abordés.
- Clarté des livrables et des justifications.

Bonne chance à tous !