

# Projet Tests unitaires

# Sommaire

- Introduction
- Présentation du projet
- Outils / Langages
- Evaluation
- Soutenance
- Ressources

# Introduction

# Introduction

- Le projet suivant reprendra les notions vues en cours, d'anciennes notions, ainsi que de nouvelles que vous allez pouvoir explorer.
- Le projet sera réalisé en groupe. Au vu de cela, une présentation de schéma reflétant votre collaboration vous sera demandée.
- Vous serez amenés à développer une API REST et un frontend. Une présentation en fin de semaine vous sera demandée.
- Vous serez évalués sur le contenu du projet, votre aisance dans les explications techniques et sur la qualité générale de la soutenance.

Projet

# Le projet : La billetterie du Havre

Située en Normandie, la ville du Havre est reconnue à l'international pour son port permettant le transport de marchandises dans toute la partie nord de l'Europe.

La mairie vient d'inaugurer un nouveau centre culturel pouvant accueillir jusqu'à 7000 personnes et souhaite la mise en place d'une nouvelle billetterie en ligne afin de faciliter les réservations et l'administration de la billetterie.

Le site aura trois grandes parties :

- Administration (Slide 7)
- Utilisateur (slide 8)
- Utilisateur connecté (slide 9 à 10)

# Le projet : La billetterie du Havre (Admin)

L'administrateur devra pouvoir créer les nouveaux événements qui se tiendront dans le centre. Un événement peut se produire sur plusieurs dates. Il y a différents types d'événements disponibles : concert, festival, brocante, jeux vidéo, tout ce qui peut promouvoir la culture.

L'administrateur peut également annuler un événement mais doit pouvoir justifier l'annulation (ex : chanteur hospitalisé, confinement COVID-19, alerte attentat, etc.) pour que les clients sachent pour quelle raison l'événement est annulé.

# Le projet : La billetterie du Havre (Utilisateur)

Arrivés sur le site, les utilisateurs pourront voir la liste des concerts disponibles, avec les différentes dates ainsi qu'une image de l'événement. S'ils souhaitent voir plus de détails, ils pourront cliquer sur un événement pour obtenir davantage d'informations (ex : description, nombre de places restantes, etc.). Si un événement est complet, il faudra afficher « complet ».

Il y aura également une page de planification qui permettra de montrer sur un calendrier la liste des événements prévus.



# Le projet : La billetterie du Havre (Utilisateur connecté)

Une page de connexion et d'inscription sera disponible sur la plateforme. Une personne connectée pourra s'inscrire à un événement.

Lors de l'inscription, l'utilisateur devra renseigner la ou les dates auxquelles il souhaite participer, ainsi que le nombre de personnes qui viendront. Attention, il y aura un système de restriction d'âge, car certains événements peuvent être limités à un âge minimum. Donc, si la personne vient seule, on vérifiera son âge par rapport à l'âge de son compte. Cependant, s'il y a plusieurs personnes, il faudra cocher une case confirmant que l'utilisateur confirme que les personnes avec lui auront toutes l'âge requis pour participer.

# Le projet : La billetterie du Havre (Utilisateur connecté)

Une fois inscrit à l'événement, l'utilisateur verra sur son compte un récapitulatif de sa réservation comprenant la date de réservation, la/les dates, l'heure, le nom de l'événement et le nombre de personnes.

Un bouton « générer un PDF » sera disponible, permettant de créer un PDF contenant ces informations ainsi qu'un code QR représentant l'**UUID** de la réservation. Cela permettra à la billetterie de vérifier facilement l'authenticité de la réservation.

# Le projet : La billetterie du Havre

Remarques :

Pour les tests unitaires, vous êtes libre de choisir votre approche, que ce soit le TDD ou une autre méthode, tant que tous les tests sont fonctionnels et réalistes, notamment avec des tests sur chaque fonction.

La génération de PDF et de code QR peut être réalisée en utilisant des bibliothèques spécialisées.

Outils / Languages

# Outils / Langage

Les langages à utiliser sont :

- ✓ Symfony 6 ou 7 (back)
- ✓ API Platform pour créer une API en Symfony
- ✓ MySQL/MariaDB
- ✓ ReactJS (front)
- ✓ Jest, react-test-library et phpunit pour les tests unitaires
- ✓ Git et github/gitlab pour le versioning
- ✓ Gantt
- ✓ Looping, draw.io ou autres outils pour le diagramme de classe

# Evaluations

# Evaluation

Travail Technique 20 points

Soutenance orale – 20 points

Soutenance



# Soutenance

Votre soutenance est fixée le dernier jour de votre module.

Vous devez préparer une présentation, PowerPoint ou autre.

La première slide doit inclure :

- Votre numéro de groupe
- Nom et prénom des membres du groupe
- Le logo de l'IPSSI
- Un titre bref de la présentation de votre projet

# Soutenance

Votre discours devra respecter la structure suivante :

- Présenter le contexte
- Présenter votre groupe et le rôle de chacun (trello peut être utiliser ici pour appuyer vos propos)
- Présenter la solution proposée / développée (vous pouvez également mettre des diagramme UML2 comme un diagramme de classe).
- Conclusion et retours d'expériences
  - Vous devez faire le rapprochement entre ce que vous avez vu en cours et ce que vous avez réalisé durant le projet

# Ressources

# Ressources

Documentations :

- [Insh.xyz/886ab5](#)
- [Insh.xyz/85438f](#)
- [Insh.xyz/85148b](#)
- [Insh.xyz/e8c152](#)
- [Insh.xyz/cf14c6](#)

# Ressources

- Vidéos :
- [Insh.xyz/611f80](#)
- [Insh.xyz/b8c1b6](#)
- [Insh.xyz/dc1b4c](#)
- [Insh.xyz/0609ba](#)
- [Insh.xyz/39e0fb](#)
- [Insh.xyz/fec20a](#) (Attention : sur ancienne version : symfony 5)

À votre clavier,

Prêt ?

Go !