

Projet Annuel

Introduction

Le projet annuel est un projet pluridisciplinaire mettant en pratique les compétences acquises au cours de la 3ème et de la 4ème année.

Cette année, le sujet du projet sera libre, néanmoins, les technologies à utiliser seront imposées.

Fonctionnalités (15 points)

Gestion de projet & Historisation de code

Le projet doit être organisé en méthode Agile. Vous devez utiliser une plateforme qui vous permet de gérer le projet de cette façon. Tous les membres du groupes doivent travailler sur toutes les fonctionnalités du projet de manière équitable. Vous pouvez utiliser l'une de ces plateformes par exemple (liste non-exhaustive) :

- GitHub Project
- Trello
- Jira

Le code doit être historisé avec Git.

Conception et Interface

L'application doit être adaptée sur ordinateur, mobile, tablette, et plus généralement sur n'importe quelle taille d'appareils les plus utilisés.

Le design doit être cohérent entre les pages et agréable visuellement.

Des maquettes des pages les plus utilisées doivent être conçu en amont et présentée le jour de la présentation.

Framework côté client

Le client Web doit être conçu obligatoirement avec le framework JavaScript Vue.js.

Vous pouvez utiliser les librairies de l'écosystème Vue.js et JavaScript à votre convenance.

Langage côté serveur

Un langage côté serveur doit être adopté pour ce projet, au choix entre l'un de ceux-ci (liste exhaustive) :

- **PHP**
- JavaScript

À savoir que vous pouvez également utiliser les deux si vous le souhaitez, sous la forme de microservices par exemple.

Modélisation, Requêtage SQL & Base de données

L'application doit stocker ses données dans une base de données relationnelle.

La base de données doit obligatoirement être une base de données SQL, et le modèle de données doit être conçu en amont et présenté et justifié le jour de la soutenance.

Vous pouvez utiliser l'une de ces bases de données au choix entre :

- **MySQL**
- MariaDB
- PostgreSQL

Docker

La base de données et le serveur doit obligatoirement être déployée sous la forme d'un conteneur Docker.

Vous devez avoir une infrastructure Docker qui expose uniquement au public un proxy inverse comme (liste non-exhaustive) :

- **Nginx**
- Apache
- Caddy
- Traefik

Tout le reste doit rester interne au réseau Docker et non-exposé publiquement.

Serveur & Nom de domaine

L'application doit être disponible en ligne sur un nom de domaine publique avec certificat SSL validé par les autorité de certification.

Vos conteneurs Docker doivent être déployés sur le serveur en utilisant un Docker Context.

Vous devez créer un utilisateur non-privilégié avec le nom de votre choix, qui sera le seul utilisateur autorisé à déployer les conteneurs Docker, et les conteneurs doivent être déployé en son nom, et pas celui de l'administrateur.

Recherche Naturelle

Le site doit être optimisé pour la recherche naturelle, avec entre autres (liste non-exhaustive) :

- Des titres uniques et cohérents par page
- Des description unique et cohérente par page
- Des pages de contenu avec une stratégie de mots-clés optimisé pour les moteurs de recherche et cohérent par rapport à votre métier

Bonus (5 points)

Authentification à deux facteurs

Vous devez implémenter (sans librairie) une authentification à deux facteurs utilisant un algorithme TOPT (Time-based One Time Password).

Les utilisateurs doivent choisir de pouvoir ajouter s'ils le souhaitent une authentification à deux facteurs via une des applications suivantes (liste non-exhaustive) :

- Google Authenticator
- Microsoft Authenticator

Campagne AdWords

Une campagne de publicité ciblée doit être mise en place pour mettre en avant votre application et votre produit, avec une stratégie claire et autour d'un périmètre géographique précis.

Analytics

Votre application doit être instrumentalisé via des outils Analytics comme Matomo, et doivent remonter les informations suivantes (liste non-exhaustive) :

- Nombres d'utilisateurs ayant visité l'application
- Durée de session moyenne
- Taux de rebond

Gestion des erreurs

Vous devez utiliser un conteneur GlitchTip (ou directement Sentry avec votre compte GitHub Student) et le SDK Sentry pour pouvoir faire remonter les éventuelles erreurs qui pourraient remonter de l'exécution du client Vue.js et du serveur.

Sécurité

Vous devez mettre en place des outils d'analyse de failles de sécurité parmi la liste suivante (non-exhaustive) :

- Snyk.io
- Docker Scout
- SQLMap

Analyser votre serveur et vos applications, construire un rapport de vulnérabilité et montrer les différentes remédiations mises en place pour corriger ces erreurs.

Contraintes

- Rendu au format ZIP
- Pas de lien GitHub, GitLab, BitBucket, ...
- Tous les membres du groupes doivent travailler équitablement
- La soutenance doit être préparée
- La tenue vestimentaire et professionnelle et obligatoire pour la soutenance
- Groupes de 2 à 4

Séances

~~Séance 1~~

~~Réflexion sur le sujet et la constitution des groupes.~~

Séance 2

Validation du sujet et de la constitution des groupes.

Séance 3

Séance de suivi.

Séance 4

Séance de suivi.

Séance 5

Pré-présentation et dernier sprint.