

TP BLANC

LOGE ME

Compétences obligatoires



Description du projet

Site de recherche d'appartements ou maison où il est possible que les anciens locataires notent le logement et son propriétaire et où ce dernier peut aussi noter ses anciens locataires afin de les aider à trouver un nouveau logement plus facilement.

Lien du projet: <https://loge-me.hopto.org/>

Synthèse des compétences personnellement mises en oeuvre

Compétences Obligatoire:

Activité type 1:

Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

- Maquetter une application
 - Réalisation de wireframes pour l'interface desktop et mobile de l'application et de l'api en prenant en compte les cas d'utilisations et les scénarios utilisateurs
- Développer des composants d'accès aux données
 - Utilisation de l'Object Relational Mapping Doctrine et des annotations, les validations sont gérées via les constraints et le composant Validator
 - Création de requêtes sql dans les repository pour récupérer/envoyer des datas (à faire)
- Développer la partie front-end d'une interface utilisateur web
 - Mise en place du frontend sur le site de l'api: authentification, documentation du projet, aperçu des routes et de leur fonctionnement, administration du site (à faire)

- Développer la partie back-end d'une interface utilisateur web
 - Génération de fixtures pour créer des utilisateurs
 - Création d'entités via Doctrine et affinements à la main
 - Création de scripts pour simuler des activités utilisateurs (à faire)
 - Mise en place de l'application de monitoring Sentry pour relever les erreurs pouvant découler de ces activités utilisateurs (à faire)
 - Mise en place de logs pour relever les activités utilisateurs (à faire)
 - Mise en place de jsonWebToken pour sécuriser les échanges entre le frontend et le backend

Activité type 2:

Concevoir et développer la persistance des données en intégrant les recommandations de sécurité

- Mettre en place une base de données
 - Mise en place d'une base de donnée via Doctrine
 - Consultation et Modifications des données en phase de test via phpMyAdmin
 - Modifications effectuées via des migrations générées par Doctrine
 - Mise en place d'un backup de la bdd via tâche cron tout les jours(à faire)
 - Les informations sensibles sont cryptées (à faire, juste password actuellement)
 - Mise en place de rôles admin et utilisateurs

Activité type 3:

Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

- Concevoir une application
 - Rédactions de cas d'utilisations
 - Conception de diagrammes uml (à faire)
- Développer des composants métier
 - Réalisation de tests unitaires (à faire)
- Construire une application organisée en couches
 - Application découpés en plusieurs couches: Entités, Repository, Controllers et Services

Compétences Non Obligatoire:

Activité type 1:

Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

- Développer une interface utilisateur de type Desktop
 - Pas fait

Activité type 2:

Concevoir et développer la persistance des données en intégrant les recommandations de sécurité

- Développer des composants dans le langage d'une base de donnée
 - Création de requêtes sql dans les repository pour récupérer/envoyer des datas (à faire)

Activité type 3:

Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

- Collaborer à la gestion d'un projet informatique et à l'organisation de l'environnement de développement
 - Utilisation de l'outil de versionning git et d'hébergement github
 - Utilisation de google docs partagés pour la réalisation de la documentation
 - Utilisation de Trello pour le suivi des plannings
 - Mise en place d'un journal de bord pour le suivi personnel
- Développer une application mobile
 - Non fait, non prévu
- Collaborer à un projet informatique et à l'organisation d'un environnement de développement
 - Mise en place d'un serveur hébergé chez moi tournant sur un raspberry pi 3 via Ubuntu Mate, ce serveur héberge notre environnement de test à savoir la bdd, l'api et l'application
 - Nom de domaine gratuit via no ip à renouveler tous les 30 jours
 - Certificat en https via lets encrypt
- Préparer et exécuter le déploiement d'une application
 - Mise en place de déploiement via l'outil deployer (<https://deployer.org/>) qui déploie selon ce qui est hébergé sur la branche master sur github, va ensuite mettre à jour composer sur le serveur, effectuer les migrations de la bdd et exécuter les fixtures, copier les clés jwt et le fichier .env