

Modèle de copie : Évaluation en cours de formation



Développeur Web et Web Mobile

GDWDWWMECFIII1A

Ceci est un modèle de copie. N'oubliez pas de renseigner vos prénom/nom, ainsi que le nom et le lien vers le projet.

Vous pouvez bien sûr agrandir les cadres pour répondre aux questions sur la description du projet si nécessaire.

Prénom : Ludovic

Nom : VACHON

ATTENTION ! PENSEZ À RENSEIGNER VOS NOM ET PRÉNOM DANS LE TITRE DE VOS FICHIERS / PROJETS !

Nom du projet : Hypnos

Lien Github du projet : <https://github.com/ludooo2204/hypnos>

Lien Trello : <https://trello.com/invite/b/TE16OXmE/2e309d2807464af65b5244ca2f9ade76/hypnos>

URL du site (si vous avez mis votre projet en ligne) : www.lomano.fr

Description du projet

1. Liste des compétences du référentiel qui sont couvertes par le projet

Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité :

- Maquetter une application
- Réaliser une interface utilisateur web statique et adaptable
- Développer une interface utilisateur web dynamique
- Réaliser une interface utilisateur avec une solution de gestion de contenu ou e-commerce

Développer la partie back-end d'une application web ou web mobile en intégrant les recommandations de sécurité :

- Créer une base de données

- Développer les composants d'accès aux données
- Développer la partie back-end d'une application web ou web mobile
- Elaborer et mettre en oeuvre des composants dans une application de gestion de contenu ou e-commerce

2. Résumé du projet en français d'une longueur d'environ 20 lignes soit 200 à 250 mots, ou environ 1200 caractères espaces non compris

Le but de ce projet est de proposer une application web à un groupe hotelier « HYPNOS » afin de proposer à leurs clients une solution de réservation en ligne autre que les portails généraliste (booking.com, etc...). Ce projet à été réalisé dans le cadre de l'ECF Studi pour l'obtention du titre RCNP « Développeur Web et Web Mobile » du 14/03/2022 au 21/04/2022.

Cette application web doit permettre à un administrateur du groupe de pouvoir créer, modifier et supprimer des établissements en totale autonomie via un panel d'administration sécurisée. Elle doit également lui permettre de nommer des gérants d'établissements.

Ces gérants ont, eux aussi, accès à un panel d'administration dédié à la gestion de leur propre établissements. Ce panel leur permet de créer, modifier et supprimer leurs suites.

Ces panels d'administrations sont destinés à employés du groupe. A ce titre, l'application doit être simple, ergonomique et facile d'accès. Un manuel d'utilisation sera fourni à Hypnos.

Les clients ou visiteurs du site ont la possibilité de naviguer à travers les différents établissements et leurs suites associées. Ils pourront réserver une suite uniquement s'ils ont créé un compte et s'ils sont connectés. Ces utilisateurs auront la possibilité de consulter leur réservations passées et d'en annuler seulement s'il reste plus de 3 jours avant l'échéance. Une section du site permettra de contacter l'administrateur du site pour avoir des informations complémentaires. Ces demandes seront consultables sur la messagerie Email de l'administrateur désigné.

3. Cahier des charges, expression des besoins, ou spécifications fonctionnelles du projet

- Une unique personne, l'administrateur, peut gerer les etablissements du groupe. Elle doit pouvoir créer un hotel avec un nom, une adresse, une description et une image de presentation. Elle peut nommer un utilisateur existant en manager d'un hotel. Elle pourra modifier, à tout moment, une de ces informations. Elle aura également la possibilité de supprimer un établissement.
- Un gérant peut lui créer, modifier et supprimer des suites de son hotel pour lequel il a été nommé. Il pourra modifier les informations lié aux suites mais pas à l'hotel. Les suites auront un titre, une image principale (mise en avant) , une galerie de photo, un prix, un lien vers le site BOOKING.com
- Un visiteur pourra visiter les etablissements et les suites. Il doit pouvoir contacter l'administrateur du groupe pour poser certaines questions. Il pourra créer un compte avec son nom, prénom et adresse mail. Il pourra ainsi reserver une suite. Il aura également la possibilité de consulter ses anciennes reservations et d'en annuler une s'il reste plus de 3 jours avant échéance. Le client pourra reserver via un lien « reserver » qui lui permettra de choisir entre differents hotels ou directement via la page de la suite avec un bouton reserver.
- La sécurité des données du site est un élément prioritaire. Pour tout type de compte, les mots de passe doivent etre sécurisés, ils ne doivent pas apparaitre dans la base de données du serveur.
- Le site doit correspondre au modèle « single page application » afin d'être réactif pour un utilisateur. Il doit etre optimisé dans ce sens.
- L'affichage du site doit également être adapté aux tablettes et téléphones.
- Le design du site doit respecter la charte graphique du groupe et garder notre esprit « minimaliste mais luxueux ».

4. Spécifications techniques du projet, élaborées par le candidat, y compris pour la sécurité et le web mobile

Pour le front-end :

Le choix s'est porté sur React Js afin de répondre au besoin du site en SPA (Single Page Application). Cela permet d'avoir un site où la navigation est très rapide.

Avec ce framework, j'ai utilisé les librairies suivantes :

- react-router-dom : afin d'avoir un routing maîtrisé côté navigateur
- Axios : pour simplifier les requêtes ajax pour l'échange de données avec le serveur
- Material-ui : pour avoir des icônes reconnues du grand public (login, logout, supprimer...)
- Match-sorter : pour avoir une recherche dynamique de données
- React-modal : pour simplifier l'utilisation de modal

Pour le back-end :

Le choix s'est porté sur du nodeJs avec l'utilisation du framework Express. Il y a un gros avantage d'utiliser le même langage de programmation côté client et côté serveur. Cela permet de simplifier les échanges sans jongler en deux langages différents. La rapidité de node et son système non-bloquant semble tout à fait adapté à ce projet à fort trafic potentiel. Les librairies suivantes ont été utilisées :

- sequelize : ORM permettant de simplifier les requêtes avec la base de données SQL
- bcryptjs : pour hasher les mots de passe et de ne pas les avoir en « clair » dans la base de données (sécurité first!!!)
- jsonwebtoken : pour simplifier la manipulation des tokens d'identification
- nodemailer : pour l'envoi de mail via le serveur.

5. Description de la veille, effectuée par le candidat durant le projet, sur les vulnérabilités de sécurité

Mon choix s'est rapidement porté sur react JS pour le front end. Sa capacité à segmenter une application en composant permet de gérer facilement une application relativement complexe. Sa rapidité, une fois chargée, n'est plus à démontrer. Le choix de react JS peut également simplifier le passage à une application mobile (avec react native) ; le code étant en grande partie réutilisable.

Pour le backend, le choix de nodeJS peut être plus discutable. Les hébergements (et les développeurs pour une maintenance ou évolution futur) en node sont moins nombreux que du php classique. Cependant, certains avantages sont indéniables. Nodejs et son framework le plus connu, Express, sont extrêmement rapide pour le traitement de multiple petite requêtes. Cela coïncide parfaitement avec le framework client reactJS qui « s'alimente » de données via des appels à l'API du serveur.

Pour la sécurisation de l'application, les routes (coté navigateur) réservées à l'administrateur ou aux gérants ne sont accessible que si l'utilisateur est connecté avec un compte adapté (composant non rendu si pas autorisé).

Les routes de l'API, pour leur part, sont également protégées des accès non autorisés grâce à des middleware d'authentications.

6. Description d'une situation de travail ayant nécessité une recherche, effectuée par le candidat durant le projet, à partir de site anglophone

J'avais besoin d'informations complémentaires sur le fonctionnement de l'authentification avec les JSON web token. Ce site montrait des exemples d'utilisations de ces tokens avec les technologies que j'avais choisis pour le backend ; à savoir node / express et sequelize.

<https://www.bezkoder.com/node-js-jwt-authentication-mysql/>

7. Extrait du site anglophone, utilisé dans le cadre de la recherche décrite précédemment, accompagné de la traduction en français effectuée par le candidat sans traducteur automatique (environ 750 signes).

Extrait Anglophone.

Token Based Authentication.

Comparing with Session-based Authentication that need to store Session on Cookie, the big advantage of Token-based Authentication is that we store the JSON Web Token (JWT) on Client side: Local Storage for Browser, Keychain for IOS and SharedPreferences for Android... So we don't need to build another backend project that supports Native Apps or an additional Authentication module for Native App users.

Traduction française :

Authentification base sur les tokens (jeton en français, mais l'anglicisme reste adaptée).

En comparaison avec l'authentification basée sur les sessions qui ont besoin de sauvegarder la session sur un « Cookie », le gros avantage des authentifications avec les tokens est que l'on sauvegarde les JSON Web Token (JWT) coté client : « Local storage » pour les navigateurs, « keychain » pour IOS et « SharedPreferences » pour Android... Ainsi , nous n'avons pas besoin de construire un autre projet backend qui supporte les applications native ou un module d'authentification supplémentaire pour les utilisateurs d'applications natives.

8. Autres ressources

/

9. Informations complémentaires

L'application n'est malheureusement pas parfaite. Il m'a manqué du temps afin mieux structurer le code, d'y ajouter des commentaires. Le système de perte de mot de passe n'a pas été fait. Le responsive design nécessiterai également quelques améliorations.