|  |  |  |
| --- | --- | --- |
|  |  |  |
| *Nom de naissance* |  | Marais |
| *Nom d’usage* |  | Marais |
| *Prénom* |  | Roxanne |
| *Adresse* |  | 69 chemin de l’Aubarède, les Mirandoles bât O  06 110 Le Cannet |
|  |  |

|  |  |
| --- | --- |
| **Titre professionnel visé** | |
|  | |
| Développeur web et web mobile | |
|  | |
| **Modalité d’accès :** | |
|  | |
| ☒ | Parcours de formation |
| ☐ | Validation des Acquis de l’Expérience (VAE) |
|  | |

|  |  |
| --- | --- |
| **Présentation du dossier** | |
|  | |
|  | |
| Le dossier professionnel (DP) constitue un élément du système de validation du titre professionnel. **Ce titre est délivré par le Ministère chargé de l’emploi.**  Le DP appartient au candidat. Il le conserve, l’actualise durant son parcours et le présente **obligatoirement à chaque session d’examen**.  Pour rédiger le DP, le candidat peut être aidé par un formateur ou par un accompagnateur VAE.  Il est consulté par le jury au moment de la session d’examen.  **Pour prendre sa décision, le jury dispose :**   1. des résultats de la mise en situation professionnelle complétés, éventuellement, du questionnaire professionnel ou de l’entretien professionnel ou de l’entretien technique ou du questionnement à partir de productions. 2. du**Dossier Professionnel** (DP) dans lequel le candidat a consigné les preuves de sa pratique professionnelle. 3. des résultats des évaluations passées en cours de formation lorsque le candidat évalué est issu d’un parcours de formation 4. de l’entretien final (dans le cadre de la session titre).   *[Arrêté du 22 décembre 2015, relatif aux conditions de délivrance des titres professionnels*  *du ministère chargé de l’Emploi]*  **Ce dossier comporte :**   * pour chaque activité-type du titre visé, un à trois exemples de pratique professionnelle ; * un tableau à renseigner si le candidat souhaite porter à la connaissance du jury la détention d’un titre, d’un diplôme, d’un certificat de qualification professionnelle (CQP) ou des attestations de formation ; * une déclaration sur l’honneur à compléter et à signer ; * des documents illustrant la pratique professionnelle du candidat (facultatif) * des annexes, si nécessaire. | |
| *Pour compléter ce dossier, le candidat dispose d’un site web en accès libre sur le site.* | |
|  | [**http://travail-emploi.gouv.fr/titres-professionnels**](http://travail-emploi.gouv.fr/titres-professionnels) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Sommaire** | | | |
| **Exemples de pratique professionnelle** | | |
| **Développer la partie front-end d’une application web ouweb mobile sécurisée** | | **p.** | **5** |
|  | Maquetter une interface utilisateur p. | p. |  |
|  |  Réaliser une interface utilisateur statique p. | p. |  |
|  | Développer une interface utilisateur dynamique p | p. |  |
|  |  |  |  |
| **Développer la partie back-end d’une application web ou web mobile sécurisée** | | **p.** |  |
|  | Mettre en place une base de données p. | p. |  |
|  | Développer des composants d’accès aux données p. | p. |  |
|  | Développer des composants métier p | p. |  |
|  |  |  |  |
| **Intitulé de l’activité-type n° 3** | | **p.** |  |
|  | Intitulé de l’exemple n° 1  p. | p. |  |
|  | Intitulé de l’exemple n° 2  p. | p. |  |
|  | Intitulé de l’exemple n° 3  p | p. |  |
|  |  |  |  |
| **Intitulé de l’activité-type n° 4** | | **p.** |  |
|  | Intitulé de l’exemple n° 1  p. | p. |  |
|  | Intitulé de l’exemple n° 2  p. | p. |  |
|  | Intitulé de l’exemple n° 3  p | p. |  |
|  |  |  |  |
| **Titres, diplômes, CQP, attestations de formation** *(facultatif)* | | **p.** |  |
| **Déclaration sur l’honneur** | | **p.** |  |
| **Documents illustrant la pratique professionnelle** *(facultatif)* | | **p.** |  |
| **Annexes** *(Si le RC le prévoit)* | | **p.** |  |

**Exemples de pratique**

**professionnelle**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activité-type** | **1** | **Développer la partie front-end d’une application web et web mobile en y intégrant les recommandations de sécurité** | | | |
| ***Exemple n°1*** | | *Maquetter un projet* | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
| J’ai entrepris la création d’un **site e-commerce** spécialisé dans la vente de créations artisanales.   Pour commencer, j’ai étudié les produits que le site allait proposer ainsi que les valeurs qu’ils véhiculaient. L’objectif était de créer une identité visuelle cohérente qui serait en harmonie avec l’univers des créations vendues, tout en m’assurant de la fluidité de la navigation.  J’ai alors créé pour chaque page du site un premier jet des maquettes visuelles. L’interface utilisateur devant être intuitive et fonctionnelle pour toutes et tous, j’ai gardé en tête son accessibilité etai partagé ces maquettes avec des collègues et autres utilisateurs afin de recueillir leurs retours. J’ai ensuite apporté des ajustements en fonction de leurs commentaires.  J’ai ensuite configuré mon environnement de développement en fonction des besoins du projet. Cela incluait l’installation de **Symfony**, le framework PHP, ainsi que des dépendances nécessaires telles que le Maker Bundle ou EasyAdmin. J’ai également mis en place Webpack Encore afin de gérer les assets CSS et Javascript du projet. | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
| Pour créer les maquettes des interfaces utilisateur, j’ai combiné plusieurs approches et outils.   * **Trello** : un outil de gestion et organisation de projet. * **Figma** : un outil de conception et prototypage pour les projets numériques. Figma m’a permis de visualiser et structurer les éléments clés de l’interface, et de créer les maquettes. * **UX design** : pour garantir une expérience utilisateur agréable, fluide et intuitive. * **Règles ergonomiques** : afin de proposer une navigation claire et efficace. * **Accessibilité** : j’ai accordé une certaine attention aux critères de la RGGA pour que mes pages soient faciles d’accès pour tout public. | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
| J’ai travaillé de manière autonome sur ces tâches, mais non pas sans les conseils et retours de mon tuteur, tout en utilisant Trello comme outil central.  Trello m’a permis de lister mes tâches, suivre leur avancement et de les organiser par ordre de priorité. De plus, j’ai pu y enregistrer les retours de mes utilisateurs afin de continuellement améliorer mon travail. | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | Cliquez ici pour taper du texte. | |
| Chantier, atelier, service | | | *Cliquez ici pour taper du texte.* | | |
| Période d’exerciceDu : *08/04/2024*au :*11/04/2024* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
|  | | | | | |
| Certaines maquettes sont disponibles dans l’***Annexe*** ***1***. | | | | | |
|  | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activité-type** | **1** | **Développer la partie front-end d’une application web et web mobile en y intégrant les recommandations de sécurité** | | | |
| ***Exemple n°2*** | | *Créer une interface utilisateur statique* | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
|  | | | | | |
| Pour créer un site e-commerce, j’ai dû en faire toutes les pages en utilisant les connaissances acquises pendant mon année de formation.  Le projet étant fait en Symfony, je me suis servi de Twig pour générer les vues côté serveur.  J’ai principalement structuré mes pagesen utilisant des balises HTML sémantiques et stylisé les éléments de l’interface en utilisant des feuilles de style CSS.  Le framework Bootstrap accélérant le développement de certains éléments (formulaires) oudu responsive web design, je l’ai également intégré au projet.  Voici les points principaux auxquels j’ai payé attention lors de la confection de ces interfaces :   * **Conformité aux Standards W3C** : J’ai fait en sorte que mon code respecte les normes du W3C afin d’en garantir la qualité et compatibilité avec différents navigateurs. * **Accessibilité**: J’ai utilisé des balises sémantiques et des attributs d’accessibilité pour améliorer l’accessibilité du site aux personnes en situation de handicap. J’ai vérifié la compatibilité du site avec des technologies d’assistance (lecteurs d’écran, etc.) et ai adapté la navigation clavier. * **Responsivité** : J’ai adapté le site pour qu’il fonctionne correctement sur différents types d’appareils afin de garantir une expérience utilisateur optimale peu importe l’écran utilisé. | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
| Pour créer ces interfaces utilisateur statiques et responsives, j’ai donc utilisé :   * **Npm** (Node package manager) m’a permis d’installer les dépendances nécessaires à mon projet. Cela inclut des bibliothèques et frameworks tels que Bootstrap, ainsi que des outils de build et de développement comme Webpack. * **Bundle Webpack Encore** : Grâce à ce bundle, j’ai pu gérer la compilation des fichiers SCSS (Sass) en CSS. Webpack Encore m'a également permis de minifier et d'optimiser ces fichiers pour améliorer les performances de chargement des pages et offrir une expérience utilisateur fluide et rapide. * **Twig étant le moteur de templates de Symfony, il fut crucial pour structurer et générer mes vues HTML, et ses logiques conditionnelles ont simplifié le processus de création d’interfaces évolutives.** * **Bootstrap grâce à ses plugins a facilité l’intégration de certains éléments au site (fenêtre modale), et ses classes prédéfinies ont permis d’accélérer la création d’interfaces responsives et adaptées pour plusieurs types d’écrans.** | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
| J’ai travaillé sur ces fonctionnalités en autonomie mais non pas sans les précieux conseils de mon responsable de stage, qui a su me guider lorsque j’avais des questions ou difficultés. | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | *Cliquez ici pour taper du texte.* | |
| Chantier, atelier, service | | | Cliquez ici pour taper du texte. | | |
| Période d’exerciceDu : *22/04/2024*au :*30/04/2024* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
|  | | | | | |
| En résumé, c’est grâce à Npm, Webpack Encore, Twig, Bootstrap et HTML/CSS que j’ai pu créer des interfaces utilisateurs fluides et agréables pour les utilisateurs sur de nombreux types d’écrans.  Voir ***Annexe 2***. | | | | | |
|  | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activité-type** | **1** | **Développer la partie front-end d’une application web et web mobile en y intégrant les recommandations de sécurité** | | | |
| ***Exemple n°3*** | | *Développer une interface utilisateur dynamique* | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
|  | | | | | |
| Dans le cadre de la création de mon site web avec Symfony, j’ai mis en place plusieurs éléments dynamiques afin d’améliorer l’expérience utilisateur ou de la rendre plus interactive et agréable.  **Navigation dynamique :**L’un des éléments clés de l’interface utilisateur.   * Afin de la rendre plus intuitive et claire, les éléments changent de couleur lorsqu’ils sont survolés avec la souris, ou lorsqu’ils sont cliqués. * La navigation comporte un sous-menu déroulant permettant aux utilisateurs de rapidement accéder aux différentes catégories de produits proposés par le site.   **Fenêtre modale pour le détail des produits :** Pour donner un accès rapide aux détails de mes produits à partir de la page d’accueil de façon fluide et non intrusive, j’ai intégré une fenêtre modale sur les grands écrans. Elle s’ouvre et se ferme grâce à un simple bouton et évite la redirection des utilisateurs.  **Fonction asynchrone de tri des produits :** Pour améliorer l’expérience de navigation des utilisateurs, j’ai développé une fonction asynchrone permettant de trier les produits du site par catégorie. Cette fonctionnalité a été réalisée en utilisant notamment Javascript et des requêtes AJAX. Les utilisateurs peuvent donc trier les produits rapidement et sans recharger la page.  **Chargement progressif des produits** : Afin de gérer l’affichage de nombreux produits et sans alourdir le temps de chargement initial de la page, j’ai mis en place une fonction de chargement progressif :   * Les produits sont chargés petit à petit à mesure que l’utilisateur fait défiler la page, rendant l’expérience de chacun plus fluide, personnalisée et adaptée à ses besoins. * Optimisation des ressources :Ce chargement étant progressif et optionnel, il répartit la charge côté client et serveur sur une période plus longue, évitant des pics de charge.   **Affichage du nombre d’articles dans le panier :** Dans un souhait d’améliorer l’expérience utilisateur, j’ai implémenté une fonctionnalité permettant d’afficher dynamiquement le nombre de produits présents dans le panier, à travers toutes les pages. | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
|  | | | | | |
| * **JQuery et AJAX :** la bibliothèque Javascript m’a majoritairement servi à manipuler le DOM et gérer des événements côté client. J’ai pu notamment sélectionner des éléments, appliquer des animations, modifier des contenus en réponse à des interactions utilisateur ou bien charger des contenus sans rechargement de page. * **Bootstrap** : c’est grâce au framework Bootstrap et ses modales que j’ai pu faciliter l’accès aux détails des produits depuis la page d’accueil. * **Javascript**: Javascript a servi à la création d’éléments du DOM (les produits à charger), gérer l’état de l’application grâce à des variables, gérer des interactions avec les utilisateurs et développer les fonctions nécessaires. * **KnpPaginator :** ce bundle installé via Composer m’a permis d’afficher un nombre limité de commentaires par page, améliorant la fluidité de navigation des utilisateurs du site. * **Twig :** la plupart des fonctionnalités citées plus haut ont été possibles ou facilitées grâce à Twig. J’ai en effet utilisé Twig afin de générer dynamiquement le contenu des pages web en fonction des données reçues du back-end. Le moteur de templates propose des balises capables d’afficher les variables de vues ainsi que des boucles permettant d’itérer sur mes données (produits, catégories, commentaires). Les filtres Twig ont également servi à formater certaines données comme les dates ou les descriptions des produits. | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
|  | | | | | |
| J’ai travaillé sur ces fonctionnalités en autonomie mais non pas sans les précieux conseils de mon responsable de stage, qui a su me guider lorsque j’avais des questions ou difficultés. | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | *Cliquez ici pour taper du texte.* | |
| Chantier, atelier, service | | | *Cliquez ici pour taper du texte.* | | |
| Période d’exerciceDu : *01/05/2024*au :*21/05/2024* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
|  | | | | | |
| En résumé, c’est en combinant Javascript, jQuery, Ajax, Boostrap et Twig que j’ai pu créer des interfaces utilisateurs dynamiques, réactives et agréables pour les utilisateurs du site.  Voir ***Annexe 3***. | | | | | |
|  | | | | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Activité-type** | |  | **2** | **Développer la partie back-end d’une application web ou web mobile sécurisée** | | | |
|  | ***Exemple n°1*** | | | Mettre en place une base de données | | | |
|  |  | | | | | | |
|  |  | | | | | | |
|  | **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | | |
|  |  | | | | | | |
|  | Pour mettre en place la base de données de mon application, j’ai suivi un processus structuré afin d’assurer une gestion efficace des données ainsi que la sécurité des données des utilisateurs.   * **MCD :** J’ai d’abord conçu un modèle conceptuel de données détaillé qui a servi à m’organiser et identifier les entités principales nécessaires au site ainsi que leurs relations. * **Relations et contraintes :** Une fois la base de données créée, j’ai pu implémenter les contraintes et relations aux tables grâce à des clés étrangères qui ont garanti l’intégrité ainsi que la cohérence des données. * **Hashing (hachage) :** Afin de renforcer la sécurité du site, j’ai mis en place un système de hachage pour les mots de passe des utilisateurs. Il a servi à ce que les mots de passe des utilisateurs restent protégés et non lisibles. * **Rôles :** En parallèle, j’ai configuré les différents rôles d’utilisateurs avec des permissions spécifiques afin de contrôler l’accès aux données sensibles (Backoffice, page de profil). | | | | | | |
|  |  | | | | | | |
|  |  | | | | | |
|  |  | | | | | | |
|  | **2. Précisez les moyens utilisés :** | | | | | | |
|  | * **PHPMyAdmin**: le mode concepteur de PHPMyAdmin m’a offert la possibilité de visualiser et planifier la structure de ma base de données de manière simple et efficace. * **Symfony** : tout en suivant les recommandations de sécurité de l’ANSSI, c’est grâce au framework PHP que j’ai pu hasher simplement les mots de passe des utilisateurs en utilisant l’algorithme de hachage configuré par défaut. Symfony grâce à Doctrine m’a aussi permis de gérer la persistance des objets en base de données de manière efficace et simplifiée. * **MySQL**: j’ai opté pour MySQL comme système de gestion de base de données afin de stocker, organiser et administrer les données du projet. * **Contraintes et accès** : à l’aide des migrations de Symfony, j’ai pu configurer les contraintes de base de données nécessaires au projet. J’ai également pu profiter du système d’authentification de Symfony et configurer le rôles (admin, user) afin de gérer l’accès aux données. | | | | | | |
|  |  | | | | | | |
|  |  | | | | | | |
|  |  | | | | | | |
|  |  | | | | | | |
|  | **3. Avec qui avez-vous travaillé ?** | | | | | | |
|  |  | | | | | | |
|  | J’ai confectionné la base de donnée en autonomie mais non pas sans les précieux conseils de mon responsable de stage, qui a su me guider lorsque j’avais des questions. | | | | | | |
|  |  | | | | | | |
|  |  | | | | | |
|  |  | | | | | | |
|  | **4. Contexte** | | | | | | |
|  |  | | | | |  | |
|  | Nom de l’entreprise, organisme ou association | | | | | *Cliquez ici pour taper du texte.* | |
|  | Chantier, atelier, service | | | | *Cliquez ici pour taper du texte.* | | |
|  | Période d’exerciceDu : *15/04/2024*au :*18/04/2024* | | | | | | |
|  |  | | | | | |
|  |  | | | | | |
|  | **5. Informations complémentaires** *(facultatif)* | | | | | |
|  |  | | | | | | |
|  | En résumé, la combinaison de Symfony, Doctrine et du MCD fait à l’aide de PHPMyAdmin, j’ai pu créer une base de données sécurisée et conforme aux besoins de mon site e-commerce.  Voir ***Annexe 4***. | | | | | | |
|  |  | | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activité-type** | **3** | **Développer la partie back-end d’une application web ou web mobile sécurisée** | | | |
| ***Exemple n°2*** | | Développer des composants d’accès aux données SQL | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
|  | | | | | |
| Dans le cadre de la création d’un site e-commerce, j’ai dû développer un certain nombre de composants d’accès aux données SQL. Ces composants sont essentiels pour interagir efficacement avec la base de données et assurer la gestion fluide des données de l’application.  Voici une liste des différents composants d’accès aux données SQL que j’ai programmés :   * Certains concernaient les produits vendus sur le site.   + Récupération et affichage dynamique des produits du site   + Filtrage et affichage des produits par catégories   + Tri et affichage des produits en fonction de leur prix   + Accès aux détails spécifiques d’un produit unique * D’autres concernaient la gestion des transactions et du panier d’achats. * Ajouter un ou des produits au panier * Supprimer un ou des produits du panier * Vider le panier * Système de paiement sécurisé * Envoi de mail et de facture aux utilisateurs une fois qu’un achat est réussi * Et enfin, grâce à un Backoffice EasyAdmin, j’ai pu utiliser des composants CRUD (créer, lire, modifier et supprimer) simplifiant la gestion des entités de l’application. | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
|  | | | | | |
| Afin de développer ces composants d’accès aux données, j’ai principalement utilisé les outils suivants :   * **Doctrine :** l’ORM intégré à Symfony a servi à gérer l’accès aux données de ma base SQL. J’ai ainsi pu facilement définir mes entités ainsi que leurs relations sans avoir à écrire de requêtes SQL directes. * **Query Builder :** le query builder de **Doctrine** m’a servi à construire et exécuter des requêtes SQL dynamiques de façon sécurisée. J’ai ainsi pu manipuler les données de manière flexible et efficace, en ajoutant par exemple des conditions ou clauses (produits, input search..). * **EntityManager :** j’ai utilisé cette classe dans mon projet pour simplifier la gestion des entités PHP et leur interaction avec la base de données. L’outil m’a permis de créer, charger ou modifier des entités, et a optimisé les performances en minimisant le nombre de requêtes SQL nécessaires au bon fonctionnement du projet. * Le **ServiceEntityRepository :** cette classe Symfony héritant des fonctionnalités de base de l’EntityRepository, je l’ai utilisée afin d’avoir accès aux méthodes prédéfinies de ce dernier. * **EasyAdmin :** pour le backoffice du site, j’ai utilisé EasyAdmin, un bundle Symfony puissant pour créer des interfaces d’administration. J’ai ainsi pu rapidement générer des CRUD pour gérer les entités de l’application telles que les produits, les catégories de produits et les commandes. * **API Stripe :** pour la gestion des transactions financières, j’ai intégré l’API Stripe au projet. Ce système a nécessité un accès aux données pertinentes de la base de données pour traiter les transactions financières de manière sécurisée. * **Dompdf :** la bibliothèque PHP permettant de convertir du HTML en PDF, j’ai pu créer pour les utilisateurs des factures professionnelles et détaillées en format PDF à partir de templates HTML qui récupéraient les données nécessaires de la base de données. | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
|  | | | | | |
| J’ai développé ces composants en autonomie mais non pas sans les précieux conseils de mon responsable de stage, qui a su me guider lorsque j’avais des questions. | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | *Cliquez ici pour taper du texte.* | |
| Chantier, atelier, service | | | *Cliquez ici pour taper du texte.* | | |
| Période d’exerciceDu : *22/05/2024*au :*20/06/2024* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
| Parmi les commandes les plus importantes ou celles que j’ai le plus utilisées se trouvent les commandes suivantes :   * Php bin/console **doctrine:database:create** m’a permis de créer la base de données du projet. * Php bin/console **make:entity** a servi à créer les entités du projet. * Php bin/console **make:migrations** / **doctrine:migrations:migrate** pour mettre à jour la base de données. * Php bin/console **make:fixtures** pour remplir la base de données de données fictives lors du développement et des tests du site.   Voir ***Annexes 5 et 6.*** | | | | | |
|  | | | | | |
|  | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activité-type** | **3** | **Développer la partie back-end d’une application web ou web mobile sécurisée** | | | |
| ***Exemple n°3*** | | Développer des composants métier côté serveur | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
|  | | | | | |
| Dans le cadre de mon projet e-commerce, j’ai développé plusieurs composants métier essentiels pour assurer une gestion fluide et sécurisée des interactions utilisateur. Voici les principaux que j’ai créés :   * **Inscription avec contrôle des champs** : gestion de l’inscription des utilisateurs avec validation des données saisies pour garantir leur conformité. * **Validation** **du compte par email** * **Connexion** : gestion de l’authentification des utilisateurs, assurant la sécurité des comptes. * **Modification du profil** : permet aux utilisateurs de mettre à jour leurs informations personnelles. * **Mot de passe oublié** : fonctionnalité permettant aux utilisateurs de réinitialiser leur mot de passe via un lien envoyé par mail. * **Supprimer le compte** : gestion de la suppression des comptes utilisateurs et de leurs données personnelles conformément aux réglementations de confidentialité. * **Ajout au panier** : permet aux utilisateurs d’ajouter des produits à leur panier d’achat. * **Supprimer du panier** : permet de supprimer un article du panier. * **Vider le panier** : permet aux utilisateurs de supprimer tous les produits de leur panier. * **Système de paiement** : intégration et gestion des achats via Stripe pour des paiements sécurisés. * **Envoi de mail et facture PDF aux clients :** envoi automatique de confirmation de commande et de facture après un achat réussi. | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
|  | | | | | |
| Pour développer des composants métier dans le cadre du projet, j’ai utilisé une combinaison de technologies et de pratiques assurant leur fonctionnement efficace et sécurisé :   * **Doctrine ORM :** a servi pour la gestion des entités et des interactions avec la BDD, assurant une persistance des données efficace et sécurisé. * **Validation Symfony :** j’ai utilisé les contraintes de validation Symfony (NotBlanc, length, email, etc.) pour garantir l’intégrité des données saisies par les utilisateurs et prévenir les erreurs de saisie. * **Twig :** le moteur de template a servi à l’intégration de logique métier dans les templates. * **Security Component :** j’ai configuré le security component pour gérer l’authentification, l’autorisation et la gestion de rôles utilisateurs, assurant ainsi la sécurité des données sensibles et des fonctionnalités. * **Gestion des formulaires :** construction de formulaires complexes avec **Symfony Form Component** pour la saisie et validation des données utilisateur, incluant la gestion des fichiers et la protection CSRF. * **Service Symfony :** création de service pour regrouper la logique métier et faciliter sa réutilisation dans toute l’application. J’ai notamment créé un ImageService qui a joué un rôle essentiel dans la gestion de l’upload et la modification des images de profil des utilisateurs. * **APIs externes :** j’ai intégré l’API Stripe au projet afin d’avoir un système de paiement sécurisé. * **Dompdf :** La bibliothèque PHP m’a permis de créer les pdf de factures pour les clients à partir de code HTML/CSS. * **Tests unitaires :** écriture de tests unitaires et fonctionnels pour valider le bon fonctionnement des composants métier et assurer la qualité du code. * **Style défensif :** j’ai adopté de bonnes pratiques de programmation telles que la gestion des exceptions avec try-catch, la validation stricte des entrées utilisateur et la sécurisation des interactions avec la base de données pour prévenir les erreurs et attaque potentielles. * **Fichier .env.local** : ce fichier m’a permis de gérer les variables d’environnement spécifiques à mon projet sans risquer de partager des informations sensibles. Je l’ai utilisé par exemple pour définir une clé Stripe. | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
|  | | | | | |
| J’ai développé ces composants en autonomie mais non pas sans les précieux conseils de mon responsable de stage, qui a su me guider lorsque j’avais des questions. | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | *Cliquez ici pour taper du texte.* | |
| Chantier, atelier, service | | | *Cliquez ici pour taper du texte.* | | |
| Période d’exerciceDu : *22/05/2024*au :*20/06/2024* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
| En résumé, j’ai créé des des composants métiers liés aux utilisateurs et à la possibilité de faire des achats sur le site en utilisant Symfony, afin de garantir une expérience utilisateur sécurisée et cohérente. | | | | | |
| Voir ***Annexes 7 et 8.*** | | | | | |
|  | | | | | |

|  |
| --- |
| **Déclaration sur l’honneur** |
|  |
|  |

Roxanne MARAIS

Je soussigné(e) [prénom et nom] ,

déclare sur l’honneur que les renseignements fournis dans ce dossier sont exacts et que je suis l’auteur(e) des réalisations jointes.

Cannes

10/07/2024

Fait à le

pour faire valoir ce que de droit.

Signature :

|  |
| --- |
| **Documents illustrant la pratique professionnelle** |
|  |
| *(facultatif)* |
| **Intitulé** |
| **Annexe 1 -** Maquettes d’interface utilisateur |
| **Annexe 2 –** Interfaces statiques |
| **Annexe 3 –** Exemple dynamiques |
| **Annexe 4 –** MCD |
| **Annexe 5 –** Backoffice EasyAdmin |
| **Annexe 6 –** ORM Doctrine |
| **Annexe 7 –** Contraintes et validations des formulaires |
| **Annexe 8 –** API Stripe |
|  |
|  |

|  |
| --- |
| **Annexes** |
|  |
| *(Si le RC le prévoit)* |

**Annexe 1 : Maquettes**

**Une image contenant texte, capture d’écran, logiciel, conception

Description générée automatiquement**

**Annexe 2 : Page statique (contact)**

**Une image contenant texte, capture d’écran

Description générée automatiquement**

|  |
| --- |
| **Annexes** |
|  |
| *(Si le RC le prévoit)* |

**Annexe 3 – Exemples dynamiques**

Navigation  **Une image contenant texte, Logiciel multimédia, capture d’écran

Description générée automatiquement**

Affichage des commentaires

**Une image contenant texte, capture d’écran, conception

Description générée automatiquement**

|  |
| --- |
| **Annexes** |
|  |
| *(Si le RC le prévoit)* |

**Annexe 3 (suite) – Exemples dynamiques**

Page produits, tri par catégorie

**Une image contenant texte, capture d’écran, Site web, Page web

Description générée automatiquement**

Chargement progressif

**Une image contenant texte, capture d’écran, conception

Description générée automatiquement**

|  |
| --- |
| **Annexes** |
|  |
| *(Si le RC le prévoit)* |

**Annexe 4 – MCD**

Une image contenant texte, capture d’écran, nombre, logiciel

Description générée automatiquement

|  |
| --- |
| **Annexes** |
|  |
| *(Si le RC le prévoit)* |

**Annexe 5 -** Backoffice EasyAdmin

Une image contenant capture d’écran, texte, logiciel, Logiciel multimédia

Description générée automatiquement

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

|  |
| --- |
| **Annexes** |
|  |
| *(Si le RC le prévoit)* |

**Annexe 6**

ORM Doctrine et Query Builder

Une image contenant texte, capture d’écran

Description générée automatiquement

|  |
| --- |
| **Annexes** |
|  |
| *(Si le RC le prévoit)* |

**Annexe 7 - Contraintes et validations des formulaires avec Symfony**

**Une image contenant texte, capture d’écran, logiciel

Description générée automatiquement**

|  |
| --- |
| **Annexes** |
|  |
| *(Si le RC le prévoit)* |

**Annexe 8 – API Stripe**

**Une image contenant texte, capture d’écran, logiciel, Police

Description générée automatiquement**

**Une image contenant texte, capture d’écran, Police, ligne

Description générée automatiquement**