|  |  |  |
| --- | --- | --- |
| *Nom de naissance* | ► | **Corin** |
| *Nom d’usage* | ► | **Corin** |
| *Prénom* | ► | **Gaëtan** |
| *Adresse* | ► | **23 rue Auguste Renoir 31200 Toulouse** |
|  |  |

| **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 ou web mobile en intégrant les recommandations de sécurité** | | **p.** | **8-17** |
|  | |  |  |
| **Compétences liées à l’activité :**   * 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 | |  |  |
|  | ► Intitulé de l’exemple n° 1  **Intégration d’une maquette Photoshop** | p. | 8-9 |
|  | ► Intitulé de l’exemple n° 2  **Projet personnel de front-end dynamique** | p. | 12-13 |
|  | ► Intitulé de l’exemple n° 3  **Front-end d’un site d’événement en Django** | p. | 16-17 |
|  |  |  |  |
| **Développer la partie back-end d’une application web ou web mobile en intégrant les recommandations de sécurité** | | **p.** | **20-29** |
|  | |  |  |
| **Compétences liées à l’activité :**   * 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 œuvre des composants dans une application de gestion de contenu ou e-commerce | |  |  |
|  | ► Intitulé de l’exemple n° 1  **Création d’une Api Rest en NodeJs** | p. | 20-21 |
|  | ► Intitulé de l’exemple n° 2  **Projet personnel d’Api Rest en Python Flask** | p. | 24-25 |
|  | ► Intitulé de l’exemple n° 3  **Back-end d’un site de paris sportif en Php et Sql** | p. | 28-29 |
|  |  |  |  |
|  |  |  |  |
| **Titres, diplômes, CQP, attestations de formation** *(facultatif)* | | p. | 31 |
| **Déclaration sur l’honneur** | | p. | 32 |
| **Documents illustrant la pratique professionnelle** *(facultatif)* | | p. | 33-46 |
| **Annexes** *(Si le RC le prévoit)* | | p. |  |

****

**Exemples de pratique**

**professionnelle**

****

|  |  | | |  | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Activité-type** | **1** | | | **Développer la partie front-end d’une application web ou web mobile en intégrant les recommandations de sécurité** | | | | |
| ***Exemple n°1*** ► | | | | ***Intégration d’une maquette Photoshop*** | | | | |
|  | | | |  | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | | | | |
| **Sous la supervision de mon responsable de stage chez Activus Group, j’ai intégré une maquette en utilisant les technologies Html, Css et la librairie Bootstrap pour le responsive. La maquette étant réalisée par un Webdesigner, la première mission a été de récupérer depuis un fichier Photoshop l’intégralité des éléments de la maquette nécessaire à son développement. (images, logos et polices d’écriture)**  **Ayant uniquement la maquette en format grand Desktop, J’ai ensuite créé un Zoning pour chaque taille de responsive en respectant les points d’arrêts natifs de Bootstrap (format moyen Desktop, petit Desktop, tablette et téléphone). J’ai pour cela découper la maquette en différentes sections afin de faciliter le zoning et la future intégration.**  **J’ai pu par la suite commencer le développement de l’interface graphique. Mon responsable m’a été d’une grande aide afin de mieux maîtriser les ‘Médias-queries’ qui sont les points d’arrêts propre au Css, ainsi que pour comprendre le fonctionnement des points d’arrêts de Bootstrap. Cela m’a permis de partir sur de bonnes bases durant tout le long de l’intégration.**  **Afin de respecter au maximum la maquette, l’importation de la librairie Font-Awesome a été nécessaire afin d’utiliser les mêmes éléments graphiques qui sont propres à cette librairie. Différentes polices d’écritures de Google-Fonts ont aussi été importées.**  **Les images en arrière-plan ont été la difficulté la plus importante rencontrée sur cette intégration. En effet, ses images ne respectent pas les points d’arrêts de la page qui donne la responsive au texte. Ces images s’étendent sur l’intégralité de la largeur de l’écran de l’utilisateur.**  **La seconde plus grande difficulté a été le mini carrousel d’images et de textes. Il a été créé en utilisant la propriété Css “overflow-x: auto”. Cette propriété permet de créer une barre de défilement sur la balise html ciblé et à l’horizontal. Cela à permis à l’utilisateur de naviguer de manière horizontal sur cette section grâce à cette barre de défilement tout en dévoilant les différents éléments les uns après les autres. Ce carrousel est surtout utile sur les petits responsives.** | | | | | | | | |
|  | | | | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | | | | |
| **Afin de réaliser ce projet, l’utilisation de Photoshop a été nécessaire afin d’ouvrir et travailler sur la maquette qui est en fichier .psd.**  **L’utilisation de Google-Fonts et de la librairie FontAwesome m’ont permis de respecter au mieux la charte graphique de la maquette.**  **J’ai utilisé l’environnement de développement Vscode ainsi que Git et Github qui me permettent de stocker, gérer le code et contrôler les modifications qui lui sont apportées.**  **L’ordinateur permettant de réaliser ce projet a été celui fourni par Activus-Group durant mon stage de fin d’année.** | | | | | | | | |
|  | | | | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | | | | |
| **J’ai travaillé seul sur cette intégration avec l’aide de mon responsable de stage.** | | | | | | | | |
|  | | | | | | | | |
| **4. Contexte** | | | | | | | | |
|  | | | | | |  | | |
| **Nom de l’entreprise, organisme ou association** ► | | | | | | ***Activus-Group*** | | |
|  | |  | | | | | | |
| **Chantier, atelier, service** | | ► | **Service développement Web de Activus-Group** | | | | | |
| **Période d’exercice** | | ► | **Du** | | **23/05/2022** | | **au** | **29/07/2022** |
|  | | | | | | | | |
|  | | | | | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | | | | | |
| Cliquez ici pour taper du texte. | | | | | | | | |

****

|  |  | | |  | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Activité-type** | **1** | | | **Développer la partie front-end d’une application web ou web mobile en intégrant les recommandations de sécurité** | | | | |
| ***Exemple n°2*** ► | | | | ***Projet personnel de front-end dynamique*** | | | | |
|  | | | |  | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | | | | |
| **Durant mon temps personnel, j ’ai réalisé un projet à présenter aux futurs employeurs afin de pouvoir me vendre lors de ma recherche d’alternance.**  **Ce projet consiste à présenter une liste de produits contenant une image, un titre, un prix et une petite description. Lors du clique sur un produit, nous avons accès à sa page spécifique.**  **Cette application est réalisée avec un front-end dynamique, ainsi qu’un back-end Api python, que je vous présente en exemple 2 de la partie back-end.**  **Je vous présente ici la partie front-end du projet, qui est développé en Html, Css, et Javascript dynamique.**  **La partie Html de ce projet est un template de base contenant les importations de mes fichiers Css et Javascript.**  **L’entête de cette application est statique. développé directement en balise Html et Css.**  **Pour le corps de l’html, il ne contient que des balises <div> vides qui seront remplis en javascript lors des différents appels à l’API.**  **Le Javascript représente donc le cœur de cette application. C’est lui qui va créer la plupart des éléments visibles sur le front-end.**  **Pour créer ses éléments, le javascript va avoir besoin d’informations. Pour cela, j’utilise la méthode asynchrone “async()” et “await”. C’est une méthode qui consiste à réaliser une action demandant un délai, attendre la réponse, puis continuer la suite du script une fois la réponse reçue.**  **Dans mon application, cette méthode est utilisée pour réaliser une requête Http vers mon Api servant à fournir les informations des différents produits à afficher.**  **Le script attend le résultat, puis une fois reçu en format json, il le transforme en format compréhensible par javascript en utilisant la fonction “.json()”.**  **Le javascript peut ensuite utiliser ses informations afin de créer les éléments représentant les différents produits.**  **Chacun de ses éléments aura un id unique, ainsi qu’un “addEventListener” servant à écouter le clique.**  **Si l’utilisateur clique sur un de ses éléments, il déclenche une fonction qui réalise une nouvelle requête vers l’api afin d’afficher en détail le produit sur lequel l’utilisateur a cliqué.** | | | | | | | | |
|  | | | | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | | | | |
| **Pour réaliser cette application, j’ai utilisé l’Html, le Css, le Javascript, ainsi que mon Api Python.**  **Je me suis aidé de Google ainsi que différents forums sur internet.** | | | | | | | | |
|  | | | | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | | | | |
| **J’ ai travaillé seul sur ce projet.** | | | | | | | | |
|  | | | | | | | | |
| **4. Contexte** | | | | | | | | |
|  | | | | | |  | | |
| **Nom de l’entreprise, organisme ou association** ► | | | | | | **Aucun** | | |
|  | |  | | | | | | |
| **Chantier, atelier, service** | | ► | **Projet personnel** | | | | | |
| **Période d’exercice** | | ► | **Du** | | **04/07/2022** | | **au** | **11/07/2022** |
|  | | | | | | | | |
|  | | | | | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | | | | | |
| Cliquez ici pour taper du texte. | | | | | | | | |

****

|  |  | | |  | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Activité-type** | **1** | | | **Développer la partie front-end d’une application web ou web mobile en intégrant les recommandations de sécurité** | | | | |
| ***Exemple n°3*** ► | | | | ***front-end d’un site d'événement en Django*** | | | | |
|  | | | |  | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | | | | |
| **Lors d’un projet en groupe durant ma formation à l’Adrar, nous avons dû réaliser un site de présentation d'événements en utilisant le Framework Django.**  **Nous avons donc débuté ce projet en réalisant un cahier des charges.**  **Nous avons ensuite réalisé la conception de la base de données en réalisant un modèle conceptuel de données ainsi qu’un modèle logique de données, puis avons réalisé une arborescence des différents templates du projet afin de correctement l’architecturer.**  **Par la suite, nous nous sommes réparties les tâches à effectuer. Ma responsabilité dans ce projet a été de réaliser l’ensemble de la partie front-end du projet pendant que mon collègue était en charge de la partie back-end.**  **Le projet ayant une durée de développement très limitée, je me suis donc dirigé vers la librairie Bootstrap afin de répondre aux exigences de temps demandées.**  **Je me suis servie des différents éléments fournis par le back-end afin de réaliser un carrousel d’image sur la page d’accueil. La difficulté sur cette conception a été que Django fonctionne sur un principe d’itération d’instances permettant de récupérer les informations des événements sur le front-end. Bootstrap, lui, utilise une classe ‘active’ sur son carrousel afin de savoir lequel des événements présenter. Il a donc fallu que j’ ajoute cette classe ‘active’ exclusivement sur l’itération de la première instance, mais qu’elle n’apparaisse pas sur toutes les suivantes.**  **Par la suite, j’ ai réalisé le formulaire de la page d’inscription, le formulaire de connexion, ainsi que la page d’erreur.**  **J’ ai ensuite réalisé la page présentant l’ensemble des événements affichés sur le site, ainsi que la page présentant en détail chaque événement. Cette page n'apparaît que lorsque l’on clique sur un événement en particulier.**  **Afin de rendre le site responsive, je me suis servi des différents points d’arrêts de Bootstrap sur toutes les pages.** | | | | | | | | |
|  | | | | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | | | | |
| **Les moyens utilisés sont le Html, le Css, le framework Django ainsi que Bootstrap.**  **J’ ai aussi utilisé le logiciel Looping.exe afin de réaliser le modèle conceptuel de données et le modèle logique de données, et le logiciel Gloomaps afin de créer un schéma de l’arborescence du site.** | | | | | | | | |
|  | | | | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | | | | |
| **J’ ai travaillé en équipe avec un collègue de formation.** | | | | | | | | |
|  | | | | | | | | |
| **4. Contexte** | | | | | | | | |
|  | | | | | |  | | |
| **Nom de l’entreprise, organisme ou association** ► | | | | | | **Aucun** | | |
|  | |  | | | | | | |
| **Chantier, atelier, service** | | ► | **Projet de groupe durant la formation** | | | | | |
| **Période d’exercice** | | ► | **Du** | | **25/03/2022** | | **au** | **28/03/2022** |
|  | | | | | | | | |
|  | | | | | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | | | | | |
| Cliquez ici pour taper du texte. | | | | | | | | |

****

|  |  | | |  | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Activité-type** | **2** | | | **Développer la partie back-end d’une application web ou web mobile en intégrant les recommandations de sécurité** | | | | |
| ***Exemple n°1*** ► | | | | ***Création d’une API Rest en NodeJs*** | | | | |
|  | | | |  | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | | | | |
| **Le travail a consisté à réaliser le Backend d’une application qui permet à des utilisateurs de créer un compte et se connecter. Ils pourront accéder à une page où seront affichés les objets à vendre des autres utilisateurs. Il sera possible pour l’utilisateur de mettre ses propres objets à vendre, modifier uniquement ses objets et les supprimer.**  **Pour cela, j‘ ai réalisé une API Rest en méthode CRUD. Cette méthode consiste à écouter les requêtes HTTP qui peuvent être en méthode Get, Post, Update et Delete. Suivant ses requêtes, l’application va réaliser les tâches les plus importantes pour une API qui sont la création(C), la lecture(R), la modification(U) et la suppression(D).**  **Pour réaliser ce Backend, j’ai utilisé les technologies Javascript, Nodejs et MongoDb. Un front-end adapté à mes requêtes APIs m’a été fourni afin de s’adapter au mieux à mon travail.**  **Afin de réaliser ce projet, j’ ai utilisé “Npm” qui est le gestionnaire de paquets officiel de NodeJs.**  **Grâce à lui, j’ai pu installer le framework Express.js de Nodejs qui permet de créer des points de terminaisons pour communiquer avec le front-end en transmettant des fichiers Json.**  **J’ai connecté l’application avec la base de données MongoDb en utilisant l’extension Mongoose ainsi que mongoose-unique-validator qui fournit des messages explicites lors de problèmes avec la base de données.**  **J’ai par la suite créé un modèle d’utilisateur ainsi qu’un modèle d’objet à vendre afin de pouvoir transformer des instances de ses modèles en enregistrements dans la base de données.**  **J’ai ensuite réalisé plusieurs Endpoints qui se servent de ces modèles pour exécuter des actions de Création(C), lecture(R), modification(U) et suppression(D). Ces points de terminaisons sont accessibles par un routeur qui se sert de l’url et de la méthode HTTP pour connaître l’action à exécuter.**  **Sur ces points de terminaisons, J'ai dû utiliser l‘asynchrone de Javascript “.then” et “.catch” qui me permet d’attendre la réponse de la base de donnée MongoDb avant de continuer mon script et de renvoyer le résultat sur le front-end.**  **Les mots de passe des utilisateurs sont sécurisés en base de données avec l’extension bcrypt qui hash les mots de passe. Elle permet aussi de vérifier la connexion de l‘utilisateur en comparant le mot de passe fourni lors de la connexion ainsi que son hash dédié sur mongoDB, puis retourne simplement true ou false afin de valider la connexion de manière sécurisée.**  **Il est possible pour l’utilisateur de rester connecté grâce à un Token fourni lors de sa connexion. Pour cela, l’extension jsonwebtoken est utilisée. L’Id de l‘utilisateur est crypté à l’intérieur de ce token, et c’est le front-end qui enverra ce token à chaque requête HTTP sur l’emplacement Header Authorization. Il est donc possible de savoir quel utilisateur est responsable de chaque requête HTTP en utilisant un middleware qui lit ce token.** | | | | | | | | |
|  | | | | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | | | | |
| **Pour réaliser ce projet, j’ai utilisé les technologies Javascript, nodeJs et de nombreuses dépendances de celui-ci. J’ai aussi utilisé le format Json afin de faire communiquer mon API. Le service Postman a été utilisé afin de tester les différents Endpoints et sa connexion à la base de données MongoDb.**  **J’ai utilisé l’environnement de développement Vscode ainsi que Git et Github qui me permettent de stocker, gérer le code et contrôler les modifications qui lui sont apportées.**  **L’ordinateur permettant de réaliser ce projet est celui fourni par Activus-Group durant mon stage de fin d’année.** | | | | | | | | |
|  | | | | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | | | | |
| **J’ai travaillé seul durant la réalisation de ce projet.**  **J‘ ai pu demander conseil auprès de mon responsable de stage ainsi qu’à différents collègues lorsque je rencontrais des difficultés.** | | | | | | | | |
|  | | | | | | | | |
| **4. Contexte** | | | | | | | | |
|  | | | | | |  | | |
| **Nom de l’entreprise, organisme ou association** ► | | | | | | ***Activus-Group*** | | |
|  | |  | | | | | | |
| **Chantier, atelier, service** | | ► | **Service développement Web de Activus-Group** | | | | | |
| **Période d’exercice** | | ► | **Du** | | **23/05/2022** | | **au** | **29/07/2022** |
|  | | | | | | | | |
|  | | | | | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | | | | | |
| Cliquez ici pour taper du texte. | | | | | | | | |

****

|  |  | | |  | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Activité-type** | **2** | | | **Développer la partie back-end d’une application web ou web mobile en intégrant les recommandations de sécurité** | | | | |
| ***Exemple n°2*** ► | | | | ***Projet personnel d’Api Rest en Python Flask*** | | | | |
|  | | | |  | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | | | | |
| **Durant mon temps personnel, j ’ai réalisé un projet à présenter aux futurs employeurs afin de pouvoir me vendre lors de ma recherche d’alternance.**  **Ce projet consiste à présenter une liste de produits contenant une image, un titre, un prix et une petite description. Lors du clique sur un produit, nous avons accès à sa page spécifique.**  **Cette application est développée avec un back-end python en API , ainsi qu’un front-end dynamique, que je vous présente en exemple 2 de la partie front-end.**  **Je vous présente ici la partie back-end du projet, qui est développé en Python utilisant le framework Flask, ainsi que la dépendance MongoEngine permettant de se connecter à la base de données MongoDb.**  **Désirant créer un microservice pour ce back-end, j’ ai utilisé un environnement virtuel propre à python en utilisant virtualenv. Cela m’a permis de créer un environnement vierge de toute dépendance python, puis d’installer les dépendances nécessaires à mon projet tout en créant automatiquement un fichier requirement.txt qui sera utilisable lors de la création de mon futur container docker.**  **J’ ai débuté l’application en installant flask, puis en créant mon premier point de terminaison.**  **Je me suis servi du site internet Postman.co afin de réaliser des requêtes HTTP pour tester mon back-end.**  **Une fois que tout était fonctionnel, j’ ai installé la dépendance MongoEngine et créé mon modèle me permettant de créer des instances de produits.**  **Puis j’ ai créé des points de terminaisons de création, d’affichage de un produit, d’affichage de tous les produits, de modification, et de suppression. Cela en utilisant les différentes méthodes de requête http GET, POST, PUT et DELETE.**  **La difficulté a était dans l’affichage de un produit car un id doit être passé dans l’url, puis récupéré en paramètre par le point de terminaison.**  **Pour les autres points de terminaisons, les informations sont transférées sous format de requête ne passant pas par l’url.**  **Une fois l’application fonctionnelle, j’ ai créé mon Dockerfile qui utilise le fichier requirement.txt pour les dépendances, puis j’ ai créé mon image Docker.**  **Je me suis ensuite servi de cette image pour créer mon conteneur.** | | | | | | | | |
|  | | | | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | | | | |
| **Les technologies utilisées sont python, le framework Flask, MongoEngine, MongoDb, ainsi que Docker.**  **Je me suis servi de Google pour faire des recherches, youtube pour regarder des tutoriels ainsi que Postman.co pour tester mon API.** | | | | | | | | |
|  | | | | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | | | | |
| **Étant un projet personnel, j’ai travaillé seul sur cette application.** | | | | | | | | |
|  | | | | | | | | |
| **4. Contexte** | | | | | | | | |
|  | | | | | |  | | |
| **Nom de l’entreprise, organisme ou association** ► | | | | | | **Aucun** | | |
|  | |  | | | | | | |
| **Chantier, atelier, service** | | ► | **Projet personnel** | | | | | |
| **Période d’exercice** | | ► | **Du** | | **04/07/2022** | | **au** | **11/07/2022** |
|  | | | | | | | | |
|  | | | | | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | | | | | |
| Cliquez ici pour taper du texte. | | | | | | | | |

|  |  | | |  | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Activité-type** | **2** | | | **Développer la partie back-end d’une application web ou web mobile en intégrant les recommandations de sécurité** | | | | |
| ***Exemple n°3*** ► | | | | ***back-end d’un site de paris sportif en Php et Sql*** | | | | |
|  | | | |  | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | | | | |
| **Lors d’un projet en groupe durant ma formation à l’Adrar, nous avons dû réaliser un site de paris sportifs.**  **Nous avons donc débuté ce projet en réalisant un cahier des charges.**  **Nous avons ensuite réalisé la conception de la base de données en réalisant un MCD et MLD, puis avons réalisé une arborescence des différents templates du projet afin de l’architecturer.**  **Par la suite, nous nous sommes réparties les tâches à effectuer. Ma responsabilité dans ce projet a été de réaliser l’ensemble de la partie back-end du projet pendant que mon collègue était en charge de la partie front-end.**  **Nous avons décidé de réaliser ce projet en php et sql, en utilisant l’architecture MVC, qui signifie modèle, vue, contrôleur.**  **Le principe étant que le modèle est une classe permettant de se connecter à la base de données. Il récupère les informations des enregistrements afin d’en faire des instances qui seront par la suite transmises à la vue. La vue représente l’ensemble des templates, soit l’ensemble des pages visibles par l’utilisateur. Lorsqu'un événement est réalisé sur le template par l’utilisateur, comme un clique sur un bouton par exemple, il va déclencher un contrôleur. C’est cette page qui va orchestrer toute l'algorithme de l’action qui est initié. Il va appeler les différents modèles nécessaires, réaliser des transformations de données en faisant appel à des méthodes de classes ou à des fonctions, puis renvoyer le tout à un template ou à un autre contrôleur.**  **Le projet sur lequel nous travaillons étant relativement petit et par manque d’expérience en php durant la création du projet, je n’ai pas créé de routeur pour naviguer entre les pages, l’appel des contrôleurs se faisant directement dans les templates.**  **Le site utilisant un principe d’inscription et de connexion, j’ai utilisé la session de php afin de savoir sur chaque contrôleur si l’utilisateur est connecté ou non. La session m’a aussi permis de savoir quel utilisateur est connecté en enregistrant le pseudo de l’utilisateur à l’intérieur de la session.** | | | | | | | | |
|  | | | | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | | | | |
| **Pour réaliser le Back-end de ce site Web, j’ ai utilisé le langage de programmation Php ainsi que la base de données Mysql. J’ ai utilisé Mysql Workbench afin de tester mes requêtes Sql, ainsi que PhpmyAdmin afin d’avoir une vision d’ensemble des tables de ma base de données.**  **J’ ai aussi utilisé le logiciel Looping.exe afin de réaliser le modèle conceptuel de données et le modèle logique de données, et le logiciel Gloomaps afin de créer un schéma de l’arborescence du site.** | | | | | | | | |
|  | | | | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | | | | |
| **J’ ai travaillé en équipe avec un collègue de formation.** | | | | | | | | |
|  | | | | | | | | |
| **4. Contexte** | | | | | | | | |
|  | | | | | |  | | |
| **Nom de l’entreprise, organisme ou association** ► | | | | | | **Aucun** | | |
|  | |  | | | | | | |
| **Chantier, atelier, service** | | ► | **Projet de groupe durant la formation** | | | | | |
| **Période d’exercice** | | ► | **Du** | | **24/04/2022** | | **au** | **30/04/2022** |
|  | | | | | | | | |
|  | | | | | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | | | | | |
| Cliquez ici pour taper du texte. | | | | | | | | |

| **Titres, diplômes, CQP, attestations de formation** | | |
| --- | --- | --- |
|  | | |
| *(facultatif)* | | |
| **Intitulé** | **Autorité ou organisme** | **Date** |
| Cliquez ici. | Cliquez ici pour taper du texte. | Cliquez ici pour sélectionner une date. |
| Cliquez ici. | Cliquez ici pour taper du texte. | Cliquez ici pour sélectionner une date. |
| Cliquez ici. | Cliquez ici pour taper du texte. | Cliquez ici pour sélectionner une date. |
| Cliquez ici. | Cliquez ici pour taper du texte. | Cliquez ici pour sélectionner une date. |
| Cliquez ici. | Cliquez ici pour taper du texte. | Cliquez ici pour sélectionner une date. |
| Cliquez ici. | Cliquez ici pour taper du texte. | Cliquez ici pour sélectionner une date. |
| Cliquez ici. | Cliquez ici pour taper du texte. | Cliquez ici pour sélectionner une date. |
| Cliquez ici. | Cliquez ici pour taper du texte. | Cliquez ici pour sélectionner une date. |
| Cliquez ici. | Cliquez ici pour taper du texte. | Cliquez ici pour sélectionner une date. |
| Cliquez ici. | Cliquez ici pour taper du texte. | Cliquez ici pour sélectionner une date. |

****

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

Je soussigné(e) [prénom et nom]  **Gaëtan Corin** ,

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

Fait à **Toulouse** le **22 aout 2022**

pour faire valoir ce que de droit.

Signature :

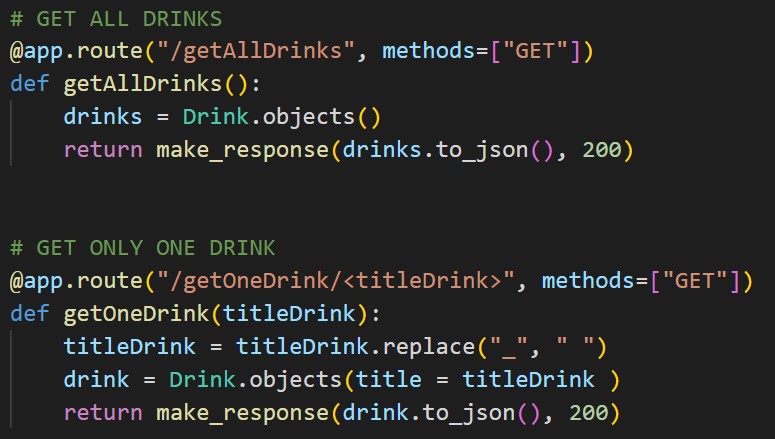
****

| **Documents illustrant la pratique professionnelle** |
| --- |
|  |
| *(facultatif)* |
| **Intitulé** |
| **p. 35 Front-end\_1 : Image dimensionnée à la largeur de l’écran superposé du texte responsive.** |
| **p. 36 Front-end\_1 : Importation des dépendances de la page Html** |
| **P. 36 Front-end\_1 : Mini carrousel d’image et texte** |
| **p. 37 Front-end\_1 : Section dynamique et Footer** |
| **p. 38 Front-end\_2 : Résultat visuel du front-end Javascript** |
| **p. 39 Front-end\_2 : Présentation de la page html** |
| **p. 39 Front-end\_2 : Appel à l’api pour récupérer les informations de toutes les boissons** |
| **p. 40 Front-end\_2 : Appel à l’api pour récupérer les informations de une boisson** |
| **p. 41 Front-end\_3 : Résultat visuel du Carrousel Django et Bootstrap** |
| **p. 41 Front-end\_3 : Code du Carrousel Django et Bootstrap** |

| **Documents illustrant la pratique professionnelle** |
| --- |
|  |
| *(facultatif)* |
| **Intitulé** |
| **p. 42 Back-end\_1 : Front-end fournis pour s’adapter à l’Api NodeJs** |
| **p. 43 Back-end\_1 : Routeur des objets de l’Api NodeJs** |
| **p. 43 Back-end\_1 : Modèle de Mongoose.** |
| **p. 44 Back-end\_1 : Présentation de deux Points de terminaisons NodeJs** |
| **p. 44 Back-end\_2 : Présentation de deux points de terminaison de Python Flask** |
| **p. 45 Back-end\_2 : Présentation du modèle de MongoEngine permettant de récupérer les enregistrements de MongoDb en Programmation orienté objet.** |
| **p. 45 Back-end\_2 : Présentation du Dockerfile utilisé pour transformer l’ API python en microservice Dockerisé** |
| **p. 46 Back-end\_3 : Présentation du résultat visuel du site de paris sportif** |
| **p. 46 Back-end\_3 : La vérification de connexion de l’utilisateur par la session Php** |
|  |

****

| **Annexes** |
| --- |
|  |
| *(Si le RC le prévoit)* |
| **FRONT END Exemple 1:**  L’image de respectant pas le responsive du texte  **Une image contenant texte, capture d’écran, moniteur, écran  Description générée automatiquement**  **FRONT END Exemple 1:**  Les différentes importations situées dans le Head de la page Html  **Une image contenant texte  Description générée automatiquement**  **FRONT END Exemple 1:**  Le mini Carrousel d’image utilisant la barre de navigation latérale.  **FRONT END Exemple 1:**  Le footer de l’intégration ainsi que les éléments réagissant au hover    **FRONT END Exemple 2:**  Présentation du résultat visuel du front-end javascript dynamique    **FRONT END Exemple 2:**  La page html ne contenant que le header et des <div> vide    **FRONT END Exemple 2:**  Appel à l’api pour récupérer les informations de toutes les boissons    **FRONT END Exemple 2:**  le addeventlistener et l’appel à l’api pour récupérer les informations de une boisson.    **FRONT END Exemple 3:**  Voici le résultat visuel du Carrousel réalisé en Django et Bootstrap    **FRONT END Exemple 3:**  Voici le code du Carrousel réalisé en Django et Bootstrap    **BACK END Exemple 1:**  Le front-end qui m’a été fourni pour s’adapter à l’Api NodeJs.      **BACK END Exemple 1:**  Voici le routeur des objets de l’API nodeJs  Une image contenant texte  Description générée automatiquement  **BACK END Exemple 1:**  Voici le modèle d’objet Express de Nodejs permettant de communiquer avec MongoDb en POO.    **BACK END Exemple 1:**  deux points de terminaisons en NodeJs.    **BACK END Exemple 2:**  Présentation de deux points de terminaison en python Flask |

****

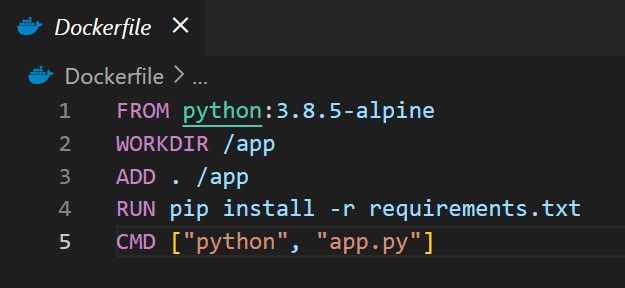
**BACK END Exemple 2:**

Voici le modèle d’objet MongoEngine de Python permettant de communiquer avec MongoDb en POO.

****

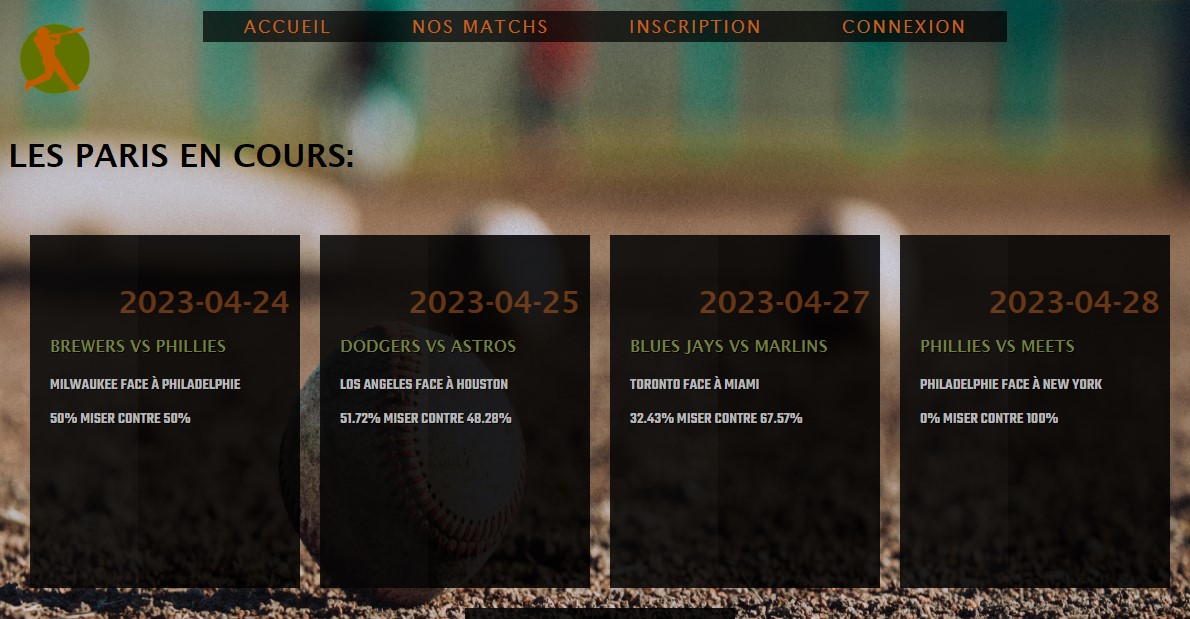
**BACK END Exemple 2:**

Voici le Dockerfile qui m’a permis de créer mon image Docker, puis mon container, afin de transformer l’Api python en microservice Dockerisé.

****

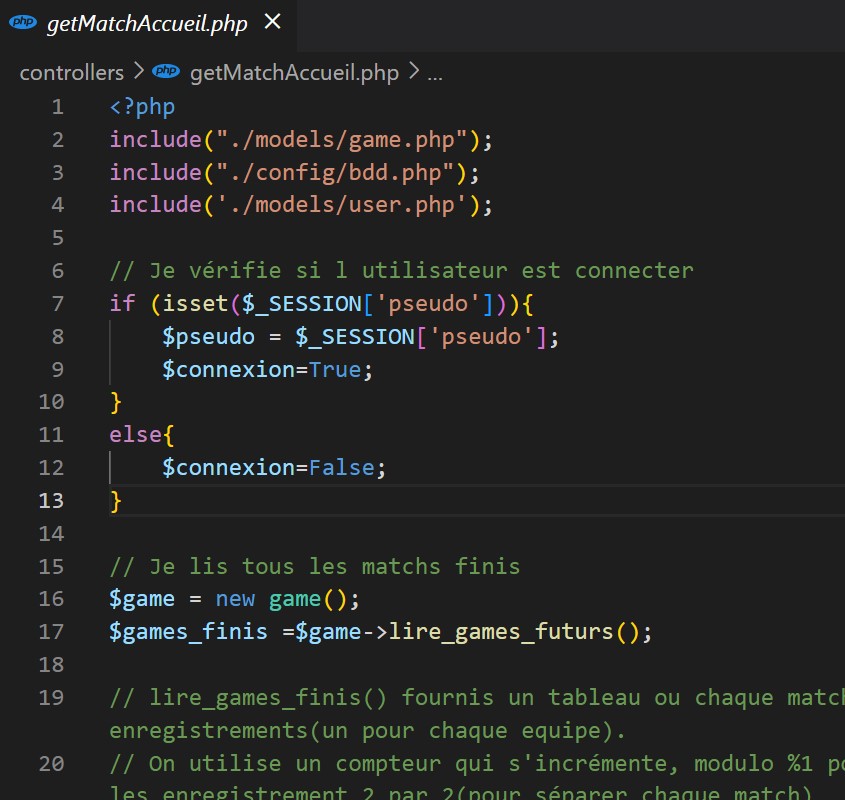
**BACK END Exemple 3:**

Présentation du résultat visuel du site de paris sportif

****

**BACK END Exemple 3:**

La vérification de connexion de l’utilisateur par la session Php

****