|  |  |  |
| --- | --- | --- |
|  |  |  |
| *Nom de naissance* |  | Pac |
| *Nom d’usage* |  | Entrez votre nom d’usage ici. |
| *Prénom* |  | Pierre |
| *Adresse* |  | 7 rue Rober Bordeux 50580 Pont-Remy |
|  |  |

|  |  |
| --- | --- |
| **Titre professionnel visé** | |
|  | |
| Concepteur Développeur d’Application | |
|  | |
| **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** | | |
| **Intitulé de l’activité-type n° 1 : Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité** | | **p.** | **5** |
|  |  Intitulé de l’exemple n° 1 Développer des composants d’accès aux données p. | p. | 5 |
|  |  Intitulé de l’exemple n° 2 Développer la partie front-end d’une interface utilisateur web p. | p. |  |
|  |  Intitulé de l’exemple n° 3 Développer la partie back-end d’une interface utilisateur web p | p. |  |
|  |  |  |  |
| **Intitulé de l’activité-type n° 2 : Concevoir et développer la persistance des données en intégrant les recommandations de sécurité** | | **p.** |  |
|  |  Intitulé de l’exemple n° 1 Concevoir une base de données p. | p. |  |
|  |  Intitulé de l’exemple n° 2 Mettre en place une base de données p. | p. |  |
|  |  Intitulé de l’exemple n° 3 Développer des composants dans le langage d’une base de données p | p. |  |
|  |  |  |  |
| **Intitulé de l’activité-type n° 3 : Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité** | | **p.** |  |
|  |  Intitulé de l’exemple n° 1 Concevoir une application p. | p. |  |
|  |  Intitulé de l’exemple n° 2 Développer des composants métier p. | p. |  |
|  |  Intitulé de l’exemple n° 3 Construire une application organisée en couche 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** | **Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité** | | | |
| ***Exemple n°1***  | | Développer des composants d’accès aux données | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
|  | | | | | |
| ans notre application, élaborée avec Spring Boot, JPA, Hibernate, et Angular, les composants d'accès aux données sont essentiels, faisant le lien entre les utilisateurs et nos bases de données, et assurant une gestion des données fiable. Nous utilisons JPA et Hibernate pour gérer la persistance des données, guidés par nos modèles et repositories. La Java Persistence API (JPA) offre une interface pour gérer la persistance des données dans nos applications Java, favorisant une gestion orientée objet des données relationnelles et augmentant la productivité des développeurs grâce à sa portabilité et sa syntaxe simplifiée. Hibernate, une implémentation notable de JPA, facilite les opérations CRUD et s’intègre harmonieusement avec Spring Boot, optimisant le développement et réduisant la configuration grâce à l'automatisation de Spring Boot.  Configuration de la Source de Données  Ma décision d’opter pour PostgreSQL comme système de gestion de base de données est motivée par sa fiabilité et ses performances dans la gestion des données relationnelles. Ainsi, j'assure la configuration de la connexion à la base de données avec soin : | | | | | |
| Je spécifie ici les paramètres essentiels comme l'URL, le nom d'utilisateur, le mot de passe, et le pilote JDBC pour garantir une connexion fluide et fonctionnelle à la base de données.  Configuration JPA et Hibernate  L’intégration de JPA et Hibernate vise à simplifier l'accès et la gestion des données dans mon application :    Ici, le dialecte PostgreSQL est employé, et la propriété ddl-auto est configurée à update pour synchroniser automatiquement le schéma de la base de données avec les entités au démarrage de l'application.    Le package spécifié indique à JPA où localiser les entités, assurant ainsi une gestion appropriée des objets dans le contexte de persistance.  Afin de faciliter le débogage durant le développement, j’active l’affichage des requêtes SQL et leur formatage :    Cette pratique m’aide à comprendre et optimiser les interactions avec la base de données, améliorant ainsi la performance globale de l’application.  La configuration des composants d’accès aux données a été minutieusement effectuée pour assurer une persistance des données robuste et performante tout au long du cycle de vie de mon application. Bien entendu, cette configuration est susceptible d’être ajustée selon les évolutions et les besoins futurs du projet, en garantissant toujours l’intégrité et la sécurité des données manipulées.  Modélisation des Données et Relations entre Entités  Lors de la création de mon application, j'accorde une attention particulière à la définition des modèles qui serviront à créer les tables dans la base de données et à définir la logique métier.  Rôle des Modèles et Création des Tables  Dans le contexte JPA/Hibernate, le modèle Sandbox dans mon application joue un rôle crucial. Les modèles, annotés avec @Entity, se transforment en tables dans la base de données au démarrage de l'application, suivant la logique déclarative inscrite dans le code.  L’annotation @Entity indique à Hibernate que la classe Sandbox est une entité et doit être mappée à une table de base de données. @Table(name = "SANDBOX") spécifie le nom de cette table. Chaque instance de Sandbox correspond à une entrée dans la table "SANDBOX" de la base de données.    La propriété id est annotée avec @Id, indiquant qu’elle est la clé primaire de la table. @GeneratedValue détermine comment les valeurs de la clé primaire sont générées. Ici, GenerationType.IDENTITY signifie que la base de données génère automatiquement la valeur lors de l'insertion de l'entité.  Relations entre Entités : OneToMany  La gestion des relations entre les entités est cruciale pour refléter la logique métier dans la base de données.    @OneToMany représente une relation un-à-plusieurs entre Sandbox et Entretien. Le paramètre mappedBy indique que la propriété sandbox dans l’entité Entretien est la clé étrangère de cette relation. CascadeType.ALL signifie que toutes les opérations effectuées sur une instance de Sandbox (comme la sauvegarde, la mise à jour, ou la suppression) sont propagées aux entités associées dans entretiens. FetchType.EAGER, d’autre part, signifie que les entités associées sont chargées automatiquement avec l’entité parente.  La compréhension et l'implémentation précises des modèles et des relations sont vitales pour assurer que l'application interagit avec la base de données de manière cohérente et performante, alignant ainsi la persistance des données avec la logique métier souhaitée.  Le Repositoire : Facilitateur d'Accès aux Données  Dans mon projet, le repositorie joue un rôle essentiel en tant que pont entre la logique métier de l'application et les opérations de la base de données, facilitant ainsi la récupération et la persistance des données.  Structure et Fonctionnalité des Repositories    L’annotation @Repository indique à Spring que cette interface est un repositorie, un composant chargé de gérer la persistance des entités Candidat. Il hérite de JpaRepository, qui contient des méthodes pour les opérations de base de données les plus courantes, telles que la sauvegarde, la suppression, et la recherche d’entités, sans qu’il soit nécessaire d’implémenter ces méthodes manuellement. | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
|  | | | | | |
| Spring Boot, JPA, Hibernate, PostgreSQL | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
|  | | | | | |
| Mentoré par le référent technique de l’équipe | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | *Mipih* | |
| Chantier, atelier, service | | | DO-GRH | | |
| Période d’exerciceDu : *17/07/2023*au :*06/10/2023* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activité-type** | **1** | **Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité** | | | |
| ***Exemple n°2***  | | Développer la partie front-end d’une interface utilisateur web | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
|  | | | | | |
| Mentoré par le référent technique de l’équipe | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | *Mipih* | |
| Chantier, atelier, service | | | DO-GRH | | |
| Période d’exerciceDu : *17/07/2023*au :*06/10/2023* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activité-type** | **1** | **Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité** | | | |
| ***Exemple n°3***  | | Développer la partie back-end d’une interface utilisateur web | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
|  | | | | | |
| Mentoré par le référent technique de l’équipe | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | *Mipih* | |
| Chantier, atelier, service | | | DO-GRH | | |
| Période d’exerciceDu : *17/07/2023*au :*06/10/2023* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activité-type** | **2** | **Concevoir et développer la persistance des données en intégrant les recommandations de sécurité** | | | |
| ***Exemple n° 1***  | | Concevoir une base de données | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
|  | | | | | |
| Concevoir une base de données  Avant de concevoir une base de données, il est crucial de conceptualiser et de visualiser le flux de fonctionnalités et d'interactions au sein du système. Pour ce faire, des outils tels que les diagrammes de séquence et d'activité sont indispensables.    Les diagrammes de séquence, illustrés ci-dessus pour un candidat, détaillent les interactions entre différents objets dans un système en mettant l'accent sur l'ordre des interactions. Ils offrent une vue claire et détaillée des fonctionnalités et des responsabilités de chaque composant, permettant ainsi de comprendre précisément le flux de processus et les interactions au sein du système.    D'autre part, les diagrammes d'activité, comme montré ci-dessus pour un candidat, mettent en lumière le flux de travail au sein du système. Ils illustrent les étapes et les activités réalisées par les utilisateurs ou les systèmes, les conditions et les décisions qui structurent le parcours de l'utilisateur.  En illustrant les étapes, les activités, et les décisions prises par un administrateur, ces diagrammes jouent un rôle crucial dans la transmission d'une compréhension unifiée des exigences et comportements souhaités de l'application entre les différentes parties prenantes du projet.  Ces diagrammes préparent le terrain pour la conception de la base de données en offrant une fondation solide pour élaborer une architecture cohérente et réaliser une implémentation sans faille.  Dans le cadre de ce projet, la conceptualisation et la réalisation de diagrammes, notamment les diagrammes de séquence et d'activité, étaient indispensables avant de s'avancer dans la conception de la base de données. Ils permettent de visualiser les interactions et le flux de travail entre les différents utilisateurs du système, qu'il s'agisse des candidats ou des administrateurs, et facilitent une compréhension holistique du comportement attendu et des exigences fonctionnelles de l'application. Les diagrammes d'activité, tels qu'illustrés pour le parcours du candidat, sont particulièrement utiles pour dépeindre de manière structurée les différents scénarios d'utilisation, les conditions et les décisions qui informent l'expérience utilisateur au sein du système.  En regardant à travers le prisme de ces diagrammes, les équipes projet peuvent acquérir une vision commune et unifiée des fonctionnalités requises et des interactions entre les divers composants du système, jetant ainsi les bases d'une conception de base de données rationnelle et cohérente.  Intégrer ces éléments préliminaires avec une structure de base de données bien pensée était donc crucial. Pour ce faire, après une série d'échanges approfondis avec l'équipe projet et une analyse détaillée du cahier des charges, j'ai opté pour l'utilisation de Looping pour construire cette base de données. Cela a été réalisé en dépit de l'absence de certaines fonctionnalités dans Looping, telles que l'intégration d'énumérations ou de classes abstraites, qui auraient pu optimiser davantage la représentation visuelle de la base de données (Annexe N°1).  La nécessité de gérer deux types d'utilisateurs, à savoir les administrateurs et les candidats, a engendré la création d'une classe abstraite, évitant ainsi la redondance et regroupant les champs communs entre ces deux entités distinctes. Les classes administrateur et candidat en héritent, promouvant la modularité et facilitant les modifications futures, tout en maintenant une cohérence structurelle.  En outre, pour satisfaire à des relations complexes, des tables de liaison ont été incorporées pour établir des relations « Many-to-Many » entre divers éléments comme les candidats et les propositions ainsi qu'entre les QCM et les questions, avec des champs supplémentaires pour assurer une flexibilité et une organisation optimales.  Ainsi, cette structure de base de données robuste et rationalisée sert de fondement à l'application, permettant des interactions fluides entre les différentes entités, et garantissant l'intégrité et la durabilité des données manipulées. En alignant rigoureusement la conception de la base de données sur les besoins définis par l'équipe et le cahier des charges, un environnement de données efficace et évolutif a été mis en place, prêt à supporter les évolutions et adaptations futures du projet. | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
|  | | | | | |
| Draw.io, Looping | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
|  | | | | | |
| Mentoré par le référent technique de l’équipe | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | *Mipih* | |
| Chantier, atelier, service | | | DO-GRH | | |
| Période d’exerciceDu : *17/07/2023*au :*06/10/2023* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activité-type** | **2** | **Concevoir et développer la persistance des données en intégrant les recommandations de sécurité** | | | |
| ***Exemple n° 2***  | | Mettre en place une base de données | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
|  | | | | | |
| Dans le développement d'applications avec Spring Boot, la mise en place d'une base de données est une étape cruciale, et pour ce faire, l'utilisation de Java Persistence API (JPA) et Hibernate s'avère souvent indispensable. JPA est une interface de programmation standard qui sert de spécification pour la gestion relationnelle des données dans les applications Java. Elle permet de définir la manière dont les données doivent être persistées, gérées et récupérées, offrant une abstraction élevée par rapport à la base de données sous-jacente.  Hibernate, quant à lui, est l'un des fournisseurs de persistance les plus populaires pour JPA. Il implémente les spécifications de JPA et offre des fonctionnalités supplémentaires, tels que la gestion de cache de second niveau et des mécanismes de requêtage avancés, qui optimisent les performances de l'application.  En utilisant JPA et Hibernate dans un contexte Spring Boot, les développeurs peuvent bénéficier d'une intégration fluide et d'une configuration simplifiée. Cette combinaison permet de manipuler les entités de base de données de manière intuitive et orientée objet, en minimisant le besoin d'écrire des requêtes SQL explicites. Elle encourage ainsi le développement rapide, tout en garantissant la qualité et la robustesse de l'accès aux données dans l'application, rendant la gestion de la persistance des données plus flexible et moins sujette à erreurs.  La mise en place d'une architecture orientée objet efficace est cruciale dans un tel environnement. Une classe abstraite comme Personne a été définie dans notre cas pour représenter les éléments communs à différentes entités, les class Admin et Candidat qui sont en réalité des spécialisations de la classe Personne. Cette approche d'utilisation de classes abstraites et d'héritage, combinée à l'abstraction des données offerte par JPA et Hibernate, permet une représentation cohérente et logique du modèle de données.    Les classes Admin et Candidat, en étendant la classe Personne, héritent de ses attributs et méthodes, mais peuvent également déclarer des propriétés et des comportements qui leur sont propres. La stratégie d'héritage SINGLE\_TABLE implique que toutes les entités de cette hiérarchie sont représentées dans une seule table, et la colonne type\_personne sert à discriminer entre les différentes instances de sous-classes.    32 - class admin qui étend Personne class Candidat qui étend Personne  Pour interagir avec la base de données, des repositories JPA, tels que AdminRepository et CandidatRepository, sont définis. Ces repositories fournissent une interface pour effectuer des opérations CRUD sur les entités correspondantes et définissent des méthodes pour effectuer des requêtes sur la base de données selon divers critères.      L'utilisation de classes abstraites et de l'héritage, en harmonie avec JPA et Hibernate dans Spring Boot, permet de créer un modèle de données structuré et extensible. Cela facilite le développement et la maintenance du code tout en optimisant l'interaction avec la base de données grâce à une interface de repository intuitive, minimisant ainsi les erreurs potentielles et favorisant un développement plus rapide et plus robuste.  Pour établir des relations entre les entités dans le contexte de JPA et Hibernate, divers types de mappages relationnels sont mis à disposition pour refléter fidèlement les relations de base de données dans le modèle objet. Ces mappages facilitent l'interaction avec la base de données en gérant de manière transparente la persistance des données, les mises à jour et les requêtes, tout en préservant les principes de la modélisation objet. Voici deux des mappages relationnels les plus couramment utilisés : OneToMany.  OneToMany  La relation @OneToMany indique qu'une instance d'une entité est associée à plusieurs instances d'une autre entité. L'exemple de votre entité Question illustre une telle relation. La classe Question possède plusieurs instances de l'entité Proposition grâce à la propriété propositions, indiquant ainsi qu'une question peut avoir plusieurs propositions :    Dans ce cas, le mappedBy signifie que la relation est bidirectionnelle et que l'autre entité (Proposition) contient la clé étrangère grâce à la propriété question :    Souvent, dans le développement des modèles de données, les développeurs se confrontent à des situations où une relation ManyToMany devient nécessaire pour décrire adéquatement la relation entre deux entités. Toutefois, en pratique, implémenter directement une relation ManyToMany peut parfois introduire des complications, notamment lorsqu'il s'agit de représenter des informations supplémentaires concernant la relation elle-même. En conséquence, une approche courante consiste à "casser" la relation ManyToMany en introduisant une table intermédiaire, réduisant ainsi la complexité et offrant un mécanisme pour gérer de manière élégante des attributs supplémentaires de la relation.  Prenons l'exemple de votre entité QuestionQcm, qui sert précisément de cette table intermédiaire, assurant une liaison entre les entités Question et Qcm :    La présence de l'objet QuestionQcmId montre que la clé primaire de QuestionQcm est en fait une clé composée des identifiants des deux autres entités liées :    En procédant ainsi, la relation ManyToMany originale entre Question et Qcm est décomposée en deux relations ManyToOne dans QuestionQcm : une vers Question et une autre vers Qcm. Cette technique de décomposition non seulement permet de gérer plus aisément la persistance et la récupération des données avec JPA et Hibernate, mais elle ouvre également la voie pour ajouter d'autres attributs relatifs à la relation (comme ordre dans l'exemple), chose qui aurait été complexe voire impossible avec un mappage ManyToMany direct. Par conséquent, cette approche apporte une flexibilité et une scalabilité accrues, permettant aux développeurs de construire des applications robustes et maintenables. | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
|  | | | | | |
| Spring Boot, JPA, Hibernate, PostgreSQL | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
|  | | | | | |
| Mentoré par le référent technique de l’équipe | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | *Mipih* | |
| Chantier, atelier, service | | | DO-GRH | | |
| Période d’exerciceDu : *17/07/2023*au :*06/10/2023* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activité-type** | **2** | **Concevoir et développer la persistance des données en intégrant les recommandations de sécurité** | | | |
| ***Exemple n° 3***  | | Développer des composants dans le langage d’une base de données | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
|  | | | | | |
| Mentoré par le référent technique de l’équipe | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | *Mipih* | |
| Chantier, atelier, service | | | DO-GRH | | |
| Période d’exerciceDu : *17/07/2023*au :*06/10/2023* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activité-type** | **3** | **Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité** | | | |
| ***Exemple n° 1***  | | Concevoir une application | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
|  | | | | | |
| Mentoré par le référent technique de l’équipe | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | *Mipih* | |
| Chantier, atelier, service | | | DO-GRH | | |
| Période d’exerciceDu : *17/07/2023*au :*06/10/2023* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activité-type** | **3** | **Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité** | | | |
| ***Exemple n°2***  | | Développer des composants métier | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
|  | | | | | |
| Mentoré par le référent technique de l’équipe | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | *Mipih* | |
| Chantier, atelier, service | | | DO-GRH | | |
| Période d’exerciceDu : *17/07/2023*au :*06/10/2023* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activité-type** | **3** | **Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité** | | | |
| ***Exemple n°3***  | | Construire une application organisée en couche | | | |
|  | | | | | |
|  | | | | | |
| **1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :** | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **2. Précisez les moyens utilisés :** | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| **3. Avec qui avez-vous travaillé ?** | | | | | |
|  | | | | | |
| Mentoré par le référent technique de l’équipe | | | | | |
|  | | | | | |
|  | | | | |
|  | | | | | |
| **4. Contexte** | | | | | |
|  | | | |  | |
| Nom de l’entreprise, organisme ou association | | | | *Mipih* | |
| Chantier, atelier, service | | | DO-GRH | | |
| Période d’exerciceDu : *17/07/2023*au :*06/10/2023* | | | | | |
|  | | | | |
|  | | | | |
| **5. Informations complémentaires** *(facultatif)* | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |

|  |  |  |
| --- | --- | --- |
| **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. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

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

Pierre Pac

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.

Pont-Rémy

03/10/2023

Fait à le

pour faire valoir ce que de droit.

Signature :



|  |
| --- |
| **Documents illustrant la pratique professionnelle** |
|  |
| *(facultatif)* |
| **Intitulé** |
| Cliquez ici pour taper du texte. |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

|  |
| --- |
| **Annexes** |
|  |
| Annexe 1 : Schéma base de données |