

DOSSIER PROFESSIONNEL (DP)

Nom de naissance ▶ Alo
Nom d'usage ▶
Prénom ▶ Rany
Adresse ▶ 70 rue Bernard du Bois 13001 Marseille

Titre professionnel visé

Concepteur développeur d'applications

MODALITE D'ACCES :

- ☒ Parcours de formation
- ☐ Validation des Acquis de l'Expérience (VAE)

DOSSIER PROFESSIONNEL (DP)

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.



Sommaire

Exemples de pratique professionnelle

Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité	p.	6
▶ Intitulé de l'exemple n° 1 Développement d'une panel admin web pour notre application mobile : forum - Code hub	p.	6
▶ Intitulé de l'exemple n° 2 Développement d'une application en Python : articleGenerator	p.	11
▶ Intitulé de l'exemple n° 3	p.	
Concevoir et développer la persistance des données en intégrant les recommandations de sécurité	p.	14
▶ Intitulé de l'exemple n° 1 : Développement d'une application : Forum – Code hub	p.	14
▶ Intitulé de l'exemple n° 2	p.	
▶ Intitulé de l'exemple n° 3	p.	
Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité	p.	17
▶ Intitulé de l'exemple n° 1 : Développement d'une application : Forum – Code hub	p.	17
▶ Intitulé de l'exemple n° 2	p.	

DOSSIER PROFESSIONNEL (DP)

► Intitulé de l'exemple n° 3

p.

Titres, diplômes, CQP, attestations de formation *(facultatif)*

p.

23

Déclaration sur l'honneur

p.

24

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 ► Développement d'une panel admin web pour notre application mobile : forum - Code hub

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

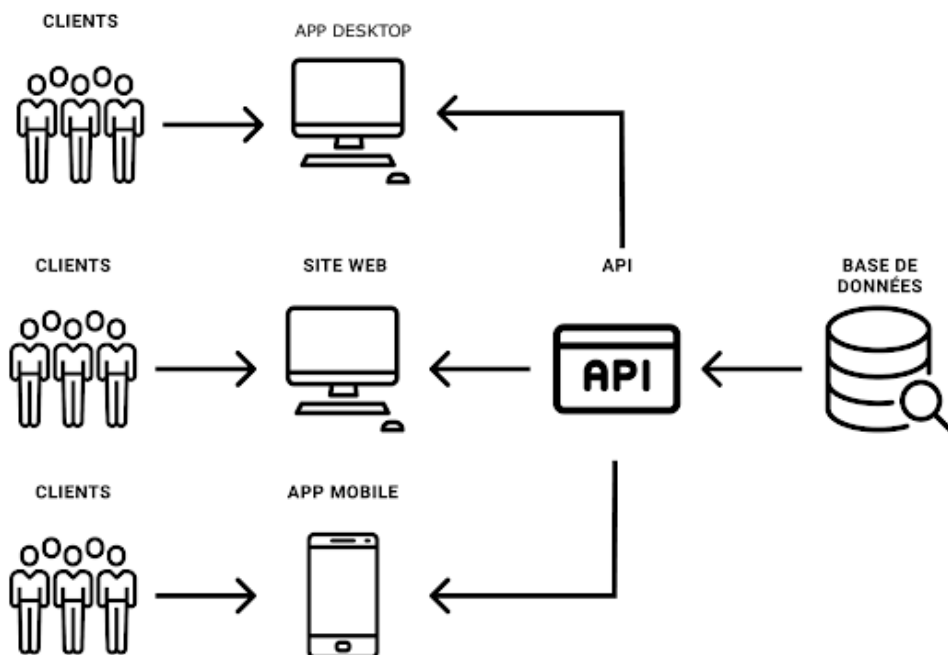
Code Hub est un forum de discussion en ligne conçu pour permettre aux utilisateurs de partager des articles et de commenter les articles des autres. Le cahier des charges pour le développement de ce forum a été fourni par notre école Laplateforme.

Notre équipe a travaillé sur la création de la panel admin de l'application. Les administrateurs ont des fonctionnalités supplémentaires par rapport aux utilisateurs standards. Ils peuvent ajouter, modifier et supprimer des utilisateurs, ainsi que des articles et des commentaires en cas de violation des règles du forum.

Dans le cadre de notre projet de développement de la panel admin de notre application de forum, nous avons effectué plusieurs tâches importantes pour la création et la mise en place de l'application. Voici un résumé des principales tâches que nous avons réalisées :

1. Création de la maquette : Nous avons élaboré une maquette de l'application qui répond aux exigences du cahier des charges de La Plateforme tout en respectant des critères esthétiques pour une bonne lisibilité de l'interface.
2. Création de l'API : Nous avons mis en place un backend pour notre application de forum en créant une API. Pour cela, nous avons choisi d'utiliser Api Platform avec Symfony, qui est un outil très puissant pour la création d'API RESTful en PHP. Api Platform fournit des fonctionnalités avancées telles que la documentation de l'API, la gestion des erreurs, la validation des données, etc.

DOSSIER PROFESSIONNEL (DP)



Nous avons créé des endpoints qui permettent de récupérer, ajouter, modifier et supprimer des données à partir de la base de données de l'application. Ces endpoints sont appelés par l'application frontend (le panel admin) pour effectuer des opérations sur les données, telles que l'ajout d'un nouvel article, la suppression d'un commentaire, la modification d'un profil utilisateur, etc.

User			
POST	/api/inscription	Creates a User resource.	✓ 🔒
GET	/api/profileMe	Retrieves a User resource.	✓ 🔒
GET	/api/user/{id}	Retrieves a User resource.	✓ 🔒
DELETE	/api/userDelete/{id}	Removes the User resource.	✓ 🔒
PATCH	/api/userProfileEdit/{id}	Updates the User resource.	✓ 🔒
GET	/api/users	Retrieves the collection of User resources.	✓ 🔒

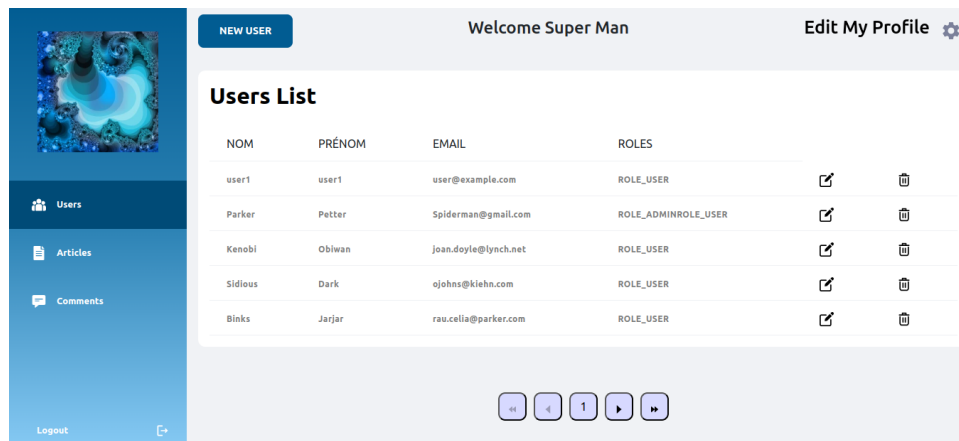
- Utilisation de l'ORM Doctrine de Symfony : Pour faciliter l'accès aux données dans l'application, nous avons utilisé l'ORM Doctrine de Symfony. L'ORM (Object-Relational Mapping) est une technique de programmation qui permet de faire correspondre les objets d'une application avec les tables d'une base de données relationnelle. Avec Doctrine, nous avons créé des entités qui représentent les différents types de données dans l'application (par exemple, les articles, les

DOSSIER PROFESSIONNEL (DP)

commentaires, les utilisateurs, etc.) et nous avons défini les relations entre ces entités.

L'avantage de l'utilisation d'un ORM est qu'il simplifie l'accès aux données en évitant d'avoir à écrire du SQL à la main. Au lieu de cela, les opérations sur les données sont effectuées en utilisant des méthodes simples et familières en PHP, telles que "find", "persist", "remove", etc.

4. Création d'un panel admin en React JS : Nous avons créé un frontend en React JS pour notre application de forum, qui permet aux administrateurs de gérer les fonctionnalités avancées du forum, telles que la modération des commentaires, la suppression de messages inappropriés, la gestion des utilisateurs, etc.



2. Précisez les moyens utilisés :

- ▶ Visual Studio Code (VSCode) est un éditeur de code source gratuit.
- ▶ PhpStorm est un environnement de développement intégré (IDE) conçu spécifiquement pour le développement en PHP. J'ai utilisé PhpStorm pour créer l'api symfony car il offre une gamme de fonctionnalités avancées pour faciliter le processus de développement.
- ▶ Symfony et API Platform : Symfony est un framework PHP open source. API Platform

DOSSIER PROFESSIONNEL ^(DP)

est une extension de Symfony qui permet de créer des API web RESTful. Nous avons utilisé Symfony et API Platform pour créer l'API backend de l'application.

- ▶ React JS : C'est une bibliothèque JavaScript open source pour la création d'interfaces utilisateur. Nous avons utilisé React JS pour développer l'interface frontend de l'application.
- ▶ GitHub : C'est une plateforme de développement logiciel qui permet de stocker et de partager du code source. Nous avons utilisé GitHub pour le versionnage du code source et la collaboration entre les membres de l'équipe.
- ▶ Trello : C'est un outil de gestion de projet en ligne basé sur la méthode Kanban. Nous avons utilisé Trello pour organiser les différentes tâches à effectuer pour réaliser le projet et pour suivre leur progression.

3. Avec qui avez-vous travaillé ?

En autonomie

4. Contexte

Nom de l'entreprise, organisme ou association

La Plateforme

Chantier, atelier, service



Dans le cadre de notre formation en tant que "Concepteur développeur d'application".

Période d'exercice

▶ Du : 02/01/2023 au : 28/02/2023

5. Informations complémentaires *(facultatif)*

DOSSIER PROFESSIONNEL ^(DP)

Exemple n°2 Développement d'une application en python : articleGenerator

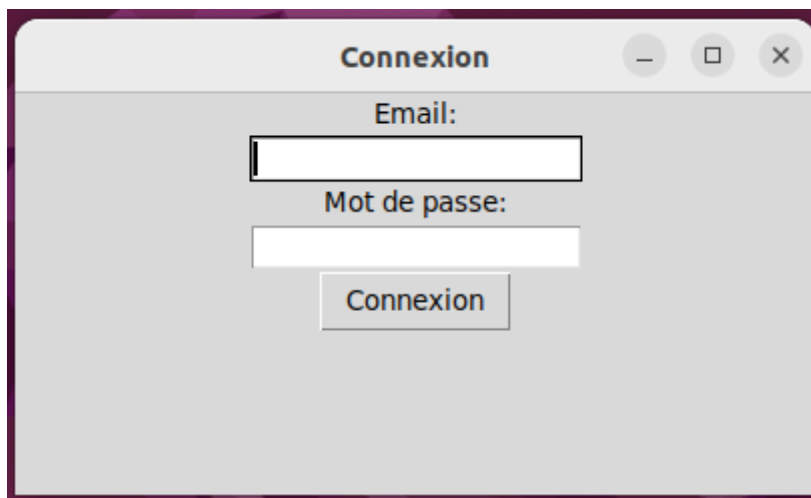


1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Le projet "articleGenerator" est une application qui utilise une interface graphique tkinter en Python pour se connecter à notre API Symfony. L'API Symfony, à son tour, effectue des appels vers l'API OpenAI pour générer un article à partir d'un titre donné.

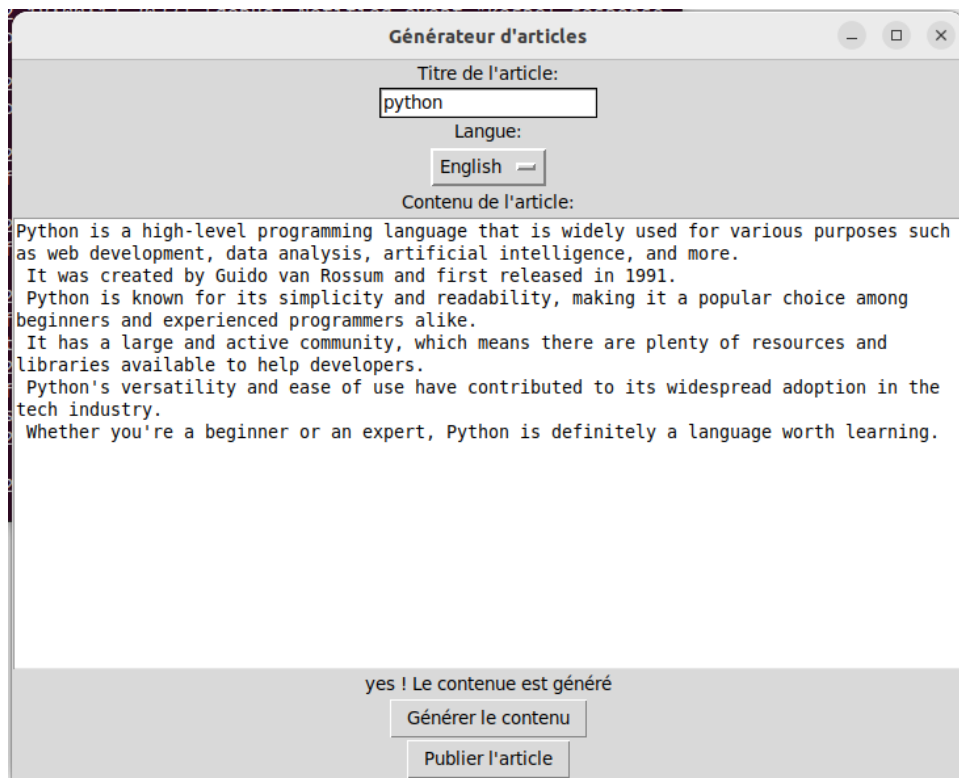
Le code est réparti en trois fichiers : "login.py", "generator.py" et "main.py".

"login.py" contient la classe "LoginWindow" qui représente la fenêtre de connexion de l'application. L'utilisateur peut entrer son email et son mot de passe, et ensuite l'application envoie une requête POST à l'API Symfony pour vérifier les informations de connexion. Si l'authentification réussit (code de statut 200), la fenêtre de génération d'articles est affichée.



"generator.py" contient la classe "GeneratorWindow" qui représente la fenêtre de génération d'articles. L'utilisateur peut saisir un titre d'article, puis l'application envoie une requête POST à l'API Symfony avec le titre pour générer le contenu de l'article à partir de l'API OpenAI. Le contenu généré est affiché dans une zone de texte. L'utilisateur a également la possibilité de publier l'article en envoyant une requête POST à l'API Symfony avec le titre et le contenu.

DOSSIER PROFESSIONNEL (DP)



"main.py" est le point d'entrée principal du programme. Il crée une instance de la classe "LoginWindow" et la lance.

Bibliothèques utilisées : tkinter pour l'interface graphique, requests pour effectuer des requêtes HTTP, json pour la manipulation de données JSON, codecs pour décoder le texte de réponse, et re pour le traitement des expressions régulières.

2. Précisez les moyens utilisés :

- ▶ Visual Studio Code (VSCode) est un éditeur de code source gratuit.
- ▶ PhpStorm est un environnement de développement intégré (IDE) conçu spécifiquement pour le développement en PHP. J'ai utilisé PhpStorm pour créer l'api symfony car il offre une gamme de fonctionnalités avancées pour faciliter le processus de développement.

DOSSIER PROFESSIONNEL ^(DP)

- ▶ **Symfony et API Platform** : Symfony est un framework PHP open source. API Platform est une extension de Symfony qui permet de créer des API web RESTful. J'ai utilisé Symfony et API Platform pour créer l'API backend de l'application.
- ▶ **Python** : J'ai utilisé python qui est un langage de programmation polyvalent, simple et lisible. Il offre une large bibliothèque standard et est portable.
- ▶ **GitHub** : C'est une plateforme de développement logiciel qui permet de stocker et de partager du code source.

3. Avec qui avez-vous travaillé ?

En autonomie

4. Contexte

Nom de l'entreprise, organisme ou association

La Plateforme

Chantier, atelier, service

Dans le cadre de notre formation en tant que "Concepteur développeur d'application".

Période d'exercice

▶ Du : 01/05/2023 au : 05/05/2023

5. Informations complémentaires *(facultatif)*

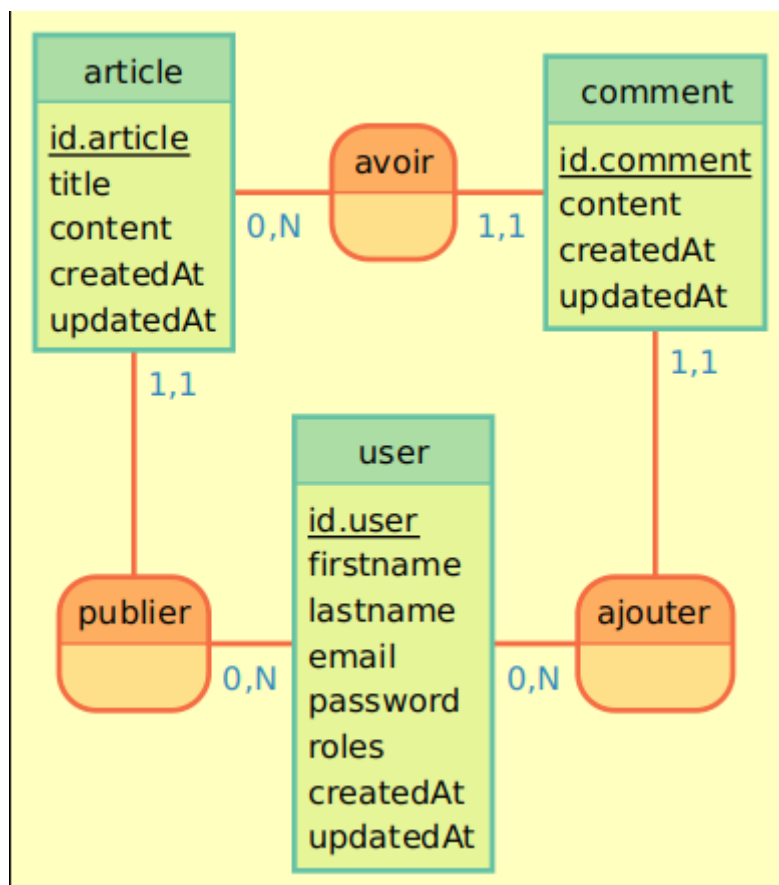
Activité- type 2

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

Exemple n° 1 ► Développement d'une application mobile : Forum – Code hub

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

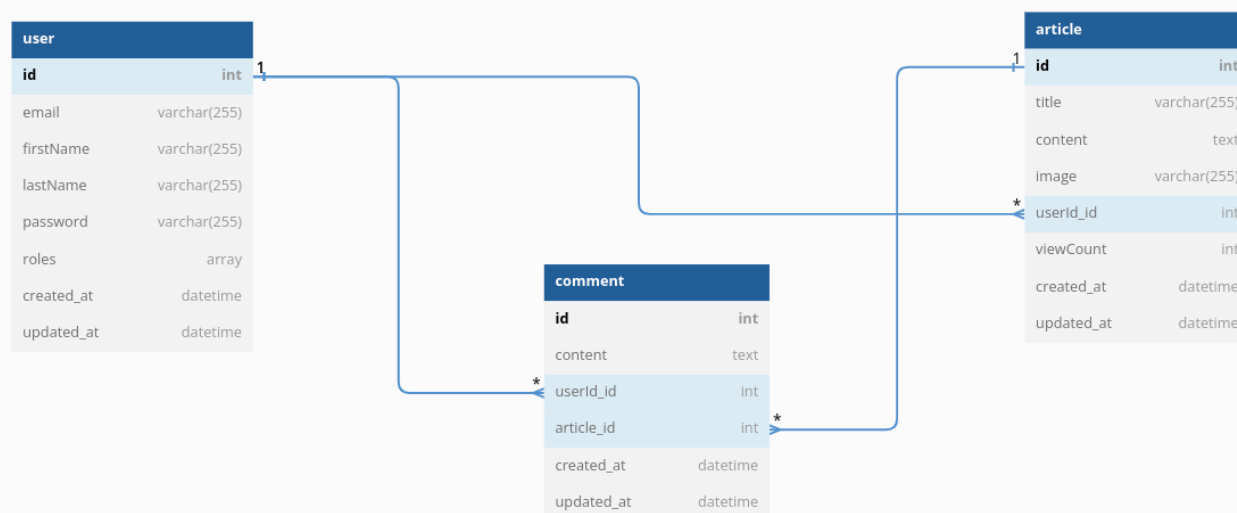
Pour notre application Code Hub, nous avons besoin d'une base de données qui contient trois tables : "user", "article" et "comment". La conception de la base de données a commencé par la création d'un Modèle Conceptuel de Données (MCD) en utilisant Mocodo. Le MCD représente les entités, les relations et les attributs des données de l'application.



À partir du MCD, nous avons créé le Modèle Logique de Données (MLD), qui décrit la structure de

DOSSIER PROFESSIONNEL (DP)

la base de données sous forme de tables, de colonnes et de relations entre les tables. La table "user" contient des informations sur les utilisateurs de l'application, telles que leur nom, leur adresse email et leur mot de passe crypté. La table "article" contient les articles créés par les utilisateurs, avec des informations telles que le titre, le contenu et la date de création. La table "comment" contient les commentaires associés aux articles, avec des informations telles que le contenu et la date de création.



Nous avons utilisé l'ORM Doctrine de Symfony pour faciliter la création et la gestion des tables de notre base de données. Nous avons commencé par définir les entités de notre application, qui ont été converties en tables dans notre base de données grâce à Doctrine. Nous avons également ajouté des contraintes de clés étrangères pour garantir l'intégrité de nos données et prévenir toute erreur de référence.

Une fois la conception de notre base de données terminée, nous avons migré nos données vers MariaDB via phpMyAdmin. MariaDB est un système de gestion de bases de données open source, qui est compatible avec MySQL et largement utilisé dans le développement web.

2. Précisez les moyens utilisés :

- ▶ Visual Studio Code (VSCode) : éditeur de code source gratuit.
- ▶ PhpMyAdmin : une application web open source pour gérer des bases de données MySQL et MariaDB.

DOSSIER PROFESSIONNEL ^(DP)

- ▶ Doctrine : ORM (Object-Relational Mapping) pour PHP, qui permet de gérer les interactions avec une base de données de manière objet.
- ▶ Mocodo : un outil en ligne qui permet de concevoir des modèles de données de manière graphique, on a utilisé Mocodo pour générer le modèle conceptuel de données (MCD).
- ▶ DBDiagram est un outil en ligne gratuit permettant de concevoir et de modéliser visuellement des bases de données relationnelles, on a utilisé dbdiagram pour générer le Modèle Logique de Données MLD.

3. Avec qui avez-vous travaillé ?

On était une équipe de quatre développeurs :

- Christophe Calmes
- Mathieu Ruiz
- Geoffrey Meca
- Rany Alo

4. Contexte

Nom de l'entreprise, organisme ou association ▶

La Plateforme

Chantier, atelier, service ▶

Dans le cadre de notre formation en tant que
"Concepteur développeur d'application".

Période d'exercice ▶ Du : 02/01/2023 au : 28/02/2023

5. Informations complémentaires *(facultatif)*

Activité- type 3

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

Exemple n° 1 ► Développement d'une application mobile : Forum – Code hub

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Le forum Code Hub est une application mobile Android, développée pour permettre aux utilisateurs de publier des articles et de commenter ceux des autres.

L'application permet aux visiteurs sans compte de consulter les articles et les commentaires, mais ils ne peuvent pas en ajouter eux-mêmes. Les utilisateurs avec un compte peuvent ajouter des articles et des commentaires. Ils ont également la possibilité de modifier leur profil et de supprimer leur compte si nécessaire.

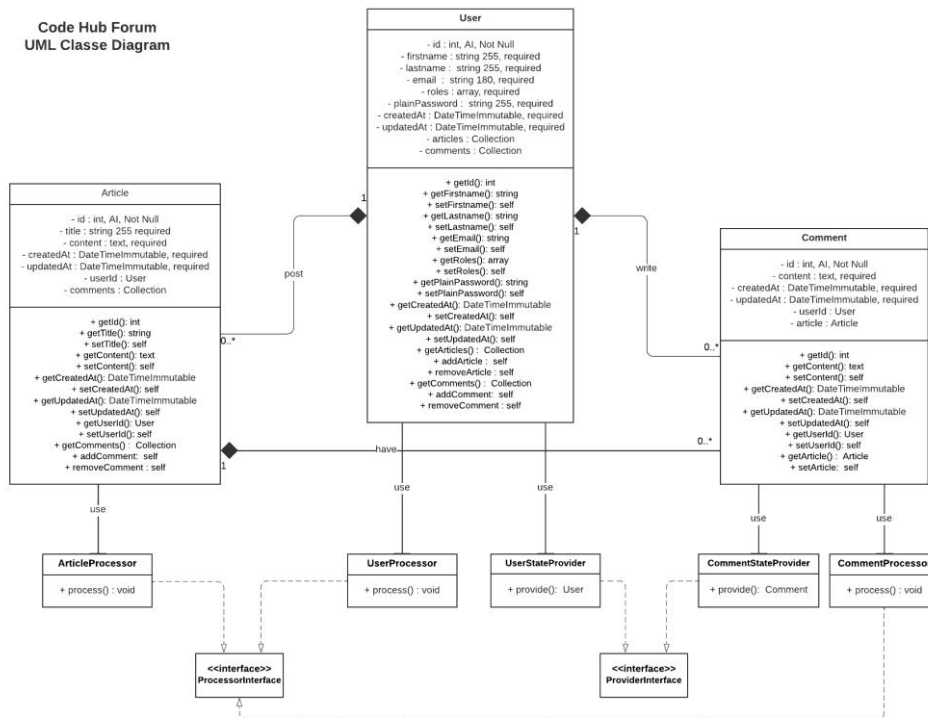
Les administrateurs ont les mêmes droits qu'un utilisateur standard, mais disposent également de fonctionnalités supplémentaires. Ils peuvent modifier, supprimer et ajouter des utilisateurs au forum. De plus, ils peuvent supprimer des articles ou des commentaires en cas de violation des règles du forum.

Les tâches réalisées :

- La première étape dans la réalisation d'un projet informatique est de construire une équipe solide de développeurs qui peuvent collaborer efficacement pour atteindre les objectifs du projet, nous avons constitué une équipe de quatre développeurs et utilisé une combinaison de méthodes Scrum et Kanban avec Trello pour suivre les tâches. Le projet a été divisé en petites tâches pour faciliter le travail et assurer une bonne gestion du temps. Nous avons également utilisé Github pour le suivi des versions et le partage du code.
- L'analyse des besoins est une étape cruciale dans la conception d'une application, car elle permet de déterminer les exigences et les attentes des utilisateurs. Dans le cadre de notre projet, nous avons reçu un cahier des charges de notre école qui a servi de base pour identifier les fonctionnalités à implémenter. Cette analyse nous a

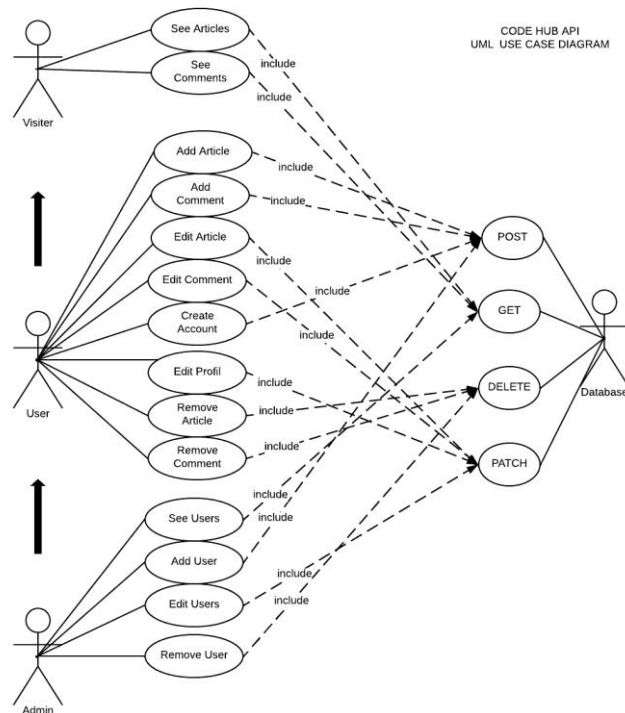
permis de comprendre les besoins des utilisateurs, d'anticiper les problématiques et d'élaborer une application adaptée.

- Création de l'UML Diagramme de classes : Le diagramme de classes UML permet de visualiser la structure de notre modèle de données et de faciliter la communication entre les membres de l'équipe de développement.



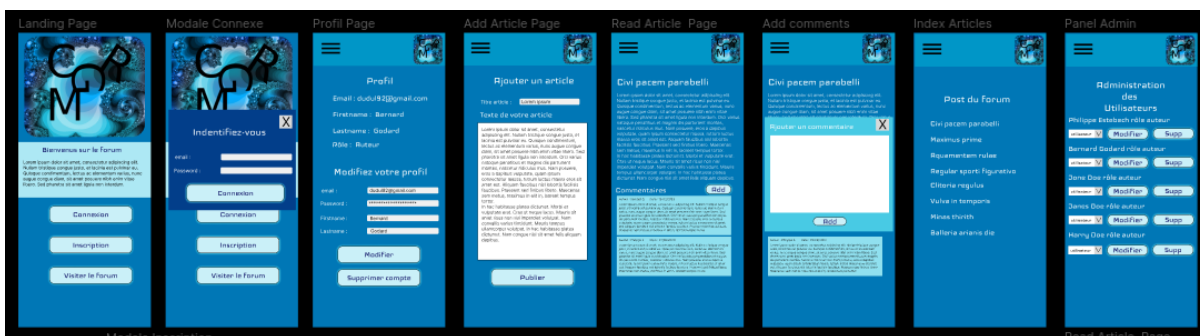
- Création de l'UML Diagramme de cas d'utilisation : permet de visualiser les fonctionnalités de notre application et de faciliter la compréhension des besoins des utilisateurs.

DOSSIER PROFESSIONNEL (DP)



- Développement de notre application mobile Code Hub organisée en quatre couches :

1. La couche de présentation : Cette couche comprend toutes les vues de notre application, c'est-à-dire les écrans que les utilisateurs voient et avec lesquels ils interagissent. Cette couche est constituée des écrans React Native de votre application. Les écrans permettent aux utilisateurs de voir les articles, les commentaires et les utilisateurs (que pour l'admin), de créer de nouveaux articles et commentaires, et de mettre à jour ou supprimer les articles et les commentaires existants.



2. La couche de logique d'application : Cette couche contient la logique métier de notre application, c'est-à-dire les règles qui déterminent comment les données sont manipulées et gérées. Cette couche est principalement gérée par l'API

Platform et Symfony, qui fournissent des fonctionnalités pour gérer les entités (User, Article, Comment) et effectuer des opérations CRUD (créer, lire, mettre à jour, supprimer) sur ces entités. La couche de logique d'application utilise également des services pour effectuer des tâches spécifiques, telles que l'authentification et l'autorisation des utilisateurs.

3. La couche d'accès aux données : Cette couche est responsable de la communication avec la source de données, dans notre cas la base de données. Cette couche est également gérée par l'API Platform et Symfony, qui utilisent Doctrine pour communiquer avec la base de données et mapper les données en entités PHP.
4. La couche de services : Cette couche comprend les différents services de notre application, qui fournissent des fonctionnalités spécifiques à différentes parties de notre application. Dans notre cas, nous avons des services pour gérer l'authentification des utilisateurs et un service qui appelle l'API OpenAi pour générer un article depuis son titre. Ces services sont également gérés par l'API Platform et Symfony.

► Les tests unitaires :

Les tests unitaires sont une étape importante dans le développement de toute application, car ils permettent de s'assurer du bon fonctionnement des différentes fonctionnalités de l'application, même lors de changements ou de mises à jour ultérieures. Dans le cadre de notre projet, nous avons effectué des tests unitaires sur les stateProvider et stateProcessor de vos entités user, article et comment, ainsi que sur les entités elles-mêmes, en utilisant l'objet mock de Symfony.

```
class UserStateProviderTest extends TestCase
{
    private $user;
    private $security;
    private $operation;
    private $entityManager;
    private $uriVariables = [];
    private $context = [];

    public function setUp(): void
    {
        $this->security = $this->createMock(Security::class);
        $this->entityManager = $this->createMock(EntityManagerInterface::class);
        $this->user = $this->createMock(User::class);
        $this->operation = $this->createMock(Operation::class);
    }

    public function testProvide()
    {
        $this->security->expects($this->once())->method('getUser')->willReturn($this->user);

        $provider = new UserStateProvider($this->security);

        $provider->provide($this->operation, $this->uriVariables, $this->context);
    }
}
```

► Déploiement de notre application :

Nous avons utilisé Plesk pour héberger notre API. Plesk est apprécié pour sa facilité d'utilisation, sa flexibilité et sa robustesse. Il est doté d'une interface graphique intuitive qui permet de gérer facilement les différents aspects du serveur Web.

2. Précisez les moyens utilisés :

- Visual Studio Code (VSCode) est un éditeur de code source gratuit.
- PhpStorm est un environnement de développement intégré (IDE) conçu spécifiquement pour le développement en PHP. J'ai utilisé PhpStorm pour créer l'API symfony car il offre une gamme de fonctionnalités avancées pour faciliter le processus de développement.
- Symfony et API Platform : Symfony est un framework PHP open source. API Platform est une extension de Symfony qui permet de créer des API web RESTful. Nous avons utilisé Symfony et API Platform pour créer l'API backend de l'application.
- React Native est une bibliothèque JavaScript open source développée par Facebook pour créer des applications mobiles multiplateformes pour iOS et Android en utilisant une seule base de code.
- GitHub : C'est une plateforme de développement logiciel qui permet de stocker et de partager du code source. Nous avons utilisé GitHub pour le versionnage du code source et la collaboration entre les membres de l'équipe.
- Trello : C'est un outil de gestion de projet en ligne basé sur la méthode Kanban. Nous avons utilisé Trello pour organiser les différentes tâches à effectuer pour réaliser le projet et pour suivre leur progression.
- Lucidchart : est un outil en ligne de création de diagrammes et de visualisations de données. Nous avons utilisé Lucidchart pour créer les diagrammes UML.

DOSSIER PROFESSIONNEL ^(DP)

3. Avec qui avez-vous travaillé ?

On était une équipe de quatre développeurs :

- Christophe Calmes
- Mathieu Ruiz
- Geoffrey Meca
- Rany Alo

4. Contexte

Nom de l'entreprise, organisme ou association ►

La PPlateforme

Chantier, atelier, service ►

Dans le cadre de notre formation en tant que
"Concepteur développeur d'application".

Période d'exercice ► Du : 02/01/2023 au : 28/02/2023

5. Informations complémentaires *(facultatif)*

DOSSIER PROFESSIONNEL ^(DP)

Titres, diplômes, CQP, attestations de formation

(facultatif)

Intitulé	Autorité ou organisme	Date
Master en Automatismes et Electronique Industrielle	Université d'Alep Syrie	2003
Certification OPQUAST Maîtrise de la Qualité en Projets Web	Opquast (705/1000)	2021
Certification des Méthodes Agiles de Gestion	Simplon	2021
Titre pro Développeur web et web mobile	Ministère du travail	2022

DOSSIER PROFESSIONNEL ^(DP)

Déclaration sur l'honneur

Je soussigné(e) Rany Alo ,
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 Marseille le 20/05/2023

pour faire valoir ce que de droit.

Signature :

DOSSIER PROFESSIONNEL ^(DP)

Documents illustrant la pratique professionnelle

(facultatif)

Intitulé
Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

ANNEXES

(Si le RC le prévoit)