
Dossier Professionnel (DP)

Nom de naissance : LABOUDIGUE
Nom d'usage :
Prénom : Quentin
Adresse : 1 Rue Marie Volait, 38700 La Tronche

Titre professionnel visé

Concepteur développeur d'applications

Modalité d'accès :

- 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.

Dossier Professionnel (DP)

Sommaire

Exemples de pratique professionnelle

Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

- Maquetter une application
- Développer une interface utilisateur de type desktop
- Développer des composants d'accès aux données
- Développer la partie front-end d'une interface utilisateur web
- Développer la partie back-end d'une interface utilisateur web

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

- Concevoir une base de données
- Mettre en place une base de données
- Développer des composants dans le langage d'une base de données

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

- Collaborer à la gestion d'un projet informatique et à l'organisation de l'environnement de développement
- Concevoir une application
- Développer des composants métier
- Construire une application organisée en couches
- Développer une application mobile
- Préparer et exécuter les plans de tests d'une application
- Préparer et exécuter le déploiement d'une application

Dossier Professionnel (DP)

Activité type n°1 : Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Compétence n°1 : Maquetter une application

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

Le projet « Raoul » consistait à développer une application mobile en utilisant la technologie « React Nativ ». Le projet a été scindé en 2 parties, la partie back-end faite de l'api et la partie front-end qui consommait ce même api sur mobile. La première étape à partir du cahier des charges client a été de retranscrire sous forme d'écrans le parcours utilisateur souhaité et ensuite de maquetter l'ensemble. Les maquettes ont eu comme fonction de s'accorder sur le travail en développement à effectuer côté « front-end » et ensuite de répartir la charge de travail entre les différentes personnes dans l'équipe.

2. Précisez les moyens utilisés :

Les moyens utilisés pour maquetter l'application peuvent se résumer à un **brouillon** pour dégrossir le parcours utilisateur et ensuite **Figma**, la web application permettant un travail collaboratif pour la production de maquettes. L'ensemble des tâches ont été formalisées comme des « Issues » dans **Gitlab**.

3. Avec qui avez-vous travaillé ?

La travail a été effectué par l'équipe « Front » du projet Raoul composée de 6 personnes de la promotion CDA 2021/2022 de Simplon Clermont.

4. Contexte

Nom de l'entreprise, organisme, ou association Formation CDA Clermont

Chantier, atelier, service Projet de développement d'une application mobile

Période d'exercice Du 06/06/2022 au 10/06/2022

Dossier Professionnel (DP)

5. Informations complémentaires (facultatif)

Login

0- Créez un événement...

1 - Liste des bars

1 - Pay as host

Register

0 - Calendrier

1 - Détail événement...

LE CHAT

Dossier Professionnel (DP)

Activité type n°1 : Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Compétence n°2 : Développer une interface utilisateur de type desktop

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

Le projet consistait à créer une application desktop basée sur le jeu du « **C'est plus c'est moins** » où l'utilisateur cherche à deviner un nombre mystère généré aléatoirement. La première étape a été la création du projet Electron à l'aide du « quickstart » officiel, qui comprenait les instructions suivantes :

- npm init
- npm install --save-dev electron

J'ai ensuite configuré le fichier package.json pour y ajouter la commande « start » d'Electron. J'ai ensuite configuré le point d'entrée « main.js » pour finalement commencer le développement :

- De la partie HTML
- De la logique en Javascript

2. Précisez les moyens utilisés :

Pour réaliser ce projet, j'ai utilisé la **documentation officielle d'Electron** et les instructions de création d'un projet. L'application a été développée en HTML / CSS / JAVASCRIPT et j'ai donc eu l'occasion de résoudre certaines impasses dans le développement à l'aide de **Stackoverflow**. Le projet étant collaboratif, **Gitlab** a été utilisé tout au long de la semaine.

3. Avec qui avez-vous travaillé ?

L'équipe était composée de 4 personnes de la promotion CDA 2021/2022 de Simplon Clermont.

4. Contexte

Nom de l'entreprise, organisme, ou association Formation CDA Clermont

Chantier, atelier, service N/A

Période d'exercice Du 09/05/2022 au 13/05/2022

Dossier Professionnel (DP)

5. Informations complémentaires (facultatif)

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <!-- https://developer.mozilla.org/en-US/docs/Web/HTTP/CSP -->
6     <meta http-equiv="Content-Security-Policy" content="default-src 'self'; script-src 'self'">
7     <meta http-equiv="X-Content-Security-Policy" content="default-src 'self'; script-src 'self'">
8     <link href=".//styles.css" rel="stylesheet">
9     <title>Hello World!</title>
10    </head>
11    <body>
12      <h1>Guess the number</h1>
13      <p id="tentatives"> Nombre de tentatives : <img id="trial"> </p>
14      <input type="number" id="guess" required min="0" max="50">
15      <button id="verify"> Vérifier </button>
16      <ul id="indications">
17
18      </ul>
19      <p id="result"> </p>
20      <img id="imageVincent">
21      <button id="reload"> Relancer une partie </button>
22      <video id="myVideo" width="420">
23        <source src="audio/JustPrix.mp4" type="video/mp4">
24      </video>
25      <script src=".//game.js"></script>
26    </body>
27  </html>
```

HTML du projet

```
function getRandomInt(max) {
    return Math.floor(Math.random() * max);
}

function checkGuess() {
    if (CheckInputValue(guessInput, guess, trialSpan)){
        ChangeAsset()
        var indications = document.createElement('li')
        if (guess == 0) {
            EndGame(indications, resultP ,checkButton)
        }
        if (!DisplayClue(guessInput.value, randomNumber, indications)){
            EndGame(indications, resultP ,checkButton, true)
        }
        indicationsUl.appendChild(indications) ;
    }
}

function reloadGame() {
    guess = 5
    trialSpan.innerText = guess
    randomNumber = getRandomInt(50)
    console.log(randomNumber)
    checkButton.disabled = false
    indicationsUl.innerText = ""
    guessInput.value = 0
    resultP.innerText = ""
    video.style.display = "none"
    image.src=''
```

Extrait de la logique en javascript du projet

Dossier Professionnel (DP)

Activité type n°1 : Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Compétence n°3 : Développer des composants d'accès aux données

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

Le projet consistait à créer un API mettant à disposition des informations sur les pays du monde, ainsi qu'une page HTML permettant de consommer l'API et d'effectuer quelques opérations du CRUD (Création d'un pays à l'aide d'un formulaire, affichage de la liste des pays, etc). La technologie utilisée était Java associée au framework Springboot. L'application possède donc un back-end et un front-end.

L'accès aux données se construit à partir du modèle « Country » sur lequel on vient connecter un repository Springboot. Un fichier 'MainController.java' va servir ensuite à détailler l'ensemble des routes permettant l'exécution des opérations du CRUD.

2. Précisez les moyens utilisés :

Documentation officielle de Java.

Documentation officielle de Springboot.

3. Avec qui avez-vous travaillé ?

Le travail a été réalisé par équipe de 2.

4. Contexte

Nom de l'entreprise, organisme, ou association

Simplon, CDA Clermont

Chantier, atelier, service N/A

Période d'exercice Du 20/12/2021 au 24/12/2021

Dossier Professionnel (DP)

5. Informations complémentaires (facultatif)

```
@Entity
public class Country {
    @Id
    @GeneratedValue(strategy=GenerationType.AUTO)
    private Integer id;

    private String name;

    private String code;

    public Integer getId() {
        return id;
    }

    public void setId(Integer id) {
        this.id = id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getCode() {
        return code;
    }

    public void setCode(String code) {
        this.code = code;
    }
}
```

Création du modèle « Country »

Dossier Professionnel (DP)

```
@Controller
@RequestMapping(path="")
public class MainController {
    @Autowired // This means to get the bean called userRepository
    // Which is auto-generated by Spring, we will use it to handle the data
    private CountryRepository countryRepository;

    @PostMapping(path="/country") // Map ONLY POST Requests
    public ResponseEntity<Country> addNewCountry (@RequestParam String name
        , @RequestParam String code) {

        Country n = new Country();
        n.setName(name);
        n.setCode(code);
        countryRepository.save(n);
        return new ResponseEntity<>(n, HttpStatus.CREATED);
    }

    @GetMapping(path="/countries")
    public @ResponseBody Iterable<Country> getAllCountries() {
        // This returns a JSON or XML with the users
        return countryRepository.findAll();
    }
}
```

Exemple de la création de plusieurs composants d'accès aux données permettant d'ajouter un pays (via POST) et d'afficher la liste des pays (via GET)

Dossier Professionnel (DP)

Activité type n°1 : Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Compétence n°4 : Développer la partie front-end d'une interface utilisateur web

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

Le projet consistait à créer un jeu de « morpion » en utilisant la librairie React. Après l'initialisation du projet à l'aide de la commande « npm Create react app », j'ai commencé par créer les différents composants de l'interface utilisateur :

- « Box » qui définit une case cliquable (affichant un rond ou une croix)
- « Board », le plateau de Jeu, lui même composé de « Box »
- « Game » ou « laGame », qui va gérer une partie soit contre un humain soit contre une intelligence artificielle.
- Enfin « Manager », le composant qui va gérer la suite de partie et l'historique des scores.

La logique de conception a été d'associer à chaque composant des variables d'états qui correspondaient au périmètre d'action du composant. « useState » a été massivement utilisé, permettant la mise à jour de l'interface graphique en fonction des actions de l'utilisateur.

2. Précisez les moyens utilisés :

Les moyens utilisés pour réaliser ce projet ont été :

- Des tutoriels sur YouTube
- Des recherches sur Stackoverflow

En terme d'outils, VS code et sa fonctionnalité Live share a été utilisée tout au long du projet afin de développer de manière collaborative.

3. Avec qui avez-vous travaillé ?

J'ai travaillé en Duo avec une personne de la promotion CDA Clermont de Simplon. Le travail n'a pas été scindé et réparti mais plutôt collaboratif notamment en utilisant la fonctionnalité Live Coding de VS code.

Dossier Professionnel (DP)

4. Contexte

Nom de l'entreprise, organisme, ou association

CDA Clermont Simplon

Chantier, atelier, service N/A

Période d'exercice Du 04/07/2022 au 08/07/2022

5. Informations complémentaires (facultatif)

```
function Game({ onGameOver, incrementGame }) {
  const [history, setHistory] = useState([Array(9).fill(null)]);
  const [stepNumber, setStepNumber] = useState(0);
  const [xIsNext, setXIsNext] = useState(true);
  const winner = calculateWinner(history[stepNumber]);
  const x0 = xIsNext ? "X" : "0";

  // Handle box click
  const handleClick = (i) => {
    const historyPoint = history.slice(0, stepNumber + 1);
    const current = historyPoint[stepNumber];
    const boxes = [...current];
    // Return if won or occupied
    if (winner || boxes[i]) return;
    // Set box
    boxes[i] = x0;
    setHistory([...historyPoint, boxes]);
    setStepNumber(historyPoint.length);
    setXIsNext(!xIsNext);
    // Trigger onGameOver
    if (calculateWinner(boxes) != null) {
      onGameOver(xIsNext);
    }
  };
}
```

Composant « Game » (partie 1)

Dossier Professionnel (DP)

```
// Reload current game (empty board)
const reloadGame = () => {
  setHistory([Array(9).fill(null)]);
  setStepNumber(0);
  setXIsNext(true);
  incrementGame();
};

// Display current game information
const displayInfo = () => {
  if (stepNumber == 9 && !winner) {
    return <h3>Well, it's a tie!</h3>;
  } else if (winner) {
    return <h3>{"Winner: " + winner}</h3>;
  } else {
    return <h3>{"Next Player: " + x0}</h3>;
  }
};

// Render
return (
  <div>
    <Board boxes={history[stepNumber]} onClick={handleClick} style="board"/>
    <div className="info-wrapper">
      {displayInfo()}
      <button className="reload-button" onClick={reloadGame}>
        {" "}
        New game{" "}
      </button>
    </div>
  </div>
);
```

Composant « Game » (partie 2)

On distingue la définition des différentes fonctions exécutées dans le composant et le rendu du composant avec la présence du mot clé « return ».

Dossier Professionnel (DP)

Activité type n°1 : Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Compétence n°5 : Développer la partie back-end d'une interface utilisateur web

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

Le projet consistait à créer un API mettant à disposition des informations sur les pays du monde, ainsi qu'une page HTML permettant de consommer l'API et d'effectuer quelques opérations du CRUD (Création d'un pays à l'aide d'un formulaire, affichage de la liste des pays, etc). La technologie utilisée était Java associée au framework Springboot. L'application possède donc un back-end et un front-end.

L'API produit présentait un ensemble de « endpoints » accessibles via des routes prédéterminées et permettait une liste d'opérations du CRUD sur une table contenant des pays et notamment :

- L'affichage de tous les pays
- La mise à jour d'un pays.
- L'affichage d'un pays unique.
- La suppression d'un pays.
- La création d'un nouveau pays.

2. Précisez les moyens utilisés :

Documentation officielle de Java.

Documentation officielle de Springboot.

3. Avec qui avez-vous travaillé ?

Le projet a été réalisé par un groupe de 2.

Dossier Professionnel (DP)

4. Contexte

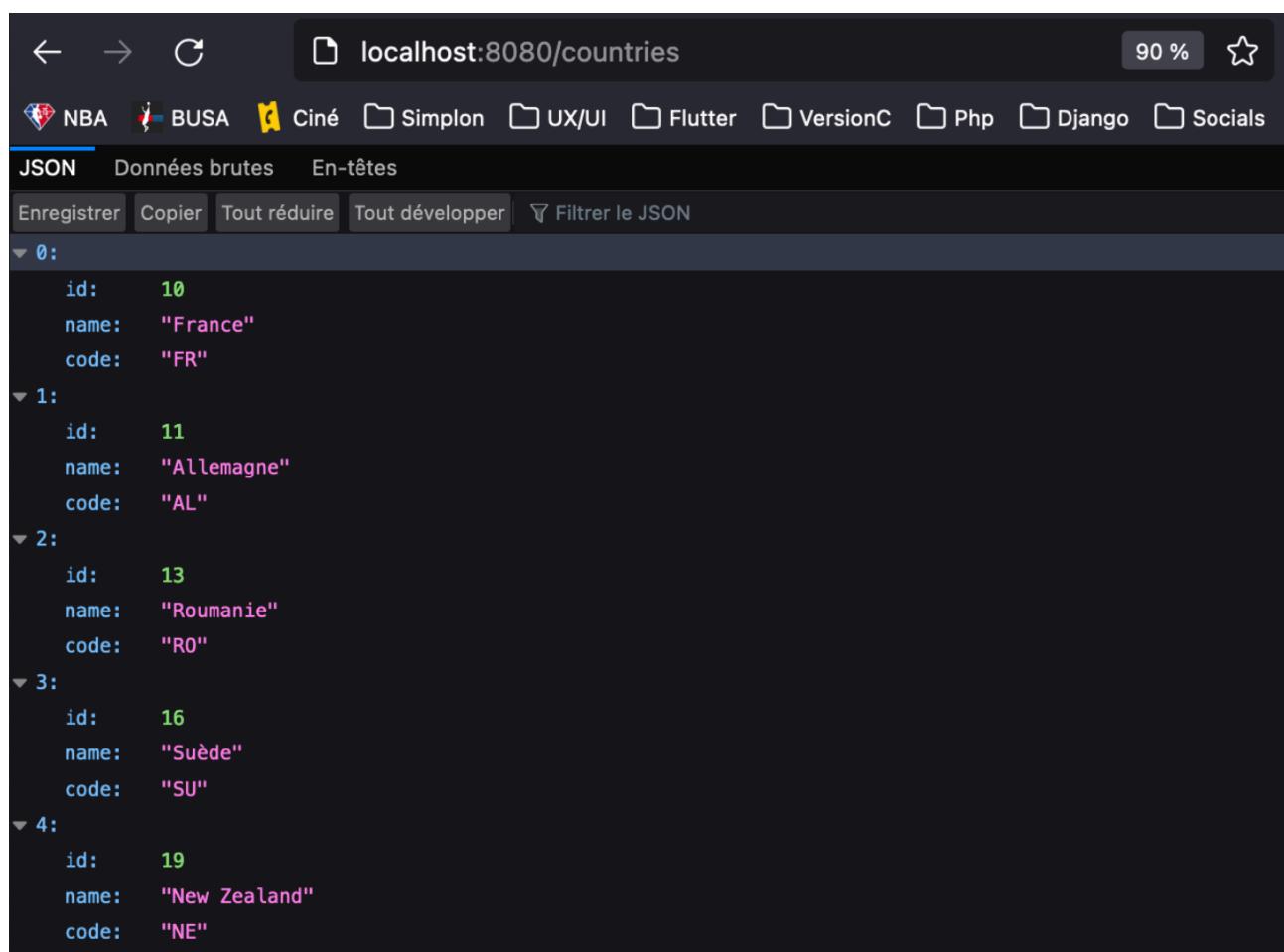
Nom de l'entreprise, organisme, ou association

CDA, Simplon Clermont

Chantier, atelier, service N/A

Période d'exercice Du 20/12/2021 au 24/12/2021

5. Informations complémentaires (facultatif)



The screenshot shows a browser developer tools JSON viewer with the URL `localhost:8080/countries`. The JSON response contains a list of five countries:

```
[{"id": 10, "name": "France", "code": "FR"}, {"id": 11, "name": "Allemagne", "code": "AL"}, {"id": 13, "name": "Roumanie", "code": "RO"}, {"id": 16, "name": "Su\u00e8de", "code": "SU"}, {"id": 19, "name": "New Zealand", "code": "NE"}]
```

Réponse de l'API avec l'affichage de la liste des pays

Dossier Professionnel (DP)

Activité type n°2 : Concevoir et développer la persistance des données en intégrant les recommandations de sécurité

Compétence n°6 : Concevoir une base de données

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

Le projet « Raoul » consistait à développer une application mobile en utilisant la technologie « React Nativ ». Le projet a été scindé en 2 parties, la partie back-end faite de l'api et la partie front-end qui consommait ce même api sur mobile. L'étape préliminaire à la création de l'API a été la conception de la base données via un diagramme de classe. Le but étant de formaliser les classes utilisés dans l'application et leurs interactions.

2. Précisez les moyens utilisés :

Lucid Chart a été utilisé pour la création du diagramme de classe ainsi que Discord pour la mise en place d'un travail collaboratif.

3. Avec qui avez-vous travaillé ?

Ce projet a été réalisée avec une équipe de 5 personnes.

4. Contexte

Nom de l'entreprise, organisme, ou association

CDA, Simplon, Clermont

Chantier, atelier, service N/A

Période d'exercice Du 09/05/2022 au 13/05/2022

Dossier Professionnel (DP)

5. Informations complémentaires (facultatif)

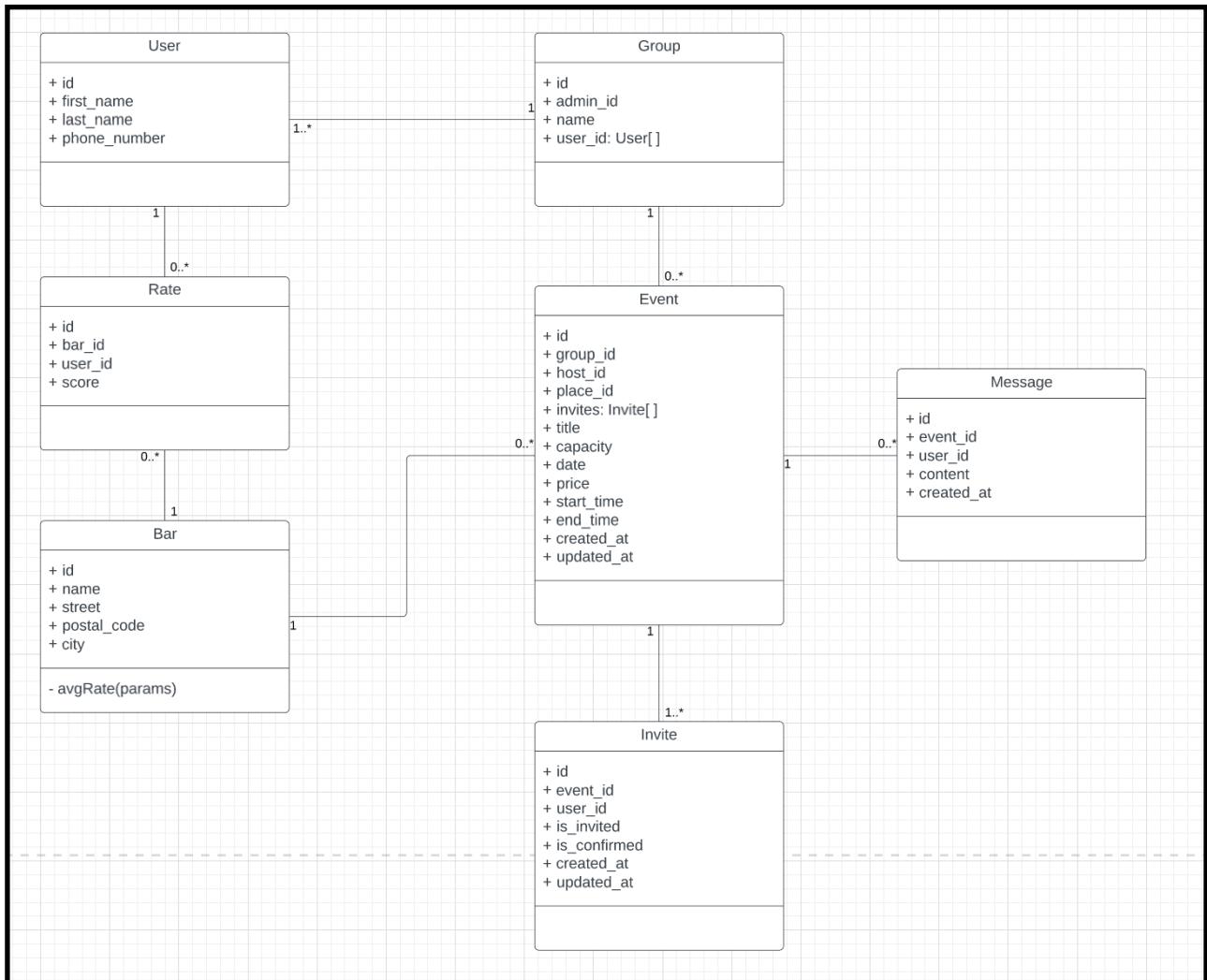


Diagramme de classe présentant les entités et leurs relations de dépendance.

Dossier Professionnel (DP)

Activité type n°2 : Concevoir et développer la persistance des données en intégrant les recommandations de sécurité

Compétence n°7 : Mettre en place une base de données

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

Le projet consistait à créer un API mettant à disposition des informations sur les pays du monde, ainsi qu'une page HTML permettant de consommer l'API et d'effectuer quelques opérations du CRUD (Création d'un pays à l'aide d'un formulaire, affichage de la liste des pays, etc). La technologie utilisée était Java associée au framework Springboot. L'application possède donc un back-end et un front-end.

La mise en place de la base données s'effectue dans un fichier de configuration Springboot en fonction de type de base de données que l'on choisit. C'est le fichier « application.properties ».

```
J application.properties ×  
src > main > resources > J application.properties  
1   spring.jpa.hibernate.ddl-auto=update  
2   spring.datasource.url=jdbc:mysql://localhost:3306/db_example  
3   spring.datasource.username=root  
4   spring.datasource.password=password  
5   spring.datasource.driver-class-name =com.mysql.jdbc.Driver  
6   spring.jpa.database-platform=org.hibernate.dialect.MySQL5InnoDBDialect
```

2. Précisez les moyens utilisés :
-

Documentation officielle de Springboot.

Documentation officielle de Java.

3. Avec qui avez-vous travaillé ?
-

Projet réalisé en duo.

Dossier Professionnel (DP)

4. Contexte

Nom de l'entreprise, organisme, ou association

CDA Simplon, Clermont

Chantier, atelier, service N/A

Période d'exercice Du 09/05/2022 au 13/05/2022

5. Informations complémentaires (facultatif)

Dossier Professionnel (DP)

Activité type n°2 : Concevoir et développer la persistance des données en intégrant les recommandations de sécurité

Compétence n°8 : 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 :
-

Le projet consistait à développer un API Rest pour une structure de donnée semblable à celle d'un blog avec des utilisateurs (Users) pouvant écrire des messages (Posts). Le cahier des charges indique que chaque message possède un auteur et un sujet (Topic). Les sujets étant eux mêmes regroupés par catégorie (Category).

Afin d'accéder aux données sur les Users, j'ai développé la classe UserDao (pour Data Access Object) permettant de déclencher des requêtes SQL à la base de données du projet. Ainsi pour récupérer un seul utilisateur dans notre base de données, on a la méthode suivante :

```
// SINGLE USER READ
public function getSingleUser($value, $key = NULL) {

    if(is_null($key)){
        $key = $this->primaryKey;
    }

    $sqlQuery = "select * from {$this->tableName} where {$key}='{$value}'";

    $stmt = $this->conn->prepare($sqlQuery);
    $stmt->bindParam(1, $this->id);
    $stmt->execute();

    $rows = $stmt->fetch(PDO::FETCH_ASSOC);

    return $rows;
}
```

Dossier Professionnel (DP)

2. Précisez les moyens utilisés :

Le projet a été développé utilisant PHP sans ORM.

Ressources vidéo sur Youtube.

StackOverflow.

Tutoriels.

3. Avec qui avez-vous travaillé ?

Projet individuel.

4. Contexte

Nom de l'entreprise, organisme, ou association

CDA Simplon Clermont

Chantier, atelier, service N/A

Période d'exercice Du 15/08/2021 au 19/08/2021

5. Informations complémentaires (facultatif)

```
// GET ALL TOPICS
public function getTopics(){
    $sqlQuery = "SELECT id, title, category_id, topic_admin_id FROM " . $this->tableName . "";
    $stmt = $this->conn->prepare($sqlQuery);
    $stmt->execute();
    return $stmt;
}
```

Exemple d'une requête SQL pour afficher les « topics » de notre base de données.

Dossier Professionnel (DP)

Activité type n°3 : Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

Compétence n°9 : Collaborer à la gestion d'un projet informatique et à l'organisation de l'environnement de développement

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

Le projet « Raoul » consistait à développer une application mobile. L'équipe projet a été scindée en 3 groupes :

- Un groupe « Back-end » responsable de la création de l'Api
- Un groupe « Front-end » responsable du développement de l'application mobile (consommant l'api mentionné précédemment)
- Un duo dans la gestion du projet constitué d'une personne responsable des échanges avec le client et de moi-même, responsable de la coordination entre les développeurs.

Assimilable à un rôle de lead-développeur, j'ai donc commencé par organiser une veille sur les technologies disponibles pour le développement d'une application mobile. Les développeurs, par équipe de 2, ont du présenter une technologie en précisant les avantages et les points faibles. À l'issue de cette veille, et compte-tenu des contraintes du cahier des charges (ampleur du projet et temps disponible), la technologie React Nativ a été choisie comme un commun accord avec l'ensemble de l'équipe.

Les développeurs se sont réparties dans les équipes Back et Front, et les tâches ont ensuite été assignées en utilisant les issues dans GitLab. Chaque « issue » s'est vu désignée un développeur et une échéance. Une fois réalisée, la fonction développée faisait l'objet d'une « marge request ».

Une dernière phase de travail collaboratif a consisté à assembler les différentes parties réalisées précédemment, pour faire fonctionner l'application selon les livrables attendus.

2. Précisez les moyens utilisés :

- Des moyens de communications pour coordonner les interactions entre les personnes dans chacun des groupes tels que **Google meet** et **Discord**
- **Gitlab** pour le « versioning », le partage et la mise en commun du travail effectué par chacun.
- Les « **issues** » Gitlab pour formaliser et attribuer les tâches aux développeurs.

Dossier Professionnel (DP)

3. Avec qui avez-vous travaillé ?

L'équipe projet était constituée de l'ensemble de la promotion CDA Simplon 2021/2022 de Clermont Ferrand.

4. Contexte

Nom de l'entreprise, organisme, ou association Simplon CDA Clermont-Ferrand

Chantier, atelier, service N/A

Période d'exercice Du 15/04/2022 au 31/05/2022

5. Informations complémentaires (facultatif)

The screenshot shows a list of issues from a GitLab project. The interface includes a header with filters for 'Open' (12), 'Closed' (1), and 'All' (13), and buttons for 'Edit issues' and 'New issue'. A search bar and a date filter for 'Created date' are also present. The list contains the following items:

- Rendre fonctionnel authentication
#13 · created 1 month ago by Clémentine Fouqué (Rendu) Jun 9, 2022 P1 · updated 1 month ago
- Intégrer un composant Map dans une page
#12 · created 1 month ago by Laboudigue Jun 9, 2022 P1 · updated 1 month ago
- Création d'événements
#11 · created 1 month ago by Laboudigue Jun 9, 2022 P1 · updated 1 month ago
- Développer les fonctions partagées des calls API
#10 · created 1 month ago by Laboudigue Jun 8, 2022 P1 · updated 1 month ago
- Intégration page profil
#9 · created 1 month ago by Clémentine Fouqué (Présentation client) Jun 8, 2022 P1 · updated 1 month ago
- Intégration page notation bar
#8 · created 1 month ago by Clémentine Fouqué (Présentation client) Jun 8, 2022 P1 · updated 1 month ago
- Intégration page afterwork
#7 · created 1 month ago by Clémentine Fouqué (Présentation client) Jun 8, 2022 P1 · updated 1 month ago
- Intégration formulaires authentification
#5 · created 1 month ago by Clémentine Fouqué (Présentation client) P1 · updated 1 month ago

Liste des « issues » du projet sur Gitlab

Dossier Professionnel (DP)

Activité type n°3 : Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

Compétence n°10 : Concevoir une application

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

Le projet concernait la conception d'une application de gestion des fabriques (unité pédagogique) chez Simplon. À partir du cahier des charges, le travail à consisté à modéliser le fonctionnement de l'application via la production d'un diagramme de cas d'usages, d'un diagramme de classe et de plusieurs diagrammes d'activité. L'application dont j'étais en charge mettait en jeu des utilisateurs dans un module post-formation visant à assurer le suivi et la gestion des anciens apprenants étant passés par une fabrique.

- Les diagrammes de cas d'usage ont été rédigés en sa basant sur la liste des besoins fonctionnels présente dans le cahier des charge.
- Le diagramme de classe représente l'ensemble des entités, classes qui vont être utilisées par le système.
- Les diagrammes d'activités reprennent chaque cas d'usage et détaillent leur exécution par le système.

2. Précisez les moyens utilisés :

« Lucid chart » a été l'outil principal utilisé dans ce projet pour la production de tous les diagrammes de manière collaborative.

3. Avec qui avez-vous travaillé ?

Ce projet a été réalisé en groupe avec 3 autres personnes de la promotion CDA Simplon Clermont-Ferrand.

4. Contexte

Nom de l'entreprise, organisme, ou association

Simplon, promotion CDA Clermont

Chantier, atelier, service N/A

Dossier Professionnel (DP)

Période d'exercice

Du 22/11/2021 au 26/11/2021

5. Informations complémentaires (facultatif)

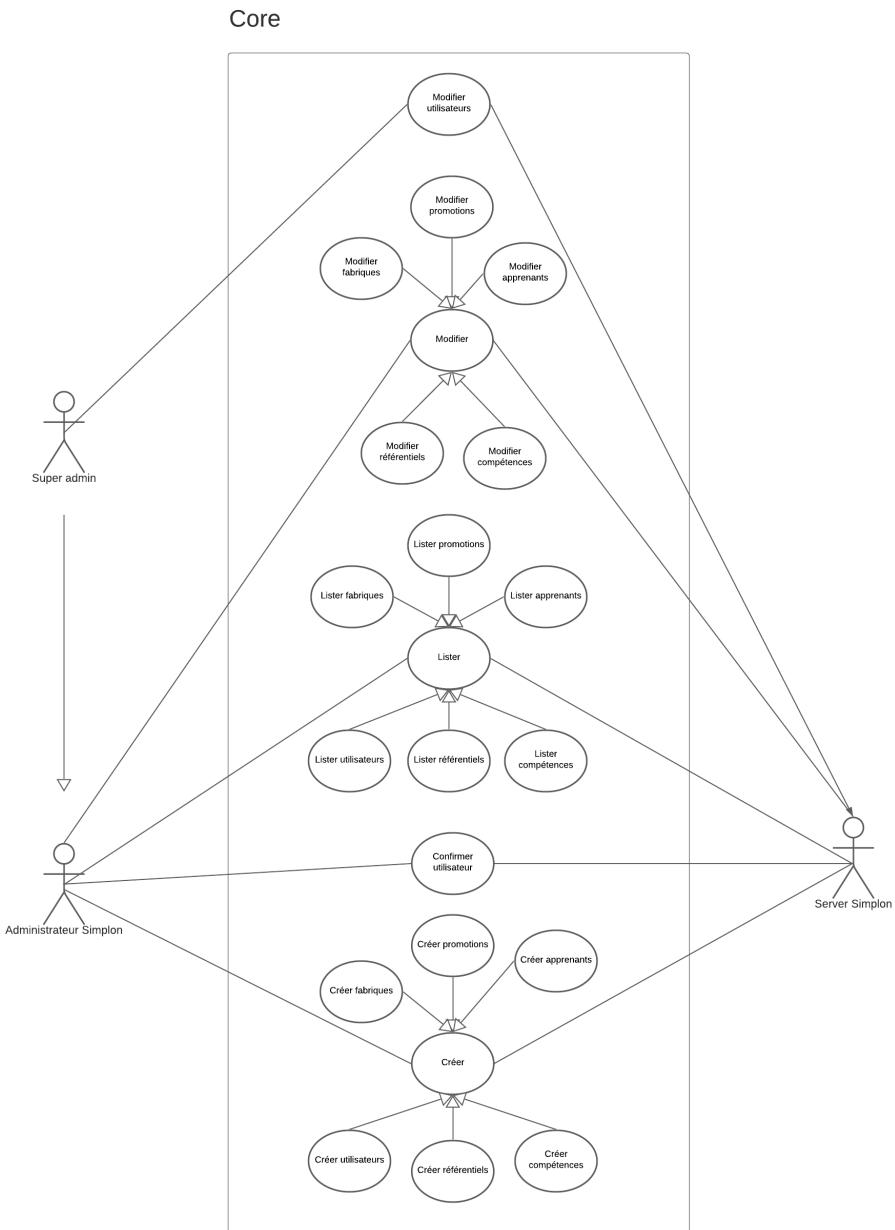


Diagramme de cas d'usage

Dossier Professionnel (DP)



Diagramme de classe

Dossier Professionnel (DP)

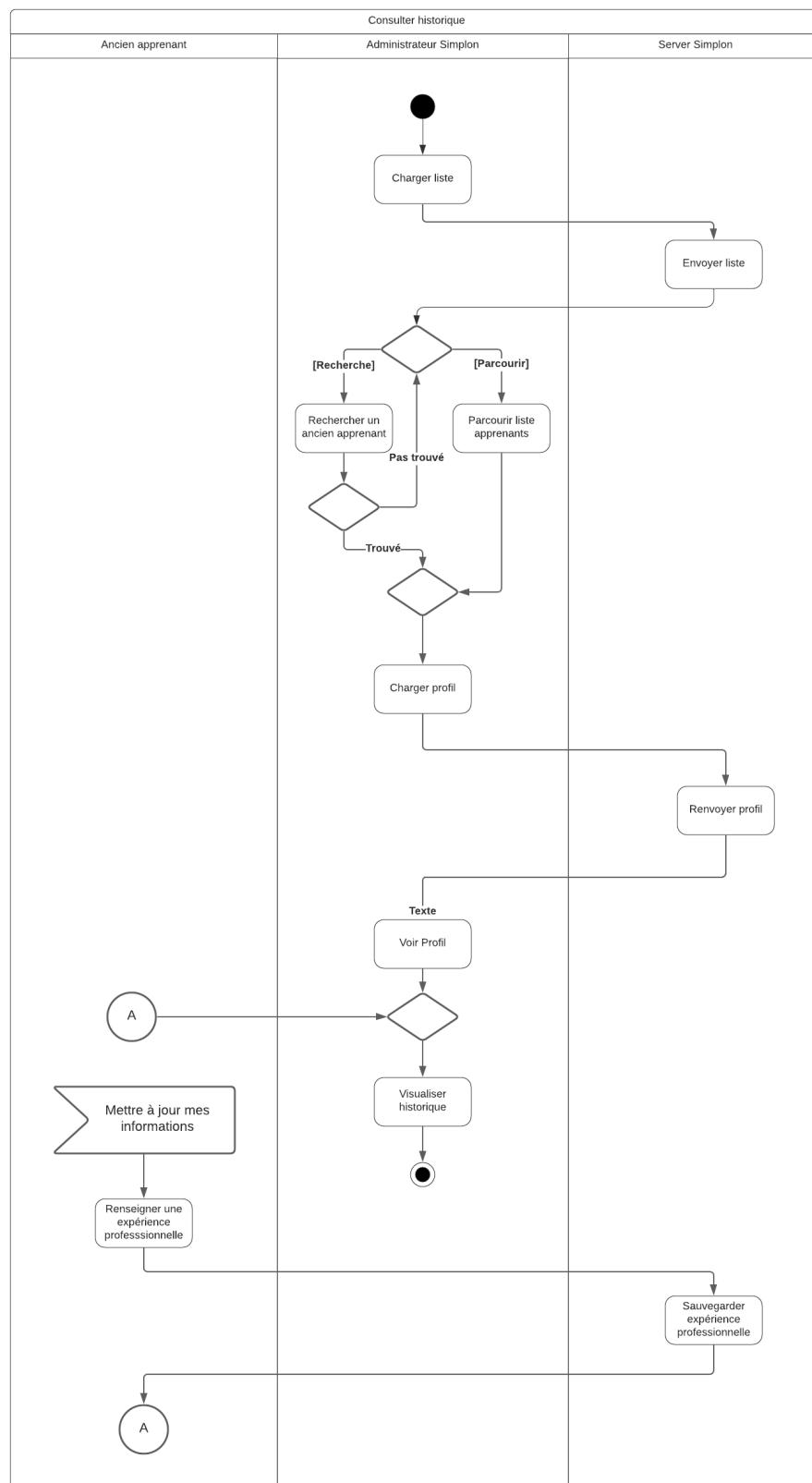


Diagramme d'activité correspondant au cas d'usage : Consulter historique

Dossier Professionnel (DP)

Activité type n°3 : Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

Compétence n°11 : Développer des composants métier

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

Le projet consistait à développer une application permettant une gestion du traitement des déchets dans une ville en fonction de leur type (Papier, verre, plastiques) et des capacités des différentes machines pouvant les traiter. L'application devait aussi permettre un calcul du CO2 généré par type de déchet

Le fichier d'entrée était formaté en Json, j'ai dû apprendre à intégrer un fichier de données dans une application Flutter et à le déserialiser. La difficulté majeure du projet résidait dans la création des modèles afin de représenter de manière optimale les déchets et leur type ainsi que les machines et leur capacité à traiter tel ou tel déchet. J'ai dû ainsi mettre en place des classes abstraites et des extensions pour modéliser correctement le problème.

Une fois les modèles verrouillés, j'ai créé une classe « Manager » permettant d'instancier le problème et de procéder aux calculs d'attribution des déchets dans les différentes machines ainsi que la quantité de CO2 créée dans l'opération de recyclage globale.

2. Précisez les moyens utilisés :

Les principes de la programmation orienté objet.

Des ressources sur YouTube.

Documentation officielle de Flutter.

3. Avec qui avez-vous travaillé ?

Projet individuel.

Dossier Professionnel (DP)

4. Contexte

Nom de l'entreprise, organisme, ou association

Simplon, CDA Clermont

Chantier, atelier, service N/A

Période d'exercice Du 27/09/2021 au 01/10/2021

5. Informations complémentaires (facultatif)

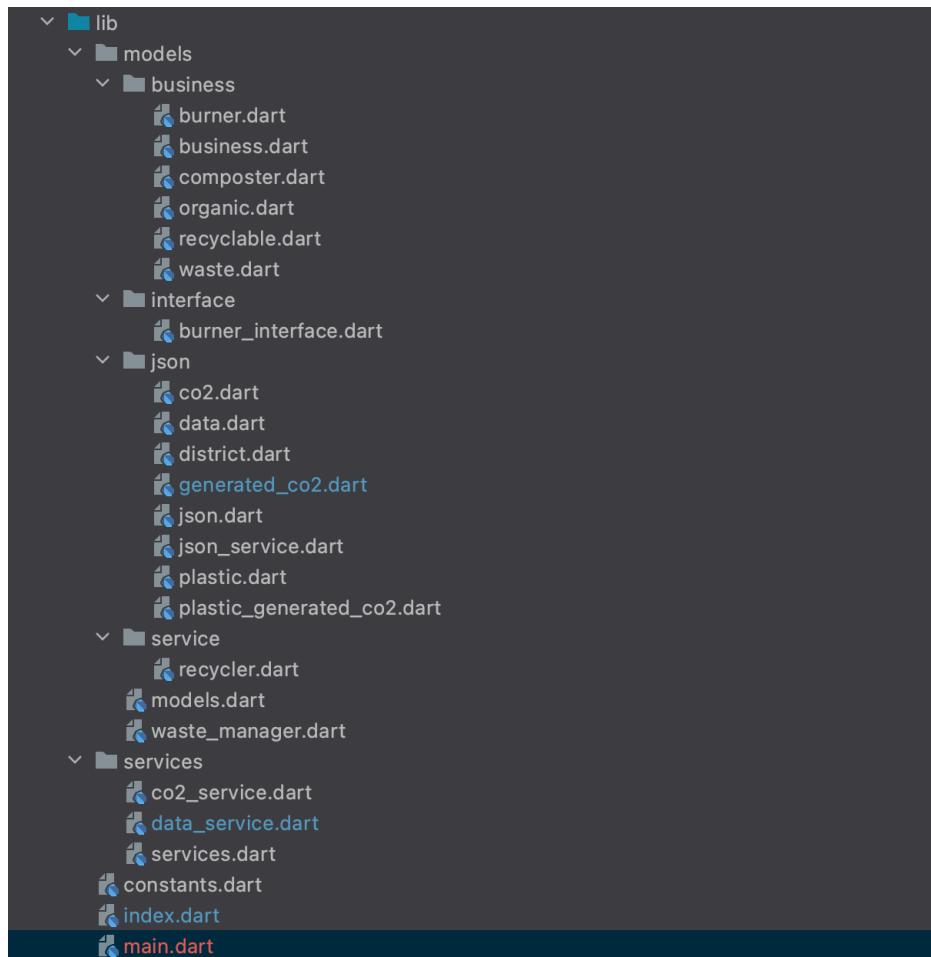
```
{
  "quartiers": [
    {
      "population": 76638,
      "plastiques": {
        "PET": 2053.9,
        "PVC": 1232.34,
        "PC": 1643.12,
        "PEHD": 2464.68
      },
      "papier": 8626.37,
      "organique": 12323.39,
      "verre": 5340.14,
      "metaux": 1643.12,
      "autre": 98587.12
    },
    {
      "population": 43824,
      "plastiques": {
        "PET": 469.79,
        "PVC": 1174.48,
        "PC": 784.69,
        "PEHD": 1174.48
      },
      "papier": 4463.04,
      "organique": 5872.42,
      "verre": 3288.55,
      "metaux": 939.59,
      "autre": 56375.19
    },
    {
      "population": 45802,
      "plastiques": {
        "PET": 736.5,
        "PVC": 2454.99,
        "PC": 736.5,
        "PEHD": 981.99
      },
      "papier": 4909.97,
      "organique": 6628.47,
      "verre": 2780.49,
      "metaux": 981.99,
      "autre": 54009.72
    }
  ],
}
```

Liste des quantités de déchets récupérés par quartier



Co2 généré par type de déchet en fonction de la méthode de traitement choisie.

Dossier Professionnel (DP)



Structure générale du projet avec la définition des classes dans le dossier « models », les « services » permettant l’interprétation des données du problème à partir du fichier json. Enfin « main.dart » constitue le point d’entrée et permet d’instancier « waste_manager » dont la fonction est de contenir la partie algorithmique de l’application.

Dossier Professionnel (DP)

Activité type n°3 : Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

Compétence n°12 : Construire une application organisée en couches

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

Le projet consistait à créer un API mettant à disposition des informations sur les pays du monde, ainsi qu'une page HTML permettant de consommer l'API et d'effectuer quelques opérations du CRUD (Création d'un pays à l'aide d'un formulaire, affichage de la liste des pays, etc). La technologie utilisée était Java associée au framework Springboot. L'application possède donc un back-end, un front-end et s'organise en couche.

- La couche de présentation (front-end) est assurée par la page HTML
- La couche d'exposition et la couche d'accès aux données (back-end) sont gérées par le MainController.java et les annotations Springboot qui permettent d'ouvrir les « endpoints » de l'api et d'y accéder via des routes précises.
- La couche de données est assurée par une base de données mySQL

2. Précisez les moyens utilisés :

Documentation officielle de Springboot.

Documentation officielle de Java.

3. Avec qui avez-vous travaillé ?

Projet réalisé en duo.

4. Contexte

Nom de l'entreprise, organisme, ou association

Ascanio, pour le client Engage-sports

Chantier, atelier, service N/A

Période d'exercice Du 09/05/2022 au 13/05/2022

Dossier Professionnel (DP)

Activité type n°3 : Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

Compétence n°13 : Développer une application mobile

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

La société **Engages-sports** organise des événements sportifs pour les licenciés de la **Fédération Française de Motocyclisme**. À ce titre, elle a souhaité développer une application permettant :

- aux participants de ces événements **d'importer** et de **consulter** la liste de leurs licences sous forme de QR code,
- aux organisateurs de **scanner** les QR codes des licenciés et ainsi vérifier la validité des licences associées.

Les spécificités techniques seront énoncées ci-dessous dans les tâches effectuées. Le framework « Flutter » se basant sur le langage Dart a été choisi pour le développement de la dite application mobile.

La liste des tâches énoncées ci-dessous correspond à l'estimation telle qu'elle a été présentée au client. La réalisation n'a pas toujours été linéaire et ne correspond donc pas forcément à l'ordre présenté.

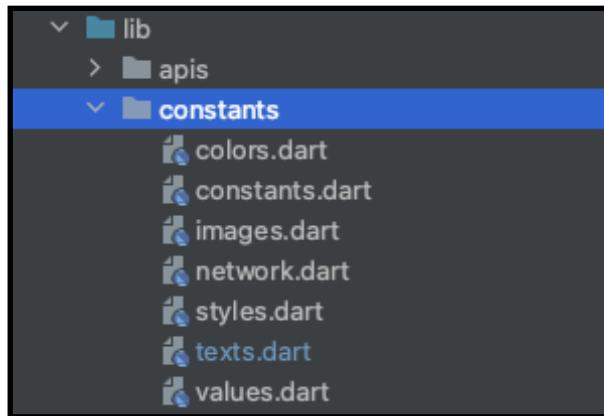
Étape 1 : Set-up du projet

Suite à la création d'un projet Flutter dans l'IDE et dans un soucis de sécuriser le travail effectué, la première étape a été de créer un repo Git pour le projet. J'ai ensuite créer l'architecture de dossiers permettant une efficacité maximale pendant la phase de développement et dans la maintenance de l'application.

Avant toute ligne de code, j'ai choisi de parcourir les maquettes réalisées en amont par un des UX-UI designer et d'intégrer le « design system » au projet Flutter (voir Annexe 1). Cette notion, venant de Figma, regroupe tous les choix de styles des composants présents dans l'application et notamment :

- La police de caractère
- Les tailles
- Les couleurs
- Les pictogrammes
- Les illustrations, photos ou images
- Le style des boutons
- Les espacements
- Les textes

Dossier Professionnel (DP)



Ces éléments sont regroupés dans un dossier « constantes » et rendus accessibles pour la suite du projet (Développement des composants graphiques).

Étape 2 : Intégration statique des maquettes

Cette étape a d'abord consisté à comprendre le parcours utilisateur tel qu'il a été représenté dans les maquettes et à lister les écrans et leur fonction principale. Une fois les fichiers créés, j'ai commencé le développement du premier écran, rajouté les logiques de navigation, jusqu'à intégrer l'ensemble des maquettes (voir Annexe 2).

C'est écran après écran, que chaque Widget (nom donné à un composant graphique dans Flutter) a été créé de manière abstraite dans le but de pouvoir les réutiliser à différents endroits dans le projet (voir annexe 3).

Étape 3 : Ajout de la logique métier

J'ai ensuite développé toute la logique métier de l'application :

- Récupération des licences d'un utilisateur via un appel API
- Stockage local des licences
- Scanner une licence via son QR Code et visualiser les informations qu'elle contient.

2. Précisez les moyens utilisés :

- Spécifications techniques et fonctionnelles du projet FFM Engage-Sports
- Documentation officielle de Flutter
- Sites communautaires d'entraide (Stackoverflow)
- Tutoriels YouTube
- Git

Dossier Professionnel (DP)

3. Avec qui avez-vous travaillé ?

Projet réalisé dans le cadre de mon alternance chez Ascanio, lors de la commande de l'application mobile FFM Engage-sports.

Équipe constituée de :

- Couvreux Benjamin, Chef de projet
- Julien, Client
- Pasha, Développeur back-end
- Fréville Maddy, UX-UI designer

4. Contexte

Nom de l'entreprise, organisme, ou association

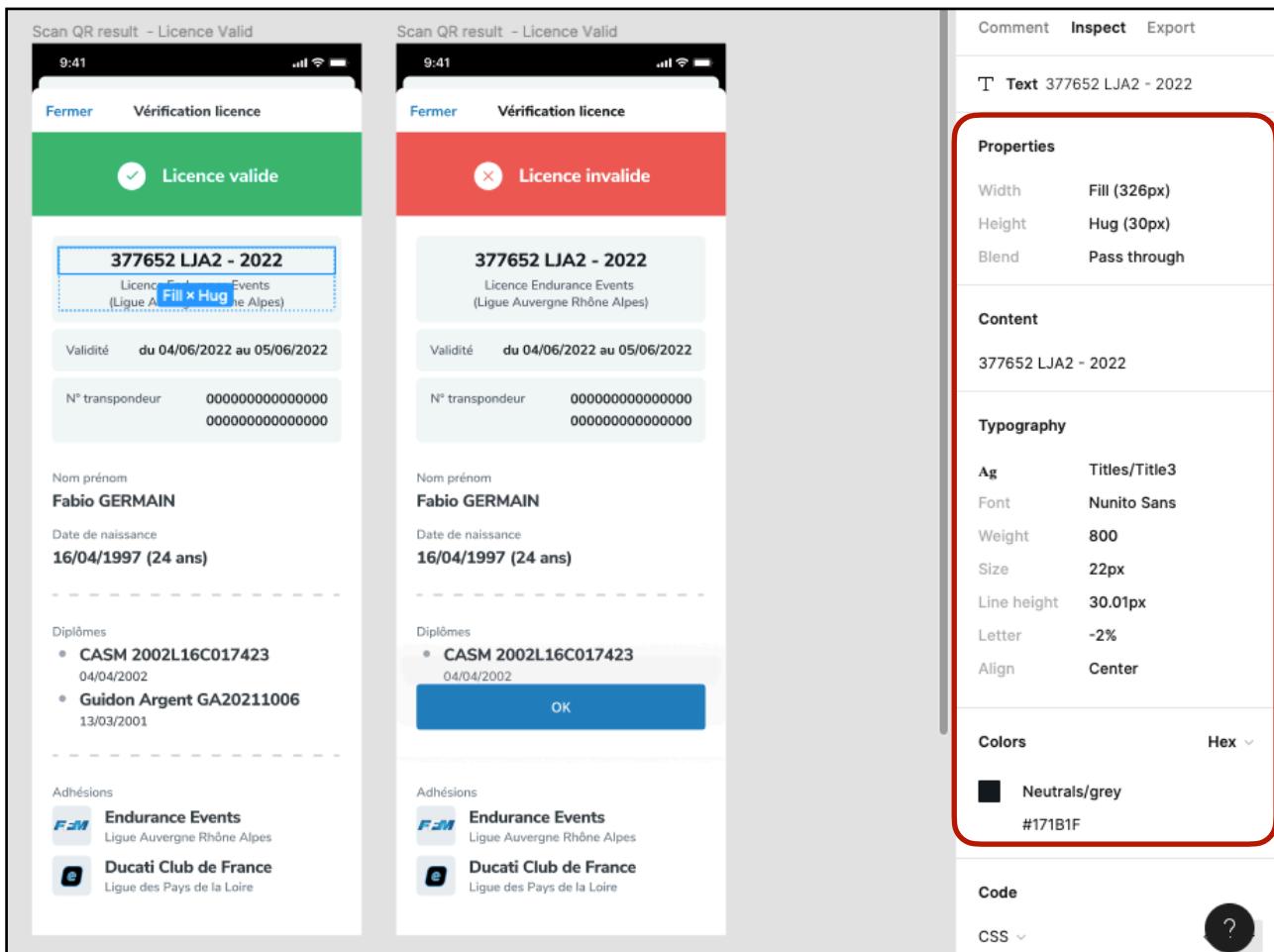
Ascanio, pour le client Engage-sports

Chantier, atelier, service N/A

Période d'exercice Du 15/04/2022 au 31/05/2022

Dossier Professionnel (DP)

5. Informations complémentaires (facultatif)

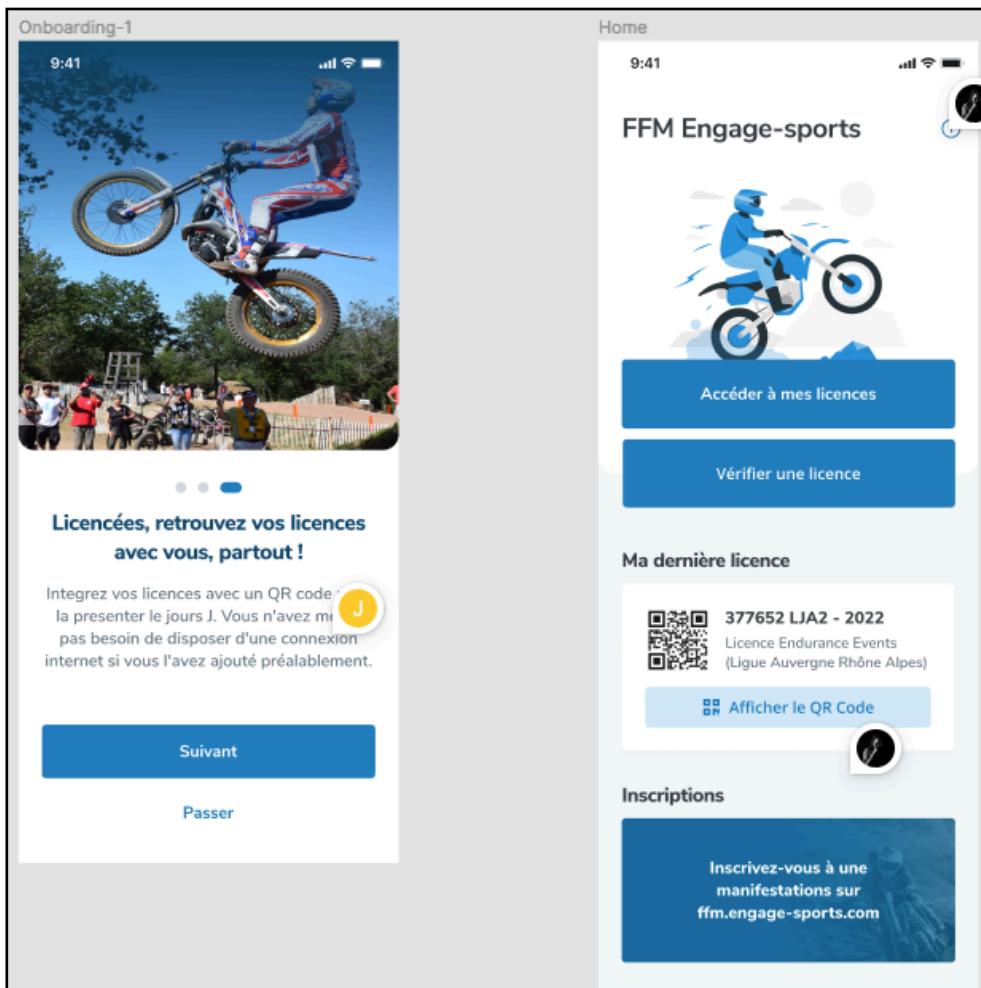


```
/// NAVIGATION
const String kHomeTitleNavText = 'FFM Engage-sports';
const String kCheckLicenceNavText = 'Vérification licence';
const String kLicenceBookNavText = 'Mon carnet de licences';
const String kAddLicenceNavText = 'Ajouter une licence';
const String kQRCodeLicenceNavText = 'QR Code licence';
const String kUpdateLicenceNavText = 'Mettre à jour une licence';
```

```
13 // IDENTITY
14 const kWhiteColor = Color(0xFFFFFFFF);
15 const kPrimaryBackgroundColor = Color(0xFFFF0F5F8);
16 const kPrimaryColor = Color(0xFF1F7EBF);
17 const kPrimaryLightTwoColor = Color(0xFF65B3E6);
18 const kPrimaryLightThreeColor = Color(0xFFD3E9F8);
19 const kPrimaryDarkTwoColor = Color(0xFF11496E);
```

Annexe 1 : Intégration du « design system »

Dossier Professionnel (DP)

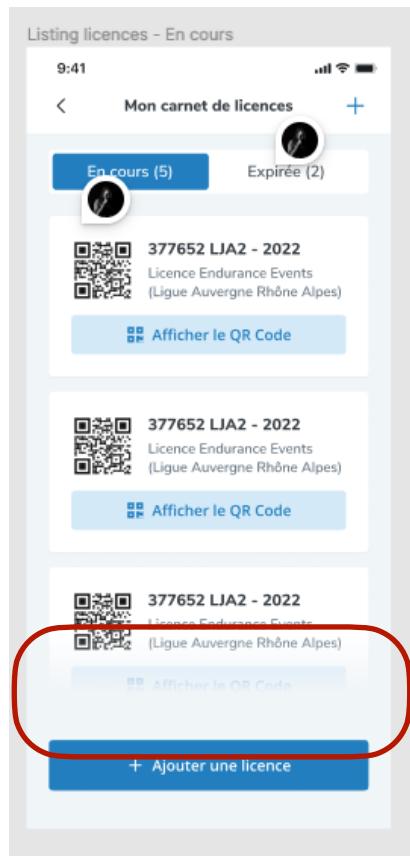


```
class OnboardingScreen extends StatefulWidget {  
  /// Screen displayed for the first user visit  
  /// providing data about the app  
  
  // CONSTRUCTOR  
  const OnboardingScreen({Key? key}) : super(key: key);  
  
  @override  
  _OnboardingScreenState createState() => _OnboardingScreenState();  
  
  class _OnboardingScreenState extends State<OnboardingScreen>  
    with SingleTickerProviderStateMixin {  
    // STATE PROPERTIES  
    int index = 0;  
    late PageController _pageController;  
    Duration pageTurnDuration = const Duration(milliseconds: 300);  
    Curve pageTurnCurve = Curves.ease;  
    List<Image> myImages = [];  
    SharedPreferences? sharedPreferences;  
  
    // INIT  
    @override  
    void initState() {  
      super.initState();  
      // Page controller  
      _pageController = PageController();  
      // Pre cache image  
      for (var imagePath in kOnboardingImages) {  
        myImages.add(Image.asset(imagePath));  
      }  
    }  
  }  
}
```

```
// DEPENDENCIES  
@override  
void didChangeDependencies() {  
  super.didChangeDependencies();  
  for (var image in myImages) {  
    precacheImage(image.image, context);  
  }  
}  
  
// DISPOSE  
@override  
void dispose() {  
  _pageController.dispose();  
  super.dispose();  
}  
  
// BUILD  
@override  
Widget build(BuildContext context) {  
  SystemChrome.setSystemUIOverlayStyle(SystemUiOverlayStyle.light);  
  return Scaffold(  
    body: Container(  
      width: double.infinity,  
      color: KWhiteColor,  
      child: Column(  
        children: [  
          _buildOnboardingPicture(),  
          _buildOnboardingText(context)  
        ],  
      ),  
    ),  
  );  
}
```

Annexe 2 : Intégration du premier écran

Dossier Professionnel (DP)



```
class ScrollShader extends StatelessWidget {  
  /// Shader mask used in licences screen to fade out  
  /// licences list  
  
  // PROPERTIES  
  final Widget child;  
  
  // CONSTRUCTOR  
  const ScrollShader({Key? key, required this.child}) : super(key: key);  
  
  // BUILD  
  @override  
  Widget build(BuildContext context) {  
    return ShaderMask(  
      shaderCallback: (Rect rect) {  
        return const LinearGradient(  
          begin: Alignment.topCenter,  
          end: Alignment.bottomCenter,  
          colors: [kWhiteColor, kWhiteColor, Colors.transparent],  
          stops: [0.0, 0.8, 1.0], // 10% purple, 80% transparent, 10% purple  
        ).createShader(rect); // LinearGradient  
      },  
      blendMode: BlendMode.dstIn,  
      child: child); // ShaderMask  
  }  
}
```

Annexe 3 : Crédit d'un composant réutilisable
(Exemple d'un masque d'opacité sur les éléments d'un liste)

Dossier Professionnel (DP)

Activité type n°3 : Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

Compétence n°14 : Préparer et exécuter les plans de tests d'une application

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

La société **Engages-sports** organise des événements sportifs pour les licenciés de la **Fédération Française de Motocyclisme**. À ce titre, elle a souhaité développer une application mobile permettant :

- aux participants de ces événements **d'importer** et de **consulter** la liste de leurs licences sous forme de QR code,
- aux organisateurs de **scanner** les QR codes des licenciés et ainsi vérifier la validité des licences associées.

Les plans de tests ont été conçus sous la forme d'un pyramide de tests :

- Les tests unitaires
- Les tests d'intégration
- Les tests End-to-End

L'application est basée sur le modèle des licences et constitue à lui seul la logique métier. J'ai donc commencé par tester tous les constructeurs personnalisés permettant d'instancier une licence (à partir d'un fichier json par exemple) ainsi que les méthodes de cette même classe. J'ai ensuite testé les fonctions permettant d'encoder et de décoder les licences dans l'application.

Les tests unitaires sont conçus en 3 phases :

- ARRANGE, qui consiste instancier ou initialiser les éléments du test
- ACT, qui consiste à appliquer une fonction sur un élément de la phase précédente.
- ASSERT, avec le mot clé « **expect** » qui permet de tester le résultat attendu (le type d'une variable par exemple)

Les tests d'intégration consistent à vérifier l'existence et/ou le comportement de certains composants graphiques. J'ai donc décidé de tester les écrans (eux-mêmes considérés comme des composants graphiques complexes) afin de m'assurer qu'ils contenaient bien les éléments souhaités.

Les tests end-to-end sont des tests manuels réalisés par une personne physique qui reprennent les différents cas d'utilisations du cahier des charges. Je me suis donc assuré que l'utilisateur puisse notamment :

- Visualiser le parcours « d'Onboarding » à sa première visite seulement.

Dossier Professionnel (DP)

- Importer ses licences Engage ou FFM pour les stocker localement sur le smartphone
- Scanner un QR code et afficher les informations correspondantes.

2. Précisez les moyens utilisés :

Les librairies de tests sont intégrées au Framework **Flutter**. Je me suis donc basé sur la documentation officielle. Quelques recherches sur **Stackoverflow** ont aussi été effectuées pour comprendre comment réaliser certains tests.

3. Avec qui avez-vous travaillé ?

Le projet a été réalisé pour le client Engage sports avec une équipe constituée d'un développeur mobile (moi-même), d'un **développeur Back-end**, d'un **UI/UX designer** ainsi qu'un **chef de projet**. Il y eu aussi beaucoup d'échanges avec le **responsable du projet chez Engage sports** tout au long de la réalisation du projet.

4. Contexte

Nom de l'entreprise, organisme, ou association Ascanio, pour le client Engage-sports

Chantier, atelier, service N/A

Période d'exercice Du 25/04/2022 au 31/05/2022

Dossier Professionnel (DP)

5. Informations complémentaires (facultatif)

```
test('Encrypt licence and decrypt should give the correct attributes quantity', () =>
    // ARRANGE
    final List<Degree> degrees = [
        Degree(typeFormation: 'typeFormation', dateDebutValidite: '12/05/1989'),
        Degree(typeFormation: 'typeFormationTwo', dateDebutValidite: '12/05/1990'),
    ];
    final List<Membership> memberships = [
        Membership(nomClub: 'nomClub', nomLigue: 'nomLigue'),
        Membership(nomClub: 'nomClub', nomLigue: 'nomLigue'),
    ];
    final licence = StoredLicence(
        id: "941364",
        saison: "2022",
        codeTypeLicence: "NCA",
        typeLicence: "NATIONALE COMPETITION A",
        codeClub: "C0855",
        nomClub: "MOTO CLUB SENAS DURANCE",
        codeLigue: "L20",
        nomLigue: "LIGUE PROVENCE",
        dateDebutValidite: "09/01/2022",
        dateFinValidite: "31/12/2022",
        etat: "Active",
        disciplineActivite: null,
        userFirstName: "MICHEL",
        userLastName: "CHEYLAN",
        userBirthDate: "08/11/1954",
        degrees: degrees,
        memberships: memberships
    );
    // ACT
    final data = EncryptionHelper.encryptMsg(licence.toReducedString()).base64;
    final decrypted = EncryptionHelper.decryptMsg(EncryptionHelper.getCode(data)).toString();
    final list = decrypted.split('|');
    // ASSERT
    expect(list.length == kLicenceLengthKey, true);
```

Exemple de test unitaire

On s'assure qu'une licence encryptée et décryptée contient bien le bon nombre d'éléments.

Dossier Professionnel (DP)

Activité type n°3 : Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

Compétence n°15 : Préparer et exécuter le déploiement d'une application

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

Le projet consistait à déployer une simple application Flask (Hello World) en mettant en place une chaîne de CI/CD.

La première étape consiste à rédiger le fichier `.gitlab-ci.yml` et de l'intégrer à la racine du projet. Ce fichier contient la liste des étapes (stages) de la chaîne CI/CD et toutes les instructions nécessaires par « stage ».

La deuxième étape consiste à exécuter un playbook Ansible, c'est-à-dire une suite de commandes permettant l'administration et la configuration d'une machine distante (machine sur laquelle est déployée notre application Flask).

2. Précisez les moyens utilisés :

Documentation sur le CI/CD de Gitlab.

Documentation de Ansible.

Ressources et tutoriels sur Youtube.

3. Avec qui avez-vous travaillé ?

Le projet a été réalisé individuellement.

4. Contexte

Nom de l'entreprise, organisme, ou association

Simplon

Chantier, atelier, service N/A

Période d'exercice Du 14/03/2022 au 19/03/2022

Dossier Professionnel (DP)

5. Informations complémentaires (facultatif)

```
! .gitlab-ci.yml > ! deploy.yml  
! .gitlab-ci.yml > {} deploy > [ ] rules > {} 0 > when  
  gitlab-ci (ci.json)  
  1   image: python:3.9  
  2   stages:  
  3     - deploy  
  4  
  5   deploy:  
  6     stage: deploy  
  7     before_script:  
  8       - python -m pip install --user ansible  
  9       - python -m pip install python-gitlab  
10      - export PATH=$PATH:~/local/bin  
11      - mkdir -p ~/.ssh  
12      - chmod 700 ~/.ssh  
13      - echo "$PRIVATE_KEY_LQ" | tr -d '\r' > ~/.ssh/id_rsa  
14      - chmod 600 ~/.ssh/id_rsa  
15      - eval "$(ssh-agent -s)"  
16      - ssh-add ~/.ssh/id_rsa  
17      - ssh-keyscan -H 163.172.180.197 >> ~/.ssh/known_hosts  
18  
19      - echo 'Check ansible commands'  
20      - ansible --version  
21      - ansible-playbook deploy/ansible/deploy.yml  
22      - ansible -i deploy/ansible/inventory -m ping all  
23     script:  
24       - ansible-playbook deploy/ansible/deploy.yml -i deploy/ansible/inventory -vvvv  
25     rules:  
26       - if: '$CI_COMMIT_BRANCH == "laboudigue_quentin"'  
27         when: manual  
28
```

Fichier gitlab-ci du projet qui s'exécute à chaque push sur le Git.

Dossier Professionnel (DP)

```
! deploy.yml 1 X

deploy > ansible > ! deploy.yml > {} 0 > [ ]tasks > {} 9
    Deployer Recipe (schema.json)
1   ---
2   - name: First Playbook
3     hosts: all
4     tasks:
5       - name: Display the config
6         command: whoami
7       - name: Install pipenv
8         command: pip install pipenv
9       - name: Create destination folder
10      command: mkdir -p /usr/src/app/quentin
11      - name: Go to the folder
12        command: chdir=/usr/src/app/quentin ls
13      - name: Change file ownership, group and permissions
14        ansible.builtin.file:
15          path: /usr/src/app/quentin
16          mode: "0777"
17      - name: Clone a repo with separate git directory
18        ansible.builtin.git:
19          repo: https://qlaboudigue:password@gitlab.com/qlaboudigue/testcicd.git
20          dest: /usr/src/app/quentin
21          version: laboudigue_quentin
22      - name: Go to the folder
23        command: chdir=/usr/src/app/quentin ls
24      - name: Install dependencies
25        command: pipenv install
26      - name: Go to the folder
27        command: chdir=/usr/src/app/quentin ls
28      - name: Run flask
29        command: chdir=/usr/src/app/quentin pipenv run flask run
30
```

Détail du script Ansible permettant l'exécution d'un série de tâche sur le Runner Gitlab.