

BTS SIO 2024  
Option SLAM  
Épreuve E5

**EPSI Montpellier**

**SOLANA-DUPLAA Léo**  
**02344511778**

## Situation n° 2/2 : Application Mobile

Flutter  
(Dart)

## Table des matières

|  |           |
|--|-----------|
| Présentation de la MAO.....  | 3         |
| Le client.....   | 5         |
| Le réseau de l'entreprise.....   | 6         |
| La réalisation .....   | 7         |
| Les fonctionnalités.....   | 7         |
| La démarche .....  | 8         |
| La base de données .....   | 9         |
| Quelques fonctionnalités expliquées .....  | 10        |
| Affichage des produits .....   | 10        |
| Explication de la fonctionnalité .....   | 10        |
| Capture d'écran .....  | 10        |
| Extrait de code .....  | 11        |
| Affichage les détails d'un produit .....   | 12        |
| Explication de la fonctionnalité .....   | 12        |
| Capture d'écran .....  | 12        |
| Extrait de code .....  | 13        |
| Ajout ou suppression d'un produit dans le panier .....                               | 14        |
| Explication de la fonctionnalité .....   | 14        |
| Capture d'écran .....  | 14        |
| Extrait de code .....  | 14        |
| Trigger et procédure.....  | 16        |
| Explication de la fonctionnalité du trigger .....                                    | 16        |
| Extrait de code .....  | 16        |
| Explication de la fonctionnalité de la procédure stockée .....                       | 17        |
| Extrait de code .....  | 17        |
| Serveur de versions.....   | 18        |
| Conclusion .....   | 18        |
| <b>ANNEXE 7-1-B : Fiche descriptive de réalisation professionnelle (recto) .....</b> | <b>19</b> |

## Présentation de la MAO

La MAO, ou Maison des Associations d'Occitanie, bénéficie du soutien financier du Conseil Régional et est gérée par le Comité Régional Olympique et Sportif du Languedoc Roussillon (CROSLR). Sa principale fonction est de proposer des locaux et des services tant aux associations sportives de la région qu'aux entités qui y sont domiciliées.

La gestion de la Maison des Ligues a été reprise en 2000 par la région, moment auquel les comités départementaux ont été promus par les athlètes, plaçant pour leur inclusion dans le soutien au développement des associations sportives. Depuis, la MAO accueille la plupart des associations sportives régionales et certains comités départementaux, grâce à une convention d'occupation de domaine public et un règlement intérieur qui définissent des critères clés, comme l'exigence pour les entités hébergées de souscrire une assurance responsabilité civile.

Le financement des opérations de la MAO provient en partie des contributions pour les frais de gestion courants, incluant le nettoyage, le chauffage, l'électricité, et l'accès à Internet. Les services fournis par la MAO sont facturés via un accord de cogestion avec le CROSLR et la Région Occitanie. La MAO offre aussi la possibilité aux ligues et comités départementaux de réserver des salles de réunion par un système de réservation en ligne. En 2009, elle a organisé l'utilisation de 2732 salles de réunion, accueillant plus de 47316 participants, avec une facturation s'appliquant seulement après plus de six réservations annuelles.

Pour assurer une tarification équitable, la MAO a établi quatre niveaux de prix. Les clubs sportifs et comités départementaux bénéficient d'un tarif préférentiel, tandis que les associations, écoles secondaires et collèges se voient appliquer un second niveau de tarification. Les tarifs les plus élevés concernent d'autres entités et les entreprises privées. L'équipe de la MAO se compose de divers professionnels, incluant deux membres du personnel du Conseil Régional et sept employés à temps plein du CROSLR.

En plus, la MAO emploie du personnel pour des services spécifiques (sécurité, nettoyage) et compte parmi ses rangs les employés des associations et des bénévoles, ces derniers étant élus parmi les membres de l'association. Les élus du Conseil Régional, les visiteurs et les stagiaires complètent l'équipe de la MAO. Avec moins de 10 employés à temps plein, la MAO bénéficie du Plan Impact Emploi Association, lui donnant accès à un logiciel spécialisé pour la gestion des salaires et des déclarations sociales et fiscales, service offert gratuitement par des tiers de confiance.

Ce système facilite la gestion des récapitulatifs périodiques, des déclarations à l'URSSAF et à l'ASSEDIC, ainsi qu'aux caisses de retraite. La MAO détient également la

certification «CRIB», lui permettant de fournir des conseils aux associations employant du personnel. Son but ultime est de superviser et de contrôler, via une plateforme de gestion, les systèmes informatiques de la MAO et ceux des entités hébergées.

Je travaille actuellement pour la MAO, au sein du département informatique, qui fait régulièrement appel à des entreprises locales spécialisées en informatique, établissant ainsi une relation contractuelle entre la MAO et ces entreprises.

## Le client

Pizza&Collect a sollicité notre équipe pour la création d'une application mobile de commerce.

En tant que pizzeria établie à Montpellier, Pizza&Collect désire s'adapter aux nouvelles habitudes d'achat de ses clients. La pandémie a renforcé la tendance au shopping en ligne, et l'entreprise souhaite répondre aux besoins de sa clientèle. Que ce soit pour ceux qui manquent de temps ou de moyens, ou simplement pour ceux qui préfèrent explorer et commander depuis le confort de leur canapé, Pizza&Collect veut offrir une expérience pratique et fluide.

L'objectif principal est de développer une application mobile qui permettra aux utilisateurs d'accéder facilement à un large éventail de produits culinaires, allant des entrées aux pizzas, en passant par les desserts et les boissons. De plus, cette application intégrera une fonctionnalité de gestion des stocks en temps réel. Voici comment elle bénéficiera à Pizza&Collect :

Expérience utilisateur améliorée : Les clients pourront naviguer aisément à travers le menu varié et passer leurs commandes sans effort.

En somme, cette application permettra à Pizza&Collect de servir ses clients de manière plus efficace.

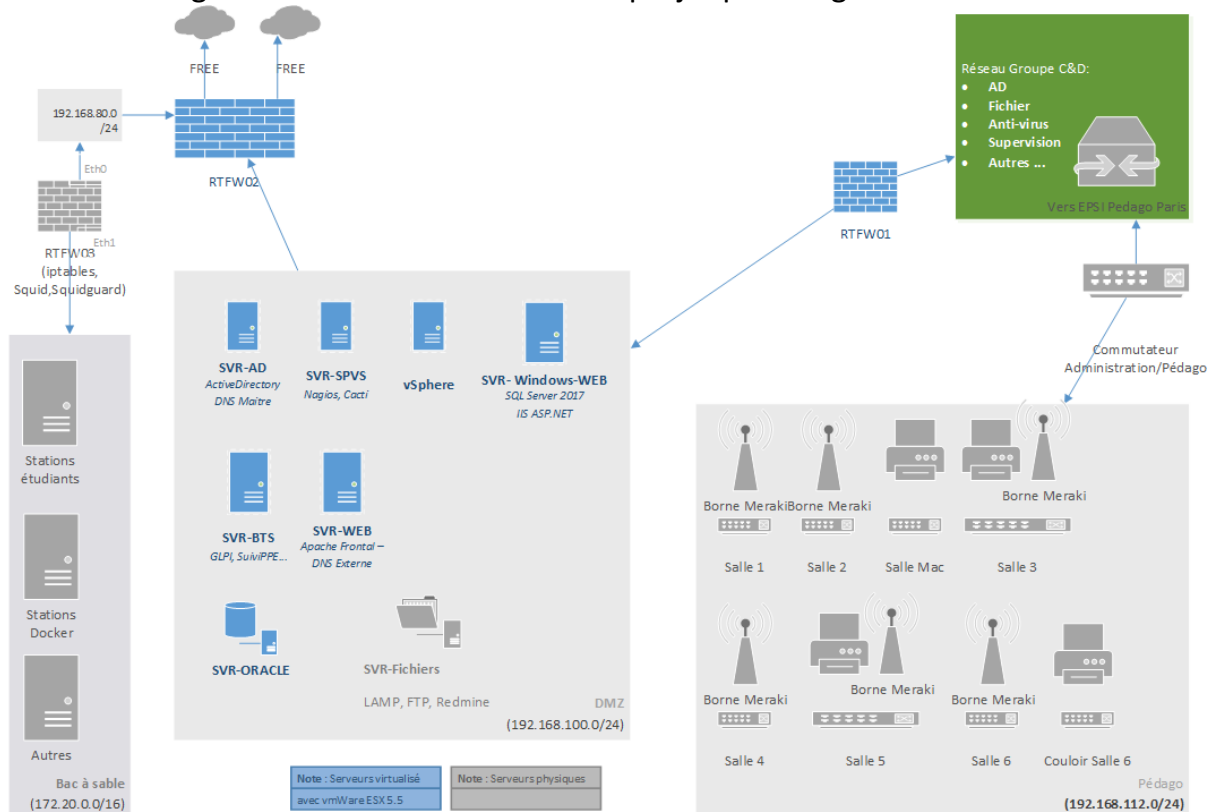
# Le réseau de l'entreprise

Présentation schématique du réseau de l'entreprise :

La MAO dispose actuellement :

- D'un réseau commun aux associations et à l'administration,
- D'un réseau de type DMZ dans lequel on trouve un ensemble de serveurs accessibles depuis l'extérieur,
- D'une connexion à internet.

Note : une segmentation des réseaux est en projet pour augmenter la sécurité.



# La réalisation

## Les fonctionnalités

Après avoir discuté avec le client il m'a énuméré les différentes fonctionnalités qu'il aimera retrouver dans cette application mobile, j'ai donc ainsi pu faire un cahier de charges :

### **Menu interactif :**

- Afficher les différentes catégories de plats (entrées, pizzas, desserts, boissons).
- Permettre aux utilisateurs de parcourir les articles dans chaque catégorie.
- Afficher les détails de chaque plat (nom, description, prix).

### **Commande en ligne :**

- Permettre aux clients de sélectionner des articles et de les ajouter à leur panier.
- Calculer le total de la commande en temps réel.
- Gérer les options (taille, garnitures, etc.) pour les pizzas.

### **Gestion du panier :**

- Afficher le contenu du panier.
- Permettre aux utilisateurs de modifier les quantités ou de supprimer des articles.
- Valider la commande et passer au paiement.

### **Interface Conviviale :**

- Design intuitif et attractif.
- Navigation fluide entre les écrans.
- Boutons d'action clairs pour commander, valider, etc.

### **Contraintes Techniques :**

- L'application sera développée en Flutter avec le langage Dart.
- Elle devra être compatible avec les systèmes d'exploitation iOS et Android.

## La démarche

Le développement de l'application s'est passé en deux étapes :

- **La création de l'interface de l'application mobile**

Cette partie correspond à la création des différentes pages en Dart.

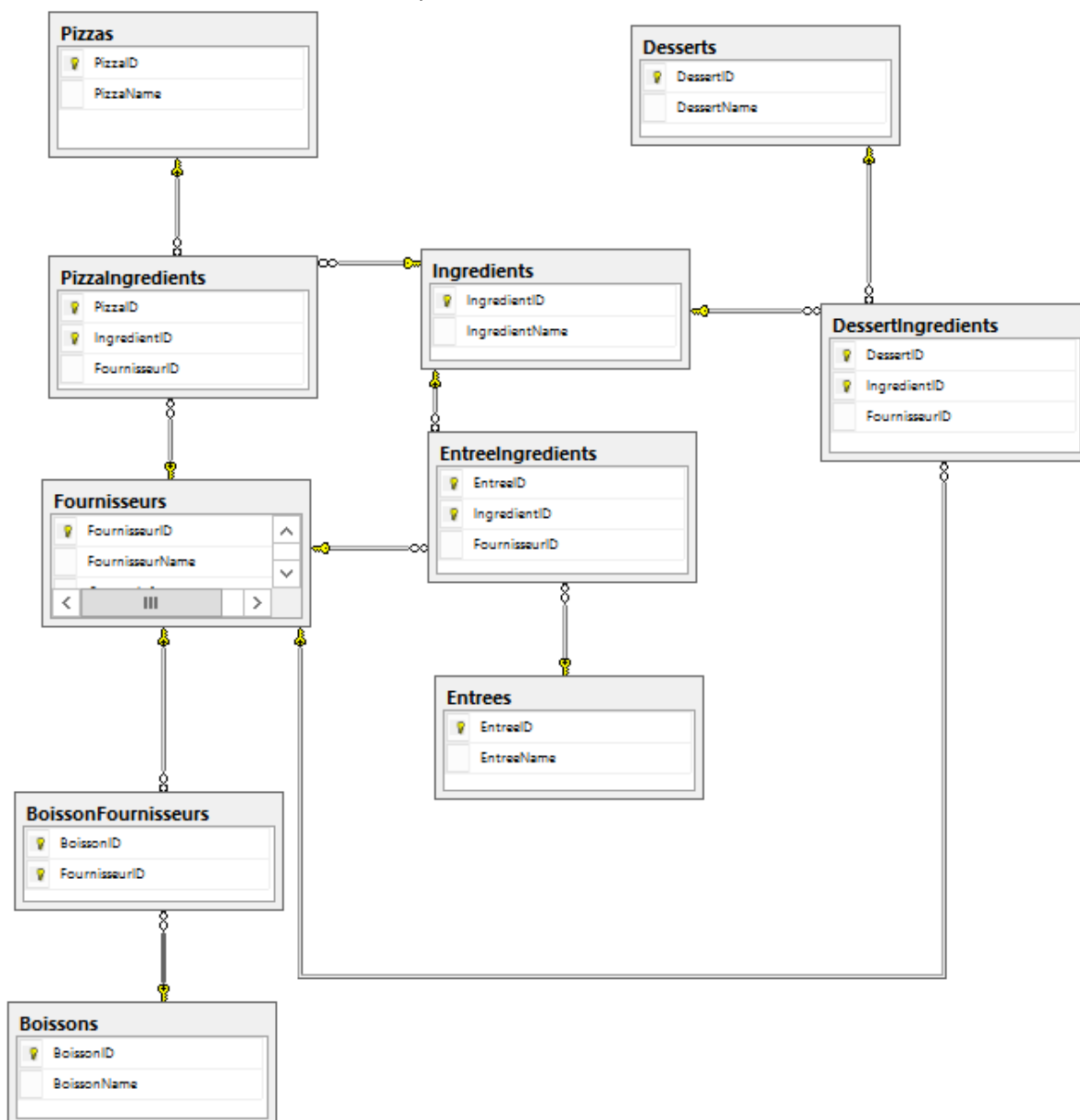
- **Mise en place des fonctionnalités**

Cette partie correspond à la mise en place du code et des différentes fonctions qui permet la récupération, l'ajout ou la suppression des éléments dans le panier.



## La base de données

La base de données est constituée de 5 tables correspondant aux fonctionnalités énumérées au début voici une représentation assez claire du MCD :



# Quelques fonctionnalités expliquées

## Affichage des produits

### Explication de la fonctionnalité

Dans cette partie de l'application pour afficher tous les produits présents dans le fichier Json il faut :

- Récupérer tous les produits présents dans un fichier Json renvoyé par l'api qui récupère les données de la base de données PhpMyAdmin hébergée en local avec Wamp.
- Puis j'affiche le contenu de la liste.

### Capture d'écran



## Extrait de code

```
class PizzeriaService {
    static final String uri = 'http://127.0.0.1/api';

    Future<List<Pizza>> fetchPizzas() async {
        List<Pizza> list = [];

        try {
            final response = await http.get(Uri.parse('$uri/pizzas'));

            if (response.statusCode == 200) {
                var json = jsonDecode(utf8.decode(response.bodyBytes));
                for (final value in json) {
                    list.add(Pizza.fromJson(value));
                }
            } else {
                throw Exception('Impossible de récupérer les pizzas');
            }
        } catch (e) {
            throw e;
        }
        return list;
    }
}
```

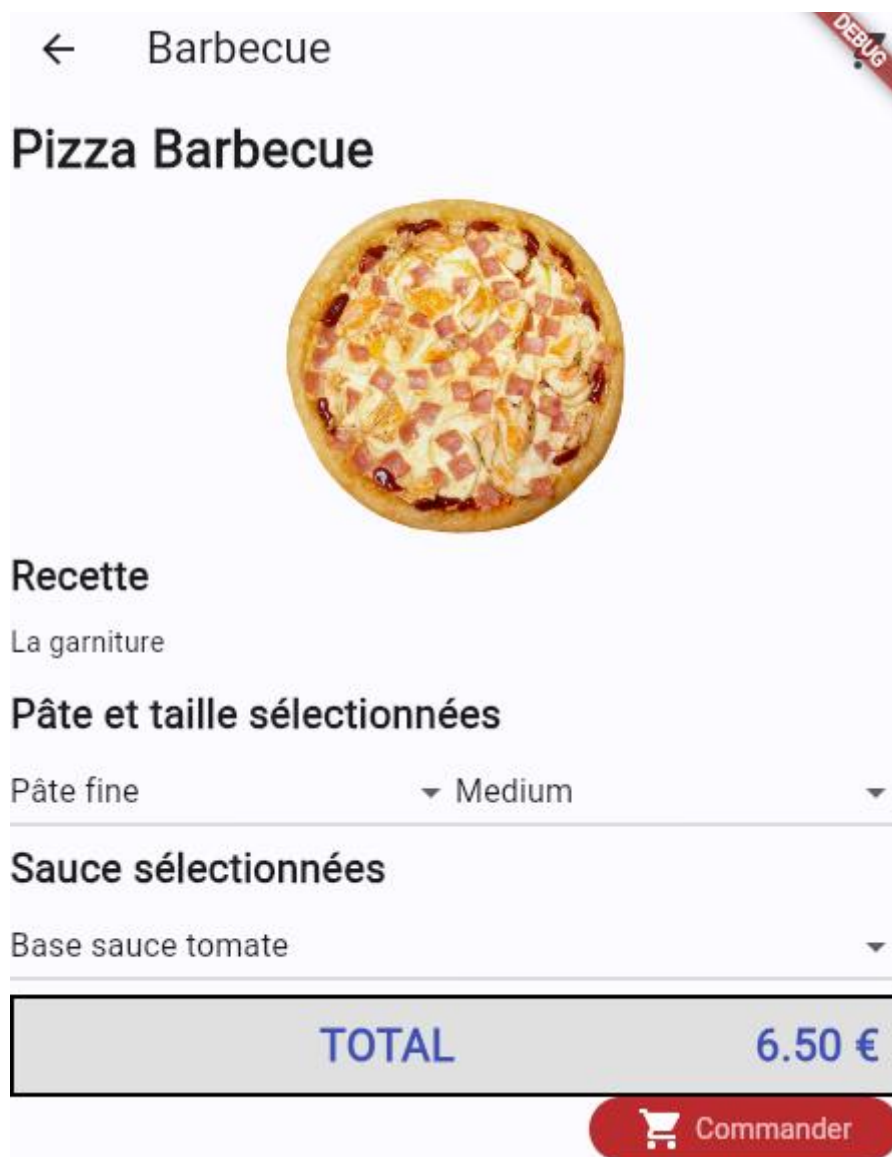
## Affichage les détails d'un produit

### Explication de la fonctionnalité

Dans cette partie de l'application pour afficher tous les détails d'un produit présent dans la liste des produits il faut :

- Récupérer tous les produits présents dans un fichier Json renvoyé par l'api qui récupère les données de la base de données PhpMyAdmin hébergée en local avec Wamp.
- Puis j'affiche le contenu de la liste des détails.

### Capture d'écran



## Extrait de code

```
Widget build(BuildContext context) {  
  return Scaffold(  
    appBar: AppBarWidget(widget._pizza.title, _cart),  
    body: ListView(  
      padding: EdgeInsets.all(4.0),  
      children: [  
        Text(  
          'Pizza ${widget._pizza.title}',  
          style: PizzeriaStyle.pageTitleTextStyle,  
        ),  
        Image.network(  
          '${widget._pizza.image}',  
          height: 180,  
        ),  
        Text(  
          'Recette',  
          style: PizzeriaStyle.headerTextStyle,  
        ),  
        Padding(  
          padding: EdgeInsets.only(top: 8.0, bottom: 12.0),  
          child: Text(  
            widget._pizza.garniture,  
          ),  
        ),  
      ],  
    ),  
  );  
}
```

## Ajout ou suppression d'un produit dans le panier

### Explication de la fonctionnalité

Dans cette partie de l'application pour augmenter le nombre de produit ou le diminuer dans le panier il faut :

- Créer deux boutons dans chaque case de produit.
- Créer les fonctions qui augmente le nombre ou le diminue

### Capture d'écran



### Extrait de code

```
void addProduct(Pizza pizza) {  
    int index = findCartItemIndex(pizza.id);  
    if (index == -1) {  
        _items.add(CartItem(pizza));  
    } else {  
        CartItem item = _items[index];  
        item.quantity++;  
    }  
}
```

```
void removeProduct(Pizza pizza) {  
    int index = findCartItemIndex(pizza.id);  
    if (index != -1) {  
        _items[index].quantity--; // Décrémentez la quantité de l'élément existant  
        if (_items[index].quantity <= 0) {  
            _items.removeAt(index); // Supprimez l'élément si la quantité est nulle ou négative  
        }  
    }  
}
```

## Trigger et procédure

### Explication de la fonctionnalité du trigger

Création du déclencheur pour enregistrer les insertions dans la table Pizzas

Pour ce faire :

### Extrait de code

```
CREATE TABLE PizzaLog (  
    LogID INT PRIMARY KEY IDENTITY(1,1),  
    PizzaID INT,  
    PizzaName NVARCHAR(100),  
    DateInserted DATETIME DEFAULT GETDATE()  
);  
  
CREATE TRIGGER trg_AfterInsertPizza  
ON Pizzas  
AFTER INSERT  
AS  
BEGIN  
    INSERT INTO PizzaLog (PizzaID, PizzaName)  
    SELECT PizzaID, PizzaName  
    FROM inserted;  
END;  
GO
```



## Explication de la fonctionnalité de la procédure stockée

Création de la procédure stockée pour ajouter une nouvelle pizza  
Pour ce faire :

### Extrait de code

```
CREATE PROCEDURE AddPizzaWithIngredients
    @PizzaName NVARCHAR(100),
    @Ingredients NVARCHAR(MAX),
    @Fournisseurs NVARCHAR(MAX)
AS
BEGIN
    INSERT INTO Pizzas (PizzaName)
    VALUES (@PizzaName);

    DECLARE @PizzaID INT = SCOPE_IDENTITY();

    INSERT INTO Ingredients (IngredientName)
    SELECT value FROM STRING_SPLIT(@Ingredients, ',')
    WHERE value NOT IN (SELECT IngredientName FROM Ingredients);

    INSERT INTO Fournisseurs (FournisseurName)
    SELECT value FROM STRING_SPLIT(@Fournisseurs, ',')
    WHERE value NOT IN (SELECT FournisseurName FROM Fournisseurs);

    INSERT INTO PizzaIngredients (PizzaID, IngredientID, FournisseurID)
    SELECT @PizzaID, I.IngredientID, F.FournisseurID
    FROM STRING_SPLIT(@Ingredients, ',') IL
    INNER JOIN Ingredients I ON IL.value = I.IngredientName
    CROSS JOIN STRING_SPLIT(@Fournisseurs, ',') FL
    INNER JOIN Fournisseurs F ON FL.value = F.FournisseurName;
END;
GO
```

## Serveur de versions

Pour sauvegarder mon travail j'ai utilisé un serveur de version. Le serveur de version que j'ai utilisé est GitHub le projet se nomme Pizza&Collect.

Voici le lien pour accéder au projet :

<https://github.com/leosdlp/Pizza-Collect>

## Conclusion

Le projet d'une application mobile demandé par la pizzeria Pizza&Collect a été mené à bien, les différentes fonctionnalités demandées ont été ajoutés à l'application. Pizza&Collect peut désormais vendre leurs produits accessibles à tout le monde, même s'ils sont dans leur canapé.

## Épreuve E5 - Conception et développement d'applications (option SLAM)

## ANNEXE 7-1-B : Fiche descriptive de réalisation professionnelle (recto)

|  |                                |                           |
|--|--------------------------------|---------------------------|
| <b>DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE</b>   |                                | <b>N° réalisation : 2</b> |
| Nom, prénom : SOLANA-DUPLAA Léo  |                                | N° candidat : 02344511778 |
| Épreuve ponctuelle X   | Contrôle en cours de formation | Date : 12 / 04 / 2024     |
| Organisation support de la réalisation professionnelle<br>EPSI Montpellier   |                                |                           |
| Intitulé de la réalisation professionnelle<br>Application mobile Pizza&Collect   |                                |                           |
| Période de réalisation : Octobre 2023 à Mars 2024 Lieu : Montpellier   |                                |                           |
| Modalité : X Seul(e) En équipe   |                                |                           |
| Compétences travaillées<br>Concevoir et développer une solution applicative<br>Assurer la maintenance corrective ou évolutive d'une solution applicative   |                                |                           |
| Conditions de réalisation <sup>5</sup> (ressources fournies, résultats attendus)<br><br>Cahier des charge / liste de fonctionnalité<br>Application mobile fonctionnelle  |                                |                           |
| Description des ressources documentaires, matérielles et logicielles utilisées <sup>6</sup><br>AndroidStudio, ordinateur personnel   |                                |                           |
| Modalités d'accès aux productions <sup>7</sup> et à leur documentation <sup>8</sup><br><br>OneDrive:<br><a href="https://drive.google.com/drive/u/1/folders/11BIUhkEyMmQEAwv2iJ6QCetJwkZbXnZ4">https://drive.google.com/drive/u/1/folders/11BIUhkEyMmQEAwv2iJ6QCetJwkZbXnZ4</a><br><br>GIT : <a href="https://github.com/leosdlp/Pizza-Collect">https://github.com/leosdlp/Pizza-Collect</a> |                                |                           |

**Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs**

**Voir la documentation**

**2.2 Les activités de l'organisation cliente s'appuient sur aux moins deux solutions applicatives opérationnelles permettant d'offrir un accès sécurisé à des données hébergées sur un site distant. Au sein des architectures de ces solutions applicatives doivent figurer l'exploitation de mécanismes d'appel à des services applicatifs distants et au moins trois des situations ci-dessous :**

| <b>Éléments</b>   | <b>Description de l'implantation dans le centre d'examen (nom du service ou de l'outil et caractéristiques techniques)</b> | <b>Remarques de la commission D'interrogation</b> |
|---|--|---|
| Du code exécuté sur le système d'exploitation d'une solution technique d'accès fixe (type client lourd) |  |   |
| Du code exécuté dans un navigateur Web (type client léger ou riche)                                     | Visual studio code - Framework Angular(HTML, CSS, TypeScript)  |   |
| Du code exécuté sur le système d'exploitation d'une solution technique d'accès mobile                   | AndroidStudio – Flutter (Dart)   |   |
| Du code exécuté sur le système d'exploitation d'un serveur  | PhpMyAdmin (Procédures stockées, Trigger)  |   |

**2.3 Une solution applicative peut être issue d'un développement spécifique ou de la modification du code d'un logiciel notamment open source.**

**2.4 Les solutions applicatives présentes dans le contexte sont opérationnelles et leur code source est accessible dans un environnement de développement opérationnel au moment de l'épreuve.**