

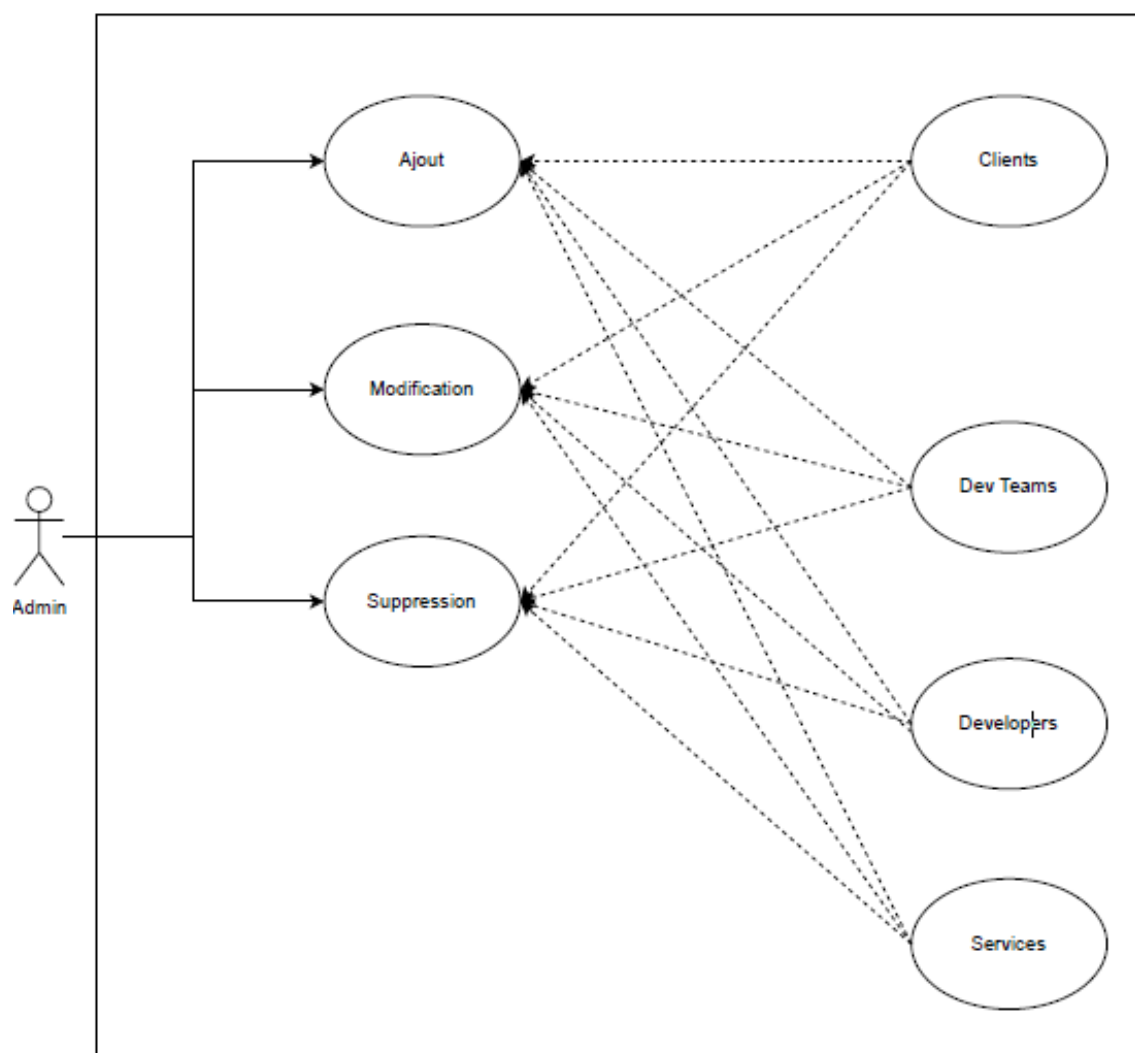
Compte rendu:

I. Introduction

Le livrable comprend la partie backend du site web de la société YourInterface. Elle permet à l'administrateur une gestion de contenu et de maintenir la base de données à travers des opérations CRUD(create, read, update, delete).

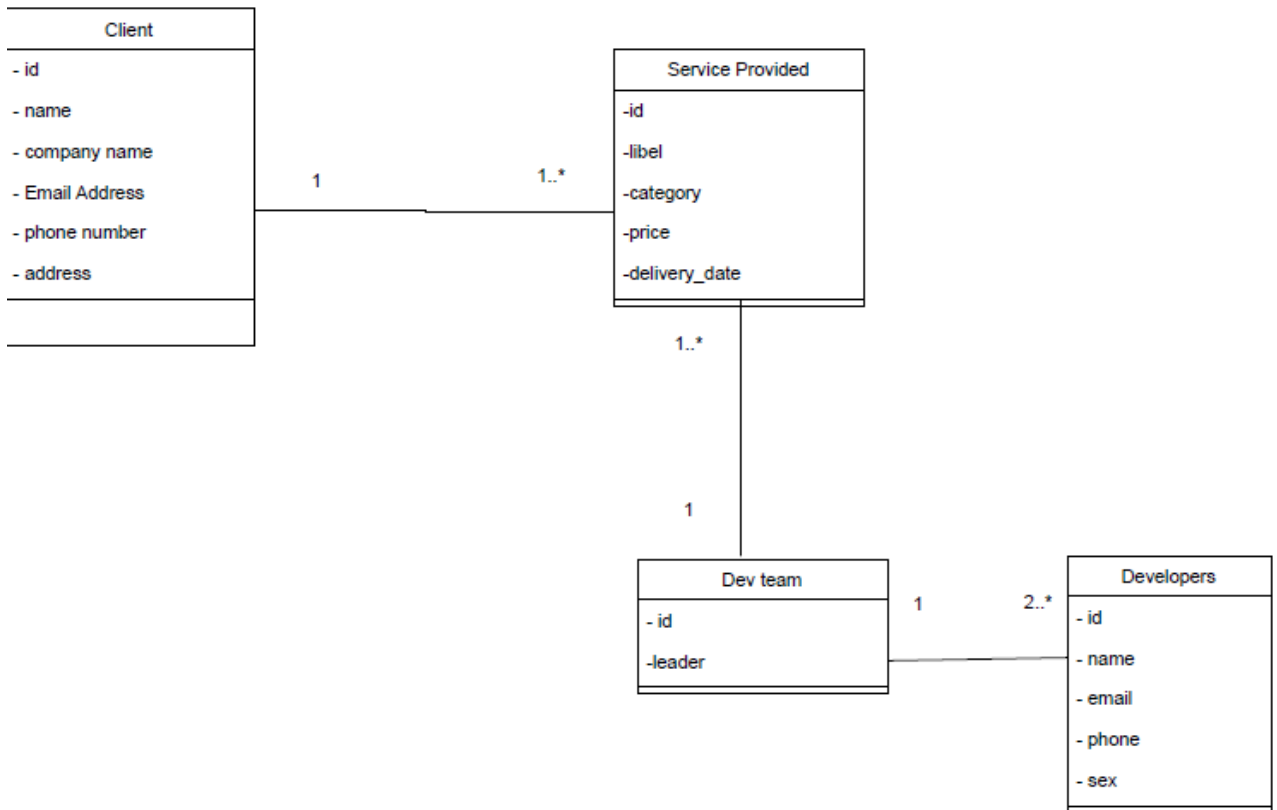
II. Démarche

- Dans le but de bien se préparer et de maîtriser les points du brief j'ai commencé par :
 - i. Compréhension de l'énoncé du brief.
 - ii. Discussion en squad du brief.
 - iii. Documentation sur le php.
 - iv. Installation de l'environnement de travail: XAMPP.
- Réalisation des diagrammes UML(de cas d'utilisations, classes). Avec description des relations entre les entités :



Dans le diagramme de cas d'utilisation l'acteur principale est l'administrateur qui aura des 3 cas d'utilisations l'ajout la modification et la suppression des données des clients, développeurs et équipes et les services.

Diagramme de classe-YoureInterface












Dans le diagramme de classe on a 4 classes : La classes "service provided" comprend les services vendu par l'agence, elle a deux relations d'association une relation one to many avec la classe client : un client peut avoir un ou plusieurs services alors q' un service vendu ne peut avoir qu'un acquéreur. La deuxième relation et avec la classe qui comprend les équipes de développeurs chaque équipe peut travailler sur plusieurs services alors que chaque service ne peut être réaliser que par une équipe. La classe des équipes a aussi une associations avec la classe qui comprend les développeurs: chaque équipe se compose de deux ou plus de développeurs alors qu'un développeur ne peut appartenir qu' a une seule équipe.

- Création de la base de donnée et des tableaux sur phpMyAdmin :







```
CREATE TABLE utilisateurs(  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    name VARCHAR(50) NOT NULL,  
    email VARCHAR(20),  
    phone VARCHAR(20),  
    adress VARCHAR(50),  
    created_at DATETIME DEFAULT CURRENT_TIMESTAMP  
);  
CREATE TABLE developers(  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    name VARCHAR(50) NOT NULL,  
    email VARCHAR(20),  
    phone VARCHAR(20),  
    sex VARCHAR(3),  
    team_id  
);  
CREATE TABLE services(  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    libel VARCHAR(50) NOT NULL,  
    category VARCHAR(50),  
    price FLOAT  
);  
CREATE TABLE developers_teams(  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    developer_id INT,  
    team_id INT,  
    created_at DATETIME DEFAULT CURRENT_TIMESTAMP  
);
```

- Structure de mon dossier de rendu :

	customers
	developers
	services_provided
	Diagramme-cas-utilisation.pdf
	README.md
	connection.php
	diagramme-class.pdf
	requettes.sql
	style.css

Chaque table de données est présenter dans un dossier. Le dossier comprend 4 fichiers un pour le code de l' affichage, la modification, la suppression, l' ajout comme suit :

	create.php
	delet.php
	edit.php
	index.php

- Aperçu du dashboard :

YI Yourinterface

Dashboard

Customers

Dev teams

Services

List of services provided

Add service

id	libel	category	price	delivery_date	team_id	client_id	client_name	company_name	action
1	Website creation	Front-end	3000	2023-11-26 14:30:05	1	1	Moussaoui	Azure	<div>Edit</div> <div>Delet</div>
2	Content & Copywriting	Front-end	8000	2023-11-26 14:30:05	1	3	Mohamed	Dolphine	<div>Edit</div> <div>Delet</div>
6	Search engine optimization	Front-end	6000	2023-11-26 17:50:07	1	4	Kamal	Lagoon	<div>Edit</div> <div>Delet</div>
9	Content & Copywriting	Front-end	6000	2023-11-27 09:54:33	7	8	Alj Yasser	Placebo	<div>Edit</div> <div>Delet</div>

Il comprend un tableau qui affiche les données un menu latéral qui permet de naviguer entre les différentes pages et tableaux des buttons pour ajouter, éditer, supprimer.

- Pour l’insertion des données on a un formulaire et des buttons submit et cancel :

Add

Label

Category

Price

Team

Client_id

Submit

Cancel

- Pour la modifications des données on a un formulaire avec les données insérer :

Modification

Libel	Content & Copywriting
Category	Front-end
Price	6000
TEAM_ID	7
CLIENT_ID	8
name	Alj Yasser
company_name	Placebo

Submit

Cancel

- Pour ce qui est du code j ai travailler les opérations CRUD par la connection de la base de données avec le php comme suit :

```
1  <?php
2  $servername = "localhost";
3  $username = "root";
4  $password = "";
5  $database = "brief2";
6  // Create connection
7  $connection = mysqli_connect($servername, $username, $password, $database);
8  ?>
```

- Pour l’affichage le code lit la requête select du tableau et permet d’ afficher les données du tableau :

```

<?php
require '../connection.php';
$requete="SELECT * FROM utilisateurs ";
$query=mysqli_query($connection,$requete);
// read data of each row
while($row = mysqli_fetch_assoc($query)){

    $editUrl="edit.php?id=$row[id]";
    $deleteUrl="delete.php?id=$row[id]";
    echo"
    <tr>
    <td>$row[id]</td>
    <td>$row[name]</td>
    <td>$row[company_name]</td>
    <td>$row[email]</td>
    <td>$row[phone]</td>
    <td>$row[adress]</td>
    <td>$row[created_at]</td>
    <td>
        <a class='btn btn-primary btn-sm' href='$editUrl'>Edit</a>
        <a class='btn btn-danger btn-sm' href='$deleteUrl'>Delete</a>
    </td>
    </tr>

```

- Pour la modification le code permet d'afficher les données de la ligne ou on veut faire des modifications:

```

require '../connection.php';
if(isset($_GET["id"])){
$id=$_GET['id'];
$requete="SELECT * FROM utilisateurs WHERE id='$id' ";
$query=mysqli_query($connection,$requete);
$row=mysqli_fetch_assoc($query);
$name = $row["name"];
$company_name = $row["company_name"];
$email = $row["email"];
$phone = $row["phone"];
$adress = $row["adress"];

}

```

- Et avec la méthode POST et requête SQL "UPDATE" on modifie les données comme suit:


```

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $id= $_POST["id"];
    $name = $_POST["name"];
    $company_name = $_POST["company_name"];
    $email = $_POST["email"];
    $phone = $_POST["phone"];
    $adress = $_POST["adress"];

    $requete ="UPDATE utilisateurs
        SET name='$name',company_name='$company_name',email='$email',phone='$phone',adress='$adress'
        WHERE id='$id'";
    $query=mysqli_query($connection,$requete);

```

- Pour la suppression des données on fait appelle a une requête "DELETE":

```

<?php

require '../connection.php';
$id=$_GET['id'];
$requete="DELETE FROM utilisateurs WHERE id=$id";
$query=mysqli_query($connection,$requete);
if(isset($requete)){
    echo"utilisateur supprime";
    header("location:../index.php");
}

?>

```

