

Projet Intranet

Table des matières

1. Mission 5 :	1
----------------------	---

1.Mission 5 :

Installez un environnement de développement pour PHP avec une base de données ou utilisez une machine virtuelle existante.

On va utiliser VS Code et Netbeans pour environnement de développement pour utiliser PHP. Également notre base de données s'appellera comme demandé « corpany_project_intranet ». Nous avons installé Netbeans comme on peut le voir ci-desous :

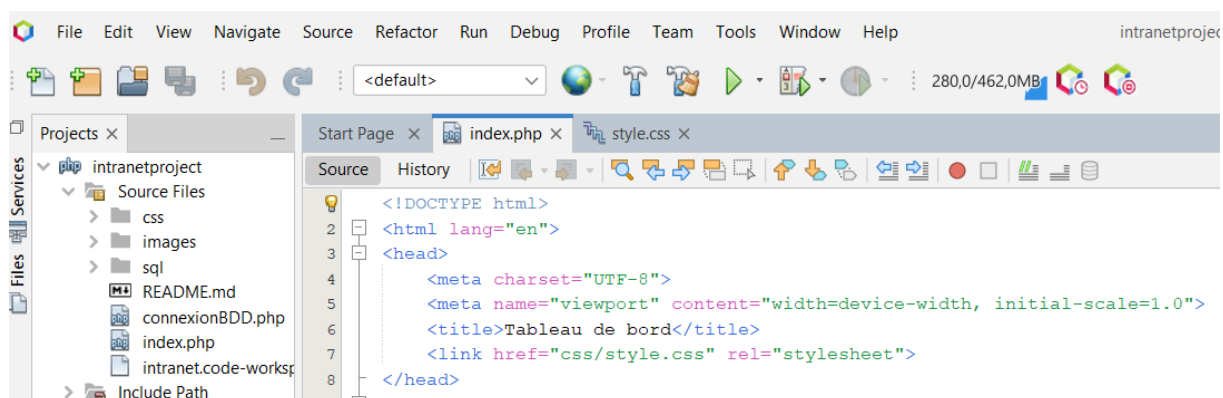


Figure 1 : Environnement de développement Netbeans

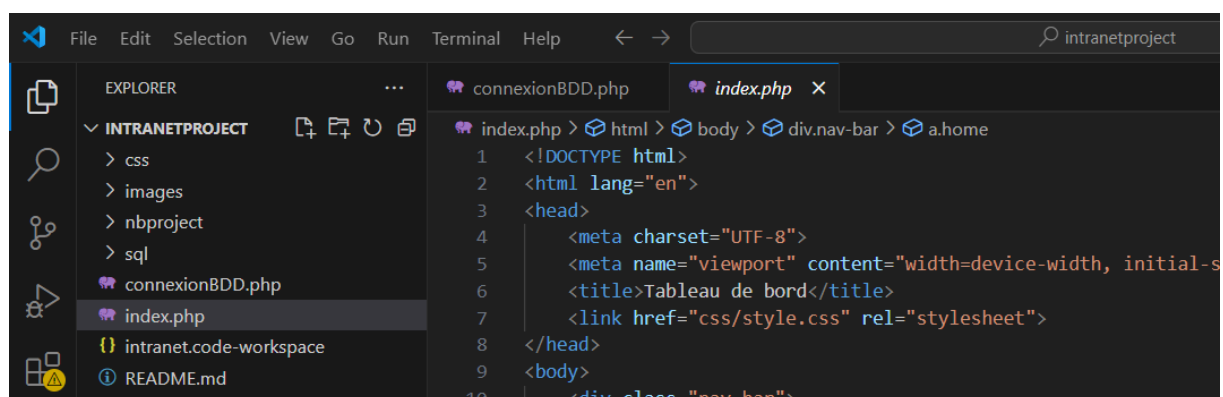
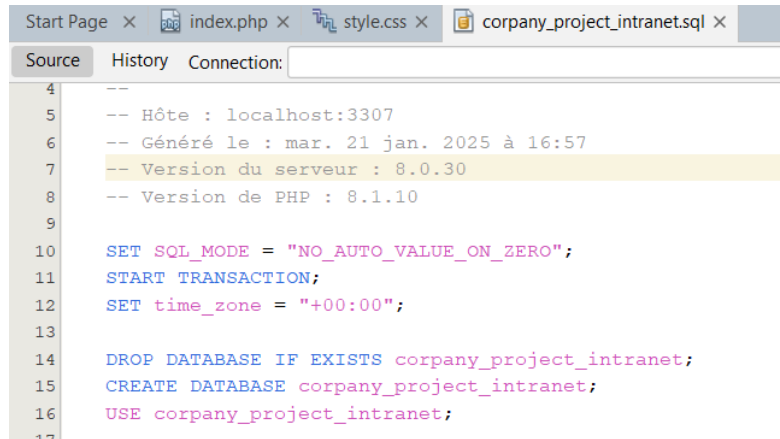


Figure 2 : Environnement de développement VS Code

AT Professionnel – Projet Intranet

Créez une base de données dans MariaDB avec le nom "corpany_project_intranet".

Tout d'abord dans le script SQL de la base de données, on supprime la base si elle existe déjà pour éviter les erreurs de type : la base de données est déjà existante. Puis on crée la base de donnée avec la commande « CREATE DATABASE nom_table ». On donne le nom demandé c'est-à-dire "corpany_project_intranet".



```
4  --
5  -- Hôte : localhost:3307
6  -- Généré le : mar. 21 jan. 2025 à 16:57
7  -- Version du serveur : 8.0.30
8  -- Version de PHP : 8.1.10
9
10 SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
11 START TRANSACTION;
12 SET time_zone = "+00:00";
13
14 DROP DATABASE IF EXISTS corpany_project_intranet;
15 CREATE DATABASE corpany_project_intranet;
16 USE corpany_project_intranet;
17
```

Figure 3 : Création de la base de donnée corpany_project_intranet

Puis on a créé différentes tables qui nous seront utiles pour les différentes fonctionnalités que nous voulons mettre en place :

Voici une explication des différentes tables de la base de données corpany_project_intranet et leur utilité :

Table actualites :

- **Utilité** : Stocker les actualités de l'entreprise ou du projet.
- **Champs principaux** :
 - titre et description : Contenu de l'actualité.
 - grp : Groupe concerné par l'actualité, lié à la table groupes.
 - creation_date : Date de création de l'actualité.
- **Relations** :
 - Le champ grp fait référence au groupe dans la table groupes.

Table documents :

- **Utilité** : Conserver les informations relatives aux documents partagés dans l'intranet (va servir notamment pour l'espace personnel des utilisateurs de l'intranet).
- **Champs principaux** :

- nom_fichier et chemin_fichier : Identité et emplacement du fichier.
- taille_fichier et type_fichier : Métadonnées techniques du document.
- uploader : Utilisateur ayant ajouté le document, lié à utilisateurs.
- **Relations :**
 - uploader fait référence à un utilisateur dans la table utilisateurs.

Table evenements :

- **Utilité :** Gérer les événements internes, tels que réunions ou formations.
- **Champs principaux :**
 - titre, description, date_debut, date_fin : Informations sur l'événement.
 - organisateur : Personne responsable de l'événement, liée à utilisateurs.
- **Relations :**
 - organisateur fait référence à la table utilisateurs.

Table groupes :

- **Utilité :** Définir les groupes ou départements de l'organisation.
- **Champs principaux :**
 - codeGrp : Code unique du groupe.
 - libelle : Nom descriptif du groupe (ex. : Informatique, Logistique).
- **Relations :**
 - Utilisé comme clé étrangère dans plusieurs tables (actualites, utilisateurs).

Table historique connexions :

- **Utilité :** Voir les dernières connexions des utilisateurs
- **Champs principaux :**
 - utilisateur : Identifie qui s'est connecté, lié à utilisateurs.
 - ip_connexion : Adresse IP utilisée.
 - date_connexion : Horodatage de la connexion.
- **Relations :**

- utilisateur est relié à la table utilisateurs.

Table historique taches :

- **Utilité** : Suivre les tâches assignées dans l'organisation.
- **Champs principaux** :
 - titre, description : Détails de la tâche.
 - responsable : Utilisateur chargé de la tâche, lié à utilisateurs.
 - date_realisation et date_echeance : Suivi temporel.
 - statut : État de la tâche (par défaut "en cours").
- **Relations** :
 - responsable est relié à la table utilisateurs.

Table infos employes :

- **Utilité** : Contient les informations personnelles des employés.
- **Champs principaux** :
 - nom, prenom, login : Identité de l'employé.
 - leposte : Poste occupé, lié à postes.
 - tel, mail : Coordonnées.
 - chemin_image : Photo de profil de l'employé.
- **Relations** :
 - leposte fait référence à la table postes.
 - login est lié à utilisateurs.

Table messages :

- **Utilité** : Gérer la messagerie interne entre utilisateurs.
- **Champs principaux** :
 - expéditeur, destinataire : Utilisateurs concernés, liés à utilisateurs.
 - message : Contenu du message.
 - date_envoi : Horodatage de l'envoi.
- **Relations** :
 - expéditeur et destinataire font référence à la table utilisateurs.

Table notifications :

- **Utilité** : Informer les utilisateurs des nouvelles actions ou événements.
- **Champs principaux** :
 - utilisateur : Utilisateur destinataire, lié à utilisateurs.
 - type_notification : Type de notification (ex. : message, événement).
 - message : Contenu de la notification.
- **Relations** :
 - utilisateur est relié à la table utilisateurs.

Table postes :

- **Utilité** : Répertoire les différents postes dans l'organisation.
- **Champs principaux** :
 - idposte : Code unique du poste.
 - libelleposte : Description du poste (ex. : Directeur, Employé).
- **Relations** :
 - Utilisé comme clé étrangère dans infos_employes.

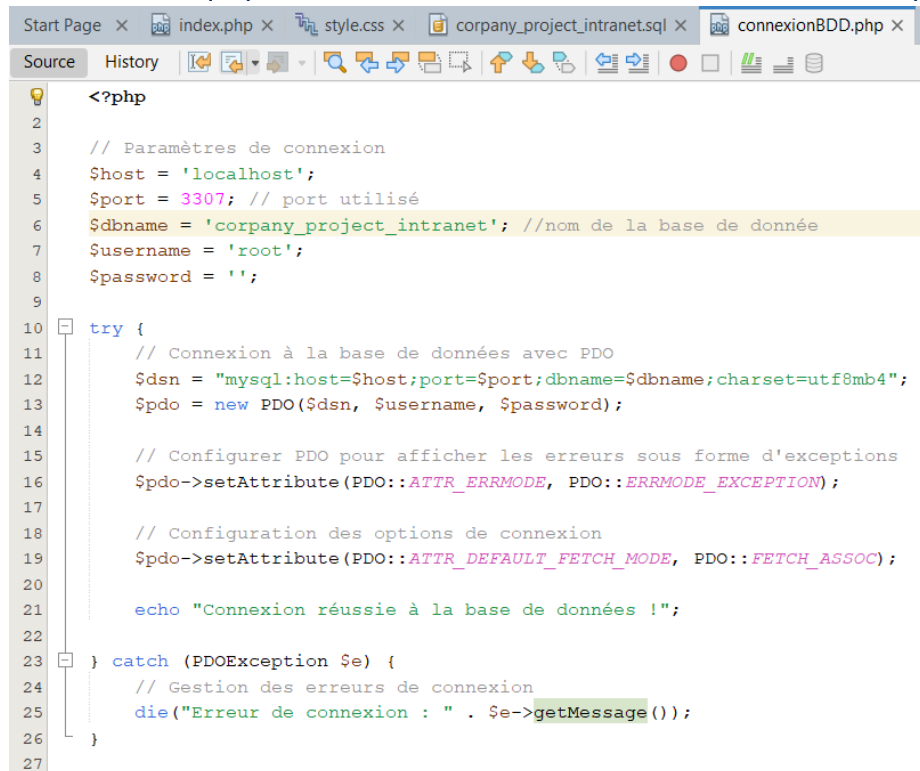
Table utilisateurs :

- **Utilité** : Stocke les informations de connexion des utilisateurs.
- **Champs principaux** :
 - login, password : Identifiants de connexion.
 - codeGrp : Groupe auquel appartient l'utilisateur, lié à groupes.
- **Relations** :
 - codeGrp fait référence à groupes.

Créez un fichier php avec du code permettant de tester l'accès à la base de données

AT Professionnel – Projet Intranet

Voici le fichier qui permet de tester l'accès à la base de données il s'appelle connexionBDD :



```
<?php
2
3 // Paramètres de connexion
4 $host = 'localhost';
5 $port = 3307; // port utilisé
6 $dbname = 'corpany_project_intranet'; //nom de la base de donnée
7 $username = 'root';
8 $password = '';
9
10 try {
11 // Connexion à la base de données avec PDO
12 $dsn = "mysql:host=$host;port=$port;dbname=$dbname;charset=utf8mb4";
13 $pdo = new PDO($dsn, $username, $password);
14
15 // Configurer PDO pour afficher les erreurs sous forme d'exceptions
16 $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
17
18 // Configuration des options de connexion
19 $pdo->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE, PDO::FETCH_ASSOC);
20
21 echo "Connexion réussie à la base de données !";
22
23 } catch (PDOException $e) {
24 // Gestion des erreurs de connexion
25 die("Erreur de connexion : " . $e->getMessage());
26 }
27
```

Figure 4 : fichier permettant de tester l'accès à la base de données

Il faut d'abord définir les paramètres de connexion avec :

- **host** : qui est l'adresse du serveur (souvent localhost pour un serveur local).
- **port** : c'est le port utilisé par le serveur MySQL (souvent 3306 ou 3307).
- **dbname** : c'est le nom de la base de données cible donc ici c'est corpany_project_intranet.
- **username** : c'est le nom d'utilisateur.
- **password** : enfin c'est le mot de passe de cet utilisateur.

Ensuite on va pouvoir créer une instance PDO, grâce à l'objet PDO qui est utilisé pour établir la connexion grâce :

- Le type de base de données (mysql).
- Les informations d'hôte, de port et de base (\$host, \$port, \$dbname).
- L'encodage des caractères (utf8mb4) pour gérer les caractères spéciaux.

Ensuite pour configurer le PDO, on a les options importantes sont définies avec le mode d'erreur. Les erreurs sont traitées sous forme d'exceptions, ce qui permet une gestion plus précise. Et également avec le mode de récupération des résultats où les résultats des requêtes sont retournés. Également, on peut afficher un message de réussite si la connexion réussit, un message est affiché pour confirmer. Enfin on peut gérer les erreurs grâce au bloc try...catch qui va permettre de capturer les exceptions levées par PDO en cas d'échec

CADIX Axel

PORTOLLEAU Anaïs

Groupe 1

AT Professionnel – Projet Intranet

(mauvais identifiants, serveur inaccessible, etc.) et affiche un message d'erreur descriptif.

Avant de lancer le projet, on doit bien configurer le projet comme ceci par exemple :

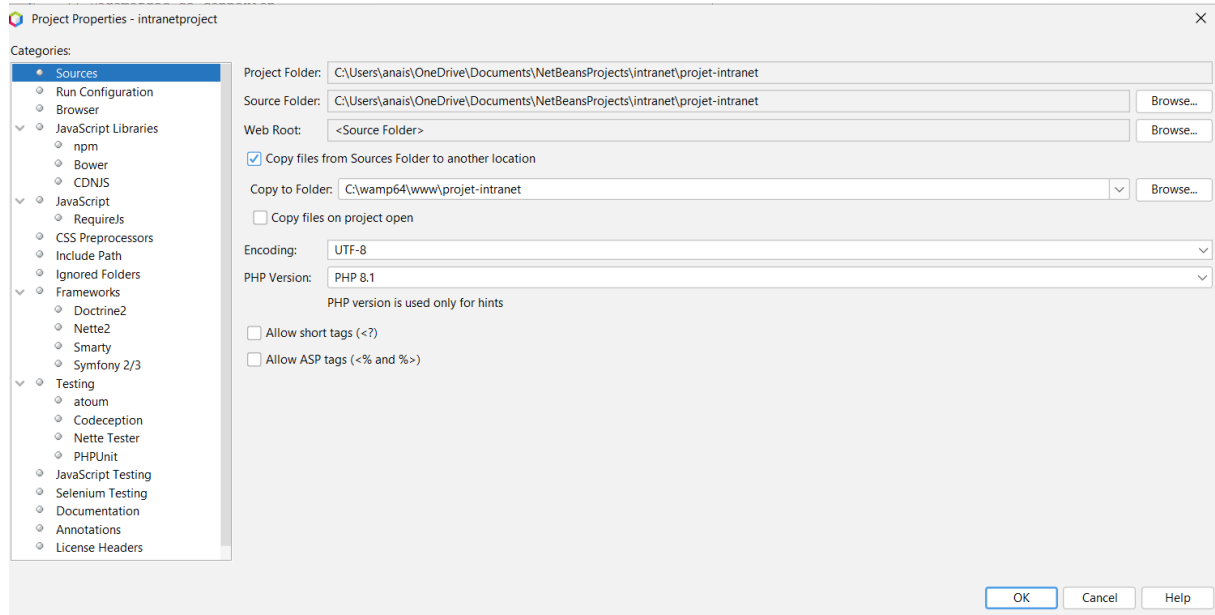


Figure 5 : Configuration du projet

On peut voir que le projet est situé dans les documents personnels. On copie le projet dans le www. Puis on va dans les run configuration pour copier la partie après le www (ici si on regarde ci-dessus c'est \projet-intranet qui est après le www donc on le copie dans le run configuration ci-dessous :

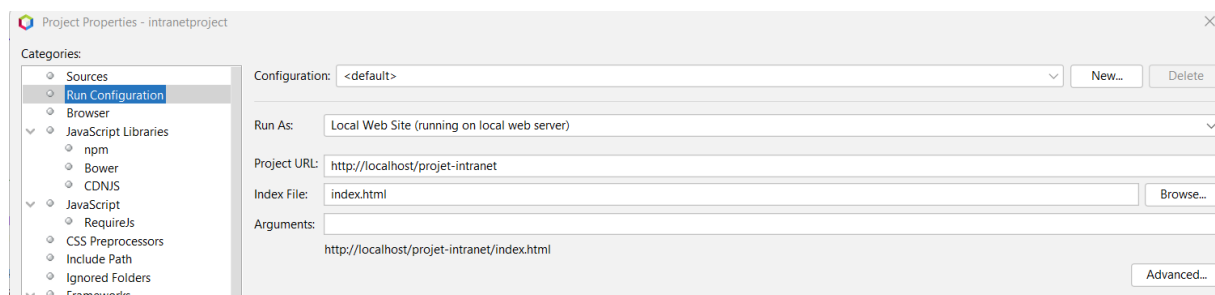


Figure 6 : Configuration 2 du projet

On a donc notre url <http://localhost/projet-intranet>

Capture de la connexion réussie sur l'ordinateur d'Anaïs :

Il faut bien ne pas oublier de démarrer WAMP pour tester et cela donne ceci comme on peut

AT Professionnel – Projet Intranet

le voir ci-dessous après l'exécution du fichier connexionBDD cela marche :

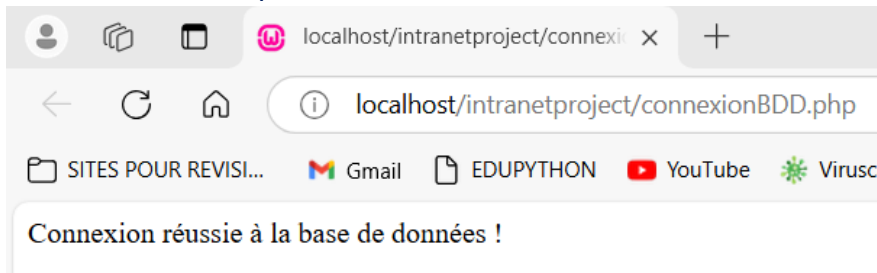


Figure 7 : Accès réussie à la base de données pour Anaïs

Capture de la connexion réussie sur l'ordinateur d'Axel :

Cela marche également chez Axel comme on peut le voir ci-dessous :



Figure 8 : Capture de la tentative de connexion d'Axel

Capture de la connexion réussie sur l'ordinateur de Benjamin :

Enfin également chez Benjamin comme on peut le voir ci-dessous :

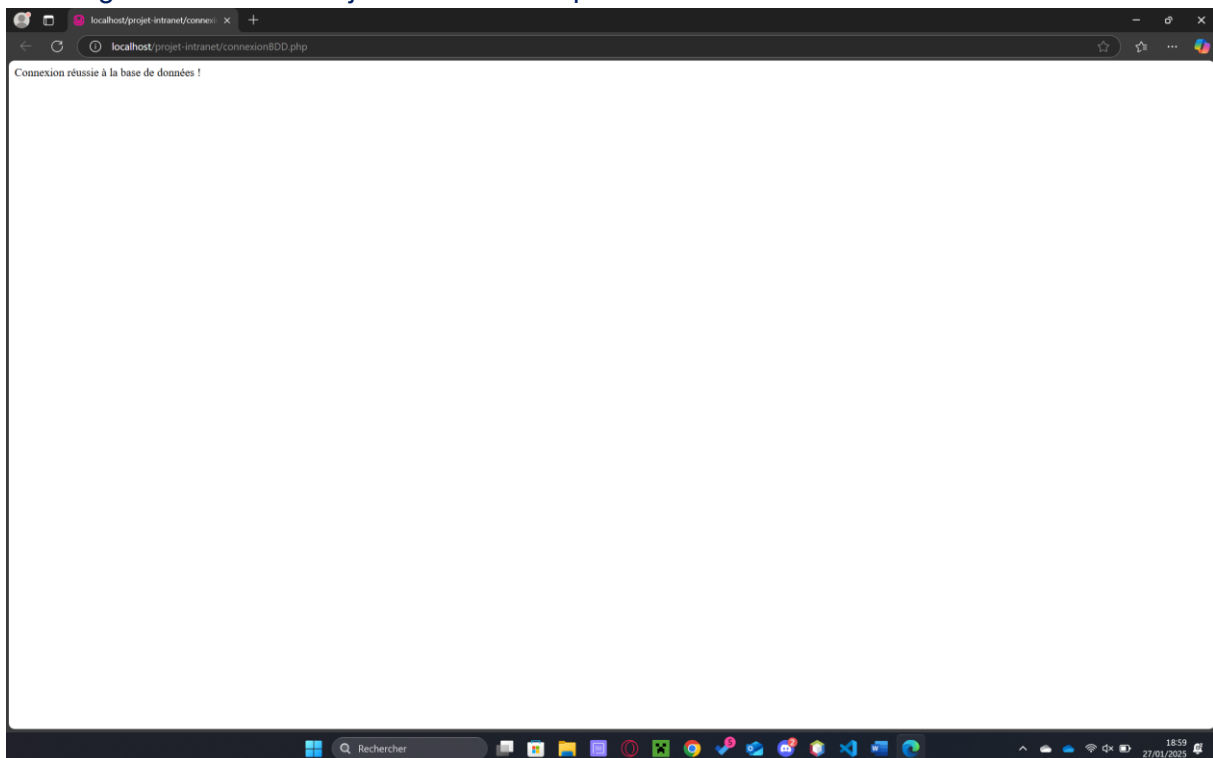


Figure 9 : Capture de la tentative de connexion (et d'accès) réussie du côté de Benjamin