



# DOSSIER PROFESSIONNEL<sup>(DP)</sup>

*Nom de naissance*

► BERTHOLLET

*Nom d'usage*

►

*Prénom*

► Arnaud

*Adresse*

► 22/08/95

## Titre professionnel visé

Titre Professionnel Développeur Web et Web mobile

### MODALITÉ D'ACCÈS :

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

## 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.*



**<http://travail-emploi.gouv.fr/titres-professionnels>**

# Sommaire

## Exemples de pratique professionnelle

<b>Développer la partie front-end d'une application web ou web mobile sécurisée</b>	<b>p.5</b>
▶ Exemple n° 1 : Installation de WampServer, qui permet d'héberger et d'exécuter localement des applications web développées en PHP .....	p.5
▶ Exemple n° 2 : Mise en place d'une fonctionnalité « dark mode » grâce à un script JS pour une interface utilisateur .....	p.7
<b>Développer la partie back-end d'une application web ou web mobile sécurisée</b>	<b>p.9</b>
▶ Exemple n° 1 : Mise en place d'une base de donnée relationnelle avec la méthode Merise .....	p.9
▶ Exemple n° 2 : Connexion à la base de données et manipulation de données via PDO et des requêtes SQL .....	p.11
<b>Titres, diplômes, CQP, attestations de formation</b> <i>(facultatif)</i>	<b>p.</b>
<b>Déclaration sur l'honneur</b>	<b>p.15</b>
<b>Documents illustrant la pratique professionnelle</b> <i>(facultatif)</i>	<b>p.</b>
<b>Annexes</b> <i>(Si le RC le prévoit)</i>	<b>p.</b>

# **EXEMPLES DE PRATIQUE PROFESSIONNELLE**

## Activité-type 1

Développer la partie front-end d'une application web ou web mobile sécurisée

**Exemple n°1** ► Installation de WampServer, qui permet d'héberger et d'exécuter localement des applications web développées en PHP

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

Pour pouvoir développer localement mon application web, j'ai installé et configuré **WampServer** sur mon ordinateur personnel sous Windows 10. J'ai commencé par télécharger l'installateur depuis le site officiel, en prenant soin de choisir la bonne version (64 bits, compatible avec ma machine).

Avant l'installation, j'ai désactivé temporairement les logiciels susceptibles de bloquer le port 80 (comme Skype ou un autre serveur local) pour éviter tout conflit avec Apache. J'ai ensuite lancé l'installation pas à pas, en suivant les instructions (choix du répertoire, vérification des versions de PHP, Apache et MySQL installées avec le package). Il a ensuite fallu attendre que tous les packages aient été installés sur mon ordinateur.

Une fois l'installation terminée, j'ai vérifié que les services Apache et MySQL fonctionnaient correctement via l'icône de Wamp dans la barre des tâches ; l'icône étant de couleur verte si tous les services ont pu être lancés correctement. J'ai également testé l'accès au *localhost* depuis mon navigateur pour m'assurer que le serveur local était bien opérationnel.

Ensuite, j'ai créé un répertoire pour mon projet dans le dossier *www* de Wamp, puis j'ai placé les premiers fichiers PHP et HTML de mon application. J'ai pu ainsi lancer mes premiers tests d'affichage, de connexions et de scripts en PHP dans un environnement local.

### 2. Précisez les moyens utilisés :

Pour installer et configurer WampServer, j'ai utilisé mon **ordinateur personnel** équipé de **Windows 10** et d'un **microprocesseur i7**. L'installation a été faite à partir de l'**installateur officiel** de WampServer, téléchargé depuis le **site officiel du projet** (<https://www.wampserver.com/>), afin de garantir la fiabilité et la sécurité du logiciel.

J'ai utilisé un **navigateur web (Google Chrome)** pour accéder à l'interface locale via *localhost*, ainsi qu'un **explorateur de fichiers** pour organiser mes projets dans le dossier *www*.

Pour configurer et tester les services (Apache, MySQL, PHP), je me suis appuyé sur :

- L'**interface graphique de WampServer**, accessible depuis la barre des tâches ;
- L'**éditeur de texte Visual Studio Code**, que j'utilise pour écrire et modifier mes fichiers HTML, CSS, JavaScript et PHP ;

### 3. Avec qui avez-vous travaillé ?

Pour l'installation de WampServer, durant le processus, j'ai pu échanger **ponctuellement des collègues de formation**, notamment pour résoudre de petits problèmes techniques (téléchargement des packages, configuration PHP).

Ce travail s'est donc fait à la fois **individuellement** et **dans un esprit d'entraide** entre apprenants. Nous étions également guidés par notre tuteur de formation car cela s'est fait en début de formation.

### 4. Contexte

Nom de l'entreprise, organisme ou association ► CEEFii, organisme de formation

Chantier, atelier, service ► Prérequis pour pouvoir travailler sur du PHP tout au long de l'année

Période d'exercice ► en septembre 2024, au début de la formation

### 5. Informations complémentaires (*facultatif*)

## Activité-type 1

Développer la partie front-end d'une application web ou web mobile sécurisée

*Exemple n° 2 ► Mise en place d'une fonctionnalité « dark mode » grâce à un script JS pour une interface utilisateur*

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

J'ai mis en place pour une interface utilisateur une fonctionnalité permettant à l'utilisateur de basculer entre un affichage clair (mode classique) et un affichage sombre (dark mode). Cette fonctionnalité améliore l'**accessibilité** et le **confort de lecture**, tout en répondant aux **habitudes modernes de navigation**.

J'ai commencé par créer un bouton HTML identifié par #darkModeBtn, destiné à déclencher le changement de thème. J'ai ensuite développé un **script JavaScript** qui applique dynamiquement une classe CSS (darkMode) à un conteneur principal de la page (#bodyContainer), permettant de modifier l'apparence globale du site selon les préférences de l'utilisateur.

Pour garantir la **persistance de ce choix** entre deux sessions, j'ai stocké dans le **localStorage**, grâce aux méthodes de l'interface Storage : *setItem()* et *getItem()*, une variable *darkmode*, de type *boolean*, pour enregistrer l'état du mode sombre dans le navigateur de l'utilisateur. Le script vérifie cet état au chargement de la page pour appliquer automatiquement le bon thème.

J'ai testé cette fonctionnalité dans un environnement local sous WampServer, en ajustant les classes CSS liées au thème sombre, puis en vérifiant que l'expérience utilisateur restait cohérente lors de l'activation/désactivation du mode, et après rechargement de la page.

J'ai finalisé cette tâche pendant une séance d'exercice durant la formation, puis amélioré le script en dehors des heures de cours pour le rendre plus fiable et ergonomique.

## 2. Précisez les moyens utilisés :

Pour mettre en place la fonctionnalité de dark mode, j'ai utilisé plusieurs **outils et ressources techniques**.

J'ai travaillé sur mon **ordinateur personnel**, avec le **navigateur Google Chrome** pour tester les rendus en direct, et l'**éditeur de code Visual Studio Code** pour écrire et organiser mes fichiers HTML, CSS et JavaScript.

Le script JavaScript repose sur des **fonctions de base du DOM** (querySelector, addEventListener, classList.toggle) et utilise l'API **localStorage** pour enregistrer la préférence de l'utilisateur de façon persistante.

Côté interface, j'ai utilisé une **structure HTML simple** contenant un bouton et un conteneur global (#bodyContainer), ainsi qu'un **fichier CSS** définissant les styles appliqués quand la classe darkMode est active (par exemple : couleurs de fond sombres, textes plus clairs).

J'ai également utilisé les **outils de développement du navigateur** (console JavaScript, inspection des éléments) pour vérifier le bon déclenchement des événements, l'ajout/retrait de la classe darkMode, et la bonne lecture/écriture des données dans le localStorage.

## 3. Avec qui avez-vous travaillé ?

J'ai principalement réalisé cette fonctionnalité **de manière autonome**, dans le cadre d'un exercice de mise en pratique pendant la formation. Elle faisait suite à un **cours théorique** sur le JavaScript, les événements, le DOM et le stockage local, animé par notre **tuteur de formation**.

Lors de l'implémentation en travaux pratiques, j'ai pu m'appuyer sur ses **explications du matin** pour structurer mon code et comprendre comment rendre l'affichage dynamique en fonction des actions de l'utilisateur.

Même si cette tâche a été menée individuellement, j'ai pu bénéficier ponctuellement de **l'entraide entre collègues de formation**, notamment pour comparer nos approches, identifier quelques erreurs ou tester les comportements sur différents navigateurs. **Notre formateur** était également présent pour répondre à nos éventuelles questions et réflexions afin de nous accompagner pendant la réalisation de l'exercice,

## 4. Contexte

Nom de l'entreprise, organisme ou association ► CEFii

Chantier, atelier, service ► Exercice d'entrainement sur JavaScript

Période d'exercice ► en février 2025

## 5. Informations complémentaires (facultatif)



## Activité-type 2

### Développer la partie back-end d'une application web ou web mobile sécurisée

*Exemple n° 1 ► Mise en place d'une base de donnée relationnelle avec la méthode Merise*

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

Pour concevoir la base de données de mon projet **Festiv'All** réalisé en milieu de parcours, j'ai appliqué la **méthode Merise**, en commençant par la définition des **règles de gestion** de la base données (exemple : un utilisateur peut réserver plusieurs événements ; un événement peut être réservé par plusieurs utilisateurs, etc...). J'ai ensuite réalisé le **dictionnaire de données** pour réfléchir sur les types à attribuer aux colonnes, pour chaque entité.

Après cela, j'ai donc pu réaliser le **Modèle Conceptuel de Données (MCD)** à l'aide de l'outil **Looping**. J'ai d'abord identifié les **entités** principales liées au projet, telles que utilisateur, événement, catégorie, commentaire, et leurs **associations** (réservation, publier, commenter...). Pour chaque entité, j'ai défini les **attributs** pertinents comme email, date\_début, ou encore places\_disponibles, ainsi que les **cardinalités** précises, conformément aux règles de modélisation.

Une fois le MCD terminé, j'ai poursuivi avec la transformation en **Modèle Logique de Données (MLD)**, puis en **Modèle Physique de Données (MPD)** pour pouvoir générer le schéma SQL correspondant dans MySQL (le MLD et le MPD ont été générés automatiquement par **Looping** à partir du MCD). Cette étape m'a permis de structurer correctement la base relationnelle et de garantir la cohérence des données pour l'ensemble des fonctionnalités prévues dans l'application (réservations, commentaires, affichage des événements...).

#### 2. Précisez les moyens utilisés :

Pour réaliser cette tâche, j'ai utilisé plusieurs **outils et supports techniques**.

Le principal a été **Looping**, un logiciel de modélisation permettant de concevoir des schémas Merise, notamment le **Modèle Conceptuel de Données (MCD)**. C'est avec cet outil que j'ai créé le diagramme que j'ai joint au dossier.

J'ai également eu recours à **MySQL** (via phpMyAdmin dans mon environnement WampServer) pour traduire le MCD en **structure de base de données** relationnelle, et pour effectuer des tests sur les tables générées.

En complément, j'ai utilisé mon **ordinateur personnel**, un **navigateur web** (pour accéder à phpMyAdmin) et mon **éditeur de texte/code** habituel (comme VS Code) pour rédiger les requêtes SQL ou noter mes schémas et correspondances MCD ↔ MLD ↔ MPD.

Enfin, les supports de cours fournis par le formateur : celui concernant la **méthode Merise** et celui sur les **bases de données relationnelles**, m'ont servi de référence pour appliquer la méthodologie.

### 3. Avec qui avez-vous travaillé ?

J'ai réalisé cette tâche **principalement en autonomie**, dans le cadre d'un projet de mise en pratique du cours sur la modélisation des bases de données avec la méthode Merise et qui reprenait l'ensemble des compétences vu jusque là (milieu de parcours) en formation.

Pendant l'exercice, il y a eu parfois de **l'entraide ponctuelle** entre collègues de formation, notamment pour s'assurer que nos modélisations respectaient bien les règles de cohérence ou pour échanger sur des cas ambigus.

Dans l'ensemble, ce travail s'inscrit dans un **contexte pédagogique**, où chacun devait produire sa propre application et base de données à partir du même cahier des charges (ici, celui du projet Festiv'All).

### 4. Contexte

Nom de l'entreprise, organisme ou association ► CEFii

Chantier, atelier, service ► Projet Festiv'all

Période d'exercice ► Du : 11/12/2024 au : 19/12/2024 *(durée totale pour réaliser le projet)*

### 5. Informations complémentaires *(facultatif)*

## Activité-type 2

### Développer la partie back-end d'une application web ou web mobile sécurisée

**Exemple n° 2** ► Connexion à la base de données et manipulation de données via PDO et des requêtes SQL

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

Pour cet exemple, je devais développer les **composants d'accès aux données SQL** de mon application web *Festiv'All*, en me basant sur une architecture MVC. Cela consistait à mettre en place une **connexion sécurisée et réutilisable à la base de données** MySQL grâce à l'extension **PDO** de PHP, puis à l'utiliser pour exécuter différentes requêtes SQL depuis les modèles de l'application.

Pour cela, j'ai créé une classe *DbConnect*, utilisée comme **classe mère de tous les modèles**, qui instancie un objet PDO dans son constructeur. Cette classe assure :

- la **connexion au serveur MySQL** via les paramètres (hôte, base, utilisateur, mot de passe) ;
- la **gestion des erreurs** via PDO::ATTR\_ERRMODE en mode exception ;
- le **paramétrage du jeu de caractères** en UTF-8 ;
- et le **format de récupération des résultats** par défaut sous forme d'objet (PDO::FETCH\_OBJ), pour en simplifier la manipulation dans les vues du MVC.

Les classes modèles comme *EvenementModel* héritent de *DbConnect* et peuvent donc accéder à `$this->connection`, l'objet PDO, pour exécuter des requêtes SQL.

Par exemple :

- avec `query()` pour les requêtes simples (comme `findAll()`) ;
- avec `prepare()` et `bindParam()` pour les requêtes préparées (comme `findByCategorie()`), ce qui permet **d'éviter les injections SQL** et de gérer dynamiquement les filtres côté utilisateur.

Les résultats sont ensuite **récupérés avec `fetchAll()`** et retournés au contrôleur, ce qui permet de séparer proprement la logique de traitement des données du reste de l'application.

## 2. Précisez les moyens utilisés :

Pour développer les composants d'accès aux données de mon application *Festiv'All*, j'ai utilisé les moyens suivants :

- **Matériel :**
  - Un **ordinateur personnel** équipé de Windows 10.
- **Environnement de développement :**
  - **Visual Studio Code** comme éditeur de code principal.
  - **WampServer** pour héberger localement le serveur Apache, MySQL, et PHP.
  - Le **navigateur Chrome** pour tester l'application en local.
- **Technologies et langages utilisés :**
  - Le langage **PHP**, en particulier l'extension **PDO (PHP Data Objects)** pour gérer la connexion à la base de données et exécuter les requêtes SQL de manière sécurisée.
  - Le langage **SQL** pour interroger la base de données relationnelle (MySQL).
  - L'**architecture MVC (Modèle-Vue-Contrôleur)** pour organiser mon code : les classes PHP du modèle héritent d'une classe DbConnect qui encapsule la configuration PDO.
- **Outils complémentaires :**
  - **phpMyAdmin** pour visualiser, tester et vérifier les requêtes et le contenu de la base de données.
  - Le **navigateur web** pour tester les requêtes dynamiques envoyées depuis l'interface utilisateur vers les modèles (via les contrôleurs).
- **Documentation et cours :**
  - Les supports de cours fournis par notre formateur sur PDO et l'architecture MVC.
  - La documentation officielle de PHP pour les méthodes PDO, `prepare()`, `bindParam()` et `fetchAll()`.

### 3. Avec qui avez-vous travaillé ?

Pour réaliser la connexion à la base de données et manipuler les données via PDO et des requêtes SQL, malgré le fait que le projet *festiv'all* était un projet à réaliser en solo, j'ai néanmoins un peu échangé avec plusieurs personnes :

- **Mon responsable de formation / tuteur** : Il m'a guidé dans la compréhension des bonnes pratiques pour sécuriser les accès à la base de données, notamment sur l'utilisation des requêtes préparées avec PDO pour éviter les injections SQL. Il intervenait directement en validant mes choix techniques et en répondant à mes questions.
- **Les autres stagiaires de la formation** : Nous avons parfois échangé sur nos implémentations respectives, partagé des conseils et des exemples de code, ce qui m'a aidé à améliorer la robustesse de mes requêtes car à ce moment de la formation, je découvrais pour la première fois les requêtes SQL « JOIN ». Leur rôle était a été minime mais cependant utile pour enrichir mes solutions.

En résumé, ces interactions m'ont aidé à concevoir un accès aux données sécurisé et conforme aux exigences du projet.

### 4. Contexte

Nom de l'entreprise, organisme ou association ► CEFii

Chantier, atelier, service ► Projet Festiv'all

Période d'exercice ► Du : 11/12/2024 au : 19/12/2024 (*durée totale pour réaliser le projet*)

### 5. Informations complémentaires (*facultatif*)

## Titres, diplômes, CQP, attestations de formation

*(facultatif)*

Intitulé	Autorité ou organisme
Cliquez ici.	Cliquez ici pour taper du texte.

## Déclaration sur l'honneur

Je soussigné(e) Arnaud BERTHOLLET,

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 Angers le 25 mars 2025

pour faire valoir ce que de droit.

Signature :

*Berthollet Arnaud*

## Documents illustrant la pratique professionnelle

*(facultatif)*

Intitulé
Cliquez ici pour taper du texte.



## ANNEXES

*(Si le RC le prévoit)*