

Maison des ligues



Application de gestion de formations
(Web)

Spécifications techniques

Table des matières

Introduction :.....	3
Partie Base de données.....	4
Système de couches	4
MLD	5
Script SQL.....	6
Phpmyadmin	6
Environnement de développement	8
Logiciels utilisés	8
Langages utilisés.....	10
Frameworks	10
Gestion des bugs	11
Gestion des versions.....	11
Tests	11
Objectifs.....	11
Tests techniques.....	11
Tests fonctionnels	11

Introduction :

La mise en place de la solution applicative web de gestion des formations est soumise à des contraintes et des spécifications techniques : Langages utilisées, logiciels utilisés, structure de la base de données correspondante, moyens de gestions de bugs et de versions.

Ce document a pour but de lister les éléments ci-dessus et d'expliquer les différents choix techniques mis en place dans le cadre de la réalisation de la solution applicative.

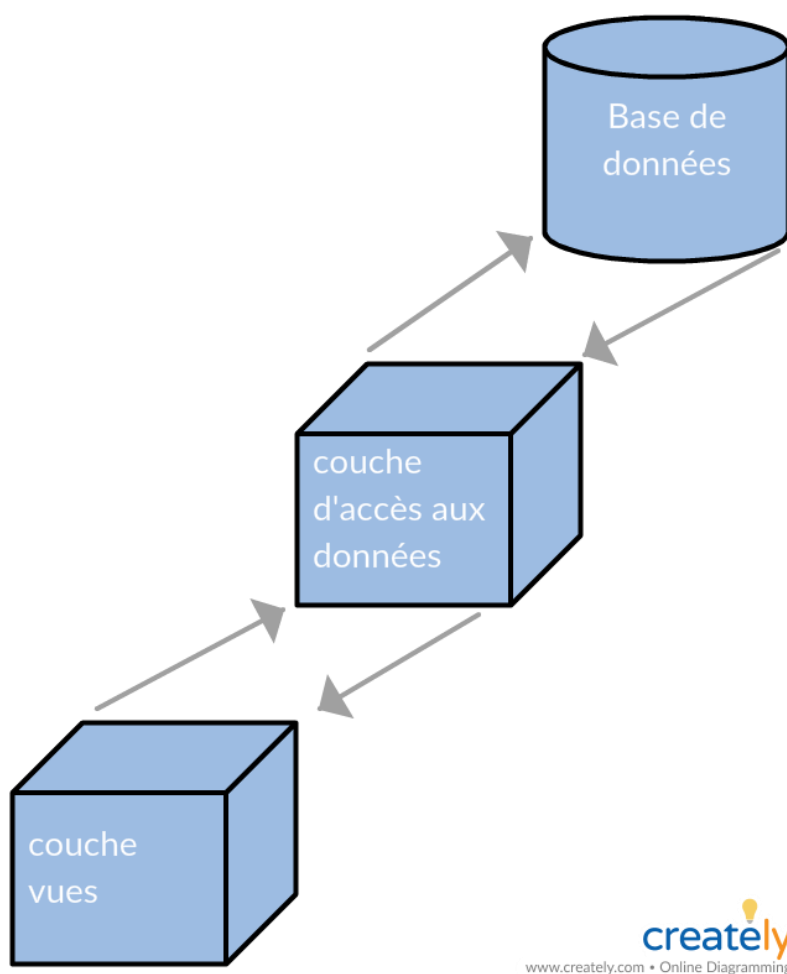
Partie Base de données

Dans cette partie sera mis en avant les choix techniques liés à la base de données.

Tout d'abord, le logiciel **Jmerise** a été utilisé pour mettre en place le MCD (modèle conceptuel de données) qui représente la structure de la base de données. A la suite de ce MCD, le MLD (modèle logique de données) est mis en place et il est possible d'exporter la structure de la base de données sur Phpmyadmin et MySQLWorkbench.

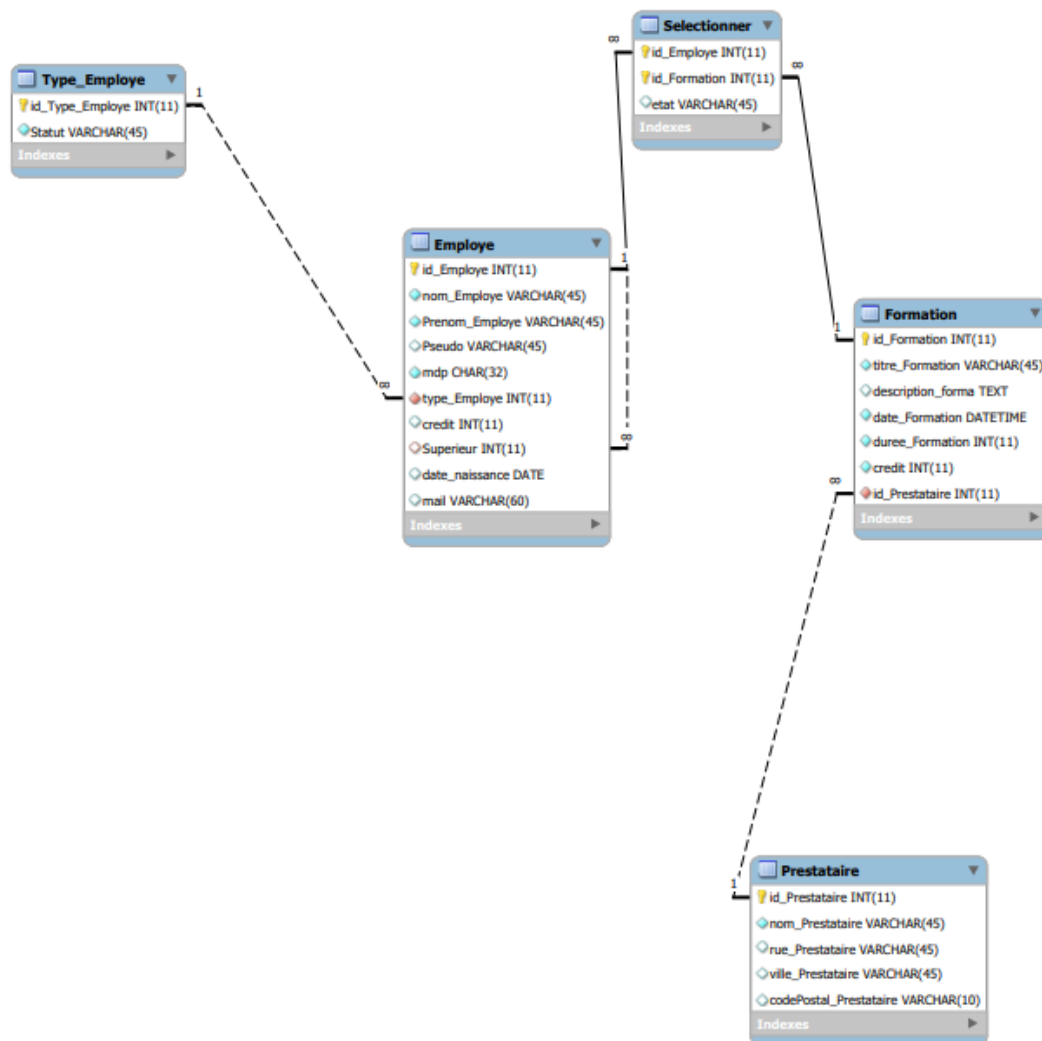
Voici le MLD obtenu suite à l'importation de la base sur MySQLWorkbench

Système de couches



Couches de données

MLD



Voici un extrait du script SQL ayant permis la conception de la base de données

Script SQL

```

22 -- Table `u668252900_m21`.`Employee`
23 -- -----
24 CREATE TABLE IF NOT EXISTS `u668252900_m21`.`Employee` (
25   `id_Employe` INT(11) NOT NULL AUTO_INCREMENT,
26   `nom_Employe` VARCHAR(45) CHARACTER SET 'utf8' COLLATE 'utf8_unicode_ci' NOT NULL,
27   `Prenom_Employe` VARCHAR(45) CHARACTER SET 'utf8' COLLATE 'utf8_unicode_ci' NOT NULL,
28   `Pseudo` VARCHAR(45) CHARACTER SET 'utf8' COLLATE 'utf8_unicode_ci' NULL DEFAULT NULL,
29   `mdp` CHAR(32) CHARACTER SET 'utf8' COLLATE 'utf8_unicode_ci' NOT NULL,
30   `type_Employe` INT(11) NOT NULL,
31   `credit` INT(11) NULL DEFAULT '3000',
32   `Superieur` INT(11) NULL DEFAULT NULL,
33   `date_naissance` DATE NULL DEFAULT NULL,
34   `mail` VARCHAR(60) CHARACTER SET 'utf8' COLLATE 'utf8_unicode_ci' NULL DEFAULT NULL,
35   PRIMARY KEY (`id_Employe`),
36   UNIQUE INDEX `Pseudo` (`Pseudo` ASC),
37   INDEX `fk_Employe_Type_Employe_idx` (`type_Employe` ASC),
38   INDEX `fk_Employe_Employe1_idx` (`Superieur` ASC),
39   CONSTRAINT `fk_Employe_Employe1`
40     FOREIGN KEY (`Superieur`)
41     REFERENCES `u668252900_m21`.`Employee` (`id_Employe`)
42     ON DELETE NO ACTION
43     ON UPDATE NO ACTION,
44   CONSTRAINT `fk_Employe_Type_Employe`
45     FOREIGN KEY (`type_Employe`)
46     REFERENCES `u668252900_m21`.`Type_Employe` (`id_Type_Employe`)
47     ON DELETE NO ACTION
48     ON UPDATE NO ACTION)
49 ENGINE = InnoDB
50 AUTO_INCREMENT = 6
51 DEFAULT CHARACTER SET = utf8
52 COLLATE = utf8_unicode_ci;
53
54

```

La gestion globale de la base de données se fait sur Phpmyadmin, ainsi les tests concernant les insertions / updates / deletes des tables se sont fait via cet outil dans un soucis de simplification et de rapidités.

Phpmyadmin













Récentes | Préférences

information_schema
u668252900_m21
Nouvelle table
Employee
Formation
Prestataire
Selectionner
Type_Employe

Colonne	Type	Fonction	Null	Valeur
id_Employe	int(11)			
nom_Employe	varchar(45)			
Prenom_Employe	varchar(45)			
Pseudo	varchar(45)		<input checked="" type="checkbox"/>	
mdp	char(32)			
type_Employe	int(11)			
credit	int(11)		<input type="checkbox"/>	3000
Superieur	int(11)		<input checked="" type="checkbox"/>	
date_naissance	date		<input checked="" type="checkbox"/>	
mail	varchar(60)		<input checked="" type="checkbox"/>	

Exécuter

☒ Ignorer

+ Options				id_Employe	id_Formation	etat
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	1	2	validé
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	1	3	validé
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	3	1	en attente
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	3	2	en attente

La création des requêtes adaptées à la solution applicative se fait sur MySQLWorkbench dans un souci d'efficacité et de clarté.

De plus Phpmyadmin et Mysqlworkbench ont été choisi suivant un critère de gratuité et de facilité d'utilisation.

Environnement de développement

Dans cette partie, les langages utilisés ainsi que les frameworks associés, les logiciels utilisés, les outils de gestion de bugs et de versions seront explicités.

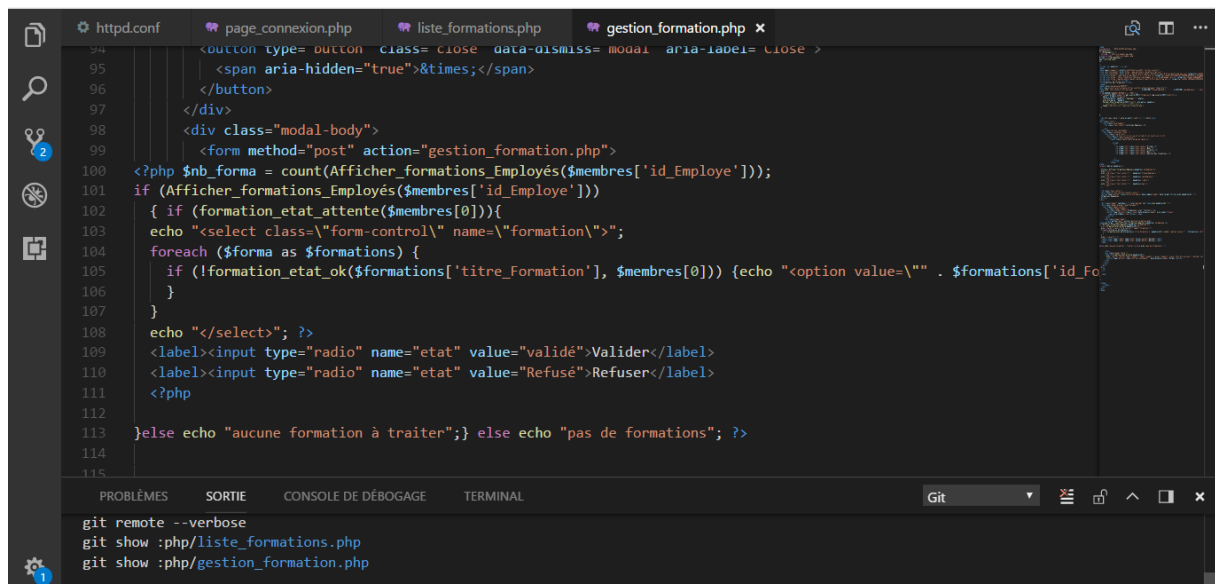
Logiciels utilisés

Editeur de code : Visual Studio Code

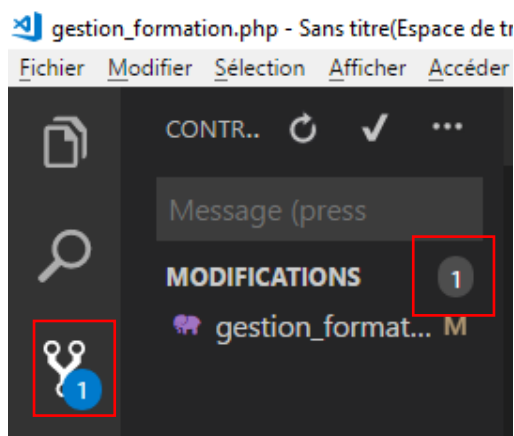
Visual Studio Code développé par Microsoft, gratuit, disposant de la possibilité d'ajouter des extensions et pouvant permettre la gestion de version via Github directement sur son interface.

Visual studio Code a été utilisé pour éditer et gérer le code composant l'application web.

Des extensions permettant notamment de gérer l'indentation ont été ajoutées.



Visual Studio Code

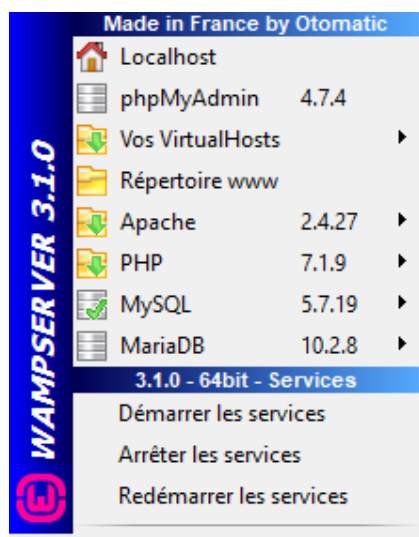


Gestion de version : Github

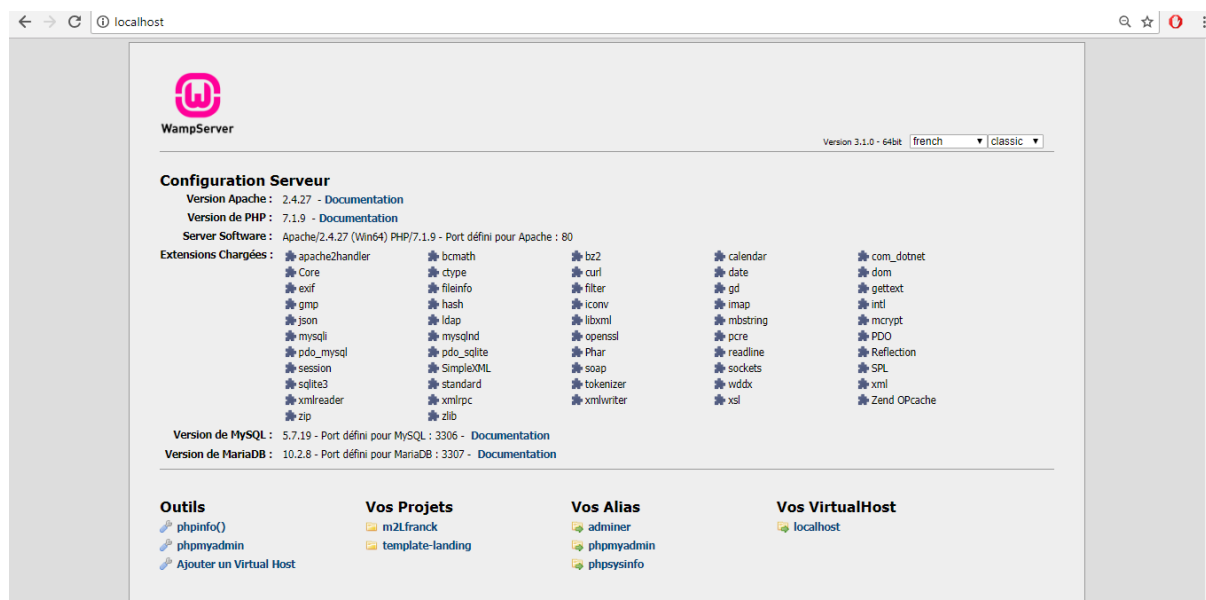
Environnement de test : WAMP (Windows, Apache, Mysql, Php)

La solution applicative Web en développement est hébergée sous WAMP, gratuit et simple d'utilisation.

Les fichiers nécessaires étant placés dans le répertoire WWW, l'accès au site se faisant via l'utilisation dans le navigateur de **localhost / nom du site**.



WAMP



Localhost

Langages utilisés

HTML5 :

La structure du site est réalisée via le langage HTML5.

CSS4 :

L'aspect graphique du site es en partie réalisée via le CSS, Bootstrap permettant a réalisation de la majorité de l'aspect graphique (voir Frameworks).

PHP : PHP Hypertext Preprocesseur (version : 7.1.9)

PHP est utilisé comme langage de communication avec le serveur hébergeant la base de données. Ce langage a été utilisé de part la facilité de son utilisation et sa grande permissivité.

La classe PDO (PHP data object) a été utilisée pour permettre la communication avec la base de données via des requêtes préparées.

Le site PHP.net a servis de documentation principale.

SQL :

Langage de la base de données.

Frameworks

Bootstrap : Version 4

Bootstrap est un Framework de type front-end (aspect graphique) permettant notamment de rendre les applications Web responsive (utilisable sur smartphone, tablettes, etc.).

Gratuit et facile d'accès il a permit de réaliser la majorité de la partie graphique de l'application Web.

SweetAlert : JS

SweetAlert a été utilisé pour modifier l'aspect graphique des alertes utilisées en JavaScript afin de rendre ses dernières en adéquation avec l'aspect graphique de l'application.

Gestion des bugs

La gestion des bugs est réalisée par des réunions hebdomadaires entre développeurs, les version de test des applications étant disponibles librement.

La gestion des bugs est supervisée via l'outil GitHub, avec le système d'issues.

Gestion des versions

La gestion des versions de l'application Web se fait sous GitHub (<https://github.com/>).

L'interface Code / Github est notamment facilitée grâce à l'utilisation de Visual Studio Code.

Tests

Objectifs

L'intérêt de réaliser des tests est de mesurer le bon fonctionnement de notre application sur le plan technique et fonctionnel.

Nous procéderont à 2 types de tests, les tests dits techniques et les tests fonctionnels.

Tous les tests seront supervisés et réalisés par le développeur.

Les tests concerneront la version finale de l'application en local.

Tests techniques

Ils ont pour but de vérifier le bon fonctionnement des méthodes utilisées en PHP, et de vérifier si la valeur retournée par exemple est bonne ainsi que son type.

Les tests techniques ont été réalisés sans outils spécifiques.

Ils ont été réalisés à l'aide de l'utilisation de variables de tests implémentées dans le code et affichés à des endroits prévus à cet effet.

Par exemple : la méthode `CreditEmploye()` a été testée en comparant sa valeur de retour avec une variable dont la valeur était préétablie.

La méthode `formation_etat_attente($id)` a été testée en comportant la chaîne de retour avec des chaînes préétablies (exemple : « en attente », « enattente »).

Tests fonctionnels

Les tests fonctionnels sont établis et réalisés par le développeur aux moyens de fiches de tests.

Ici on testera :

- La réussite de connexion à un compte
- La fluidité de navigation entre les différents onglets
- L'affichage des informations relatives à une formation
- La récupération du fichier PDF relatif aux formations
- La sélection d'une formation avec un nombre de crédits suffisant
- La sélection de plusieurs formations avec un nombre de crédits suffisant

- La non sélection de formations avec un nombre de crédits insuffisant.
- La décrémentation des crédits en cas de sélection de formations
- L'affichage des formations sélectionnées
- La validation de formations par le manager
- Le refus de formations par le manager et la réaffectation des crédits
- L'affichage des informations personnelles
- La déconnexion avec destruction de la session

Voici des exemples de tests :

Connexion à un compte utilisateur

Fonctionnalité à tester	Connexion à un compte utilisateur	Version	Version finale
Projet	M2Lfranck	Période	mai 2018
Date	24/05/2018		

Procédure de test :

- Lancer le navigateur de votre choix
- Accéder à la ressource en ligne en utilisant l'url à disposition
- Entrer un pseudonyme ou un mot de passe incorrect
- Appuyer sur **se connecter**

Résultat(s) attendu(s) :

- Non connexion
- Affichage d'un message d'erreur
- Réinitialisation des champs

Résultat(s) obtenu(s) :

Echec de la connexion : votre pseudo ou votre mot de passe est incorrect.

Vous connecter

Se connecter

Conclusion : Réussite du test.

Sélection d'une formation avec nombre de crédits suffisant

Fonctionnalité à tester	Sélection d'une formation	Version	Version finale
Projet	M2Lfranck	Période	mai 2018
Date	25/05/2018		

Procédure de test :

- Lancer le navigateur de votre choix
- Accéder à la ressource en ligne en utilisant l'url à disposition et se connecter
- Aller dans l'onglet liste des formations et sélectionner une formation
- Appuyer sur **Valider**

Résultat(s) attendu(s) :

- Message de confirmation
- Retour à la page d'accueil, affichage de la formation
- Déduction des crédits et suppression de la formation dans la liste de formations

Résultat(s) obtenu(s) :

The image shows two parts of the application interface. On the left, a modal window titled 'Confirmation' with a green circular progress indicator. The text inside says 'Vous avez sélectionné 1 formation(s)' and there is an 'OK' button. Below this is the label 'message de confirmation'. On the right, a vertical flow diagram shows the credit balance. It starts with a blue box 'Il vous reste 2200 crédits' labeled 'crédits initiaux'. An arrow points down to another blue box 'Il vous reste 1200 crédits' labeled 'crédits finaux'.

Formation	Description	Date de la formation	durée de la formation (en jours)	prestataire
Bureautique	Plus d'infos	22-08-2018	3	Bureautique-plus
Base programmation	Plus d'infos	24-09-2018	5	Dev-pack

liste des formations sélectionnées

Conclusion : Réussite du test.