Documentation Site Web Ligue 1

Sommaire

[Documentation Site Web Ligue 1 1](#_Toc164925674)

[Sommaire 1](#_Toc164925675)

[1. Introduction 2](#_Toc164925676)

[1.1. Différence entre Front-end et back-end 2](#_Toc164925677)

[2. Prérequis 3](#_Toc164925678)

[3. Installation et configuration 3](#_Toc164925679)

[4. Structure du projet 3](#_Toc164925680)

[4.1. Présentation rapide du modèle MVC 3](#_Toc164925681)

[4.2. Détails du dossier src 3](#_Toc164925682)

[4.2.1. Détails du dossier model 3](#_Toc164925683)

[4.2.2. Détails du dossier view 3](#_Toc164925684)

[4.2.3. Details du dossier control 3](#_Toc164925685)

[4.2.4. Details du dossier script 3](#_Toc164925686)

[4.2.5. Détails du dossier img 3](#_Toc164925687)

[4.2.6. Détails du dossier Css 3](#_Toc164925688)

[5. La Base de données 3](#_Toc164925689)

[6. Fonctionnement du site web 4](#_Toc164925690)

[7. Contact 4](#_Toc164925691)

[8. Conclusion 4](#_Toc164925692)

1. Introduction

Lors de ma 2ᵉ année de BTS SIO SLAM, nous avons étudié les languages Web (HTML, CSS, PHP, JavaScript), mais nous nous sommes particulièrement intéressés au côté back-end de la programmation au travers d’un projet Web. Ce projet consiste à créer un site internet de foot sur la ligue 1, comportant différentes pages et interrogeant une Base de données PostgreSQL.

Nous avons travaillé sous le modèle MVC qui est une bonne façon de travailler, car elle permet de répartir les logiques de code dans différentes classes.

* 1. Différence entre Front-end et back-end

Front-end : Imaginez que vous êtes dans un restaurant. Le front-end serait la salle de restaurant où vous vous asseyez, le menu que vous parcourez, et le serveur qui prend votre commande. En termes de développement web, le front-end est tout ce que l’utilisateur voit et avec quoi il interagit sur le site web. Cela comprend la conception, la mise en page, les couleurs, les boutons, les formulaires et tout autre élément visuel. Les technologies couramment utilisées dans le développement front-end comprennent HTML, CSS et JavaScript.

Back-end : En revenant à notre analogie du restaurant, le back-end serait la cuisine, où le chef prépare votre repas, et tous les processus qui se déroulent en coulisses pour s’assurer que votre repas arrive à votre table comme vous l’avez commandé. Dans le développement web, le back-end fait référence à tout ce qui se passe en coulisses d’un site web. Cela comprend la gestion des bases de données, la logique métier (comme les calculs), la sécurité, l’authentification, et tout ce qui est nécessaire pour traiter les demandes de l’utilisateur et renvoyer les bonnes données. Les technologies couramment utilisées dans le développement back-end comprennent Node.js, Ruby, Python, PHP, Java, et d’autres.

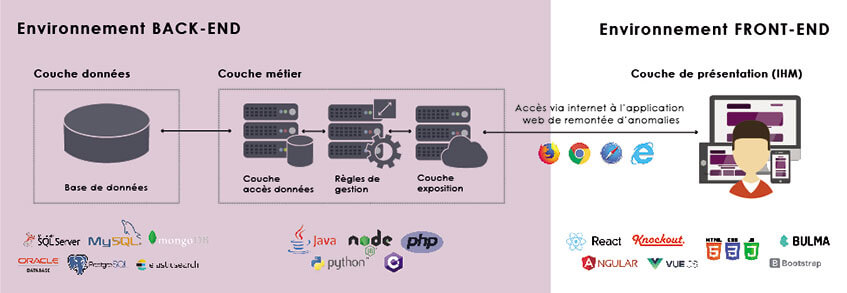


Figure 1 : schéma représentatif de la différence entre back-end et front-end

La différence principale entre le front-end et le back-end réside donc dans leur rôle et leur interaction avec l’utilisateur. Le front-end est tout ce que l’utilisateur voit et interagit, tandis que le back-end est tout ce qui se passe en coulisses pour rendre cela possible.

1. Prérequis

Le site internet a été développé en utilisant les langages de programmation principaux du web, à savoir PHP, HTML et JavaScript. Actuellement, le site fonctionne en local car il n’est pas encore hébergé sur un serveur web. Pour la gestion des données, nous utilisons une base de données PostgreSQL, également en local.

Le code source du site est disponible sur GitHub. Pour y accéder, veuillez cliquer [ici](https://github.com/Paul-berne/Site-foot).

1. Installation et configuration

Pour pouvoir accèder au site web et profiter de celui-ci, je vais vous présenter comment pouvoir faire tourner un site web multi page sur une machine sur Microsoft Visual Studio, via une commande.

Vous devez avoir télécharger PHP préalablement puis avoir mis dans les variables d’environnement le path vers le dossier de PHP que vous venez de télécharger.

Ensuite il vous suffit de lancer Visual studio et d’ouvrir le projet, pour ensuite dans le terminal entrer cette commande : « php -S localhost :3000 -t src »

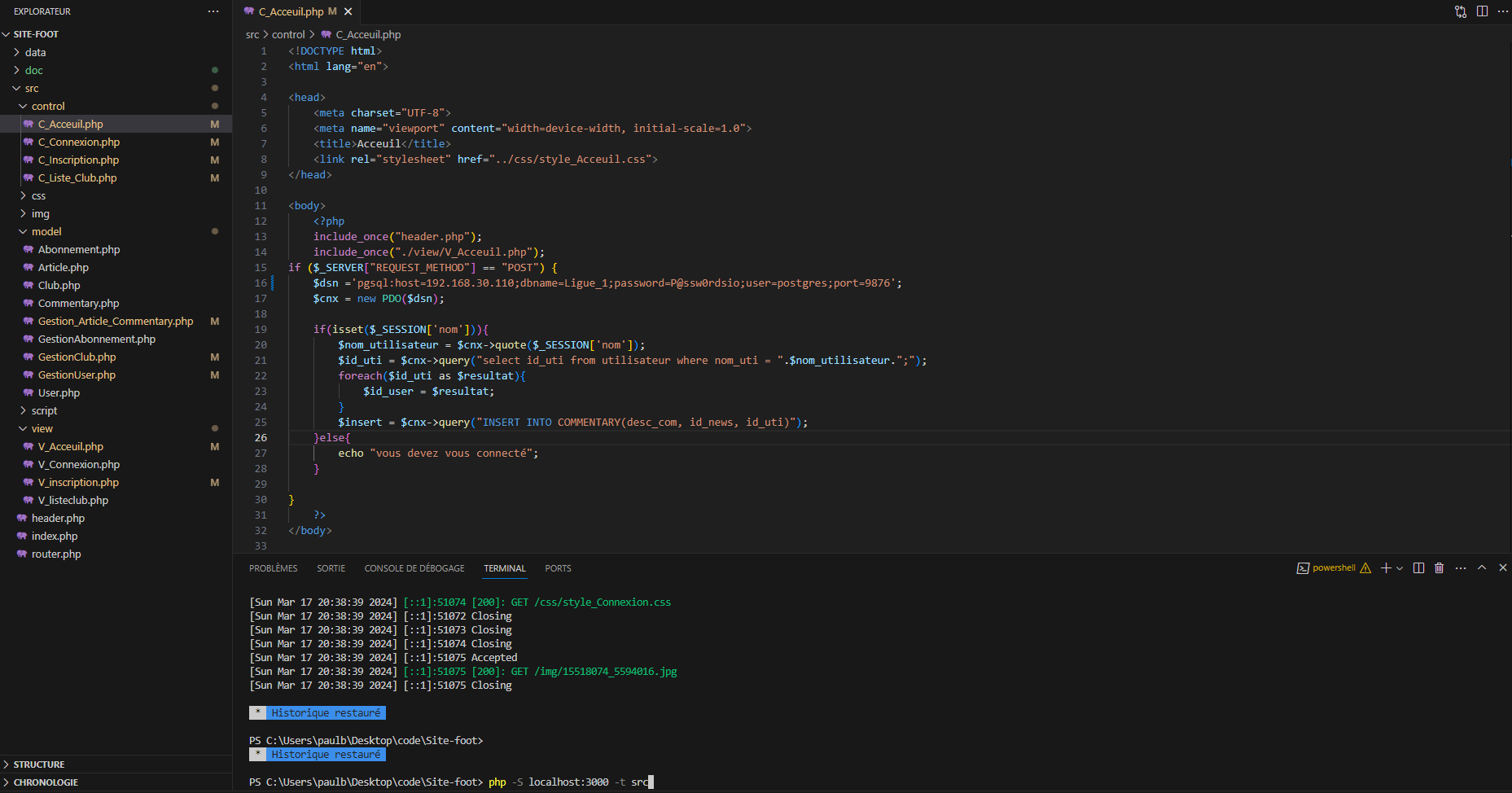


Figure 2 : Screenshot de VisualStudio

Ensuite vous pouvez accéder au site internet via le lien « localhost :3000 ».

1. Structure du projet

Maintenant nous verrons toute la structure du code derrière le site internet.

* 1. Présentation rapide du modèle MVC

Le modèle MVC est composé de 3 grands packages le modèle, la vue et le contrôleur, chacun a son rôle dans une application, voici leurs définitions respectives :

**Modèle** : Dans le modèle MVC, le “Modèle” représente les données et les règles métier de votre application. Il interagit avec la base de données pour créer, lire, mettre à jour et supprimer des informations.

**Vue** : La “Vue” est responsable de la présentation des données à l’utilisateur. Elle prend les données du modèle et les affiche d’une manière que l’utilisateur peut comprendre.

**Contrôleur** : Le “Contrôleur” agit comme un intermédiaire entre le modèle et la vue. Il traite les entrées de l’utilisateur, interagit avec le modèle pour obtenir ou modifier des données, et met à jour la vue en conséquence.

* 1. Détails du dossier src

Le dossier src comporte tout le code du projet et permet d’accèder a toute la logique et le fonctionnement du site internet.

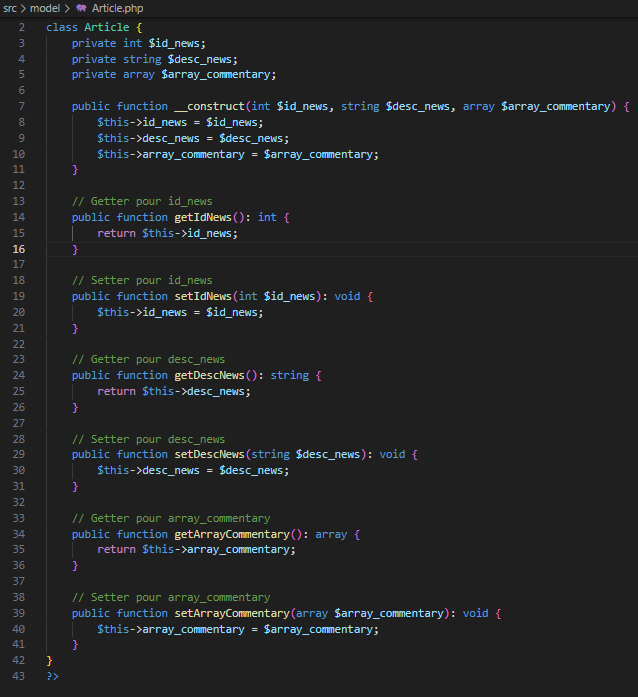
* + 1. Détails du dossier model

Le dossier model comme expliquer précedemment comporte toutes les classes Modèle, c’est-a-dire que les classes modèle vont intéragir avec la base de donnée, il y’aura des classes qui ont des paramètres réprésentant des tables sur la base de donnée.

* + - 1. La classe Article

Elle représente un article avec des commentaires. Voici les détails :

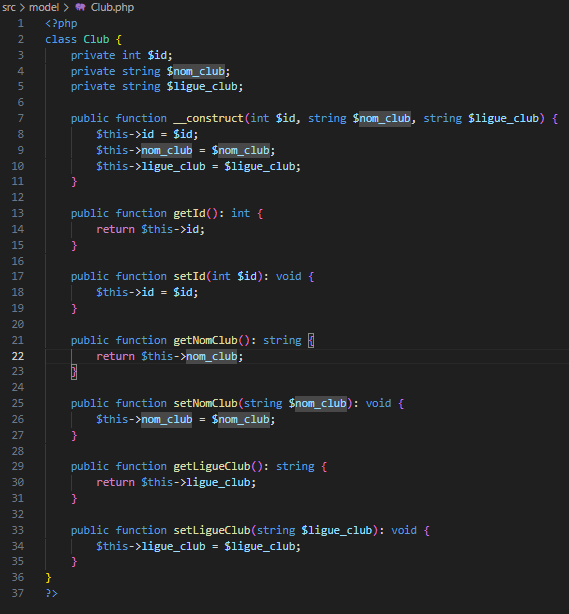
* Propriétés : La classe a trois propriétés privées :
  + $id\_news : Un entier qui représente l’identifiant unique de l’article.
  + $desc\_news : Une chaîne de caractères qui représente la description de l’article.
  + $array\_commentary : Un tableau qui contient les commentaires associés à l’article.
* Constructeur : Le constructeur de la classe prend trois paramètres ($id\_news, $desc\_news, $array\_commentary) qui sont utilisés pour initialiser les propriétés de l’objet lors de sa création.
* Getters : Les méthodes getIdNews(), getDescNews() et getArrayCommentary() sont des getters. Elles permettent d’accéder aux valeurs des propriétés privées de l’objet.
* Setters : Les méthodes setIdNews(), setDescNews() et setArrayCommentary() sont des setters. Elles permettent de modifier les valeurs des propriétés privées de l’objet.



* + - 1. La classe Club

Elle représente un club de football dans une application web. Voici les détails :

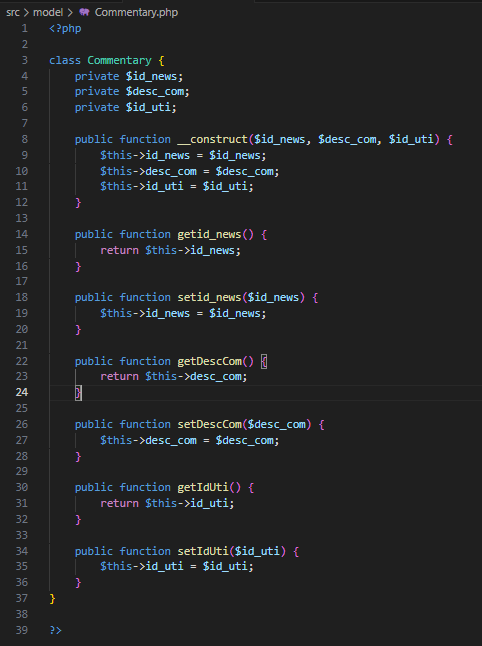
* Propriétés : La classe a trois propriétés privées :
  + $id : Un entier qui représente l’identifiant unique du club.
  + $nom\_club : Une chaîne de caractères qui représente le nom du club.
  + $ligue\_club : Une chaîne de caractères qui représente la ligue dans laquelle le club joue.
* Constructeur : Le constructeur de la classe prend trois paramètres ($id, $nom\_club, $ligue\_club) qui sont utilisés pour initialiser les propriétés de l’objet lors de sa création.
* Getters : Les méthodes getId(), getNomClub() et getLigueClub() sont des getters. Elles permettent d’accéder aux valeurs des propriétés privées de l’objet.
* Setters : Les méthodes setId(), setNomClub() et setLigueClub() sont des setters. Elles permettent de modifier les valeurs des propriétés privées de l’objet.



* + - 1. La classe Commentary

La classe s’appelle Commentary. Voici les détails :

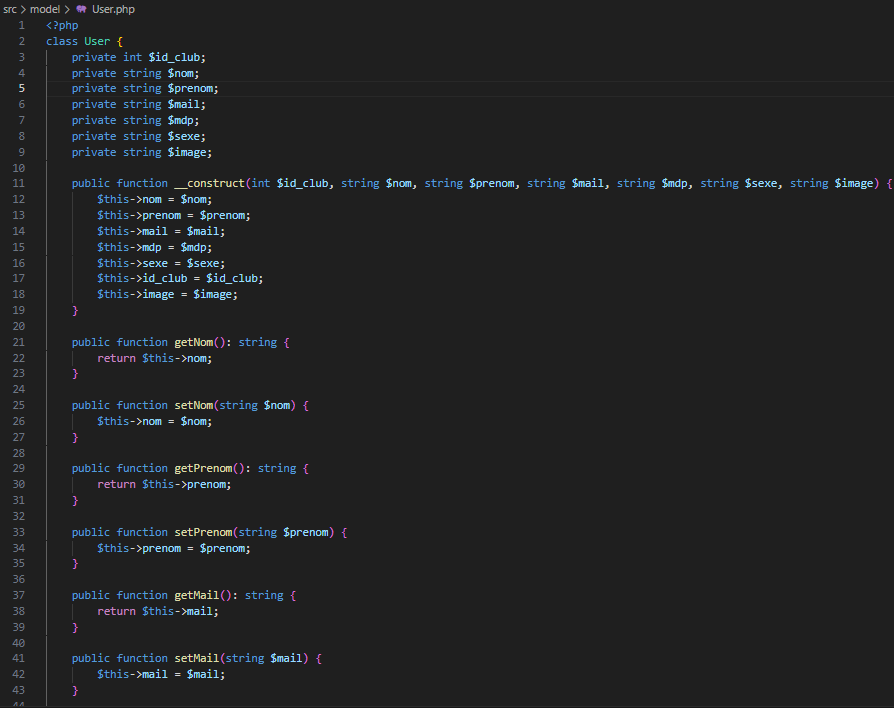
* Propriétés : La classe a trois propriétés privées :
  + $id\_news : Il s’agit probablement de l’identifiant de l’article auquel le commentaire est associé.
  + $desc\_com : Il s’agit de la description du commentaire, c’est-à-dire le texte du commentaire lui-même.
  + $id\_uti : Il s’agit probablement de l’identifiant de l’utilisateur qui a posté le commentaire.
* Constructeur : Le constructeur de la classe prend trois paramètres ($id\_news, $desc\_com, $id\_uti) qui sont utilisés pour initialiser les propriétés de l’objet lors de sa création.
* Getters : Les méthodes getid\_news(), getDescCom() et getIdUti() sont des getters. Elles permettent d’accéder aux valeurs des propriétés privées de l’objet.
* Setters : Les méthodes setid\_news(), setDescCom() et setIdUti() sont des setters. Elles permettent de modifier les valeurs des propriétés privées de l’objet.



* + - 1. La classe User

La classe s’appelle User. Voici les détails :

* Propriétés : La classe a sept propriétés privées :
  + $id\_club : Il s’agit de l’identifiant du club auquel l’utilisateur est associé.
  + $nom : Il s’agit du nom de l’utilisateur.
  + $prenom : Il s’agit du prénom de l’utilisateur.
  + $mail : Il s’agit de l’adresse mail de l’utilisateur.
  + $mdp : Il s’agit du mot de passe de l’utilisateur.
  + $sexe : Il s’agit du sexe de l’utilisateur.
  + $image : Il s’agit de l’image de profil de l’utilisateur.
* Constructeur : Le constructeur de la classe prend sept paramètres ($id\_club, $nom, $prenom, $mail, $mdp, $sexe, $image) qui sont utilisés pour initialiser les propriétés de l’objet lors de sa création.
* Getters : Les méthodes getNom(), getPrenom(), getMail(), getMdp(), getSexe(), getId\_club() et getImage() sont des getters. Elles permettent d’accéder aux valeurs des propriétés privées de l’objet.
* Setters : Les méthodes setNom(), setPrenom(), setMail(), setMdp(), setSexe(), setId\_club() et setImage() sont des setters. Elles permettent de modifier les valeurs des propriétés privées de l’objet



* + 1. Détails du dossier view

Le dossier View comporte les Classe PHP qui sont le front-end des pages du site. Elles sont la pour contenir le code de ce que l’utilisateur voit.

* + - 1. La classe V\_Acceuil
      2. La classe V\_Connexion
      3. La classe V\_inscription
      4. La classe V\_inscription
      5. La classe V\_listeclub
    1. Details du dossier control
    2. Details du dossier script
    3. Détails du dossier img
    4. Détails du dossier Css

1. Fonctionnement du site web
2. Contact
3. Conclusion