# **L’ARCHITECTURE MVC DU BACK-END**

## **SCHEMA**

***API***

***Node JS***

***SITE***

Language de programmation utiliser:

* HTML
* CSS
* JS

**DATABASE**

* **Table**
* **Procedure stocker**

***BACK-END***

Language de Programmation utiliser:

* PHP
* JS

## **Explication de la structure**

* **API**

Le rôle de l'API (Interface de Programmation Applicative) entre une base de données (BDD) et un site web (frontend et backend) est de permettre la communication et l'échange de données entre ces différentes composantes d'une application. Voici comment cela fonctionne :

**1. Communication entre le backend et la BDD :** L'API permet au backend de communiquer avec la base de données en envoyant des requêtes pour récupérer, insérer, mettre à jour ou supprimer des données stockées dans la BDD. L'API agit comme une interface qui expose les fonctionnalités de la BDD de manière sécurisée et contrôlée.

**2. Communication entre le frontend et le backend :** L'API permet au frontend de communiquer avec le backend en envoyant des requêtes pour récupérer des données à afficher sur le site web, ou pour envoyer des données saisies par l'utilisateur au backend pour traitement. Cela permet une séparation claire des responsabilités entre le frontend (interface utilisateur) et le backend (logique métier et accès aux données).

**3. Format des données échangées :** L'API définit les formats de données acceptés et retournés par les différentes requêtes. Il peut s'agir de formats tels que JSON, XML, ou d'autres formats structurés. Cela garantit une communication cohérente et normalisée entre les différentes parties de l'application.

En résumé, l'API agit comme un pont entre la base de données, le backend et le frontend d'une application, facilitant ainsi l'échange de données et assurant une communication efficace et sécurisée entre ces composantes.

* **BDD**

La base de données (BDD) d'un site web joue un rôle essentiel dans le stockage et la gestion des données nécessaires au bon fonctionnement du site. Elle permet de stocker des informations telles que les utilisateurs, les articles, les commandes, etc. La BDD permet d'organiser et de structurer les données de manière à ce qu'elles puissent être facilement récupérées, modifiées et supprimées par le site web.

En résumé, la BDD d'un site web est essentielle pour assurer le stockage et la manipulation efficace des données du site.

* **BACK-END**

Le backend d'un site web est la partie invisible qui assure le bon fonctionnement et la gestion des fonctionnalités du site. Il se charge du traitement des requêtes des utilisateurs, de l'accès à la base de données, de la logique métier, de l'authentification des utilisateurs, et de la génération des pages web dynamiques à afficher côté client.

En somme, le backend est responsable de la gestion et du traitement des données en coulisses pour fournir une expérience utilisateur fluide et interactive sur le site web.

* **FRONT-END**

Le frontend d'un site web est la partie visible et interactive avec laquelle les utilisateurs interagissent directement. Il est responsable de l'interface utilisateur, du design, de la mise en page, des animations et de tout ce que les utilisateurs voient et avec quoi ils interagissent sur le site. Le frontend communique avec le backend pour récupérer et afficher les données de manière conviviale pour les utilisateurs.

En résumé, le frontend est chargé de présenter le contenu d'un site web de manière attrayante et intuitive pour les visiteurs.