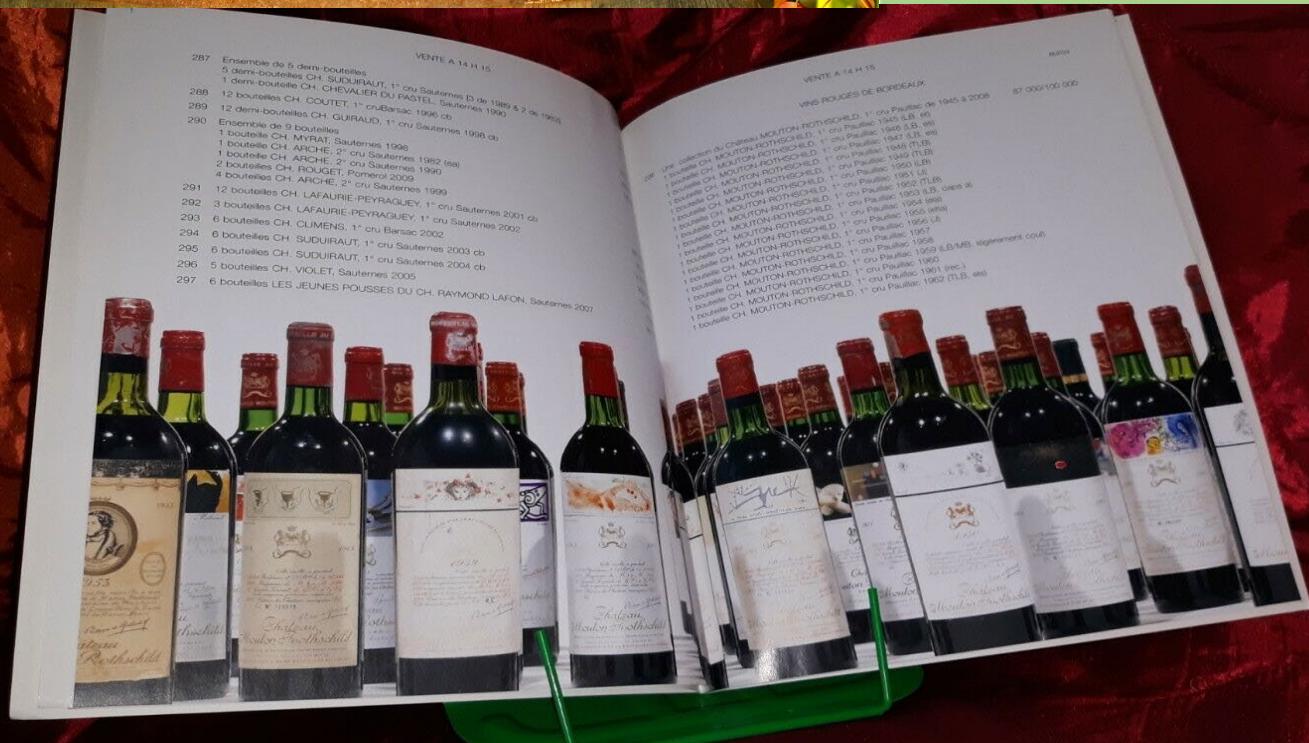


Projet de Cave à Vin

2023

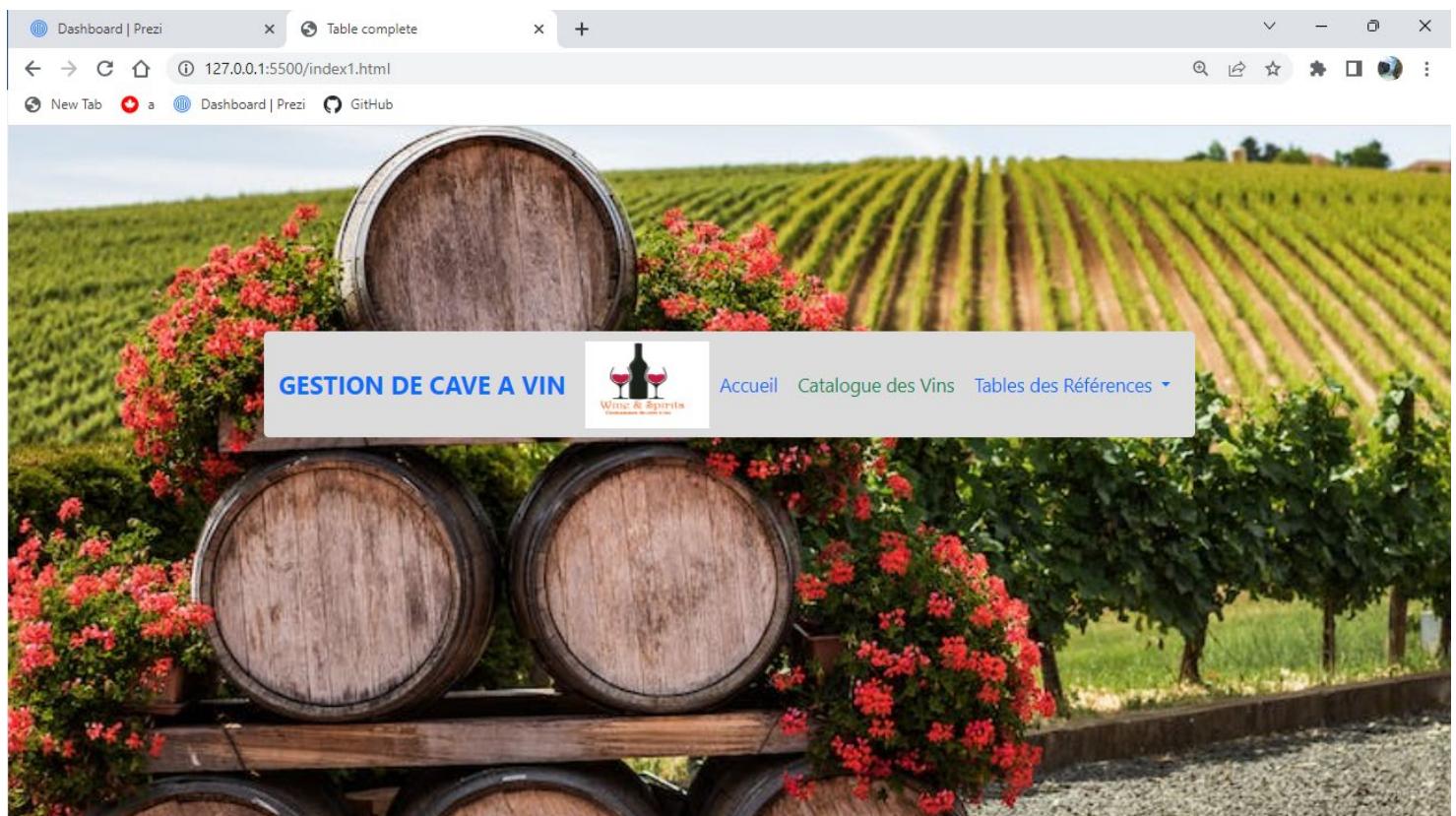


Carmela Urbano

Formation Developpeur Web

26/07/2023

PROJET DE GESTION D'UNE CAVE A VIN



Introduction

1	Compétences couvertes.....	3
2	Résumé du projet.....	3
3	Cahier des charges.....	4
	3.1 Maquette de Présentation.....	4
4	Spécifications Technique.....	6
	4.2 Schéma de la base de données.....	7
	4.1 Comment fonction une API Rest ?.....	9
	4.3 La structure des Tables.....	10
	4.4 L'accès aux tables.....	13
	4.5 Les clés des tables.....	14
5	Rédaction du Projet :	15
	5.1 Organisation des dossiers et fichiers.....	15
	5.2 Le Programme.....	16
	5.3 Visuel de l'interface sur écrans.....	22
6	Utilisation du programme de Gestion des Vins.....	23
	6.1 L'onglet 'Catalogue des vins.....	24
	6.2 Comment saisir un nouveau vin.....	25
	6.3 Comment apporter des modification sur un vin.....	27
	6.4 Comment supprimer un vin.....	28
	6.5 Comment voir la liste des vins.....	28
	6.6 Comment rechercher un critère de vin.....	29
7	L'onglet « Tables des Références ».....	30
	7.1 Gestion du tableau « Cépage ».....	30
	7.2 Gestion du tableau « Région ».....	33
	7.3 Gestion du tableau « Pays ».....	35
	7.4 Gestion du tableau « Appellation ».....	37
	7.5 Gestion du tableau « Couleur ».....	39
8	Tests du programme.....	40
9	La veille technique.....	41
10	Bilan.....	42

1. Liste des compétences couvertes par le projet

Wine & Spirits est un projet front-end qui consiste en la gestion d'une cave à vin. Ce projet permet de mettre en évidence plusieurs compétences techniques dans le titre professionnel développeur web - web mobile. Les compétences sont inclut la capacité de maquetter une application afin de permettre au client et au professionnel de visualiser une ébauche du projet et de pouvoir se mettre d'accord quant au visuel. La compétence de réaliser une interface utilisateur web statique et adaptable, créer une interface visuelle tout en la rendant compatible avec différents appareils et tailles d'écran. Enfin, le développement d'une interface utilisateur web dynamique, ce qui implique la création d'éléments interactifs au sein de l'interface pour une expérience utilisateur plus riche et fluide.

2. Résumé de Projet

Ce site de gestion de cave à vins est réalisé dans le cadre d'un projet de développement de la partie front-end d'une application web & web mobile. Un client Wine & Spirits me demande de réaliser un site pour gérer sa cave à vin. Il s'agit de concevoir un site internet pour aboutir à une interface graphique. Celle-ci sera ensuite transformée sous la forme de code exécutable par un navigateur.

L'objectif de ce projet est de réaliser un interface web statique et dynamique, analyse et satisfaire la demande de client souhaitant pouvoir gérer et enrichir la base de données de ses vins ainsi que la conception des composants front-end et back-end.

L'utilisateur doit pouvoir rentrer dans la base de donnée, créé auparavant, les bouteilles de sa cave à vin. Il doit pouvoir visualiser le contenu de sa cave à un instant donné, avec toutes les informations concernant ses vins, mais également la possibilité d'ajouter de nouvelles bouteilles, la modifier ou d'en retirer.

Le site facilitera la recherche de vins spécifiques en fonction de divers critères tels que le nom du vin, la région viticole, le cépage, l'appellation, et la couleur.

Il y a plusieurs avantages de ce projet, il permettra le gain du temps, l'efficacité et améliorer la gestion des stocks.

Ce projet se réalise avec une application de codage prenant en charge le code HTML, le code CSS, le code JavaScript. On va utiliser la méthode CRUD (Create - créer, Request- lire, Update – mettre à jour, Delete- supprimer) pour pouvoir mettre à jour les données. En plus, je vais utiliser l'API REST les demandes AJAX format JSON pour la mise à jour dynamique de l'interface utilisateur sans recharger la page entière dans le navigateur. Les applications fonctionnent plus rapidement et sont plus réactives aux actions de l'utilisatrice ou de l'utilisateur.

3. CAHIER DES CHARGES

Un client Wine & Spirits souhaite commander une application pour gérer sa cave sous la forme d'une base de données avec des formulaires adaptés. Les fonctions souhaitées sont les suivantes :

1. Listing des bouteilles

On souhaite pouvoir visualiser la liste des vins à jour à tout moment en tenant compte des mouvements des bouteilles effectuées.

2. Ajouter une bouteille

Lorsque le client acquiert de nouvelles bouteilles de vin, cela doit être pris en compte dans le logiciel de gestion. Plusieurs caractéristiques sont à saisir sur ce nouveau vin : la région dont il est issu, son appellation, le nom du domaine, les informations sur le vin en particulier (cuvée, cultures et couleur). Pour faciliter la saisie de ces informations, le formulaire d'ajout de bouteille est découpé en parties logiques, chacun concernant une famille de critères.

3. Modifier une bouteille

On souhaite pouvoir modifier les caractéristiques des vins et puis les enregistrer.

4. Supprimer une bouteille

La sortie d'une ou plusieurs bouteilles est un mouvement comme l'entrée d'une bouteille. Cette manipulation doit être réalisable rapidement et facilement pour l'utilisateur.

5. Gestion des pays, des régions et des appellations

Le domaine du vin est réparti selon les régions viticoles de chaque pays. Un vin est issu d'une région, qui appartient à un pays. Il faut donc intégrer à la base de données une liste de pays producteurs de vin, avec leurs régions respectives. Toutes les régions possèdent des appellations.

6. Gestion des cépages

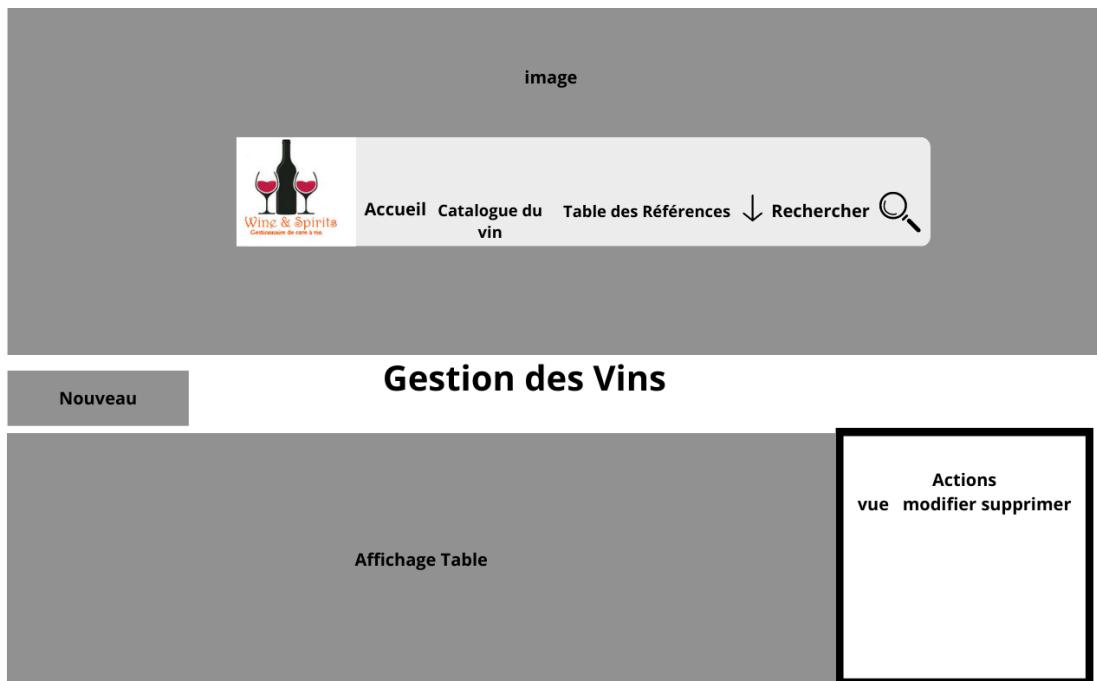
Un vin est composé d'un ou de plusieurs cépages. Une liste de cépages sera également entrée dans la base de données avec toujours la possibilité pour utilisateur d'en ajouter.

7. Gestion de couleur

La couleur est un des principaux critères pour choisir un vin car il permet de donner un indicatif sur son âge, mais également sur le cépage dont il provient. Elle permettra aussi, par le biais de l'intensité, d'estimer l'acidité du vin (en général) : un vin clair sera plus acide qu'un vin foncé.

1. Maquette de présentation

La maquette ci-dessous donne une idée de la présentation du site de gestion d'une cave à vin.



4. SPECIFICATIONS TECHNIQUE

Choix des technologies utilisées :

HTML (Hypertext Markup Language) :

J'ai utilisé ce langage pour structurer ma page web et son contenu. Il est essentiel pour proposer aux algorithmes une structure lisible, riche, originale et qui obtiendra un bon résultat d'indexation. C'est aussi un langage qui favorise une navigation en ligne fluide et agréable, qui contribue à fidéliser une clientèle sur le web.

CSS (Cascading StyleSheets) :

J'ai choisi d'utiliser un peu de CSS pour contrôler exactement l'affichage de chaque élément HTML dans le navigateur et présenter les documents avec la mise en forme. Il sert à styliser le contenu de la page et le positionnement des éléments.

Bootstrap :

J'ai utilisé ce framework CSS pour rendre mon site responsive design.

FontAwesome :

J'ai utilisé cette librairie pour insérer des icônes pour illustrer rapidement mes pages.

JavaScript :

La majorité du code de l'interface a été codé en JavaScript. J'ai utilisé ce langage de programmation pour concevoir des sites web interactifs et dynamique. De plus, j'ai choisi de privilégier la Programmation Orientée Objet afin d'organiser au mieux les dossiers de l'interface.

API (Application Programming Interfaces) et API REST :

Les API sont des mécanismes qui permettent à deux composants logiciels de communiquer entre eux à l'aide d'un ensemble de définitions et de protocoles.

API REST définit un ensemble de fonctions comme GET, PUT, DELETE, etc., que les clients peuvent utiliser pour accéder aux données du serveur. Les clients et les serveurs échangent des données au moyen du protocole HTTP.

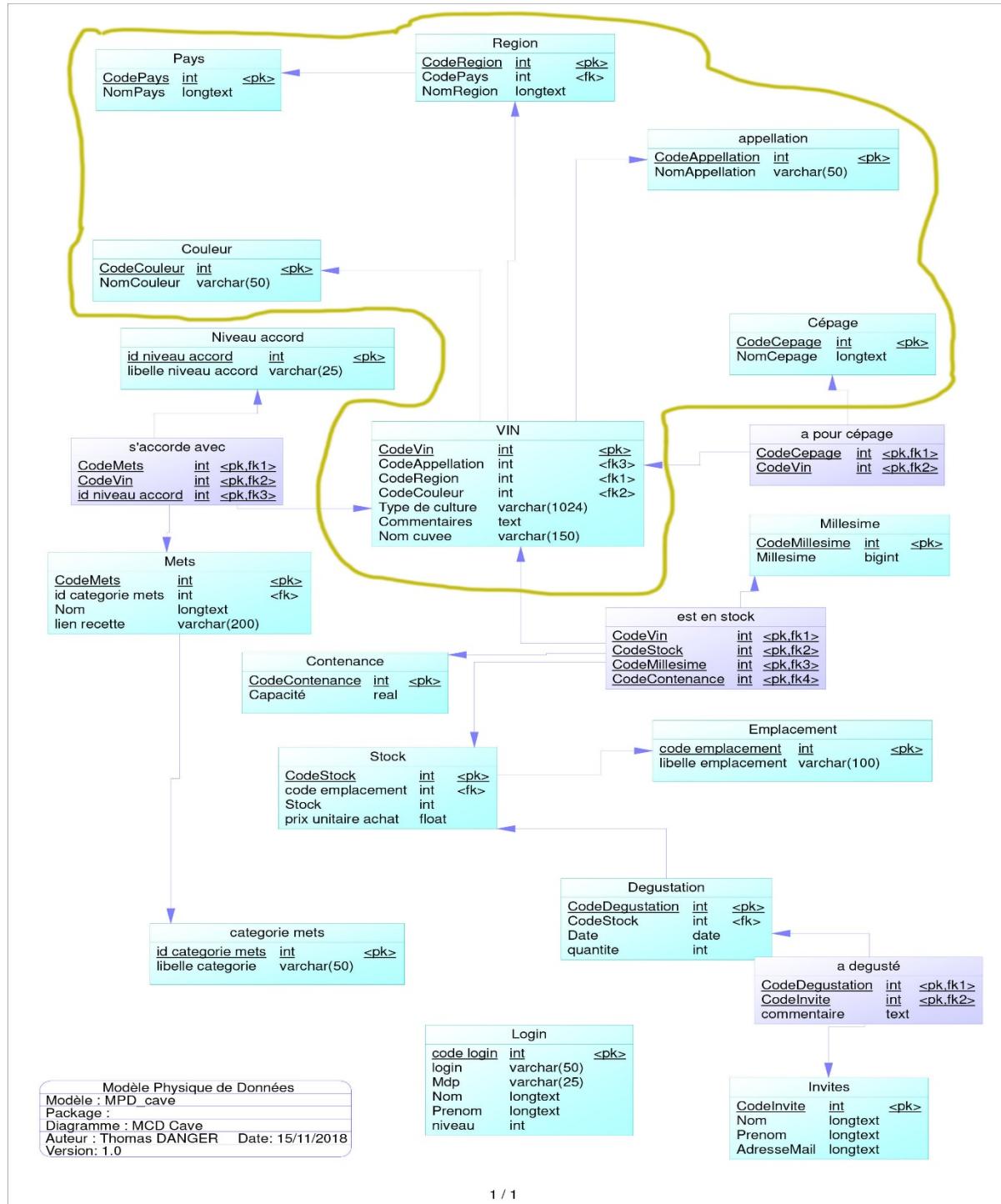
Les données présentes dans l'interface proviennent d'une API distante. Pour y accéder, des requêtes via AJAX ont été réalisées.

AJAX (Aysnchronous Javascript and XML):

Ajax est une méthode alliant plusieurs technologies existantes, dont : HTML, CSS, JavaScript, DOM, XML et qui réalise des requêtes HTTP de manière asynchrone. Lorsque ces technologies sont combinées dans le modèle AJAX, les applications web sont capables de réaliser de la mise à jour rapides et incrémentales de l'interface utilisateur sans devoir recharger la page entière dans le navigateur. Je l'ai exploitée pour communiquer avec l'API distante via des requêtes de type GET, POST, PUT et DELETE. Grâce à cela, j'ai pu dynamiser l'affichage tout en évitant le rechargelement régulier des pages de l'interface.

4.2 Schéma de la base de données

Le schéma ci-dessous est le M.P.D (Modèle Physique des données) issu de l'analyse du projet avec la méthode « MERISE » qui est l'acronyme pour « Méthode d'étude et de réalisation informatique pour les systèmes d'entreprise ».



Le M.P.D permet de construire la structure finale de la base de données. Il fait suite au M.C.D qui permet d'identifier les principales entités à représenter, leurs relations et leurs attributs, et d'analyser la structure conceptuelle du système d'information.

Ainsi, les tables qui seront dans un premier temps utilisées sont les tables :

- CEPAGE
- REGION
- PAYS
- APPELLATION
- COULEUR
- VIN

On remarque que la table vin est générée en faisant appel aux contenus de nombreuses autres tables.

Les tables

Les tables seront consultées à partir d'une API (Application Programming Interface ou Interface de programmation d'application en français).

Concrètement, c'est une interface de programmation permettant l'accès à un ou plusieurs services tels que des données ou des fonctionnalités fournies par un système tiers.

Il existe deux types d'API :

1. API SOAP (Simple Object Access Protocol).

SOAP est un protocole de messagerie qui peut être construit en utilisant la technologie Microsoft appelée WCF (Windows Communication Foundation), le SOAP ne renvoie que du XML et les services web ne sont rien d'autres que vos services SOAP.

SOAP permet au client d'obtenir des réponses indépendamment des plateformes et des langages.

1. API REST (Representational State Transfert).

Les **APIs** basés sur l'architecture REST, appelés API RESTful, sont beaucoup plus flexibles et rapides que celles utilisant le protocole SOAP.

- L'API REST renvoie du JSON, XML ou du YAML tandis que l'API SOAP ne renvoie que du XML

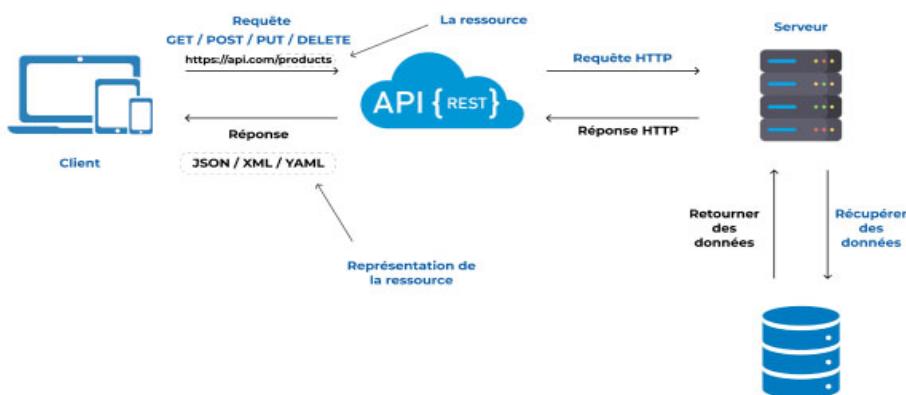
- L'API REST est un style architectural et SOAP est un protocole

- Les API REST sont légères contrairement aux services web SOAP

4.1 Comment fonctionne une API REST ?

Le client envoie une requête HTTP en précisant la ressource, le serveur traite la requête en récupérant les informations demandées dans sa base de données et ensuite renvoie une représentation de la ressource.

Comme le montre le schéma ci-dessous un client va faire des requêtes AJAX (Asynchronous JavaScripts And XML) de type « GET/POST/PUT et DELETE » à l'API qui va consulter la « BDD » (base de données). La BDD retournera la réponse à l'API qui à son tour la renverra au client.



Généralement, L'API REST expose des points de terminaison (endpoints) qui correspondent à des ressources spécifiques. Chaque endpoint représente une URL unique qui peut être appelée pour effectuer des actions spécifiques.

Les méthodes HTTP (GET , POST , PUT , DELETE, etc) sont utilisées pour déterminer quelle action doit être effectuée sur la ressource associée à l'endpoint. Par exemple, GET est utilisée pour récupérer des données, POST pour créer de nouvelles données, PUT pour mettre à jour des données existantes et DELETE pour supprimer des données.

Lorsqu'une requête est envoyée à l'API REST, le serveur répond avec une réponse HTTP appropriée. Cela peut inclure des données demandées, des codes de statut (comme 200 pour une réussite, 404 pour une ressource introuvable, etc.) et des en-têtes d'entête HTTP.

L'architecture REST est "stateless", ce qui signifie que chaque requête envoyée au serveur contient toutes les informations nécessaires pour comprendre et traiter la demande. Le serveur ne garde pas de trace de l'état du client entre les requêtes.

Sécurité : Pour accéder à certaines ressources ou effectuer certaines actions, l'API peut nécessiter une authentification à l'aide de jetons d'accès ou d'autres méthodes de sécurité pour s'assurer que seuls les utilisateurs autorisés peuvent interagir avec les ressources.

Plus précisément, on utilise CRUD qui est étroitement lié avec la gestion des données numériques. CRUD signifie la création d'un compte(create), utilisation à tout moment(read), la mise à jour (update) ou encore la suppression(delete)

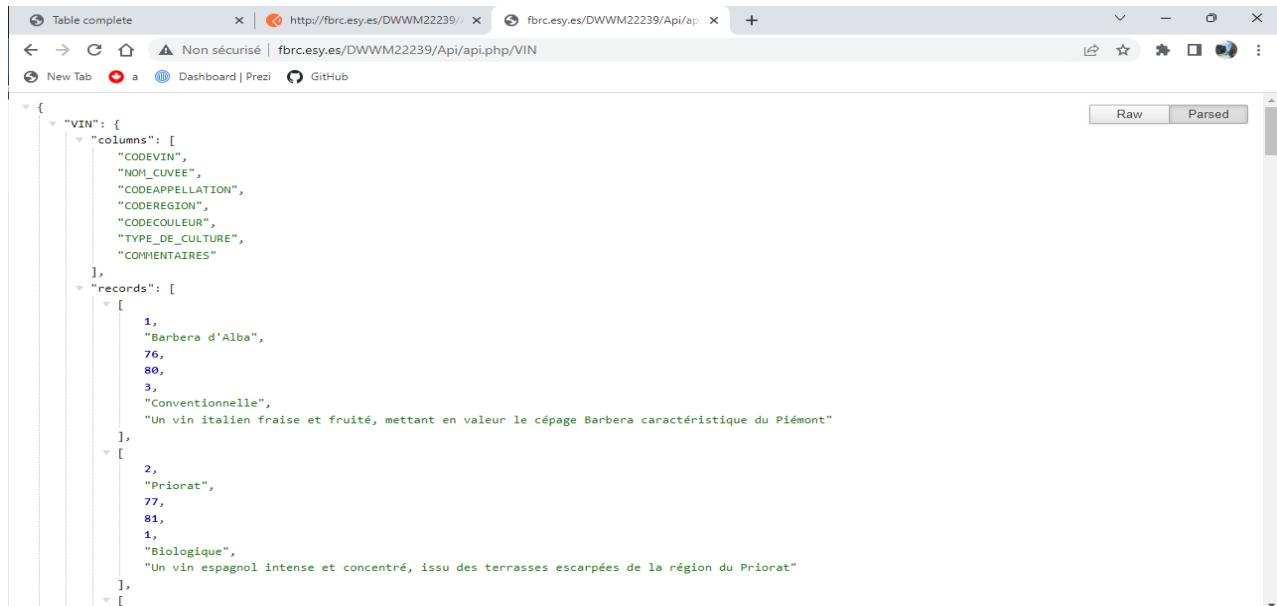
- **Create** (créer)
- **Read** ou **Retrieve** (lire)
- **Update** (mettre à jour)
- **Delete** ou **Destroy** (supprimer)

En résumé, une API Rest permet à des applications de communiquer et d'échanger des données via des requêtes HTTP standard, en utilisant des endpoints spécifiques et des méthodes bien définies pour accéder aux ressources et effectuer des actions sur ces ressources. Cela rend l'intégration entre différentes applications plus facile et permet une communication flexible et cohérente.

4.3 La structure des tables

1. La table « VIN »

La table 'VIN' se présente sous la forme suivante :



```
Table complete http://fbrc.esy.es/DWWM22239/ fbrc.esy.es/DWWM22239/Api/api.php/VIN Raw Parsed
{
  "VIN": {
    "columns": [
      "CODEVIN",
      "NOM_CUVEE",
      "CODEAPPELLATION",
      "CODEREGION",
      "CODECOULEUR",
      "TYPE_DE_CULTURE",
      "COMMENTAIRES"
    ],
    "records": [
      [
        1,
        "Barbera d'Alba",
        76,
        80,
        3,
        "Conventionnelle",
        "Un vin italien fraise et fruité, mettant en valeur le cépage Barbera caractéristique du Piémont"
      ],
      [
        2,
        "Priorat",
        77,
        81,
        1,
        "Biologique",
        "Un vin espagnol intense et concentré, issu des terrasses escarpées de la région du Priorat"
      ]
    ]
  }
}
```

2. La table « CEPAGE »

La table ‘CEPAGE’ se présente sous la forme suivant :

```

{
  "CEPAGE": {
    "columns": [
      "CODECEPAGE",
      "NOMCEPAGE"
    ],
    "records": [
      [
        1,
        "Sauvignon"
      ],
      [
        2,
        "Cabernet"
      ],
      [
        3,
        "Chenin"
      ],
      [
        4,
        "Romorantin"
      ],
      [
        5,
        "Gamay"
      ],
      [
        6,
        "Côt"
      ]
    ]
  }
}
    
```

3. La table « REGION»

La table ‘REGION’ se présente sous la forme suivante :

```

{
  "REGION": {
    "columns": [
      "CODEREGION",
      "CODEPAYS",
      "NOMREGION"
    ],
    "records": [
      [
        1,
        154,
        "Bourgogne"
      ],
      [
        3,
        154,
        "Aquitaine"
      ],
      [
        4,
        1,
        "Vallée du Rhône"
      ],
      [
        5,
        1,
        "Champagne"
      ],
      [
        6,
        1,
        "Languedoc-Roussillon"
      ],
      [
        7,
        1,
        "Pays de la Loire"
      ],
      [
        8,
        1,
        "Centre-Val de Loire"
      ],
      [
        9,
        1,
        "Normandie"
      ],
      [
        10,
        1,
        "Bretagne"
      ],
      [
        11,
        1,
        "Provence-Alpes-Côte d'Azur"
      ],
      [
        12,
        1,
        "Corse"
      ]
    ]
  }
}
    
```

4. La table « PAYS »

La table 'PAYS' se présente sous la forme suivante :

```

{
  "PAYS": {
    "columns": [
      "CODEPAYS",
      "NOMPAYS"
    ],
    "records": [
      [
        1,
        "AZER"
      ],
      [
        2,
        "ITALIE"
      ],
      [
        3,
        "ESPAGNEE"
      ],
      [
        17,
        "ETATS UNIS"
      ],
      [
        18,
        "CHINE"
      ],
      [
        20,
        "AUSTRALIE"
      ]
    ]
  }
}
    
```

5. La table « APPELLATION »

La table 'APPELLATION' se présente sous la forme suivante :

```

{
  "APPELLATION": {
    "columns": [
      "CODEAPPELLATION",
      "NOMAPPELLATION"
    ],
    "records": [
      [
        2,
        "Touraine Mesland2"
      ],
      [
        3,
        "Cheverny"
      ],
      [
        4,
        "Chateauneuf du Pape"
      ],
      [
        6,
        "Champagne"
      ],
      [
        51,
        "Bordeaux"
      ],
      [
        64,
        "Bourgogne"
      ]
    ]
  }
}
    
```

6. La table « COULEUR »

La table ‘COULEUR’ se présente sous la forme suivante :

```

{
  "COULEUR": {
    "columns": [
      "CODECOULEUR",
      "NOMCOULEUR"
    ],
    "records": [
      [
        1,
        "Rouge2"
      ],
      [
        2,
        "Blanc"
      ],
      [
        3,
        "Rosé"
      ],
      [
        4,
        "Pétillant Blanc"
      ],
      [
        5,
        "Pétillant Rosé"
      ],
      [
        6,
        "Pétillant Rouge"
      ]
    ]
  }
}
    
```

4.4 L'accès aux tables

Les tables sont accessibles à partir de l'url suivante

<https://afpafabrice.space/DWWM22239/Api/api.php/>

A laquelle il faut ajouter le nom de la table que l'on souhaite consulter (en majuscule).

Pour consulter la table ‘VIN’ on renseignera l'url suivante :

<https://afpafabrice.space/DWWM22239/Api/api.php/VIN>

Par exemple, pour consulter la table ‘CEPAGE’ on renseignera l'url suivante : <https://afpafabrice.space/DWWM22239/Api/api.php/CEPAGE>

Pour consulter la table ‘REGION’ on renseignera l'url suivante

<https://afpafabrice.space/DWWM22239/Api/api.php/REGION>

Pour consulter la table ‘PAYS’ on renseignera l'url suivante

<https://afpafabrice.space/DWWM22239/Api/api.php/PAYS>

Pour consulter la table ‘APPELLATION’ on renseignera l'url suivante

<https://afpafabrice.space/DWWM22239/Api/api.php/APPELLATION>

Pour consulter la table ‘COULEUR’ on renseignera l'url suivante

<https://afpafabrice.space/DWWM22239/Api/api.php/COULEUR>

4.5 Les clés des tables

Chaque table est constituée d'une clé primaire ou « pk » (primary key). Par exemple, la clé primaire ou « pk » de la table 'PAYS' est 'CODEPAYS'. La clé primaire ou « pk » de la table 'REGION'est 'CODEREGION'. La clé primaire ou « pk » de la table 'VIN' est 'CODEVIN'. Et ainsi de suite pour chacune des tables (cf chapitre 4.1 'Schéma de la base de données').

Certaines tables ne possèdent que des clé primaire « pk » comme les tables « PAYS », « APPELLATION », « COULEUR » et « CEPAGE ».

D'autres tables comme « REGION » possèdent une clé primaire « pk » et une clé étrangère « fk » (foreign key). Pour la table « REGION » la clé primaire « pk » est « CODEREGION » et la clé étrangère « fk » est « CODEPAYS ».

Enfin, certaines tables, comme la table « VIN » possèdent une clé primaire « pk » et plusieurs clés étrangères « fk ». Pour la table «VIN » la clé primaire « pk » est « CODEVIN » et les clés étrangères « fk » sont « CODEREGION », « CODECOULEUR » et « CODEAPPELLATION ».

L'objectif final de la clé primaire est de relier les enregistrements liés dans d'autres tables.

La différence de clé primaire « pk » et clé étrangère « fk ».

Le site [geeksforgeeks.org](https://www.geeksforgeeks.org/difference-between-primary-key-and-foreign-key/) l'explique très bien. Voici un extrait contenu dans la page :

<https://www.geeksforgeeks.org/difference-between-primary-key-and-foreign-key/>

A **primary key** is used to ensure that data in the specific column is unique. A column cannot have NULL values. It is either an existing table column or a column that is specifically generated by the database according to a defined sequence.

A **foreign key** is a column or group of columns in a relational database table that provides a link between data in two tables. It is a column (or columns) that references a column (most often the primary key) of another table.

The table with the foreign key is called the child table, and the table with the primary key is called the referenced or parent table. »

5. Rédaction du Projet

5.1. Organisation des dossiers et fichiers

```
> ress
< appall1.html
JS appall1.js
< cepage1.html
JS cepage1.js
< couleur1.html
JS couleur1.js
< index1.html
JS init_appall.js
JS init_cepage.js
JS init_couleur.js
JS init_pays.js
JS init_region.js
JS init_Vin.js
JS init.js
< pays1.html
JS pays1.js
< region2.html
JS region2.js
# style1.css
JS Table.js
< vincomplet3.html
JS vincomplet3.js
```

Le dossier de Wine & Spirits se présente comme ceci :

Il y a 7 fichiers .html correspondant aux différentes pages de l'interface.

Le **dossier CSS** contient un seul fichier avec l'extension .css : il gère la mise en forme visuelle de l'interface.

Le **dossier ressources (ress)** regroupe toutes les images présentes sur l'application : l'image de la page d'accueil ainsi que l'image de l'arrière-plan.

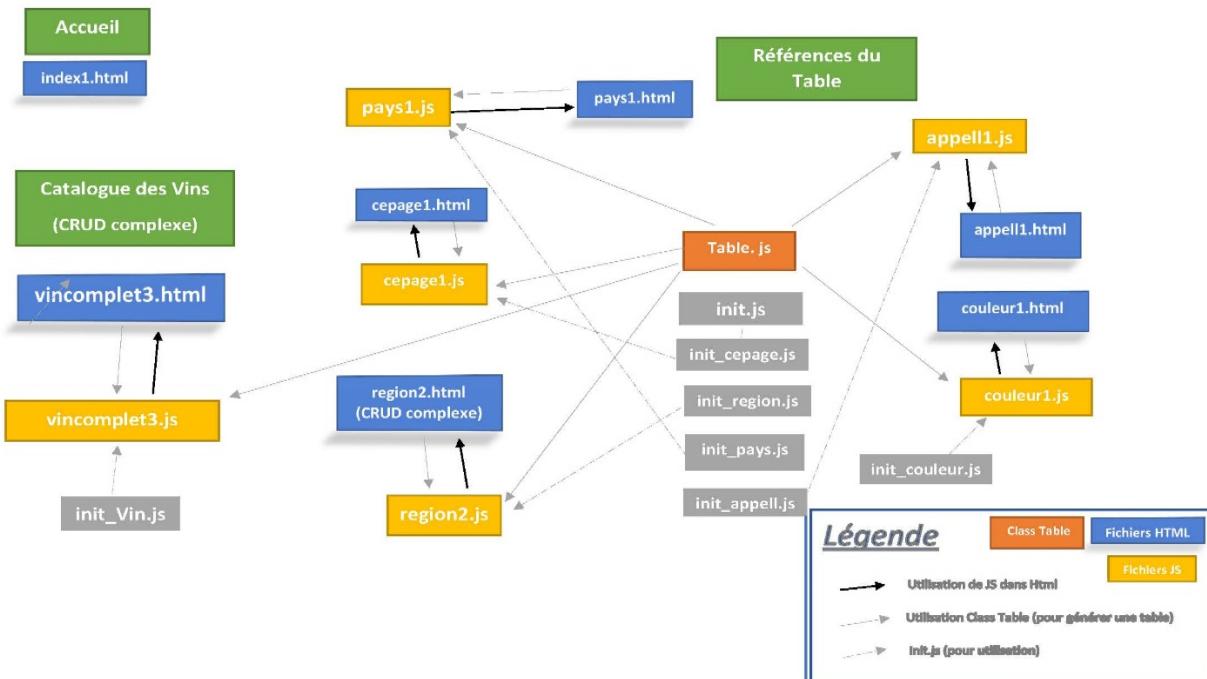
Les pages html sont associées au fichiers JavaScript suivants :

Le **dossier Javascript** comprend 14 fichiers :

«appall1.js » et « init_appall.js » pour Appellation,
« cepage1.js » et « init_cepage.js » pour Cépage,
« region1.js » et init_region.js » pour Région,
« couleur.js » et « init_couleur.js » pour Couleur,
« pays1.js » et init_pays.js » pour Pays,
« vincomplet3.js » et init_Vin.js pour le Catalogue du Vins
« Table.js »
« init.js »

Le programme principal en JavaScript et le fichiers « url et la classe Table.js », ont pour fonction de centraliser la génération de ces rendus spécifiques.

SCHEMA DIAGRAM



5.2 Le Programme

Détails du développement : exemple de la page « Table des vins »

- index.html (page d'accueil)
- pays1.html
- region2.html
- appell1.html
- couleur1.html
- cepage1.html
- vincomplet3.html

a. Les fichiers communs

Head

```
<!DOCTYPE html>
<html lang="fr">

<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Table complete</title>
    <link rel="stylesheet" type="text/css" href="style1.css" />

    <!-- Les ICONS FONTAWESOME-->
    <link rel="stylesheet" href="https://pro.fontawesome.com/releases/v5.10.0/css/all.css"
        integrity="sha384-AYmEC3Yw5cVb3ZcuHtOA93w35dYTsvhLPVnYs9eStHfGJvOvKxVfELGroGkvsg+p" crossorigin="anonymous" />

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
        integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztqQTwfspd3yD65VohhpuuCOMLASjc" crossorigin="anonymous" />
    <!-- Les ICONS BS5-->
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.9.1/font/bootstrap-icons.css" />
```

Body

```
<body>
    <div class="d-flex justify-content-center align-items-center "
        style="background-image: url('ress/vigne.jpeg'); background-size: cover; height:100vh">
        <nav class="navbar navbar-expand-sm navbar-expand-mg rounded-2" style="background-color: #gainsboro" ;>
            <div class="container-fluid">

                <a class="navbar-brand fw-bold " href="#"></a>
                
                <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
                    data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
                    aria-expanded="false" aria-label="Toggle navigation">
                    <span class="navbar-toggler-icon"></span>
                </button>
                <div class="collapse navbar-collapse" id="navbarSupportedContent">
                    <ul class="navbar-nav me-auto mb-2 mb-lg-0 ">
                        <li class="nav-item">
                            <a class="nav-link active link-primary " aria-current="page"
                                href="index1.html">Accueil</a>
                        </li>
                        <li class="nav-item">
                            <a class="nav-link active link-success" href="vincomplet3.html">Catalogue </a>
                        </li>

                        <li class="nav-item dropdown">
                            <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown"
                                aria-expanded="false">
                                Table des Références
                            </a>
                            <ul class="dropdown-menu">
                                <li><a class="dropdown-item" href="cepage1.html">Cépage</a></li>
                                <li><a class="dropdown-item" href="region2.html">Région</a></li>
                                <li><a class="dropdown-item" href="pays1.html">Pays</a></li>
                                <li><a class="dropdown-item" href="appell1.html">Appellation</a></li>
                                <li><a class="dropdown-item" href="couleur1.html">Couleur</a></li>
                            </ul>
                        </li>
                    </ul>
                </div>
            </div>
        </nav>
    </div>
```

Ln 1, Col 1 Spaces: 4 UTF-8 CR

Ce fichier représente le rendu de la page « Table des vins ».

Le head :

Plusieurs éléments sont préalablement chargés via les liens dans le head.

Le premier lien charge la feuille de style css qui s'occupe de la mise en page. Le second lien permet d'afficher les icônes issues du site fontawesome.com. Le troisième lien cible le chemin vers Bootstrap 5.

Le body :

Il définit le corps du document. Tout le contenu d'un document HTML, vous verrez des hyperliens, des images, des tableaux, des listes etc.

Les scripts :

Pour réaliser le rendu dynamique, des scripts relatifs au Javascript sont essentiels. Le premier script appelle le programme principal dont les fichiers présents dans le dossier «vincomplet3.js » ont besoin pour fonctionner. Le second script fait le lien avec la class Table qui gère l'affichage du tableau des vins dont les fichiers présents dans le dossier « Table.js ». Le troisième fichier est le dossier « init_Vin.js » qui est en lien pour se connecter à l'API Rest.

b. La page index.html

La page index.html intègre différents liens nécessaires à son fonctionnement. Le lien ci-dessous permet la gestion des icônes présentes dans le programme.

```
<!-- Les ICONS FONTAWESOME-->
<link rel="stylesheet" href="https://pro.fontawesome.com/releases/v5.10.0/css/all.css"
      integrity="sha384-AYmEC3Yw5cVb3ZcuHt0A93w35dYTsvhLPVnYs9eStHfGJvOvKxVfELGroGkvsg+p" crossorigin="anonymous" />
```

Ce lien va permettre de générer les icônes 'voir', 'modifier' et 'supprimer' et auront les représentations suivantes :



Les liens ci-dessous permettent l'utilisation de Bootstrap afin d'améliorer la présentation des pages html dans le programme, ainsi que la possibilité de le rendre web responsive .

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
      integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztQQTwFspd3yD65VohhpuuCOMLASjC" crossorigin="anonymous" />
```

c. L'emploi des fichiers JavaScript

Chaque page html fera appel à un script JavaScript avec, comme par exemple, la syntaxe utilisée ci-dessous dans la page vincomplet3.html :

```
<script src="Table.js"></script>

<script src="init_Vin.js"></script>

<script src="vincomplet3.js"></script>
```

Cela permet, entre autre, de séparer le code JavaScript du code html.

Par exemple, si un code JavaScript est présent dans une page html et que l'on souhaite l'utiliser dans une autre page html cela oblige à le dupliquer dans chaque page où il sera nécessaire.

Quand le programme lit le code ci-dessous que se passe-t-il ?

```

        }

        function enregAjoutVin() {
            alert("Ajout enregistrer");
            // La fonction enregAjoutVin est définie ici.
            let nouvelleLigne = [
                // Un nouvel objet "nouvelleLigne" est créé avec une propriété "NOM_CUVEE" ayant la valeur de l'élément avec l'ID "txtcuvee"
                NOM_CUVEE: document.getElementById("txtcuvee_ajout").value,
                CODEAPPELLATION: document.getElementById("comboAppellationAjout").value,
                CODEREGION: document.getElementById("comboRegionAjout").value,
                CODECOULEUR: document.getElementById("comboCouleurAjout").value,
                TYPE_DE_CULTURE: document.getElementById("txtculture_ajout").value,
                COMMENTAIRES: document.getElementById("txtcomment_ajout").value,
            ];

            let requestOptionsAdd = {
                method: "POST",
                headers: {
                    "Content-Type": "application/json",
                },
                body: JSON.stringify(nouvelleLigne),
                // Les options de la requête POST sont définies, y compris le corps de la requête qui contient l'objet "nouvelleLigne"
                // converti en chaîne JSON à l'aide de JSON.stringify().
            };

            fetch(urlApiVin, requestOptionsAdd)
                .then((response) => response.json())
                .then(function (data) {
                    location.reload();
                })
        }
    
```

COMMENTAIRE :

- a) La **function 'enregAjoutVin()**' est définie. Cette fonction est appelée lorsque l'utilisateur souhaite enregistrer un nouveau vin. Le code dans cette fonction effectue les étapes nécessaires pour ajouter un nouvel enregistrement de vin à l'aide d'une API.
- b) 'alert(« Ajout enregistrer »)' ; Lorsque l'utilisateur déclenche cette fonction , une boîte de dialogue d'alerte s'affiche avec le message « **Ajout enregistré** ».
- d) Pour chaque propriété du nouvel enregistrement de vin, le code récupère la valeur de l'élément HTML correspondant. Par exemple :

-NOM_CUVEE : document.getElementById(« txtcuvee_ajout »).value, Recupère la valeur de l'élément avec ID « txtcuvee_ajout et attribue à la propriété 'NOM_CUVEE' de l'objet 'nouvelleLigne'.
-De même, les autres propriétés ('CODEAPPELLATION','CODEREGION','CODECOULEUR','TYPE_DE_CULTURE' et 'COMMENTAIRES') sont également définies en récupérant les valeurs des éléments HTML correspondants.

let requestOptionsadd = {.....} : Ici, un nouvel objet 'requestOptionsAdd' est créé pour configurer les options de la requête POST à envoyer à l'API. Ces options incluent :

-method : « POST » : Indique que la requête est une requête POST pour ajouter un nouvel enregistrement.

headers : {« Content-Type » : « application/json »} : Définit les en-têtes de la requête pour spécifier que le corps de la requête est au format JSON.

body : JSON.stringify (nouvelleLigne) : Le corps de la requête est configuré en utilisant JSON.stringify () pour convertir l'objet nouvelleLigne en une chaîne JSON. Cela permet d'envoyer les données du nouvel enregistrement de vin à l'API sous forme JSON.

- e) **'fetch (urlApivin, requestOptionsAdd)** : Ici, la **function fetch()** est utilisée pour envoyer la requête POST à l'URL spécifiée par la variable urlApiVin, qui doit être définie d'appeler la **function enregAjoutVin()**. L'objet requestOptionsadd contenant les options de la requête est également passé en paramètre.
- f) **'then ((response) => response.json())** : Une fois que la réponse de l'API est reçue, cette ligne de code prend la réponse('réponse) et la convertit en format JSON en appelant 'response.json()'. Cela renvoie une promesse qui sera résolue lorsque la réponse aura été convertie en JSON.
- g) **'.then(function(data) {location.reload() })**: Une fois que les données JSON sont disponibles, la fonction anonyme définie ici est appelée avec ses données en tant qu'argument (le nom « data » est utilisé pour les données JSON). Dans ce cas, la fonction recharge simplement la page (location.reload()) après l'ajout réussi du nouvel enregistrement de vin à la base de données via l'API.

En RESUME :

En résumé, ce code permet d'ajouter un nouvel enregistrement de vin en récupérant les valeurs des champs de saisie de l'utilisateur, en envoyant ces valeurs à une API via une requête POST au format JSON, en rechargeant ensuite la page pour refléter les mises à jour après l'ajout du vin.

On obtient alors la page html ci-dessous :

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "Ajout d'un Vin" (Add a Wine). The form contains the following data:

Code	Cuvée
1	Barbera d'Alba
2	Priorat

Form fields:

- Cuvée : Bordeaux
- Appellation : Touraine Mesland
- Région : Bourgogne
- Couleur : Rouge
- Cultures : Saisir une Culture
- Commentaires : Saisir un Commentaire

Buttons at the bottom:

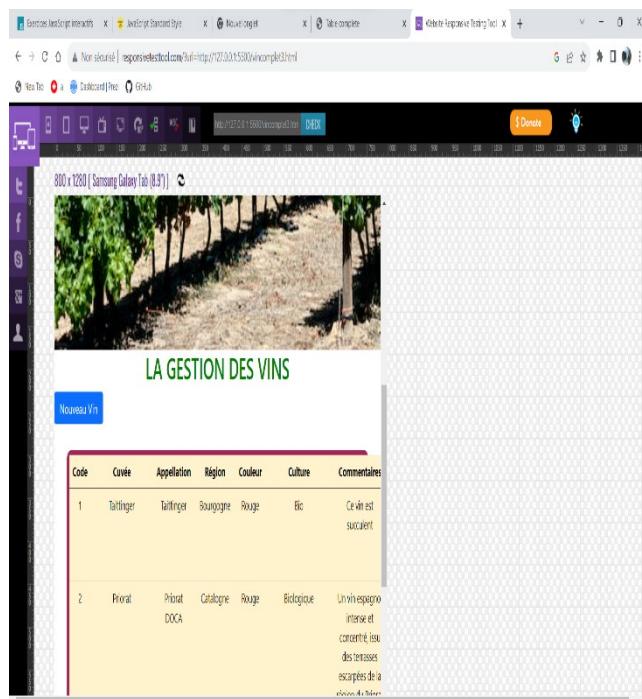
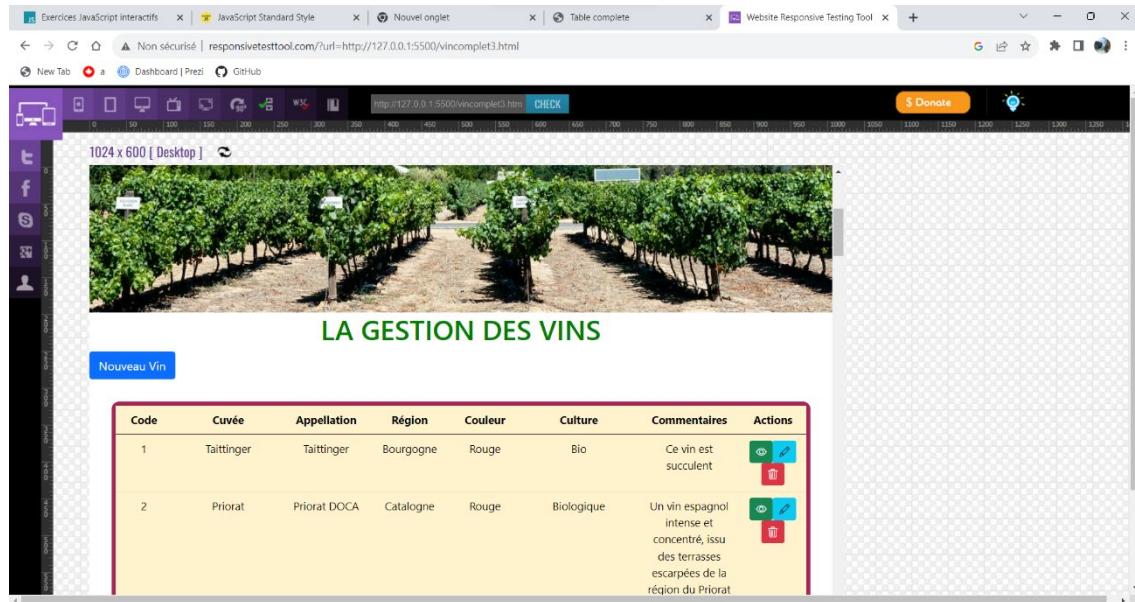
- Enregistrer (Register)
- Annuler (Cancel)

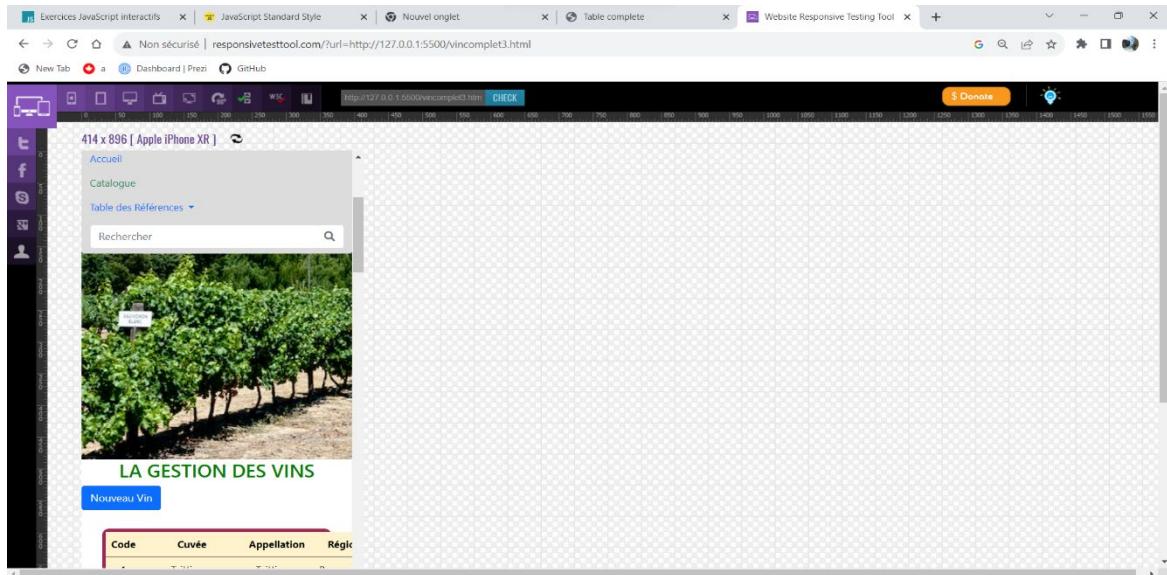
On the right, there is a sidebar with two entries:

- Commentaires** (Comments): Un vin italien fraise et fruité, mettant en valeur le cépage Barbera caractéristique du Piémont. Actions: Edit, Delete.
- Actions**: Un vin espagnol intense et concentré, issu des terrasses escarpées de la région du Priorat. Actions: Edit, Delete.

Visuel de l'interface sur différents écrans

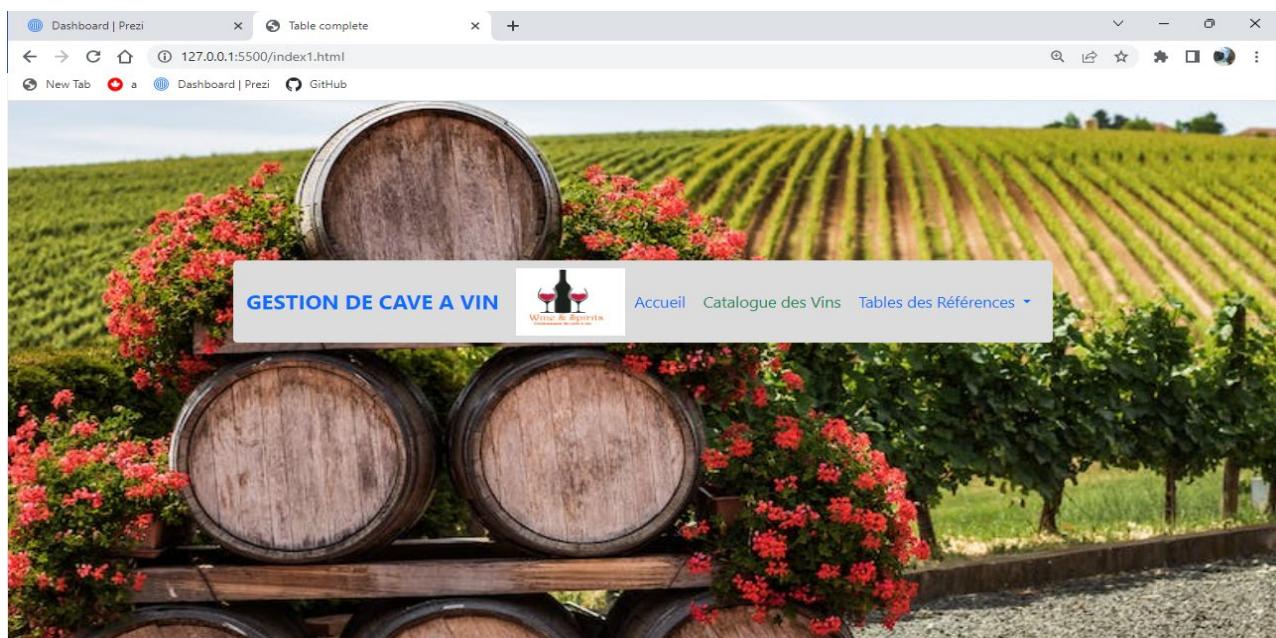
L'utilisateur a la possibilité de visualiser l'interface de gestion des vins sur différents supports : ordinateur, tablette et smartphone.





6– Utilisation du programme de Gestion des vins

Ce programme permet la gestion d'une cave à vin. La page d'accueil se présente sous la forme suivante :



La barre de navigation est constituée de 4 onglets dont 1 menu déroulant :

ACCUEIL
CATALOGUE
TABLEAU DES REFERENCES
-Cépage
-Région
-Pays
-Appellation
-Couleur
RECHERCHER

6.1 L'onglet 'Catalogue'

En cliquant sur cet onglet on accède à l'ensemble des vins répertoriés dans cette application comme le montre la saisie d'écran ci-dessous :

The screenshot shows a web browser window with the URL 127.0.0.1:5500/vincomplet1.html. The page features a header with a logo, a search bar, and navigation links for Accueil, Catalogue, Table des Références, and Rechercher. Below the header is a scenic image of a vineyard with mountains in the background. The main content area is titled "LA GESTION DES VINS". At the bottom left is a blue button labeled "Nouveau Vin". A table below lists three wines with columns for Code, Cuvée, Appellation, Région, Couleur, Culture, Commentaires, and Actions.

Code	Cuvée	Appellation	Région	Couleur	Culture	Commentaires	Actions
1	Barbera d'Alba	Alsace	Abruzzes	Rouge	Conventionnelle	Un vin italien fraise et fruité, mettant en valeur le cépage Barbera caractéristique du Piémont	
2	Priorat	Priorat DOCA	Catalogne	Rouge	Biologique	Un vin espagnol intense et concentré issu des terrasses escarpées de la région du Priorat.	
3	Brunello di Montalcino	Brunello Di Montalcino DOCG	Toscane	Rouge	Biodynamique	Un vin toscan élégant et complexe, vieilli pendant de longues périodes avant d'être dégusté.	

On retrouve la dénomination des vins sous forme d'un tableau constitué de 8 colonnes.

La première colonne « Code Vin » est le code généré par l'application au fur et à mesure lorsqu'ont saisi de vins. Ce code sera incrémenté automatiquement.

La deuxième colonne « Cuvée » contient le nom du vin. Ce vin sera saisi au moment de sa création dans l'application.

La troisième colonne « Appellation », c'est un type de vin. Chaque appellation vient d'une région.

La quatrième colonne « Région » indique l'origine du vin. La région d'origine sera choisie au moment de la création du vin dans l'application.

La cinquième colonne « Couleur » donne le couleur du vin. La couleur sera choisie au moment de la création du vin dans l'application.

La sixième colonne « Culture » permet d'ajouter, si besoin, des précisions sur le type de culture.

La septième colonne « Commentaires » permet de mettre des précisions que l'on souhaite avoir à propos du vin.

La huitième colonne « Actions » permet de voir le détail d'un vin, de le modifier ou de le supprimer.

6.2 Comment saisir un nouveau vin

A partir de la page « La gestion des vins », il est possible de cliquer sur le bouton « Nouveau Vin» pour ajouter un vin:

Code	Cuvee	Appellation	Region	Couleur	Culture	Commentaires	Actions
1	Barbera d'Alba	Alsace	Abruzzes	Rosé	Conventionnelle	Un vin italien fraise et fruité, mettant en	

Une fenêtre s'ouvre :

Code	Cuvee
1	Barbera d'Alba

Commentaires
Actions

Un vin italien fraise et fruité, mettant en valeur le bouquet Barbera caractéristique

Il suffit alors de renseigner les différents champs contenus dans la fenêtre. Dans le champ « Cuvée » on saisira le nom du vin que l'on souhaite ajouter dans l'application.

Pour les champs 'Appellation', 'Région' et 'Couleur', on peut choisir dans la liste déroulante. Il suffira de faire dérouler la liste jusqu'à la dénomination qui nous intéresse et de cliquer dessus pour la sélectionner comme le montre la saisie d'écran ci-dessous :

Code	Cuvee	Appellation	Culture	Commentaires	Actions		
1	Barbera d'Alba	Alsace		Un vin italien fraîche et fruité, mettant en valeur le cépage Barbera caractéristique du Piémont			
2	Priorat	Priorat DOCA	Rouge2	Un vin espagnol intense et concentré, issu des terrasses escarpées de la région du Priorat			
3	Brunello di Montalcino	Brunello Di Montalcino DOGC	Toscane	Rouge2	Biodynamique	Un vin toscan élégant et complexe, vieilli pendant de longues périodes avant d'être mis en bouteille	

Pour terminer la saisie, cliquez sur « Enregistrer ». Le nouveau 'vin' est maintenant enregistré et présent dans le tableau.

Remarque importante :

La saisie d'un nouveau vin est conditionnée au renseignement préalable de certaines tables d'application. Les tableaux « Région », « Appellation » et « Couleur » devront être renseignés avant la saisie d'un nouveau vin.

6.3 Comment apporter de modification sur un vin

A partir de la page « La gestion des vins » (accessible en cliquant sur ‘Catalogue’ dans la barre de navigation), il y a 3 boutons en face de chaque ligne contenant le nom d'un vin.

Code	Cuvee	Appellation	Région	Couleur	Culture	Commentaires	Actions
1	Barbera d'Alba	Alsace	Abruzzes	Rouge	Conventionnelle	Un vin italien fraise et fruité, mettant en valeur le cépage Barbera caractéristique du Piémont	
2	Priorat	Priorat DOCA	Catalogne	Rouge	Biologique	Un vin espagnol intense et concentré, issu des terrasses escarpées de la région du Priorat	
3	Brunello di Montalcino	Brunello Di Montalcino DOCG	Toscane	Rouge	Biodynamique	Un vin toscan élégant et complexe, vieilli pendant de longues périodes avant d'être mis en bouteille	

Pour modifier un vin, il faut cliquer sur le bouton représentant un crayon :



Une fenêtre s'ouvre :

Code	Cuvee	Appellation
1	Barbera d'Alba	Alsace
2	Priorat	Priorat
3	Brunello di Montalcino	Brunello Di Montalcino
5	Chianti Classico	Chianti Classico DOCG

Modification

Cuvee:

Appellation:

Région:

Couleur:

Cultures:

Commentaires:

Apporter les modifications souhaitées et cliquez sur le bouton « Enregistrer ». La modification est alors faîte.

6.4 Comment supprimer un vin

A partir de la page « La gestion des vins, (accessible en cliquant sur ‘Catalogue’ dans la barre de navigation, il y a 3 boutons en face de chaque ligne contenant le nom d'un vin.

Code	Cuvee	Appellation	Region	Couleur	Culture	Commentaires	Actions
1	Barbera d'Alba	Alsace	Abruzzes	Rouge	Conventionnelle	Un vin italien fraîche et fruité, mettant en valeur le cépage Barbera caractéristique du Piémont	
2	Priorat	Priorat DOCA	Catalogne	Rouge	Biologique	Un vin espagnol intense et concentré, issu des terrasses escarpées de la région du Priorat	
3	Brunello di Montalcino	Brunello Di Montalcino DOGC	Toscane	Rouge	Biodynamique	Un vin toscan élégant et complexe, vieilli pendant de longues périodes avant d'être mis en bouteille	

Pour supprimer un vin, il faut cliquer sur le bouton représentant une poubelle :



Le message suivant apparaît :

127.0.0.1:5500 indique
Voulez-vous supprimer ce vin ?

Code	Cuvee	Appellation	Region	Couleur	Culture	Commentaires	Actions
1	Barbera d'Alba	Alsace	Abruzzes	Rosé	Conventionnelle	Un vin italien fraîche et fruité, mettant en valeur le cépage Barbera caractéristique du Piémont	
2	Priorat	Priorat DOCA	Catalogne	Rouge	Biologique	Un vin espagnol intense et concentré, issu des terrasses escarpées de la région du Priorat	
3	Brunello di Montalcino	Brunello Di Montalcino DOGC	Toscane	Rouge	Biodynamique	Un vin toscan élégant et complexe, vieilli pendant de longues périodes avant d'être mis en bouteille	

Cliquez sur le bouton «OK ». La suppression est alors effectuée. Si l'on ne souhaite pas effectuer la suppression il suffit de cliquer sur le bouton « Annuler ».

6.5 Comment voir la liste des vins

Code	Cuvée	Appellation
1	Ferme	Bordeaux
2	Rouge	Bordeaux
3	Brunello di Montalcino	Brunello Di Montalcino D.O.C.G.
4	Chianti Classico	Chianti Classico D.O.C.G.
5	Bordeaux Bourgogne	Bourgogne
6	Gavi di Gavi	Gavi di Gavi D.O.C.
7	Château Margaux	Château Margaux
8	Château Latour	Château Latour
9	Château Mouton Rothschild	Château Mouton Rothschild
10	Château Pichon Longueville Baron	Château Pichon Longueville Baron
11	Château Pichon Longueville Comtesse de Lalande	Château Pichon Longueville Comtesse de Lalande
12	Château Lafite Rothschild	Château Lafite Rothschild
13	Château Margaux	Château Margaux
14	Château Latour	Château Latour
15	Château Mouton Rothschild	Château Mouton Rothschild
16	Château Pichon Longueville Baron	Château Pichon Longueville Baron
17	Château Pichon Longueville Comtesse de Lalande	Château Pichon Longueville Comtesse de Lalande
18	Château Lafite Rothschild	Château Lafite Rothschild
19	Château Margaux	Château Margaux
20	Château Latour	Château Latour

Cliquez sur le bouton pour voir les différents critères de vins notamment le code, la cuvée, l'appellation, la région, la couleur, la culture et les commentaires.

6.6 Comment rechercher un critère de vin

Cliquer sur pour rechercher un critère ou chercher un vin spécifique.

7. L'onglet 'Gestion des Tables de référence'

Comme abordé à la fin du chapitre 6, il est important de procéder aux renseignements de certaines tables avant la saisie d'un nouveau vin.

Pour cela il est nécessaire d'aller sur l'onglet ' Tables de référence' présent dans la barre de navigation.

L'action de cliquer sur cet onglet ouvre une liste ou apparaît toutes les tables nécessaires au fonctionnement de l'application.

The screenshot shows a web browser window with multiple tabs. The active tab displays a vineyard landscape with mountains in the background. A navigation bar at the top includes links for Accueil, Catalogue, Table des Références (which is currently selected and has a dropdown menu), and Rechercher. The dropdown menu lists: Cépages, Appellation, Couleur, Pays, and Région. Below the menu, a table titled 'LA GESTION DES VINS' is visible, showing two rows of wine information. At the bottom left, there is a blue button labeled 'Nouveau Vin'.

Code	Cuvee	Appellation	Région	Couleur	Culture	Commentaires	Actions
1	Barbera d'Alba	Alsace	Abruzzes	Rouge	Conventionnelle	Un vin italien fruité et fruité, mettant en valeur le cépage Barbera caractéristique du Piémont	
2	Priorat	Priorat DOCA	Catalogue	Rouge	Biologique	Un vin espagnol intense et concentré, issu des terrasses escarpées de la	

7.1 Gestion du tableau 'Cépage'

Avant toute saisie d'un nouveau vin, la table 'Cépage' doit être renseignée. Pour cela il suffit de cliquer sur 'Cépage' présent dans la liste comme montré ci-dessous :

The screenshot shows a web browser window with multiple tabs. The active tab displays a vineyard landscape with mountains in the background. A navigation bar at the top includes links for Accueil, Catalogue, Table des Références (which is currently selected and has a dropdown menu), and Rechercher. The dropdown menu lists: Cépages, Appellation, Couleur, Pays, and Région. Below the menu, a table titled 'LA GESTION DES VINS' is visible, showing an empty table structure with columns: Cuvee, Appellation, Région, Couleur, Culture, Commentaires, and Actions. At the bottom left, there is a blue button labeled 'Nouveau Vin'.

Cuvee	Appellation	Région	Couleur	Culture	Commentaires	Actions
-------	-------------	--------	---------	---------	--------------	---------

L'action de cliquer sur 'Cépage' va ouvrir une nouvelle page qui ressemblera à celle vu ci-dessous :

The screenshot shows a browser window with several tabs open. The active tab is titled '127.0.0.1:5500/cepage1.html'. The page content includes a navigation bar with links for 'Accueil', 'Catalogue', 'Tables des Références', and a search bar. A dropdown menu is open over the 'Tables des Références' link, showing options: 'Cépage', 'Région', 'Pays', 'Appellation', and 'Couleur'. Below the menu is a large image of red grapes. The main section is titled 'GESTION DES CEPAGES' and contains a table with two rows:

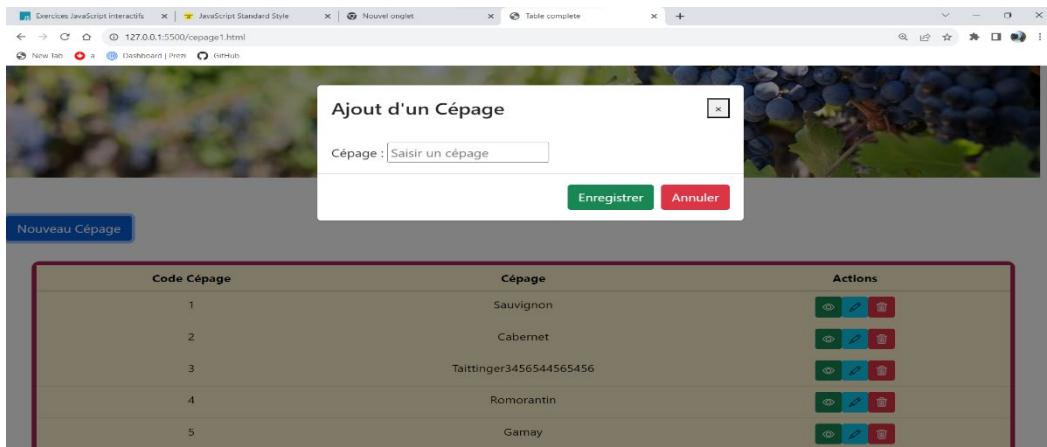
Code Cépage	Cépage	Actions
1	Sauvignon	
2	Cabernet	

Pour créer un nouveau 'Cépage', il faut cliquer sur le bouton « Nouveau Cépage » :

The screenshot shows a browser window with the same tab configuration as the previous one. The page content is identical, including the 'GESTION DES CEPAGES' table. A new row has been added at the top of the table:

Code Cépage	Cépage	Actions
1	Sauvignon	
2	Cabernet	
3	Taittinger3456544565456	
4	Romorantin	
5	Gamay	

Une fenêtre s'ouvre :



Il suffit alors de renseigner le champ ‘Ajout d’un Cépage’ contenu dans la fenêtre.

Pour terminer la saisie, cliquez sur « Enregistrer ». Le nouveau ‘Cépage’ est maintenant enregistré et présent dans la liste.

Pour voir, modifier ou supprimer un cépage, il suffit de suivre la même procédure que celle exposé au chapitre 6.3, 6.4 et 6.5 avec l’utilisation des boutons



7.2 Gestion du tableau 'Région'

Pour gérer la table 'Région' il suffit de cliquer sur 'Région' présent dans la liste comme montré ci-dessous :



L'action de cliquer sur 'Région' va ouvrir une nouvelle page qui ressemblera à celle-ci :

Code	Nom Pays	Nom Region	Actions
1	FRANCE	Bourgogne	
3	FRANCE	Aquitaine	

Pour créer une nouvelle 'Région' cliquez sur le bouton « Nouvelle Région » :

GESTION DES REGIONS

Code	Nom Pays	Nom Region	Actions
1	FRANCE	Bourgogne	
3	FRANCE	Aquitaine	

Une fenêtre s'ouvre :

Ajout d'un Région

Nom Pays :
FRANCE

Région :
Saisir un region

Enregistrer Annuler

Il suffit de renseigner les différents champs contenus dans la fenêtre. Dans le champ 'Région' on saisira le nom de la région que l'on souhaite ajouter dans l'application et choisissez le « nom pays » à l'aide de menu déroulant.

Code	Nom Pays	Nom Region	Actions
1	FRANCE	Aquitaine	
3	FRANCE	Vallée du Rhône	
4	AZER	Champagne	
5	AZER	Vallée de la Loire	
6	AZER	Toscane	
7	ITALIA	Piemont	
8	ITALIA	Rioja	
9	ESPAGNE	Navarre	
10	AZER		

Pour terminer la saisie, cliquez sur « Enregistrer ». La nouvelle ‘Région’ est maintenant enregistrée et présente dans la liste.

Pour voir, modifier ou supprimer une région, il suffit de suivre la même procédure que celle exposé au chapitre 6.3, 6.4 et 6.5 avec l’utilisation des boutons :



7.3 Gestion du tableau ‘Pays’

Pour gérer la table ‘Pays’ il suffit de cliquer sur ‘Pays’ présent dans la liste comme montré ci-dessous :

The screenshot shows a web browser window with a header 'GESTION DE CAVE A VIN'. Below the header is a banner featuring two wooden barrels and red flowers. To the right of the banner is a navigation bar with links: Accueil, Catalogue des Vins, Tables des Références (with a dropdown menu), and a search bar. The 'Tables des Références' link is highlighted in blue. The dropdown menu under 'Tables des Références' contains the following items: Cépage, Région, Pays (which is highlighted in grey), Appellation, and Couleur. The background of the page shows a vineyard landscape.

L'action de cliquer sur ‘Pays’ va ouvrir une nouvelle page qui ressemblera à celle-ci :

The screenshot shows a web browser window with a header 'Gestion des Pays'. Below the header is a banner featuring a bunch of green grapes and a bottle of white wine. To the right of the banner is a search bar labeled 'Rechercher'. The main content area has a title 'GESTION DES PAYS'. Below the title is a button labeled 'Nouveau Pays'. A table is displayed with the following data:

Code Pays	Pays	Actions
1	FRANCE	

Pour créer un nouveau 'Pays' cliquez sur le bouton « Nouveau Pays » :

Code Pays	Pays	Actions
1	FRANCE	
2	ITALIE	

Une fenêtre s'ouvre :

Il suffit de renseigner le champ contenu dans la fenêtre. Dans le champ 'Pays' on saisira le nom de pays que l'on souhaite ajouter dans l'application.

Pour terminer la saisie, cliquez sur « Enregistrer ». Le nouveau 'Pays' est maintenant enregistré et présent dans la liste.

Pour voir, modifier ou supprimer un Pays, il suffit de suivre la même procédure que celle exposée au chapitre 6.3, 6.4 et 6.5 avec l'utilisation des boutons :

7.4 Gestion du tableau 'Appellation'

Pour gérer la table 'Appellation' il suffit de cliquer sur 'Appellation' présent dans la liste comme montré ci-dessous :



L'action de cliquer sur 'Appellation' va ouvrir une nouvelle page qui ressemblera à celle vu ci-dessous :

A screenshot of a web browser window titled 'Table complete'. The address bar shows '127.0.0.1:5500/appell1.html'. The page features a header with 'GESTION DES VINS' and a photograph of several wine bottles. Below the header is a blue button labeled 'Nouvelle Appellation'. A table is displayed with columns for 'Code Appellation', 'Appellation', and 'Actions'. The data in the table is as follows:

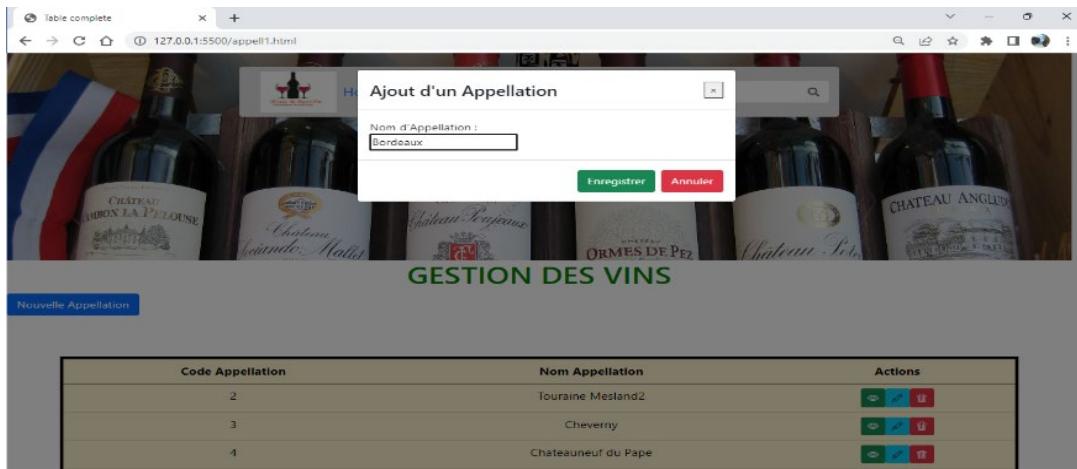
Code Appellation	Appellation	Actions
2	Touraine Mesland2	
3	Cheverny	
4	Chateauneuf du Pape	
6	Champagne	

Pour créer une nouvelle 'Appellation' cliquez sur le bouton « Nouvelle Appellation » :

A screenshot of a web browser window titled 'Table complete'. The address bar shows '127.0.0.1:5500/appell1.html'. The page features a header with 'GESTION DES VINS' and a photograph of several wine bottles. Below the header is a blue button labeled 'Nouvelle Appellation'. A table is displayed with columns for 'Code Appellation', 'Appellation', and 'Actions'. The data in the table is as follows:

Code Appellation	Appellation	Actions
2	Touraine Mesland2	

Une fenêtre s'ouvre :



Il suffit de renseigner le champ contenu dans la fenêtre. Dans le champ ‘Nom d’Appellation’ on saisira le nom de l’appellation que l’on souhaite ajouter dans l’application.

Pour terminer la saisie, cliquez sur « Enregistrer ». La nouvelle ‘appellation’ est maintenant enregistrée et présente dans la liste.

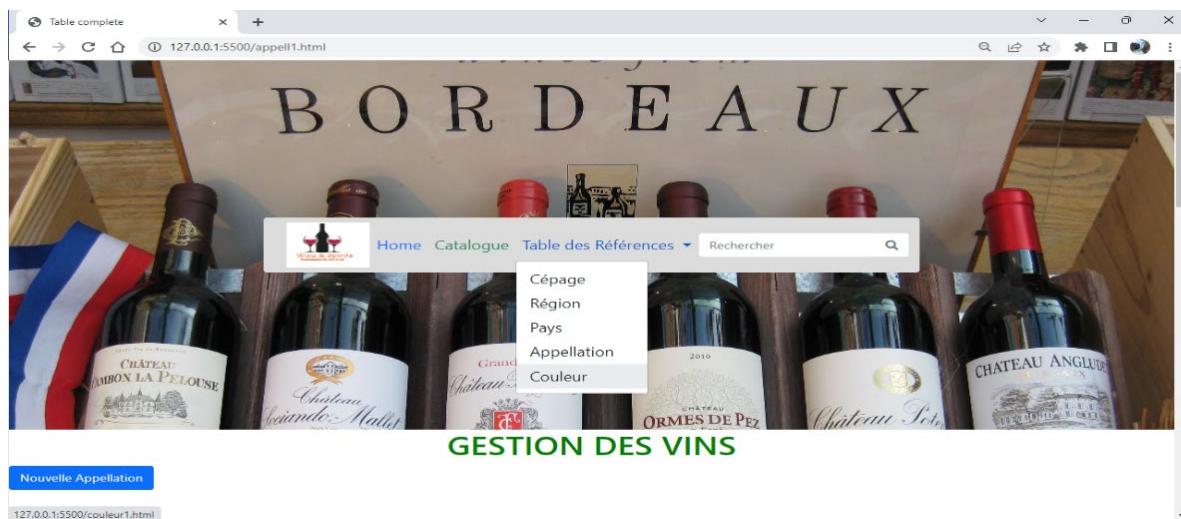
Pour voir, modifier ou supprimer un Appellation, il suffit de suivre la même procédure que celle exposé au chapitre 6.3, 6.4 et 6.5 avec l’utilisation des boutons :



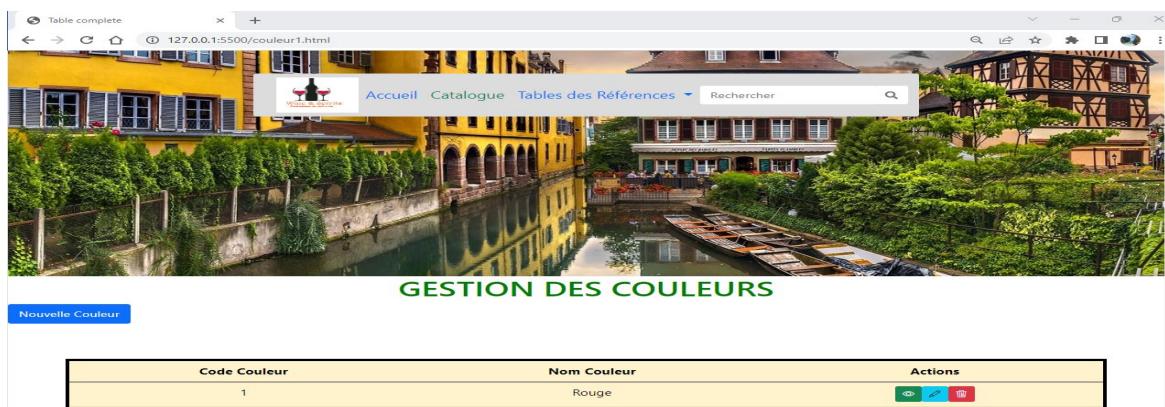
Pour faire une recherche, le bouton rechercher est présent dans chaque onglet, il suffit de saisir le code ou le nom dans le champ de saisie pour le trouver.

7.5 Gestion du tableau ‘Couleur’

Pour gérer la table ‘Couleur’ il suffit de cliquer sur ‘Couleur’ présent dans la liste comme montré ci-dessous :



L'action de cliquer sur ‘Couleur’ va ouvrir une nouvelle page qui ressemblera à celle vu ci-dessous :



Pour créer un nouvelle ‘Couleur’ cliquez sur le bouton « Nouvelle Couleur » :



Une fenêtre s'ouvre :



Il suffit de renseigner le champ contenu dans la fenêtre. Dans le champ 'Couleur' on saisira une couleur que l'on souhaite ajouter dans l'application.

Pour terminer la saisie, cliquez sur « Enregistrer ». La nouvelle 'couleur' est maintenant enregistré et présent dans la liste.

Pour voir, modifier ou supprimer une Couleur, il suffit de suivre la même procédure que celle exposé au chapitre 6.3, 6.4 et 6.5 avec l'utilisation des boutons



8. TESTS

Test	Affichage	Nouveau	Visualisation	Modification	Suppression	Recherche
index	OK	-	-		-	-
Pays	OK	OK	OK	OK	OK	OK
Région	OK	OK	OK	OK	OK	OK
Appellation	OK	OK	OK	OK	OK	OK
Couleur	OK	OK	OK	OK	OK	OK
Cépage	OK	OK	OK	OK	OK	OK
Catalogue	OK	OK	OK	OK	OK	OK

9 LA VEILLE TECHNIQUE LIENS UTILES ANGLOPHONE

<https://www.w3schools.com/js/>

The screenshot shows the W3Schools JavaScript tutorial page. The left sidebar lists various JS topics from basic to advanced. The main content area starts with a brief introduction to JavaScript, followed by a "Start learning JavaScript now" button. Below this, there's a section titled "Examples in Each Chapter" with a note about the "Try it Yourself" editor. A "My First JavaScript" example is shown in a code editor window. On the right side, there are promotional banners for "Bootcamps", "Spaces", "Get Full Access", and a "Join our Mini JavaScript Bootcamp".

<https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX>

The screenshot shows the Mozilla Developer Guide for Ajax. The left sidebar has a navigation tree for various web development topics. The main content area is titled "Ajax" and discusses the XMLHttpRequest object and its asynchronous nature. It also mentions JSON as a preferred data format over XML. The right sidebar contains a "In this article" section with links to "Examples", "Step 1 – How to make an HTTP request", "Step 2 – Handling the server response", "Step 3 – A Simple Example", "Step 4 – Working with the XML response", "Step 5 – Working with data", "Simple timed XHR example", and "Related technologies".

<https://www.guru99.com/comparison-between-web-services.html>

The screenshot shows a comparison between SOAP and REST services. It includes a table of contents, sections on "What is SOAP?" and "What is REST?", and a section on "Difference Between SOAP and REST". To the right, there are two ads: one for "Tout Creative Cloud" and another for "Active Directory SSPR Tool".

<https://www.geeksforgeeks.org/difference-between-primary-key-and-foreign-key/>

10.

BILAN

Ce projet m'a permis d'apprendre à utiliser des langages HTML, CSS et JavaScript pour développer des sites internet statiques et dynamiques.

C'est très intéressant d'apprendre la technologie AJAX pour envoyer des requêtes asynchrones vers des serveurs sans recharger la page. Grâce à cela, j'ai pu rendre l'application plus réactive et dynamique.

C'est une expérience enrichissante de pouvoir réussir à faire marcher le programme malgré toutes les erreurs rencontrées auparavant. D'autres outils importants utilisés dans ce projet sont le MPD et les API's. Ces outils indispensables m'ont permis de mieux comprendre comment créer une interface entre le front-end et le back-end. La méthode CRUD (Create, Read, Update, Delete) (Créer, lire, mettre à jour, supprimer) m'a donné une meilleure compréhension de la gestion de données. Il me permet d'approfondir mes compétences en développement web qui sont fondamentales pour évoluer dans le monde de l'informatique.

Ce projet m'a encouragé à adopter de bonnes pratiques de développement telles que la validation des données, la gestion des erreurs et la sécurisation des requêtes pour prévenir les vulnérabilités potentielles.

Certes, la réalisation de ce projet m'a apporté une satisfaction et de la motivation pour continuer mon évolution en tant que développeuse web.