

Université de
Paul-Valéry et de
Montpellier



Projet de Webmapping

LES ÉTABLISSEMENTS CINÉMATOGRAPHIQUES EN FRANCE

ARCHITECTURE 3/3: CLIENT, SERVEUR, BASE DE DONNÉES



Geomagicien

M2 Géomatique, 2024/2025

Sommaire

du rapport du projet de
Webmapping sur les établissements
cinématographiques

- 01 **ARCHITECTURE DU PROJET** page 1

- 02 **NOS CHOIX TECHNOLOGIQUES** page 2

- 03 **DIAGRAMME DU PROJET** page 5

- 04 **LES DONNÉES ET LANGAGES DE PROGRAMMATIONS** page 6

- 05 **INTERFACES HOMME-MACHINE** page 8

- 06 **CONCLUSION ET SOURCES** page 10

Une architecture 3/3

Tout d'abord, je vais vous présenter le but et le fonctionnement du projet.

Le but final du projet est de créer une application web de cartographie en ligne sur une thématique choisie avec des formulaires pour interagir avec notre base de données en utilisant son pc comme serveur local (localhost).



Cela permettra de d'effectuer des recherches depuis une page web, d'interroger une base de données et de renvoyer et afficher les résultats dans la page web.

Mais pour faire fonctionner une telle application, plusieurs outils logiciels et matériels sont requis. Voici une liste des besoins pour ce projet à avoir pour pouvoir le réaliser dans de bonnes conditions:

Les besoins

- Un ordinateur fonctionnel avec une connexion internet
- OS Windows ou Linux de préférence
- Editeur de code (Notepad++, VS Code...)
- Serveur web avec Apache sous Firefox comme client pour le rendu final
- Une base de données spatiale (PostgreSQL, MySQL...)
- Un serveur cartographique (Mapserver, QGISServer, GeoServer)
- Interface Web (Leaflet, OpenLayers)
- PHP pour la connexion à la base de données
- Données cartographiques sur la thématique choisie en WFS/WMS
- Fonds de carte

Tableau présentant nos choix de technologies pour réaliser notre projet

	NOM	Utilisation
Base de données	PostgreSQL avec extension PostGIS	Qgis via DB Manager et PGAdmin
Serveur cartographique	GeoServer	Site web en localhost
Client Web	Firefox	Logiciel du navigateur
Serveur Web	Apache	Installation via leur site officiel et UniGetUI avec Apache HTTP Server ou Tomcat ou Wampserver ou XAMPP pour récupérer mes données via le formulaire de recherche
Interface web pour le client	OpenLayers	Installer des fichiers dans mon arborescences de dossiers
Éditeur de code	VS Codium	Logiciel pour coder
Système d'exploitation	Windows 11	PC Portbale
Thématique	Établissements cinématographiques en France	Données ponctuelles de 2023 en shp via le site d'OpenData du Ministère de la culture (data.culture.gouv.fr)
Choix des langages de programmations	HTML/JS/PHP/CSS	Code via VS Codium et ordonné dans une arborescence de dossier claire

Pour construire notre architecture 3/3 en webmapping, nous avons choisi les outils technologiques les plus accessibles, les plus complets et facile d'utilisation pour que le plus de personnes, même les nons spécialistes géomaticiens puissent comprendre le fonctionnement et créer une application de webmapping sans difficulté.

De plus, nous voulions des logiciels OpenSource, Libre et gratuit notamment ceux conseillées par l'OGC (l'Open Geospatial Consortium) et OSGeo (Open Source Geospatial) qui promeut des logiciels standards et des normes internationales fiables pour une meilleure interopérabilité.

C'est pour cela que nous avons développé notre application web sous le système d'exploitation Windows 11, car c'est le plus répandu au monde et plus facile d'utilisation pour des personnes lambda.

Nous avons donc du prendre des logiciels compatibles et optimaux avec Windows 11 comparés aux spécificités plus complexes de Linux.

1. Tout d'abord, la base de données spatiales choisie est: PostgreSQL avec l'extension PostGIS. C'est l'une des base de données les plus connues utilisable sous windows, et elle peut intégrer des données spatiales, grâce à l'extension PostGIS qui effectuera les traitement en SQL liés aux analyses spatiales et donc à mes données spatiales que je vais télécharger. Pour utiliser ce logiciel, nous avons besoin de le télécharger sur notre ordinateur (client lourd) afin de créer et configurer notre base de données pour le projet via pgAdmin.

Ensuite, pour insérer nos données dans la base de données, nous sommes passés par DB Manager de QGIS en connectant QGIS à notre base de données PostgreSQL pour y ajouter des données notamment SHP dedans. Un autre solution possible était de passer par l'extension PostGIS de PostgreSQL pour en ajouter sans passer par QGIS.

2. Puis, le serveur cartographique choisi est: Geoserver, logiciel Open Source et libre car directement utilisable en localhost via le navigateur web comme Firefox pour le configurer, insérer des données et créer des flux WFS/WMS.

C'est le serveur cartographique le plus simple d'utilisation sous Windows car les autres sont plus optimisés sous Linux et nous avons fait le choix de rester sur Windows.

3. Pour le rendu cartographique final, nous avons choisi: OpenLayers qui est plus complexe que Leaflet mais qui est plus complet avec plus d'options possibles pour la carte en ligne. De plus, conseillé par l'OGC et développé par l'OSGEO, la communauté d'utilisateurs est assez importante ainsi que la documentation officielle d'aide. Les mises à jour sont aussi plus régulières, ce qui stabilise et renforce la robustesse de cet outil comparé à Leaflet qui est mieux pour des cartes basiques et des données moins lourdes. Cet outil OpenSource est adapté aux utilisateurs avancés de cartographies en ligne.

4. Pour faire fonctionner l'application web et rendre dynamique la page web avec PHP et la relier à la base de données PostgreSQL, nous avons besoin d'un serveur web. Sous Windows, nous avons plusieurs choix à notre disposition comme Xampp, Wampserver ou Tomcat qui sont des environnements de travail tout en un, mais nous avons fini par choisir: Apache HTTP Server, qui est le serveur web le plus utilisé au monde ; il est gratuit et opensource. Nous l'avons donc installé via son site web officiel et nous avons configuré ses fichiers comme httpd.conf pour changer les chemins de fichiers et le port (Port: 8083). Ainsi, pour faire fonctionner l'application de webmapping, il faut ouvrir notre page web en localhost et activer ce serveur.

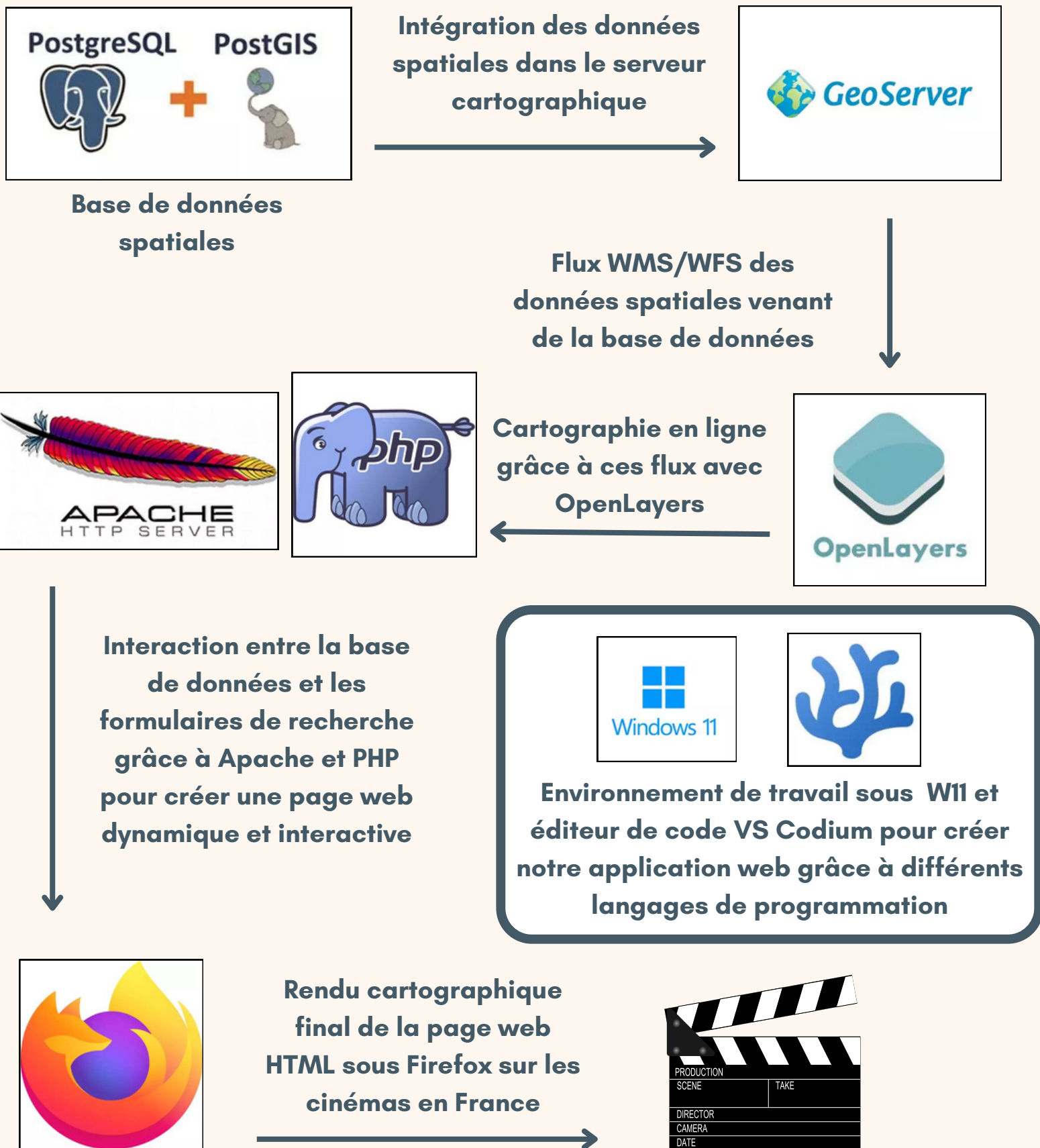
5. Ensuite, pour avoir une page web dynamique et la relier à notre base de données spatiales, nous avons opté pour PHP version 8.3.14 qui sera relié à notre serveur web Apache. Car c'est PHP avec des extensions comme pgsql (activée dans le fichier php.ini) qui permet de récupérer les données par des scripts en SQL, de les filtrer et de les afficher dans la carte selon les critères de recherches choisis via les formulaires disponibles dans notre application web.

6. Pour notre thématique finale, nous avons choisi: les établissements cinématographiques en France métropolitaine que nous avons téléchargés en shp sur le site officiel des données ouvertes du Ministère de la Culture (data.culture.gouv.fr). Ensuite, nous les avons insérées dans notre base de données PostgreSQL. Ce sont des données statistiques du Centre National du Cinéma et de l'image animée (CNC) de 2023, ce qui rend la donnée plus qualitative car presque à jour car récente.

De plus, pour compléter ces données, nous avons récupéré des données de l'IGN comme les limites administratives des communes, des départements et des régions grâce à ADMIN-EXPRESS. Avec aussi des flux WMS de l'IGN et d'OSM pour les fonds de carte dont une image satellite.

Diagramme du projet

Pour mieux comprendre le projet, un diagramme beaucoup plus visuel est parfait pour analyser les liens et les relations entre les logiciels et outils utilisés pendant le projet.



Les données et sources



1. Données thématiques:

Nous avons choisi de travailler sur les établissements cinématographiques de France Métropolitaine car nous sommes passionnés de films et séries, et l'industrie du cinéma en France est très importante.

Nous avons donc trouvé intéressant de comparer les cinémas et les territoires français entre eux.

De plus, nous avons récupéré les données des limites des régions, des départements et des communes pour avoir un meilleur contexte du lieu où se trouve le cinéma.

Sources:

data.culture.gouv.fr
IGN (ADMIN-EXPRESS)



2. Données pour les fonds de cartes:

Pour rendre la carte en ligne plus interactive, nous avons fait le choix de mettre 3 fonds de carte différents: un plan classique OpenStreetMap, un plan plus formelle français de l'IGN et une image satellite de l'IGN du satellite SPOT datant de 2023. Le fond de carte par défaut est celui d'OSM car c'est le plus répandu et connu, les utilisateurs de cette application et page web ne seront donc pas perdus.

Sources:

OSM
IGN (Géoservices)

Les fonctionnalités liées aux données et à la carte

- WFS pour la couche des cinémas et interagir avec la base de données
- WMS pour les fonds de carte grâce au serveur cartographique
- Formulaire de recherches avec bouton de recherche et de réinitialisation pour filtrer les cinémas par communes, régions, départements, programmeurs, et si le cinéma est d'art et d'essai ou non en interaction avec ma base de données grâce à PHP et au serveur web Apache.
- Choix des couches de données et des fonds de carte à utiliser
- Informations sur les cinémas en cliquant dessus
- Vue Monde
- Coordonnées en Lambert-93 et échelle cartographique
- Bouton "Home" pour revenir sur la France entière
- Bouton de zoom et de plein écran

Les langages de programmations utilisés et leurs fonctions

Pour programmer, nous avons séparé chacun de nos codes pour plus de visibilité, de lisibilité et plus de facilité pour déboguer le code. Nous avons donc 4 fichiers distincts (HTML/JS/PHP/CSS).



HTML:

Ce langage permet de créer une page web et de configurer l'Interface Web.

Elle récupère notamment via des liens vers les fichiers dont OpenLayers a besoin pour fonctionner mais aussi le chemin vers mes fichiers de styles ou images ou encore mon fichier JS pour faire afficher la carte dans la page web.

De plus, c'est dans ce code que nous avons inséré mes formulaires de recherches pour filtrer les cinémas selon plusieurs critères différents.



PHP:

Ce langage permet de créer des pages web "vivantes" et dynamiques avec une interaction et un lien vers PostgreSQL/PostGIS pour notamment récupérer et mettre à jour les données sur les cinémas via des requêtes SQL sur les différents champs que possède cette couche géographique.



JavaScript (JS):

Ce langage est spécifique à la cartographie en ligne et permet donc de créer une carte avec OpenLayers ici ou bien Leaflet.

Ce langage nous permet de récupérer les couches géographiques de la base de données via le serveur cartographique GeoServer grâce aux flux WMS ou WFS qu'il génère en localhost sur l'ordinateur.

De plus, je peux configurer ma carte en ligne avec le zoom, l'échelle, la vue monde etc. pour la personnaliser et avoir une expérience optimale de visualisation des cinémas. C'est aussi ici que je relie mon code PHP aux formulaires créés dans HTML pour les rendre utilisables et dynamiques.

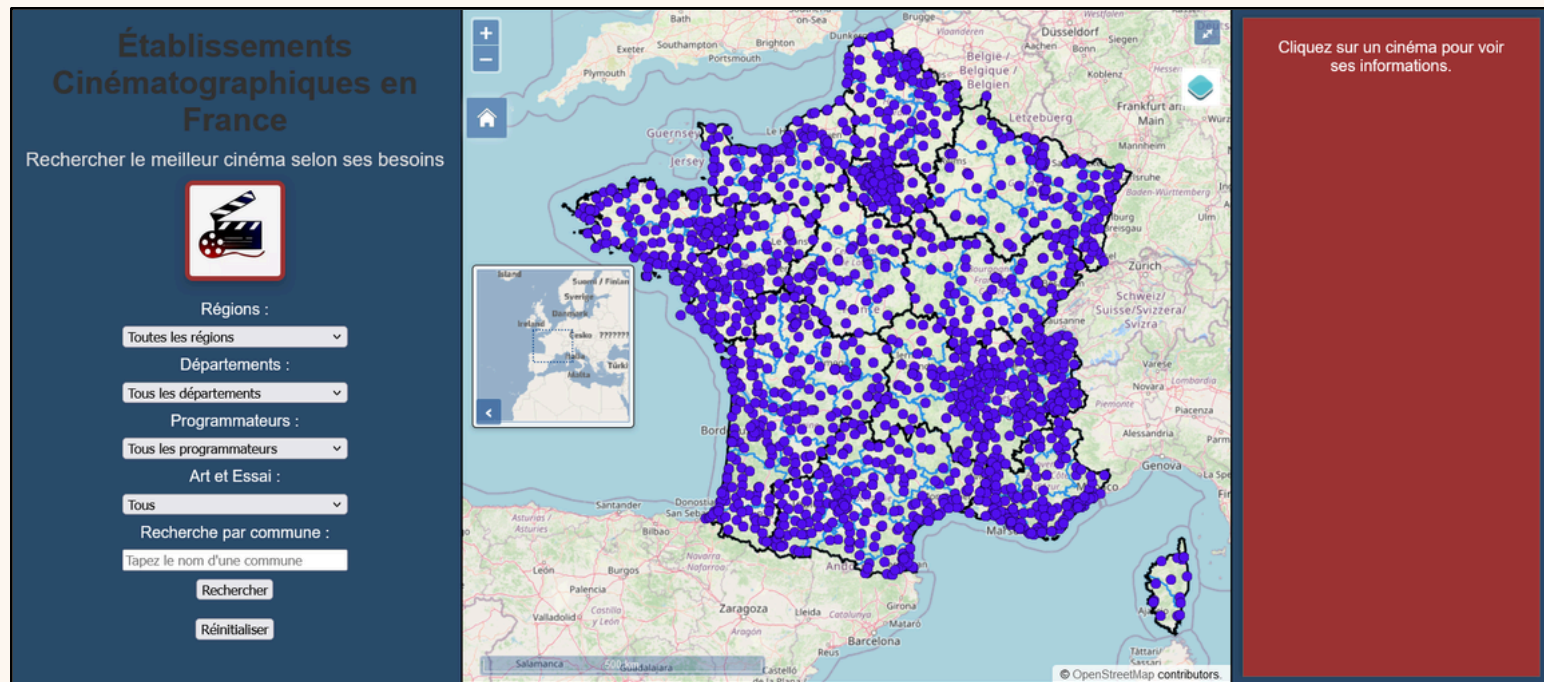


CSS:

Ce langage permet de changer la symbologie et le style de la carte mais aussi de la page HTML toute entière avec des paramètres et des options plus précis que dans le code HTML directement. Chaque élément présent dans la page web HTML est donc personnalisable individuellement. Cela permet alors de changer la position, la couleur, l'arrière-plan, la taille de la police, la police...

Page web HTML finale en localhost

Grâce au serveur web Apache HTTP Server et PHP, la page web finale (cinemas.html) fonctionne correctement en localhost sur notre machine et peut se connecter facilement à la base de données PostgreSQL pour effectuer les recherches de cinémas par des contraintes SQL via les formulaires.



L'interface Homme-Machine ou Front End est la partie visible d'un site web, incluant la mise en page, les graphiques et l'interface utilisateur. C'est la partie finale du projet et la plus importante pour les utilisateurs externes car c'est ce qui rend le site attractif, compréhensible et facile d'utilisation pour les clients web:

- Les technologies principales du Front End sont HTML, CSS et JavaScript, qui permettent de structurer, styliser et rendre interactif le contenu.
- Il joue un rôle crucial dans l'expérience utilisateur, influençant directement l'esthétique et la fonctionnalité des sites et applications web.
- La carte en ligne avec OpenLayers est a été pensée pour qu'elle soit le plus simple d'utilisation et la plus épurée possible, car le plus important ici, c'est de localiser l'emplacement des établissements cinématographiques.

De plus, pour améliorer l'expérience utilisateur de l'interface web, plusieurs options possibles sont à ajouter pour des projets qui seront réellement mis en ligne: par exemple, nous pouvons ajouter un système de géolocalisation qui sélectionne les cinémas de la commune dans laquelle nous nous trouvons.

Comme nous montre l'image, la page est séparée par trois blocs distincts:

- à gauche, nous avons tous les formulaires disponibles et utilisables pour faire de recherches sur la base de données des cinémas de France avec le bouton "Rechercher" pour lancer la recherche et le bouton "Réinitialiser" pour désélectionner les cinémas sélectionnés par la recherche précédente. De plus, les recherches combinées entre plusieurs formulaires sont possibles. Par exemple, rechercher les cinémas d'art et d'essai dans une commune spécifique est possible pour plus de précision dans sa recherche.
- ensuite au centre nous avons la carte en ligne avec OpenLayers qui prend le plus de place dans la page car le plus important pour des données et recherches géographiques et spatiales est de les visualiser dans un projet de webmapping. Cela permet donc de visualiser la localisation des différents cinémas de France avec différentes couches disponibles comme les limites régionales, départementales et communales ainsi que différents fonds de carte (OSM, IGN ou satellite) que l'on peut changer à souhait dans le LayerSwitcher d'OpenLayers pour la gestion des couches, la vue "Monde" pour un contexte international, une échelle graphique, un mode plein écran et un bouton "Home" qui permet de revenir sur la page d'accueil des cinémas et sur le zoom de la France. Nous avons aussi implémenter les coordonnées en Lambert-93 qui s'affichent au déplacement de la souris et donne la localisation du point sur lequel la souris se trouve.
- Pour finir, la partie de droite est celle qui prend le moins d'espace. Elle permet après avoir cliqué sur un cinéma, d'avoir les informations de ce dernier disponibles depuis la base de données des établissements cinématographiques.

Plusieurs traitements ont donc été réalisés pour créer cette page web finale notamment en Front End pour l'interface mais aussi en Back End pour les interactions entre les technologies en arrière-plan pour faire fonctionner l'application web ici en localhost.

C'est la partie invisible responsable de la gestion des données, de la logique d'application et de la communication avec le Front End expliqué précédemment permettant d'avoir une expérience utilisateur stable et optimale.

- Nous utilisons ici des technologies comme PHP et la base de données PostgreSQL pour assurer le bon fonctionnement des systèmes.

Conclusion

Ce projet de webmapping sur les établissements cinématographiques en France a permis de concevoir et de déployer une application web cartographique interactive, offrant une visualisation et une exploration détaillées des cinémas, répartis sur le territoire national. En combinant des technologies modernes et performantes telles que PostgreSQL/PostGIS, GeoServer, OpenLayers, Apache et PHP, nous avons créé une solution robuste et évolutive répondant aux besoins d'analyse géospatiale et de recherche des utilisateurs sur la thématique des cinémas mais notre application web est également ouverte à de nombreuses possibilités d'amélioration et d'expansion.

Ce projet illustre alors l'importance des technologies géospatiales pour la visualisation et l'analyse des données territoriales sur des données et thématiques variées. En alliant des compétences en développement web, en gestion des bases de données spatiales et en cartographie, ce projet constitue une vitrine idéale pour faire connaître le domaine du webmapping et de la géomatique.

Sources et bibliographies

- <https://geoservices.ign.fr/catalogue>
- <https://geoservices.ign.fr/adminexpress>
- <https://geoservices.ign.fr/documentation/services/utilisation-sig/tutoriel-qgis/wms-wmts>
- <https://wiki.openstreetmap.org/wiki/FR:Serveurs/wms.openstreetmap.fr>
- <https://data.culture.gouv.fr/explore/dataset/etablissements-cinematographiques/information/?location=5,46.57405,2.40395>
- <https://openlayers.org/>
- https://geotribu.fr/articles/2008/2008-08-22_introduction-a-openlayers/
- <https://docs.geoserver.org/latest/en/user/tutorials/index.html>
- <https://docs.geoserver.org/latest/en/user/>
- <localhost:8080/geoserver/web/?3>
- <https://httpd.apache.org/download.cgi>
- <https://httpd.apache.org/docs/2.4/platform/windows.html>
- <https://www.php.net/>
- <https://www.postgresql.org/docs/16/index.html>
- <https://postgis.net/>
- https://postgis.net/documentation/getting_started/install_windows/
- <https://www.sigterritoires.fr/index.php/debuter-avec-postgresqlpostgis-introduction-a-pgadmin3/>
- <https://www.sigterritoires.fr/index.php/debuter-avec-postgrespostgis/>
- Claude AI
- ChatGPT
- Mistral AI