

# Une SPA à la main

Rendu 1

**4TTV906U** : Programmation web

**Étudiant** : *Aguibou Barry*

**Responsable :** *M. Laurent Réveillère*  
**Département :** Informatique & Master 2 (Génie Logiciel)  
**Date :** November 7, 2022

---

# 1 Introduction

L'objectif dans ce devoir est de réaliser une SPA (Single Page Application) permettant de faire des requêtes sur le site <https://www.prevision-meteo.ch/services/> pour chercher des flux de données *JSON* correspondant à la météo d'une ville donnée sur toutes les localités de Suisse, France et Belgique.

## 2 Fonctionnalités

### 2.1 Version de base

Dans la version de base, l'objectif est de pouvoir réagir au clic sur le bouton *Afficher* pour aller chercher les flux de données *JSON* sur le site <https://www.prevision-meteo.ch/services/> correspondant à la ville saisie par l'utilisateur pour ensuite les afficher sur notre site.

### 2.2 Version finale

Ici, le but est d'intégrer une carte sur la version de base et quand l'utilisateur clic sur cette dernière, on récupère les coordonnées GPS de cet endroit pour ensuite chercher les informations correspondantes sur le site <https://www.prevision-meteo.ch/services/> et afficher par la suite le résultat obtenu par la requête sur notre site.

## 3 Technologies

Dans ce rendu, seuls les langages **JavaScript**, **HTML** et **CSS/Bootstrap** sont permis donc aucun framework front-end n'est autorisé.

- **HTML** : Mon code est quasiment le même que celui qui nous est fourni de base, avec l'ajout de quelques affichages par défaut (message d'accueil, carte et pied de page). L'affichage de la météo est fait dynamiquement dans le code JavaScript.
- **JavaScript** : Tout le travail est fait par celui-ci de la recherche à l'affichage des données sur le site et comme dans mon code *HTML* aucune arborescence n'est créée au lancement pour l'affichage des données de la météo, après avoir récupéré les données, je crée dynamiquement cette arborescence dans le DOM pour l'affichage des données.
- **CSS** : J'ai choisi à n'utiliser que du CSS pour toute la mise en forme du site même si cela aurait pu être plus simple avec du **Bootstrap**. Ce choix est tout simplement pour me permettre d'améliorer mon niveau.
- **Leaflet** : Une bibliothèque JavaScript libre, pour l'intégration de la carte.

## 4 Travail réalisé

Toutes les fonctionnalités des deux versions ont été réalisées et marchent correctement. La figure 1 présente la page d'accueil au lancement du site.

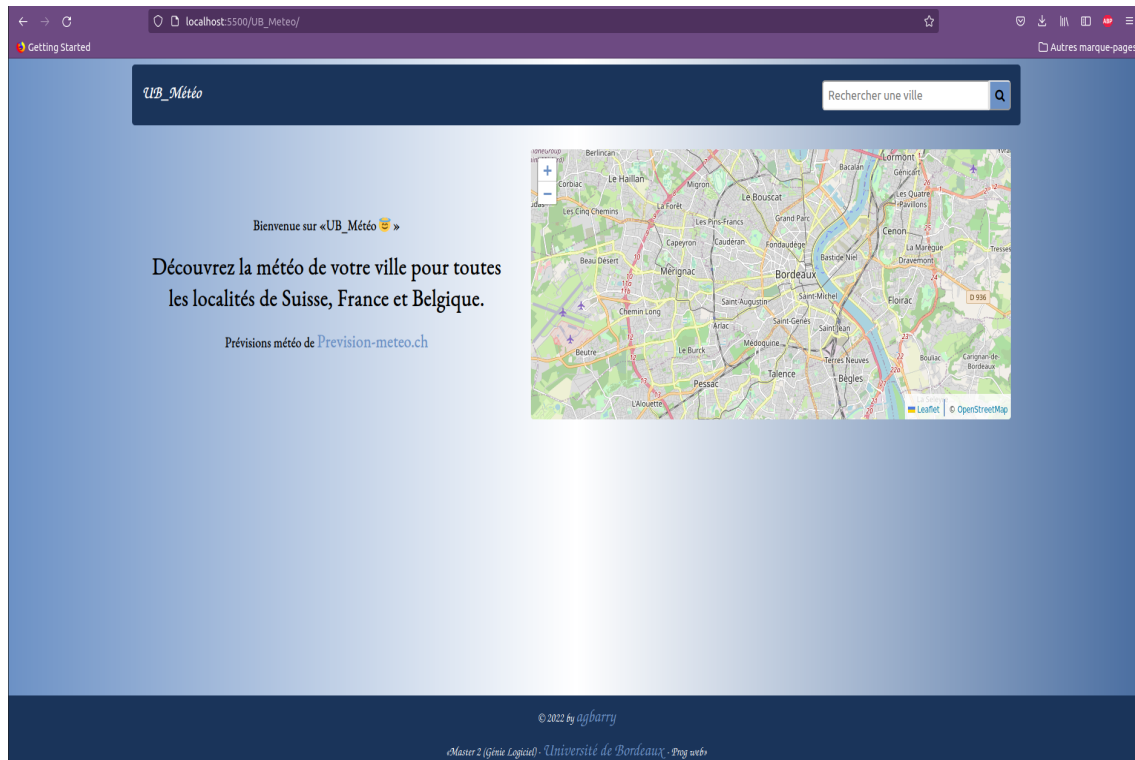


Figure 1: Page d'accueil

### 4.1 Recherche par la barre de recherche

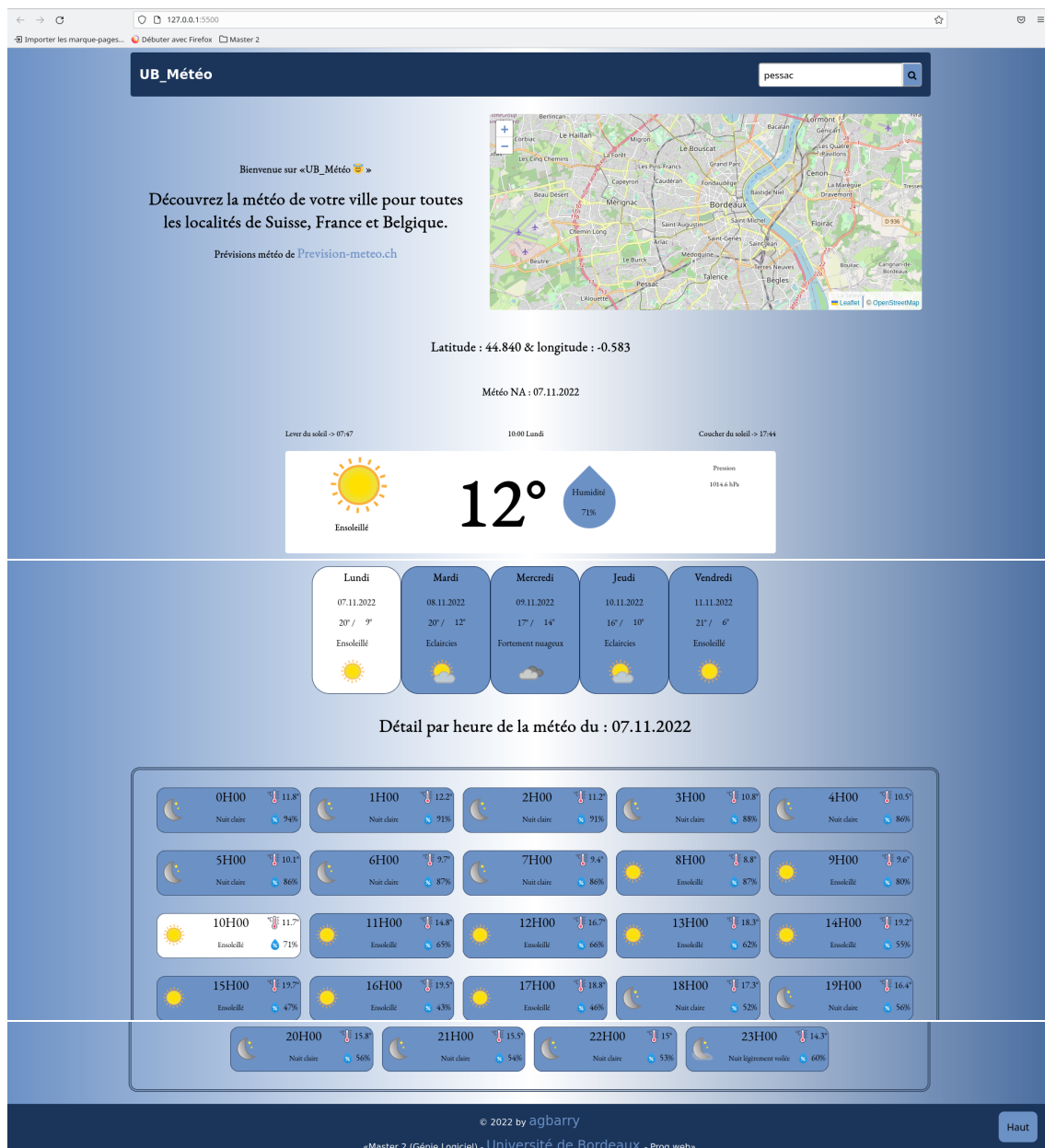
Dans la figure 2, un exemple de recherche de la ville d'**Orléans** par la barre de recherche, le site hébergeur des flux de données n'autorisant pas les accents, j'ai fait en sorte de supprimer celles-ci avant de faire la requête et ici, on voit bien qu'on a un résultat de notre requête et cela marche pour toutes les autres villes. Sur cet affichage, j'ai fait en sorte de mettre en évidence (font *blanc*) toutes les données courantes (*jour* et *heure*). Le détail par heure de la météo dépend du jour sélectionné par l'utilisateur par défaut, c'est le jour-j qui est affiché, on peut éventuellement afficher la prévision des autres villes en cliquant tout simplement sur celui-ci.



Figure 2: Recherche de la ville Orléans

## 4.2 Recherche par la carte

Dans celle-ci, j'ai fait la recherche par la *longitude* et la *latitude* de la zone cliquée par l'utilisateur et le résultat est quasiment le même que celui de la recherche par la saisie à la seule différence que certaines données (*Pays* et *Ville*) ne sont pas affichées.



**Figure 3:** Recherche par la carte au clic sur Bordeaux