

## BRIEF PROJET

# *Afficher la météo dans les transports en commun*

Projet pour la préparation à la journée de sélection pour le parcours de formation  
Concepteur.trice Développeur.euse D'application



## Prérequis

- HTML/CSS (intégration de composants et interfaces standards)
- JavaScript (DOM, fetch)

**Durée estimée :** 1 à 2 journées

# Énoncé du brief

## Scénario

Vous êtes développeur web pour le compte d'une agence web. Votre agence a signé un contrat pour le développement d'interfaces météo à destination des usagers du réseau de transport en commun de plusieurs villes de taille moyenne en France. Les écrans seront intégrés aux écrans d'information dans les stations et dans les transports.

Les écrans doivent être programmés avec les technologies web, embarquées dans la webview du système des écrans de la compagnie de transports en commun de la ville.

Exemples d'interfaces météo :

- <https://github.com/PritamSarbajna/simple-weather-website>
- <https://github.com/abdellatif-laghaj/weather-web-app>

L'interface ne doit pas inclure de moteur de recherche pour la localisation de l'information météo.

Mais, elle doit inclure un fichier de configuration (JSON par exemple) dans lequel l'information de la ville concernée sera entrée par l'entreprise de transport, et utilisée par votre code pour récupérer les bonnes données météo.

## Livrables

Les fichiers source seront partagés via l'url d'un dépôt Git accessible en ligne (Github, Gitlab...).

cf.  Les Tutos - Débuter avec Git et Github en 30 min

# L'organisation de votre travail

## S'auto-former

Suivez la proposition de [parcours d'auto-formation](#) au besoin.


## Développer l'application

- chercher et choisir une API ouverte pour la récupération des données météo,
- créer les différents fichiers nécessaires, c'est à dire :
  - un fichier HTML (index.html),
  - à minima un fichier CSS (style.css),
  - le fichier de configuration de la ville (conf.json)
  - et à minima un fichier Javascript (meteo.js),
- programmer la récupération des données météo depuis l'API. Déboguer en console afin de valider votre procédure,
- intégrer l'interface d'affichage des données météo,
- intégrer la mise à jour des données météo une fois par heure,

## Tester l'application et livrer

- validez, puis livrez votre projet sur un dépôt Git en ligne.

## Ressources

- Git et un dépôt en ligne :  Les Tutos - Débuter avec Git et Github en 30 min

## Critères de qualité

### Critères pour pour le projet

- Les consignes sont respectées dans leur intégralité,
- la structure html est fonctionnelle et cohérente,
- les règles css sont fonctionnelles et effectivement liées à la page html.  
→ *il n'y a pas de critères concernant les visuels, le design et la mise en page. La ville concernée et les informations météo doivent cependant être lisibles sans doute possible,*
- les données météo récupérées sont effectivement celle de la ville configurée dans le fichier dédié,
- les données météo sont correctement restituées depuis l'interface,
- le rafraîchissement des données météo toutes les heures est intégré au programme,
- les données sont effectivement rafraîchies toutes les heures.

### **Critères pour le jour J**

- Le candidat décrit les tâches qu'il a effectuées pour parvenir à un résultat,
- le·a candidat·e explique le fonctionnement logique de la récupération des données météo,
- le·a candidat·e explique les règles d'intégration des interfaces et des données météo,
- le·a candidat·e a identifié les améliorations possibles,
- le·a candidat·e a identifié comment ces améliorations pourraient être implémentées concrètement et les apprentissages qu'il ou elle devrait faire pour y arriver.