

# **Document d'initialisation et de spécification**

# Table des matières

## [Table des matières](#)

### [I/ Introduction](#)

### [II/ Planning](#)

#### [Organisation](#)

### [III/ Diagramme de cas d'utilisation](#)

#### [i. Acteurs](#)

#### [ii. Cas d'utilisation](#)

#### [iii. Hypothèses](#)

#### [iv. Diagramme](#)

### [IV/ Exigences fonctionnelles et non-fonctionnelles](#)

#### [i. Exigences fonctionnelles](#)

#### [ii. Exigences non-fonctionnelles](#)

### [V/ Analyse des risques liés à la sécurité](#)

### [VI/ Tests](#)

### [VII/ Manuel d'utilisation](#)

# I/ Introduction

L'objectif de cette série de travaux pratiques est d'appliquer les principes de l'ingénierie logicielle pour développer une application logicielle de haute qualité. Dans ce contexte, nous avons pour mission de concevoir une application logicielle appelée "AirWatcher".

Ce document spécifie les fonctionnalités de l'application AirWatcher, développée pour une agence gouvernementale chargée de la surveillance de la qualité de l'air. Cette application permettra l'analyse des données de capteurs installés sur un large territoire. Le but de cette application est ainsi de générer des statistiques sur la qualité de l'air en proposant diverses fonctionnalités comme calculer la moyenne de la qualité de l'air ou générer des valeurs de qualité de l'air pour des positions géographiques précises.

En outre, AirWatcher permettra de mesurer l'impact des purificateurs d'air industriels, d'identifier les zones ayant une qualité d'air similaire et de permettre aux particuliers de participer à la génération de données sur la qualité de l'air en installant des capteurs fixes à leur domicile. Les utilisateurs recevront des points pour l'utilisation de leurs données, mais l'application permettra également à l'agence gouvernementale d'analyser la fiabilité des données fournies par les capteurs des particuliers.

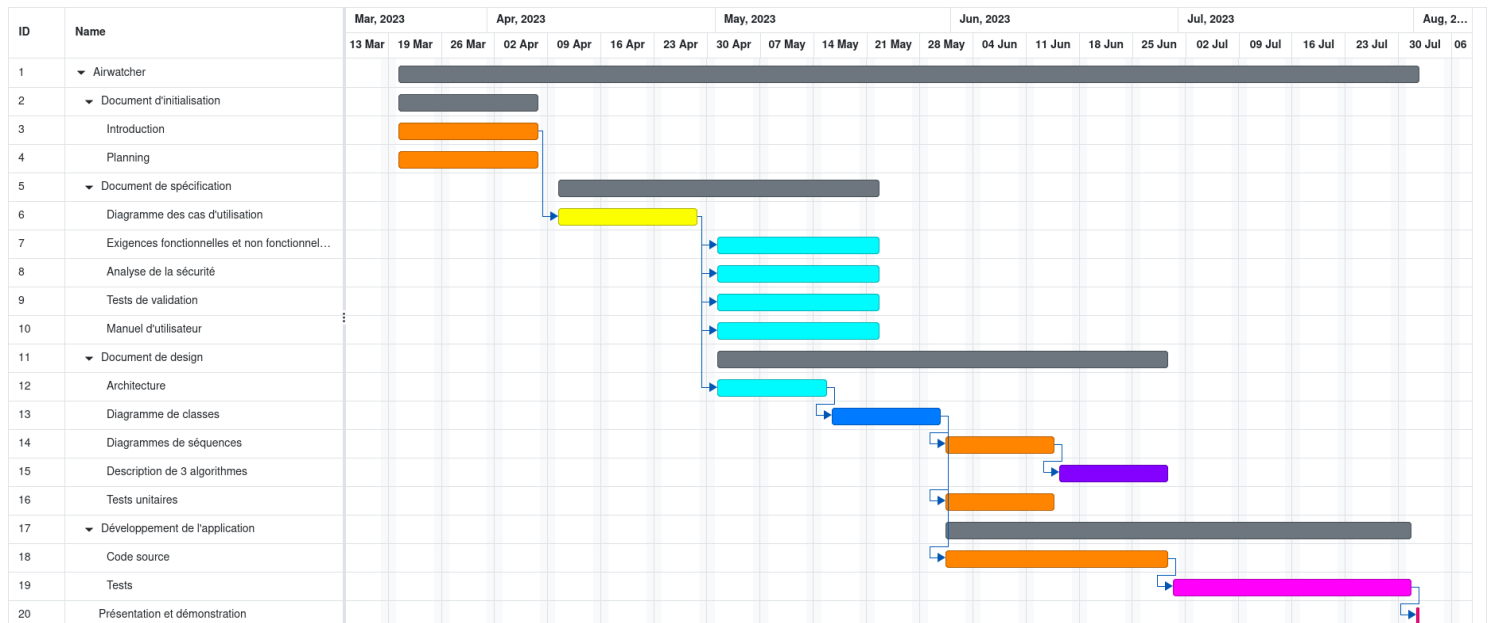
Les algorithmes de l'application AirWatcher seront efficaces et mesurables, et l'application offrira une interface utilisateur adaptée aux exigences et aux privilèges de chaque utilisateur.

## II/ Planning

### Organisation

L'équipe est constituée des binômes 3427 (NIZON Martin et NGUYEN Mathis) et B3421 (CHANTREL Thibaut, SAYSANA Hugo)

### Diagramme de Gantt :



(pour voir l'image en plus grand :

[https://drive.google.com/file/d/1jEuGSAi5UeJnT8e2CSzDBZQ0ZALemgRe/view?usp=share\\_link](https://drive.google.com/file/d/1jEuGSAi5UeJnT8e2CSzDBZQ0ZALemgRe/view?usp=share_link))

## III/ Diagramme de cas d'utilisation

### i. Acteurs

- Membre de l'agence gouvernementale : il s'agit d'un employé de l'agence gouvernementale qui travaille sur l'application. Il devra s'être authentifié. Il sera dénommé "membre de l'agence".
- Employé de l'un des fournisseurs de purificateur d'air. Il devra s'être authentifié. Il sera dénommé "fournisseur".
- Particulier possédant un capteur : il s'agit d'un utilisateur qui possède au moins un capteur inscrit dans le serveur. Il devra s'être authentifié. Il sera dénommé "particulier".
- Utilisateur non-authentifié : il s'agit d'un utilisateur non-connecté au serveur d'authentification. Il sera dénommé "utilisateur non-authentifié".
- Le serveur : il s'agit de la machine sur laquelle les données sont récupérées par requêtes provenant de l'application. Il sera dénommé "serveur".

### ii. Cas d'utilisation

- Obtenir la moyenne de la qualité de l'air sur une position exacte à un moment donné
- Obtenir la moyenne de la qualité de l'air sur une position exacte sur une période donnée
- Obtenir la moyenne de la qualité de l'air dans un rayon en un moment donné
- Obtenir la moyenne de la qualité de l'air dans un rayon sur une période donnée
- Trouver les capteurs similaires à un capteur sélectionné
- Identifier les zones à qualité d'air similaire
- Donner la qualité de l'air à une position exacte
- Voir ses points pour les propriétaires privés de capteurs et les employés de l'agence
- Observer les bénéfices d'un purificateur d'air dans sa zone d'action
- Mesurer la performance des algorithmes en temps d'exécution (pour que les membres de l'agence puissent voir le temps d'exécution des algorithmes)
- Ajouter un point (*lorsqu'une requête utilise les données d'un particulier*)
- Exclure lors d'une requête les données d'un capteur appartenant à un utilisateur qui possède des capteurs présents dans la liste noire. La liste noire renseigne tous les capteurs ayant été marqués (automatiquement ou manuellement) comme étant défectueux et/ou appartenant à un utilisateur malveillant.
- Détecter des capteurs donnant des informations fausses et inscription de ce dernier dans la liste noire.

### iii. Hypothèses

On décide de faire les hypothèses suivantes :

- Tous les utilisateurs n'ont pas besoin de créer un compte pour utiliser certaines fonctionnalités de l'application. En se connectant, ils accéderont ensuite aux fonctionnalités dédiées à leur rôle.
- Tous les utilisateurs peuvent obtenir la qualité de l'air et la moyenne de la qualité de l'air à n'importe quel endroit pour un moment donné ou sur une période de temps donnée.
- Seuls les fournisseurs et les membres de l'agence peuvent voir l'impact de n'importe quel purificateur d'air, qu'il lui appartienne ou pas.
- Lorsqu'il n'y a pas de capteur dans une zone, on calcule la qualité de l'air dans celle-ci par interpolation (on précisera le fonctionnement plus tard).
- Seuls l'utilisateur et les membres de l'agence peuvent voir les points de cet utilisateur.
- L'ensemble des capteurs d'un particulier malveillant seront placés dans une liste noire. Les capteurs présents dans cette liste noire seront exclus lors de toute requête.

#### iv. Diagramme

Ainsi, on en vient au diagramme de cas d'utilisation suivant :

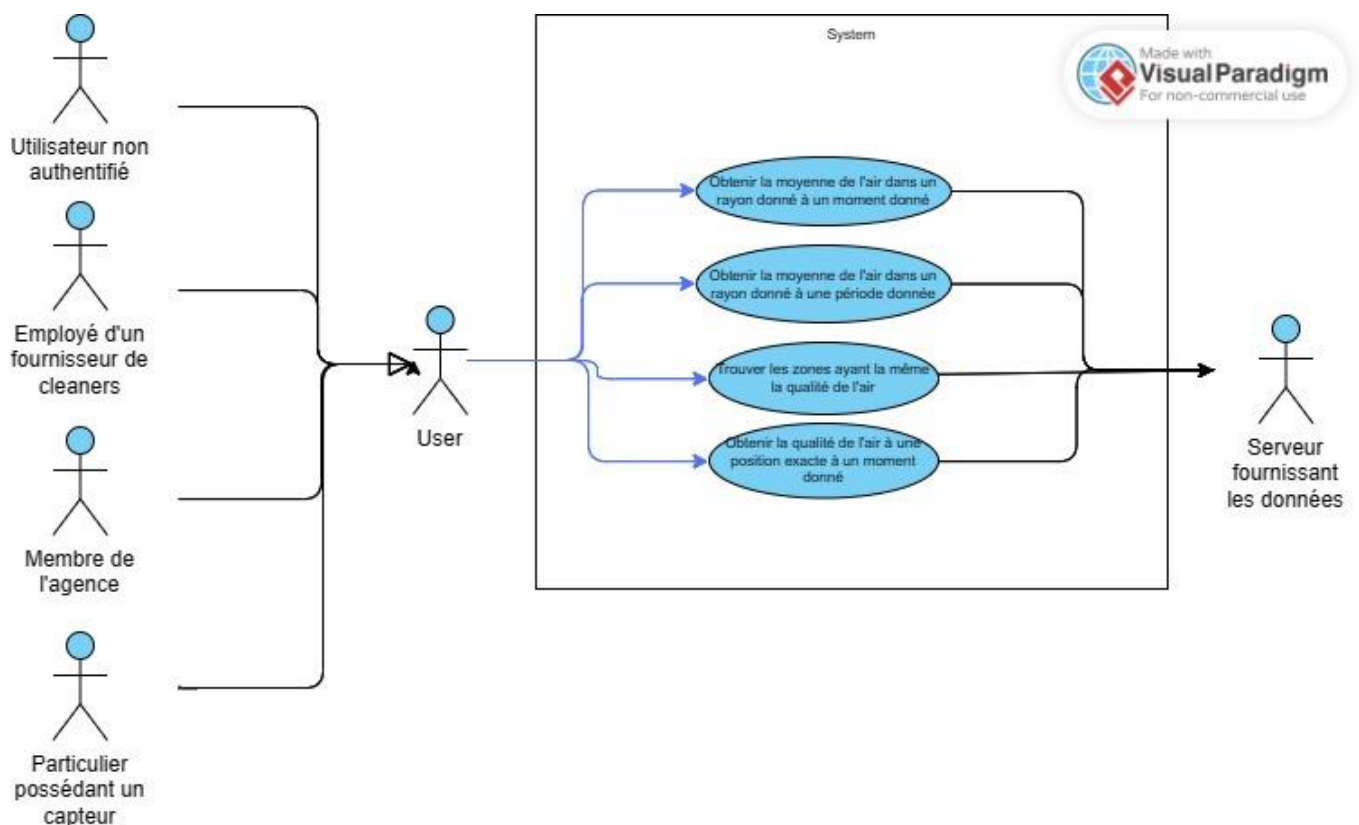


Figure 1 : Diagramme partiel de cas d'utilisation des users de l'application AirWatcher

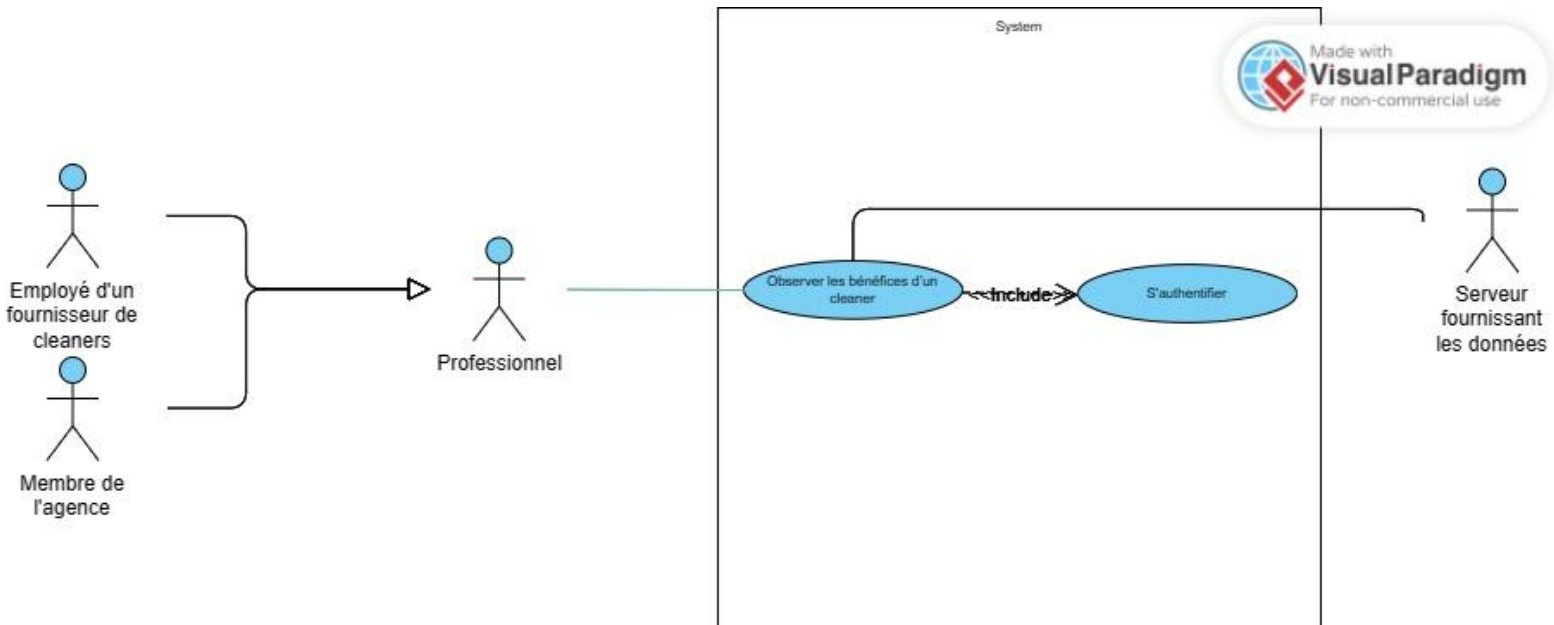


Figure 2 : Diagramme partiel de cas d'utilisation des professionnel de l'application AirWatcher

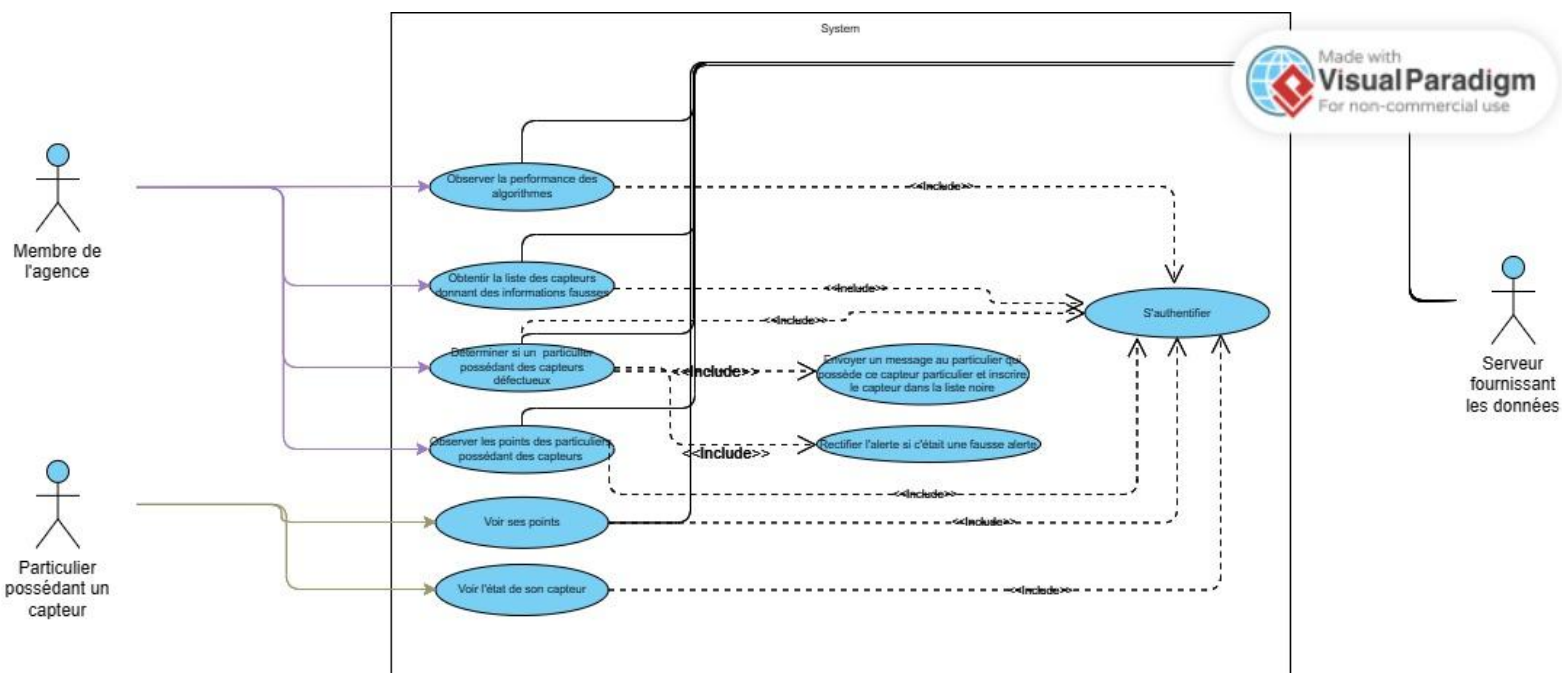


Figure 3 : Diagramme partiel de cas d'utilisation des membre de l'agence et des particuliers possédant un capteur de l'application AirWatcher

## IV/ Exigences fonctionnelles et non-fonctionnelles

### i. Exigences fonctionnelles

**1** Le système permettra aux utilisateurs de se connecter et de se déconnecter

**1.1** Le système permettra aux membres de l'agence, aux fournisseurs et aux particuliers de s'authentifier. Il octroiera à chacun de ces utilisateurs leurs droits associés (qu'un utilisateur non-authentifié ne possède pas).

**1.2** Le système permettra aux membres de l'agence, aux fournisseurs et aux particuliers de se déconnecter. Il retirera les droits octroyés.

**2** Le système sera capable d'informer les utilisateurs de la qualité de l'air. L'utilisateur pourra spécifier une position géographique précise ou un rayon, ainsi qu'un jour ou une période donnée.

**2.1** Si l'utilisateur choisit une position, si la position donnée par l'utilisateur est identique à celle d'un capteur, la qualité de l'air retournée sera celle du capteur. Sinon, la valeur retournée sera la valeur moyenne des trois capteurs les plus proches, pondérés par leur distance à la position donnée par l'utilisateur.

**2.2** Si l'utilisateur spécifie un rayon, la valeur retournée sera la valeur moyenne des capteurs présents dans la zone. Si aucun capteur n'est présent dans la zone, la valeur retournée sera nulle.

**2.3** Si l'utilisateur spécifie une période, le système lui retournera la valeur moyenne de la qualité de l'air sur cette période.

*Note : Les capteurs de la liste noire ne seront pas pris en compte dans les calculs.*

**3** Le système devra être capable de classer les capteurs par similitude avec un capteur sélectionné par l'utilisateur. Le système devra noter tous les autres capteurs en fonction de la ressemblance des mesures de la qualité de l'air, et fournir à l'utilisateur un classement de ces capteurs en fonction de la note. L'objectif de cette exigence sera d'identifier les zones à qualité d'air similaires.

*Note : Les capteurs de la liste noire ne seront pas pris en compte dans les calculs.*

**4** Le système devra gérer un système de points :

**4.1** Un point sera attribué à un particulier à chaque fois qu'un de ses capteurs sera utilisé lors d'une recherche.



*Remarque : si l'un des capteurs du particulier est dans la liste noire, aucun de ses capteurs ne pourra être utilisé lors de recherches. Son nombre de points ne pourra pas évoluer.*

**4.2** Le système permettra aux particuliers qui se sont authentifiés de voir leurs points.

**4.3** Le système pourra fournir les points de tous les utilisateurs uniquement aux membres de l'agence

**5** Le système permettra aux membres de l'agence d'indiquer si le capteur d'un particulier est fiable ou non.

**5.1** Si un capteur est indiqué comme non-fiable alors l'ensemble des capteurs appartenant au particulier sont déclarés non-fiables. L'ensemble de ces capteurs sera ajouté à la liste noire et ils ne pourront plus être utilisés pour des recherches.

**5.2** Si un capteur est indiqué comme fiable, ce capteur pourra être retiré de la liste noire manuellement par un membre de l'agence, il pourra apparaître dans les recherches futures, et le compteur de points de son propriétaire (utilisateur privé) pourra augmenter de nouveau.

**6** Le système permettra aux membres de l'agence et aux fournisseurs de consulter les informations relatives à l'impact d'un purificateur d'air sélectionné.

**6.1** Le système pourra fournir le rayon de la zone impactée par le purificateur.

**6.2** Le système pourra fournir un indice du niveau d'amélioration de la qualité de l'air du purificateur. Cet indice sera calculé en faisant le rapport entre la moyenne de la qualité de l'air durant la semaine actuelle et la moyenne de la qualité de l'air durant la semaine juste avant l'installation du purificateur, à sa localisation. Ce rapport est calculé à titre indicatif, il est censé être comparé à d'autres chiffres tel que le rapport de l'évolution de la pollution pour pouvoir exprimer tout son sens. Ce dernier est calculé de la sorte :

$$I_{\text{niveau d'amélioration}} = \frac{\text{Moyenne(semaine actuelle)}}{\text{Moyenne(semaine précédant l'installation)}}$$

**7** Le système devra fournir le temps d'exécution de chaque algorithme à un membre de l'agence s'il le demande. Ce temps d'exécution sera donné en millisecondes.

## ii. Exigences non-fonctionnelles

**1** Le système devra permettre d'obtenir des résultats aux requêtes dans un temps convenable.

**1.1** Le résultat d'une requête ne doit pas prendre plus de 5 secondes. Cette valeur a été choisie après concertation au sein de notre groupe afin d'éviter de trop longues attentes pour les utilisateurs.

**1.2** Si le délai est trop long  $\Rightarrow$  abandon de la requête et affichage d'un message d'erreur.

**2** Le service devra être utilisable 24/7.

**3** L'interface mise en place doit être facile d'utilisation et optimisée pour les usages en fonction de l'utilisateur. Elle devra être claire et faire en sorte de minimiser les erreurs de saisie de l'utilisateur.

**4** Il faut que l'application offre un service fiable dans le temps, peu importe la série d'actions effectuée, c'est-à-dire que les performances de l'application (services, accès aux données, ...) ne doivent pas se dégrader avec son utilisation (par exemple en veillant à ce qu'il n'y ait pas de fuites de mémoire).

**5** L'application doit être développée dans un souci de laisser place à de futures fonctionnalités.

## V/ Analyse des risques liés à la sécurité

Figure 2 : Tableau de l'analyse des risques liés à la sécurité

Système	Atout	Vulnérabilité	Attaque	Risque	Niveau d'impact	Contre-mesure (ne seront pas implémentées)
Application AirWatcher	Compte d'un utilisateur	Mot de passe devinable (par bruteforce ou par fuite de données, etc)	Attaque des informations d'authentification d'un particulier par force brute ou par dictionnaire, etc	Modification, suppression ou falsification des données liées au(x) capteur(s) et/ou au(x) purificateur(s) d'air	Elevé	Ajouter un système d'authentification avec mot de passe vérifié et suivant certains critères
				Accéder et modifier les données liées à n'importe quels utilisateurs (login, points, capteur(s) ou cleaner(s))	Elevé	
	Points d'un utilisateur	Les utilisateurs peuvent faire des requêtes sollicitant n'importe quel capteur	Un utilisateur effectue volontairement beaucoup de requêtes avec ses propres capteurs	Un utilisateur peut artificiellement augmenter son nombre de points	Moyen	Implémenter la règle suivante : un utilisateur ne gagne pas de point si une requête qui utilise un de ses capteurs provient de ce même utilisateur
	Données d'un utilisateur	Un membre de l'agence a accès à toutes les données renseignées par les utilisateurs	Un membre de l'agence accède aux données d'un utilisateur	Usurpation d'identité, vente de données (à voir selon la politique d'Airwatcher)	Elevé	Faire signer aux membres de l'agence une charte les engageant à ne pas accéder aux données avec des intentions frauduleuses
		Un membre de l'agence peut manuellement modifier les données des utilisateurs	Un membre de l'agence modifie les données d'un utilisateur dans le but de commettre des actions frauduleuses	Falsification des données utilisateurs (nom, prénom, etc, mais aussi marquer un capteur comme malveillant ou l'inverse)	Elevé	Introduire un système d'historique des accès et modifications pour pouvoir traquer les suspects en cas de
	Serveur	Ressources	Pas de limite du nombre de requête par utilisateur	Possibilité d'attaque par déni de service (DDOS)	Déni de service : impossibilité d'accéder aux informations du serveur	Elevé

## VI/ Tests

Pour chaque utilisateur :

1) Faire des requêtes pour :

- Obtenir la moyenne de la qualité de l'air sur une position géographique donnée, à un moment donné et sur une période donnée
- Obtenir la moyenne de la qualité de l'air sur un rayon donné, à un moment donné et sur une période donnée
- Pouvoir identifier toutes les zones similaires à un capteur sélectionné
- Établir un classement des zones similaires à une certaine zone

⇒ Vérifier que les valeurs renvoyées sont correctes en utilisant des données réduites (pour gagner du temps dans un premier temps).

⇒ Vérifier que les résultats sont cohérents sur un set plus grand de données (pour être certain que tout fonctionne bien).

Pour un particulier :

1) S'authentifier

⇒ Vérifier que le système reconnaît bien l'utilisateur comme un particulier en vérifiant qu'il possède tous les droits qui lui sont accordés

2) Se déconnecter

⇒ Vérifier que l'utilisateur n'a accès qu'aux commandes communes

3) Voir ses points

⇒ Vérifier que l'utilisateur peut consulter ses points

⇒ Faire une recherche avec un capteur donné pour vérifier que les points de son particulier sont correctement incrémentés

4) Voir l'état de son(ses) capteur(s)

⇒ Vérifier que l'utilisateur puisse consulter l'état de l'ensemble de ses capteurs

⇒ Faire déclarer un capteur donné comme non fiable et vérifier que l'état donné par l'application est cohérent

Pour un fournisseur :

1) S'authentifier et être reconnu en tant que fournisseur

⇒ Vérifier que le système reconnaît bien l'utilisateur comme un fournisseur en vérifiant qu'il possède tous les droits qui lui sont accordés

2) Se déconnecter

⇒ Vérifier que l'utilisateur n'a accès qu'aux commandes communes

3) Voir l'impact de tous les cleaners, même s'ils ne lui appartiennent pas

⇒ Vérifier que les valeurs renvoyées sont correctes en utilisant des données réduites.

⇒ Vérifier que les résultats sont cohérents sur un set plus grand de données.

Pour un membre de l'agence :

- 1) S'authentifier et être reconnu en tant que membre de l'agence  
⇒ Vérifier que le système reconnaît bien l'utilisateur comme un membre de l'agence en vérifiant qu'il possède tous les droits qui lui sont accordés
- 2) Se déconnecter  
⇒ Vérifier que l'utilisateur n'a accès qu'aux commandes communes
- 3) Voir l'impact de tous les cleaners  
⇒ Vérifier que les valeurs renvoyées sont correctes en utilisant des données réduites.  
⇒ Vérifier que les résultats sont cohérents sur un set plus grand de données.
- 4) Voir les points de tous les particuliers  
⇒ Vérifier que l'utilisateur peut consulter l'ensemble des points des particuliers  
⇒ Faire une recherche avec un capteur donné pour vérifier que les points de son particulier sont correctement incrémentés
- 5) Voir les performances du serveur  
⇒ Vérifier que le temps d'exécution des algorithmes rentrent dans les critères établis (temps < 5 secondes)
- 6) Marquer des données d'un capteur comme fausses  
⇒ Déclarer un capteur donné comme non fiable et vérifier que l'état donné par l'application est cohérent  
⇒ Vérifier qu'un capteur donné déclaré non fiable est bien ajouté dans la liste noire

## VII/ Manuel d'utilisation

### I. Lancement de l'application

L'application peut prendre deux arguments : le nom d'utilisateur et le mot de passe, ou aucun argument. S'il n'y a pas d'arguments, l'application considère que l'utilisateur n'est pas authentifié. Sinon, elle vérifie quel type d'utilisateur (particulier, fournisseur ou membre de l'agence) le compte possède. Cette implémentation peu sécurisée sera utilisée dans un premier temps durant la phase de développement. On redirigera cette fonctionnalité vers une plateforme plus sécurisée dans le futur. On utilisera ainsi la syntaxe suivante:

`./AirWatcher` pour lancer l'application sans s'authentifier.

`./AirWatcher username password` pour lancer l'application en s'authentifiant.

### II. Choix disponibles dans les menus

Une fois lancée, l'application propose plusieurs commandes selon le type de compte de l'utilisateur. L'utilisateur pourra voir la liste des commandes en tapant `help`

#### II.1 Commandes communes

Tous les utilisateurs, dont les utilisateurs non-authentifiés, pourront choisir parmi les commandes suivantes, qui demanderont chacune un prompt, et donc ne nécessitent pas d'options/d'arguments de la part de l'utilisateur, sauf s'il s'agit d'un membre de l'agence souhaitant voir les performances de l'exécution d'une fonctionnalité (cf Commandes des membres de l'agence) (dans l'implémentation actuelle, nous n'avons pas d'autres options disponibles, d'autres pourront être ajoutées dans le futur). Il est aussi important d'ajouter qu'il est possible de retourner au menu principal si l'utilisateur ne souhaite plus effectuer une commande/prompt ou qu'il a tapé une commande par erreur en tapant `back`. De plus, avant chaque exécution d'une fonctionnalité, une confirmation (y/n) sera demandée à l'utilisateur. Cela pourra permettre de réduire les erreurs.

- `avgairqlty` → choix du type de requête parmi les 4 suivants (sélection avec index 1, 2, 3 et 4) :
  - Obtenir la moyenne de la qualité de l'air sur une position géographique donnée, à un moment donné (la console demandera l'utilisateur sur : des coordonnées (longitude et latitude), une date au format JJ/MM/AAAA HH:MM) :
  - Obtenir la moyenne de la qualité de l'air sur une position géographique donnée, sur une période donnée (la console demandera l'utilisateur sur : des coordonnées (longitude et latitude), une date de début et une date de fin au format JJ/MM/AAAA HH:MM)
  - Obtenir la moyenne de la qualité de l'air sur un rayon donné, à un moment donné (la console demandera l'utilisateur sur : des coordonnées (longitude et latitude), un rayon, une date au format JJ/MM/AAAA HH:MM)
  - Obtenir la moyenne de la qualité de l'air sur un rayon donné, sur une période donnée (la console demandera l'utilisateur sur : des

coordonnées (longitude et latitude), un rayon, une date de début et une date de fin au format JJ/MM/AAAA HH:MM)

- rank → Obtenir un classement des zones similaires à la zone où se situe un capteur qu'il aura sélectionné. (La console demandera à l'utilisateur de sélectionner un capteur à l'aide de son index (1 à n) parmi la liste qu'elle proposera)
- quit → Quitter l'application (termine le programme et retourne dans le terminal)
- connect → Se connecter si l'utilisateur n'est pas authentifié (La console demandera à l'utilisateur quel type de connexion il souhaite effectuer (Fournisseur, employé de l'agence ou particulier), puis il demandera le login et le mot de passe)
- disconnect → Se déconnecter si l'utilisateur est authentifié (Déconnecte l'utilisateur)

## II.2 Commandes des particuliers

Les particuliers auront accès aux fonctions suivantes, en plus des précédentes :

- mypoints → Voir ses points (commande d'affichage)
- mysensors → Voir l'état de son(ses) capteur(s) (commande d'affichage)

## II.3 Commandes des fournisseurs

Les fournisseurs auront accès aux fonctions suivantes, en plus des commandes communes :

- cleaner → Voir l'impact d'un cleaner, même de ceux qui ne leur appartiennent pas (la console demandera l'utilisateur sur : un capteur parmi la liste que l'utilisateur sélectionnera à l'aide de l'index du capteur (1 à n)).

## II.4 Commandes des membres de l'agence

Les membres de l'agence auront accès aux fonctions suivantes, en plus des commandes communes :

- cleaner → Voir l'impact d'un cleaner (la console demandera l'utilisateur sur : un capteur parmi la liste de capteurs que l'utilisateur sélectionnera à l'aide de l'index du capteur (1 à n)). Il s'agit de la même commande que celle des fournisseurs.
- points → Voir les points d'un particulier (la console demandera l'utilisateur sur : un utilisateur parmi la liste de particuliers que l'utilisateur sélectionnera à l'aide de l'index du capteur (1 à n)). Il s'agit d'une commande différente que celle des particuliers.
- Voir les performances de l'exécution d'un algorithme en ajoutant -perf en option lorsqu'il entre une commande
- addtoblocklist → Marquer des données d'un capteur comme fausses manuellement (la console demandera l'utilisateur sur : un capteur parmi la liste des capteurs de particuliers que l'utilisateur sélectionnera à l'aide de l'index du capteur (1 à n)).
- blacklist → Voir la liste noire des capteurs (commande d'affichage)