

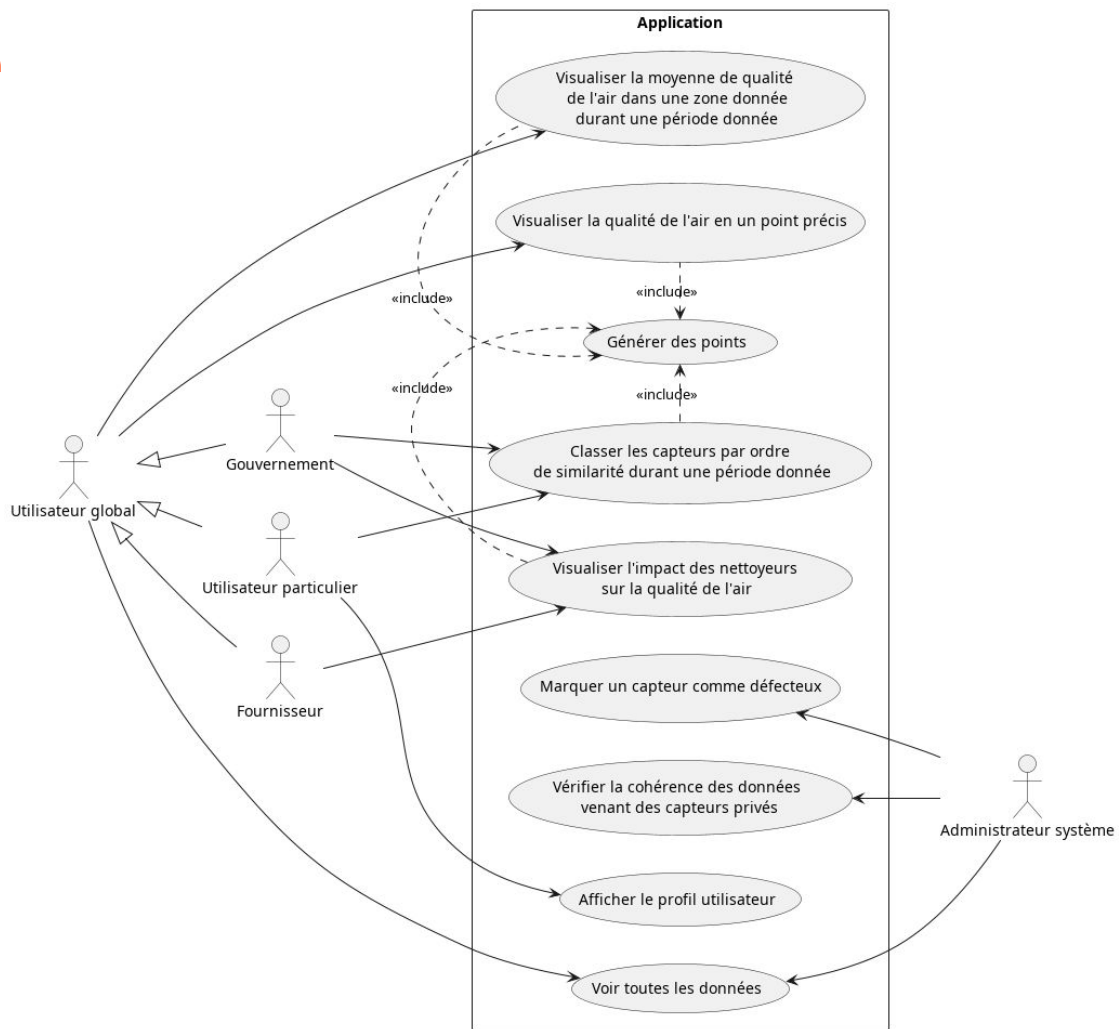
Présentation du projet Airwatcher



Contexte du projet

- Objectif : permettre au gouvernement de surveiller la qualité de l'air
- **Spécification** et **conception** du projet
- Développement d'une partie de l'application :
 - en respectant un cahier des charges
 - en utilisant un dataset existant

Use case



Des exigences fonctionnelles...

*Visualiser la moyenne de
qualité de l'air dans une
zone donnée durant une
période donnée*

- **Inputs** : latitude, longitude, rayon, date de début et de fin
- **Source** : base de données des capteurs et mesures
- **Output** : indice de qualité de l'air

- **Inputs** : l'ID du capteur, date de début et de fin
- **Source** : base de données des mesures
- **Output** : classement des capteurs par ordre de similarité

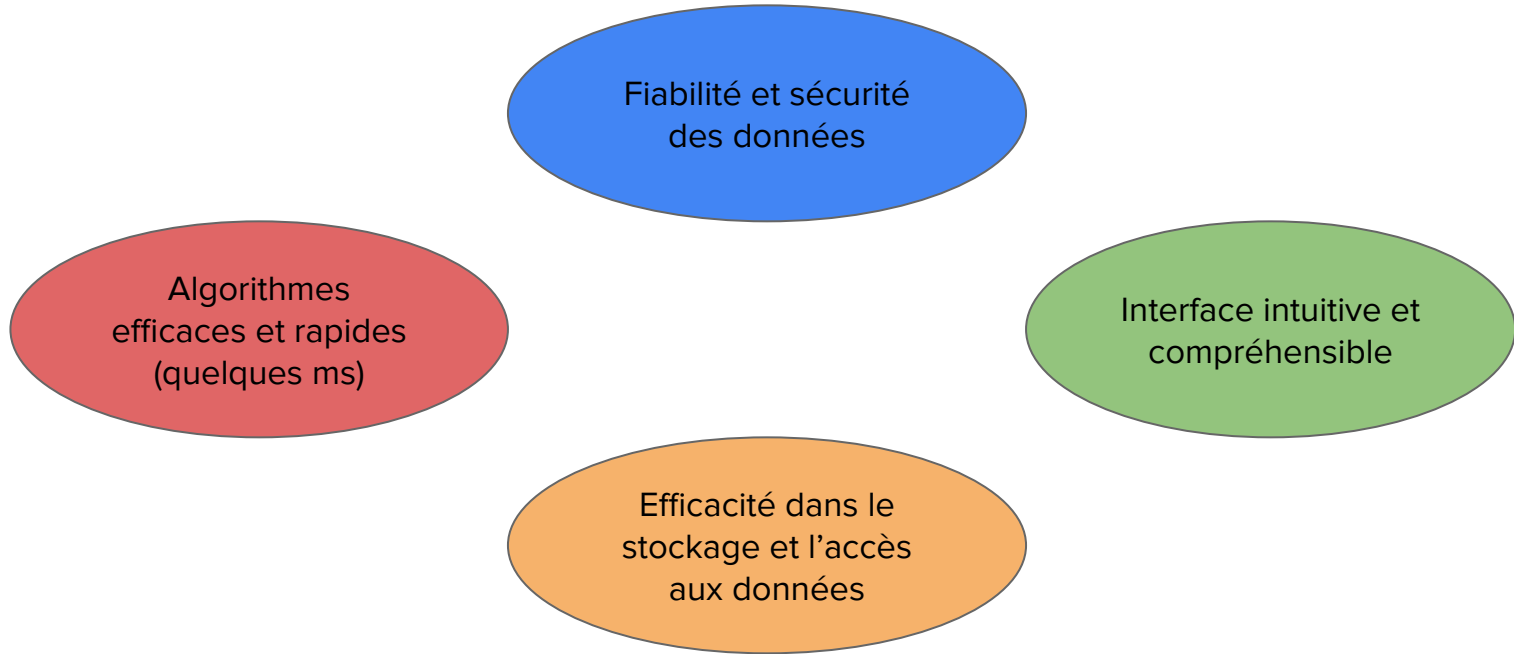
*Classer les capteurs par
ordre de similarité durant
une période donnée*

Des exigences fonctionnelles...

Indice ATMO ^{25, 23}	O ₃	SO ₂	NO ₂ ⁻	PM ₁₀	Niveau
1	0 à 29	0 à 39	0 à 29	0 à 6	Très bon
2	30 à 54	40 à 79	30 à 54	7 à 13	Très bon
3	55 à 79	80 à 119	55 à 84	14 à 20	Bon
4	80 à 104	120 à 159	85 à 109	21 à 27	Bon
5	105 à 129	160 à 199	110 à 134	28 à 34	Moyen
6	130 à 149	200 à 249	135 à 164	35 à 41	Médiocre
7	150 à 179	250 à 299	165 à 199	42 à 49	Médiocre
8	180 à 209	300 à 399	200 à 274	50 à 64	Mauvais
9	210 à 239	400 à 499	275 à 399	65 à 79	Mauvais
10	≥ 240	≥ 500	≥ 400	≥ 80	Très mauvais

→ Source : Wikipedia

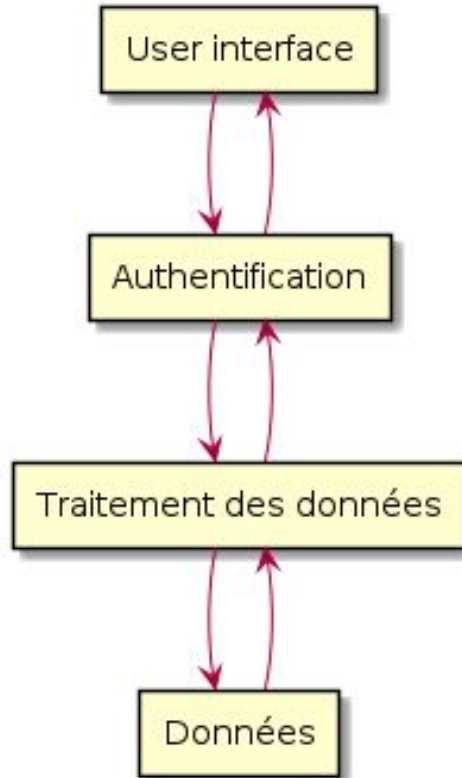
... et non-fonctionnelles



Risques de sécurité

Atout	Vulnérabilité	Attaque	Risque	Impact	Contre-mesure
Stockage des données (mesures des capteurs, données utilisateur, ...)	> Stockage sans chiffrement	> Déchiffrement et obtention des données stockées	> Accès aux mesures de qualité de l'air > Corruption des données (insertion de fausses données, ...) > Accès aux données des utilisateurs > Corruption des points obtenus par l'utilisateur > Accès à la localisation des capteurs, privés ou publics	> Elevé > Elevé > Elevé > Moyen > Elevé	> Chiffrement du stockage
Transmission des données (mesures des capteurs, données utilisateur)	> Transmission sans chiffrement	> Interception de la communication	> Accès aux mesures de qualité de l'air > Accès aux données des utilisateurs > Accès à la localisation des capteurs, privés ou publics	> Elevé > Elevé > Elevé	> Chiffrement de la transmission
Consultation des données sur l'application <i>AirWatcher</i>	> Authentification faible (nom d'utilisateur, mot de passe, ...)	> Authentification devinée par l'attaquant	> Accès aux mesures de qualité de l'air > Corruption des données (insertion de fausses données, ...) > Accès aux données des utilisateurs > Corruption des points obtenus par l'utilisateur > Accès à la localisation des capteurs, privés ou publics	> Elevé > Elevé > Elevé > Moyen > Elevé	> Vérifications de la robustesse de l'authentification (mot de passe)

Conception : architecture

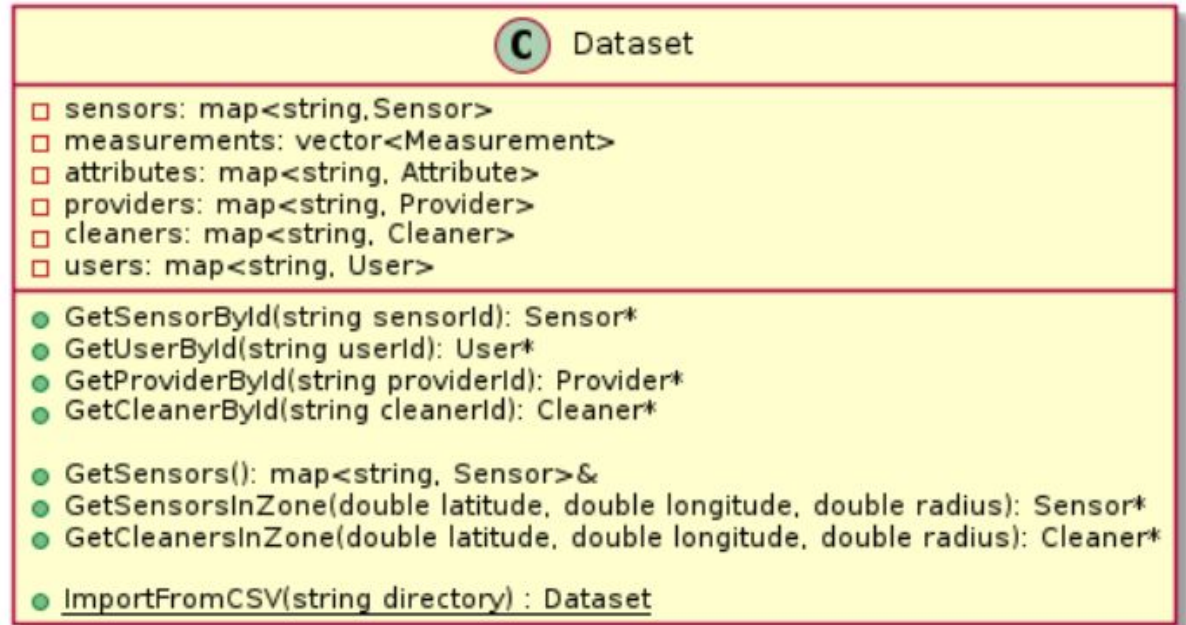
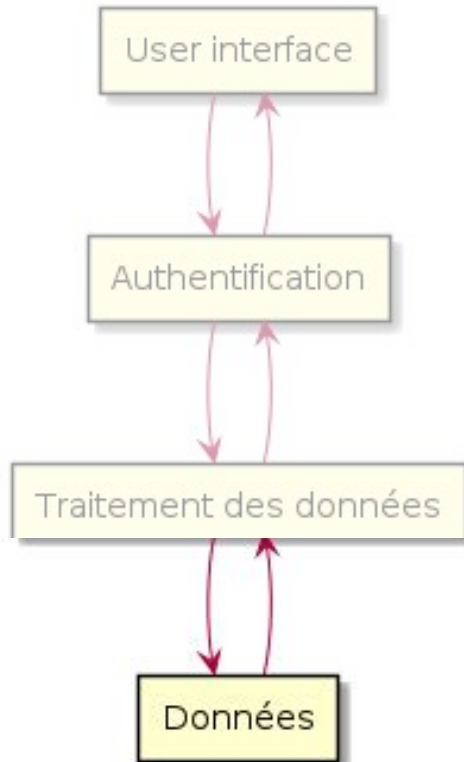


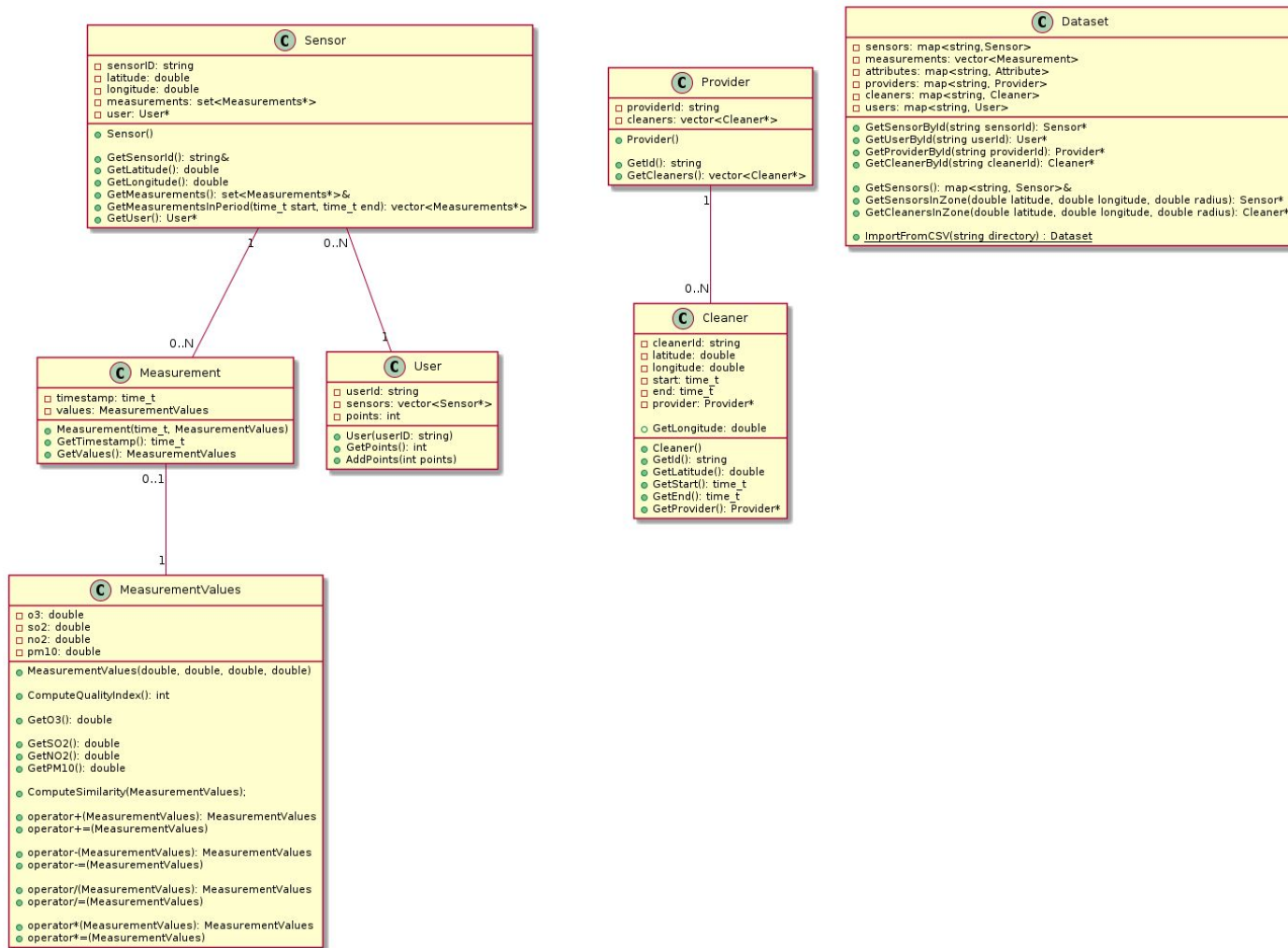
- La couche d'interface utilisateur qui devra afficher les résultats à la console et gérer les entrées
- La couche d'authentification qui décidera quelles fonctionnalités montrer à l'utilisateur courant
- La couche de traitement de données qui sera chargé de faire les calculs demandés
- La couche de données qui permettra de stocker efficacement les données

Conception : architecture

Avantages	<ul style="list-style-type: none">→ Modularité: Chaque couche peut-être isolée et testée indépendamment.→ Abstraction: Interface simplifiée et cohérente pour les couches supérieures.
Inconvénients	<ul style="list-style-type: none">→ Maintenance: Les couches rendent une application plus difficile à maintenir. Chaque changement nécessite une analyse.→ Performance: Surcharge liée à la communication entre couches supérieures et inférieures.

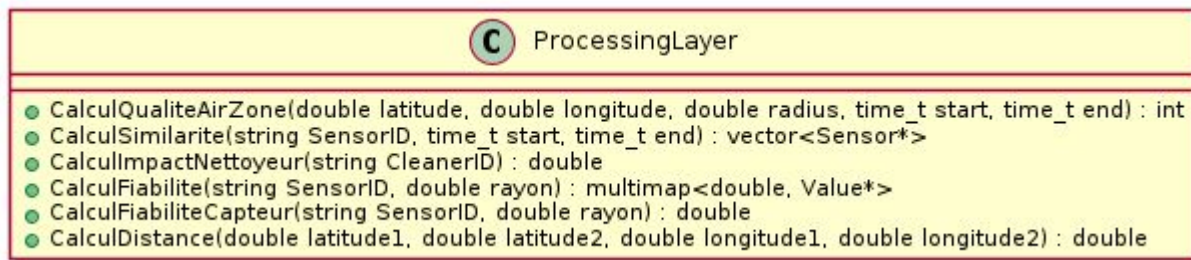
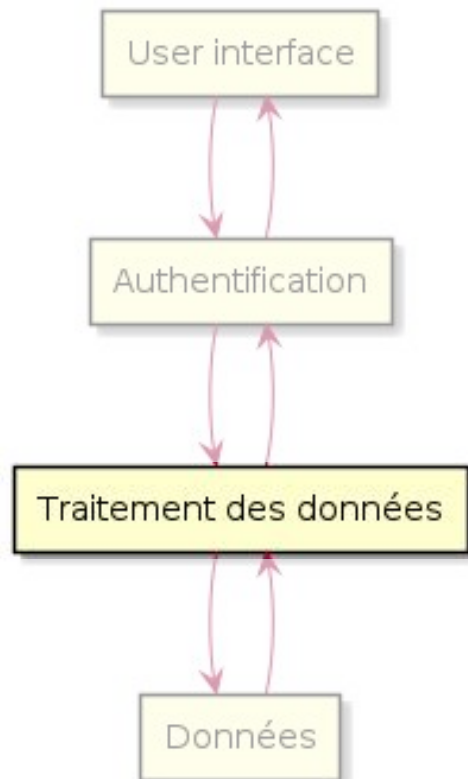
Conception : classes



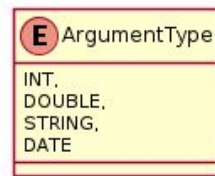
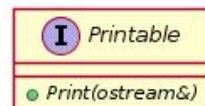
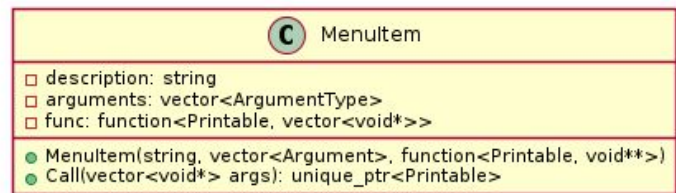
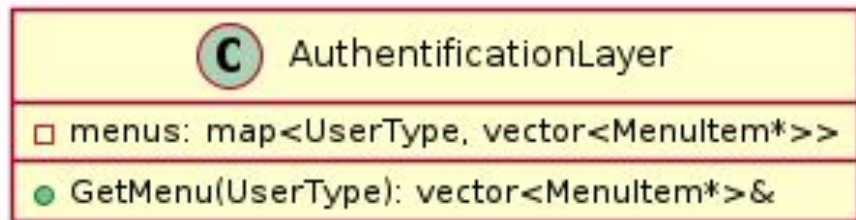
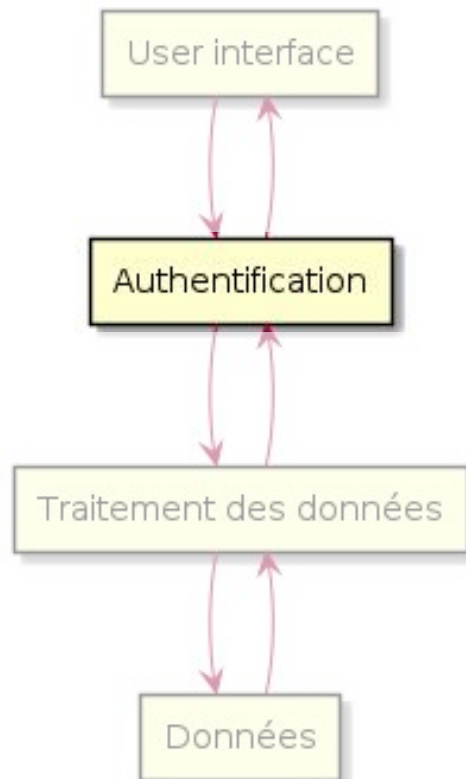


Conception: classes

Conception : classes



Conception : classes



Conception : scénarios principaux

- 3 scénarios principaux
 - Calcul de la qualité de l'air dans une zone
 - Récupération des capteurs similaires
 - Vérification de l'impact des nettoyeurs (non développé)

Scénario 1 : calcul de qualité de l'air dans une zone

→ Input :

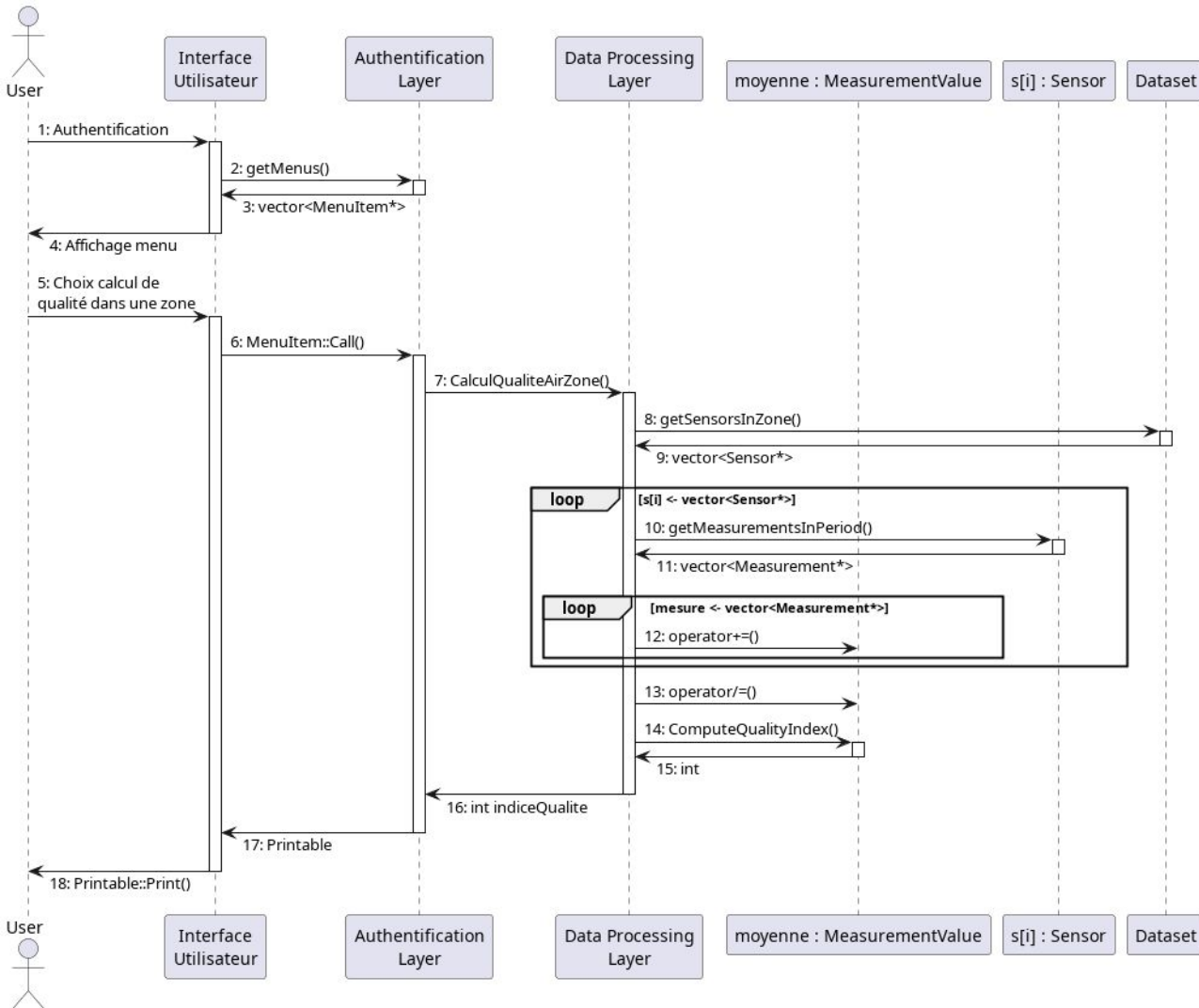
Zone circulaire (centre + rayon), période de temps (dates de début et de fin)

→ Output :

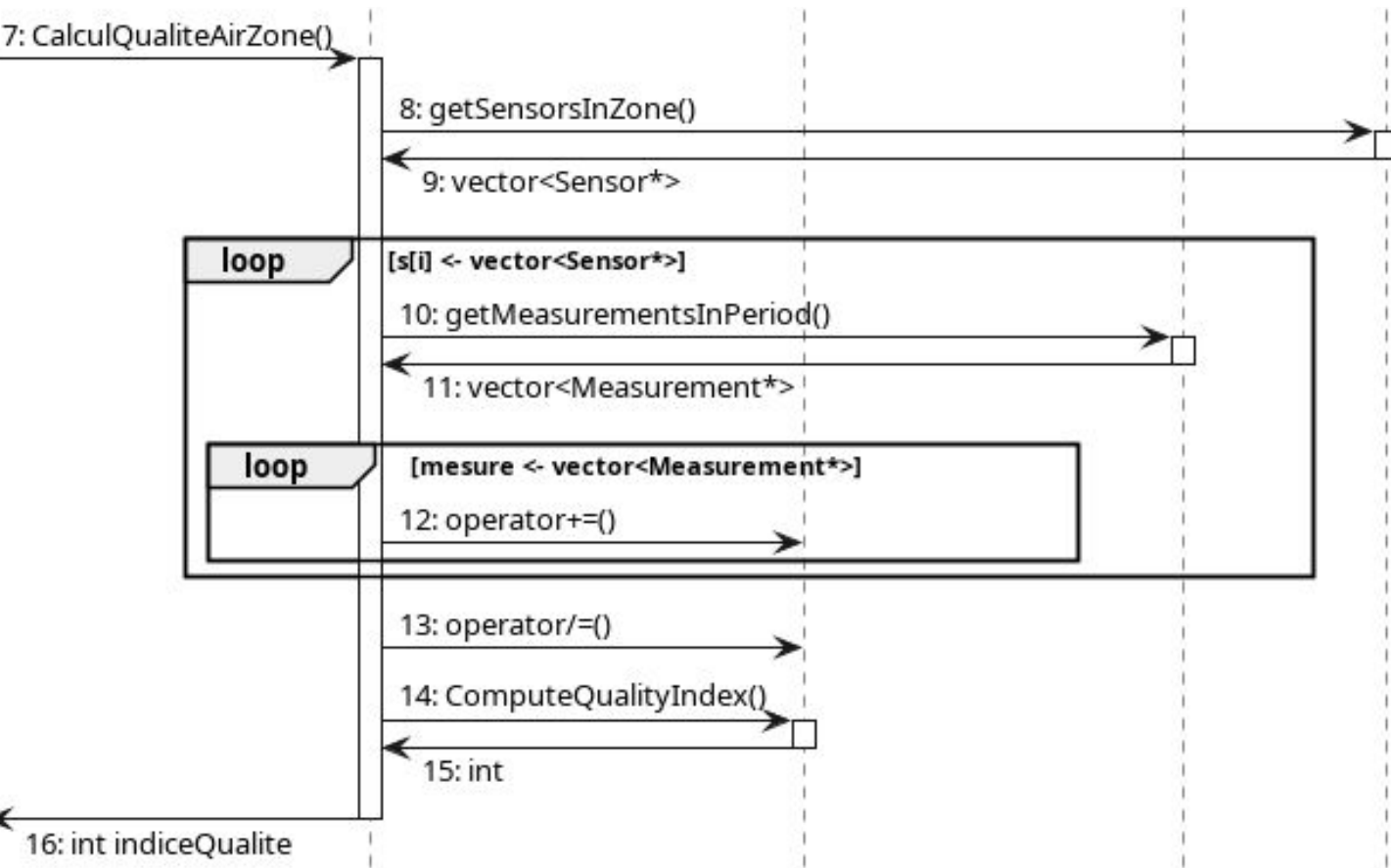
Indice ATMO correspondant (Entier)

→ Calcul :

- 1) Moyenne des mesures de tous les capteurs de la zone
- 2) Tableau des indices ATMO



Scénario 1 : calcul de qualité de l'air dans une zone



**Scénario 1
: calcul de
qualité de
l'air dans
une zone**

Scénario 2 : classement des capteurs similaires

→ Input :

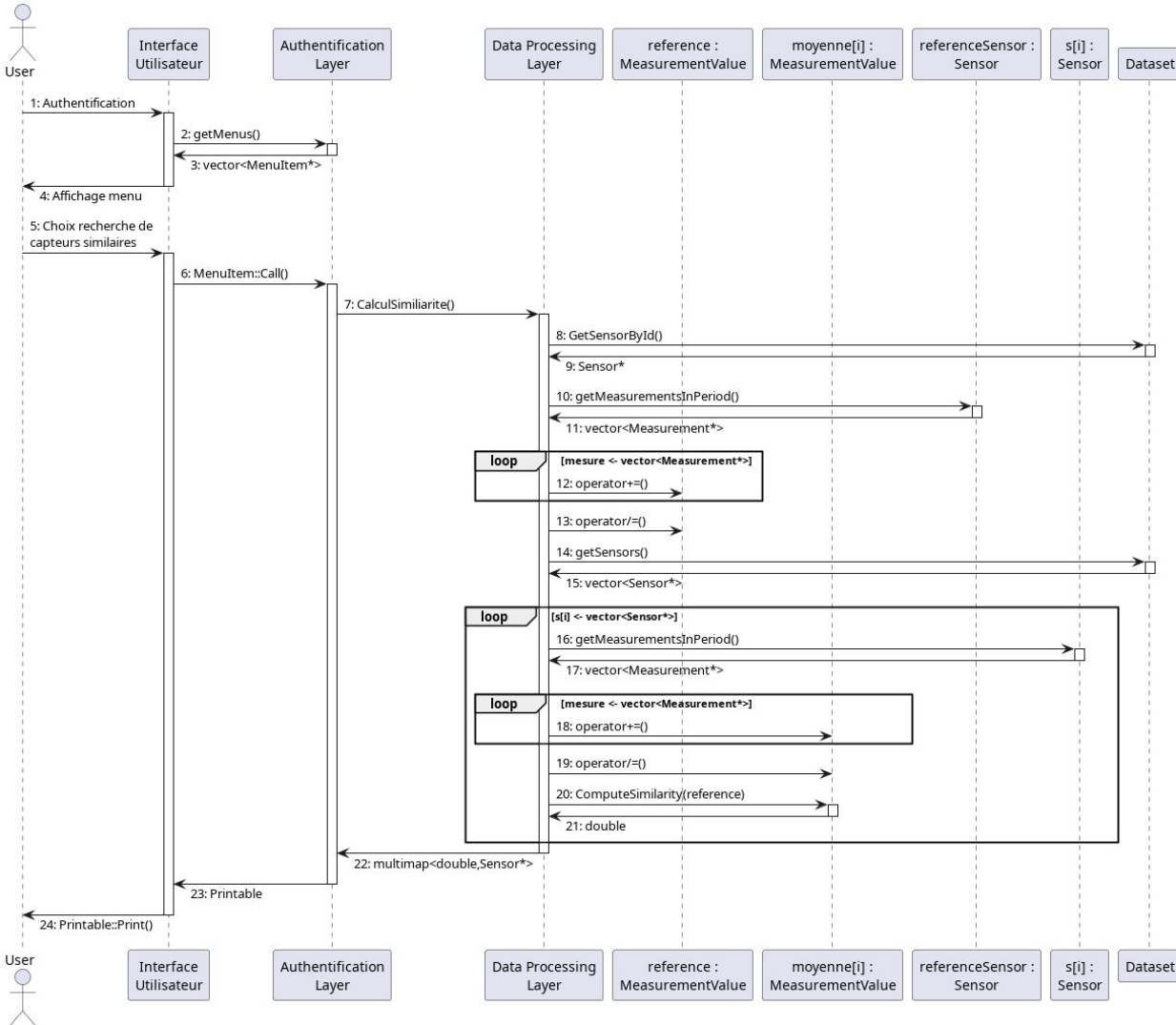
Capteur (SensorID), période de temps (dates de début et de fin)

→ Output :

Classement des capteurs similaires (Multimap <double, Sensor*>)

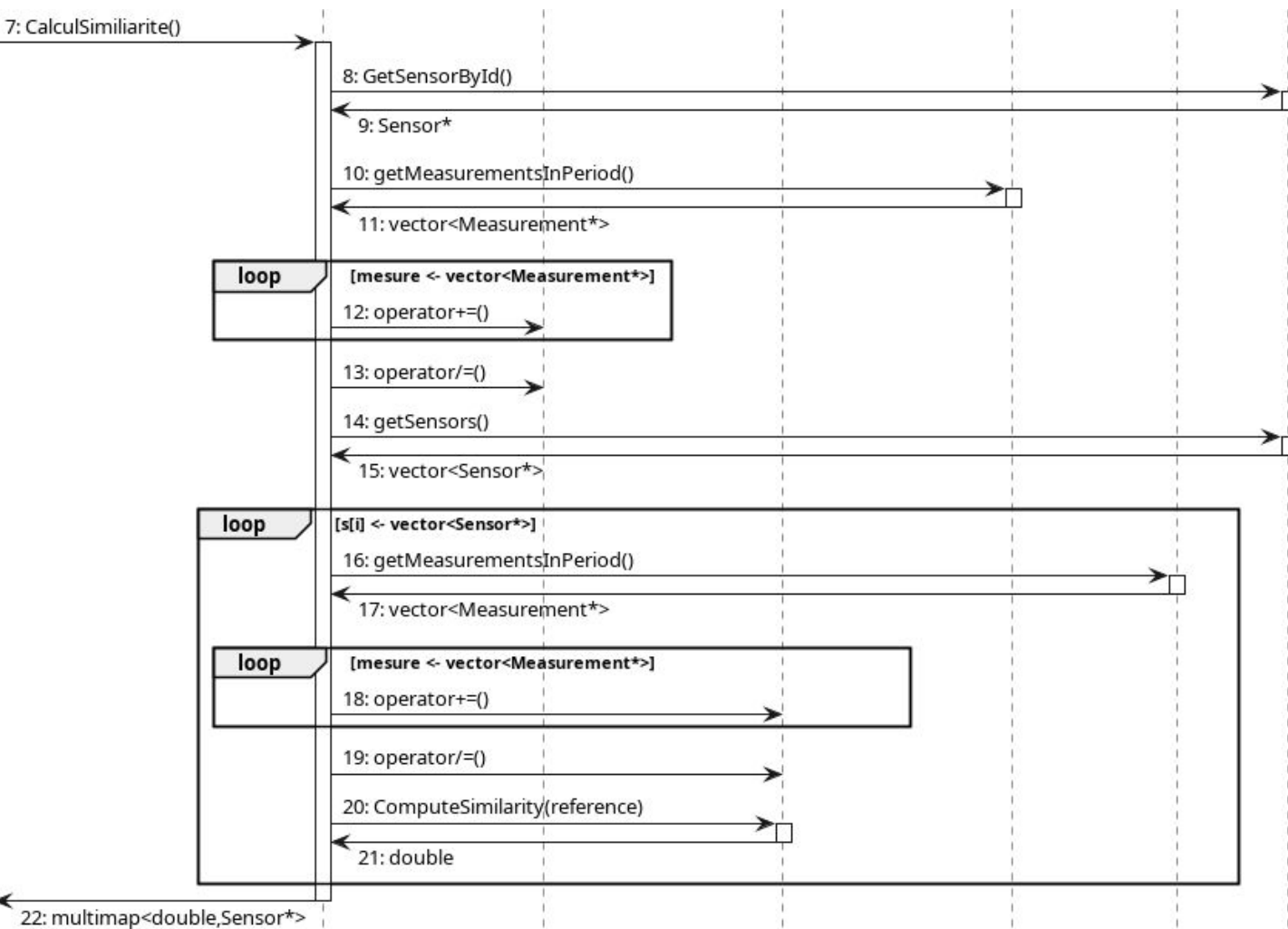
→ Calcul :

- 1) Pour le capteur de référence : moyenne des mesures**
- 2) Pour chaque capteur : moyenne des mesures**
- 3) Remplissage Multimap avec calcul de norme**



Scénario 2 : classement des capteurs similaires

7: CalculSimilarite()



**Scénario 2 :
classement
des
capteurs
similaires**

Scénario 3 : quantification impact nettoyeur

→ Input :

Nettoyeur (CleanerID)

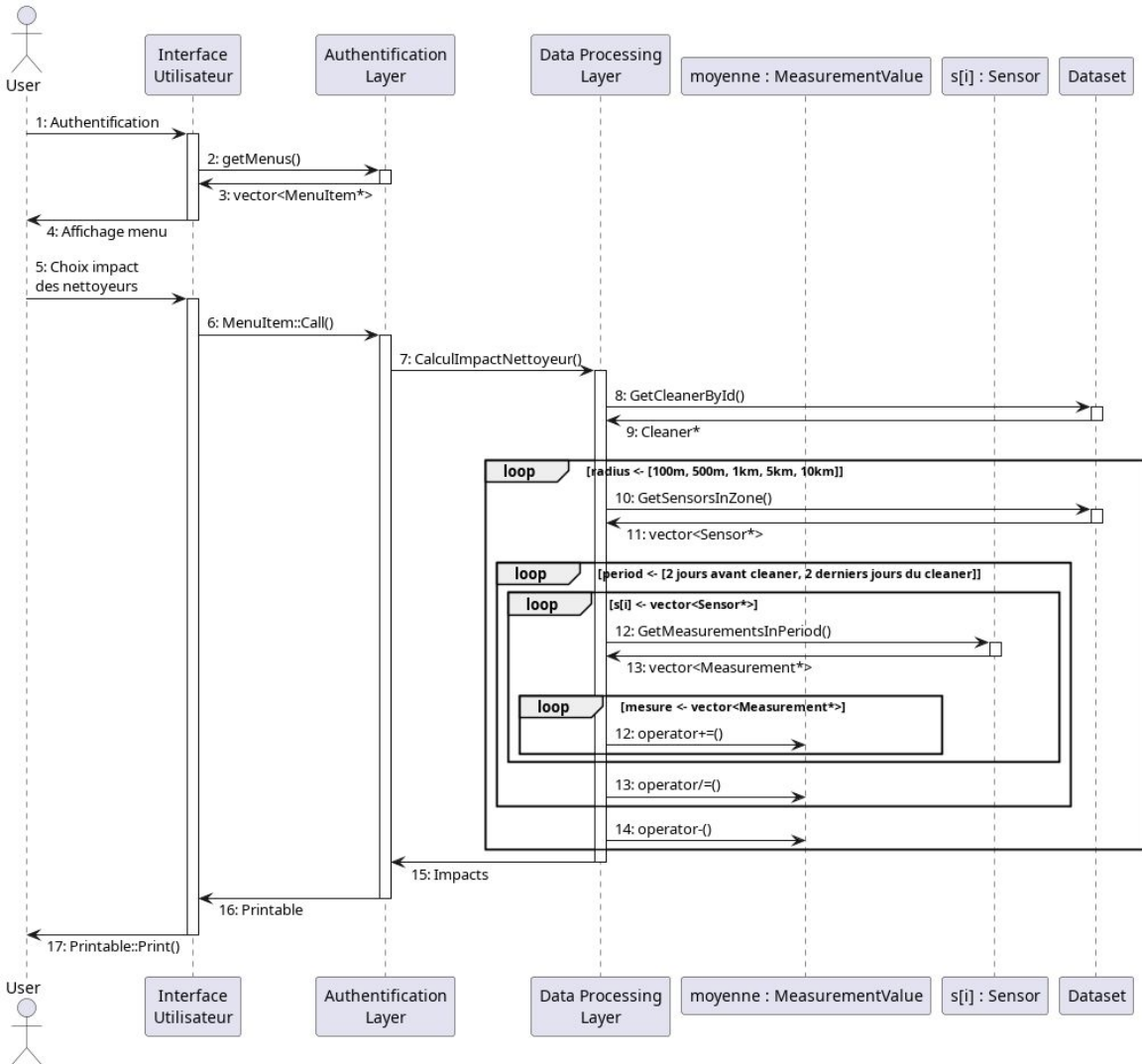
→ Output :

Quantification impact nettoyeur (100m, 500m, 1km, 5km, 10km)

→ vector <MeasurementsValue*>

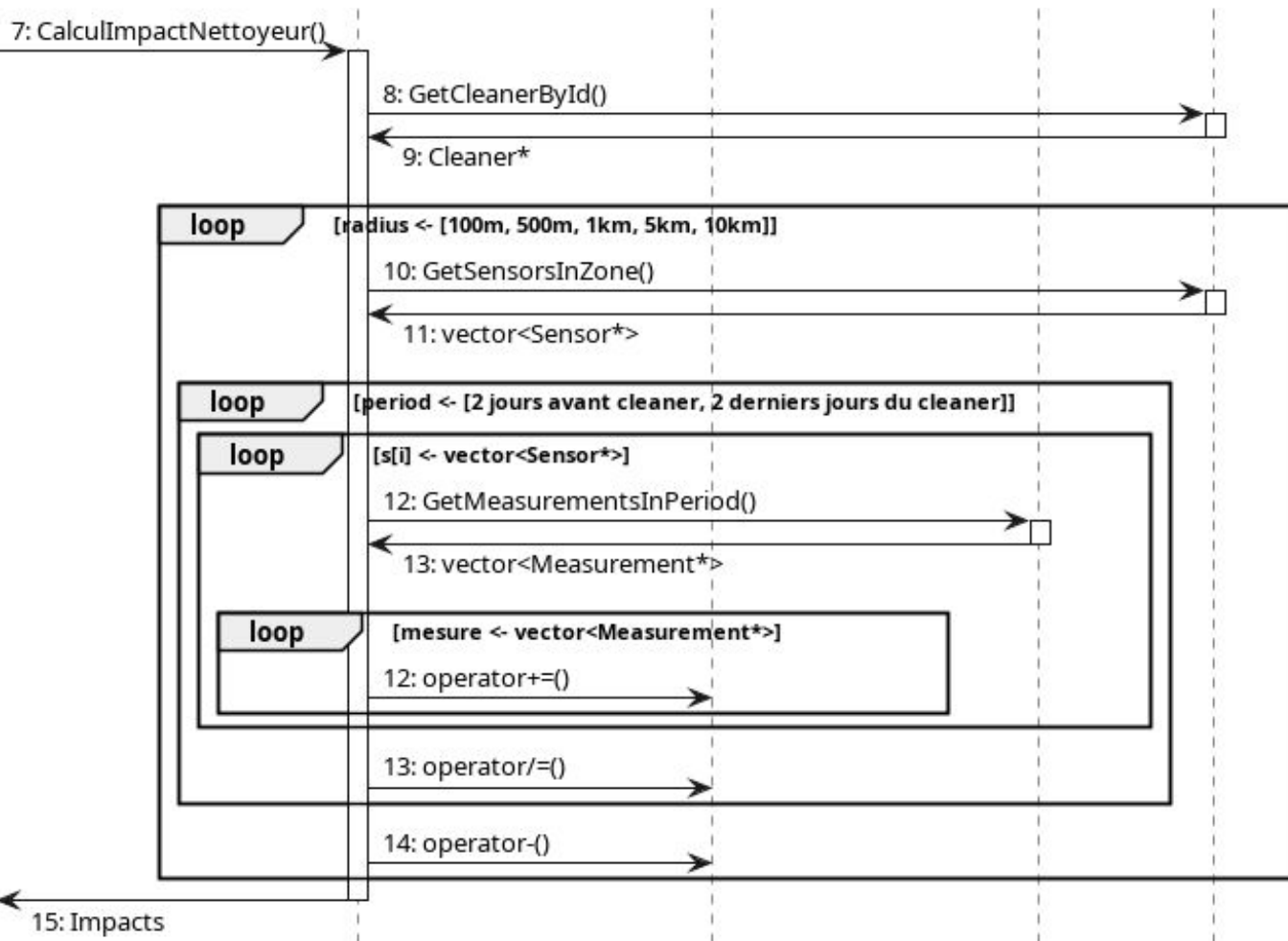
→ Calcul :

- 1) Sur la période de 2j avant l'installation : moyenne des mesures**
- 2) Sur la période de 2j avant désinstallation : moyenne des mesures**
- 3) Différence des mesures, pour les différents rayons considérés**



Scénario 3 : quantification impact nettoyeur

7: CalculImpactNettoyeur()



**Scénario 3 :
quantification
impact
nettoyeur**

Live demo !

