

Dossier commun du projet

Sommaire

[I. Cahier des charges 3](#_Toc506900197)

[1.1) Récapitulatif du cahier des charges 3](#_Toc506900198)

[1.2) Expression du besoin 3](#_Toc506900199)

[1.3) Contraintes imposées par le client 3](#_Toc506900200)

[1.4) Matériel fourni par le client 3](#_Toc506900201)

[1.5) Logiciels fournis par le client 3](#_Toc506900202)

[1.6) Synoptique de la réalisation 3](#_Toc506900203)

[1.7) Répartition des tâches 3](#_Toc506900204)

[II. Spécifications techniques 4](#_Toc506900205)

[2.1) Étudiant 1 : Spécification des blocs Raspberry/Anémomètre 4](#_Toc506900206)

[2.2) Étudiant 2 : Spécification des blocs Web/Solarimètre 4](#_Toc506900207)

[2.3) Étudiant 3 : Spécification des blocs Base de données/Sonde température 4](#_Toc506900208)

[2.4) Étudiant 4 : Spécification des blocs Android/Pluviomètre 4](#_Toc506900209)

[III. Analyse 5](#_Toc506900210)

[3.1) Étudiant 1 : Analyse des blocs Raspberry/Anémomètre 5](#_Toc506900211)

[3.2) Étudiant 2 : Analyse des blocs Web/Solarimètre 5](#_Toc506900212)

[3.3) Étudiant 3 : Analyse des blocs Base de données/Sonde température 5](#_Toc506900213)

[3.4) Étudiant 4 : Analyse des blocs Android/Pluviomètre 5](#_Toc506900214)

[IV. Intégration et versionning 6](#_Toc506900215)

[V. Communication de groupe 7](#_Toc506900216)

[VI. Conclusion 8](#_Toc506900217)

[VII. Annexes 9](#_Toc506900218)

[7.1) Diagramme de Gantt 9](#_Toc506900219)

[7.2) Procédure d’installation 9](#_Toc506900220)

[VIII. Glossaire 10](#_Toc506900221)

# Cahier des charges

## Récapitulatif du cahier des charges

**Les bienfaits d’une serre :**

Une serre, même petite, ouvre de vastes possibilités de culture, des primeurs aux orchidées. Son équipement peut comprendre des dispositifs automatiques pour assurer les opérations les plus astreignantes : **chauffage**, **arrosage**, **aération**.

Une serre libère le jardinier des caprices des intempéries. A l'abri du verre, il peut cultiver des plantes qui, dehors, souffriraient du froid, des pluies et des vents violents.

Les serres peuvent être chauffées ou non :

• Les serres froides

Une serre froide, non chauffée, ne protège pas du gel, mais elle entoure les plantes d'une atmosphère plus tiède en période de végétation, ce qui hâte le mûrissement des fruits et améliore la floraison de nombreuses espèces décoratives. Elle protège les plantes des intempéries, d'une humidité excessive en hiver et des attaques des oiseaux, des insectes et autres animaux nuisibles.

Mais sa fonction la plus précieuse est d'allonger la période de végétation. On peut y mettre les plantes en culture en début de printemps et soit les conserver en serre, soit les planter dans le jardin. En serre, les plantes continuent de prospérer durant l'automne.

Cette prolongation artificielle de la période de végétation est causée, au printemps, par la chaleur du soleil captée par le verre et à l'automne, par la conservation de la chaleur accumulée pendant l'été dans le sol.

• Les serres chaudes

Dans une serre chauffée artificiellement, la gamme des végétaux cultivables augmente de façon considérable, car les plantes molles, qui ne résistent pas aux hivers rigoureux, s'y conservent admirablement.

Des dispositifs automatiques de chauffage, de ventilation, d'ombrage et arrosage rendent presque totale la maîtrise des conditions climatiques intérieures. De plus, toutes les serres sont équipées d'aérateurs qui permettent le renouvellement de l'atmosphère de la serre. Il existe également des ventilateurs électriques. Enfin, par les jours très chauds d'été, l'aération ne suffit pas. Pour prévenir une montée de la température préjudiciable aux plantes, l'ombrage sera assuré par des stores intérieurs que l'on baisse lorsque le soleil est trop fort et que l'on remonte lorsqu'il faiblit. Il existe des stores automatiques extérieurs, mais ils sont coûteux.

Une serre demande donc une **température**, une **hydrométrie**, une **intensité lumineuse** une anémométrie contrôlée.

**Le commanditaire du projet :**

Le Groupe Olivier est spécialisé dans la production de tomates et de concombres sous serres verre chauffées. Il regroupe 2 sociétés de production de légumes, la SCA Olivier Frères et la SCA des Courtines.

L’exploitation d’une surface de 18,5 ha de serres est répartie sur 2 sites de production dans la région nantaise à Haute-Goulaine et Saint-Julien-de-Concelles.

Dirigé par Henri, Jean-Luc et Stéphane Olivier, le Groupe emploie environ 170 équivalents temps plein. Les productions de légumes en culture hors sol sont commercialisées par la coopérative OCEANE, crée en 1993 par 9 Maraîchers Nantais, dont la famille Olivier.

Pour plus d’informations sur l’activité du Groupe Olivier : <http://www.groupe-olivier.fr/>

**Aperçu du projet :**

La globalité de ce projet aura pour objet la gestion automatique d'une serre maraîchère et se décomposera en deux parties :

1. La **supervision** de l'état de la serre avec récupération et stockage de l'ensemble des données nécessaires ;

2. L'**automatisation** de la régulation de la température, l'hydrométrie et de l'intensité lumineuse de la serre.

Il a été décidé que ce projet sera porté sur deux années. Le projet décrit par la suite se limite donc à la première partie : **La supervision de l'état de la serre**.

Ce projet aura donc pour but de superviser l’état d’une serre :

- L’acquisition et le stockage de l’ensemble des données suivantes :

* L’intensité lumineuse
* La pluviométrie
* La force et la direction du vent
* La température intérieure de la serre et de l’eau des tuyaux de chauffage

- La mise en place de systèmes pour visualiser les différentes données acquises :

* Un système (Raspberry) récupèrera automatiquement les différentes mesures énoncées ci-dessus.
* Un second système (BDD) va permettre la mémorisation de ces mesures sur plusieurs années.
* Possibilité à l’aide d’une application Web de consulter les mesures par des histogrammes et de visualiser en temps réel.
* Information en temps réel de l’état de fonctionnement de l’ensemble du système grâce à une application Android.

**Finalité du projet :**

Le client souhaite :

- un système situé dans la serre permettant la récupération automatique des différentes mesures nécessaires au projet global ;

- un système situé dans un local proche de la serre permettant la mémorisation des mesures sur une période de plusieurs années ;

- une application web permettant la consultation des mesures par des histogrammes ;

- une application web permettant la visualisation en temps réel des différentes mesures effectuée dans la serre ;

- une application Android permettant d’être informé en temps réel de l'état de fonctionnement de l'ensemble du système.

**Mesures à effectuer**

Le système devra enregistrer six mesures distinctes.

1. L'intensité lumineuse devra être mesurée par le solarimètre SDE-SOLAR-10000W de la société ARIA. Celui-ci dispose d'une plage allant de 0 à 1000 W/M2. Ce capteur fonctionne en boucle de courant de 4-20 mA.
2. La température intérieure sous serre devra être mesurée par un capteur SDTS ARIA de plage 0 à 45°C. Ce capteur fonctionne également en boucle de courant de 4-20 mA.
3. La température d'eau chaude des tuyaux de chauffage devra être mesurée par un capteur SDTS ARIA de plage 0 à 100°C. Ce capteur fonctionne également en boucle de courant de 4-20mA.
4. L'hydrométrie sera mesurée par le pluviomètre Rain Collector II de la société DAVIS. Ce capteur fonctionne par impulsion.
5. La direction et la force du vent seront mesurée par l'anémomètre-girouette 7911 de la société DAVIS.

## Expression du besoin

L'exploitant de la serre est l'acteur principal du système.

L'exploitant doit pouvoir visualiser en temps réel l'état de la serre.

L'exploitant doit pouvoir consulter facilement l’évolution de chaque mesure, en choisissant la période de visualisation (jour, semaine, mois ou année).

L'exploitant doit pouvoir être informé en temps réel sur son smartphone de l'état de fonctionnement du système.

## Contraintes imposées par le client

**Contraintes financières (budget alloué) :**

Le budget alloué est de 100 euros permettant l'achat éventuel de carte(s) Adaptateur 4-20 mA. Le reste du matériel nécessaire au projet est disponible dans la section.

**Contraintes de développement (matériel et/ou logiciel imposé / technologies utilisées) :**

* Carte de gestion : développement sur carte Raspberry en langage python
* Développement de l'application Android avec Android Studio
* Développement de l'application Web sous le patron Modèle-Vue-Contrôleur (éventuellement sous le framework Symfony)

**Contraintes qualité :**

* La partie logicielle devra être intégralement testée avant sa présentation au commanditaire en fin de projet. Le client final devra pouvoir utiliser le système sans compétence informatique particulière (hormis éventuellement pour certains paramétrages de configuration).
* Le projet devra être réalisé afin de pouvoir ajouter facilement de nouvelles mesures par la suite. Par exemple, si le client désire ajouter une mesure pH du sol, le travail à réaliser devra être minimal, voire automatique si possible : la mise à jour de l'interface web de visualisation devra être aisée, la visualisation de l'évolution pourra être mise à jour automatiquement. **Ceci est une contrainte d'évolutivité forte impliquant une analyse poussée et un travail de développement plus exigeant. Cette contrainte sera notée (\*ContE) et précisera, dans le tableau de répartition des tâches, les tâches pour lesquels cette contrainte devra être prise en compte.**
* Une documentation complète sur le système proposé devra être fournie au client. Elle doit comprendre : un dossier de présentation du système, un mode d’emploi de l’interface, une procédure illustrée d’installation... Les étudiants devront aussi fournir un exemplaire des sources de leur travail, ainsi qu’une nomenclature précise du matériel utilisé pour permettre aux bénéficiaires de donner une suite au projet, par une autre voie que l’équipe d’étudiants.
* Une documentation automatique du code devra être fourni au format html.

**Contraintes de fiabilité, sécurité :**

* La carte de gestion devra journaliser en interne tout problème d'acquisition de mesures ou de problème d'accès à la base de données.

## Matériel à disposition des étudiants

Ressources matérielles mises à disposition des étudiants :

- Solarimètre SDE-SOLAR-10000W

- Deux capteurs SDTS ARIA (0 à 45°C et 0 à 100°C)

- Un pluviomètre DAVIS

- Un anémomètre/girouette DAVIS

- Carte de mesure : module boucle 4-20 mA...

- Carte de gestion : carte raspberry Pi 3, arduino…

- 4 PC Windows 7

*Remarque : le smartphone android n’est pas fourni, ce sera donc celui de l’étudiant qui sera utilisé. Une tablette pourra être prétée si besoin.*

## Logiciels à disposition

**Ressources logicielles pour le développement**

- Environnement de développement pour Python 2.7

- Environnement de développement Android Studio

- Environnement MySQL pour la base de données

- Environnement Netbeans pour l'application Web, éventuellement le framework Symfony

**Autres ressources logicielles disponibles durant le projet**

- Suites bureautiques Microsoft Office 2007 et LibreOffice 5

- MagicDraw 17.0.3 avec plugin SysML

- Microsoft Project

- Gestionnaire de version Git

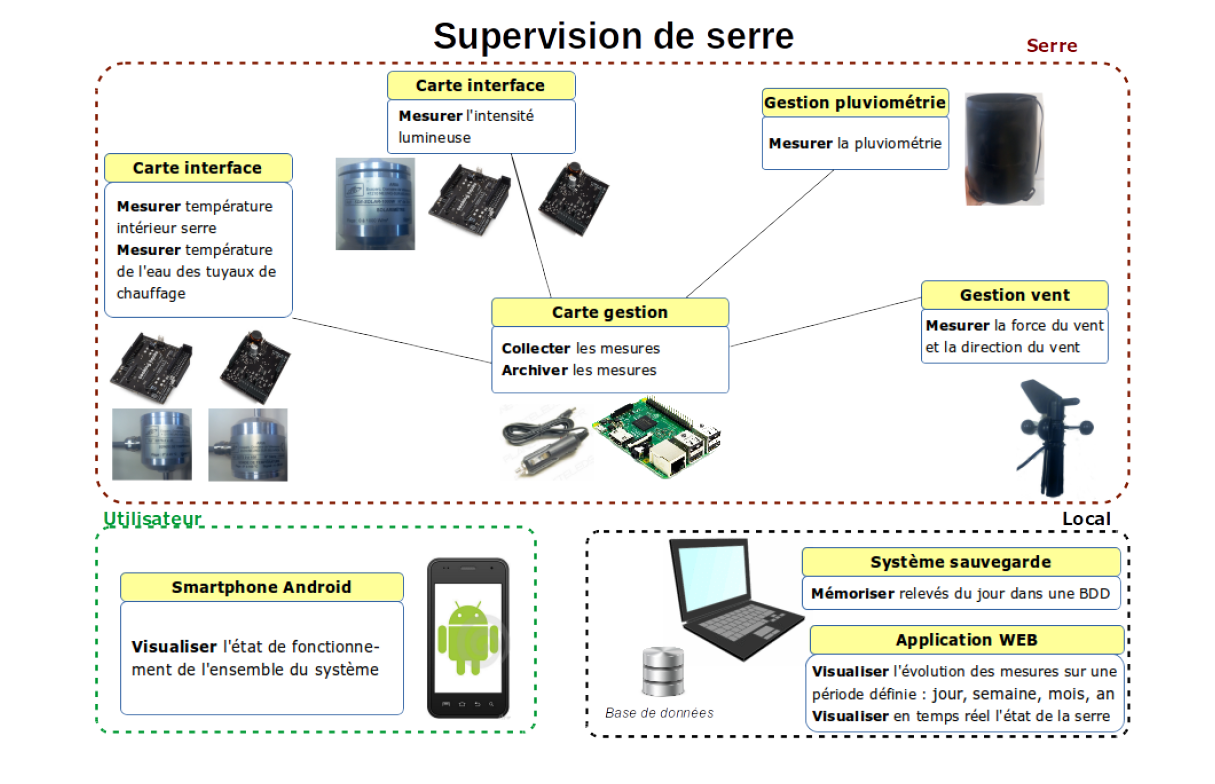
## Synoptique – Fonctions du système et échange d’informations

Le principe de fonctionnement du système sont résumés dans le synoptique de la page suivante.

Dans la serre, un système effectue une prise des mesures à intervalle régulier. Ces informations sont enregistrées dans une base de données distante. Le type de liaison sera à déterminer par les étudiants et devra correspondre aux spécificités du site.

Dans le local proche de la serre, se trouvent le PC d’archivage. La visualisation des informations récoltées est effectuée sur le PC à l’aide d’une application web qui permet d’afficher les courbes dévolution sur une période définie, pouvant aller jusqu’à un an. Il est également possible de visualiser l'état actuel de la serre depuis l'application web.

L’utilisateur dispose aussi d’un moyen de contrôle du système, une application Android. Celle-ci lui permet de visualiser le bon fonctionnement de chaque entité du système.



## Répartition des tâches

Pour la réalisation du projet, les tâches ont été réparties de la façon suivante :

* **Etudiant 1 : Steven NGUENE**, partie Raspberry/Anémomètre
* **Etudiant 2 : Willy RINEAU**, partie Web/Réseau/Solarimètre
* **Etudiant 3 : Samuel GERARD**, partie BDD/Web/Sonde température
* **Etudiant 4 : Dylan CHESNOUARD**, partie Android/Pluviomètre

# Spécifications techniques

## 2.1) Étudiant 1 : Spécification des blocs Raspberry/Anémomètre

## 2.2) Étudiant 2 : Spécification des blocs Web/Solarimètre

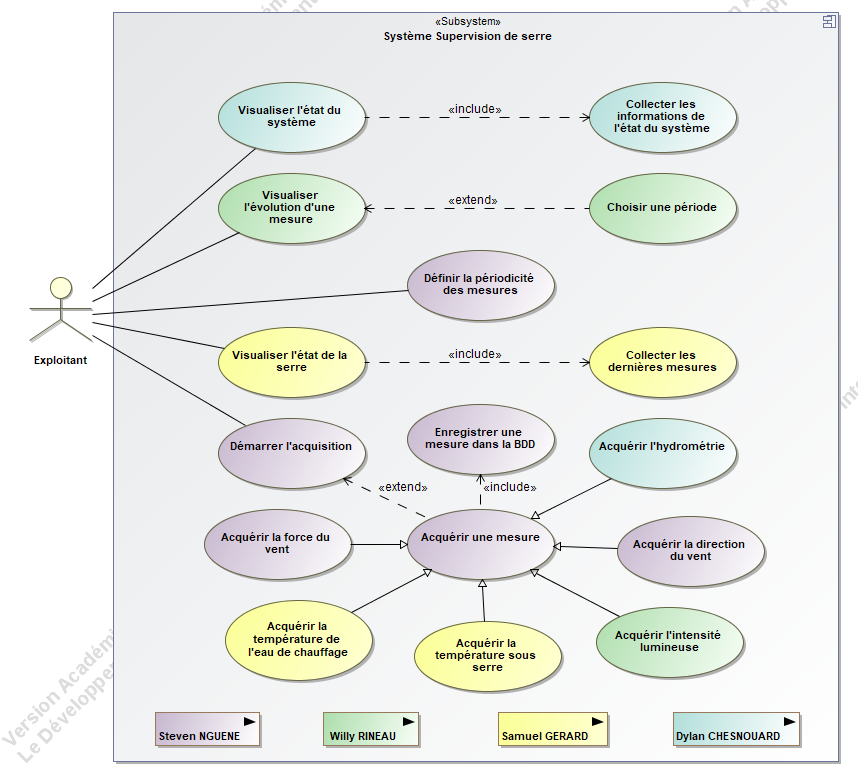
## 2.3) Étudiant 3 : Spécification des blocs Base de données/Sonde température

## 2.4) Étudiant 4 : Spécification des blocs Android/Pluviomètre

# Analyse

Pour l’analyse, nous nous intéresserons dans un premier temps aux différents cas d’utilisations de chaque bloc à créer. Voici un diagramme d’utilisation imposé par le client dans le cahier des charges. Etant très complet, nous nous en sommes inspirés pour s’en servir de base.

**Diagramme d’utilisation :**



**Diagramme de définition de blocs :**



Ce diagramme permet de visualiser l’architecture du système et représente les éléments matériels.

Les liens sont également représentés par :

* Des compositions entre l’Arduino et les capteurs ce qui signifie que **l’Arduino** contient tous les **capteurs** et ceux-ci seront détruits avec elle.
* Des associations qui montrent une connexion entre les différents systèmes.

**Diagramme de classe :**

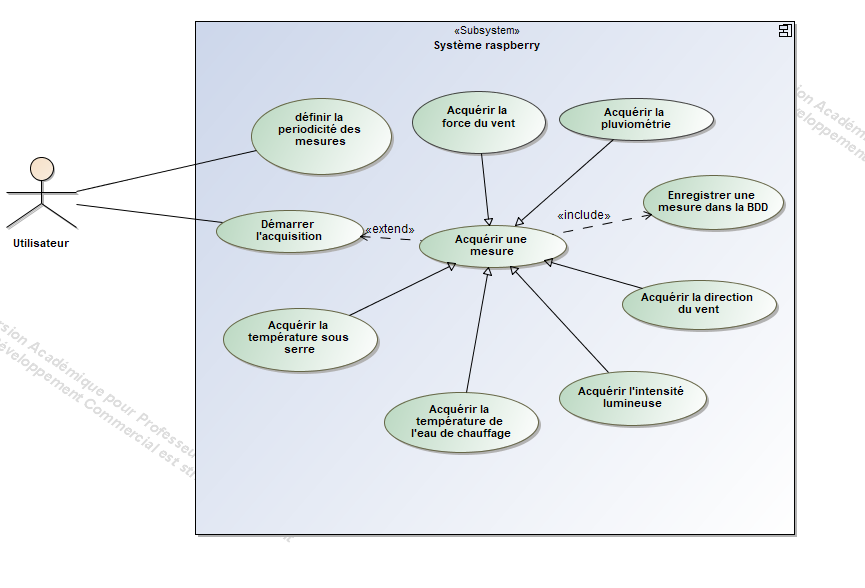
****

Ce diagramme de classe nous présente les différents liens entre les classes.

Les capteurs héritent de la classe Capteurs où ils récupèrent la méthode virtuelle « *mesurer () »*.

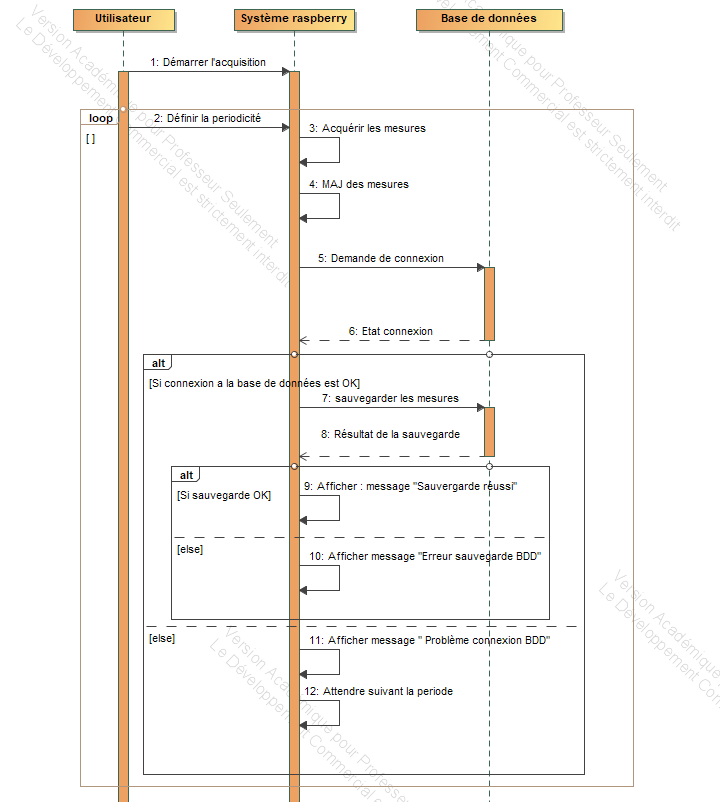
Les capteurs étant connectés directement à l’Arduino, il y a donc une composition, de même entre l’Arduino et la Raspberry. Le lien entre la Raspberry et l’interface BDD montre la connexion distante entre ces derniers.

## 3.1) Étudiant 1 : Analyse des blocs Raspberry/Anémomètre

**Diagramme de cas d’utilisation :**

Pour le bloc Raspberry, l’utilisateur pourra définir la périodicité des mesures, démarrer l’acquisition et ainsi acquérir les mesures pour chaque capteur. Les mesures seront ensuite automatiquement enregistrées dans la base de données.

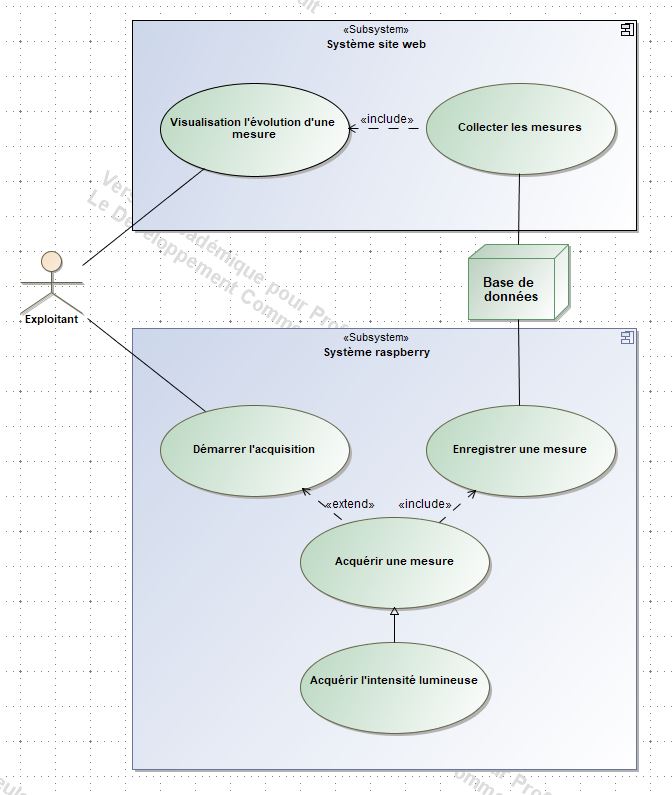
**Diagramme de séquence :**

****

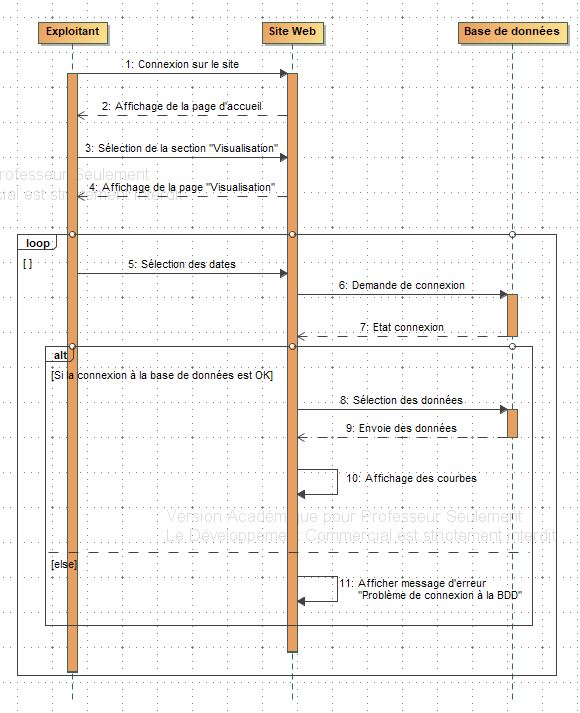
L’utilisateur pourra acquérir les mesures puis si la connexion à la base de données est effective, il pourra les sauvegarder dans celle-ci.

## 3.2) Étudiant 2 : Analyse des blocs Web/Solarimètre

**Diagramme de cas d’utilisation :**

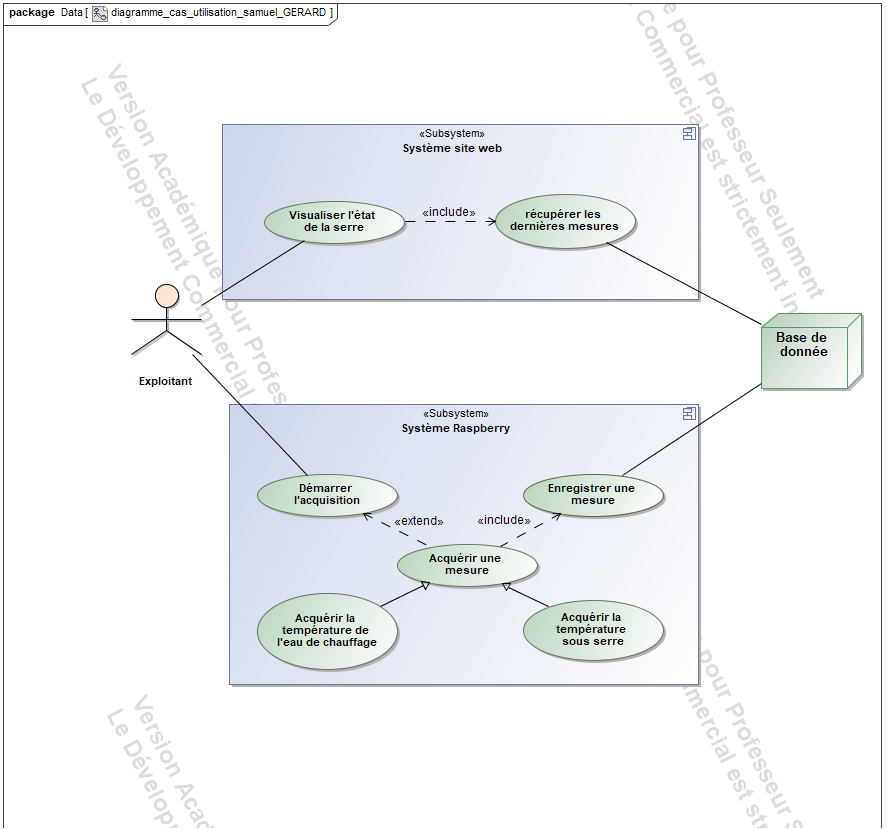


**Diagramme de séquence :**

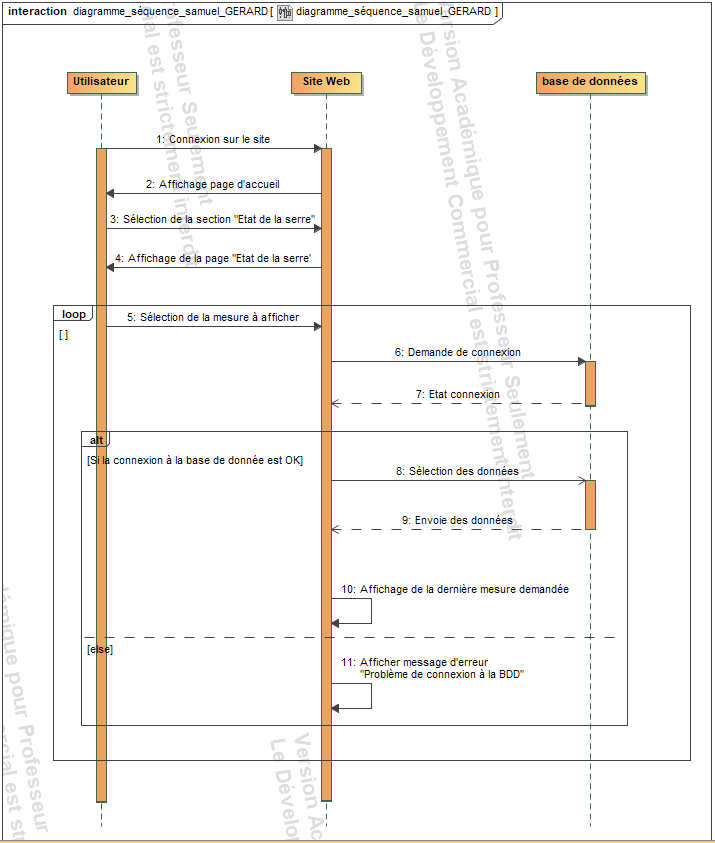


## 3.3) Étudiant 3 : Analyse des blocs Base de données/Sonde température

**Diagramme de cas d’utilisation :**

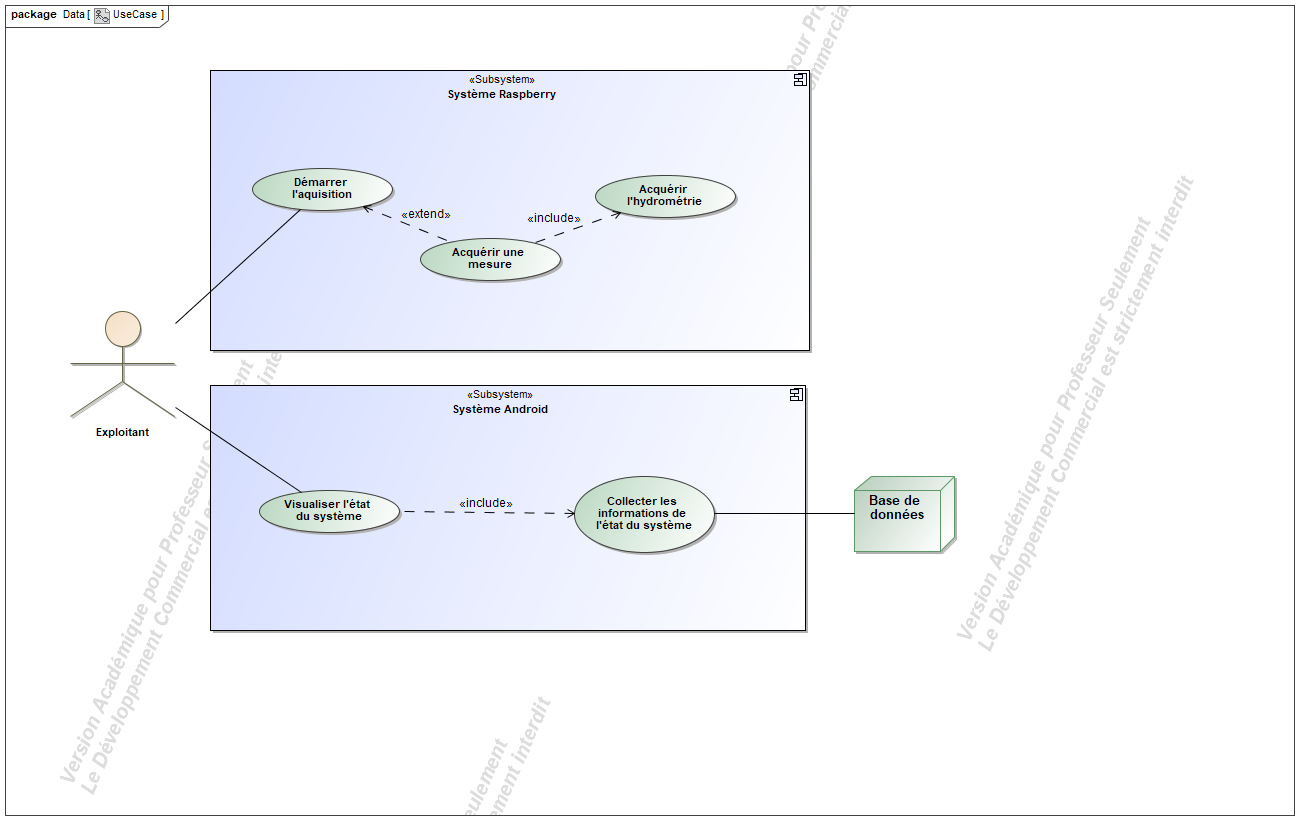


**Diagramme de séquence :**

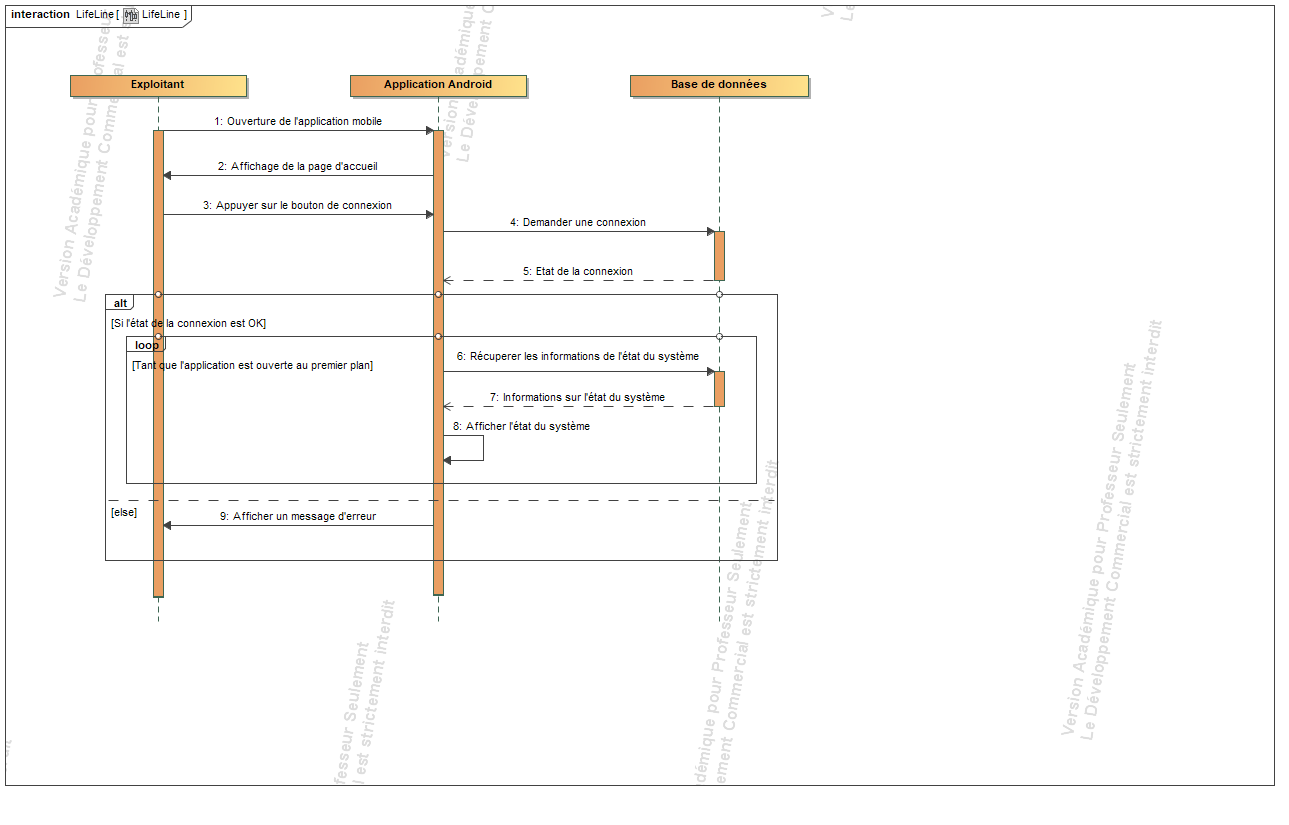
****

## 3.4) Étudiant 4 : Analyse des blocs Android/Pluviomètre

**Diagramme de cas d’utilisation :**

****

**Diagramme de séquence :**

****

# Intégration et versionning

Comme nous travaillions tous les trois sur un seul et même logiciel, nous avons décidé de mettre en place un dépôt privé avec l’utilitaire GitHub afin de nous simplifier grandement la tâche d’intégration de chaque bloc. Après avoir exporté le projet initial vers le dépôt, il nous a été possible de le cloner sur chacun de nos ordinateurs.

Durant la réalisation du projet, après chaque phase de travail, chacun pouvait enregistrer ses modifications puis les exporter vers le projet global en ligne. Ainsi, nous avions tous en permanence les modifications des uns et des autres ce qui permettait d’avoir une intégration continuelle. Cela nous a permis de gagner du temps de conception et de réalisation.

Pour cela, GitHub propose un utilitaire Windows nommé GitHub Desktop qui permet d’effectuer toutes ces opérations très simplement. Chaque modification du code ou des fichiers sources sont réversibles et on peut explorer les versions plus anciennes du projet. Les lignes de code ajoutées et supprimées apparaissent de manière très lisible.

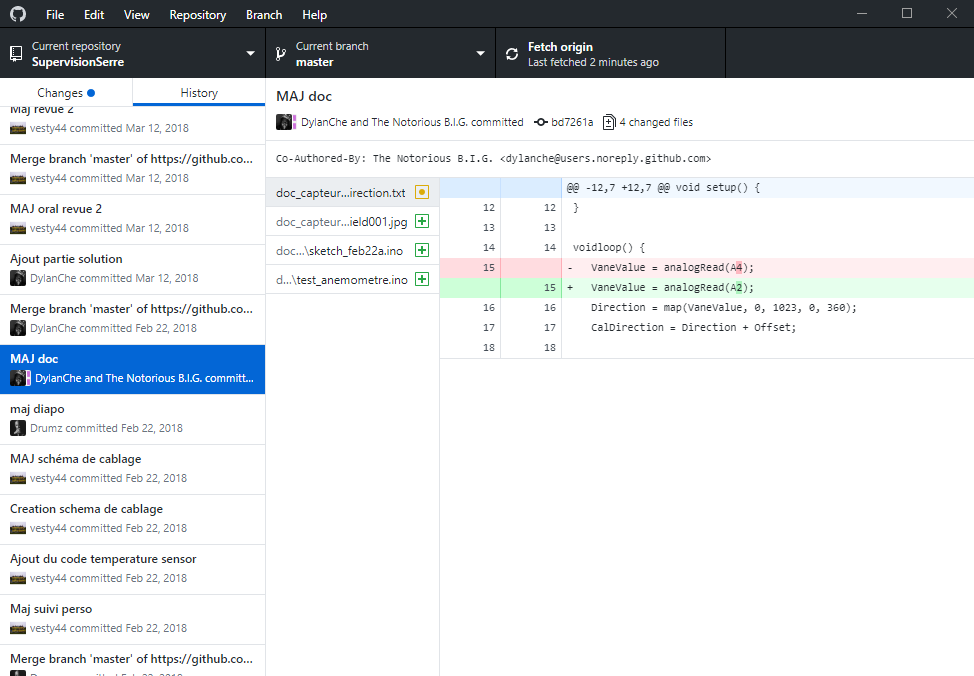


Figure 1 : Fenêtre principale de l’utilitaire GitHub Desktop

# Communication de groupe

# Conclusion

# Annexes

## 7.1) Diagramme de Gantt

## 7.2) Procédure d’installation

# Glossaire