

Table des matières

[I. Présentation du projet 3](#_Toc505870062)

[1. Information générale 3](#_Toc505870063)

[2. Contexte 3](#_Toc505870064)

[3. Objectifs du projet 3](#_Toc505870065)

[4. Finalité du projet 4](#_Toc505870066)

[5. Equipement 4](#_Toc505870067)

[6. Synoptique de la réalisation 8](#_Toc505870068)

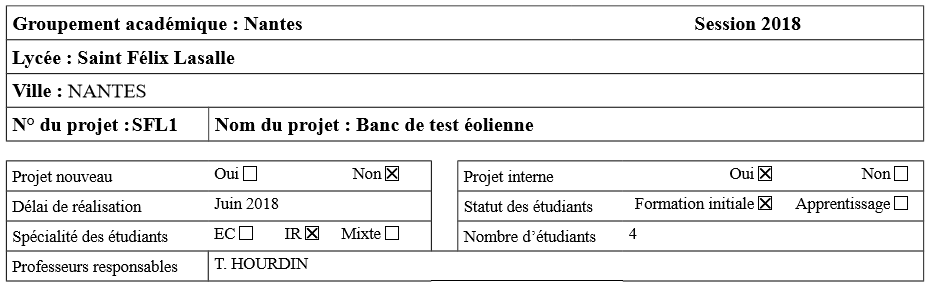
[7. Répartition des tâches 12](#_Toc505870069)

[8. Gestion des fichiers 9](#_Toc505870070)

[9. Communication de groupe 9](#_Toc505870071)

# Présentation du projet

## Information générale



## Contexte

En France, d’après l’Association des Professionnels du Petit Eolien (AFPPE), 2 500 petites éoliennes ont été installées entre 2010 et 2012. Le marché compte aujourd’hui plusieurs fabricants français, mais ceux-ci éprouvent des difficultés à se développer : en effet, la rentabilité économique pour l’utilisateur (le producteur ou l’auto consommateur) n’est pas toujours garantie, et la concurrence de fabricants étrangers produisant de petites éoliennes en grande série reste forte.

Elles permettent de s’alimenter en électricité en total autonomie. Si elle n’est pas consommée immédiatement, elle est stockée dans des batteries de stockage. Malheureusement, leur prix élevé des batteries marginalise cette solution. De plus leur durée de vie excède rarement 7 ans en moyenne c’est pourquoi le choix de l’éolienne est important.

Mais en milieu urbain une éolienne avec axe horizontal n’est pas adaptée, c’est pourquoi Monsieur Jacky ROBIN a donc conçu et fabriqué une première éolienne à axe vertical, type Savonius. Celle-ci fut testée avec la soufflerie disponible dans l’établissement.

L’objectif principal de ce projet est donc de vérifier la rentabilité de l’éolienne et en évaluer son efficacité.

## Objectifs du projet

* Simplifier la mise en œuvre du banc par la création d'un logiciel de commande et de supervision.
* Rendre possible la création de séquences de test, et ainsi évaluer les différences de performances d'éoliennes selon des scénarios définis.
* Stocker l'information (données de performances éoliennes), afin de consulter des données mesurées.
* Simplifier la consultation des relevés (instantanés ou passés) via un appareil connecté au réseau local.
* Sécuriser l'accès à la soufflerie.

## Finalité du projet

Différentes éoliennes seront testées dans ce banc de test. En effet Monsieur ROBIN a pu tester une première éolienne. Seulement il souhaite pouvoir commander et superviser le banc grâce à une application et aussi avoir la possibilité d’exécuter une séquence de test. Pour évaluer différentes performances d’éoliennes selon les scénarios définis.

Il sera aussi possible de consulter les données mesurées instantanément ou archivées sur l’application ou la page web.

Le banc de test sera contrôler par une application qui permettra de commander la consigne de 2 modes: commande de debit d’air (instantanné) et commande d’un scenario de test.(scenario).

Afin d’obtenir ce résultat l’application s’adresse a l’ordinateur embarqué via le réseau (socket), c’est ce dernier qui se charge d’effectuer les commandes adéquat avant de mettre en base de données chacun des relevés.

Depuis une interface web il sera possible de consulter ces relevé

Des capteurs de detection de l’état de la porte et de presence seront également installer afin de renforcer la securité de ce banc de test.

## Equipement

##### Liste du materiel

* Raspberry



Carte mère Raspberry Pi 3 Type B

Processeur intégré Quad-core ARM

RAM : 1024 Mo

GPU Dual Core VideoCore IV Multimedia Co-Processor

Lecteur de cartes Micro SD

Ports : HDMI, 4x USB, RJ45, jack 3.5 mm, connecteurs pour APN et écran tactile

Wi-Fi et Bluetooth

Support des distributions dédiées basées sur Linux et Windows 10

* Soufflerie



Commander de 0 à 10V

* **Le capteur CV3F** est un capteur de mesure du vent (en m/s), direction du vent (en degres) et temperature (en degres).



Câblage:

Câble coaxial RG58 entre le capteur et le boîtier de jonction.

Connexions sur bornes à vis dans le boîtier de jonction

- +12Vdc (9 à 16Vdc, 24mA)

- 0V

- sortie NMEA0183 +

- sortie NMEA0183 -

- âme coaxial

- tresse coaxial

### Gamme de mesures:

Force du vent: 0,5 à 99 Nœuds

Angle du vent: 0 à 359°

Température: -10°C à +50°C

Sensibilité des mesures:

Force du vent: 0,50 Nœuds

Angle du vent: +/-1,5°

Température du vent: +/- 0,5°C

Résolutions:

Force du vent: 000,00

Angle du vent: 000,0

Température: +/-00,0

### Température de fonctionnement:

- 10°C (hors givre) à +50°C

Phrases NMEA0183® utilisées:

IIMWV: angle, force du vent et validité de la mesure, transmission et rafraichissement 2 fois par seconde.

WIXDR: Température du vent

PLCJ: données constructeur

[](http://www.lcjcapteurs.com/wp-content/uploads/2012/07/Logo_CE.jpg)PLCJE: données constructeur

Vitesse de transmission: 4800 Bauds.

Conforme aux normes Européennes

Pendant le projet nous disposons de l’eolienne Breeze:

**Présentation**

L’**Air Breeze Marine** a été adoubée par de nombreux passionnés de la navigation comme la meilleure éolienne de sa catégorie. Son application ne se limite d’ailleurs pas aux **bateaux**, l’Air Breeze est aussi destinée aux **sites côtières** et aux plateformes **off-shore**.

Elle s’avère être un complémentaire efficace aux panneaux solaires pour un système d’alimentation **hybride**. Ce système permet de maintenir le fonctionnement énergétique, même en temps orageux ou nuageux (baisse de rendement des panneaux solaires).

Son design compact lui confère une élégance esthétique qui s’intègre très bien aux lignes des bateaux. Une gamme d’Air Breeze pour **utilisation terrestre** (pompage) existe également.

**Caractéristiques**

Cette éolienne à axe horizontal est protégée par une **peinture anti-corrosion** certifiée par les normes **aéronautiques**. Le kit de fixation (visseries) est composé de pièces **inoxydables**.

L’Air Breeze Marine est optimisée pour une production électrique **stable**, avec un régulateur électronique à **microprocesseur** pour un réglage automatique de la vitesse de rotation.

L’Air Breeze Marine est dotée d’une option d’**arrêt** ou de ralentissement **automatique**, programmée par l’utilisateur selon le niveau de tension désiré dans les batteries.

Le système d’arrêt automatique peut également se déclencher en cas de **sur-vitesse** lors de vents violents.

Selon le constructeur, l’Air Breeze démarre à un vent de **2,2m/s**.

|  |  |
| --- | --- |
| Fiche technique de l’Air Breeze | |
| **Diamètre du rotor** | 1,17m |
| **Poids** | 5,9Kg |
| **Pâle** | composite souple moulé |
| **Vitesse de démarrage** | 2,2m/s |
| **Puissance nominale** | 200W,300W,400W |
| **Alternateur** | Aimant permanent brushless |
| **Protection survitesse** | Controleur électronique |
| **Vitesse de vent max** | 49,2m/s |
| **Production** | 38KWh/mois à 6,0m/s de vent en moyenne |
| **Hauteur du mât recommandée** | 2,7m avec un diamètre de 48mm |

##### Ressources mise à disposition

Logiciel :

Les schemas d’analyse UML seront crées via le logiciel MagicDraw

L’application sera crée en c++ avec le logiciel QtCreator

Le gantt sera réalisé sur Microsoft Project

Suites bureautiques Microsoft Office 2007

Les pages Web seront développées en HTML/PHP/CSS sous NetBEANS

Matériel :

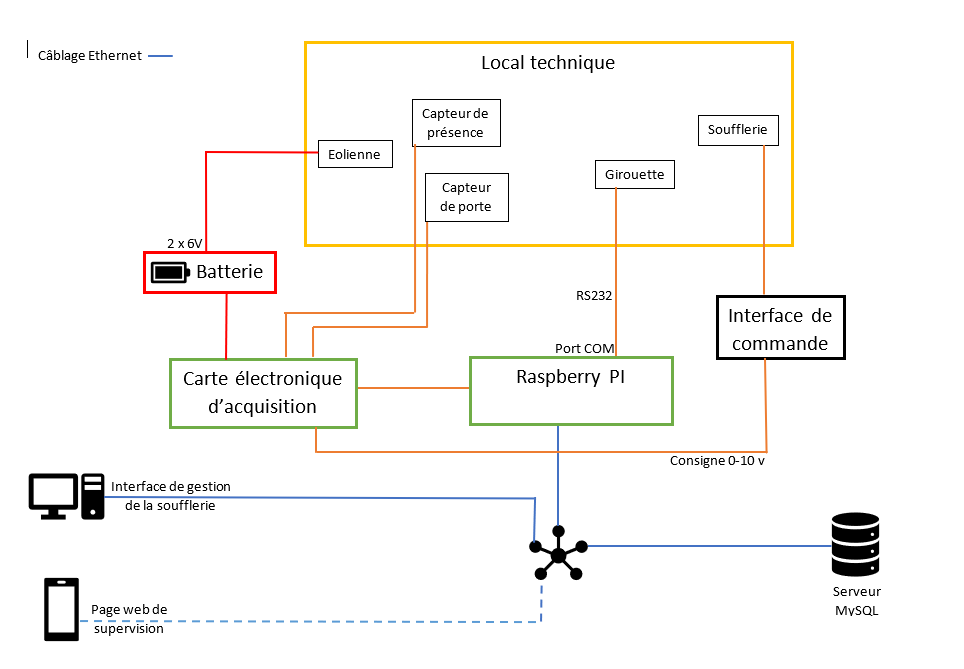
1 poste de développement par étudiant

1 carte Raspberry PI 3

Accès au banc de test

1 carte électronique + extensions

## Synoptique de la réalisation



## Gestion des fichiers

Nous avons mis en place un dépot privé via l’utiliaire GitHub afin de fascilité les échanges (notamment pour les fichiers commun). Une fois le dossier “banc\_de\_test\_eolienne” exporter vers le dépôt nous avons pu le cloner sur chacun de nos ordinateurs.

Tout au long de la réalisation du projet, après chaque séance de travail, chacun pouvait donc enregister les différentes modifications apportés grâce a GitHubDesktop puis les exporter en ligne.

Cela nous a donc permis de gagner du temps et de sauvegarder notre travail en ligne.



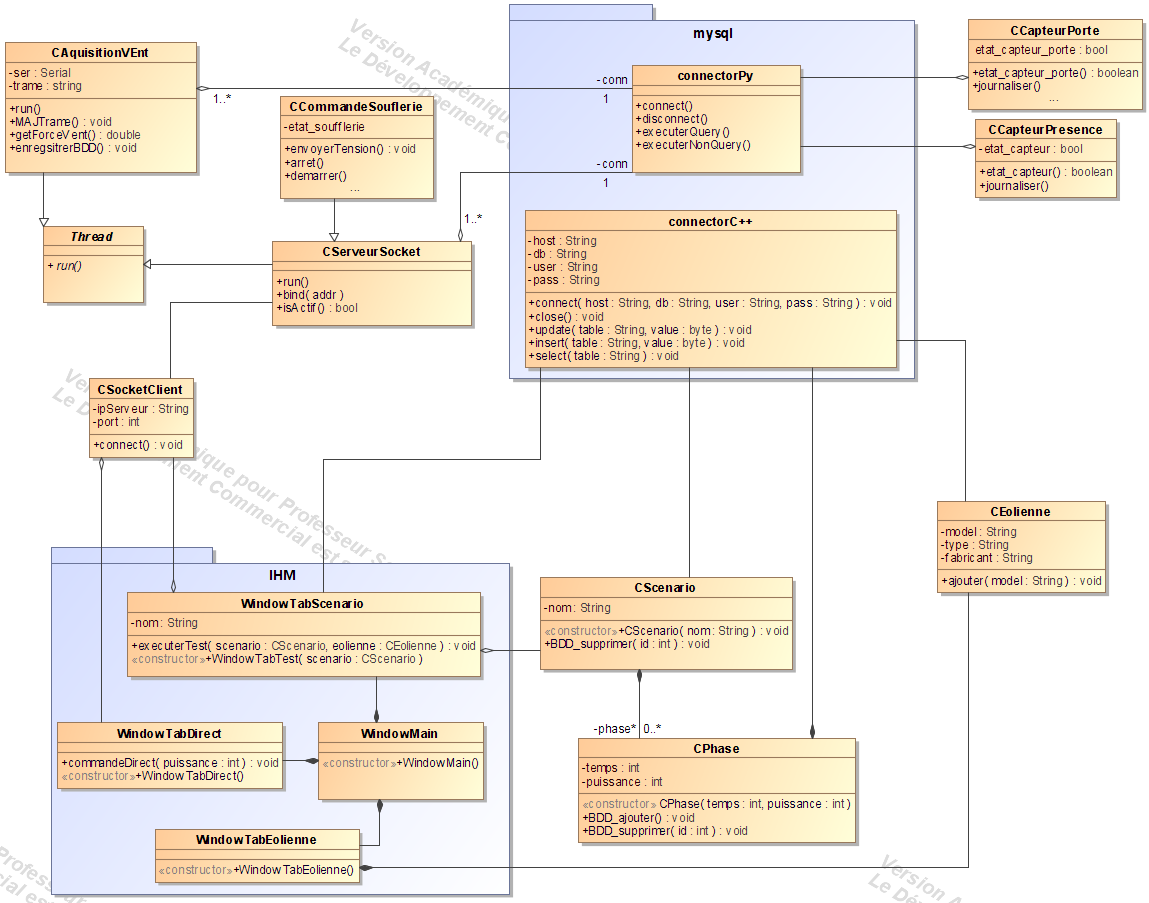
## Communication de groupe

Afin d’organiser les taches réaliser dans le diagramme de Gantt, nous avons avons utilisé le site internet Trello. Il fonctionne par “carte” que l’on peut organiser dans différents groupes. Il permet aussi d’identifier une ou plusieurs personne.

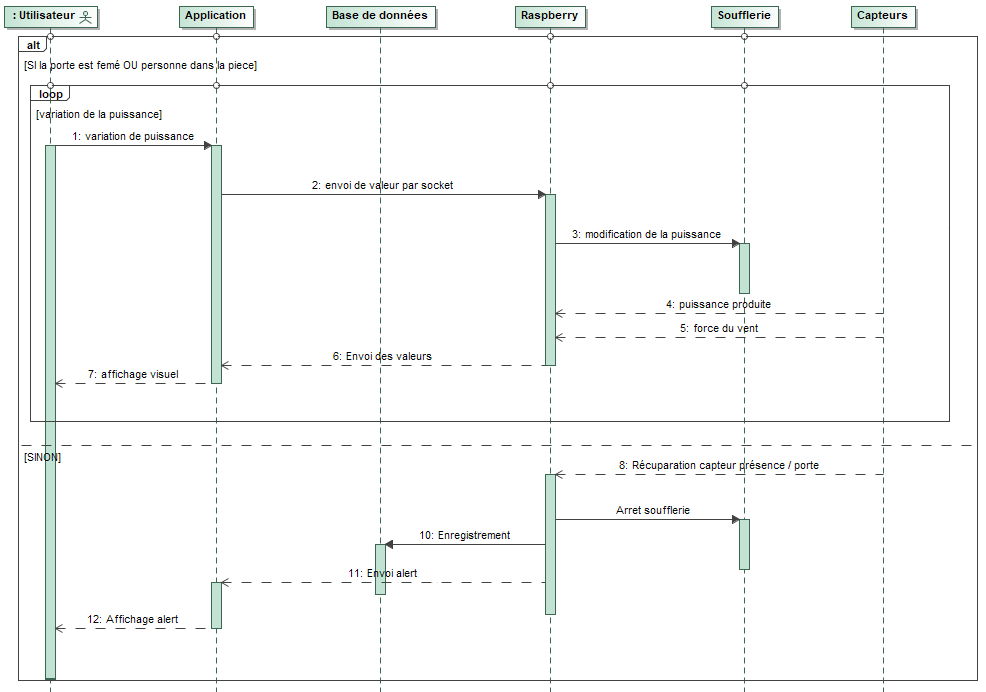
# Analyse

## Analyse UML

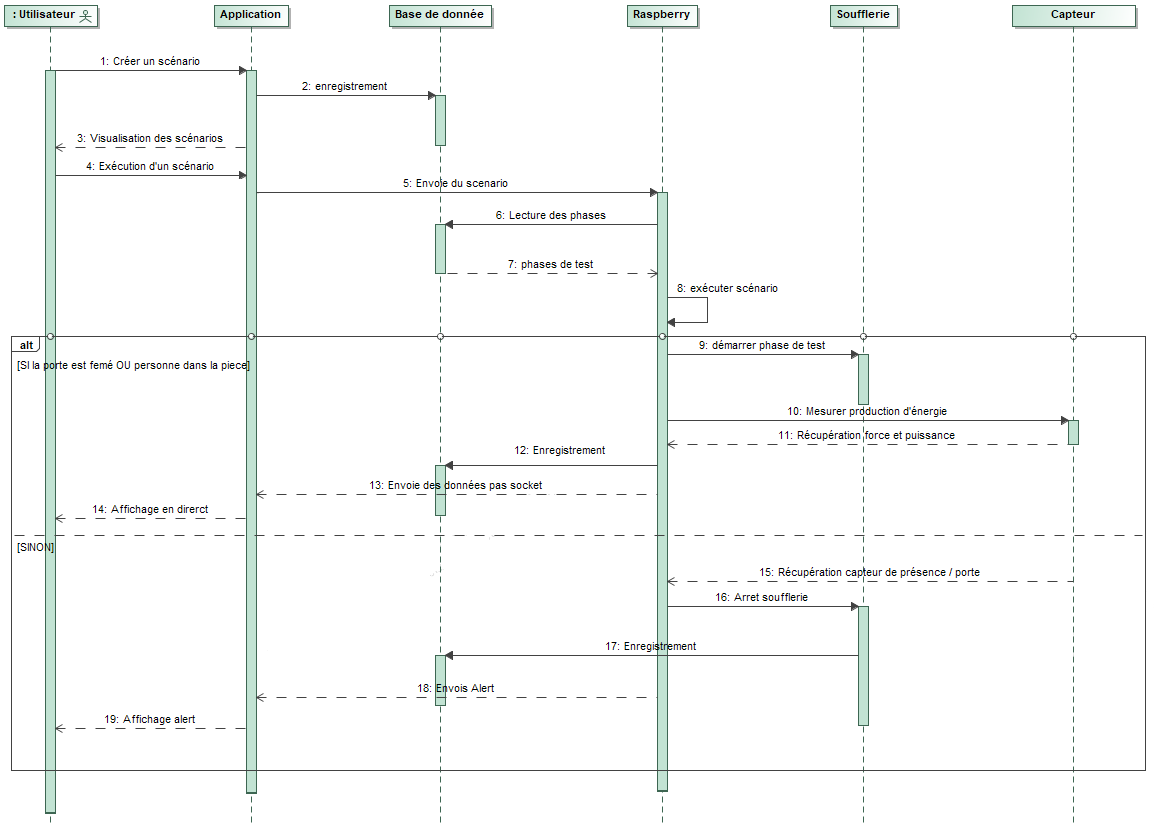
### Diagramme de classe



### Diagramme de sequence (commande direct)



### Diagramme de sequence (mode scenario)



# Répartition des tâches

### Etudiant 1, Alan Carrer :

Actuellement

La commande de consigne (0-10v) se fait manuellement par le biais d'un potentiomètre situé sur un pupitre à proximité de la soufflerie.

Contrôler la commande depuis une application avec deux options de contrôles. Au démarrage de l'application, les 2 modes sont proposés. :

Mode instantané

Une interface affiche un curseur permettant de modifier la consigne (entre 0 et 100 % du max de la soufflerie). L'interface affiche en instantané la force du vent et la puissance éolienne produite.

Mode scenario

Ce mode de test implique :

- de sélectionner ou créer un scénario de test (périodes de puissances)

- de sélectionner, ou enregistrer, une éolienne afin de retrouver les performances liées à une éolienne

En mode scénario, l'opérateur sélectionne l'éolienne et le scénario, puis il démarre sa séquence de test. Au fur et à mesure du déroulement du test, la force du vent et la puissance produite sont mises à jour en instantané.

Les tests effectués sont archivés dans une base de données, et accessible via l'interface Web.

Pour chaque éolienne, les informations à enregistrer seront à définir (modèle, type, fabricant, …). Cependant, il sera possible d'attacher un document technique à l'éolienne (permettant ainsi d'accéder aux données constructrices en vue de les confronter avec les mesures effectuées).

*Un compte rendu de scénario est affiché en fin de test.*

### Etudiant 2, Arnaud Jullien

**Outils**

-Raspberry

-Carte électronique d’interface

**Objectifs**

-Capturer la puissance et la commande de la consigne (0-10v)

-Être en relation avec un PC via le réseau (socket)

-Mise à jour des informations dans la base de données

### Etudiant 3, Manon Morille :

Développement de pages Web de consultation

**Objectifs**

-Site responsif

-Consultation en direct de la force du vent et des historiques de scenario de test

**Par défaut**

Force de vent de 0 m/s (m/s ou km/h).

Production de 0Watts. La force du vent peut être affichée en m/s ou km/h.

*L'énergie produite par l'éolienne charge 2 batterie de 12V montées en série. Ainsi, la mesure du courant de charge multiplié par la tension de charge nous donnera la puissance instantanée.*

### Etudiant 4, Killian Labattut :

**Actuellement**

Aucune sécurité n’est mise en place ce qui fait qu’une personne peut pénétrer dans la pièce alors que celle-ci est en fonctionnement.

**Objectifs**

-Mise en place d’un capteur afin de connaître l’état de la porte (ouverte ou fermée) afin de stopper la mise en route de la soufflerie si la porte est ouverte (arrêt d’urgence)

-Mise en place d’un capteur de détection présence afin de ne pas démarrer la soufflerie

-Indication sur le logiciel de l’arrêt d’urgence ou de l’impossibilité de démarrer

-Indication sur le logiciel de l’état du capteur de la porte et de présence

*Possibilité de rajouter une alarme visuelle ou sonore à l'extérieure de la soufflerie pour alerter l'opérateur en cas d'ouverture de porte ou de présence non prévus.*

# Glossaire

Soufflerie: Ventilateur surdimensionné, dans le cas présent il sert le banc de test éolien.

CSS: Le terme CSS est l'acronyme anglais de Cascading Style Sheets qui peut se traduire par "feuilles de style en cascade". Le CSS est un langage informatique utilisé sur l'internet pour mettre en forme les fichiers HTML ou XML. Ainsi, les feuilles de style, aussi appelé les fichiers CSS, comprennent du code qui permet de gérer le design d'une page en HTML.

PHP: Le PHP est un langage informatique utilisé sur l'internet. Le terme PHP est un acronyme récursif de "PHP: Hypertext Preprocessor".

Ce langage est principalement utilisé pour produire un site web dynamique. Il est courant que ce langage soit associé à une base de données, tel que MySQL.

GitHub: GitHub (exploité sous le nom de GitHub, Inc.) est un service web d'hébergement et de gestion de développement de logiciels, utilisant le logiciel de gestion de versions Git. Ce site est développé en Ruby on Rails et Erlang par Chris Wanstrath, PJ Hyett et Tom Preston-Werner.

Trello: Trello est un outil de gestion de projet en ligne, lancé en septembre 2011, et inspiré par la méthode Kanban de Toyota. Il est basé sur une organisation des projets en planches listant des cartes, chacune représentant des tâches. Les cartes sont assignables à des utilisateurs et sont mobiles d'une planche à l'autre, traduisant leur avancement.