



Le génie pour l'industrie

RAPPORT TECHNIQUE
PRÉSENTÉ À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
DANS LE CADRE DU COURS GTI795 PROJET DE FIN D'ÉTUDES EN GÉNIE DES TI

PROJET AI MATRICIS

MIGUEL FIGUEROA ROSALES
FIGM05108306

GUILLAUME TOUBACHE
TOUG27029407

RONI MOUFARREJ
MOUR16049403

DÉPARTEMENT DE GÉNIE LOGICIEL ET DES TI

Professeur-superviseur

Alain April

MONTREAL, 12 AOÛT 2019
ÉTÉ 2019

REMERCIEMENTS

Merci à l'équipe de Matricis pour nous avoir permis de réaliser notre projet de fin d'études avec vos experts.

PROJET AI MATRICIS

MIGUEL FIGUEROA ROSALES
FIGM05108306

GUILLAUME TOUBACHE
TOUG27029407

RONI MOUFARREJ
MOUR16049403

RÉSUMÉ

Ce rapport va tout d'abord d'écrire l'entreprise qui nous a donné comme mandat d'effectuer ce projet. Puis l'objectif en tant que telle de ce projet qui était de présenter le flux de travail pour un projet d'apprentissage machine dans l'application ThingWorx. Par la suite, nous allons présenter la façon dont nous avons travaillé tout au long de ce projet, soit avec des rencontres hebdomadaires ainsi qu'en priorisant la communication par courriel pour laisser une traçabilité plus simple à suivre au cours du projet. Puis nous allons expliquer la conception dans l'application ThingWorx, soit intégrés les datas à l'aide de ThingWorx Analytics Builder, puis en créant le modèle dans ce module nous pouvions ensuite le déployer dans ThingWorx Analytics Manager et l'exécuter. Avec ceci il était possible d'obtenir les résultats nécessaires.

Puis à l'aide de ceux-ci nous avons réussi à établir un flux de travail avec ThingWorx Foundation dans le mélange pour être en mesure de prendre les données de la turbine et les ajouter à ThingWorx Analytics. Qui est celui recommandé à Matricis pour être en mesure de compléter le projet.

Il y a eu plusieurs problèmes rencontrés, mais ceux-ci bien qu'ils ont ralenti le projet non pas empêcher sa complétion.

TABLE DES MATIÈRES

CHAPITRE 1 - INTRODUCTION	1
CHAPITRE 2 - PROBLÉMATIQUE.....	2
2.1 Description du projet.....	2
2.2 Mandat du projet.....	2
2.3 Outils utilisés	3
2.4 Résultats Attendues.....	3
2.5 Risques du projet.....	3
2.6 Membres de l'équipe.....	5
CHAPITRE 3 – MÉTHODOLOGIE DE TRAVAIL	6
3.1 Gestion du projet.....	6
3.2 Progrès du projet.....	6
3.3 Distribution des Tâches.....	7
3.4 Communication.....	7
3.5 Outils et technologies utilisés	7
3.5.1 ThingWorx.....	8
3.5.2 Extension Java	8
CHAPITRE 4 – CONCEPTION.....	9
4.1 Extension Java	9
4.2 ThingWorx.....	13
4.2.1 ThingWorx Analytics.....	13
4.2.2 ThingWorx Foundation.....	14
CHAPITRE 5 – RÉSULTATS	15
5.1 Extension Java	15
5.2 ThingWorx.....	15
CHAPITRE 6 - DIFFICULTÉS RENCONTRÉS	22
6.1 Compréhension du projet et problème d'équipe.....	22
6.1.1 Description Initial du projet.....	22
6.1.2 Limité sur le temps.....	22
6.1.3 Conflit d'horaire.....	23
6.2 ThingWorx	23
6.2.1 Compréhension de ThingWorx.....	23
6.2.2 Limitation par les accès	24
6.2.3 Utilisation des Extensions ThingWorx	24
6.3 Extension Java	24
6.3.1 Librairies ThingWorx	25
6.3.2 Librairies des descriptions	25
CHAPITRE 7 - SOLUTIONS	26
7.1 Solution ThingWorx	26

CHAPITRE 8 - ENJEUX ENVIRONNEMENTAUX	27
8.1 Réduction de l’empreinte écologique	27
CHAPITRE 9 - CONCLUSION	28
CHAPITRE 10 - LISTE DE RÉFÉRENCES	29
CHAPITRE 11 - BIBLIOGRAPHIE	30
CHAPITRE 12 - APPENDICES	31

LISTE DES FIGURES

	Page
Figure 1 Tableau des risques	4
Figure 2 Tableau des membres et rôles de l'équipe	5
Figure 3 Code dans Eclipse pour ThingWorx	9
Figure 4 Build Gradle et l'exécution de Gradle.....	10
Figure 5 Importation dans ThingWorx	10
Figure 6 Sélection de type d'importation dans ThingWorx	11
Figure 7 Sélection du fichier .Zip	11
Figure 8 Importation complété	12

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

KNN	K-nearest neighbors algorithm (Méthode des k plus proches voisins)
SVM	Support Vector Machine (Machines à vecteurs de support)
NN	Neural networks (Réseau de neurones)
NASA	National Aeronautics and Space Administration
IDE	Integrated development environment (Environnement de développement)

CHAPITRE 1 - INTRODUCTION

L'internet des objets est la connexion entre les objets, les lieux et les environnements de la vie avec l'internet. Avec l'internet des objets, nous allons voir une croissance de l'interconnexion des machines, soit les téléphones intelligents, mais aussi tout ce qui était dans à la maison. Il sera possible de connecter la laveuse, la sècheuse, le lave-vaisselle, avec l'innovation nous pourrons complètement changer notre style de vie et être en mesure de contrôler l'entièreté de sa maison avec son ordinateur ou même avec son téléphone portable. Tout ceci grâce à l'interconnexion entre ces objets et l'internet.

Matricis, est une entreprise qui se spécialise en intégration d'applications d'entreprise à l'intégration de données et à la mise en place d'architecture orientée services. Matricis a proposé un projet à notre équipe ayant pour but d'explorer les fonctionnalités de l'intelligence artificielle dans l'application ThingWorx. L'équipe de projet avait donc pour mandat de comprendre comment appliquer des modèles d'entraînement sur des données générés par une turbine pour être en mesure d'arriver au même résultat que d'autres applications qui ont aussi des modèles d'entraînements différents.

Ce rapport va présenter tout d'abord la problématique ainsi que les risques liés au projet, la méthodologie de travail, puis, la conception, les difficultés rencontrées et finalement les différentes solutions mises en œuvre et les solutions proposées à Matricis.

CHAPITRE 2 - PROBLÉMATIQUE

Ce chapitre portera sur la description du projet, le mandat, les outils utilisés, les résultats attendus, les risques du projet et ce que chacun des membres de l'équipe peuvent apporter au projet.

2.1 Description du projet

Le projet étant nouveau pour tous les partis concernés étaient un projet d'exploration. Le rendu final optimal était connu, mais les façons d'y arriver ne l'étaient pas. Matricis avait besoin d'intégrer un nouveau module à leur application déjà existante, soit le module « ThingWorx Analytics ». Ce module étant tout nouveau, aurait besoin d'être exploré par l'équipe. Ce qui était connu de ce module est le fait qu'il est en mesure de produire des résultats avec des modèles d'intelligence artificielle. Grâce à ceci, il serait possible d'entraîner le modèle avec des données fournies et différents classificateurs tels que KNN, SVM et NN.

2.2 Mandat du projet

Pour ce projet, il fallait être en mesure de produire une logique dans « ThingWorx » pour être en mesure de reproduire les mêmes résultats que ceux produit par la NASA. En d'autres termes il fallait voir si avec « ThingWorx » il était possible de reproduire les mêmes résultats qu'avec les autres applications servant à l'entraînement et à la classification de modèle.

2.3 Outils utilisés

Les outils utilisés dans ce projet étaient en constante évolution puisque l'équipe explorait les différentes solutions offertes. Sur « ThingWorx » il est possible de créer des extensions en Java, pour appliquer des modifications qu'il n'offre pas normalement. En autres dans « ThingWorx » il y a aussi d'autres modules qui ont été nécessaire au projet, tout d'abord il y avait « ThingWorx Analytics » qui est un module qui permet d'automatiser les analyses à l'aide d'entraînement de modèle, il supporte des gros volumes de données et plusieurs classificateurs. Dans ce module, il y a deux sous-sections nécessaires à la création et à l'utilisation des modèles. La première étant « Analytics Builder » qui comme son nom l'indique sert à créer les modèles d'entraînement avec les données fournies. Le deuxième ce nommant « Analytics Manager » qui lui permet de déterminer les modèles à exécuter, ajouter des événements, des « jobs » et d'observer les résultats suite à l'entraînement et la classification.

2.4 Résultats Attendues

Les résultats attendus comme énoncés plutôt, sont d'être en mesure de reproduire les mêmes résultats avec « ThingWorx » qu'avec n'importe quelles autres applications d'entraînement et de classification de modèles. Puisque les données fournies pour l'entraînement et la classification sont partagés, il a été possible d'avoir un équivalent des résultats avec R script par Matricis. Bien que les données aient été réduites à cause des limitations de R, il était possible d'avoir un résultat comparatif pour voir si les classifications faites étaient acceptables ou non.

2.5 Risques du projet

Il y a quelques risques dans ce projet, mais ils sont tout de même critiques et pourraient entièrement changer la tournure du projet si ceux-ci se produisent. Les risques sont présentés dans le tableau de la figure 1.

ID	Description	Probabilité	Impact	Gestion
R01	Compatibilité des solutions et des outils	4	10	Il n'y a pas de réelle solution à ce problème, si « ThingWorx » n'est pas compatible avec le projet, alors le projet devient non-viable.
R02	Manque d'expérience avec les technologies	8	7	Nous allons faire les efforts nécessaires pour nous familiariser avec les différentes technologies et outils que nous devons connaître.
R03	Sous-estimation de la complexité	6	6	Il est dur de déterminer qu'est-ce qu'on ne connaît pas, si des parties du projet se révèlent impossible ou prendraient trop de temps, alors on en discutera avec les chargés de projet.
R04	Temps alloué pour le projet insuffisant.	5	5	Notre but serait d'avancer le projet autant que l'on peut pour laisser le moins à faire. Mais si jamais nous ne pouvons terminer le mieux que nous pouvons faire est de laisser toute la documentation possible pour que les personnes qui vont continuer puissent reprendre le projet aisément.
R05	Limite de disponibilités	3	3	Si les horaires des membres de l'équipe et celle de Matricis ne conviennent pas à personne le mieux est de bien diviser les tâches pour que même si on ne se rencontre pas souvent on puisse tout de même avancer à un bon rythme.
R06	Limite des ressources	5	4	Puisque le projet étant dans l'inconnu pour les membres de l'équipe il sera nécessaire de vérifier constamment comment les ressources sont allouées pour être en mesure qu'il ne manque pas de temps pour produire les résultats attendus.

Figure 1 Tableau des risques

2.6 Membres de l'équipe

L'équipe est composée de plusieurs membres de Matricis, du chargé responsable de l'ETS, ainsi que les étudiants qui font leurs Projet de fin d'étude. Ceux-ci sont présentés dans la Figure 2.

Membre	Rôles	Responsabilités
Réjean Ouellet	Propriétaire du projet Administrateur ThingWorx	- Émettre les exigences et les spécifications du projet - Support de l'équipe avec ThingWorx - Gérer les accès ThingWorx
Cédric Melançon	Architecte ThingWorx	- Support de l'équipe avec ThingWorx
François Blackburn-Grenon	Gestionnaire de projet	- Assurer le suivi du projet, organisation des rencontres
Alain April	Enseignant Responsable ETS	- Évaluer le projet, le rapport et les artéfacts
Christos Karras	Expert AI Architecte d'intégration	- Support de l'équipe avec toute la partie Intelligence Artificielle, transformation de données et intégration
Miguel Figueroa Rosales	Chef d'équipe Développeur ThingWorx	- Intermédiaire entre l'équipe de développeur et l'équipe de Matricis - Développement sur ThingWorx Analytics
Guillaume Toubache	Développeur Java	- Développement de l'extension Java - Production des artefacts nécessaires - Construction du guide d'utilisateur
Roni Moufarrej	Développeur Java	- Développement de l'extension Java - Production des artefacts nécessaires - Construction du guide d'utilisateur

Figure 2 Tableau des membres et rôles de l'équipe

N'ayant aucun membre de l'équipe qui connaissait préalablement ThingWorx, les membres ont dû se familiariser avec l'application avant de pouvoir déterminer qui allait faire quoi. Certains membres étant déjà familier avec le Java et plus à l'aise que d'autres ont été en mesure de prendre tout ce qui était relié à cette partie.

CHAPITRE 3– MÉTHODOLOGIE DE TRAVAIL

Cette partie du rapport a pour but de présenter comment le projet a été géré et comment celui-ci a progressé. La façon dont les tâches et rôles ont été attribués au début du projet et comment ceux-ci ont changés et évoluer à-travers le temps. Il sera également décrit les différents outils et technologies qui ont été utilisés pour faire progresser le projet et les raisons pour leurs sélections.

3.1 Gestion du projet

La méthodologie de gestion du projet a été adoptée dès la première rencontre avec les gens de Matricis. Le projet était à but exploratoire et de taille variable, donc l'approche utilisée n'as pas été très strict. Chaque semaine, les différentes personnes impliqués dans le projet se rencontraient chez Matricis (ou en utilisant un lien Skype si nécessaire) pour discuter de l'avancement du projet et poser des questions sur la direction voulue, ainsi que des questions plus techniques relié aux outils et la technologie utilisée. La journée et heure des rencontres pouvait être flexible et reporté une autre journée si nécessaire.

3.2 Progrès du projet

L'avancement du projet était décidé de façon non structuré au fur et à mesure. Il aurait été possible d'utiliser un tableau de bord pour déterminer les tâches à faire et le temps à dédier pour chacune, mais dû à la petite taille de notre équipe et le fait qu'on communiquait déjà de façon appropriée nous avons décidé de ne pas en utiliser un. En rétrospective il aurait utile de le faire simplement pour mieux simuler un projet en milieu d'entreprise et être plus professionnel. Le début du projet fut tumultueux initialement dû à son statut explorateur et le fait que personne ne connaissait bien les capacités de ThingWorx, de ce fait plusieurs semaines fûts nécessaire pour réellement commencer à avancer et produire des résultats concrets. Il aurait été mieux si les attentes et résultats attendus avait déjà été définis avant le début du projet au lieu d'avoir à faire cette étape durant celui-ci.

3.3 Distribution des Tâches

Une fois que le projet a réellement commencé il a été plus simple de diviser les tâches selon ce que chacun voulait faire et les choses ont été beaucoup plus organisées à partir de là. Initialement une partie de l'équipe de l'équipe était supposé travailler sur ThingWorx, mais il est rapidement devenu clair que cette partie était dure à diviser entre deux personnes vues que c'était plus un travail de configuration et non du développement. Au final deux membres de l'équipe ont été mis sur la partie de développement Java et de la documentation et un sur ThingWorx.

3.4 Communication

La planification des rencontres avec le client se faisaient par emails, elles étaient toutes prévues pour le lundi de 13h à 14h et étaient dans l'agenda électronique du système de l'ETS, elles pouvaient être déplacés selon les besoins de chacun. Matricis possédait également un Slack avec lequel l'équipe pouvait communiquer de façon plus simple avec le client en termes d'échange d'idées et de questions. La communication entre les membres de l'équipe se faisait principalement par Messenger pour les messages écrits et Discord pour VoIP.

3.5 Outils et technologies utilisés

Divers outils et modules ont été utilisés durant le projet, ceux-ci seront décrit dans la section suivante les dépendances et utilité de ceux-ci, comment ils sont reliés au projet et leur fonctionnement général.

3.5.1 ThingWorx

ThingWorx a été créé par PTC, une entreprise de l'industrialisation de l'internet des objets ayant pour but de permettre à ses clients d'accéder, d'analyser, monétiser et d'interagir avec leurs données. ThingWorx est la plateforme permettant d'accomplir ceci, offrant selon PTC beaucoup de fonctionnalité, flexibilité, évolutivité, sécurité et un retour rapide sur l'investissement. Avec ses modules il est possible prendre des données faire des transformations diverses sur celle-ci et d'avoir une sortie du type qu'on souhaite.

3.5.2 Extension Java

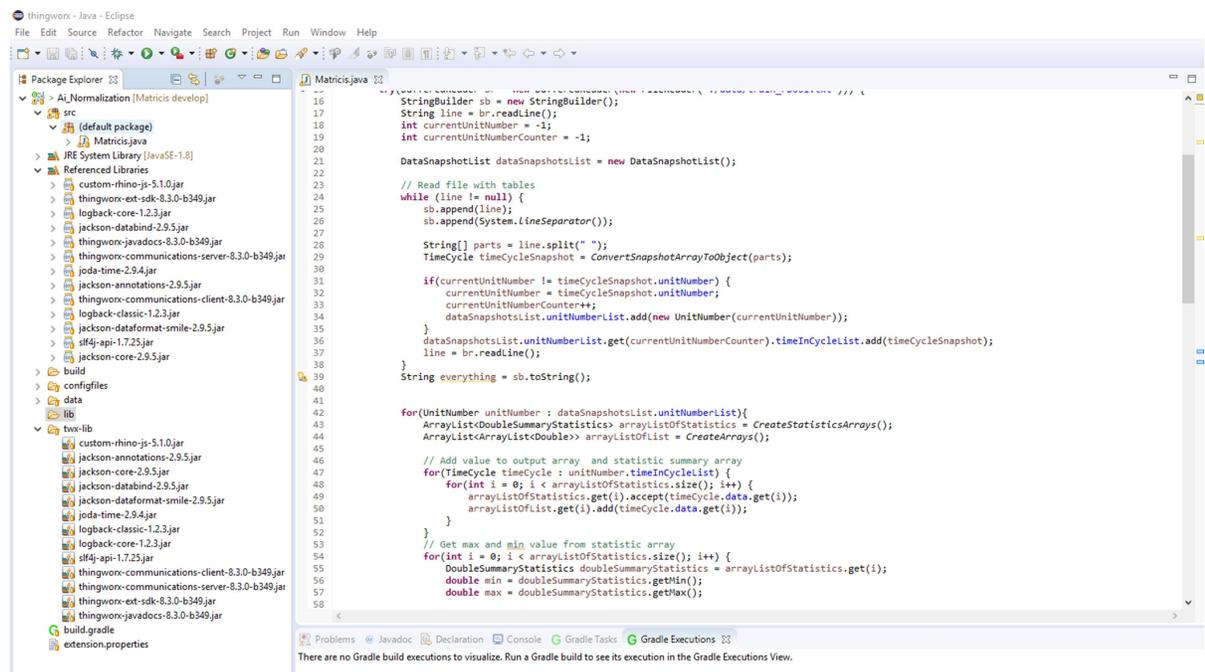
L'extension Java permet d'interagir avec ThingWorx de façon plus personnalisée, ThingWorx étant limité par ses modules et étant plus un travail de configuration que de création, ses capacités sont naturellement limitées. L'extension Java permet de contourner ses limitations en créant ses propres programmes sur un IDE compatible avec Java, dans notre cas Eclipse était utilisé avec un dépôt Git pour travailler en équipe sur le programme.

CHAPITRE 4 – CONCEPTION

Dans cette section nous présenterons ce qui a été fait pour arriver aux résultats finaux. Tout ce qui a été codé ou configuré sera présenté ici avec une explication pour faciliter la compréhension de ceux-ci.

4.1 Extension Java

L'extension que nous avons créée avait pour but initial de permettre de faire le calcul de moyenne sur les données brutes préalablement traitées par Matricis, nous n'étions pas certains si ThingWorx allait être capable de faire les calculs de la façon d'on nous souhaitons. Le but était d'explorer les capacités de cet outil et de voir s'il était réellement fonctionnel et si des futurs utilisateurs pourraient s'en servir ou s'il était préférable d'utiliser les modules prédéfinis de ThingWorx. Le processus pour créer l'extension est expliqué en détail sur <https://developer.thingworx.com/en/resources/guides/eclipse-plugin-thingworx-extension-development-tutorial>, mais brièvement il va comme ceci. Il faut créer une classe qui étend "Thing" (les bibliothèques de Thing pouvant être trouvées dans le tutoriel).



```
16  private void readDataFromFiles() {
17      StringBuilder sb = new StringBuilder();
18      String line = br.readLine();
19      int currentUnitNumber = -1;
20      int currentUnitNumberCounter = -1;
21
22      DataSnapshotList dataSnapshotList = new DataSnapshotList();
23
24      // Read file with tables
25      while (line != null) {
26          sb.append(line);
27          sb.append(System.LineSeparator());
28      }
29      String[] parts = line.split(" ");
30      TimeCycle timeCycleSnapshot = ConvertSnapshotArrayToObject(parts);
31
32      if (currentUnitNumber != timeCycleSnapshot.unitNumber) {
33          currentUnitNumber = timeCycleSnapshot.unitNumber;
34          currentUnitNumberCounter++;
35          dataSnapshotList.unitNumberList.add(new UnitNumber(currentUnitNumber));
36      }
37      dataSnapshotList.unitNumberList.get(currentUnitNumberCounter).timeInCycleList.add(timeCycleSnapshot);
38      line = br.readLine();
39  }
40
41  String everything = sb.toString();
42
43  for (UnitNumber unitNumber : dataSnapshotList.unitNumberList) {
44      ArrayList<DoubleSummaryStatistics> arrayListOfStatistics = CreateStatisticsArrays();
45      ArrayList<ArrayList<Double>> arrayListOfList = CreateArrays();
46
47      // Add value to output array and statistic summary array
48      for (TimeCycle timeCycle : unitNumber.timeInCycleList) {
49          for (int i = 0; i < arrayListOfStatistics.size(); i++) {
50              arrayListOfStatistics.get(i).accept(timeCycle.data.get(i));
51              arrayListOfList.get(i).add(timeCycle.data.get(i));
52          }
53      }
54      // Get max and min value from statistic array
55      for (int i = 0; i < arrayListOfStatistics.size(); i++) {
56          DoubleSummaryStatistics doubleSummaryStatistics = arrayListOfStatistics.get(i);
57          double min = doubleSummaryStatistics.getMin();
58          double max = doubleSummaryStatistics.getMax();
59      }
60  }
```

Figure 3 Code dans Eclipse pour ThingWorx

Il faut également installer le plugin de Gradle permettant de build un fichier zipper pour l’extension. Gradle peut être trouvé dans la marketplace d’Eclipse. Il faut ensuite créer une build configuration avec Gradle et de la lancer avec le bon “Working Directory” et “Java Home”.

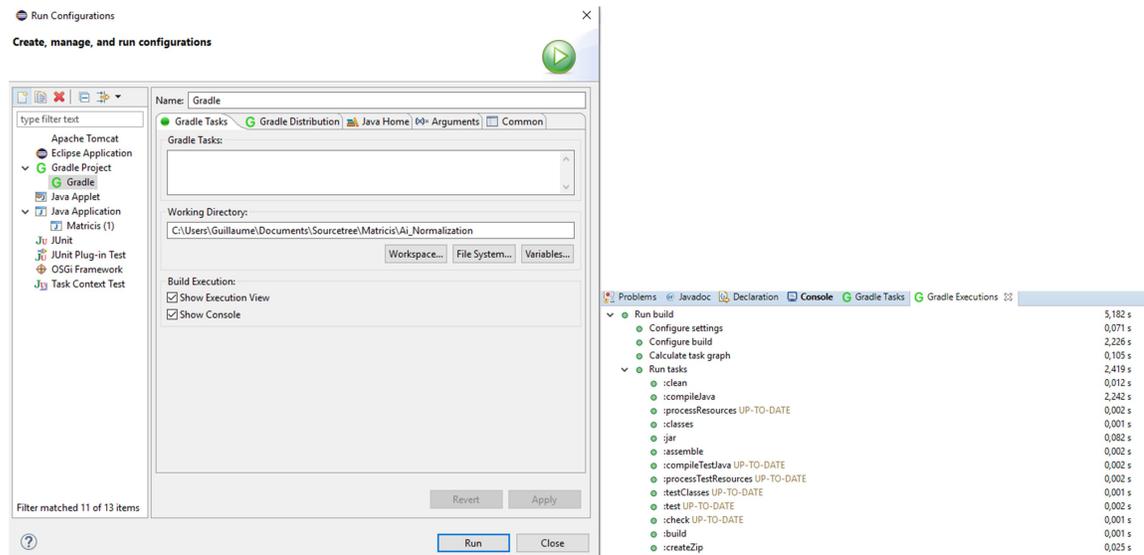


Figure 4 Build Gradle et l’exécution de Gradle

Avec le fichier créé il est maintenant possible de l’importer directement dans le projet ThingWorx comme suit.

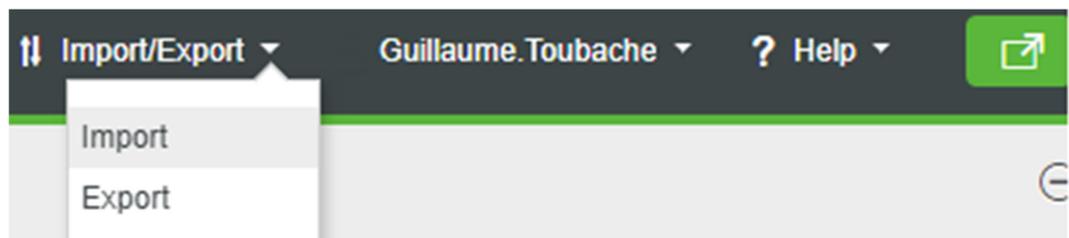


Figure 5 Importation dans ThingWorx



Figure 6 Sélection de type d'importation dans ThingWorx



Figure 7 Sélection du fichier .Zip

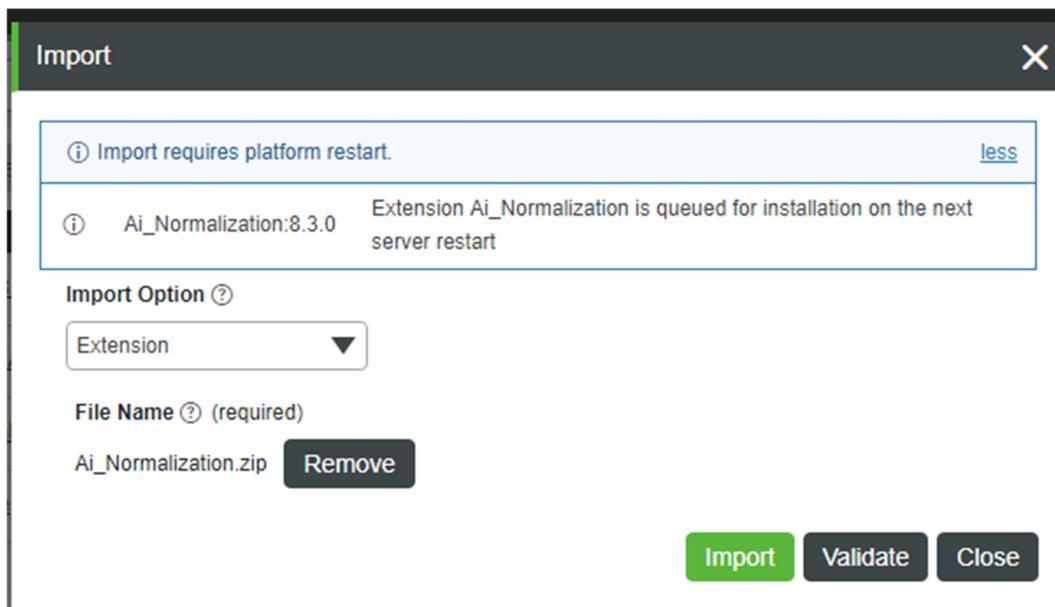


Figure 8 Importation complété

Par la suite il faut mettre la nouvelle entité dans un “Thing” et elle devrait devenir utilisable.

4.2 ThingWorx

Dans ThingWorx, il y a plusieurs modules différents, dans le cas de ce projet les deux qui nous intéressent sont ThingWorx Analytics ainsi que ThingWorx Foundation. Les deux ont été absolument nécessaires à la conception du flux de travail.

4.2.1 ThingWorx Analytics

Thingworx Analytics est le module avec lequel il est possible de créer des modèles ML et les mettre en opération. Ce module comprends deux sous-modules Builder et Manager. Le premier sous-module sert à créer et entraîner un modèle et le deuxième est utilisé pour gérer l'utilisation et l'interaction du modèle avec "Thingworx Foundation".

4.2.1.1 ThingWorx Analytics Builder

Thingworx Analytics builder est le sous-module permettant de créer et d'entraîner des modèles ML. Il y a les choix usuels de type de modèle (Decision tree, neural network, random forest, baysian, etc.) et les choix typiques d'ensemble (bagging, boosting et stacking). C'est avec ce module que l'on importe les données et qu'on les nettoie. Il est également possible de faire une analyse de corrélation entre les données et l'objectif et exclure une primitive qui ne contribue pas à la prédiction de la valeur ciblée. Une fois que le modèle est jugé satisfaisant il peut être activé afin qu'il puisse être traitée par le module Analytics Manager.

4.2.1.2 ThingWorx Analytics Manager

Une fois que le modèle est créé avec Analytics Builder, Analytics Manager connecte le modèle avec un "Thing" à travers un "mapping" de données.

4.2.2 ThingWorx Foundation

Thingworx foundation est l'application principale qui contient la logique nécessaire pour interconnecter les objets IoT. Avec ce module on peut créer des "Things" qui représentent des objets et utiliser des protocoles tel REST pour entrer les données qui caractérisent un état à un certain moment dans le temps. Il existe deux types de Thing dans Foundation, le "direct Thing" et le "remote Thing". Le "remote Thing" est capable de recevoir des données et effectuer un traitement sur celles-ci.

CHAPITRE 5 – RÉSULTATS

Dans cette section, nous allons présenter les résultats obtenues après la conception ce qui va être présenté à Matricis pour expliquer si le projet est réussi ou non.

5.1 Extension Java

L'extension Java était censée donner plus d'options en offrant un module personnalisé permettant de répondre à des besoins spécifiques ne pouvant pas être comblé par ce que ThingWorx contenait déjà. Celle-ci pouvant prendre des entrées venant de ThingWorx Foundation et des sorties dans ThingWorx Analytics. Ce que nous avons conclu était que celle-ci n'était pas vraiment nécessaire pour le projet qu'on avait à faire, les capacités de bases de ThingWorx étaient plus que suffisantes pour répondre de façon satisfaisante aux demandes du projet. Il serait possible que de futurs projets nécessitent l'utilisation d'une telle extension, mais à moins qu'une situation très complexe survienne il y a peu de chance que les capacités déjà existantes de ThingWorx ne puissent pas répondre à la demande. Et dans le cas d'une telle situation les principes de l'utilisation d'une telle extension dans notre projet explorateur offriraient un bon point de départ sur comment exécuter l'implémentation d'une extension Java.

5.2 ThingWorx

La configuration nécessaire au déploiement d'un ML model sur Thingworx s'est effectuée de la manière suivante:

- 1- Le modèle sur Thingworx Analytics fut créé.
- 2- Le Thing responsable de recevoir l'information fut créé.
- 3- La connexion entre ces deux entités fut établie.

A) La première étape de la création du modèle consiste à importer les données d'entraînement qui doivent être dans un format CSV. Dans des versions antérieures il fallait définir les champs qui allaient accueillir les données, mais ce n'est plus le cas avec la version 8. L'option upload metadata doit rester décoché pour que Thingworx détecte automatiquement le type de donnée qui réside dans chaque champ (Int, decimal, string, etc).

Figure 9 Création d'un jeu de données pour importation sur Thingworx Analytics.

- B) Lorsque ces données sont importées dans le système, il est possible de créer des filtres afin que le modèle puisse omettre les entrées invalides qui faussent le résultat final. Les données sont groupées en intervalles et on peut voir le poids d'un groupe sur l'ensemble total de données. Il est important de noter que lorsqu'une exclusion est effectuée sur une valeur c'est toute la rangée contenant la valeur qui est exclue du jeu de données lors de l'entraînement.

Values	# Of Records	% Of Total
46.93 to 47.064	33	0%
47.066 to 47.198	1352	7%
47.2 to 47.3325	3340	16%

Figure 10 Création d'un filtre sur un jeu de données.

- C) Une fois le jeu de données entrée dans le système et le filtre créé (au besoin), le modèle ML peut être créé. Le nouveau modèle se voit assigné un jeu de données. Il faut ensuite désigner un "Goal" ou étiquette et choisir le filtre précédemment créé.

New Predictive Model

Model Name (required):
test

Data Selection | Advanced Model Configuration

Dataset (required):
enginefailuredataset

Goal (required):
RUL

Filter (required):
all_data Create Filter

Filter Details
This filter contains 20,631 rows, representing 100% of all the rows in the dataset.
This filter uses all the rows in this dataset.

Excluded Fields from Model: Exclude Fields

Submit Cancel

Figure 11 Création d'un modèle partie 1.

- D) Thingworx Analytics comporte des options avancées en ce qui concerne la création de modèle. Ces options avancées permettent de choisir spécifiquement un modèle (Decision tree par exemple) ou en combiner plusieurs (Ensemble technique).

New Predictive Model

Model Name (required):
test

Data Selection | **Advanced Model Configuration**

Model Settings | **Sampling Strategy**

Validation Holdout %:
20

Max Fields
25

Redundancy Filter

Only available for Boolean goals.

Learning Techniques

Add Remove Learners

Name	Layer Count	Hidden Unit %	Max Depth	# Of Trees	# Of Iterations
DECISION_TREE			12		
GRADIENT_BOOST					
LINEAR_REGRESSION					
NEURAL_NET	3	0.20			

Ensemble Technique:
Elite Average Average Best

Comparison Metric:
Pearsons RMSE

Reset Configuration

Submit Cancel

Figure 11 Création d'un modèle partie 2.

- E) Une fois le modèle créé, on peut lancer l'entraînement dans l'interface principale de Analytics builder. L'opération prend quelques minutes.



Figure 12 Résultats de l'entraînement

- F) Il est possible d'effectuer une analyse sur les données pour trouver la contribution de chaque primitive au résultat final à travers des signaux (Signals). On peut alors modifier le filtre existant ou créer un autre filtre sur les données.

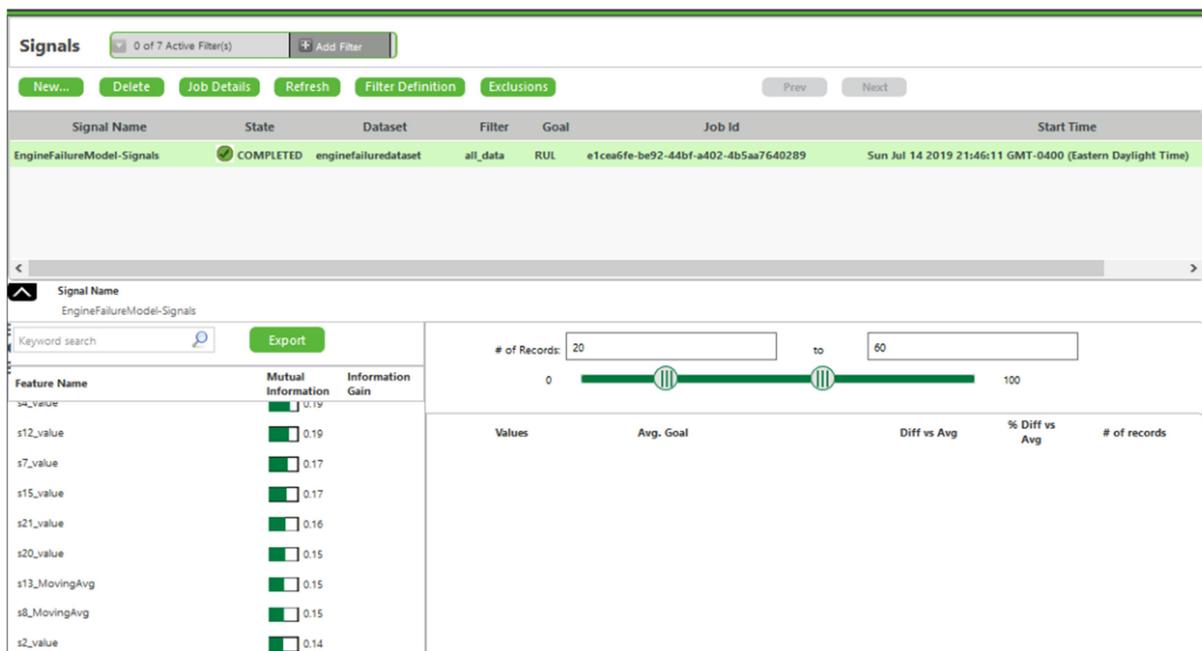


Figure 13 Corrélation.

- G) Lorsque le modèle est satisfaisant il peut être mis comme étant disponible pour qu'il puisse être traité par Analytics Manager en cliquant sur "Enable".

H) On peut maintenant créer un “Thing” représentant la turbine dans Thingworx Foundation. Puisque qu’il est nécessaire de calculer des “moving averages” un “remote Thing template” fut utilisé. On ajouta des propriétés correspondant aux capteurs (s1_value, s2_value, etcetera), des propriétés remote qui calculeront à chaque 60 secondes la moyenne des données pour chaque capteurs, une propriété RUL pour contenir le résultat de l’évaluation, une switch FlipFlop pour déclencher le processus d’évaluation et un datashape qui contient de colonnes identiques aux colonnes du modèle dans Analytics.

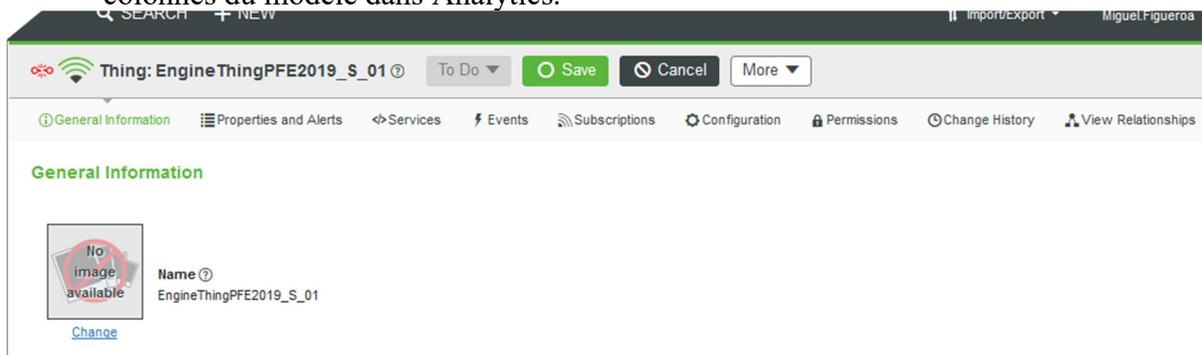


Figure 14 Création du remote Thing

- I) Matricis souhaitait déclencher l’évaluation à une certaine intervalle de temps, pour effectuer la tâche, nous avons créé un Thing avec un Timer template qui avait une fréquence de 60 secondes. Dans le Thing Timer une “subscription” a été créé pour transférer les données des propriétés au datashape et changer la valeur du FlipFlop pour déclencher le calcul.
- J) Jusqu’à ce point nous avons créé un Thing et un modèle, mais aucune connexion n’a été effectué entre les deux. Pour ce faire, il faut aller dans Thingworx Analytics manager sélectionner le modèle et cliquer sur mapping. Les valeurs entrantes sont mappés ainsi que les valeurs résultantes.

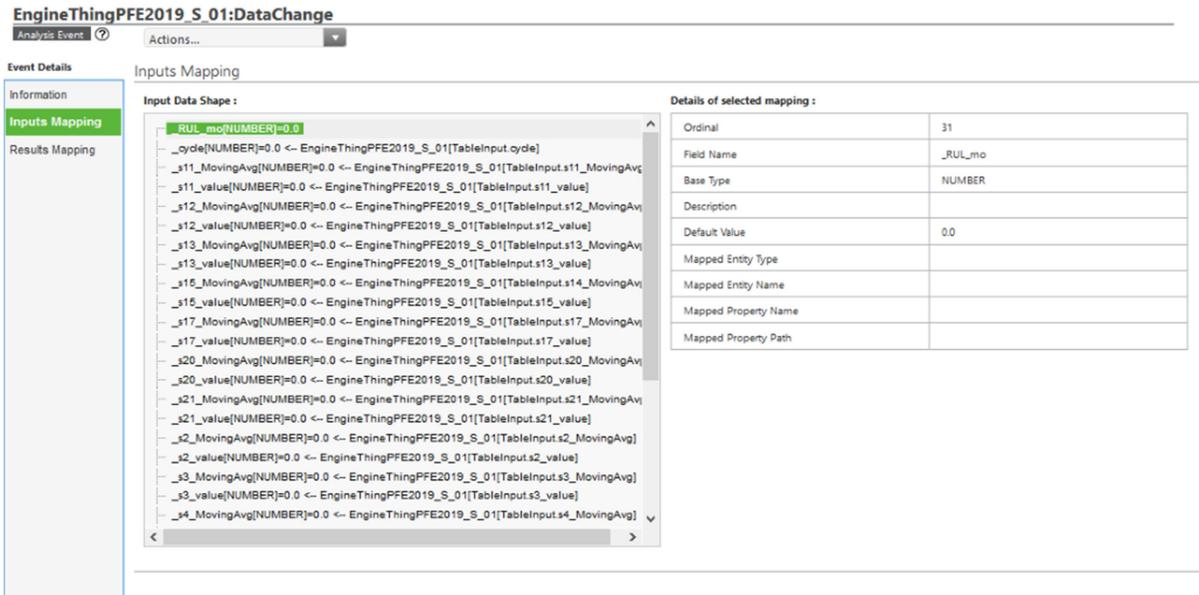


Figure 15 Mapping

K) Le graphique suivant Présente le résultat final de la configuration:

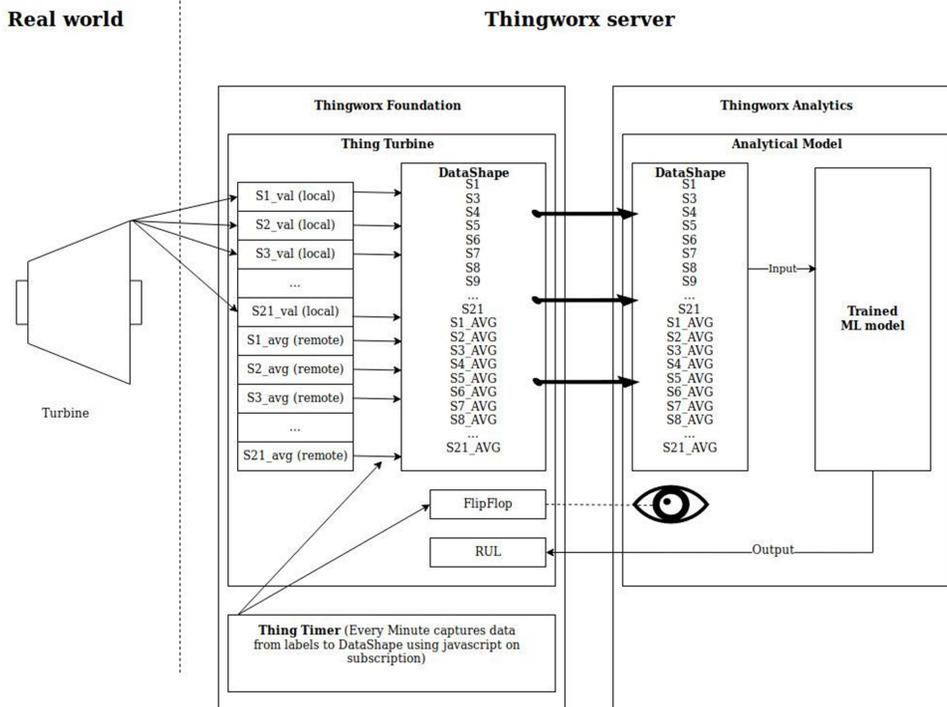


Figure 16 Architecture

La turbine envoie de données aux propriétés du thing S1 à S21.
 Les moving averages sont calculés sur les étiquettes SX_avg à chaque 60 secondes.

Le Thing Timer déclenche un calcul du RUL en copiant les valeurs des propriétés sur le datashape du Thing Turbine. Le timer change également la valeur du FlipFlop à 0 lorsque c'est 1 et 1 lorsque c'est 0.

L'Analytical model surveille la valeur du FlipFlop (défini lors du mapping). Il copie les valeurs du datashape sur son propre datashape afin d'envoyer l'information au modèle entraîné.

Une fois la prédiction effectuée, l'analytical model retourne la valeur du RUL dans la propriété du Thing (défini lors du mapping).

CHAPITRE 6- DIFFICULTÉS RENCONTRÉS

Dans le projet, nous avons rencontré plusieurs problèmes, dont ceux que nous avons considérés comme à risque dans le projet. Il y a trois sections différentes de problèmes rencontrés. Les problèmes de compréhension de projet et les problèmes d'équipe, puis il y a les problèmes rencontrés sur ThingWorx, finalement les problèmes avec Java pour ThingWorx.

6.1 Compréhension du projet et problème d'équipe

Cette section va plutôt parler des problèmes rencontrés dans la compréhension en tant que telle du projet, des problèmes vécus par les membres de l'équipe, telle que les contraintes de temps et les conflits d'horaire.

6.1.1 Description Initial du projet

Lors de la description initiale du projet nous pensions avoir compris ce qu'il fallait faire, mais plus le projet avançait plus on se rendait compte que ce que l'on avait compris n'était pas tout à fait ce qu'il fallait faire. Au tout départ certains membres de l'équipe pensaient qu'il fallait développer un nouveau modèle d'entraînement, puis après la première rencontre avec Matricis, les choses se sont clarifiées, mais tout de même encore un peu flou considérant que le projet en tant que telle était de monter un flux de travail pour démontrer que le projet est réalisable.

6.1.2 Limité sur le temps

Puisque le projet se déroule sur 3 mois, nous avons une contrainte de temps, considérant que chacun des membres de l'équipe à une situation différente. L'un des membres avait deux cours à l'ETS et travaillait 40 heures semaines, un autre travaillait 20 heures semaines, mais avait 4 cours et le dernier ne travaillait pas, mais avait 4 cours à l'école. Donc trouver du temps pour travailler était plus complexe ce qui a fait en sorte que les 3 mois étaient limites pour compléter le projet.

6.1.3 Conflit d'horaire

Comme expliqué à la section 6.1.2 les membres de l'équipe avaient un horaire assez varié ce qui faisait en sorte que les conflits d'horaire étaient très présents, nous nous sommes retrouvé plusieurs fois dans le projet ou les membres de l'équipe travail en parallèle pendant la nuit au lieu de la journée puisque nous étions tous indisponibles durant la journée à des heures différentes. Nous avons tout de même toujours été en mesure de se joindre aux rencontres chez Matricis les lundis ce qui a facilité le transfert d'information entre les membres de l'équipe et Matricis et qui a permis un avancement sur le projet continu.

6.2 ThingWorx

Pour la section ThingWorx, les problèmes rencontrés ici sont plutôt des problèmes techniques et d'accès. Ils sont séparés en 3 sections, les problèmes quant à la compréhension de ThingWorx, la limitation des accès et les problèmes reliés aux Extensions sur ThingWorx.

6.2.1 Compréhension de ThingWorx

Le projet étant de courte durée et l'équipe n'ayant aucune expérience dans ThingWorx, le tout devait être appris pendant le projet. Considérant la limite de ce qui est fourni par ThingWorx et que ThingWorx Analytics est tout de même récent comme module, il n'y a pas grande information que l'on peut trouver qui sera encore à jour et nécessaire lorsque nous faisons nos recherches sur PTC University, un compte par utilisateur nous a été fournies par Matricis. Bien que les informations étaient manquantes, il a tout de même été possible de les surmonter et d'être en mesure de présenté un flux de travail.

6.2.2 Limitation par les accès

Bien que ceci fût réglé assez rapidement de la part de Matricis, le fait est que parfois nous avons des empêchements, par le manque de connaissance sur ThingWorx. Nous avons été en mesure de réellement commencer vers la fin du mois de Juin, et c'est lorsqu'on a commencé à jouer avec ThingWorx Analytics Builder, que nous avons réalisé qu'un autre module nécessaire au projet, soit ThingWorx Analytics Manager. Lorsque découvert, nous avons fait la demande à Matricis pour obtenir le module, qui était disponible, mais nous n'avions juste pas les permissions de le voir. En moins de 4 jours, nous avons réussi à obtenir les accès, ce qui nous a permis de continuer assez rapidement.

6.2.3 Utilisation des Extensions ThingWorx

Lorsque nous avons terminé une partie de l'extension ThingWorx, nous avons essayé de l'importer, lorsque nous avons fait ceci, nous avons eu un "pop-up" pour nous avertir qu'au prochain redémarrage du serveur l'importation va être complétée, puisqu'à chaque soir entre 21h et 22h il y a un redémarrage. La fois d'après lorsque nous voulions continuer, nous avons essayé de créer un "Thing" pour l'associer à notre Extension en Java, mais elle était introuvable.

6.3 Extension Java

Finalement, cette section va présenter les problèmes rencontrés lors du développement de l'extension en Java. Ils sont séparés en deux parties, la première étant les librairies pour ThingWorx et l'autre les librairies de descriptions pour aller chercher les valeurs nécessaires.

6.3.1 Bibliothèques ThingWorx

Un tutoriel avait été suivi tout au long de la phase de rajouter l'extension sur ThingWorx, mais lorsqu'on travaillait sur le code, nous avons rencontré des problèmes tels que les bibliothèques étaient manquantes d'un programmeur à l'autre il y avait des problèmes lors du transfert du code, une fois sur Git, il y a eu d'autres problèmes, mais ceux-ci facilement réglable puisque le fichier ".git" excluait ces fichiers.

6.3.2 Bibliothèques des descriptions

Pour le cas des bibliothèques de descriptions, ceux qui ont été nécessaire pour retrouver la moyenne, le minimum, le maximum et des moyennes mobiles. La difficulté ici, était l'intégration au code existant, puisque nous avons effectué la lecture des fichiers ainsi que l'écriture des fichiers il ne restait qu'à rajouter les bibliothèques de descriptions pour être en mesure de prendre ces valeurs au bon endroit dans le code et puisque nous avons utilisé des Objets pour faciliter la lecture du fichier il fallait prendre ceci en considération lors de l'intégration des bibliothèques ce qui finalement nous a fait modifier le code pour des listes, pour que les descriptions soient plus simples à intégrer.

CHAPITRE 7 - SOLUTIONS

Dans cette section nous allons expliquer la solution fournis à Matricis et quelle type de personne sont plus aptes à travailler sur le projet.

7.1 Solution ThingWorx

La solution proposée est le flux de travail décrit dans la section Résultat 5.2 celui représenté dans la figure **XX**.

Par contre bien que ce flux de travail semble simple après les explications il faut tout de même faire attention puisque ce n'est pas n'importe qui, qui sera en mesure de travailler sur le projet, parce que bien que l'idée soit simple, l'application de ceci demande quelqu'un qui a de l'expérience et les connaissances nécessaires dans l'apprentissage machine.

Pour ce qui est en ai de l'extension le code nécessitera quelques changements pour être sûre de sortir les données dans le format voulu, pour le moment les données sortantes sont dans la console, par contre ce qui serait recherché tournerait plutôt autour d'un fichier du type de JSON.

CHAPITRE 8- ENJEUX ENVIRONNEMENTAUX

8.1 Réduction de l’empreinte écologique

Au cours de ce projet, nous avons évité le plus possible les déplacements en voiture. Nous avons majoritairement utilisé les transports en commun, aussi un des membres de l’équipe prenaient l’occasion du beau temps pour se déplacer en bicyclette. Lorsque nous étions dans la possibilité de le faire, nous participions aux rencontres hebdomadaires par Skype à distance, donc nous éliminions aux maximums tous les effets néfastes que nous causions sur l’environnement. Puisque l’entièreté du projet était sur ordinateur, nous en avons profité pour éviter tout ce qui est papier, les seuls papiers utilisés aux cours du projet sont les documents confidentiels à signer au début du projet.

CHAPITRE 9 - CONCLUSION

En conclusion, ce projet qui demandait des connaissances en apprentissage machine a permis aux membres de l'équipe de développer et mettre en action les connaissances sur un des plus gros du marché, ThingWorx. Le but étant de développer un flux de travail qui n'est disponible nulle part puisque bien que la documentation de ThingWorx existe elle est « théorique », il restait à voir si elle était applicable.

Après plusieurs mois d'effort et d'apprentissage, l'équipe a finalement été en mesure de faire le flux de travail complet et de le présenter grâce à ceci, maintenant la documentation existe à l'intérieur de Matricis et permet à ceux-ci de commencer à dupliquer la méthode de travail. Après avoir rencontrés plusieurs problèmes, nous réalisons que bien la complexité du travail est plus grande que prévue l'important dans un projet comme celui-ci est le fait que si nous sommes bien encadrés pour faire le travail, nous allons être en mesure de réussir le projet.

Ce projet nous a permis d'explorer l'univers de l'industrie 4.0 en marché du travail et de travailler avec plusieurs personnes qui vivent couramment ce mode de vie à tous les jours.

CHAPITRE 10 - LISTE DE RÉFÉRENCES

<https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/>

Données fournis par la NASA.

https://support.ptc.com/help/thingworx_hc/thingworx_analytics_52/#page/thingworx_analytics_7%2Fthingworxanalytics-welcome.html%23

ThingWorx Analytics

<https://www.ptc.com/en/>

<https://support.ptc.com/>

<http://www.ptc.com/Saba/Web/ptcu>

PTC University

<https://community.ptc.com/t5/ThingWorx-Developers/bd-p/twxdevs>

ThingWorx Forum

<http://commons.apache.org/proper/commons-math/>

Librairies Mathématique JAVA

<http://commons.apache.org/proper/commons-math/javadocs/api-3.3/org/apache/commons/math3/stat/descriptive/DescriptiveStatistics.html>

Libraires pour

<https://developer.thingworx.com/resources/guides/eclipse-plugin-thingworx-extension-development-tutorial>

Guide pour ajouter une extension

CHAPITRE 11 - BIBLIOGRAPHIE

<Texte – Style Références bibliographiques déjà activé>

CHAPITRE 12 - APPENDICES

Nous avons fournis le code Java dans un fichier .zip qui peut être utilisé pour l'extension.