**Utilisation des concepts d’industrie 4.0 pour** **l’identification de différents potentiels d'évolution d’un outil d’analyse au sein d’une production automobile.**

Promoteur : M. Fabrice Médol

Maître de stage en entreprise : Mme Salma El Gueddari

JUIN 2022

Travail de fin d’études présenté en vue de l’obtention du diplôme de Bachelier en Informatique de Gestion.

Table des matières

[1. Introduction 3](#_Toc102132694)

[2. Etat de l’art 3](#_Toc102132695)

[2.1. Industrie 4.0 3](#_Toc102132696)

[2.1.1. Evolution 3](#_Toc102132697)

[2.1.2. Définition 4](#_Toc102132698)

[2.1.3. Eléments clés 4](#_Toc102132699)

[2.2. Industrie 4.0 dans une production automobile 5](#_Toc102132700)

[2.2.1. Audi Brussels 5](#_Toc102132701)

[2.3. l’identification de potentiels d'évolution 7](#_Toc102132702)

[2.3.1. Point de départ 7](#_Toc102132703)

[2.3.2. Améliorations identifier 8](#_Toc102132704)

[2.4. Outils existants 10](#_Toc102132705)

[2.4.1. Algorithme Apriori 10](#_Toc102132706)

[2.4.2. Algorithme Expire-Span 11](#_Toc102132707)

[3. Bibliographie 18](#_Toc102132708)

Introduction

# Etat de l’art

## Industrie 4.0

### Evolution

Jusqu’à ce jour différentes révolutions industrielles ont été connu dont chacune qui sont caractériser par des inventions.

La première révolution industrielle a commencé avec l'invention de la machine à vapeur et par la construction de chemins de fer. Quelques temps après la deuxième révolution industrielle ont apporté l’électricité et les premières chaines de montage.

Ensuite les ordinateurs ont vu le jours pendant la troisième révolution grâce à l’invention des semiconducteur et **d’Internet** (Schwab, 2022, p.11-12).

Aujourd’hui on est en train d’assister à la quatrième révolution industrielle qui est souvent désigné étant une continuation de la troisième mais qui s’appuie uniquement sur cette dernière et rapporte des nombreuses avancées technologiques :

* Nanotechnology, virtual and augmented reality, the Internet of Things (IoT) (Guan et al., 2020, p.3).
* Un Internet que comparer à avant sera beaucoup plus présent et mobile, des capteurs qui seront plus petit, plus puissant et moins couteux, machine learning, artificial intelligence (Schwab, 2022, p.11-12).

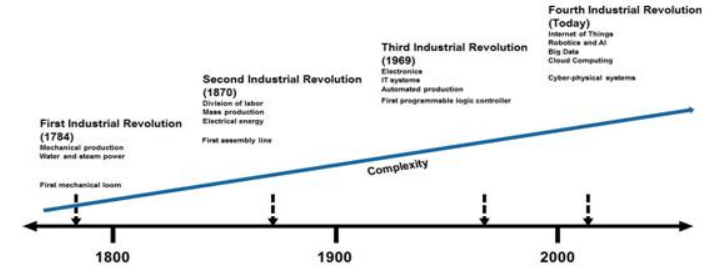


Figure 1 Time Line Chart Industrial Revolution Period (Ashwani, S., & Dr Bikram Jit, S. (2020))

### Définition

Ces nombreuses technologies ont révolutionné le fonctionnement des entreprises par la digitalisation des différents processus de production et de logistique. Cela implique une communication entre les machines et les produits afin de rendre l’organisation des différentes étapes autonome en synergie avec l’humain (Sinay & Kotianova, 2018).

### Eléments clés

**Cyber-Physical Systems**

Ces sont des systèmes intégrés dans un réseau qui ont pour but de surveiller les processus de fabrication afin de rendre les machines plus autonomes. Ces machines vont pouvoir détecter et réparer les différents dysfonctionnements qui peuvent avoir lieu. (Vermeulen & Feys, 2019)

Ces systèmes représentent le lien entre des objets réels (machines) et leur processus de traitement de l'information (Sinay & Kotianova, 2018).

**The Internet of Things**

On rend des objets physiques intelligent en les connectant à internet au moyen de capteurs.

Dans l’industrie automobile on retrouve déjà un exemple avec la voiture connecté qui a l’aide d’internet peut déclencher des appels d’urgence en cas d’accident ou recevoir des avertissements de trafic. (Werr, 2015)

**The Internet of Services**

Avec la croissance des nombres d’appareil connecte à internet on constate que leur utilité est corrompue par la diversité de ces mêmes appareils.

L’Internet des services va simplifier le processus d’intégration de tous ces dispositifs pour pouvoir les utiliser dans leur intégration. (Mukerji, 2018)

**Smart Factories**

Les Smart Factories sont un concept important dans l’industrie 4.0, il s’agit de collecter toutes les données obtenues à la réalisation d’opérations dans tous les secteurs de l’entreprise.

Avec ces données il est possible d’adapter les opérations selon les demandes et prédire des changements. (Vermeulen & Feys, 2019)

## Industrie 4.0 dans une production automobile

Dans le cadre de ce travail on prendra exemple sur Audi Brussels qui selon Sinay J. et Kotianova Z. (2018) serai au deuxième niveau d’implémentation de l’industrie 4.0 (la **digitalisation** des processus).

### Audi Brussels

Audi Brussels s’occupe exclusivement de la production d’E-tron et dans le département assemblage on retrouve une chaîne de production comme dans la plupart des industries automobile.

Cette chaîne de production est séparée en plusieurs « Proofhort », c’est le terme qui est utilisé pour parler des différents emplacements.

Chaque proofhort s’occupe d’une partie du procède d’assemblage du véhicule et ils sont illustré dans la figure suivante.

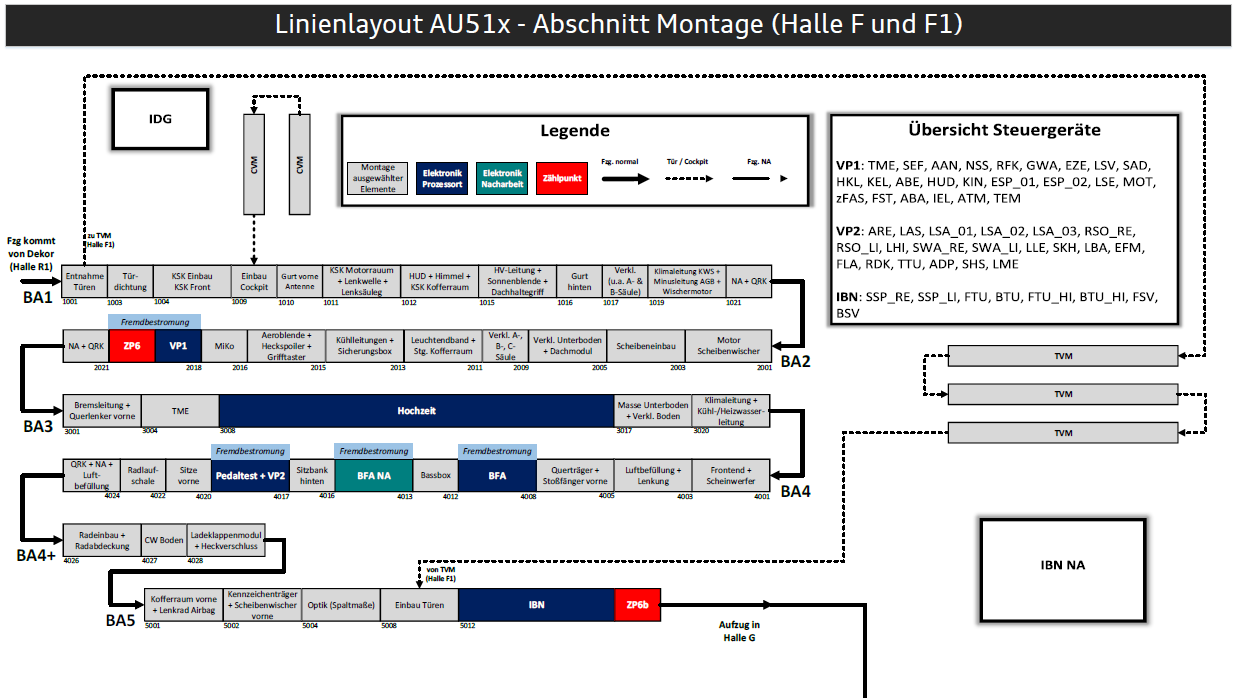


Figure 2 Abschnitt Montage (Audi Brussels.(2022))

C’est dans les rectangles blues qu’on va se concentrer, parce que c’est là où les tests électroniques prennent lieu. Grâce à un appareil qui se connecte avec le véhicule appelé « MFT » des tests sont effectué sur les composants, comme les boitiers électroniques, les batteries et les résultats obtenu sont conserver.

On procède maintenant à l’explication des rectangles blues pour avoir une idée des types de tests qui sont réalisé.

|  |  |
| --- | --- |
| **Proofhort** | **Déroulement** |
| **VP1** | Test des premiers boitiers électronique |
| **Hochzeit(Mariage)** | Une étape très importante qui consiste à assembler le châssis de la voiture avec la batterie principale à haute voltage.  **(Il n’y a pas de test effectués)** |
| **BFA** | Remplissage des liquides : liquide de refroidissement, des freins et de climatisation. |
| **VP2** | Entre VP1 et BFA des nouveaux boitiers électroniques ont pris place, on procède donc à leur test. |
| **IBN** | Ici, la voiture démarre pour la première fois. On test la communication entre tous les composants. |

Une fois que IBN prend fin, les voitures doivent continuer sur une autre chaîne où elles termineront leur procède de production.

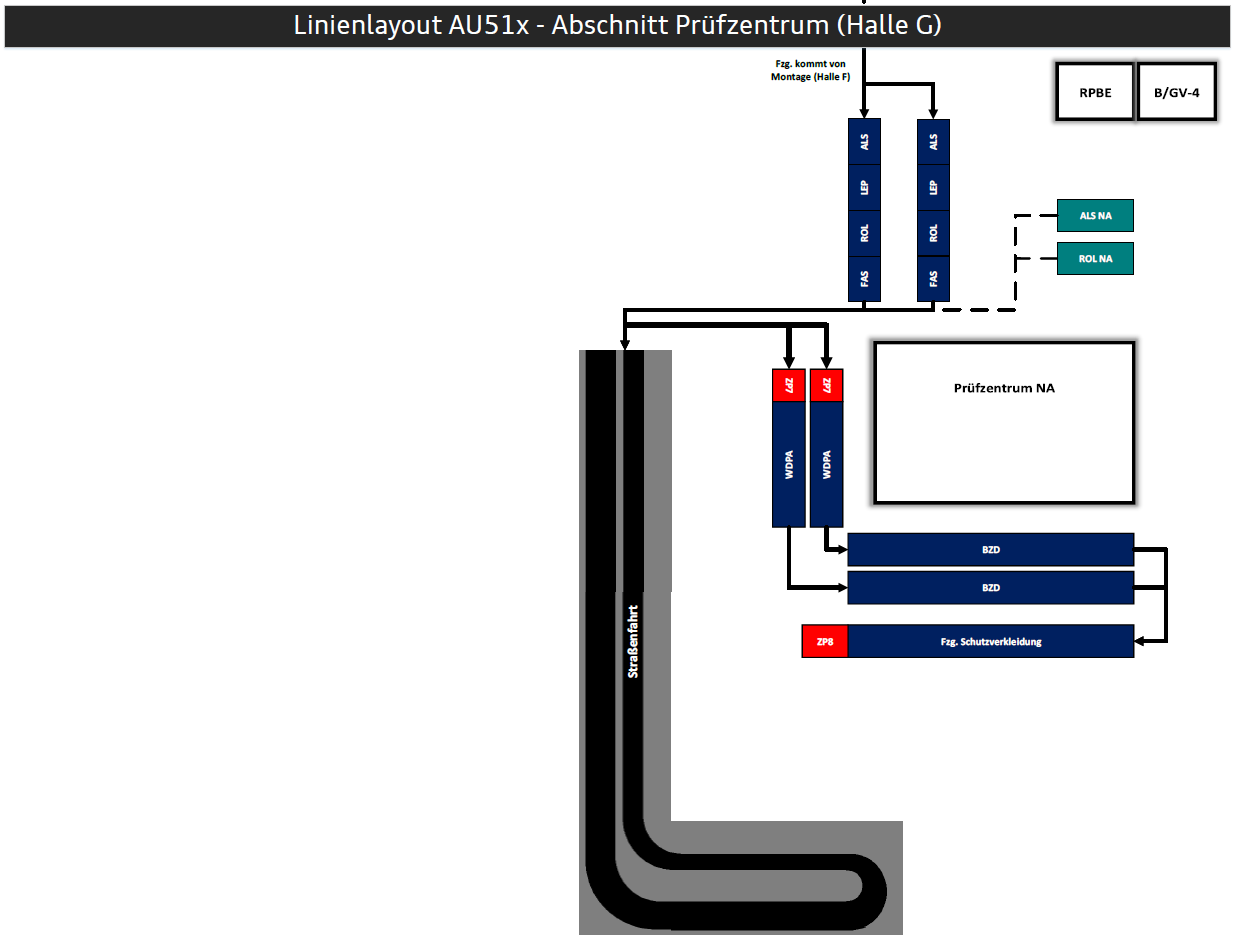


Figure 3 Abschnitt Prüfzentrum (Audi Brussels.(2022))

On retrouve sur cette chaîne d’autres poste de contrôle électronique où le MFT exécute des nouveaux tests.

|  |  |
| --- | --- |
| **Proofhort** | **Déroulement** |
| **ALS** | Calibrage des roues |
| **LEP** | Un des opérateurs braque le volant tout à gauche, puis tout à droite pour définir une valeur MAX et MIN. |
| **ROL** | On passe la voiture sur des rouleau pour tester son comportement |
| **FAS** | Les voitures selon les options possèdent des camera et des capteurs. Il y a donc un procède de calibrage. |

Dans WDPA et Fzg. Schutzverkleidung(Habillage de protection du véhicule) aucun test est réalisé, on contrôle la carrosserie du véhicule, on le mets en mode transport et on le protège avec un revêtement en plastique pour la livraison au client (Audi Brussels , 2022)[[1]](#footnote-1).

## L’identification de potentiels d'évolution

### Point de départ

Dans le cas d’Audi Brussels, comme dans la plupart des industrie automobiles qui suivent le concept de Smart Factory on retrouve une interface de visualisation. Cette interface affichera toutes les données collectées pendant les processus de fabrication.

Pour le moment cet affichage permet de connaître :

* L’état du véhicule

Parmi les tests effectués on peut détecter des erreurs et donc avoir un état général sur sa santé ou un état plus spécifique en se basant uniquement sur certain test.

* L’emplacement du véhicule

En regardant dans quel secteur de la chaine de production le dernier test a été fait, on peut localiser le véhicule de façon assez précise.

* Les erreurs les plus courantes

En calculant le nombre de fois qu’une erreur apparait sur l’ensembles des erreurs détectées on peut distinguer les plus courantes et les afficher par ordre décroissant.

La fonctionnalité déjà existante constitue un bon outil d’analyse qui sert de support pour la prévention, la détection et solution de problèmes.

Mais dans le secteur automobile il est impératif d’être constamment en évolutions afin de pouvoir réduire le délai de fabrication et augmenter le rendement.

Sur base d’un nombre de données consistant, de nombreuses améliorations peuvent être implémenter.

### Améliorations identifier

La solution de ce travail se basera sur les erreurs issues des tests de véhicules qui ont eu un résultat négatif. On a constaté avec les opérateurs de la chaîne de production qu’il serai intéressant d’avoir **des causes** relier à ces défauts. Cela devrait rendre la détection et la solution du problème plus efficace.

#### Collection de données

Pour chaque test négatif solutionné l’opérateur doit documenter son procédé de réparation. Tout d’abord il doit choisir les symptômes cette à dire les erreurs qui sont apparu dans le test. Cette étape est importante parce que la solution que le fonctionnaire va introduire peut résoudre plusieurs symptômes ou un seul.

Exemple :

L’opérateur reçoit un test négatif :

- Numéro de voiture

- Symptômes :

* Lampe pour feu arrière gauche
* Lampe pour feu arrière droit
* Airbag latéral arrière côté passager

Dans ce cas si les deux lampes sont réparées en une seul fois alors il doit choisir les deux premiers symptômes.

Ensuite il établira une description qui résumera le(s) moyen(s) utilisé.

Comme dans l’exemple il peut y avoir différents problèmes pour un même test (Airbag et les feux), dans ce cas plusieurs procédés vont être documenter.

Les données collecter grâce à ce système vont suivre le schéma suivant :

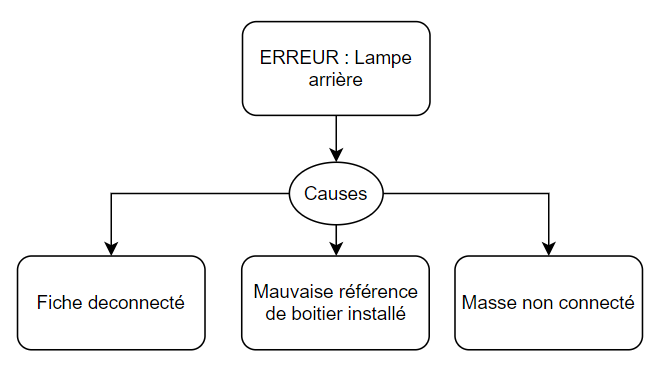


Figure 4 Schéma des causes relier à une erreur

On doit pouvoir proposer des causes différentes pour chaque erreur ayant déjà été présente au paravent pendant les tests effectués sur la chaîne de production. Comme sur la figure 4 où l’erreur « Lampe arrière » est relier à trois causes, cela veut dire que cette même erreur peut être causé par trois choses différentes.

#### Traitement de données

Dans un deuxième temps les taux de possibilité vont aider le fonctionnaire à choisir la cause la plus probable. Cela sera possible grâce des algorithmes spécialiser dans le traitement de données qui vont être aborder dans un prochain chapitre.

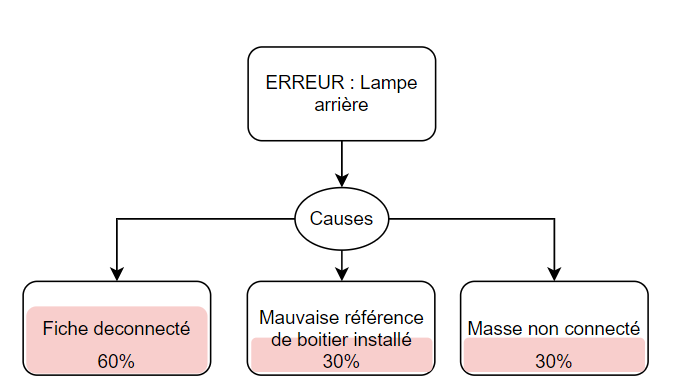
****

Figure 5 Schéma des causes relier à une erreur avec des fréquences

On constate sur la figure 5 qu’une des causes a un taux plus élevé que les autres cela veut dire qu’il y a plus de chance de résoudre l’erreur en se basant sur cette dernière.

## Outils existants

Concernant la partie traitement de données des algorithmes seront utilisé, il est donc nécessaire de choisir les plus approprié pour notre cas. Durant ce chapitre une explication de plusieurs procédés ainsi que leurs avantages et inconvénients seront présenté.

Pour finir une comparaison sera faite afin de conclure un choix final qui sera implémenter dans la solution de ce travail.

### Algorithme Apriori

**Définition**

C’est un algorithme qui générer des règles d’associations booléennes en se basant sur un ensemble d’éléments fréquents. (Chee et al., 2018)

**Explication**

Dans un premier temps il faut définir un nombre de support minimal qui va éliminer pour chaque étape les ensembles dont leur nombre d’apparition est inférieur à ce support.

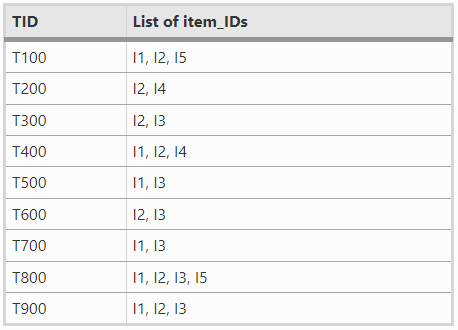


Figure 6 Echantillon de données (Chee,C. et al., (2018))

Ensuite en parcourant la base de données comme celle de la figure 6, il faut repérer tous les ensembles de taille un et compter leur nombre d’apparition pour former une première table (k). On filtre cette table à l’aide du support minimal et avec les lignes qui reste on doit être capable de crée des ensembles de taille deux (k+1).

Les prochaines étapes vont être une répétition de la première jusqu’à ce que ça ne soit plus possible à cause du support minimal.

**Avantages**

C’est un algorithme simple à comprendre et facile à utiliser.

Les résultats obtenus sont également faciles à interpréter.

L’algorithme fonctionne sur des base de données non classé (Big Data)Expliquer, ainsi il peut être utiliser dans de nombreux cas puisque la plupart des donnes sont non classé. (Sosnovshchenko, 2018)

**Inconvénients**

Il est nécessaire de parcourir constamment la base de données et vérifier la fréquence d’apparition des ensembles pour passer aux étapes suivantes.

Selon le contenu de la base de données on risque d’avoir un trop grand nombre d’ensembles d'éléments fréquents. (Chee et al., 2018)

### Algorithme FP-Growth

**Définition**

C’est également un algorithme qui a comme but de trouver les ensembles fréquents. Il a comme spécialité d’utiliser un arbre qui va permettre de scanner uniquement deux fois la base de données. (Zhang et al., 2008)

**Explication**

De même que l’algorithme Apriori il faut établir un support minimal et la première étape consiste à regrouper les différentes éléments présents dans la base de données en comptant leur nombre d’apparition sans oublier de les trier par ordre décroissant.

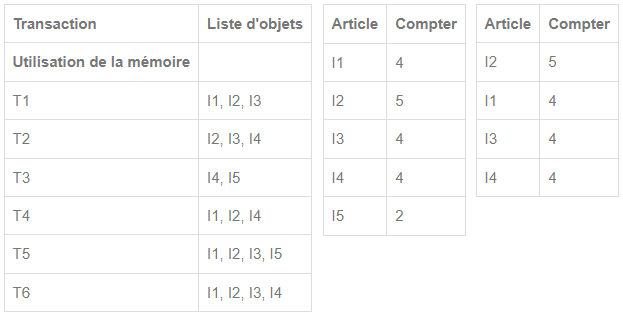
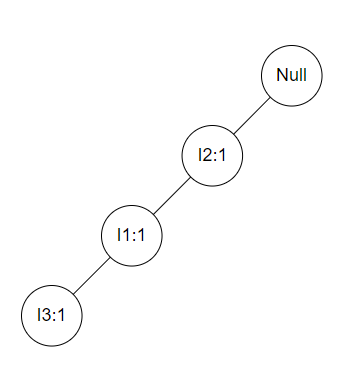


Figure 7 Base de données

<https://fr.myservername.com/frequent-pattern-growth-algorithm-data-mining#:~:text=L'algorithme%20de%20croissance%20FP%20repr%C3%A9sente%20la%20base%20de%20donn%C3%A9es,en%20utilisant%20un%20%C3%A9l%C3%A9ment%20fr%C3%A9quent>.

REFAIRE TOI-MEME LES TABLEAU FIGURE 7

Ensuite on parcoure une deuxième fois la base de données pour crée l’arbre FP (Frequent Pattern). Chaque transaction doit être ordonné selon le tableau qui a été réaliser à la première étape, par exemple la transaction I1,I2 et I3 va devenir I2,I1 et I3. La racine est égale à « Null » pour la première transaction I2 est relié à la racine, I1 est relié à I2 et I3 est relié à I1.



1

Figure 8 Première étape Arbre

L’élément est toujours suivi par son nombre d’apparition (1).

Dans la deuxième transaction I2 apparait à nouveau dans ce cas il suffit d’incrémenter le nombre d’apparition de l’élément. Contrairement à la première transaction où I3 était lié à I1 ici il est lié à I2, c’est pour cela qu’une nouvelle branche doit être rajouter suivi par I4.

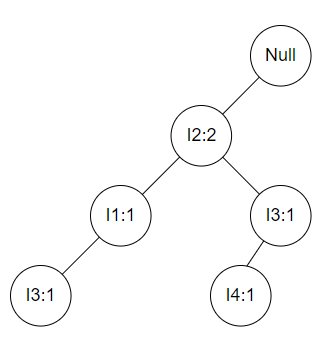
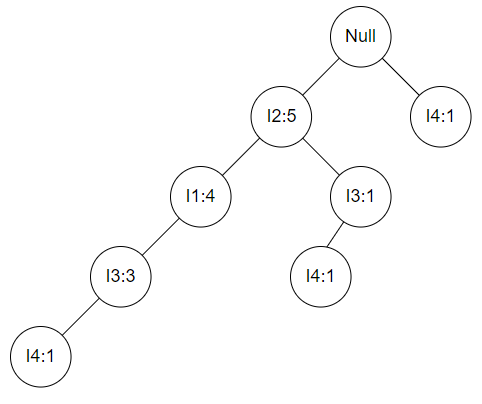


Figure 9 Deuxième étape Arbre

Une fois toutes les transactions parcours l’arbre final doit ressembler à cela.



1

Figure 10 Arbre finale

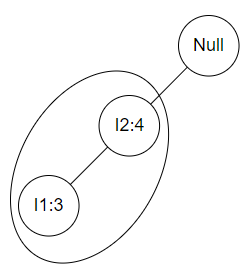
La deuxième partie de l’algorithme FP-Growth permet de trouver les ensembles fréquents.

Si un élément est isolé les chemins qui permettent de l’atteindre depuis la racine sont des transactions par exemple avec I3 les chemins sont :

* I2, I1
* I2, I1
* I2, I1
* I2

La transaction I2, I1 est répète trois fois en relation avec le nombre d’apparition de l’élément (1).

Un autre arbre FP est construit à partir de ces nouvelles transactions en faisant attention aux étapes de la première partie de l’algorithme.



3

2

1

Figure 11 Deuxième arbre FP

Avec la combinaison de l’élément isoler (I3) et les nœud obtenu les ensembles fréquents sont créer :

{I2, I3: 4} (1), {I1: I3: 3}(2), {I2, I1, I3: 3}(3)

**Avantages**

Contrairement à l’algorithme Apriori la combinaison des éléments n’est pas réaliser, ce qui le rend plus rapide. De plus il est nécessaire de parcourir uniquement deux fois la base de données ceci augmente également les performances. (Frequent Pattern (FP) Growth Algorithm In Data Mining, 2022 ; Zhang et al., 2008)

**Inconvénients**

L’arbre FP est complexe à être construit. Une partie du temps gagner en parcourant seulement deux fois la base de données est perdue en construisant ce type d’algorithme. De plus, il n’est pas très efficace voir même non fonctionnel avec de grande base de données. (Frequent Pattern (FP) Growth Algorithm In Data Mining, 2022; Kavitha & T.Tamil Selvi, 2016)

## Comparaison algorithmes

Dans le but de choisir l’algorithmes le plus adapté pour la solution de ce travail, il est fondamental de réaliser un tableau de comparaison contenant différents critères dont : nombre de fois parcourir base de donnes, temps et technique.

|  |  |  |
| --- | --- | --- |
|  | Algorithme Apriori | Algorithme FP-Growth |
| Nombre de parcours | On parcourt la base de données constamment pour vérifier le nombre d’apparition des nouveaux groupes d’éléments. | La base de données est parcourue uniquement deux fois. Une fois pour connaître le nombre d’apparition des éléments et une deuxième fois pour générer l’arbre FP. |
| Technique | Apriori fonctionne avec des groupes d’éléments qui sont initialement à un. Au fur et à mesure des étapes ces groupes s’agrandissent, jusqu’à ce que le support minimal arrête le procédé. | Pour trouver les ensembles fréquents on combine un élément isolé avec les nœuds de son arbre qui a été réalisé au préalable. |
| Complexité | Grâce à sa technique, l’algorithme est simple d’utilisation et d’intégration. | L’arbre FP est complexe à implémenter. |
| Temps | Le temps d’exécution est supérieur à celui de l’algorithme FP-Growt, à cause de la formation des groupes d’éléments. | Le temps d’exécution est inférieur à celui d’Apriori parce qu'il ne fonctionne pas avec des éléments groupés. |
| Taille | Il est efficace avec des larges bases de données et non classé. | Il n’est pas très efficace voir même non fonctionnel avec des larges bases de données. |

Tableau 1 Comparaison Apriori – FP Growth

En ce qui concerne la solution de ce travail, l’algorithme le plus approprié est Apriori. La base de données qui sera traité provient des nombreux tests qui ont déjà été effectué sur la chaine de production chez Audi Brussels. Cette dernière représente une grande quantité de données qui ne sont pas classé et comme le tableau 1 le démontre Apriori est efficace dans ce cas.

D’autre part cette solution sera mise à disposition pour les opérateurs c’est pour cela que la simplicité d’intégration que l’algorithme Apriori offre permettra d’avoir rapidement des résultats. A reformuler ou supprimer.

## Récapitulatif

# Solution

La solution comportera deux 4(analyse,implementation,limitation,analyse de résultats) parties, dans un premier temps les fonctionnalités du système pour collecter les donnes seront représentées sous forme de diagramme (2.1 Analyse). Ensuite, le développement de ces fonctionnalités va être détaillé dans l’implémentation suivie par l’algorithme pour le traitement des données (2.2 Implémentation).

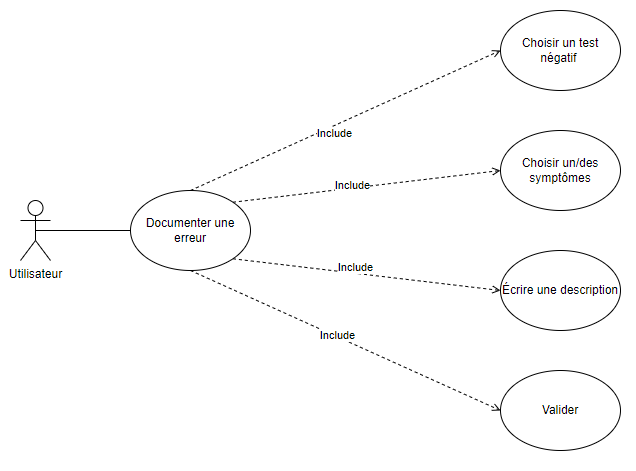
## Analyse

La section analyse comportera des diagrammes qui serviront à illustrer les différentes fonctionnalités du système de collecte de données ainsi que l’interaction de l’utilisateur avec ce dernier.

### Diagramme cas d’utilisation

**Documenter une erreur**

2



5

3

4

1

Figure 12 Diagramme cas d'utilisation - Documenter une erreur

Lorsqu’un utilisateur souhaite documenter une erreur, il doit suivre différentes étapes (1). Premièrement, il doit choisir parmi tous les tests négatifs celui concerné (2), ensuite il faut qu’il précise les symptômes dont il a apporté une solution (3). En deuxième lieu, la procédure de réparation doit être notée dans la description (4) et finalement l’utilisateur valide ses insertions (5).

**Modifier une documentation**

## Implémentation

### Partie 1 : Système de collecte de données

**Technologie et langages utilisé**

En ce qui concerne l’implémentation du système de collecte de données le langage utilisé est Javascript accompagné par le Framework AngularJS.

C’est un Framework front-end open-source développer par Google. source

AngularJS propose une extension à l’HTML classique qui permet la création de pages web dynamiques. De plus il possède les avantages d’être facile à apprendre et simple à intégrer au sein d’un projet. (AngularJS — Superheroic JavaScript MVW Framework, s. d.)

De plus ce Framework il est compatible avec de nombreuses libraires dont Bootstrap, qui a été choisi pour le visuel de ce système.

La librairie Bootstrap a été utilisé, pour sa simplicité d’installation et d’utilisation. De plus elle propose beaucoup de composants comme des tableau, des buttons et différents types de champ. (Bacinger, 2015)

Ce système de collecte de données a été implémenter au sein d’une application interne à Audi Brussels, développer également avec AngularJS, c’est qui justifie l’utilisation de cette technologie.

Le choix du langage ou du Framework n’a pas une très grande importance pour cette partie de l’implémentation. En effet la fonction principale de ce système est de collecté des données.

**Les différentes parties du système de collecte de données**

Sous ce point une explication du code ainsi qu’une visualisation des différentes parties du système de collecte de données vont être proposer.

1. **Liste des symptômes**

Une étape importante pour la documentation d’une erreur est le choix d’un ou des symptômes. Pour permettre à l’opérateur de faire ce choix, une récupération de ces symptômes doit avoir lieu.

Une image contenant texte, horloge, montre

Description générée automatiquement

Figure 13 Schéma récupération symptômes

Sur la figure 13, on peut voir les différentes étapes du procédé de récupération des symptômes. Premièrement, un opérateur sur la chaîne de montage choisi un test négatif (1). Ensuite parmi les résultats du test, on récupère ceux qui ont échoué pour établir une liste de symptômes (2).

La procédure de récupération des résultats du test est déclenchée automatiquement par le click de l’opérateur sur le test négatif.

Voici le code qui a permis cette procédure :

Une image contenant texte, moniteur, écran, intérieur

Description générée automatiquement

Figure 14 Code simplifié procédure – symptômes

La méthode « GetSymptomes » reçoit en paramètre un test négatif (1). Ensuite il est nécessaire de parcourir les résultats de ce test (4). Un résultat contient plusieurs champs dont le code qui va nous permettre d’exclure les symptômes qui n’ont pas échoué (5). Pour finir les symptômes sont introduit dans la liste (6) qui est rendu accessible à l’HTML grâce au mot clé « scope » (10).

Par la suite, ce que l’opérateur verra c’est la liste des symptômes sous forme d’un tableau (composant Bootstrap) :

Une image contenant texte, moniteur, écran, capture d’écran

Description générée automatiquement

Figure 15 Code source simplifié - tableau symptômes

Comme expliquer plus haut, à l’aide du « scope » l’HTML peut avoir accès à la liste des symptômes (1). Cette liste sera parcourue en utilisant « ng-repeat » qui créera dynamiquement les lignes du tableau (2). Pour pouvoir récupérer la sélection de l’opérateur, une variable est initialisée à « false » (3). Selon si la case est cocher ou non cette variable changera de valeur (4).

Le rendu final ressemble à cela :

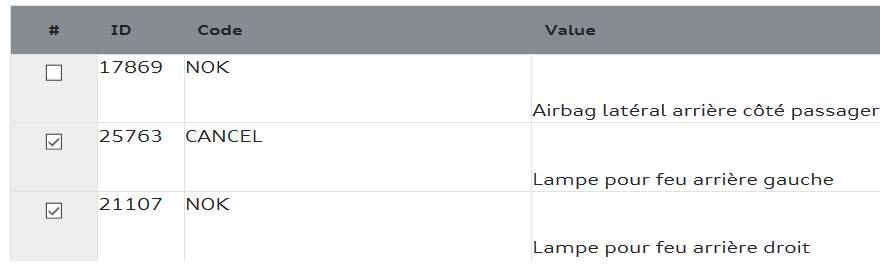


Figure 16 Screenshot tableau symptômes - données fictives

1. **Champ description**

Quant à la description de l’erreur, le type de champ qui a été choisi c’est un « textarea ». C’est un choix qui a été pris par manque de temps, en conséquence ce n’est pas la solution la plus adapté.

Ce type de champ est trop libre, cette à dire que l’utilisateur n’a pas de restrictions concernant le texte introduit. Ceci peut provoquer dans le cas de cette solution une mauvaise compréhension des données.

Par exemple, si un opérateur introduit « câble de la lampe arrière déconnecté » et un autre opérateur pour la même erreur il introduit « câble de la lumière arrière déconnecté ». Ces deux phrases veulent dire la même chose mais dans le cadre d’une base de données se sont deux entrées différentes.

Le même scenario peut se reproduire, si un opérateur écrit une description d’une erreur en allemand et un autre opérateur écrit la même description en français.

Le visuel ressemble à cela :



Figure 17 Screenshot champ description

### Partie 2 : Traitement de donnés

**Technologie et langages utilisé**

Le langage utilisé pour cette partie de l’implémentation est Python avec l’aide de deux libraires Pandas et Mlxtend.

Python est un langage très connu pour les data scientists, il est réputé pour sa simplicité d'utilisation et ses bibliothèques mathématiques intégrées qui offrent une grande puissance de calcul. (Sinha, 2022)

Il a également fait ses preuves auprès de Facebook et de Khan Academy, qui l'utilisent depuis longtemps dans leur environnement. (Vyas, 2022)

**Pandas**

Pandas est une librairie open source Python, qui offre plein d’outils pour lire, analyser, manipuler et stocker des donnés. (What Is Pandas in Python ? Everything You Need to Know, 2021)

L'outil utilisé dans cette solution est le « DataFrame », qui servira à créer les différents tableaux nécessaires et à filtrer les données pertinentes.

**Mlxtend**

La librairie Mlxtend (open source, Python) est très utile pour la science des données, elle offre des outils pour le machine learning, qui peuvent être consultés sur cette page <http://rasbt.github.io/mlxtend/>. (Raschka, s. d.)

Puisque l'algorithme Apriori va être implémenté, la fonction apriori et la fonction association\_rules que mlxtend offre seront très bénéfique.

Ces deux librairies ont déjà été utilisées pour réaliser un projet dans le cadre du cours de base de données de M. Riggio.

**Explication différentes étapes de l’algorithme apriori**

Cette section explique les différentes étapes de l'implémentation de l'algorithme Apriori à l'aide d'un diagramme. Chaque point du diagramme est expliqué et illustré.

*Schéma like Medol*

1. Clean des donnés

Si les données récolter suivent déjà le schéma représenter plus haut faire le lien alors cette étape n’est pas nécessaire.

En raison de contraintes de temps, le système de collecte n'a pas pu être mis à la disposition des opérateurs. Cependant, les données analysées proviennent d'un fichier JSON fourni par Audi Brussels, contenant des procédures d'erreurs provenant d'un ancien système interne.

Cette étape se charge de supprimer les propriétés du fichier JSON, qui ne sont pas importantes dans le contexte de cette solution.

**

Figure 18 Suppression des propriétés non nécessaires

1. Transformation en liste python (transactions Apriori)

Comme expliqué dans le point 1.4.1 faire le lien, l'algorithme d'Apriori nécessite un système de transactions comme base de données initiale.

Cette étape va transformer chaque objet du JSON (du point 1) en une liste python où chaque élément est associé à une transaction.

Une image contenant texte

Description générée automatiquement

Figure 19 Transformation JSON – liste

La propriété "Description" est hachée pour éviter des affichages trop grands (1).

1. Transformation en tableau binaire (true,false) pour la librarie mlx grâce à pandas

La dernière étape de transformation est nécessaire pour que la méthode apriori fonctionne.

Le principe est de former un tableau binaire, où chaque colonne est un élément des transactions vues au point 2. Ensuite, chaque ligne contient la valeur vrai ou faux en fonction de l'apparition de ces mêmes éléments au sein des transactions.

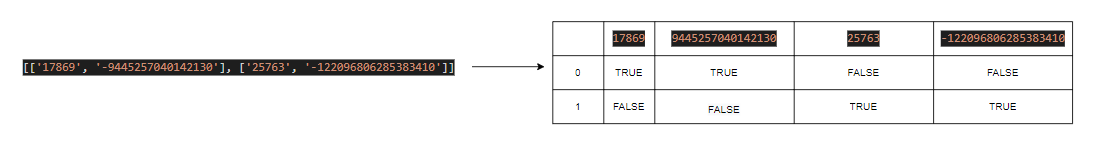


Figure 20 Transformation liste - tableau binaire

1. Utilisation de la méthode apriori de la librairie mlx

Cette méthode a comme but de trouver les ensembles fréquents, cette a dire qu’en parcourant le tableau binaire et en comparant les colonnes elle pourra savoir qu’elle couple d’éléments apparaissent ensemble.

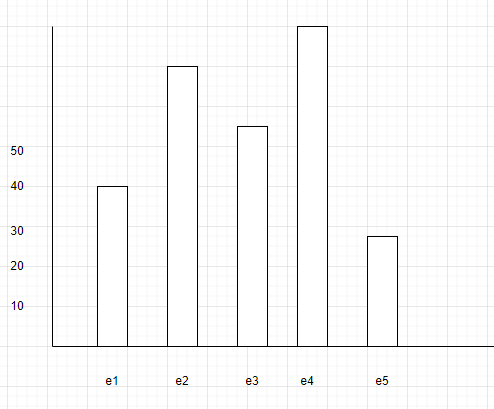
1. Utilisation de la méthode association rules de la librarie mlx

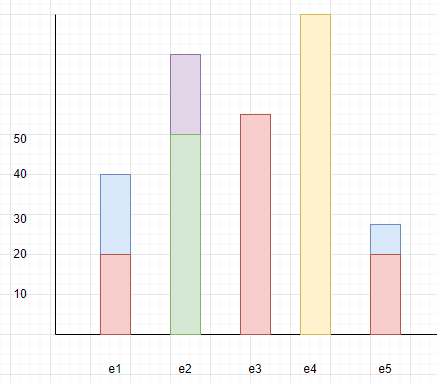
Pour finir, des règles d’associations sont définit pour les ensembles fréquents. Ces associations permettent de calculer la confidence qui suit la formule suivante :

Confidence(Symptôme -> Description) = support(Symptôme, Description) / support(Symptôme)

Cette valeur sera la fréquence d’apparition d’une solution par rapport a une erreur.

## Analyse de résultats





# Bibliographie

Audi Brussels. (2022). Au51x Linienlayout Montage & Prüfzentrum

Chee, C., Jaafar, J., Aziz, I., Hasan, M., & Yeoh, W. (2018, 24 mars). *Algorithms for frequent itemset mining : a literature review*. SpringerLink. https://link.springer.com/article/10.1007/s10462-018-9629-z?error=cookies\_not\_supported&code=22d68a2a-36e1-4063-852c-72b951946782

Dr Bikram Jit, S., & Ashwani, S. (2020, 1 septembre). *Evolution of industrial revolutions : A review*. ResearchGate. https://www.researchgate.net/publication/344366036\_Evolution\_of\_Industrial\_Revolutions\_A\_Review

Guan, C., Jiang, Z., & Ding, D. (2020). *Emerging Business Models, The (Singapore University of Social Sciences - World Scientific F)*. WSPC.

Kotianova,Z., & Sinay,J. (2018, septembre). *Automotive industry in the context of industry 4.0 strategy*. ResearchGate. https://www.researchgate.net/publication/331507570\_Automotive\_Industry\_in\_the\_Context\_of\_Industry\_40\_Strategy

Mukerji, D. (2018, 27 mars). *Industry 4.0 Defined : 4 Core Components*. Automate. https://www.automate.org/editorials/industry-4-0-defined-4-core-components

Schwab, K. (2022). *Fourth Industrial Revolution*. PENGUIN GROUP.

Sosnovshchenko, A. (2018). Machine Learning with Swift : Artificial Intelligence for iOS. Packt Publishing.

Vermeulen, T., & Feys, J. (2019, décembre). *Industry 4.0 in the automotive sector of spain*. Flanders Investment & Trade – MADRID. https://www.flandersinvestmentandtrade.com/export/sites/trade/files/market\_studies/2019-Spanje-Industry%204.o%20in%20the%20automotive%20sector%20of%20Spain%20website.pdf

Werr,P. (2015, 9 avril). *How Industry 4.0 and the Internet of Things are connected*. Iotevolutionworld. https://www.iotevolutionworld.com/m2m/articles/401292-how-industry-40-the-internet-things-connected.htm

1. Source de l’intranet d’Audi Brussels (non accessible au public). [↑](#footnote-ref-1)