

# LIVRABLES R3 THEORIGINALS

## SPECIFICATIONS FONCTIONNELLES

## I-Yuxin SHI

## I .Présentation

Réaliser : Analyser des indicateurs relatifs à l'activité de l'atelier

Brief: Destiné au chef du dépôt.

Celui-ci peut calculer et afficher de nombreux indicateurs tels que le nombre d'opérations, leur durée moyenne, le nombre de pièces consommées, etc. Il pourra les décliner dynamiquement suivant plusieurs axes combinables: par type de véhicule, par manutentionnaire, par période de temps (semaine, mois, année), etc.

Exemples: \* nombre total de réparations pour crevaison effectuées au cours du mois passé, sur des vélos en libre service \* évolution semaine par semaine sur l'année en cours du nombre total d'opérations effectuées au sein du dépôt 2

## II. Spécification fonctionnelle

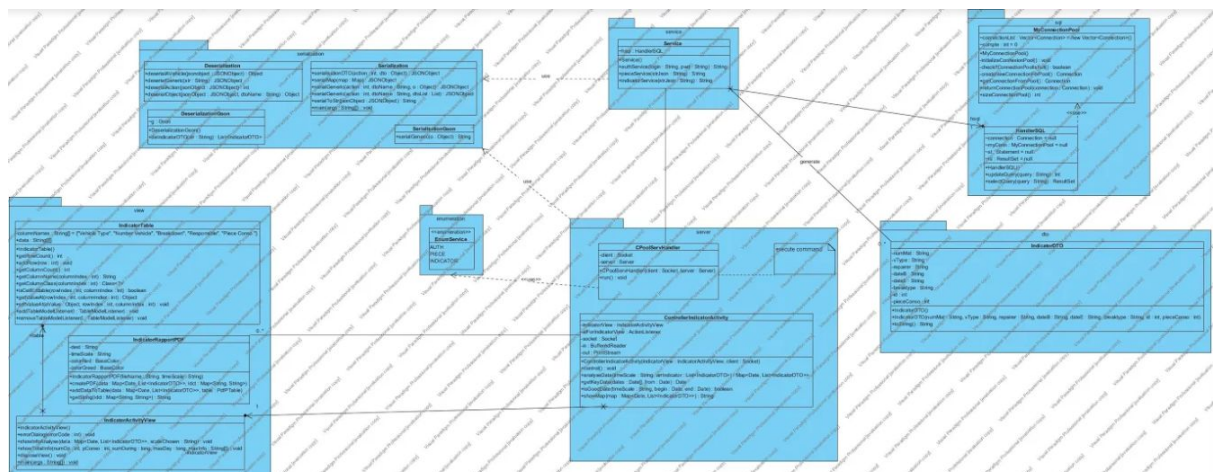
## 1-Fonctionnalité

Ma partie est destiné au chef du dépôt, elle permet de : - rechercher les opérations effectuées en affichant en mode de table - analyser les résultats récupérés par semaine, mois ou an - exporter les résultats analysés en format pdf

### III. Diagramme de classes

Ce diagramme est grand, donc je met le lien vers l'image originale:

[https://drive.google.com/open?id=0B06xT8k\\_M7qsVWqzZ25hU2IzZUE](https://drive.google.com/open?id=0B06xT8k_M7qsVWqzZ25hU2IzZUE)



## II-Bakari DEMBELE

### SPÉCIFICATIONS FONCTIONNELLES

Cette logicielle a pour objectifs d'enregistrer un véhicule à l'entrée du dépôt pour réparation.


Lorsqu'un véhicule entre au dépôt pour un motif précis (panne, simple défaut, révision, ...) il est pris en charge par un manutentionnaire. Celui-ci entre dans le système les informations fondamentales relatives à cette entrée (motif, détails sur l'état du véhicule, etc.).

Le système peut calculer la durée prévisionnelle totale des opérations de maintenance à réaliser et indiquer à quelle place il faut stocker le véhicule en attendant la réparation.

Le véhicule entre alors dans une file d'attente.

Un manutentionnaire avec le véhicule devant lui utilisera une information qui permet d'identifier ce véhicule parmi tant d'autres (un numéro matricule par exemple). Il listera tous les motifs et pannes constatés sur ce véhicule et les enregistre dans la base de données.

Information du véhicule



Matricule  Chercher

Modele

Marque

Type vehicule

Motif de visite et panne constatée

Motif visite  charger Ajouter

LISTE DE MOTIFS:

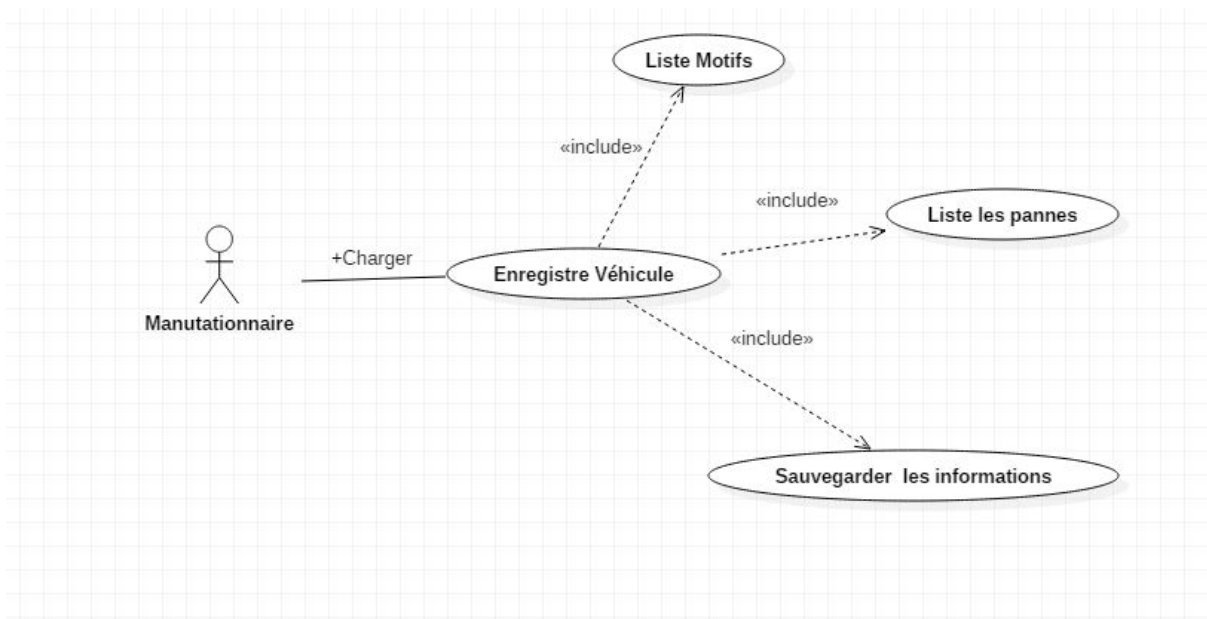
Pannes

Panne  charger Ajouter

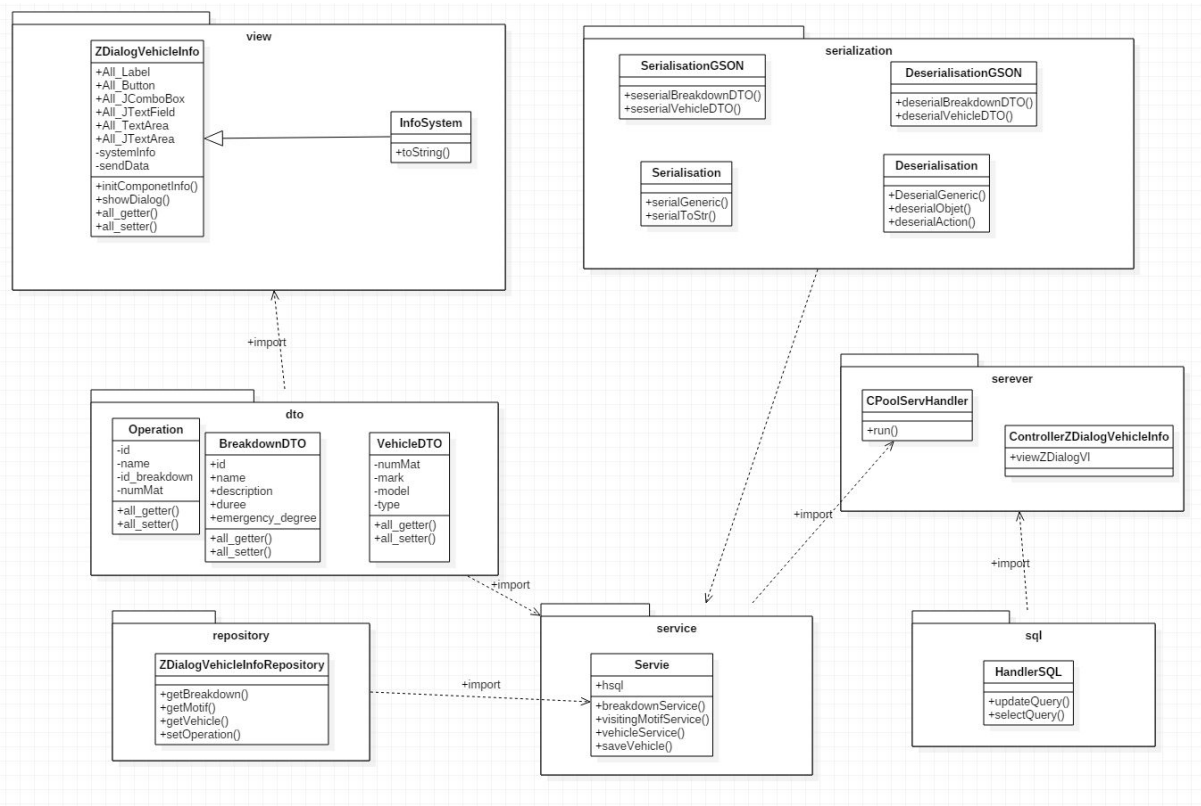
LISTE DE PANNES:

Enregistrer Annuler

Le diagramme suivant est une synthèse:



Les différents classes : D'où le diagramme de classe.



## Description du travail

### **ACTEURS :**

le réparateur

### **Description :**

le réparateur peut consulter la liste de toutes les pièces utilisé par un simple clic sur le bouton de même nom , visualiser le niveau de stock d'une pièce en rentrant son nom et en cliquant sur le bouton niveau de stock, et aussi voir l'historique (entrée / sortie ) d'une pièce en rentrant sa référence avant de cliquer sur le bouton Historique .

Indépendamment le programme ordonnera un achat si le niveau de stock d'une pièce est en dessous d'un seuil qu'on a déterminé et cela grâce à un trigger mis en place en base de donnée.

### **Scenarios :**

#### **Scenario normal :**

Le réparateur cherche à savoir le niveau de stock de la pièce "BOITE DE VITESSE AUDI" qui est dans la base de données il rentre le nom de la pièce et clic sur le bouton niveau de stock le résultat s'affiche.

Le réparateur cherche à savoir L'historique de la pièce "BOITE DE VITESSE AUDI" qui est dans la base de données il rentre la référence de la pièce qui est "AAAA8888 " et clic sur le bouton niveau de stock le résultat s'affiche.

Le réparateur veut voir toute les pièces utilisé il clic sur le bouton le résultat s'affiche.

#### **Scenario Alternatif :**

Le nom de la pièce ou la référence entré par le réparateur n'est pas dans la base de donné on affiche " erreur !! ".

On a pas encore de pièce utilisées et le réparateur clic sur pièce utilisé on affiche " 0 piece utilisé ".

**Description détaillé :**

**La view dépôt :**

depot

niveau de stock

nom p

PNEU AUDI

5

historique

ref p

GGGG4748

entré le 01/12/2015  
sortie le 05/01/2016

pieces utilisés

pneu audi  
FFFF4545  
pneu Reanaut  
FFFF4646  
volant Audi  
FFFF4747

## 1)Bouton niveau de stock :

Pour avoir le niveau de stock d'une pièce on doit rentrer le nom de cette dernière dans un JTextField le programme prendra le nom et fera un select dans la base de donnée à la table piece\_stock et additionnera toutes les lignes où se trouve le nom de la pièce et nous donnera le résultat qui sera affiché grâce à un label dans la vue

La table piece\_stock de donnée :

```
mysql> SELECT * from piece_stock ;
```

ref_piece_detached	name	login_administrator	date_reception
FFFF4646	Pneu audi	jack	2017-02-02
FFFF4647	Pneu audi	jack	2017-02-02
FFFF4648	Pneu audi	jack	2017-02-02
FFFF4649	Pneu audi	jack	2017-02-02
FFFF4747	Pneu Renault	jack	2017-02-03
FFFF4748	Pneu Renault	jack	2017-02-03
FFFF4749	Pneu Renault	jack	2017-02-03

```
7 rows in set (0.00 sec)
```

## 2)Bouton Historique :

Pour avoir l'historique (date d'entrée /date de sortie) on doit rentrer la référence de la pièce notre programme fera un select de la date de réception dans la table piece\_stock à la colonne date\_reception qui nous donnera la date d'entrée et un autre select dans la table piece\_consumption dans la colonne date toujours qui nous donnera la date\_reception de sortie de la pièce et les affichera dans un JTextArea

Les Tables piece\_stock et piece\_consumption :

```
mysql> SELECT * from piece_stock ;
```

ref_piece_detached	name	login_administrator	date_reception
FFFF4646	Pneu audi	jack	2017-02-02
FFFF4647	Pneu audi	jack	2017-02-02
FFFF4648	Pneu audi	jack	2017-02-02
FFFF4649	Pneu audi	jack	2017-02-02
FFFF4747	Pneu Renault	jack	2017-02-03
FFFF4748	Pneu Renault	jack	2017-02-03
FFFF4749	Pneu Renault	jack	2017-02-03

```
7 rows in set (0.00 sec)
```

```
mysql> SELECT * from piece_consumption ;
```

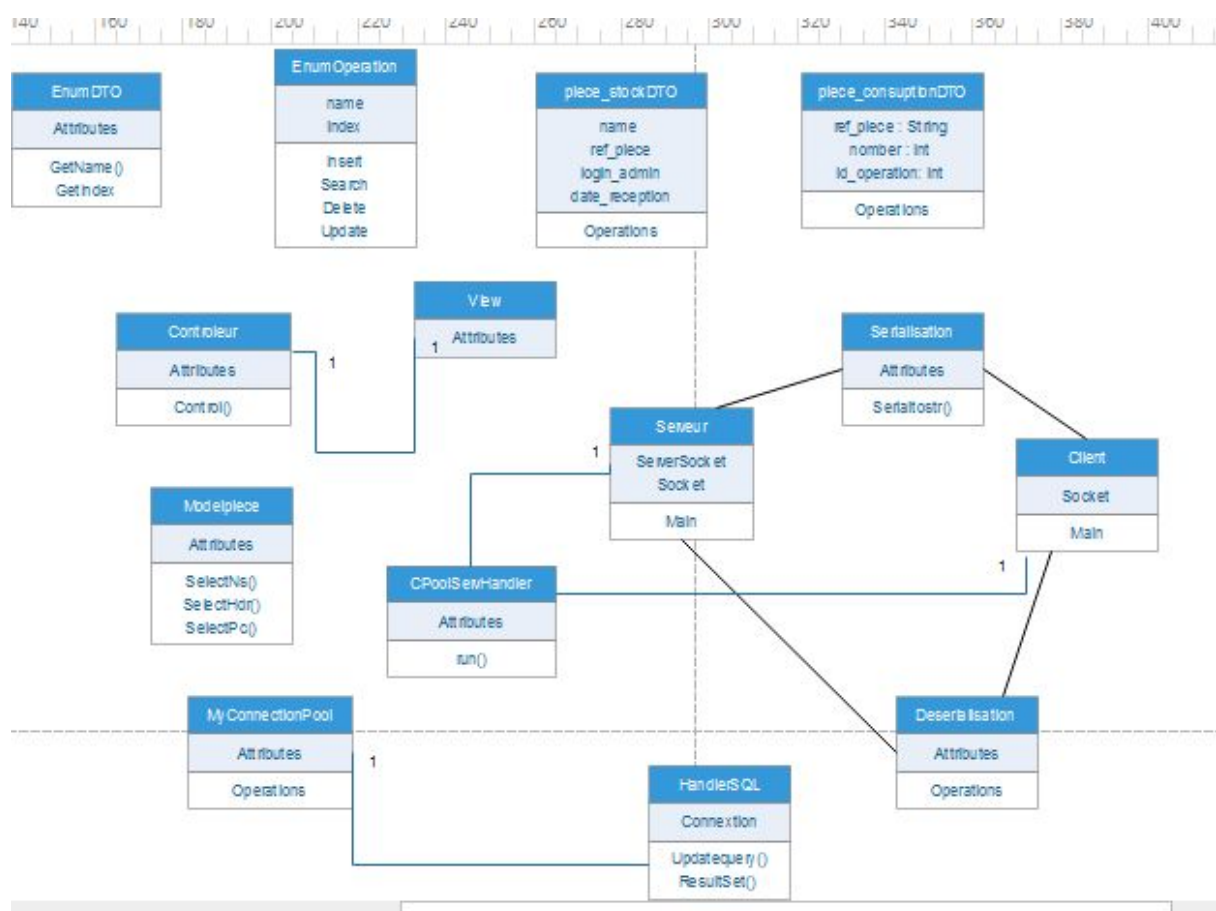
id_conso	ref_piece_stock	name	numConso	id_operation	date_reception
1	FFFF4747	pneu Renault	3	1	2017-02-10
5	FFFF4646	pneu audi	42	2	2017-02-10

```
2 rows in set (0.00 sec)
```

### 3) Bouton pièces utilisé :

Pour avoir toutes les pièces utilisées on a juste à appuyer sur le bouton toutes les pièces qui fera un select dans la table piece\_consumption colonne Name et qui affichera toutes ses pièces + leur référence dans un JText Area.

### diagramme de class :



## **IV-Anaïs HEMICI**

### **USE CASE**

**Je suis en charge du use case suivant : prioriser les opérations de maintenance.**

### **DESCRIPTION GLOBALE**

**Lorsque un véhicule entre dans le dépôt, un manutentionnaire se charge d'entrer les opérations de maintenance à effectuer. Un réparateur tout au long de son processus pourra modifier des champs dans la table Operation (par exemple le champs done : 1=fait, 2=en attente de pièces, 3=en attente de réparation) ou encore ajouter des operations qui n'ont pas été prévues à l'entrée du véhicule.**

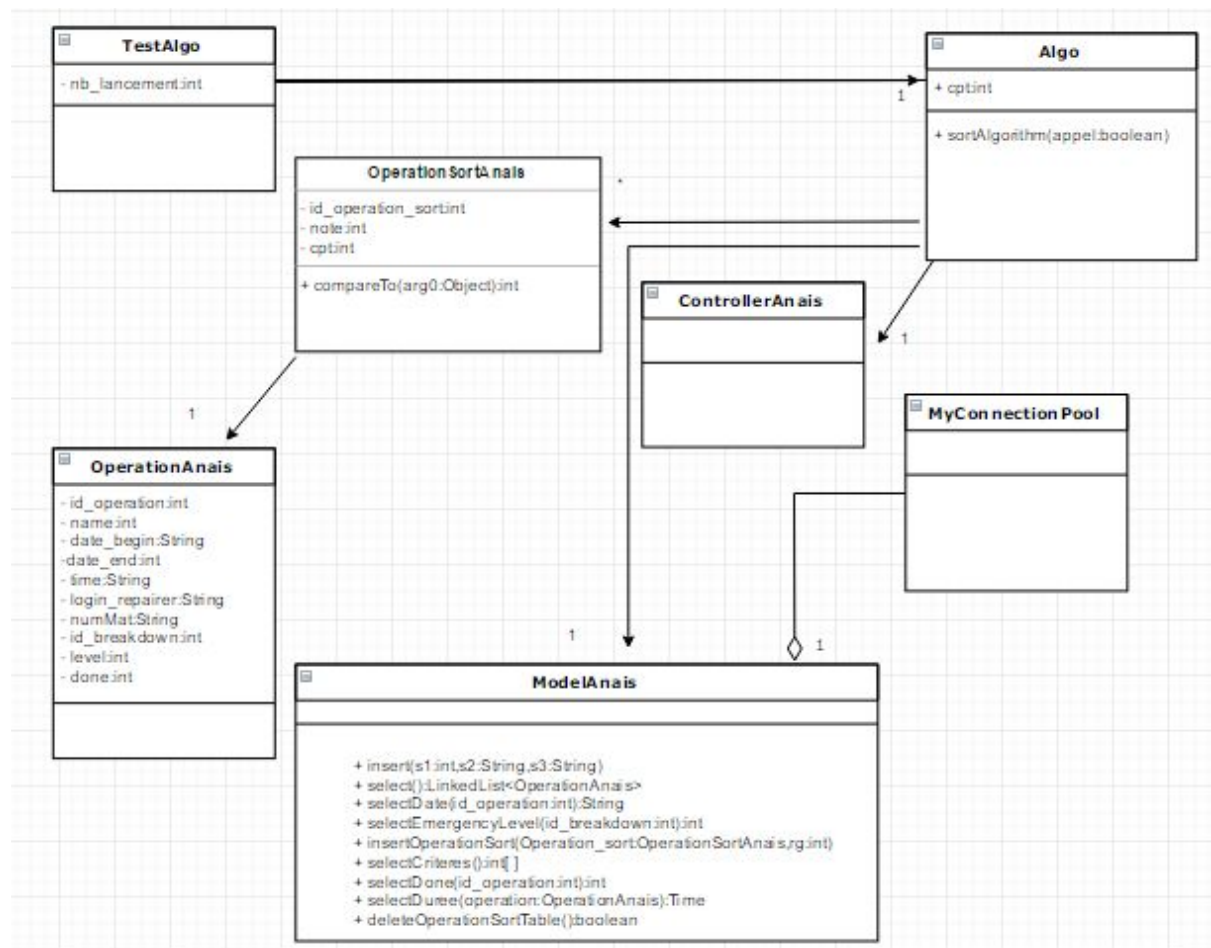
**A chaque mouvement sur la table Operation, un algorithme de priorisation basé sur un ensemble de critères, est lancée et alimente la table Operation\_Sort.**

**Aucun acteur n'est en relation avec ce use case car il tourne de façon automatisé au sein du serveur.**

### **DIAGRAMME DE CLASSE**

**Ce diagramme classe présente les principales classes utilisées pour réaliser mon UC.**





## TABLES UTILISEES

Table operation

Nom de l'attribut	Type	Description
Id_operation	INT	ID de l'opération
Id_breakdown	INT	ID de la panne
Time_begin	TIME	Horaire de début de l'opération
Time_end	TIME	Horaire de fin de l'opération

<b>Date_begin</b>	<b>DATE</b>	<b>Date de début de l'opération</b>
<b>Date_fin</b>	<b>DATE</b>	<b>Date de fin de l'opération</b>
<b>Done</b>	<b>ENUM</b>	<b>Etat de l'operation : 1=operation réalisée 2=opération en attente de pièces 3=opération en attente</b>
<b>Login_repairer</b>	<b>VARCHAR</b>	<b>Nom du réparateur en charge de l'opération</b>
<b>NumMat</b>	<b>VARCHAR</b>	<b>Numéro de matricule du véhicule nécessitant l'opération</b>

**Table operation\_sort**

<b>Nom de l'attribut</b>	<b>Type</b>	<b>Description</b>
<b>Id_operation_sort</b>	<b>INT</b>	<b>Id de l'opération triée</b>
<b>Id_operation</b>	<b>INT</b>	<b>Id de l'opération</b>
<b>NumMat</b>	<b>VARCHAR</b>	<b>Numéro de matricule du véhicule nécessitant l'opération</b>
<b>Emergency_degree</b>	<b>INT</b>	<b>La note globale de l'opération</b>
<b>Rang</b>	<b>INT</b>	<b>Classement des opération</b>

**Table critere**

<b>Nom de l'attribut</b>	<b>Type</b>	<b>Description</b>
<b>Piece_stock</b>	<b>INT</b>	<b>Voir si les pièces sont disponibles pour réaliser l'opération</b>

Emergency_level	INT	Correspond au niveau de difficulté de répartition d'une panne (de 1 à 10)
Date_entrance	DATE	Date d'entrée du véhicule
Time	TIME	Durée de l'opération

## FONCTIONNEMENT DE L'ALGORITHME

Etape	Déroulement
1	L'appel de l'algorithme avec le paramètre true en entrée incrémente un compteur qui au bout cpt=20 met en tête de liste une grosse opération de maintenance (non prioritaire habituellement). Le paramètre cpt est incrémenté dans le cas ou la réparation est effective (champs done=1=opération réalisée). Dans de cas ou le réparateur se rend compte du manque de pièces (le champs done passe à 2=en attente de pièces) l'opération est rejetée en fin de liste. Dans ce cadre l'algorithme est appelé avec le paramètre false.
2	<p>La méthode sortAlgorithm(boolean appel) commence par supprimer tous les champs de Operation_sort. Les coefficients des critères (stockés en base) sont récupérés et stockés dans un tableau : int[ ] CritereValues.</p> <p>CritereValues[0] = C0  CritereValues[1] = C1  CritereValues[2] = C2  CritereValues[3] = C3</p> <p>Pour chaque opération dans la table Opération sera crée une ligne dans la table Opération_sort.</p>
2	Le calcul de la note se fait en fonction des critères <b>ci-dessous</b> .
3	Chaque operation_sort est insérées dans la table Operation_sort suivant l'argument note.

Critère	<=5	]5 ;10]	]10 ;15]	>15
Date_entrance	C2	C2+2	C2+4	C2+6

Critère	<=2	]2 ;4]	]4 ;6]	]6;8]	>8
Emergency_level	C1 /2	C1	C1+2	C1+4	C1+6

Critère	Durée-seui l0 <=10	(seuil0) >= 0 && duree-seuil 1 < 0	(seuil1) >= 0 && duree.-seuil 2 < 0	seuil2) >= 0 && duree-seuil 3 < 0	seuil3) >= 0 && duree-se uil4 < 0
Time	C3+10	C3+6	C3+2	C3+1	C3

Critère	Done=2=en attente de pièce	Done=3=en attente de réparation
Pièce stock	Met l'opération en dernière dans la liste	Est classée en fonction des trois critères précédents

V-Laetitia ZADI

## Enregistrer une sortie de véhicule :

### Description:

A certain moment de la journée, les manutentionnaire consultent la liste des véhicules réparés et qui doivent être sortie du dépôt. Le manutentionnaire doit donc identifier ces véhicules, modifier leur statut, indiquer l'heure courante le nouveau lieu de dépôt,etc.

### Version plus détaillée :

L'acteur de ce use-case est le manutentionnaire. Il doit afficher la liste des véhicules qui ont été réparé à la date d'aujourd'hui, afficher les caractéristiques, les détails d'un véhicule ayant été réparé à la date d'aujourd'hui. Il doit également relocaliser un véhicule dans un autre dépôt en au moment de la sortie du véhicule. Le manutentionnaire indiquera également l'adresse du nouveau dépôt pour le véhicule en question.

### Afficher la liste des véhicules réparés :

Le manutentionnaire saisit la date de fin d'opération permettant d'afficher les véhicules qui ont fini d'être réparé à cette date. Une fois qu'il ai fait cela, une liste de véhicule sera affiché.

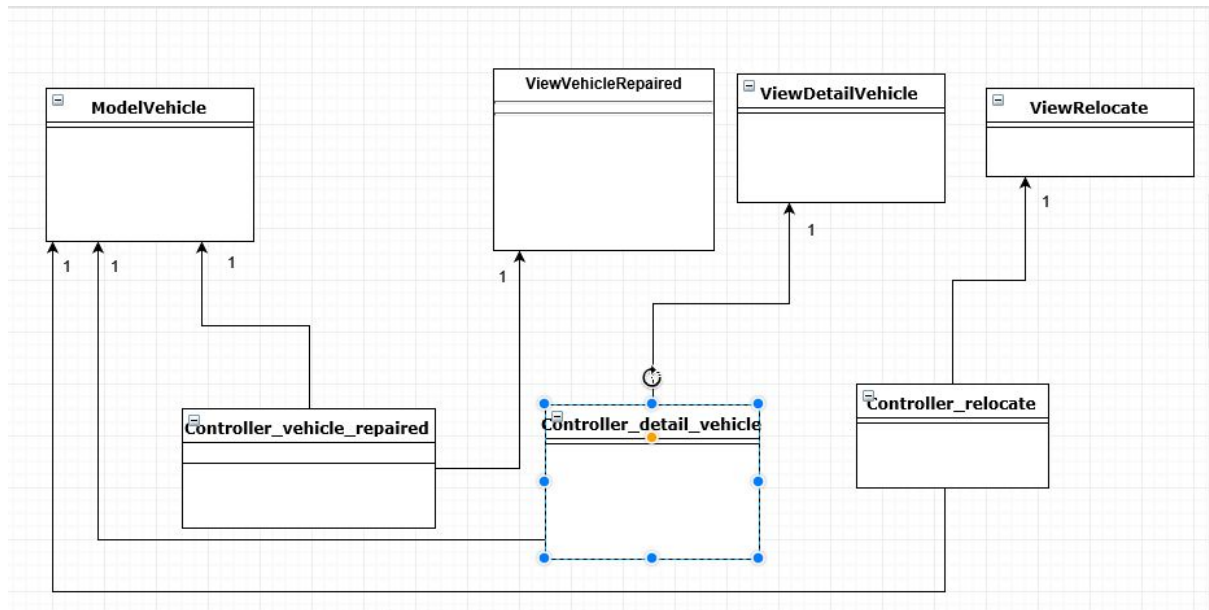
### Afficher le détail/caractéristiques d'un véhicule :

Le manutentionnaire saisit un numéro d'immatriculation d'un véhicule où il vient de finir d'être réparé. Les caractéristiques du véhicule seront affiché sur la fenêtre en question.

### Relocaliser un véhicule :

Le manutentionnaire modifie le statut, la date de sortie et indique un identifiant pour mettre le véhicule dans un nouveau dépôt. Comme cela, le manutentionnaire pourra connaître l'adresse du nouveau véhicule pour qu'il puisse être relocaliser. Il fera de plus, une mise à jour de l'heure et de la date indiquant le moment où il a sortit le véhicule.

### **Diagramme de classe:**



**VI-Youcef BOUNEKTA**

### **spécifications fonctionnelles :**

**use case :** Réaliser une opération de maintenance

### **Description du travail :**

les acteurs concernés par cette partie sont les personnes capables de faire une opération de réparation que ce soit un manutentionnaire ou un réparateur .

tout d'abord le réparateur se connecte , après avoir réussi l'authentification ce dernier doit avoir la possibilité de réaliser plusieurs tâches et évidemment plusieurs changements au niveau de l'interface graphique ainsi qu' au niveau de la base de données .

une fois le réparateur connecté il peut dans un premier temps consulter la liste des opérations existantes en récupérant toutes les lignes de la table " operation\_sort " .cette opération est effectuée après le clic sur un bouton précis .

le réparateur doit pouvoir sélectionner une opération pour s'en occuper cependant il n'a pas la possibilité de choisir quelle opération ,vu que les opérations sont priorisées il existe la plus urgente , donc de la sélection le réparateur choisit automatiquement l'opération la plus urgente ceci doit se faire par un clic sur un bouton "prendre une operation " , suite à ce clic le réparateur doit pouvoir consulter toutes les informations concernant ce véhicule afin d'avoir une idée sur le type de panne le placement au dépôt ainsi que la durée estimée pour sa réparation qui a été définie lors de son diagnostic (entrée au dépôt).

on préfère voir sur la vue principale de l'application les informations suivantes : voiture et le type de panne (motif) .

par contre pour le reste des informations (marque, type, date d'entrée au dépôt ...etc) doivent être accessibles que par un clic sur un bouton nommé (info véhicule), à ce moment le réparateur a bien choisi le véhicule qui correspond à l'opération la plus urgente, il doit procéder donc à sa réparation pour cela il en a besoin des pièces, donc l'application doit lui permettre notamment de commander des pièces ainsi que de vérifier la disponibilité des pièces.

un bouton "ajouter pièce" doit être présent sur la vue, en cliquant sur ce dernier une vue de sélection de pièce apparaît, au niveau de cette dernière le réparateur a la possibilité de sélectionner le nom de la pièce souhaitée ainsi que la quantité par contre, si le nom de la pièce est faux ou un des champs de saisie (pièce, quantité) est vide l'application doit indiquer par un message d'erreur, si la quantité saisie par le réparateur est supérieure à celle de la pièce choisie en stock le système doit indiquer par un message d'erreur que la quantité demandée est supérieure à celle de la pièce en question en stock.

si la pièce existe en stock et la quantité est bien inférieure à celle de la pièce en stock le système doit non seulement réserver cette pièce, et informer le réparateur.

le système réserve la pièce en décrémentant le nombre associé à sa disponibilité en stock suivant le calcul suivant :

**quantité stock := quantité stock - quantité saisie par le réparateur**

le système affichera pour le réparateur un message de réussite de cette tâche, et le nom de la pièce ainsi que sa quantité (qte choisie par le réparateur) doivent être visibles au niveau de la grande vue.

au cours de l'opération de réparation le réparateur peut constater de nouvelles pannes non détectées lors du diagnostic de ce véhicule et, pour cela l'application doit permettre l'ajout d'une opération (une opération est liée à une panne précise, bijection entre pannes et opérations), donc un bouton "ajouter panne" doit être présent, en effet une vue doit s'afficher cette dernière elle permet la saisie du nom de la panne en cliquant sur le bouton ok de cette dernière une ligne s'ajoutera au niveau de la base de données dans la table opération avec les attributs suivants :

- date début et date fin : nulle.
- heure début heure fin : nulle.
- id panne : c'est l'id lié au nom de la panne saisie par le réparateur.
- login réparateur : c'est le login du réparateur connecté.
- état (donne) : en attente de réparation.

bien évidemment si le réparateur saisit un nom de panne qui n'existe pas dans le référentiel des pannes, un message d'erreur lui sera affiché, ainsi, si le nom est bien valide un message lui sera affiché indiquant la réussite de l'ajout de panne à ce véhicule.

quand le réparateur termine son opération il clique sur le bouton fin d'opération, un autre bouton s'affichera "fin", en cliquant sur ce dernier le système affiche au réparateur le nombre d'opération restantes concernant ce véhicule (recherché dans la table opération par numéro d'immatriculation) et effectue des modifications au niveau de trois tables dans la base de données : "véhicule" et "opération" et "opération sort"





**VII-Ali BANHAKEIA**

**Description du travail :**

ACTEUR :

Personnel

Chef de dépôt

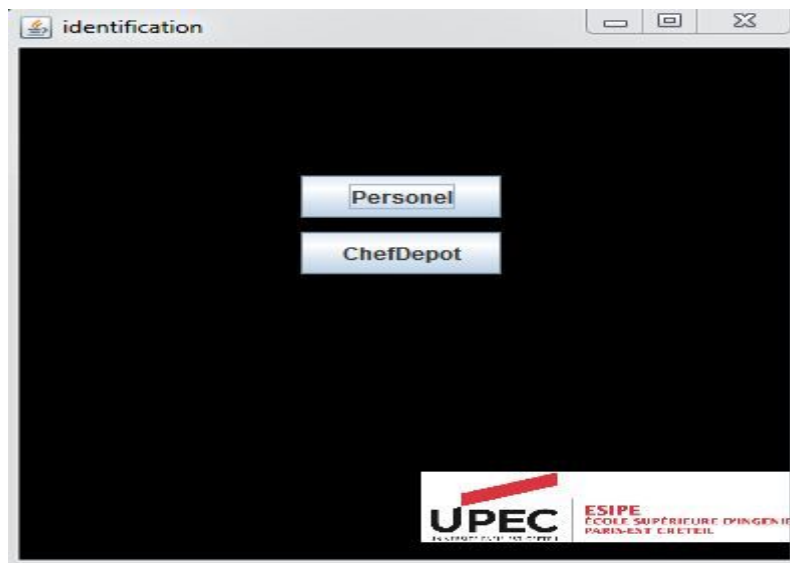
Description :

Tout d'abord on a une fenêtre identification ou on a deux boutons personnel et chef de dépôt

- 1) Chef dépôt : Si on choisit chef de dépôt on a une fenêtre d'identification qui apparait après s'être identifié on aura la fenêtre chef dépôt ou on peut entrer le nom et prénom d'un réparateur et voir toute les réparations et le nombre d'opérations qu'il a effectué.
- 2) Personnel : si on choisit personnel et après l'identification on aura la fenêtre personnel ou on pourra entré le matricule d'une voiture et voir son statut, mettre une date début et une date fin pour voir la liste de toute les voitures réparé entre temps et voir le workflow de tous les véhicules du dépôt.

### Description détaillé :

#### 1)La view identification :

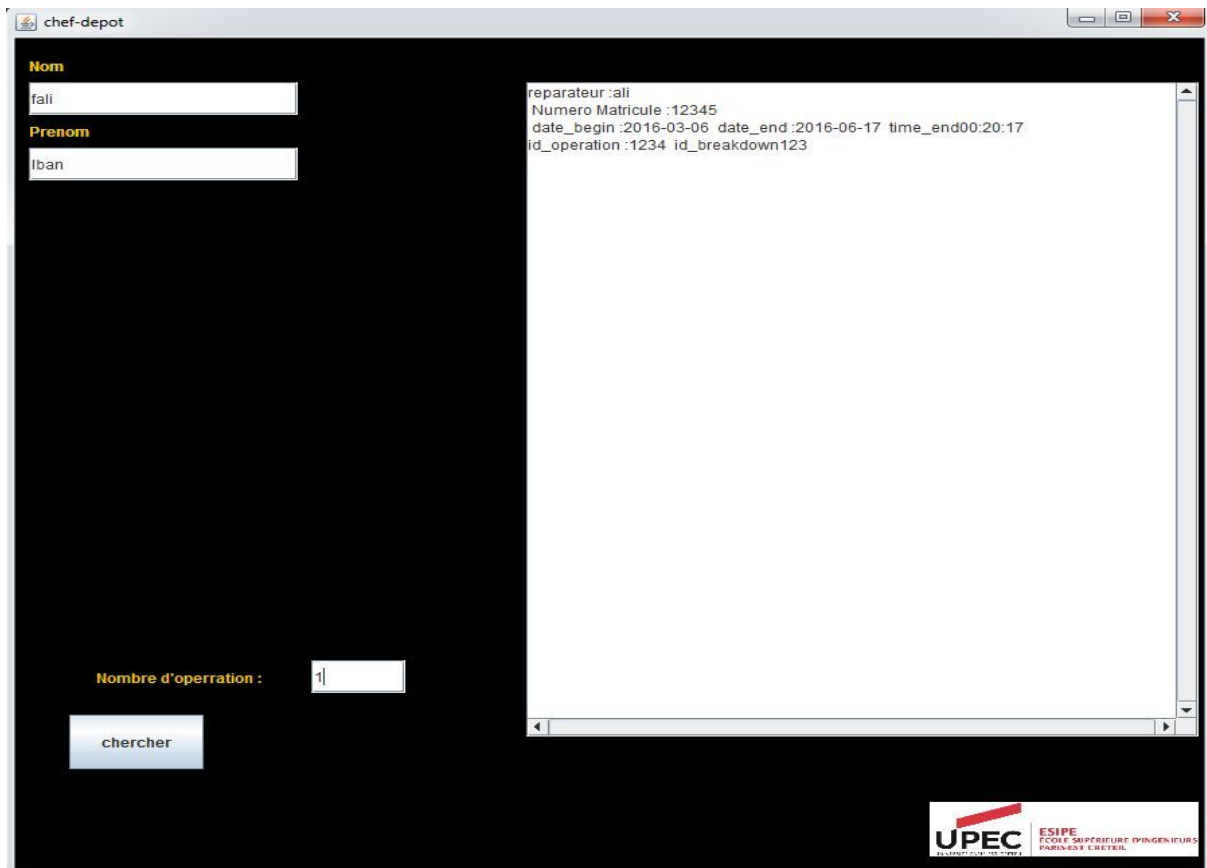


#### 2) view identification 2 :

The image shows a screenshot of a web application window titled "identification". The window has a black background. In the center, there are two white input fields. The first field is labeled "login :" in yellow text. The second field is labeled "Password :" in yellow text. Below these fields is a blue button with the text "identifier" in white. At the bottom right of the window, there is a logo for "UPEC" (Université Paris-Est Créteil) and "ESIPE" (École Supérieure d'Ingénierie Paris-Est Créteil).

on rentre le login et le password que le programme va recuperer dans la base de donnee aux tables reparateurs (personnel) ou adminisrateur (chef du depot)

3)view chef depot :



on a deux JTextField pour entrer le nom et le prenom du reparateur ,on clic sur le bouton “chercher” le programme récupère les informations dans la base de donnée a la table Operation et les affiche sur un JtextfieldArea

le nombre d’oparation se calcule automatiquement et s’affiche sur un JTextField

4)view personnel :

cette view a trois fonctionnalité :

- 1) afficher le statut d’un véhicule récupéré à la table vehicule\_warehouse dans la base de donnée à la table après avoir entré son matricule en cliquant sur le bouton chercher

Personel

Matricule

BBB

Numero Matricule :BBB satus :en attente

Date debute

Date Fin

chercher

cumulée

VehiculeMt

UPEC

ESIP  
ÉCOLE SUPÉRIEURE D'INGÉNIEURS  
PAROISSY CRETIL

2) entré une date d'entré et une date de sortie pour afficher tous les vehicules réparé entre temps l'information sera récupérée par le programme dans la base de donnée en cliquant sur cumulée

Personel

Matricule

Numero Matricule : BBB date\_entrance2017-02-12 date\_wayout2017-05-30  
Numero Matricule : BBbC date\_entrance2017-02-12 date\_wayout2017-05-30  
Numero Matricule : BBbCRAAAAAARR date\_entrance2017-02-12 date\_wayout2017-05-30  
Numero Matricule : BBbCRRR date\_entrance2017-02-12 date\_wayout2017-05-30  
Numero Matricule : Rlf2017 date\_entrance2017-02-12 date\_wayout2017-05-30

Date debut 2017-02-12 Date Fin 2017-05-30

chercher cumulée

VehiculeMt

**UPEC** ESIPÉ  
ÉCOLE SUPÉRIEURE D'INGÉNIEURS  
PARLEMENTAIRE

3) afficher le workflow (que le programme récupérera dans la base de donnee a la table vehicul\_warehouse) de tous les véhicules du dépôt en cliquant sur le bouton vehiculMT

Personel

Matricule

Date debut

2017-02-12

Date Fin

2017-05-30

chercher

cumulée

VehiculeMt

Numero Matricule :BBB satus :en attente

Numero Matricule :BBbC satus :en attente

Numero Matricule :BBbCRAAAAAARR satus :en attente

Numero Matricule :BBbCRRR satus :en attente

Numero Matricule :CCC satus :en attente

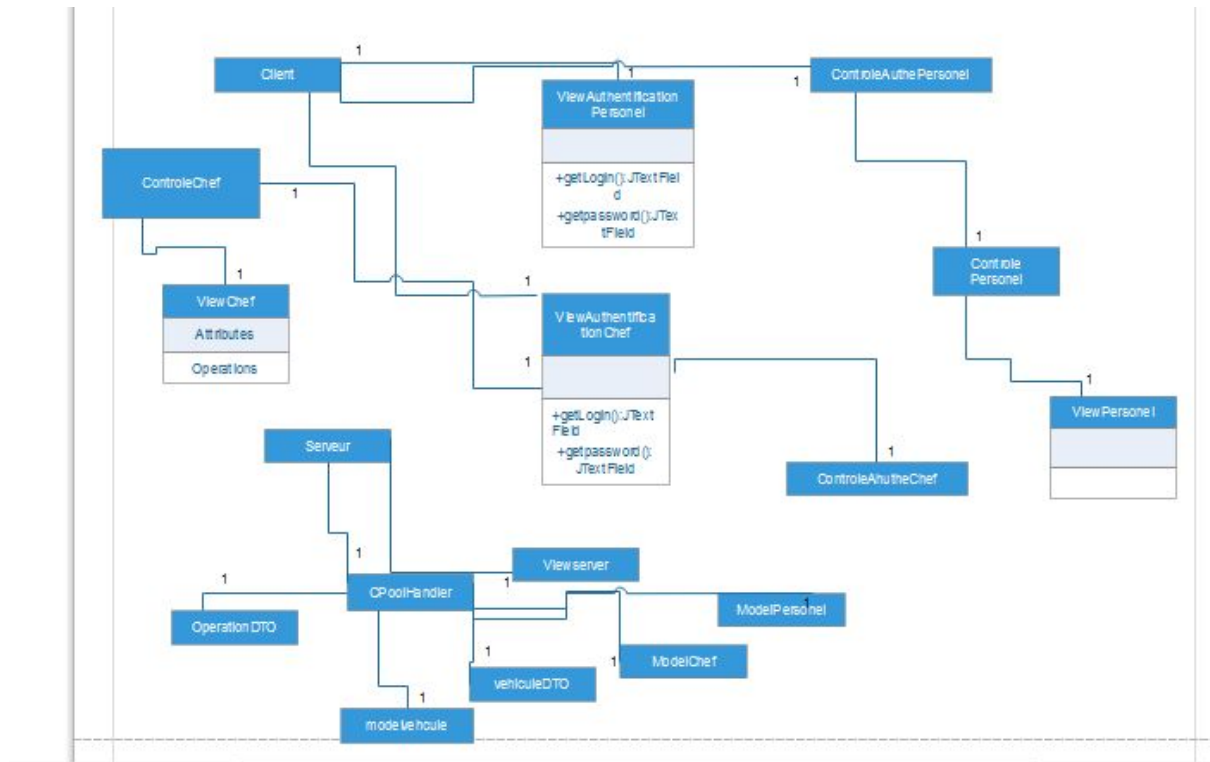
Numero Matricule :DDD satus :en attente

Numero Matricule :EEE satus :en attente

Numero Matricule :FFF satus :attente pieces

Numero Matricule :Rlf2017 satus :en attente

diagramme de class :



## MANUELS UTILISATEUR

I-Yuxin SHI

### 1. Mode d'emploi

Indicators

Here a window of application to start a request:



The user has to choose some indicators:

- beginning day(When do operations start?)

- end of day (When do operations end?)

-scale of time(there are 3 types scale to choose, weeks/month/year, these scale time is used to analyse the operation which are sent from server. But there is a constraint of date. If users choose “weeks”, the gap between the beginning day and the end of day can’t greater than 8 weeks. For month, the gap can’t be greater than 12 month, for “year”, this value is 5 years. For example, if the user choose “weeks” and the beginning day “2016/02/01” and the end of day “2016/05/01”, a popup window with error message will appear, but not for an end of day like “2016/03/01”. )

- le type de dommage(Which type of breakdown do you want to query?)

A dropdown menu with a light gray border. The text 'fenetre' is displayed in the main area, and a small downward-pointing arrow is visible on the right side of the text box.

- le statut(What statut do you query? finish or in operating?)


A dropdown menu with a light gray border. The text 'operation finie' is displayed in the main area, and a small downward-pointing arrow is visible on the right side of the text box.

- le type de véhicule(bike or car?)

A form with a light gray border. At the top, the text 'Type de Véhicule' is displayed. Below it, there are two checkboxes: 'Velo' and 'Voiture', both of which are currently unchecked.

### Table and Rapport

While users click button “comfirmer”, these indicators will be put in a HashMap and sent to the server. The server response the results of query which come from SQL server to the user. Then the data appears in the mode of table at left, after analysing results come from server, the rapped of the time period that users have choose appears at right in the scroll text field, which shows the number of operation on the basis of time scale chosen.

A form with a light gray border. It contains several sections: 1. 'Echelle du temps' with three radio buttons: 'Week', 'Month' (which is selected), and 'Year'. 2. 'De/à' with two date input fields: '2016-09-01' and '2017-06-09', each with a small calendar icon to its right. 3. 'Type de Véhicule' with two checked checkboxes: 'Velo' and 'Voiture'. 4. Two dropdown menus on the right: the top one is labeled 'operation finie' and the bottom one is labeled 'all'.

For example I choose the date that we want to request: from 2016-09-01 to 2017-06-09, how many operations of bike and car have been executed with all the type of breakdown? And I want to analyse it by month. There is the result showing below:

Vehicle Ty...	Number V...	Breakdown	Responsible	Piece Con...	Rapport
voiture	BI023	ralenti irr...	henri henri	0	<p>nombre de l'opération: 11 pieces consommées en totale: 1 Durée moyenne: 10.73 jours. Durée maximale pour une opération est 31 jours. Cette opération est effectuée par bard bard avec le véhicul</p> <hr/> <p>Les données sont analysées par mois :</p> <hr/> <p>Sat Oct 01 00:00:00 CEST 2016: Il n'y a aucune d'opération sont effectuée.</p> <hr/> <p>Tue Nov 01 00:00:00 CET 2016: Il n'y a aucune d'opération sont effectuée.</p> <hr/> <p>Thu Dec 01 00:00:00 CET 2016: Les véhicules qui sont réparés dans cette échelle : BI023   CCC   Opérations effectuées: 2 Pièces consommées: 0 Durée moyenne par l'opération: 10 Durée maximale pour une opération est 21 jours. Cette opération est effectuée par henri henri avec le vét</p>
velo	BBB	XXX	bard bard	0	
velo	BBB	FGDFF	bard bard	1	
voiture	SW60802	turbo	henri henri	0	
voiture	SW60802	vitesse dif...	bard bard	0	
voiture	SW60802	embrayage	bard bard	0	
voiture	SW60802	conso. huile	henri henri	0	
voiture	Passat02	electroniq...	bard bard	0	
velo	CCC	crevaisson	henri henri	0	
voiture	SW60802	ralenti irr...	bard bard	0	
voiture	SW60802	allumage	bard bard	0	

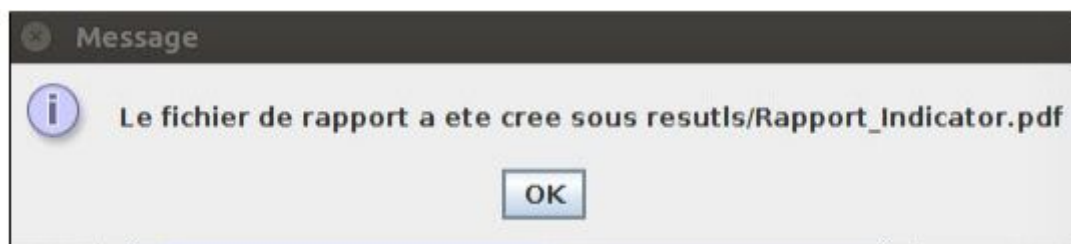
At left side, it's a table of all the operations which have been found, and in the text area at right side, it's the rapport exported according to time scale "month", it shows firstly the number of operation, all the pieces have been consumed by these operations, the maximum period to repair a vehicle and the average period for repairing.

Then it shows the information in detail, for every month, it gives:

1. the vehicles which have been repaired
2. the number of operation which have been found
3. the average period for repairing one operation
4. the maximum period for repairing one operation
5. the repairer who has execute the operation consumed longest days

Export as PDF

At the bottom, there is a button "Exporter en PDF". This button is enabled after receiving the result from server. It can export a file name Rapport\_Indicator.pdf at the folder results



There the view of report:

## TheOriginal

Rapport Indicateur de 2016-09-01 a 2017-06-09

Type de dommage : ALL

Date	Nombres opé total	Max jours/opé	Min jours/opé	Avg jours/opé	Max piece conso/opé	Mon piece conso/opé	Avg piece conso/opé
01/10/2016	0	0	0	0	0	0	0
01/11/2016	0	0	0	0	0	0	0
01/12/2016	2	21.0	0.0	11	0.0	0.0	0
01/01/2017	1	6.0	0.0	28	0.0	0.0	0
01/02/2017	0	0	0	0	0	0	0
01/03/2017	3	31.0	0.0	30.67	1.0	0.0	0.33

**II-Bakari DEMBELE**

III-Karim LOBIYED

## Depot Pieces

### USER GUIDE

Here we can see the window depot that will be used for three tasks

The screenshot shows a window titled 'depot' with three distinct sections, each with a button and associated data:

- niveau de stock**: A button labeled 'niveau de stock' is followed by the text 'nom p' and a text box containing 'PNEU AUDI'. To the right of this is the number '5'.
- historique**: A button labeled 'historique' is followed by the text 'ref p' and a text box containing 'GGGG4748'. To the right of this text box are two lines of text: 'entré le 01/12/2015' and 'sortie le 05/01/2016'.
- pieces utilisés**: A button labeled 'pieces utilisés' is followed by a list of items: 'pneu audi', 'FFFF4545', 'pneu Reanaut', 'FFFFF4646', 'volant Audi', and 'FFFF4747'.

How to use it ?

**1) Display used pieces :**

You can Consult the list of all the pieces used by clicking on the button "pieces utilisés "

**2) Display stock level :**

You can Consult the stock level of a piece by entering it's name and clicking on the button "niveau de stock"

**3) Display History :**

You can see the history (Enter/ Exit) of a piece by entering it's reference before clicking on the button "Historique".

**IV-Anaïs HEMICI**

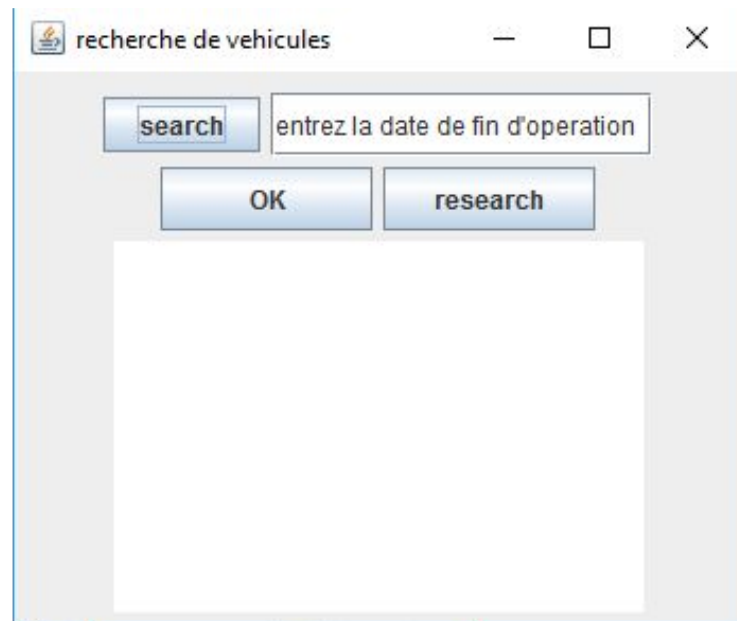
**My use case doesn't use a user.**

**In fact it is an algorithm which is running into the Server. It will run when there are modifications on table Operation. An operation add and an update on it can run the algorithm.**

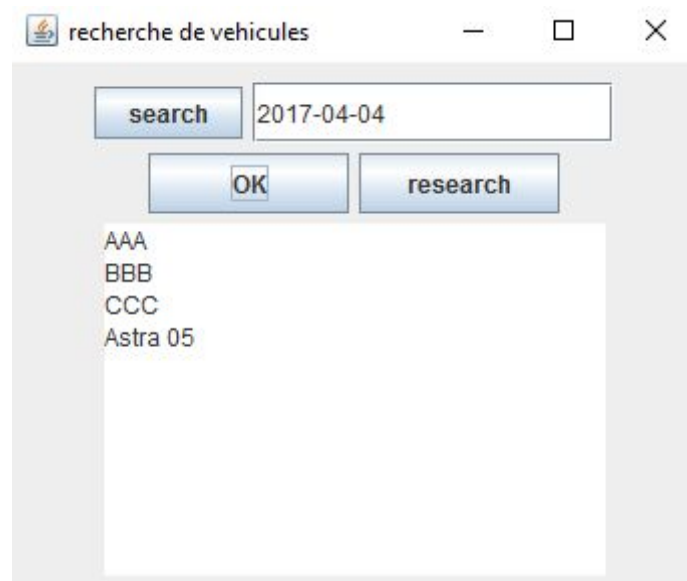
**For more details about the algorithm please refer you to fonctionnalites specifications in part IV-Anaïs HEMICI at the beginning of the document.**

**V-Laetitia ZADI**

To save a wayout vehicle, the user have to press the « SEARCH » button and it will show « entrez la date de fin d'operation »

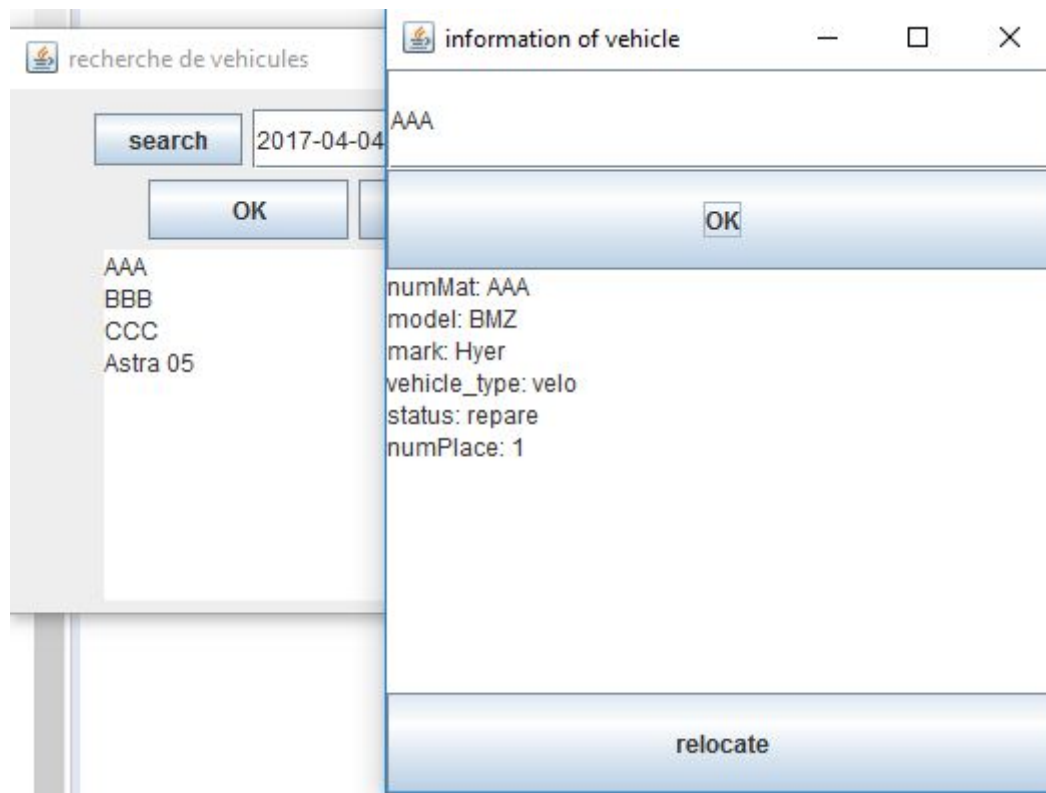


After that, the user enter a date of end of an operation and press the « OK » button. This will show a list of vehicles whose operations were completed on that date.



As we can see, there are four vehicles which have been repaired at this date.

To see, the characteristics of a vehicle, the user have to press the third button (« research ») to enter the registration number of the vehicle. The user press the « OK » button to show its characteristics.



Indeed, we can see the characteristics for this registration number

To put a vehicle in a new warehouse, the user have to press the « RELOCATE » Button which is located at the bottom. A new window will appear. The user enter the registration number the vehicle and an id\_warehouse. The user press the button and it will appear what has changed and the new address of the vehicle.



The screenshot shows a window titled 'RELOCATION' with a standard Windows interface (minimize, maximize, close buttons). The window is divided into several sections. On the left, there are labels for 'NumMat :', 'id\_warehouse :', and a summary of vehicle details. On the right, there are input fields for 'NumMat' (containing 'AAA') and 'id\_warehouse' (containing '4'). Below these, there is a blue button labeled 'OK'. To the right of the 'OK' button, there is a text area containing the following information: '1vehicule relocalise le 10 juin 2017 a 03:58:12', '1 date modifiee', and '1 vehicule mis en circulation'. At the bottom left, there is a summary of the vehicle details: 'numero d'immatriculation: AAA', 'id\_warehouse: 4', and 'adresse:bbc'.

NumMat :	AAA
id_warehouse :	4
<input type="button" value="OK"/>	
1vehicule relocalise le 10 juin 2017 a 03:58:12 1 date modifiee 1 vehicule mis en circulation	
numero d'immatriculation: AAA id_warehouse: 4 adresse:bbc	

We can see that there has been an upgrade and a relocated vehicle.

**VI-Youcef BOUNEKTA**

**PDS**

**Livrable De R3**

**BOUNEKTA Youcef**

**Manuel utilisateur :**

**Contenu**

**User**

# Manual

**This module of the application will allow the user to realize a repair operation.**

**When the application is launched, the repairman should firstly sign-in with his own login and password before getting in the application, after a verification of the system, if everything goes well and the data of the user are correct, the user can do a whole of tasks with all simplicity, here as you can see on the preview below, a first view appears to the repairman immediately after a right and full sign-in to the application:**

Reparation-vehicle

Prendre une Operation

Liste d'operations

Vehicle

Motif

Piece N°1	nul	quantite	nul
Piece N°2	nul	quantite	nul
Piece N°3	nul	quantite	nul
Piece N°4	nul	quantite	nul

Ajouter une operation

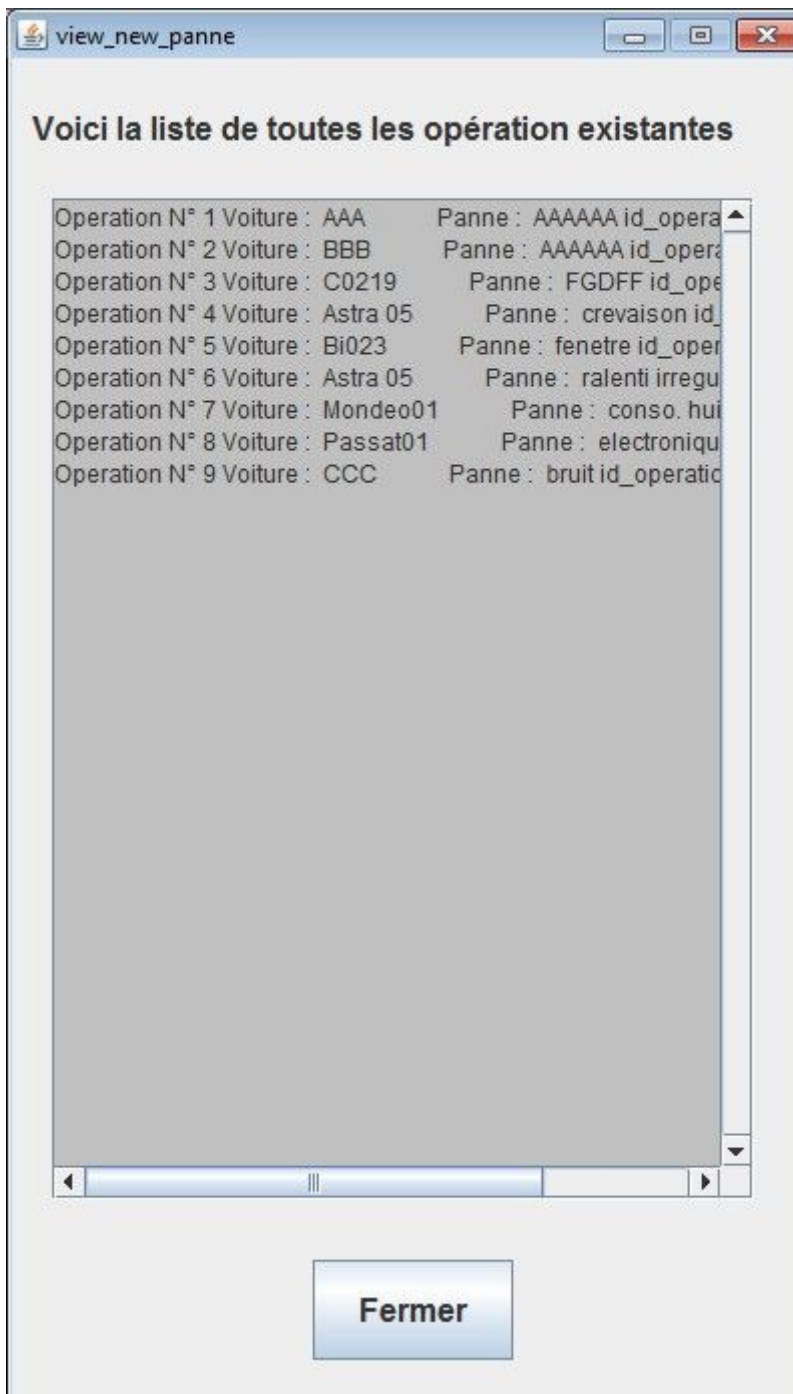
Fin d'operation

FIN

**This view as you can clearly see it, contains 5 buttons .**

**First of all, the user have the possibility to choose from a large list of the main operations that exist about any vehicle which resides in the deposit**

**Here is a preview of showing how the button « Liste d'operations » works,**



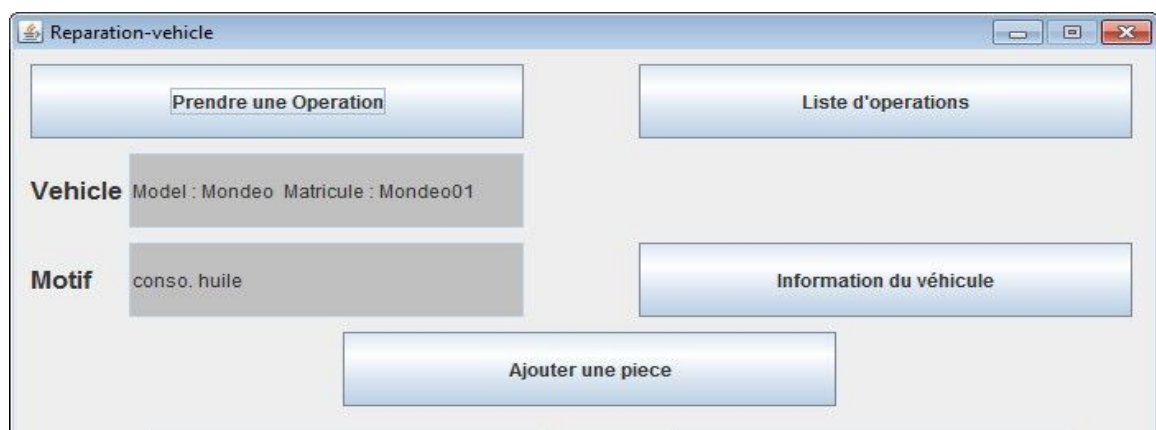
On the other hand, he will be obliged to deal with most urgent reparation, To take action, the repairer have to select this urgent case, only by clicking on the button « prendre une operation », automaticity, the name of the breakdown and the vehicle will appears in the The two text areas (car, pattern).

Here is a preview of showing how the button « prendre une operation » works,



It means two new buttons :

- 1- Vehicle information
- 2- Add piece



The repairer can consult the list of information concerning the vehicle in breakdown, in order to have a first idea on it.

By clicking on the vehicle info button, the information (model, Brand, estimated time, number of the vehicle position in the depot, etc.) will be displayed in this way.

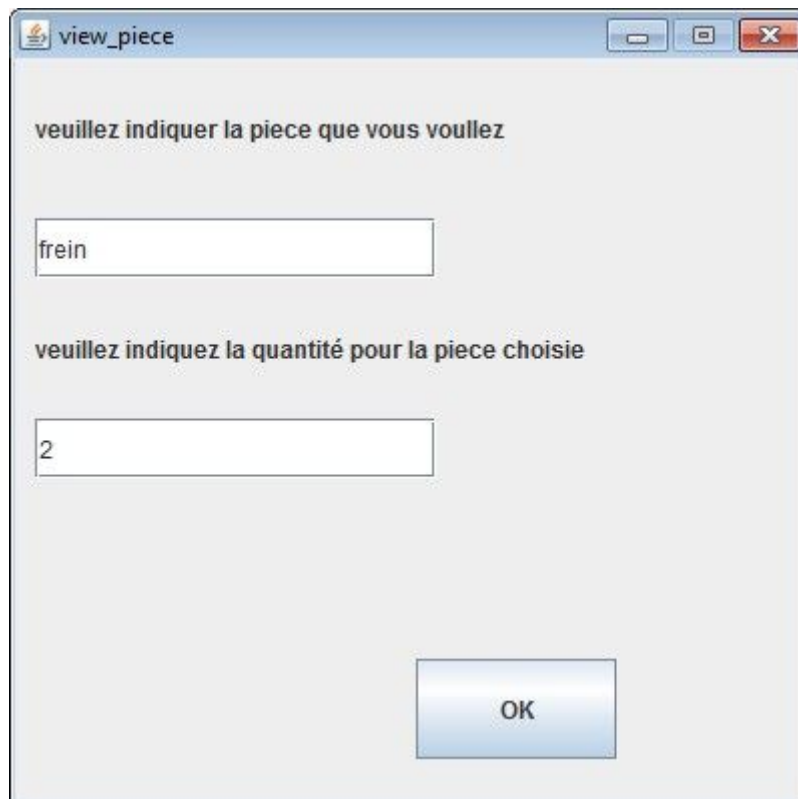
The screenshot shows a Java Swing window titled "viewInfoVehicle". It contains a form with the following fields and values:

Label	Value
Model :	Mondeo
Durée estimé :	02:30:00
Mark :	Ford
Type :	voiture
Date d'entrée :	2016-05-22
N° de place :	10

At the bottom center of the window is a button labeled "Fermer".

**The repairer must select the parts he needs to carry out this operation and for this purpose, it is necessary to click on the button << add piece >>, after that, a view will**

**be displayed, and the repairer must enter the name of the Piece concerned, And the quantity of it.**



view\_piece

veuillez indiquer la piece que vous voulez

frein

veuillez indiquer la quantité pour la piece choisie

2

OK

**By clicking on the <ok> button, the system will reserve the validated piece and it will be displayed in the main view (name piece, quantity)**



Piece N°1	frein	quantite	2
-----------	-------	----------	---

**During the operation if the repairer discovers a problem that will require another repair (new breakdown), then the repairer must indicate the problem by clicking on the <add breakdown> button, a view will be displayed, the repairer must indicate the name of the breakdown and click on the <OK> button to save this one.**



The screenshot shows a Java Swing window titled "view\_new\_panne". Inside the window, there is a text prompt "Veuillez indiquer le nom de la panne" followed by "que vous voulez ajouter". Below this is a text input field containing the word "frein". At the bottom center of the window is a button labeled "Ajouter".

**When the repairer finishes a task, he must click on the <find operation> button, another button will be displayed, by clicking on this new button the repairman will save the information concerning this reparation. The system will display a success operation message and also the number of breakdown remaining to be repaired, in order to make the vehicle operational again.**

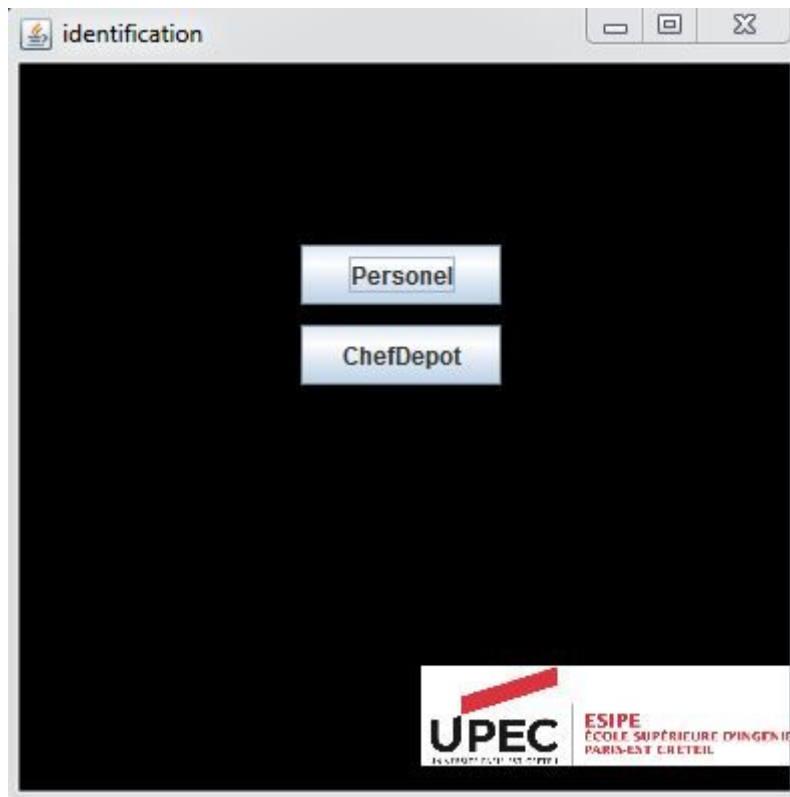
The screenshot shows the same Java Swing window "view\_new\_panne" after a successful operation. The text "-----OPERATION REUSSIE-----" is displayed at the top. Below it, the text "Le nombre actuel d'operations restante concernant ce vehicule est :" is followed by a gray square icon and the word "operations". At the bottom center is a button labeled "Terminer".

## USER GUIDE

the first window is identification

if you are repairer click on “personnel”

if you are manager click on “chefdepot”



after you have to identify yourself with login and password in this window

The image shows a screenshot of a web application window titled "identification". The window has a black background. In the center, there are two white input fields. The first field is preceded by the label "login :" in yellow text. The second field is preceded by the label "Password :" in yellow text. Below the input fields is a blue button with the text "identifier" in white. In the bottom right corner, there is a logo for "UPEC" (Université de Picardie Jules Verne) and "ESIPÉ" (École Supérieure d'Ingénierie de Picardie Est-Créteil).

after the authentication you will see a third window

- 1) the window of managers : we can see all the operations and the number of which of a repairer just in putting his first name and last name

The screenshot shows a web application window titled "chef-depot". On the left, there are two input fields: "Nom" (Last Name) with the value "fali" and "Prenom" (First Name) with the value "Iban". Below these is a "Nombre d'operation :" label and a small input field containing the number "1". A "chercher" (search) button is located below the number field. On the right, a large text area displays the search results for the repairer "ali". The results include: "reparateur : ali", "Numero Matricule : 12345", "date\_begin : 2016-03-06 date\_end : 2016-06-17 time\_end 00:20:17", and "id\_operation : 1234 id\_breakdown 123". The bottom right corner of the window features the logos for "UPEC" (Université Paris-Est Créteil) and "ESISE" (École Supérieure d'Ingénieurs Paris-Est Créteil).

2) Window of repairer : the repairer can do 3 tasks

1)display vehicle status by entring it's registration number

The screenshot shows a software window titled "Personel" with a dark background. It contains several input fields and buttons for searching vehicle status.

**Matricule**

BBB

**Date debute**

**Date Fin**

**Numero Matricule :BBB satas :en attente**

**chercher** **cumulée**

**VehiculeMt**

**UPEC** **ESIPE**  
ÉCOLE SUPÉRIEURE D'INGÉNIEURS  
PARIS-EST CRETEIL

2)Entered an entry date and an exit date to display all repaired vehicles in between

The screenshot shows a web application window titled "Personel". The interface is divided into several sections:

- Matricule:** A text input field for entering a vehicle matricule number.
- Date debut:** A date input field with the value "2017-02-12".
- Date Fin:** A date input field with the value "2017-05-30".
- Search Buttons:** Two buttons labeled "chercher" and "cumulée" are located below the date fields.
- Results Area:** A large text area displaying the following data:

```
Numero Matricule : BBB date_entrance2017-02-12 date_wayout2017-05-30
Numero Matricule : BBbC date_entrance2017-02-12 date_wayout2017-05-30
Numero Matricule : BBbCRAAAAAARR date_entrance2017-02-12 date_wayout2017-05-30
Numero Matricule : BBbCRRR date_entrance2017-02-12 date_wayout2017-05-30
Numero Matricule : Rlf2017 date_entrance2017-02-12 date_wayout2017-05-30
```
- Footer:** At the bottom right, there is a logo for "UPEC" (Université Paris-Est Créteil) and "ESIP" (École Supérieure d'Ingénieurs Paris-Est Créteil).

### 3)Display the workflow All vehicles of the depot

Personel

Matricule

Numero Matricule :BBB satus :en attente

Numero Matricule :BBbC satus :en attente

Numero Matricule :BBbCRAAAAAARR satus :en attente

Numero Matricule :BBbCRRR satus :en attente

Numero Matricule :CCC satus :en attente

Numero Matricule :DDD satus :en attente

Numero Matricule :EEE satus :en attente

Numero Matricule :FFF satus :attente pieces

Numero Matricule :Rif2017 satus :en attente

Date debute

2017-02-12

Date Fin

2017-05-30

chercher

cumulée

VehiculeMt

UPEC

ESISE

ÉCOLE SUPÉRIEURE D'INGÉNIEURS

PARIS-EST CRETEIL