

Université de La Mannouba
Ecole Nationale de Science de l'Informatique



Rapport du projet de fin d'été

Sujet :

**Application Android pour l'entretien et la maintenance
des pièces de voiture**

Réalisé par :

TABKA M'hamed

TRIFI AmanAllah

Encadré par :

MEZGHANNI Dhafer

Appréciations de l'encadrant

Remerciement

Il est nécessaire, avant de présenter notre travail, d'exprimer notre reconnaissance envers les personnes qui nous ont aidé de près ou de loin et qui nous ont apporté leur soutien et au premier lieu Mr MEZGHANNI Dhafer qui n'a ménagé aucun effort pour nous encadrer.

Nous tenons également à remercier les professeurs de l'ENSI pour tout le savoir qu'ils nous ont transmis et qui sans eux ce stage n'aurait pas eu lieu.

Nous tenons également à exprimer notre gratitude aux membres du jury pour l'attention qu'ils porteront à l'examen de notre projet.

Résumé

Notre cursus en tant qu'étudiant à l'Ecole Nationale des Science de l'Informatique prévoit un stage ou projet au cours de l'été qui suit la première année, c'est pour cela qu'on a effectué un projet sur la plateforme Android.

Table des matières

I -Etude préliminaire.....	5
1- présentation.....	5
2- Présentation de l'Android.....	5
3- Présentation du langage java.....	5
4- Présentation de SQLite.....	6
5- Problématique.....	6
6- étude de l'existant.....	6
7- solution proposée.....	6
 II – Analyse des besoins.....	 7
1- identification des acteurs.....	7
2- besoins non fonctionnels.....	7
3- besoins fonctionnels.....	7
4- besoins optionnels.....	7
 III-conception.....	 8
1-description du sujet.....	8
2-conception général.....	8
 IV -Réalisation.....	 9
1-Environnement du travail.....	9
2-Pièces de rechange	10
3-Résultats obtenus	11
 V-Conclusion et perspectives.....	 17
 VI-Netographie.....	 18

Table des figures

- 1 - Gabarit de notre application
- 2 - Architecture de l'application
- 3 – Tableau des pièces de rechange
- 4 - L'interface de l'ouverture de l'application
- 5 - L'interface du menu principale
- 6 - L'interface où est mise à jour la distance parcourus en kilomètre
- 7 - L'interface du formulaire
 - 1 – première figure
 - 2 – deuxième figure
- 8 - L'interface du diagnostic résultant
 - 1 – première figure
 - 2 – deuxième figure
- 9 – La notification

Chapitre 1

I -Etude préliminaire

Introduction

Dans ce chapitre, nous présentons notre application, les langages utilisés et l'étude préalable.

1 – présentation du sujet

C'est une application Android qui rappelle à l'utilisateur les dates où il doit changer les pièces de sa voiture (filtre à huile, filtre à air, huile...) en se basant sur la dernière date du changement et du nombre des kilomètres roulés. Lorsque la date du changement d'une pièce donnée est atteinte, une notification s'affiche.

2 - Présentation de l'Android

Android, prononcé *Androïd*, est un système d'exploitation mobile pour smartphones, tablettes tactiles, PDA et terminaux mobiles. C'est un système open source utilisant le noyau Linux. Il a été lancé par une startup du même nom rachetée par Google en 2005. D'autres types d'appareils possédant ce système d'exploitation existent, par exemple des téléviseurs, des radio-réveils, des montres connectées, des autoradios et même des voitures.⁽²⁾

3 - Présentation de langage JAVA

Le langage **Java** est un langage de programmation informatique orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au *SunWorld*.

La société Sun a été ensuite rachetée en 2009 par la société Oracle qui détient et maintient désormais Java.

La particularité et l'objectif central de Java est que les logiciels écrits dans ce langage doivent être très facilement portables sur plusieurs systèmes d'exploitation tels que UNIX, Windows, Mac OS ou GNU/Linux, avec peu ou pas de modifications. Pour cela, divers plateformes et frameworks associés visent à guider, sinon garantir, cette portabilité des applications développées en Java. ⁽²⁾

4 - Présentation de SQLite

SQLite est une bibliothèque écrite en C qui propose un moteur de base de données relationnelle accessible par le langage SQL. SQLite implémente en grande partie le standard SQL-92 et des propriétés ACID.

Contrairement aux serveurs de bases de données traditionnels, comme MySQL ou PostgreSQL, sa particularité est de ne pas reproduire le schéma habituel client-serveur mais d'être directement intégrée aux programmes. L'intégralité de la base de données (déclarations, tables, index et données) est stockée dans un fichier indépendant de la plateforme. ⁽²⁾

5 – Problématique

Les durées de vie des pièces de rechanges ne sont pas égaux exemple la durée de vie de filtre à huile est 12 mois alors que la durée de vie des galetés est 5 ans ! C'est pourquoi plusieurs propriétaire des voitures oublient les date où il faut changer ces pièces : le huile, les filtres, les pneus, les galetés...

6 - étude de l'existant

La majorité des propriétaires des voitures utilisent des blocs note pour se rappeler la date d'échange ou le kilométrage du dernier changement d'une certaine pièce.

7-solution proposée

Dans ce présent rapport, nous présenterons notre solution qui consiste en une application Android rappelant l'utilisateur des dates de changement des pièces et le nombre de kilomètres parcourus depuis le dernier changement.

Chapitre 2

II -Analyse des besoins

Introduction

Dans cette partie nous présentons l'identification des acteurs, des besoins fonctionnels et non fonctionnels.

1 -Identification des acteurs

Notre application est dédiée aux utilisateurs des voitures et spécifiquement aux utilisateurs d'Android.

2 –Besoin fonctionnel

- Consulter les dates et les kilométrages des derniers changements de toutes les pièces et communiquer à l'utilisateur combien il lui reste du temps et combien de kilomètres à parcourir avant le prochain changement.
- Recevoir des notifications lorsque la date où il faut changer une pièce donnée est atteinte.
- Rappeler les vérifications que l'utilisateur doit effectuer chaque jour.

3 –Besoin non fonctionnel

- L'application doit offrir une interface conviviale et ergonomique exploitable par l'utilisateur en envisageant toutes les interactions possibles à l'écran du support tenu.
- L'application doit optimiser les traitements pour avoir un temps de génération de schéma raisonnable.
- L'application doit être fonctionnelle indépendamment de toutes circonstances pouvant entourer l'utilisateur.
- Le code de l'application doit être lisible et compréhensible afin d'assurer son état évolutif et extensible par rapport aux besoins du marché.

Chapitre 3

III -Conception

Introduction

Dans cette partie nous présentons une modélisation de notre future application : gabarit et architecture.

1) Description du sujet

Le sujet se décompose en plusieurs parties :

- Recherche et documentation sur les pièces de rechanges
- Analyse et conception de la base de données.
- Développement de l'ensemble des interfaces de gestions de données des exploitants et accessoires.

2) Conception général

Cette figure présente le gabarit des interfaces de notre application, au-dessus c'est le nom de l'application, en bas la zone des boutons et le contenu de l'interface au milieu.



Fig 2 - Gabarit de notre application

Cette figure présente l'architecture de notre application. Au début on a une interface où il y a un menu qui nous dirige vers un formulaire ou vers une interface où on peut mettre à jour le nombre de kilomètres roulés et après on arrive à l'interface où on peut consulter tous les pièces de rechange, la période et les kilomètres restant avant le prochain changement. D'autre part lorsqu'une notification s'affiche elle nous dirige directement vers la dernière interface.

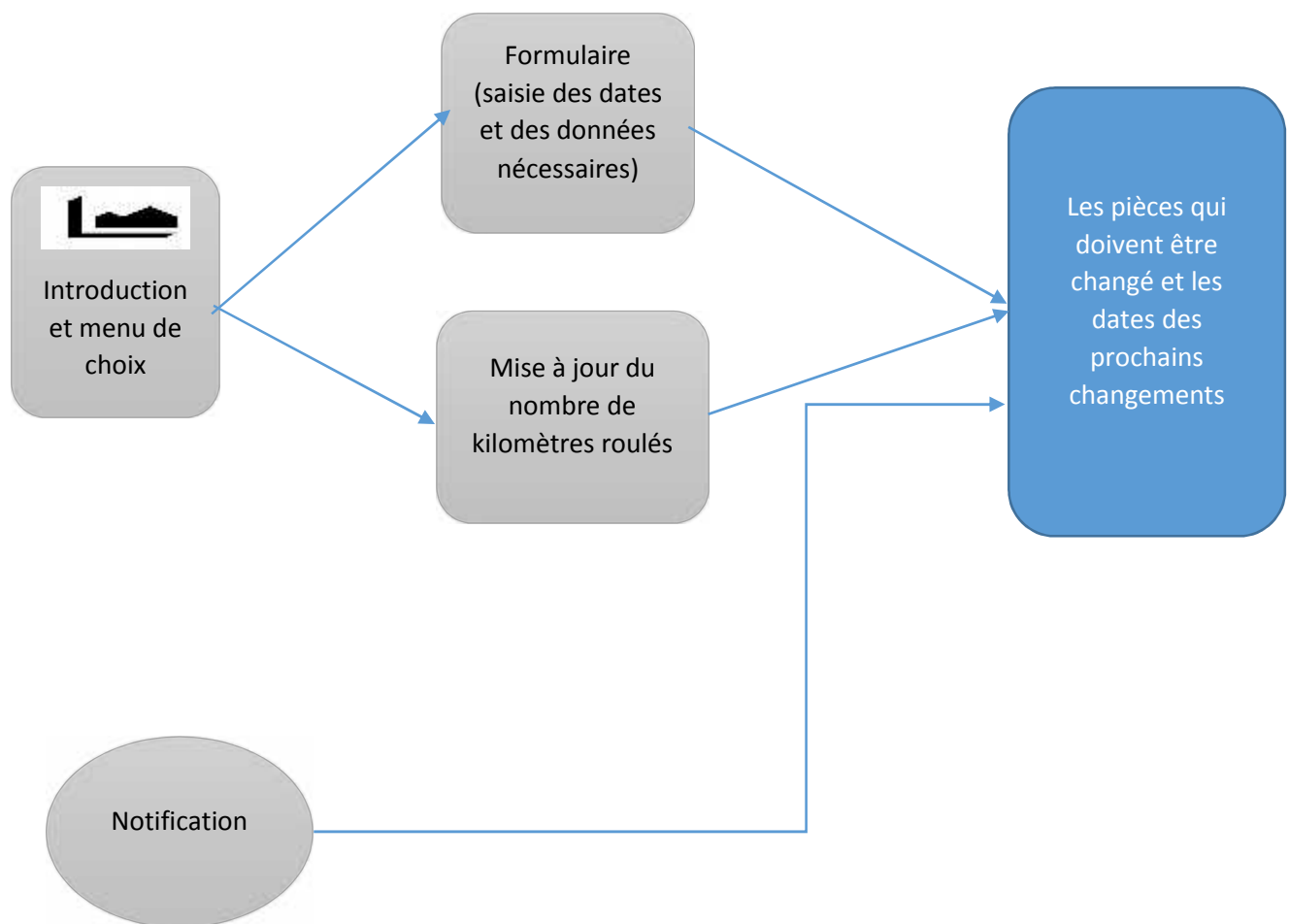


Fig 3 - Architecture de l'application

Chapitre 4

IV -Réalisation

Introduction

Ce chapitre est consacré à la présentation de l'environnement matériel et logiciel utilisés pour le développement de la solution proposée. Nous donnons ensuite une description des résultats aboutis approuvés par quelques imprimés écrans.

1-Environnement du travail

- **Environnement matériel**

L'ordinateur qui a servi pour la réalisation de cette application possède les caractéristiques suivantes :

- Processeur : Intel CORE i5
- Mémoires : 6 GO RAM
- Disque dur : 500 GO
- Système d'exploitation : Windows Seven

- **Environnement logiciel**

- Outils de développement : eclipse
- Base de données : SQLite

3-Les pièces de rechange

On présente de ce tableau les pièces de rechange et les périodes et les kilométrages max pour les changer.

Les Pièces	Période max pour le changement	Kilométrage max pour le changement
Les bougies d'allumage	24 mois	30 000km
Le filtre à huile	12 mois	10 000 km
Le filtre à air	12 mois	20 000km
Le filtre d'habitacle	12 mois	-
L'huile motrice	12 mois	10 000km
L'huile de la boîte vitesse	12 mois	30 000km
La courroie crantée	5 ans	-
Les galets	5 ans	-
La pompe à eau	5ans	-
LA SUSPENSION	2 ans	20 000km
Le disque frein	2 ans	30 000km
Les pneus	-	-
La visite technique	12 mois ou 6mois	-
L'assurance	12 mois ou 6mois	-

Fig 4 – Tableau des pièces de rechange ⁽³⁾

4-La base de données

Notre base de données se compose d'un tableau de trois colonnes une pour les pièces, une pour les dates et la dernière pour les nombres de kilomètres roulés lors du dernier changement d'une telle pièce.

5-Résultats obtenus

On a abouti à une application exécutable qui permet à l'utilisateur de se rappeler des dates des changements des pièces de sa voiture.

En premier lieu on aura une interface de démarrage. Au cours de laquelle il y'aura une connexion à la base de données.



Fig 5 - L'interface de l'ouverture de l'application

- Ensuite un menu qui permet à l'utilisateur de choisir la tâche à réaliser :
- Aller au formulaire pour saisir les dates des dernières maintenances effectuées ou les mettre à jour.
 - Mettre à jour le nombre de kilomètres parcourus.



Fig 6 - L'interface du menu principale

Le premier menu permet de saisir la nouvelle valeur du nombre de kilomètres parcourus, qui additionné aux autres éléments existants dans la base de données permettra lors de l'appui sur le bouton "Mettre à jours" de faire le calcul nécessaire :

Kilometrage

Veuillez saisir le dernier kilometrage:

125598

Mettre à jour

Fig 7 - L'interface où est mise à jour la distance parcourus en kilomètre

Le deuxième menu renvoi à un formulaire à remplir de données qui vont être enregistrés dans la base de données et qui serviront dans le calcul des prochaines dates des changements des pièces :

Formulaire

Marque de la voiture :

Marque de la voiture

Date de dernier vidange :

September 2014

S	M	T	W	T	F	S
36	31	1	2	3	4	5
37	7	8	9	10	11	12
38	14	15	16	17	18	19
39	21	22	23	24	25	26
40	28	29	30	1	2	3
41	5	6	7	8	9	10

le Killometrge de dernier vidange :

killometrage

Fig 8 - 1 - L'interface du formulaire



Fig 8 -2 - L'interface du formulaire

Après la saisie ou la mise à jour des différentes données, l'application renvoi au diagnostic affichant la liste complète des pièces nécessitant une maintenance et leurs états ainsi qu'une estimation de la date du prochain changement et/ou le nombre de kilomètres restant à parcourir.



Fig 9- 1 - L'interface du diagnostic résultant



Fig 9- 2 - L'interface du diagnostic résultant

L'application affiche aussi une notification alertant l'utilisateur l'approche ou le dépassement de la date de changement d'une pièce donnée.

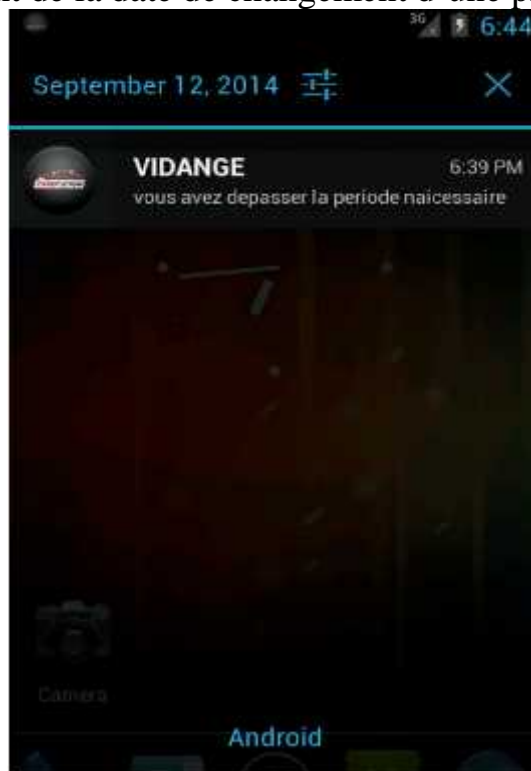


Fig 10 - Notification

V-Conclusion et Perspectives

Au cours de ce travail, nous avons mené une recherche sur les voitures et spécifiquement sur les pièces de rechange, et on a utilisé le langage java et balise XML.

Notre application permettra de :

- Se rappeler des derniers changements des pièces de rechange, de dernière visite technique, la date de fin de l'assurance, etc....
- Les vérifications quotidiennes.
- Alerté l'utilisateur de d'expiration de tel pièce.

Malgré son intérêt comme solution indispensable nous ne pouvons pas prétendre que ce travail est une solution complète et définitive car notre application se base sur les données entrées par l'utilisateur, par exemple les nombre de kilomètres parcourus.

On a appris à travailler en groupe et de savoir gérer le temps et décomposer le problème et surtout à répartir les tâches à réaliser ce qui nous a beaucoup facilité les choses.

VI- Netographie

- (1) www.openclassrooms.com
- (2) www.wikipedia.com
- (3) www.fiches-auto.fr