



**République Algérienne Démocratique et
Populaire**



**Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique**

**UNIVERSITÉ ECHAHID HAMMA LAKHDAR
ELOUED**

**FACULTÉ SCIENCES EXACTES
Mémoire de fin d'étude**

LICENCE ACADEMIQUE

Domaine: Mathématiques et Informatique

Filière: Informatique

Spécialité: système informatique

Présenté par:

Lyzidi Amor

Sichi Hamza

Zaabi Mouloud Seif Eddine

**Conception et Modélisation d'un system
de gestion de parking**

Soutenu le :

Membres du jury :

Examineur 1 : Mr. OTHMENI SAMIR

Examineur 2 : Mr. BERJEOUH CHAFIQUE

Encadreur : Mr. Khalaifa Abdennaceur

Année Universitaire 2015 - 2016

Remerciements

Nous tenons tout d'abord à remercier Mr. Kkhalaifa Abdelnaceur
notre encadreur pour tout le soutien, l'aide
l'orientation et pour ses précieux conseils et ses encouragements lors de la réalisation de
notre mémoire.

Nous remercierons également toutes les personnes qui de près ou de loin ont
participé à l'élaboration de notre travail.

Enfin Nous remercierons nos amis et collègues durant ces trois années d'études
passées ensemble
dans les meilleurs moments comme dans les pires.

Résumé

La gestion d'un parking d'automobile consiste à gérer l'entrée et la sortie du parking. A l'entrée il faut sauvegarder L'heure d'entrée s'il y a des places vides, et à la sortie il faut calculer le montant à payer selon la durée de demeure dans le parking. Le système doit en plus offrir d'autres fonctionnalités par exemple : réservation à l'avance, remise selon la fidélité, etc.

ملخص

إدارة موقف للسيارات يتمثل في إدارة دخول وخروج السيارات للموقف. عند المدخل يجب حفظ وقت الدخول إذا كانت هناك أماكن فارغة، وعند الخروج يجب حساب المبلغ المستحق حسب مدة مكوث السيارة داخل الموقف. يجب على النظام كذلك تقديم امتيازات أخرى مثل : الحجز المسبق ، والخصم للزبائن الأوفياء ، الخ

Abstract

The management of a parking is to manage the entry and exit of parking. At the entrance the system must save input time if there are empty places, and at exit the system must calculate the amount of money to pay by calculating the period of stay. The system must also offer other advantages such as: advance booking , delivery according fidelity, etc.

Sommaire

I. Introduction General	1
I.2.Problématique:	1
I.3.Objectifs :	1
II. Chapitre I Etude Conceptuel	
I.1. Introduction:	3
I.1.1.Le Processus 2TUP :	3
I.1.2.Définition :	3
I.2. Capture des besoins fonctionnels :	4
I.2.1. Identification des acteurs :	4
I.2.2. 1 Les cas d'utilisation de notre système :	5
I.2.2.2.Les Diagramme de Séquence de notre système :	6
I.2.2.3.1.Diagramme de séquence Authentification:	6
I.2.2.3.2.Ajouter Les client.....	7
I.2.2.3.3.Supprimer les client	8
I.2.2.3.4.modifier les client.....	9
I.2.2.3.5.Entrée client	10
I.2.2.3.6.Sortie client	11
I.3.Diagramme de class général :	12
I.3.1. Conception détaillée:	13
I.3.1. Description du modèle des classes :	13
I.4.Conclusion:	14
III. Chapter II Réalisation de system	
II.1. Introduction:	16
II.2. Choix des outils de développement:	16
II.3. Description de system:	17
II.3.1.Description general:	17
II.3.2.Description les Formulaire du gestion :	18
II.3.2.1.Interface d'accueil :	18
II.3.2.2. Fenêtre de paramétrage:	18
II.3.2.3.Fenêtre de gestion client:	19
II.3.2.4. Formulaire d'ajout :	19

II.3.2.5.Fenêtre de Stationnement :	20
II.3.2.6Formulaire de Modification :	21
II.3.3.Conclusion :	22
Conclusion Générale	23
Perspective	23
Références bibliographique	24

Liste de figure :

Figure 1 : Diagramme de cas d'utilisation	5
Figure 2:Diagramme de séquence Authentification	6
Figure 3:Diagramme de séquence ajouter les Client.....	7
Figure 4:Diagramme de séquence D' supprime les Client.....	8
Figure 5:Diagramme séquence modification de client.....	9
Figure 6: Diagramme de séquence D' Entrée les Client	9
Figure 7: Diagramme de séquence Sortir les Client.....	11
Figure 8: Diagramme de Class	11
Figure 9: page de accueil.....	18
Figure 10:page de paramétrage.....	18
Figure 11: page de Management Client.....	19
Figure 12: page de Ajoute Client	19
Figure 13: page de Reserve automobile	20
Figure 14: page de Sortir automobile	20
Figure 15: page de Modifier Client	21

Listes des Tableaux :

Tableau 1 : Les acteurs de système.....4

Tableau 2 : Description du modèle des classes.....13

I. Introduction Général

I.1.Introduction

La croissance du trafic automobile et ses conséquences sur la congestion des agglomérations ainsi que sur les stratégies de localisation des activités et les politiques publiques d'aménagement urbain, ont consacré les liens indéniables entre politiques publiques de mobilité d'une part et d'aménagement du territoire d'autre part.

Jusqu'il y a peu, les politiques urbaines mises en œuvre ont plutôt cherché à répondre aux besoins croissants en mobilité en adaptant le plus souvent la ville à l'automobile. Toutefois, les nuisances induites par ce mode de transport et la rareté de l'espace conduisent les pouvoirs publics à rechercher une meilleure cohérence entre les politiques de déplacement et les politiques d'aménagement du territoire.

Afin grand nombre des voitures , qui se grandit chaque jour , de façon illimite nous a donné une idée produire un programme pour facilite la tâche de stationnement les automobiles et administrer les stations d'une part et ordonner les voitures d'autre part. Ce programme c'est un exemple faisable et applicable de cette idée.

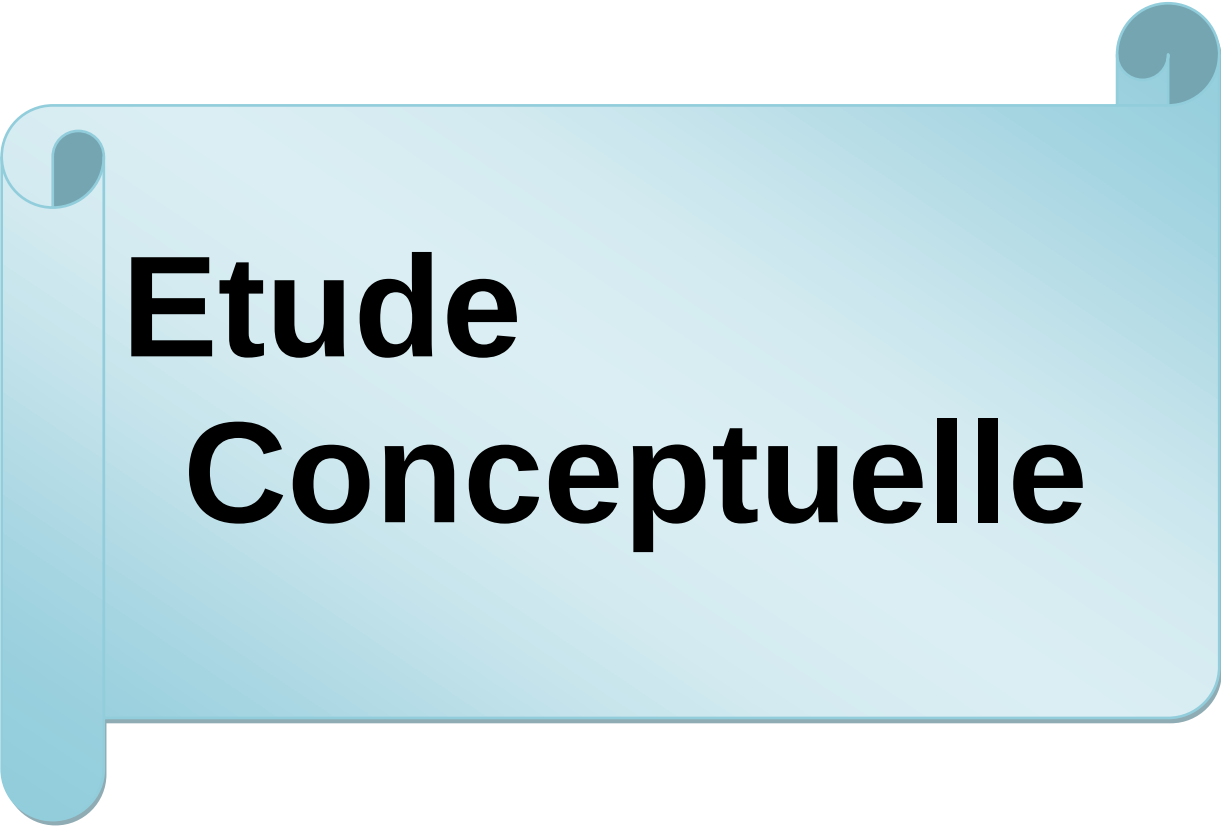
I.2.Problématique:

Chaque gestion de système rencontre des problèmes et des difficultés hors de travail :

- L'insatisfaction avec le client pour le prix .
- Problématique du développement du system gestion.
- la classification du clients par des catégorie .
- Quels sont les places disponible .

I.3.Objectifs :

L'objectif de ce programme est de trouver une solution racinée pour le stationnement d'automobiles, les administrer dans la ville et d'arrêter les stations des voiture ale atore.



Etude Conceptuelle

I.1. Introduction:

Dans ce chapitre, nous allons présenter le processus de développement de gestion .

Nous avons choisi le langage UML (UnifiedModelingLanguage) en raison de sa qualité de modélisation à objet, de son formalisme relativement simple et compréhensible qui permet de mieux visualiser et de manipuler l'ensemble des éléments constituant un système et sa rigueur qui est nécessaire au bon déroulement du projet ..

Dans ce chapitre, nous présenterons quelques définitions et concepts du langage UML. Par la suite, nous exposerons la modélisation de notre système et le processus Unifié (2TUP),

I.1.1.Le Processus 2TUP :**II.1.2.Définition :**

Le processus 2TUP (2Tracks Unified Processus) apporte une réponse aux contraintes de changement continu imposées aux systèmes d'information de l'entreprise. Idée de base du 2TUP : toute évolution imposée au SI peut se décomposer et se traiter parallèlement, suivant 2 axes (« 2 tracks ») : Un axe fonctionnel et Un axe technique

La réalisation du système consiste à fusionner les résultats des deux branches.

Dans notre cas en à besoin de concentré pour les deux cas Suivants :

➤**Capture des besoins fonctionnels .**

➤**Conception détaillée .**

I.2. Capture des besoins fonctionnels :

La capture des besoins fonctionnels est la première étape de la branche gauche du processus en Y. Elle sera complétée au niveau de la branche droite par la capture des besoins techniques pour préparer l'analyse au niveau de la branche gauche. Ce chapitre structure les exigences fonctionnelles ébauchées durant l'étude préliminaire. La technique des cas d'utilisation est la pierre angulaire de cette étape. Nous verrons successivement dans ce chapitre comment :

- Identifier les acteurs du système .
- Identifier, décrire et organiser les cas d'utilisation du système .
- Identifier les classes candidates du modèle d'analyse .

I.2.1. Identification des acteurs :

Un acteur est une catégorie d'utilisateur, il représente un rôle joué par une personne , un logiciel , un matériel , un automate qui exploite les données du système

Dans notre cas les acteurs sont :

Utilisateurs finaux	Description des besoins fonctionnels
Administrateur	Le system doit permettre à l'administrateur de : - Gestionnaire de Stationnaire(Entrée et sortir des voiture). - Gestionnaire de place (Ajoute et Supprime et visualiser les place). - Gestionnaire de client(Ajoute et Supprime et Modifier les Client). - questionnaire de tariff().

Tableau 1 : Les acteurs de système

I.2.2. 1 Les cas d'utilisation de notre système :

Après avoir identifié les acteurs du système, il est utile d'identifier les besoins de celui-ci et les fonctionnalités offertes pour chaque acteur. Pour ceci nous utilisons les diagrammes de cas utilisation afin d'illustrer des besoins, les diagrammes de séquence nous montrent les scénarios possible liées aux cas d'utilisations, et enfin le diagramme de classe nous donne une vue générale des entités et le lien qu'il y'a entre eux.

Nous avons regroupé les cas d'utilisation selon les besoins des acteurs de notre système, et par conséquent nous avons le détail suivant :

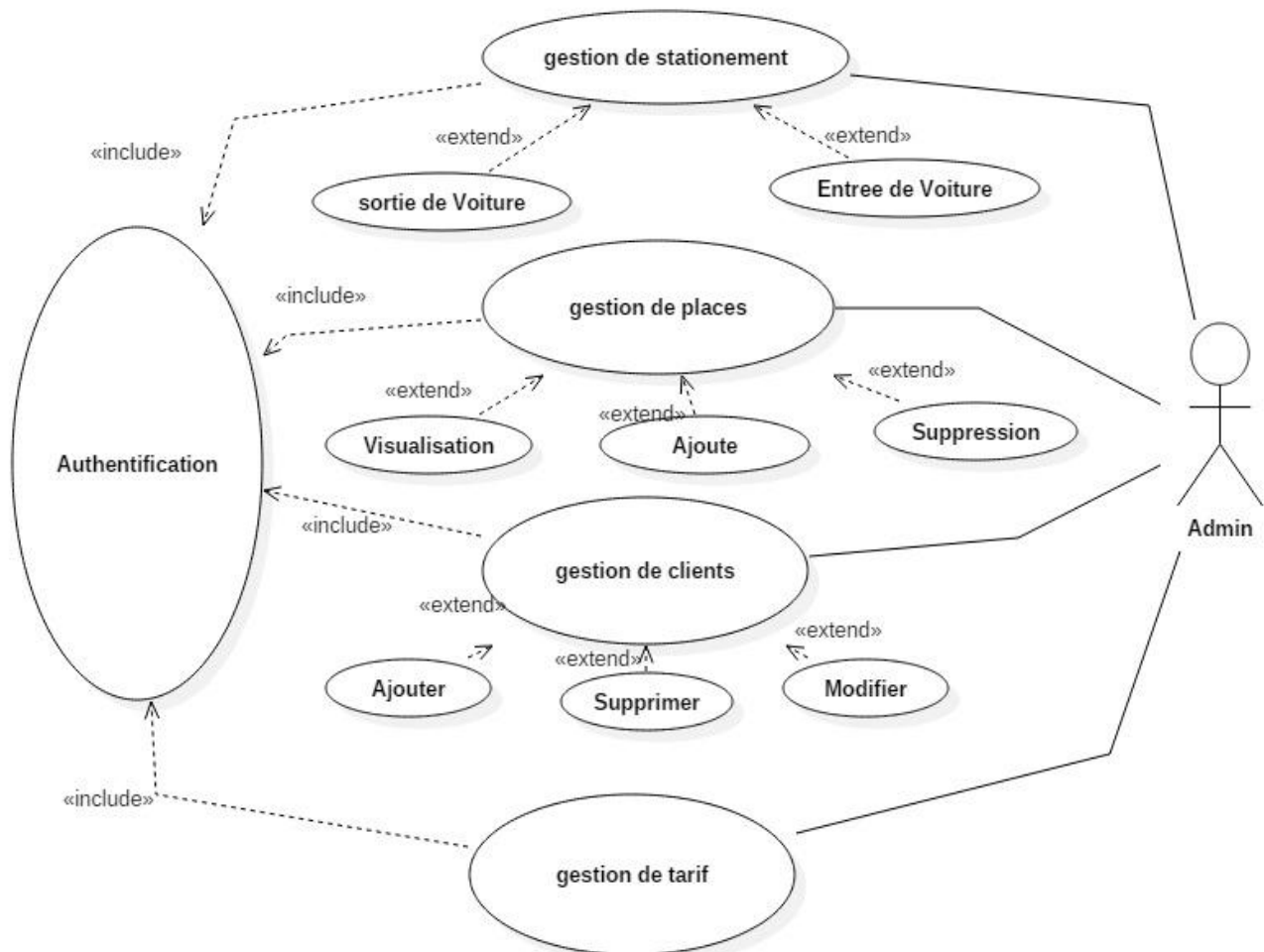


Figure 1 : Diagramme de cas d'utilisation

I.2.2.2. Les Diagramme de Séquence de notre système :

Les diagrammes de séquences mettent en valeur les échanges de messages (déclenchant des événements) entre acteurs et objets (ou entre objets et objets) de manière chronologique, l'évolution du temps se lisant de haut en bas. Chaque colonne correspond à un objet (décrit dans le diagramme des classes), ou éventuellement à un acteur, introduit dans le diagramme des cas. La ligne de vie de l'objet représente la durée de son interaction avec les autres objets du diagramme

I.2.2.3.1. Diagramme de séquence Authentification:

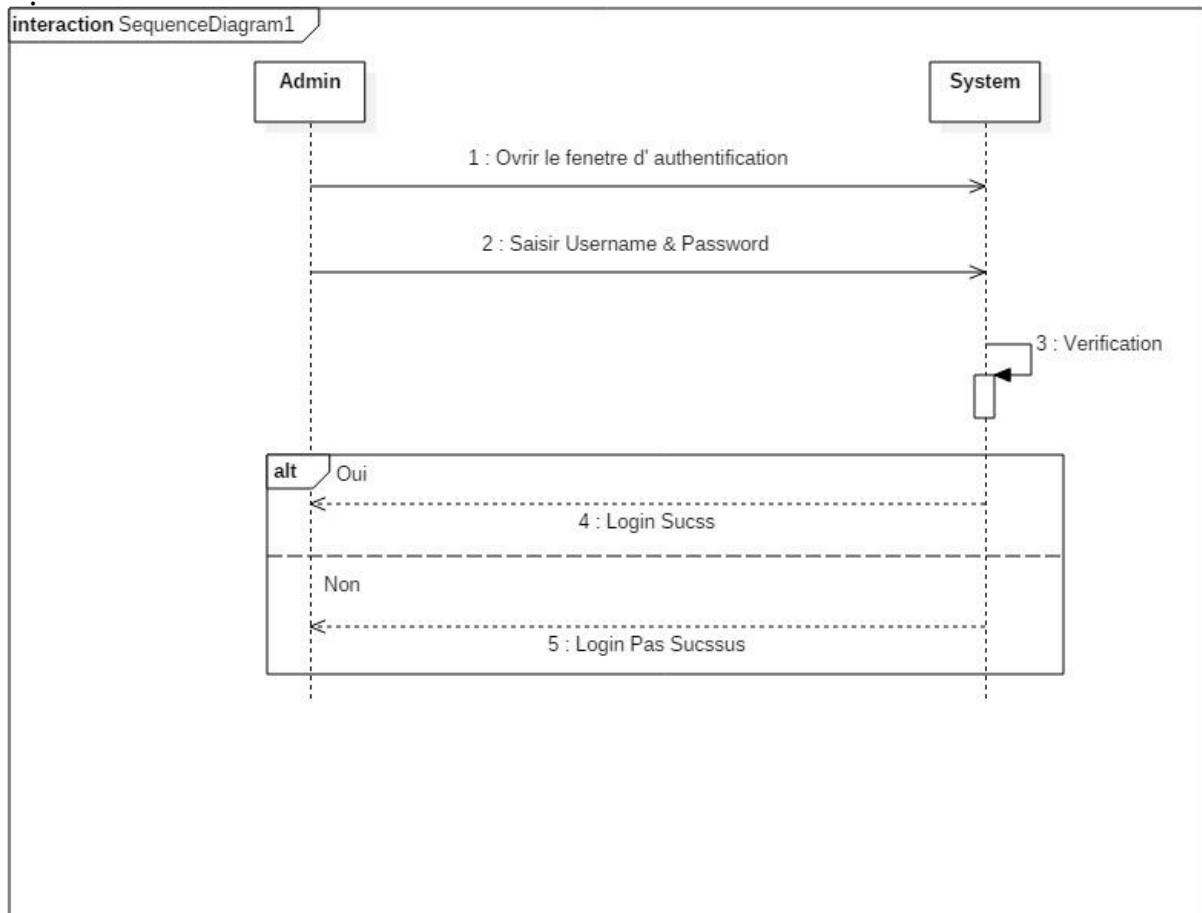


Figure 2: Diagramme de séquence Authentification

I.2.2.3.2.Ajouter Les client : Cette fonction permet D' ajouter les client au Système

Pre-condition: Administrateur doit s'authentifier

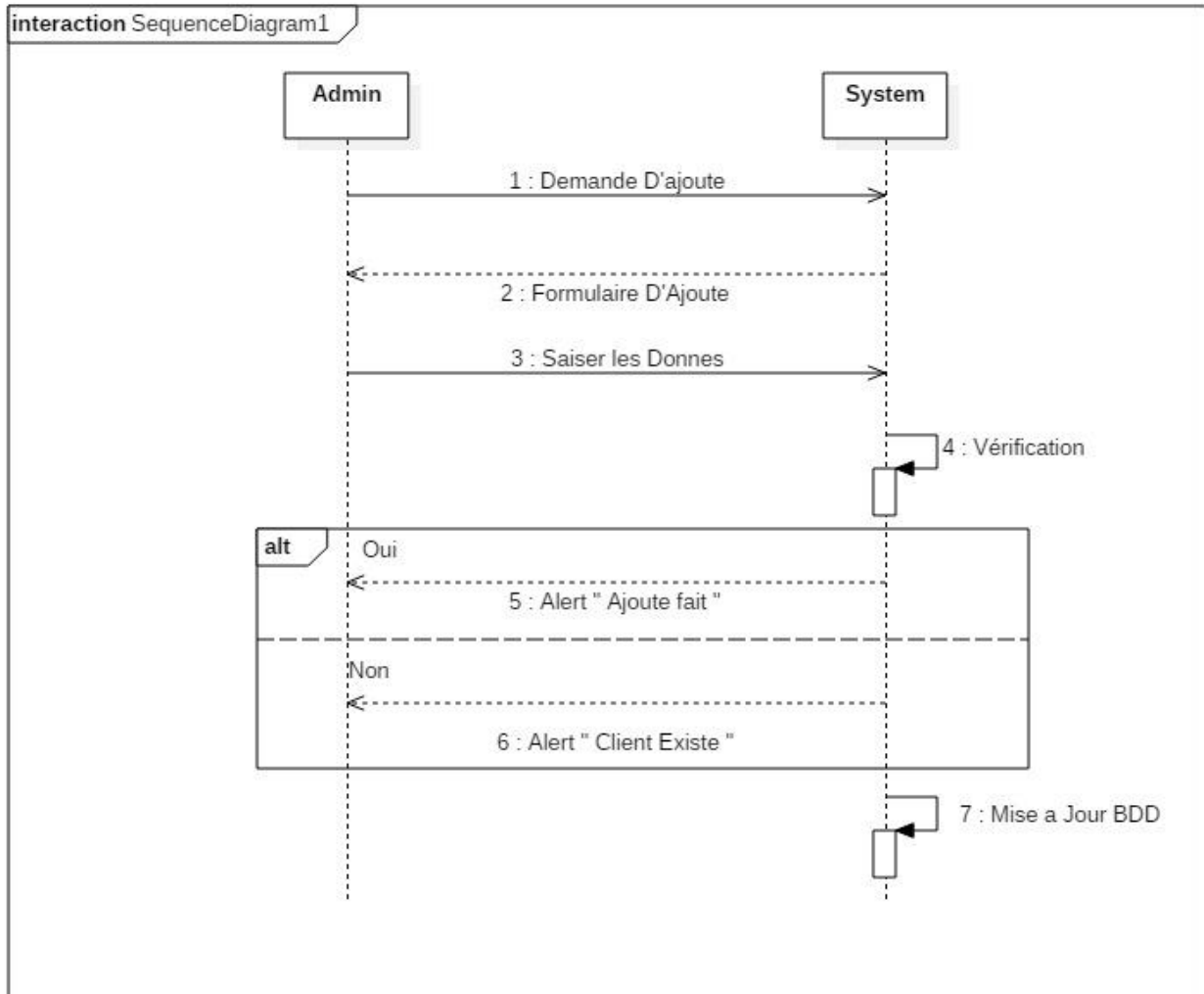


Figure 3:Diagramme de séquence ajouter les Client

I.2.2.3.3. Supprimer les client: Cette fonction permet à l'administrateur de supprimer les Client

Pre-condition: Administrateur doit S'authentifier

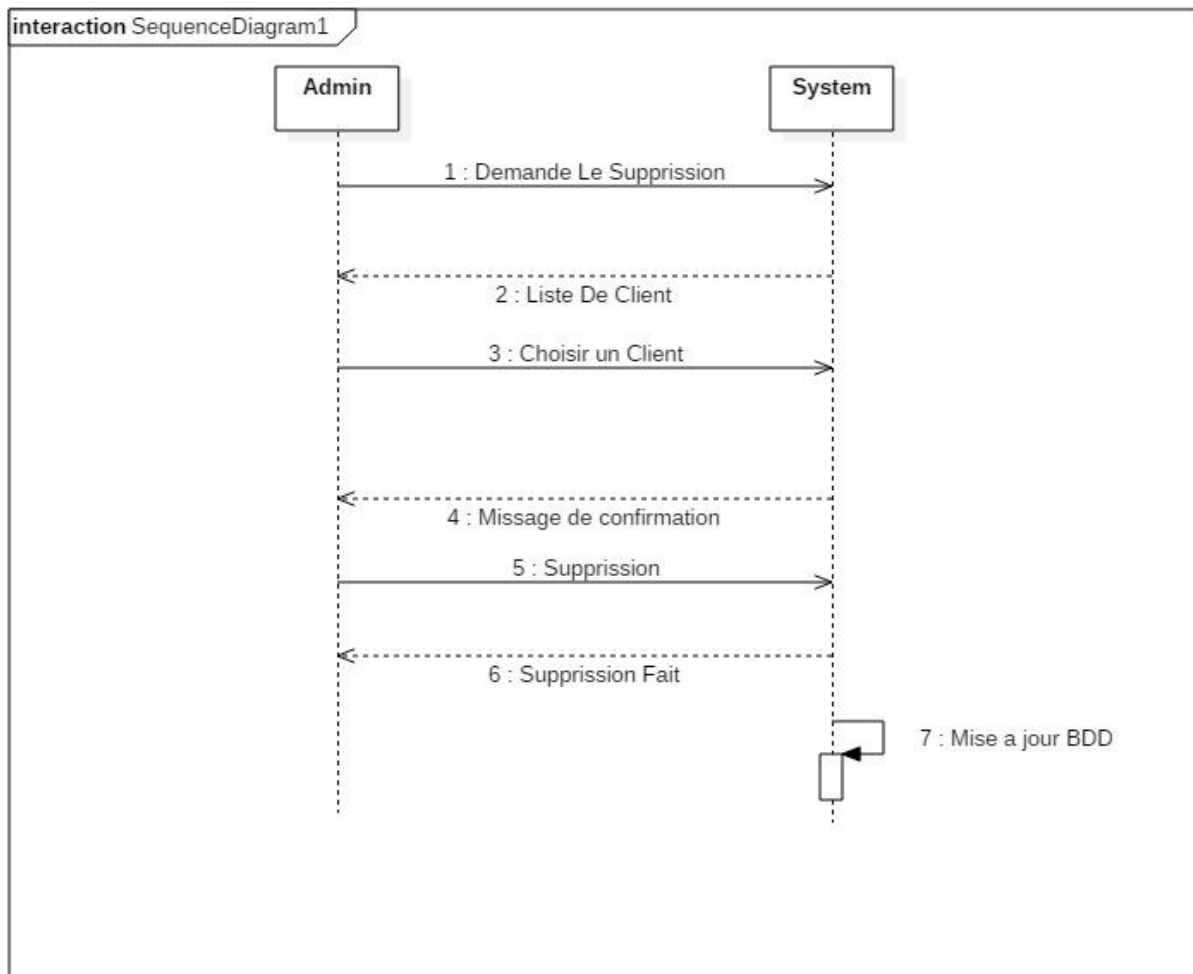


Figure 4:Diagramme de séquence D' supprime les Client

I.2.2.3.4.modifier les client: Cette fonction permet à l'administrateur de modifier les Client dans le Gestion .

Pre-condition: Administrateur doit S'authentifier.

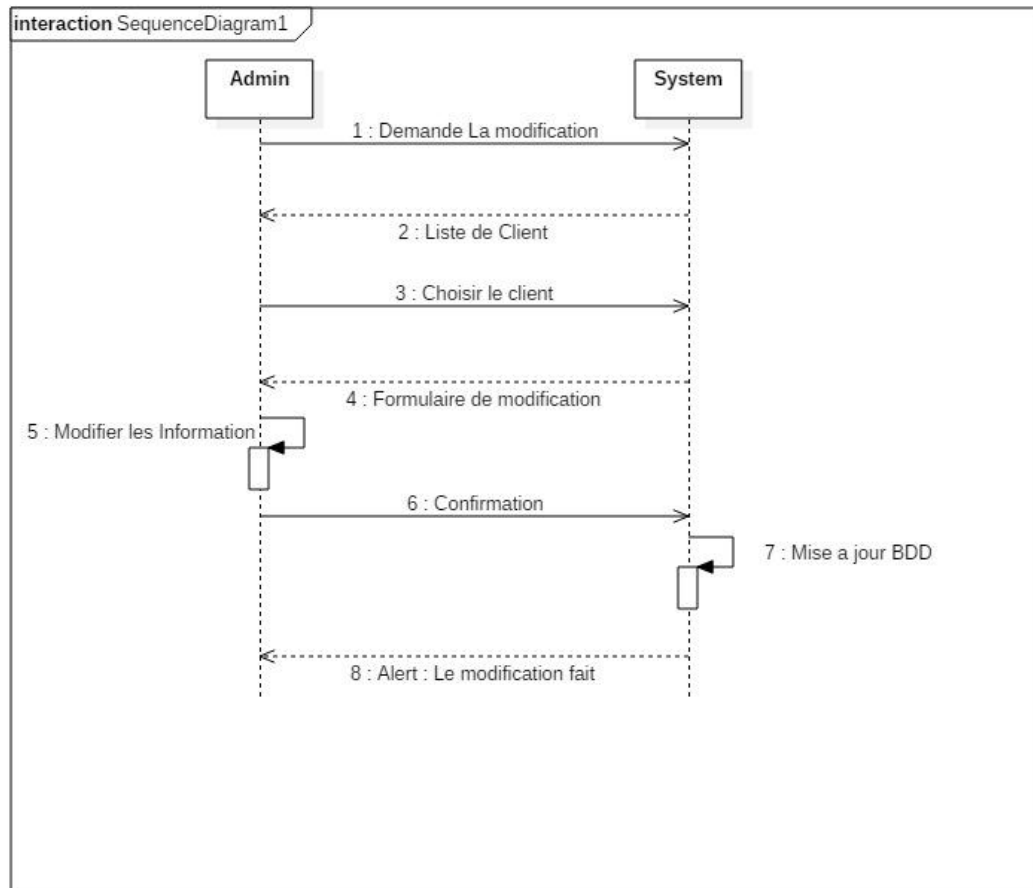


Figure5:Diagramme séquence modification de client

I.2.2.3.5. Entrée client: Cette fonction permet à l'administrateur de Entrée les Client.

Pre-condition: Administrateur doit S'authentifier

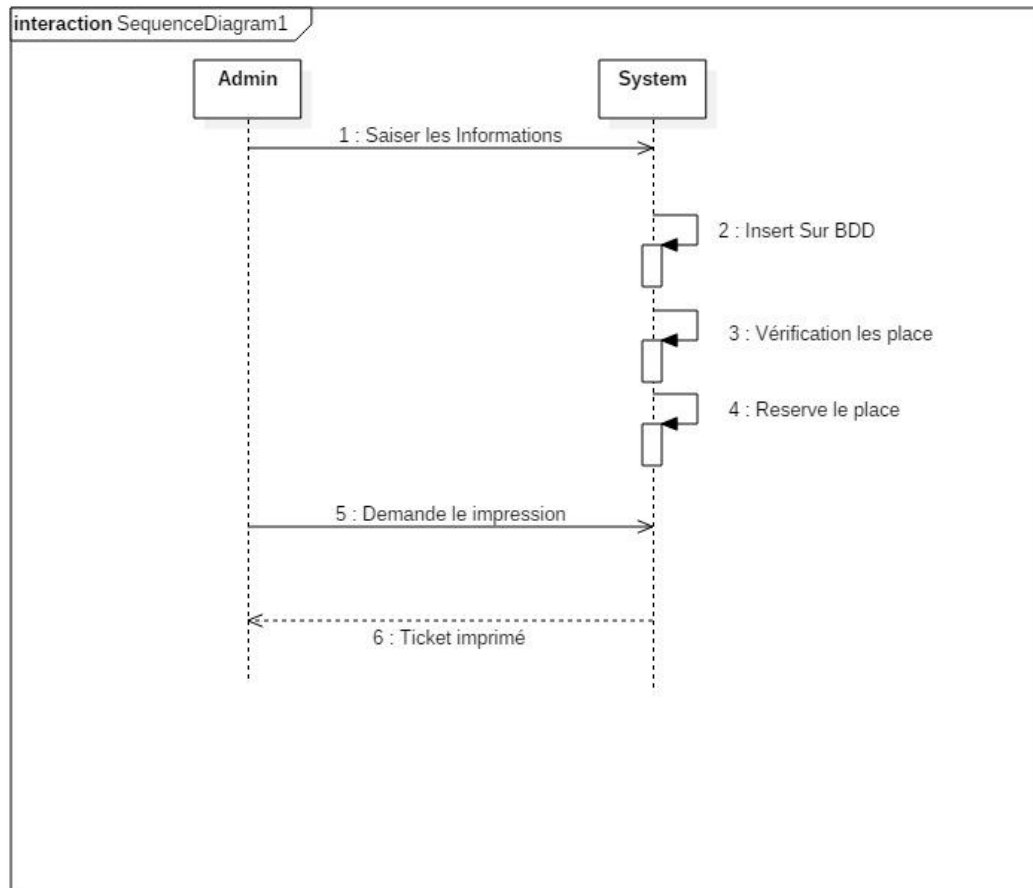


Figure 6: Diagramme de séquence D' Entrée les Client

I.2.2.3.6.Sortie client: Cette fonction permet à l'administrateur de Entrée les Client.

Pre-condition: Administrateur doit S'authentifier

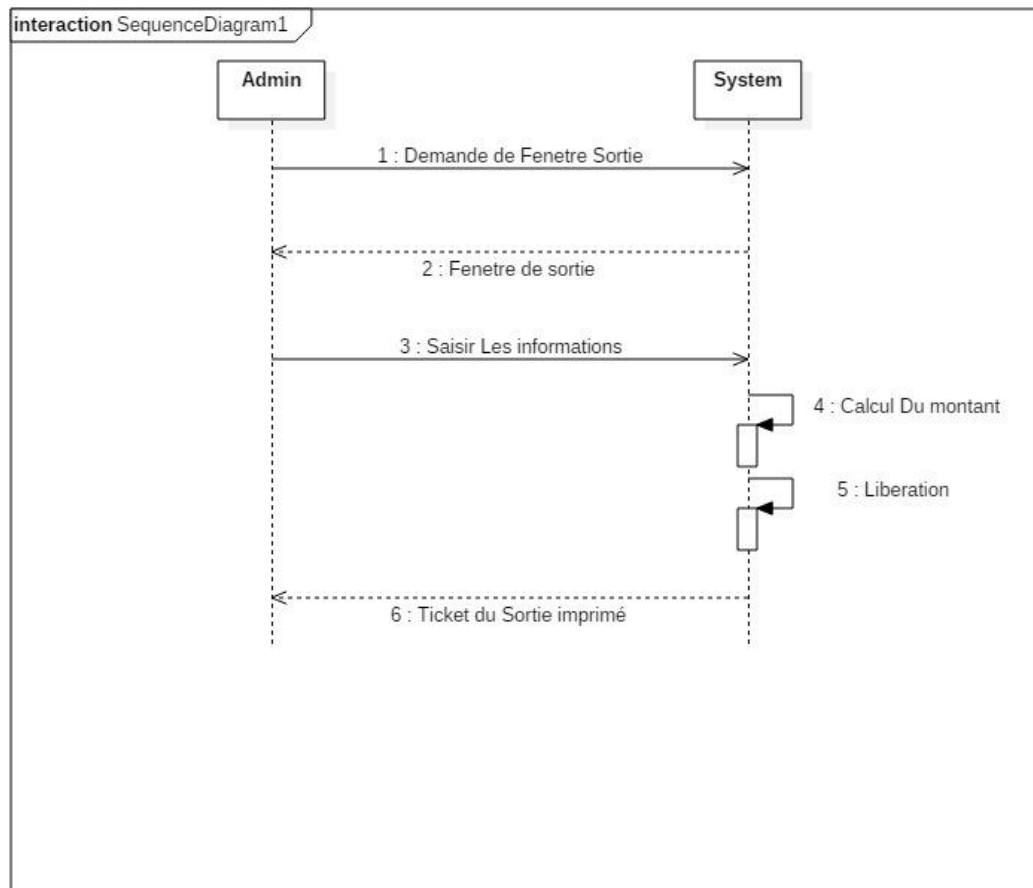


Figure 7: Diagramme de séquence Sortir les Client

I.3. Diagramme de class général :

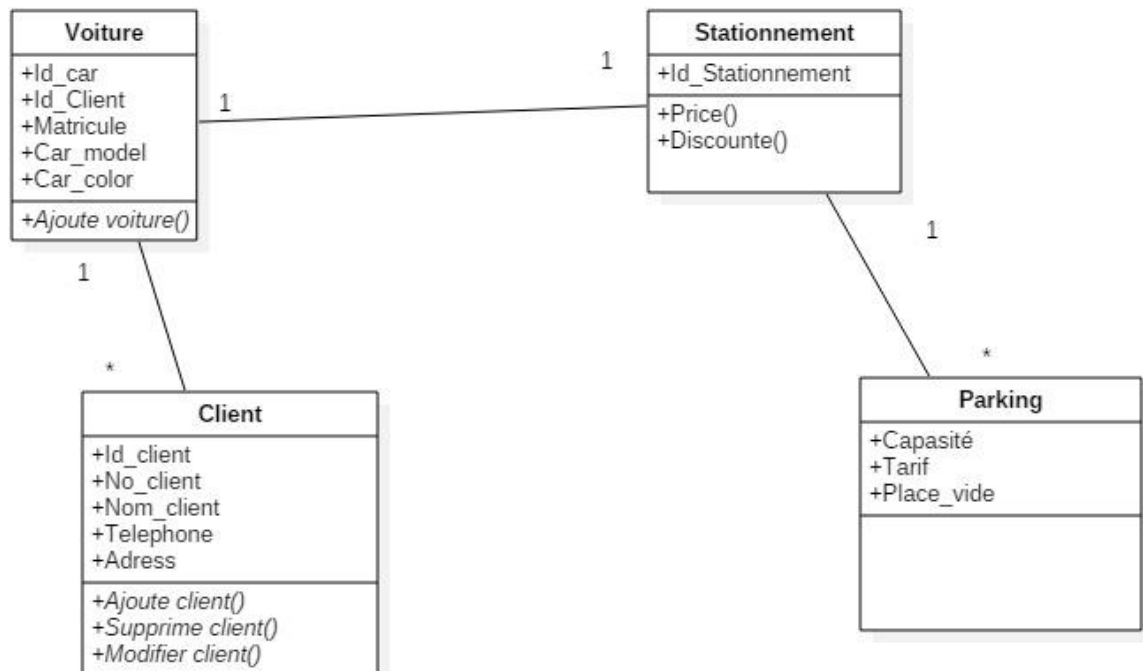


Figure 8: Diagramme de Classe

I.3.1. Conception détaillée:

La conception détaillée est la phase ultime de la modélisation qui consiste à construire et à documenter précisément les classes, les tables et les méthodes qui constituent le codage de la solution.

I.3.1. Description du modèle des classes :

Class	Attributs	Description	Type[Taille]
Voiture	Id_car	Identifiant numérique d'une voiture	Int
	IdClient	Identifiant numérique d'une Client	Int
	Matricule	Le matricule de voiture	Int
	Car_model	Le Model de voiture	Text
	Car_color	La colore de voiture	Text
Client	IdClient	Identifiant numérique d'une Client	Int
	No_Client	Numero de Client	Int
	Nom_Client	Le nom de Client	Text
	Telephone	Telephone de client	Int
	Adress	Adress de Client	Text
Stationnement	Id_Stationnement	Identification de Stationnement	Int
Parking	Capacity	Capacity de Parking	Int
	Tarif	Le Tarif	Int
	Place_vide	Les place vide	int

Tableau 2 : Description Du modèle des classes

I.4.Conclusion:

Dans ce chapitre, nous avons présenté une capture des digrammes utilisés lors de la modélisation suivi une démarche de développement. Les résultats de ce chapitre seront enrichis par de détails d'implémentation dans le chapitre suivant pour réaliser notre système

A decorative graphic consisting of a blue scroll with a white outline, featuring a vertical strip on the left and a horizontal strip at the top, both with rounded ends and a small circular detail at the corner.

Réalisation De Système

II.1. Introduction:

A ce stade du processus, les cas d'utilisation sont terminés, le problème a été analysé en profondeur. Nous avons défini une conception mieux appropriée aux besoins de system.

Dans cette phase nous allons présenter les outils de développement que nous avons utilisé, l'architecture matérielle mise en place, implémenter tous les cas d'utilisation.

II.2. Choix des outils de développement:

Pour réaliser notre application, nous avons utilisé le langage de programmation C# 2012 dédié à la POO afin de créer des applications exécutables et indépendantes dans l'environnement Client/serveur utilisant les bases de données de la plate-forme Microsoft.

Environnement Dia nous a permis dans la modélisation en nous produisant de diagrammes appropriés dans l'Orientée Objet. Pour produire des interfaces ergonomiques.

II.2.1. Le langage de programmation C#:

C# est un langage élégant, simple, de type sécurisé et orienté objet qui permet aux programmeurs en entreprise de créer une vaste gamme d'applications.

Le langage C# permet également de créer des composants de niveau système durables grâce aux fonctionnalités suivantes :

- Prise en charge totale de plate-forme/COM pour l'intégration du code existant.
- Robustesse grâce au garbage collection et à la sécurité de type.
- Sécurité assurée par des mécanismes d'approbation de code intrinsèque.
- Prise en charge intégrale des concepts de métadonnées extensibles.

Il est également possible d'inter opérer avec d'autres langages, sur plusieurs plates-formes, en reprenant les données héritées des premiers langages Basic, grâce aux fonctionnalités suivantes :

- Prise en charge intégrale de l'interopérabilité via les services COM+ 1.0 et .NET Framework avec un accès strict basé sur les bibliothèques.
- Prise en charge de XML pour l'interaction des composants Web.
- Versioning permettant de faciliter l'administration et le déploiement.

II.2.3.Choix du SGBD :SQLite

Ce qu'il faut bien garder à l'esprit, c'est que SQLite n'est pas vraiment un concurrent des serveurs de base de données relationnelles, c'est plus une extension de leur champ d'application à des applications offline (là ou avant elles n'étaient utilisé que sur des serveurs sur un modèle client/serveur).

SQLite peut aussi s'avérer très utile sur un site web pour créer des fonctionnalités qui ne sont pas directement liées au site (comme par exemple organiser un jeu concours avec un formulaire où les gens peuvent d'inscrire), évitant ainsi de polluer la base de données principale (fonctionnant sur MySQL par exemple) avec des données temporaires.

L'autre avantage est la simplicité : il n'y a aucune manipulation à faire, le fichier sqlite est créé automatiquement à la volée, toute la base est stockée dans un fichier unique qu'il est facile d'échanger en FTP...

En fait, SQLite peut être utilisé en remplacement de Microsoft Access, car il est bien plus performant, propre, portable et multiplateforme.

II.3. Description de systèm:

II.3.1.Description général:

Le system est un gestion administration , l'exécution se fait par les formulaire principal , c'est la formulaire d'accueil qui est réservée au un seul Administrateur, elle contient des formulaire faciles à manipuler , L'organisation de formulaire est réalisée en utilisant logiciel C# .

II.3.2.Description les Formulaire du gestion :

II.3.2.1.Interface d'accueil :

Toute personne désire d'utiliser le système est obligée de passer par cette page.

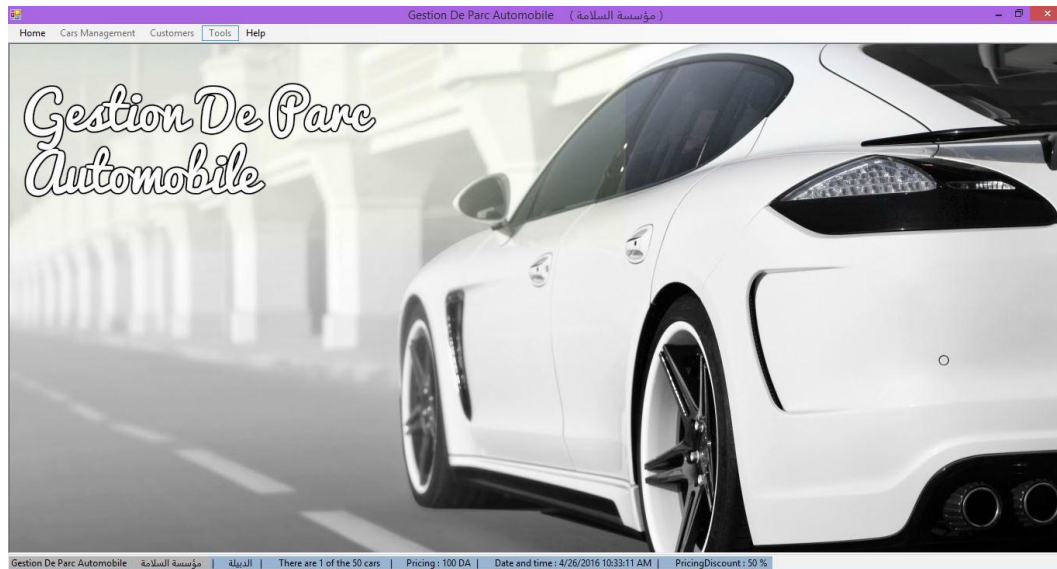


Figure 09: page de accueil

II.3.2.2.Fenêtre de paramétrage:

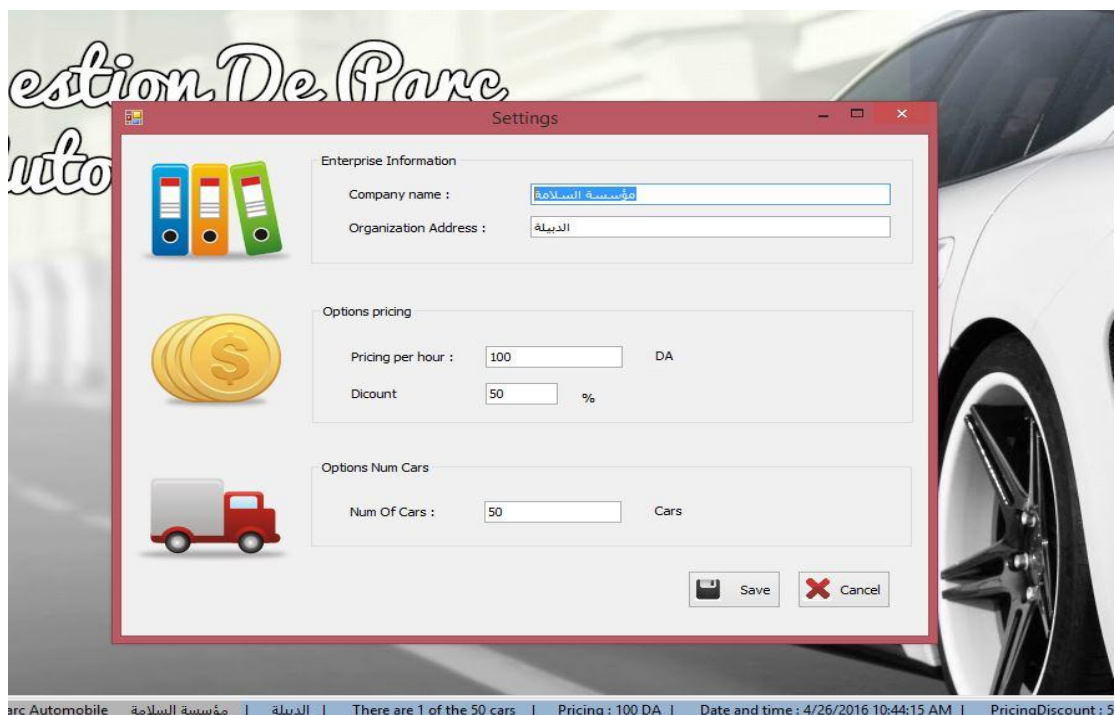


Figure 10: page de paramétrages

II.3.2.3.Fenêtre de gestion client:

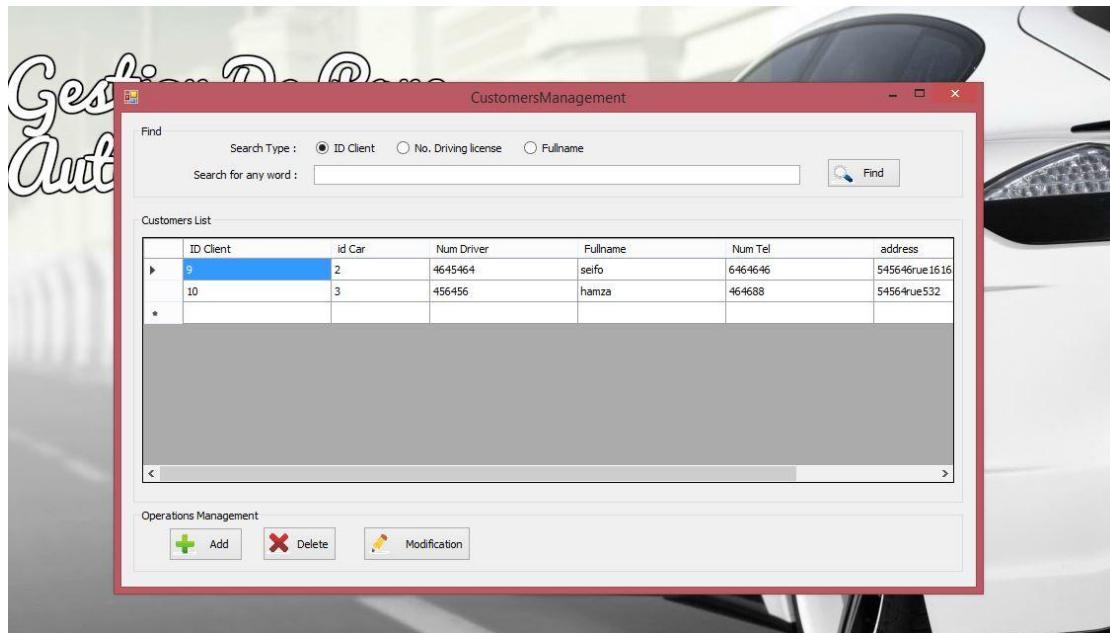


Figure 11: page de Management Client

III.3.2.4.Formulairesd'ajout :

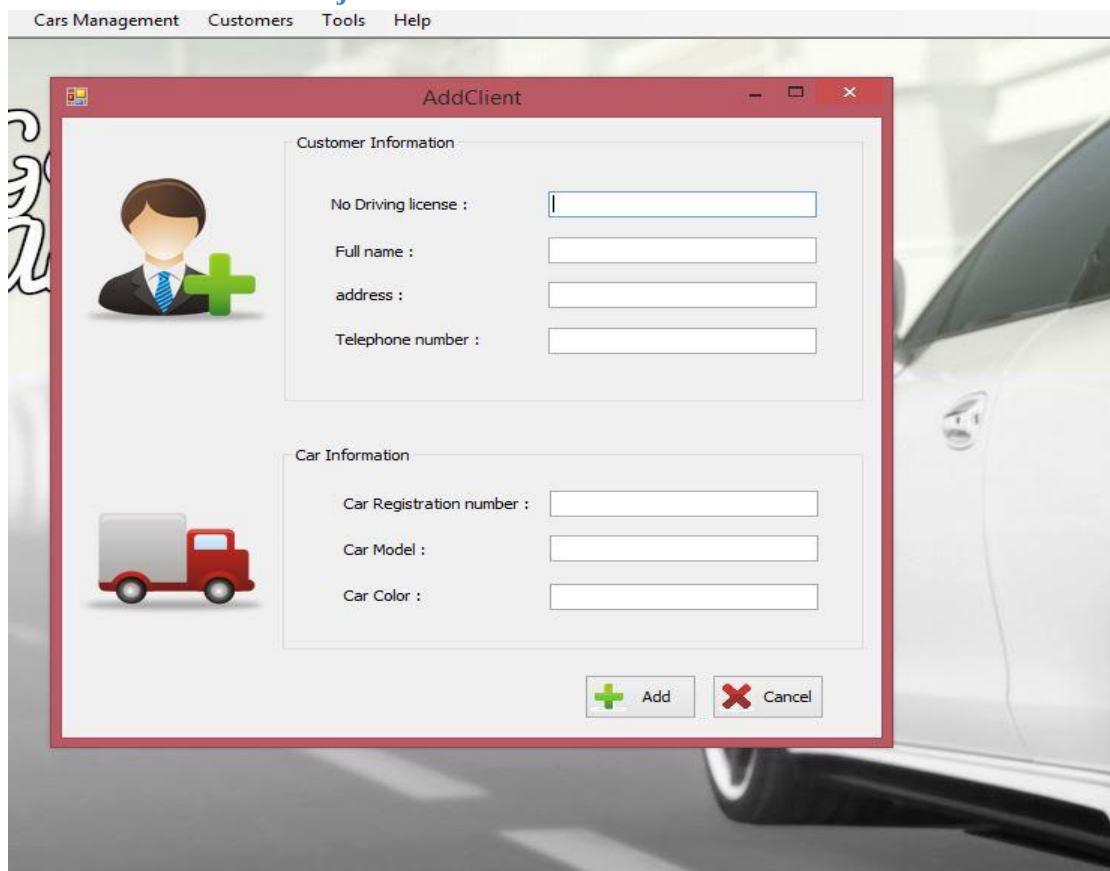


Figure 12: page de Ajoute Client .

III.3.2.5.Fenêtre de Stationnement :

ReceivingCar

Receiving Car information

No. Ticket Receipt :

Time Of Entry :

Type of Client : ☒ Client Not Reg ☐ Client Reg

Car Information

Car Registration number :

Car Model :

Car Color :

Car Driver Information

No Driving license :

Full name :

Telephone number :

ID Client

ID Client :

Print Ticket New Receipt Cancel

مؤسسة السلامة | الدبيلة | There are 1 of the 50 cars | Pricing : 100 DA | Date and time : 4/26/2016 10:35:35 AM | PricingDiscount : 50 %

Figure 13: page de Reserve automobile

DeliveryCar

Find Ticket

Search Type : ☒ No. Ticket Receipt ☐ Car Registration number ☐ No. Driving license

By No. Ticket Receipt :

☐ Discounting Find

Ticket Information

No. Ticket Receipt : No Driving license :

Car Reg number : Full name :

Time Of Entry : Telephone number :

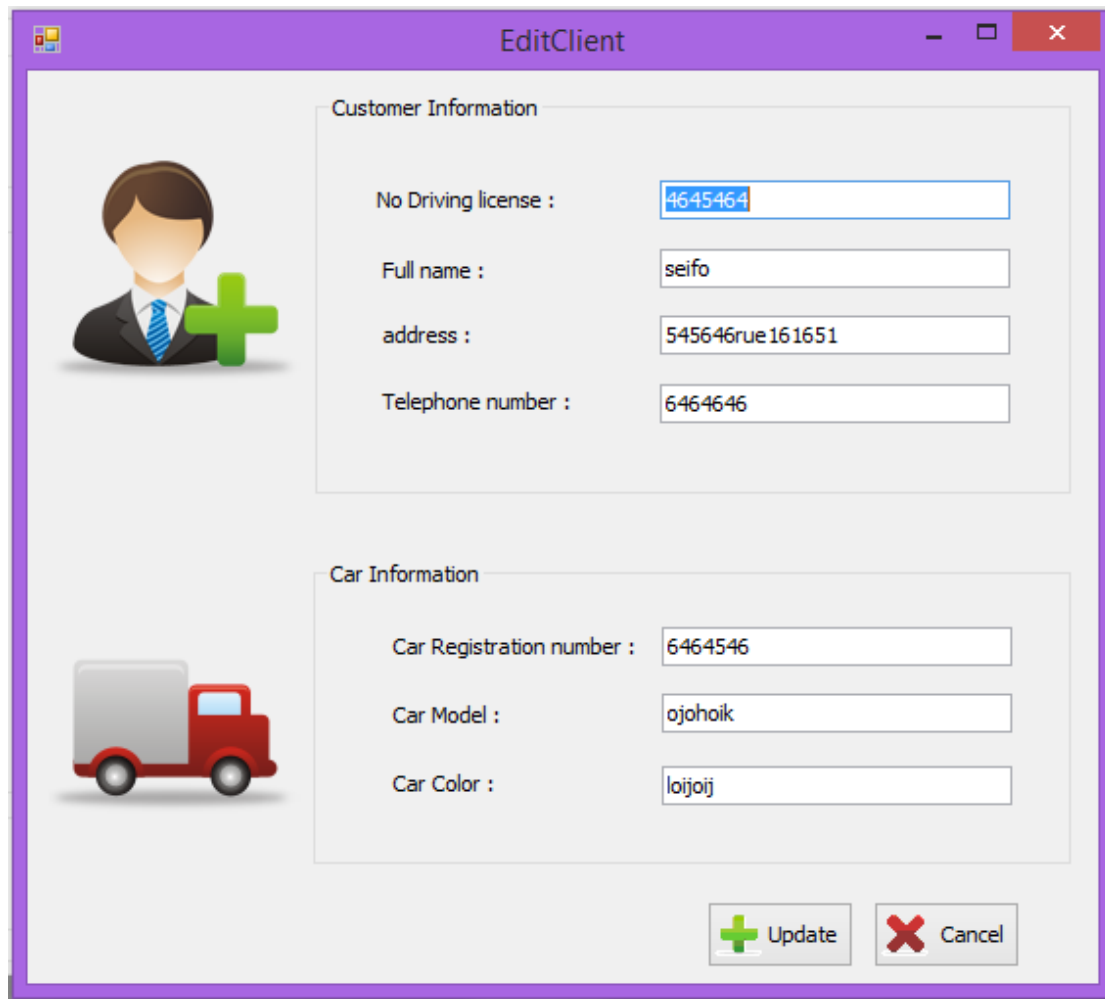
Time Out : Time spent :

Pricing DA / H : 0.0

Print Show Data Cancel

Figure 14: page de Sortir automobile

II.3.2.6 Formulaire de Modification :



The screenshot shows a software window titled "EditClient" with a purple title bar. Inside, there are two main sections: "Customer Information" and "Car Information".

Customer Information: This section is accompanied by an icon of a person in a suit with a green plus sign. It contains four text input fields:

- No Driving license : 4645464
- Full name : seifo
- address : 545646rue161651
- Telephone number : 6464646

Car Information: This section is accompanied by an icon of a red truck. It contains three text input fields:

- Car Registration number : 6464546
- Car Model : ojohoik
- Car Color : loijoi

At the bottom right of the window, there are two buttons: a green plus icon labeled "Update" and a red X icon labeled "Cancel".

Figure 15: page de Modifier Client .

II.3.3.Conclusion :

Dans ce dernier chapitre, nous avons présenté le coté réalisation de notre projet, justifié le choix du langage et les outils de développement. nous avons finit par présenter les interfaces les plus importantes.

Conclusion Générale

En conclues, nous avons arrivé développe un System de gestion de parking qui satisfait à tous les exigences principales d'un Parkingcomme l' ajoute des client , l' ajoute les place

Notre Systemcontinent les procédures suivantes :

- 1- Ajouter client
- 2- Supreme client
- 3- Calcule le montant de stationnement
- 4- Gérer les places
- 5- Statistiques De Parking

Perspective :

Nous perspective ajouter les parties suivantes du Système :

- 1- Le répartition des places par étages et par section
- 2- Gestion par réseau de plusieurs entrées et plusieurs sorties

Références bibliographique

[1] Microsoft , Pourquoi utiliser le langage C#.

<https://msdn.microsoft.com/fr-fr/library/aa287554%28v=vs.71%29.aspx>

[2] Data base Speed Comparison, SGBD :SQLite.

<http://www.sqlite.org/speed.html>