



CARS TRADING MANAGEMENT SYSTEM



Projet soumis dans le contexte du cours I3301 Génie Logiciel

Préparé par :

- Ibrahim Aref Mostapha :15338
- Ghadir Mohamad Harb:15409
- Abir Ahmad Zanati :14721

ANNEE ACADEMIQUE 2023-2024

UNIVERSITE LIBANAISE

Faculté des Sciences

Remerciements

Je tiens à exprimer mes sincères remerciements au Dr Rani Makke pour avoir pris le temps d'évaluer notre projet SRS. Sa profonde expertise dans le domaine a été une source d'inspiration et ses commentaires précieux ont grandement enrichi notre compréhension des exigences logicielles. Je suis également reconnaissant envers mes collègues de classe qui ont collaboré de manière constructive tout au long de ce processus. Leur engagement a contribué à la qualité finale du document. L'évaluation de ce SRS par Vous Dr Rani est une opportunité précieuse d'apprentissage, et ces remerciements reflètent ma gratitude envers toutes les personnes qui ont contribué à ce processus.

Table de Matières

Remerciements

Table de Matières

Chapitre I- Introduction

I.1 Le domaine métier

I.2 L'application modélisée

2.1 Profils d'Utilisateurs

2.2 Analyse des Applications Similaires Existantes

Chapitre II- Analyse des Besoins et Spécification

II.1 Introduction

II.2 Prototypage

II.3 Analyse des Besoins

3.1 Besoins Fonctionnels

II.4 Spécification

4.1 Cas d'Utilisation

4.2 Description Textuelle

Chapitre III- Conception de l'Application

III.1 Introduction

III.2 Diagrammes de Classe

III.3 Détails Techniques

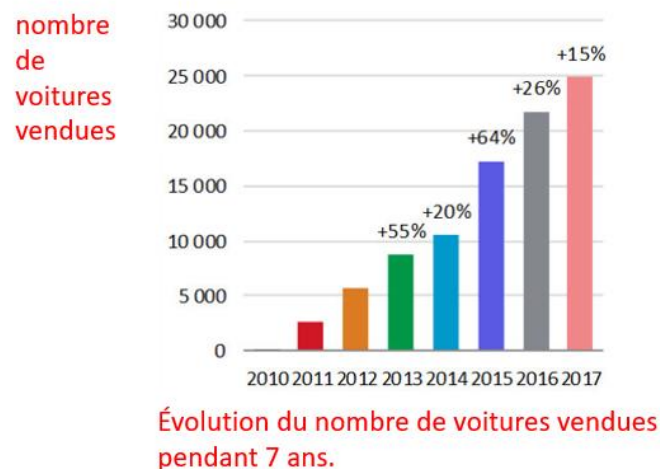
Chapitre IV- Conclusion

Chapitre I- Introduction

Le Cars Trading Management System est une application spécifiquement conçue pour répondre aux besoins du commerce automobile. Elle est destinée aux propriétaires d'entreprises comme "Cars Trading" qui se spécialisent dans l'achat et la vente de voitures. Le système vise à rationaliser et à optimiser les opérations quotidiennes en fournissant des outils de gestion complets

I.1 Le domaine métier

Le secteur de la vente de voitures offre un potentiel significatif en raison de la demande continue de véhicules neufs et d'occasion. Selon des rapports sectoriels récents, le marché automobile a connu une croissance stable au cours des dernières années. La demande croissante de mobilité personnelle, les avancées technologiques dans l'industrie automobile, et l'intérêt pour des options de véhicules plus respectueuses de l'environnement contribuent à maintenir une dynamique positive sur le marché. Cependant, il est important de noter que le succès d'une entreprise de vente de voitures dépend également de la capacité à rester concurrentielle, à s'adapter aux évolutions du marché, et à offrir une expérience client exceptionnelle.



I.2 L'application modélisée

L'application est conçue pour répondre aux besoins spécifiques de ce métier en offrant une automatisation efficace, une gestion transparente des transactions, et une vision claire de la santé

financière de l'entreprise. Elle vise à simplifier les tâches opérationnelles, à minimiser les erreurs humaines et à offrir une plateforme centralisée pour une gestion optimale des activités liées à la vente de voitures. Ainsi, elle devient un outil essentiel pour améliorer l'efficacité et la rentabilité de l'entreprise dans le domaine du commerce automobile.

2.1 Profils d'Utilisateurs

L'application que nous proposons de modéliser, le "Cars Trading Management System", est destinée à être utilisée par le propriétaire de l'entreprise (Cars Trading) pour rationaliser et automatiser divers aspects de son activité. Cette application devrait être un outil complet et efficace pour faciliter la gestion quotidienne de l'entreprise.

2.2 Analyse des applications similaires existantes

	Controle d'accès	Descriptions détaillées des voitures	Gestion du stock	Backup
Cars Trading Management System	X	x	x	x
App 1	X		x	
App 2	x	x		
App 3			x	

Chapitre II –Analyse des Besoins et Spécification

II.1 Introduction

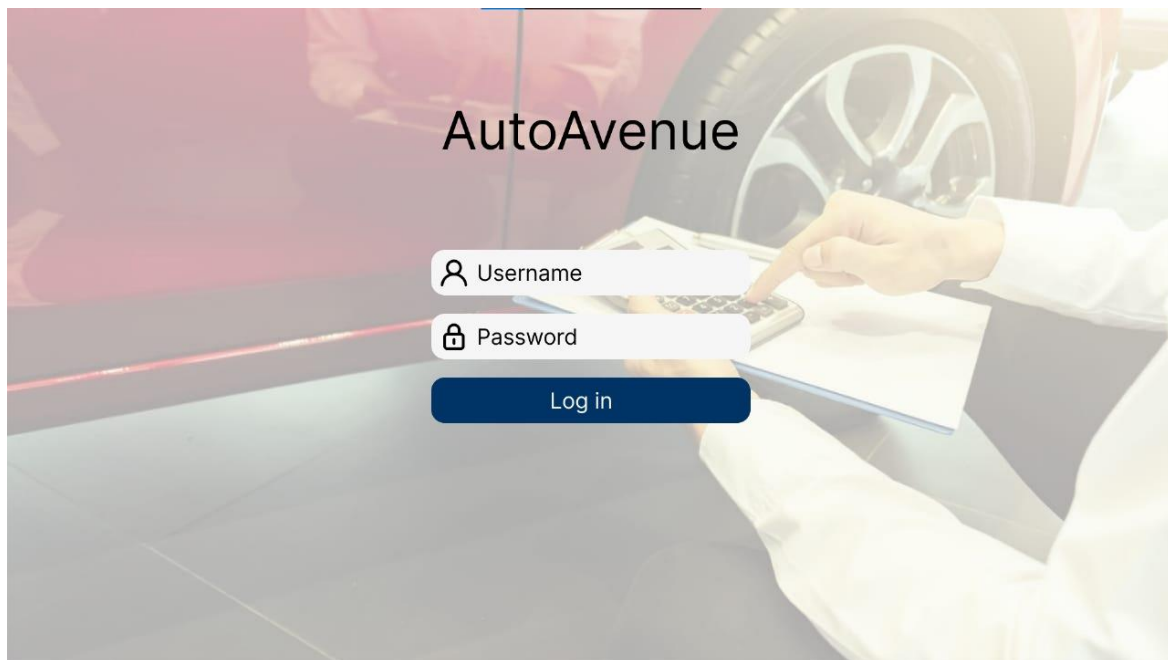
Pour collecter les besoins des clients dans le cadre de la création d'un système de gestion pour une entreprise de vente de voitures, nous avons utilisé un questionnaire bien structuré. Voici un exemple de quelques questions que nous avons posées dans l'enquête:

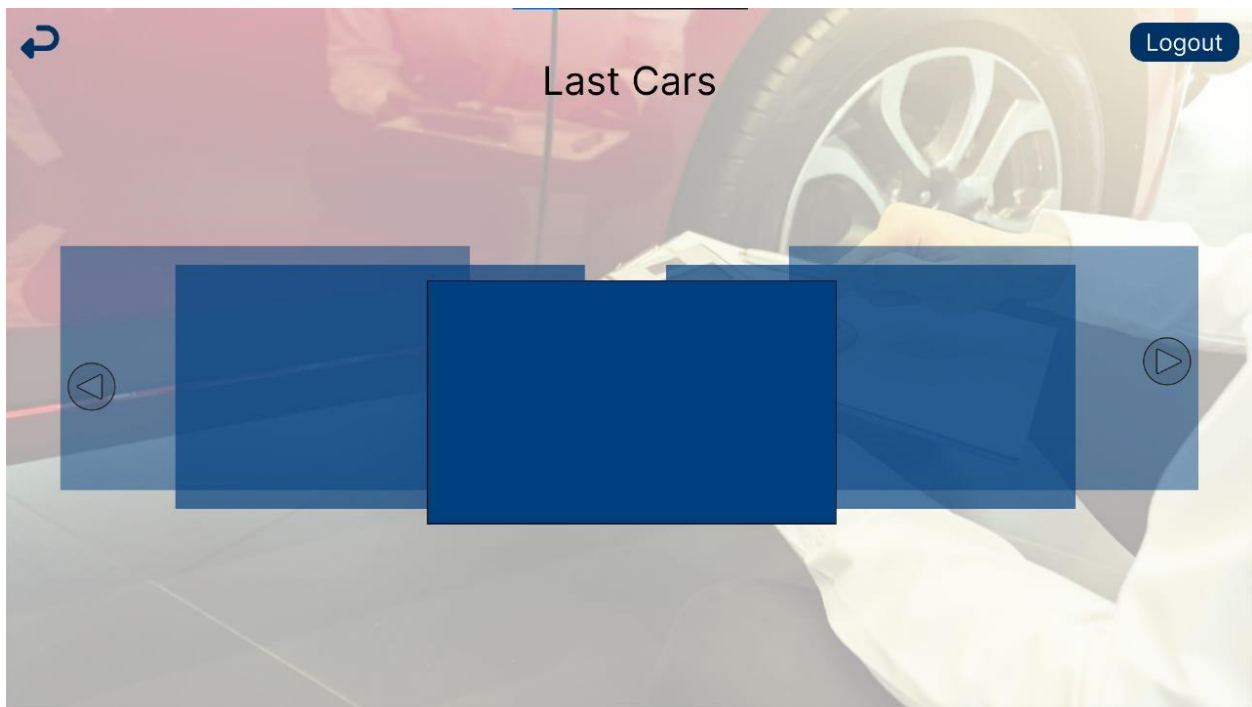
1. Gestion des Transactions :

- Quels sont les aspects les plus importants dans la gestion des transactions pour votre entreprise de vente de voitures ?

- Comment préférez-vous enregistrer et suivre les détails des transactions d'achat et de vente ?
- 2. Analyse Financière :**
- Quels indicateurs financiers sont essentiels pour évaluer la performance de votre entreprise ?
 - À quelle fréquence avez-vous besoin d'analyses financières, et quels types de rapports financiers trouvez-vous les plus utiles ?
- 3. Gestion du Stock :**
- Comment aimeriez-vous visualiser et suivre l'inventaire de voitures disponibles ?
 - Quelles informations spécifiques sur chaque voiture sont cruciales pour votre gestion de stock ?
- 4. Descriptions des Voitures :**
- Quels détails souhaitez-vous inclure dans les descriptions des voitures pour faciliter la vente ?
 - Comment préférez-vous organiser ces informations pour une consultation facile ?
- 5. Calcul des Bénéfices :**
- Quelles périodes sont les plus significatives pour le calcul des bénéfices de votre entreprise ?
 - Y a-t-il des éléments spécifiques que vous aimeriez inclure dans le calcul des bénéfices ?
- 6. Sauvegardes Automatiques :**
- Comment préférez-vous gérer les sauvegardes des données de votre entreprise ?
 - Quelle fréquence de sauvegarde vous semble la plus appropriée pour assurer la sécurité des données ?

II.2 Prototypage





[↶](#)

Inventory

Logout

Search...

Q

Filters:

Purchase Date:

Purchase Price:

Brand:

Color:

GO

from

to

from

to

bmw V

black V


ID	Car Name	Brand	Condition	Purchase Date	Purchase Price	View	Edit
1	Car 1					

Export

[↶](#)

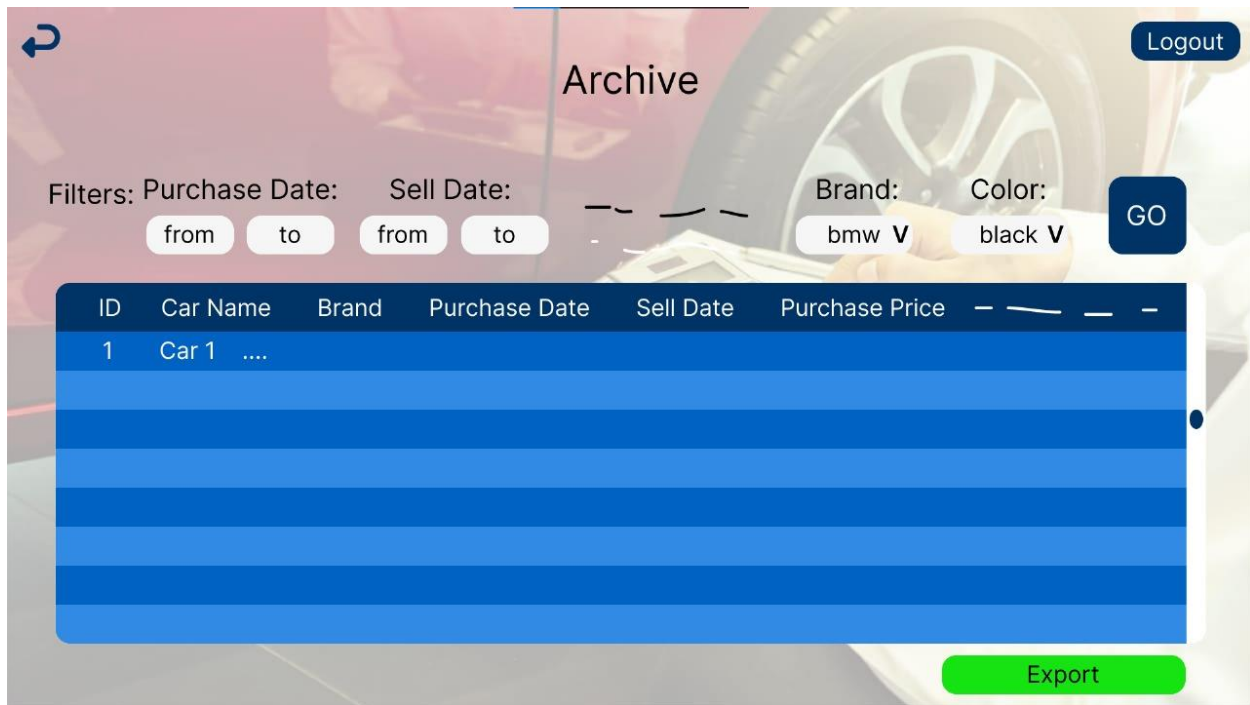
Car Details

Logout



Mercedes C300 2022

Description:



II.4 Analyse des Besoins

4.1 Besoins fonctionnels

Pour le logiciel proposé, le "Cars Trading Management System", voici une liste d'ensemble de fonctionnalités avec une ébauche de leurs scénarios :

1. Enregistrement des Transactions :

- Scenario** : Le propriétaire enregistre une nouvelle transaction, saisissant les détails de l'achat ou de la vente d'une voiture.

2. Calcul des Bénéfices :

- Scénario** : Le propriétaire spécifie une période, le système récupère les transactions pendant cette période, puis calcule et affiche le bénéfice total.

3. Gestion du Stock :

- Scénario** : Le propriétaire ajoute une nouvelle voiture au stock avec des informations détaillées telles que le nom, l'état, le prix d'achat, etc.

4. Visualisation des Voitures Disponibles :

- Scénario** : Le propriétaire visualise les voitures disponibles avec des détails complets comme le nom, l'état, la date d'achat, etc.

5. Génération de Rapports Analytiques :

- Scénario** : Le propriétaire génère un rapport analytique pour évaluer les performances financières, incluant des graphiques et des statistiques.

6. Mise à Jour des Descriptions des Voitures :

- Scénario** : Le propriétaire met à jour les détails d'une voiture existante, modifiant des informations comme le prix de vente, la couleur, etc.

7. Sauvegardes Automatiques :

- Scénario** : Le système effectue automatiquement une sauvegarde complète toutes les trois jours, assurant la sécurité des données.

8. Authentification et Gestion des Utilisateurs :

- Scénario** : Le propriétaire s'authentifie dans le système avec des identifiants uniques, accédant ainsi à leurs fonctionnalités spécifiques.

9. Notifications Automatiques :

- Scénario** : Le système envoie automatiquement une notification au gestionnaire de stock lorsque le niveau d'inventaire d'une voiture particulière est bas.

10. Histories des Transactions :

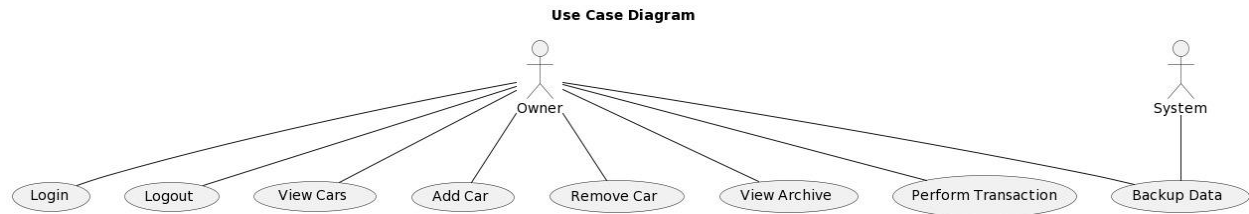
- Scénario** : Les utilisateurs peuvent consulter l'historique complet des transactions, y compris les détails de chaque achat et vente.

11. Recherche et Filtres Avancés :

- Scénario** : Les utilisateurs peuvent effectuer des recherches avancées et appliquer des filtres pour trouver rapidement des informations spécifiques dans le système.

II.5 Spécification

5.1 Diagramme de Cas d'Utilisation



5.2. Description Textuelle

Ce qui suit est un exemple de description textuelle détaillée de deux cas d'utilisation : calcul des Bénéfices, visualiser voitures disponibles.

- **Cas d'utilisation "calcul des Bénéfices"**

Number	2	
Name	Calcule des Benefices	
Summary	Ce cas d'utilisation permet au propriétaire de l'entreprise de gérer et de visualiser les bénéfices de l'entreprise pour une période spécifique	
Priority	1	
Postconditions	Le propriétaire peut visualiser le montant des bénéfices pour la période spécifiée.	
Trigger	Le propriétaire clique sur le bouton "calculer les Bénéfices"	
Main scenario	Step	Action
	1	Le propriétaire accède à la fonctionnalité de gestion des bénéfices depuis le tableau de bord principal.
	2	Le système affiche une interface permettant de spécifier la période pour laquelle les bénéfices doivent être calculés.
	3	Le propriétaire entre la date de début et la date de fin de la période.
	4	Le système récupère les données des transactions enregistrées pendant cette période.
	5	Le système calcule le montant total des bénéfices en soustrayant le coût total des achats du total des

		ventes.
	6	Le montant calculé des bénéfices est affiché à l'utilisateur.
Extensions	Branching scenario	
	Si aucune transaction n'est enregistrée pour la période spécifiée, le système affiche un message indiquant qu'aucun bénéfice n'a été généré.	
	indiquant qu'aucune voiture n'est actuellement en stock	

- **Cas d'utilisation "visualiser voitures disponibles"**

Number	4	
Name	Visualisation voitures disponibles	
Summary	Ce cas d'utilisation permet au gestionnaire de visualiser la liste des voitures disponibles dans le stock de l'entreprise "Cars Trading" avec des détails spécifiques.	
Priority	2	
Preconditions	Les informations sur le stock de voitures sont à jour.	
Postconditions	L'acteur peut visualiser la liste des voitures disponibles avec des détails complets.	
Trigger	Le propriétaire clique le bouton Visualiser Voitures Disponibles	
Main scenario	Step	Action
	1	L'acteur sélectionne l'option "Visualiser Voitures Disponibles" dans l'interface utilisateur du système.
	2	Le système affiche une liste des voitures disponibles, triées par défaut par ordre d'ajout au stock.
	3	Pour chaque voiture répertoriée, le système affiche les informations suivantes: Nom de la voiture,État (neuf, d'occasion, etc.),Date d'achat,Prix d'achat,Prix de vente,Couleur,Type de moteur,Autres détails pertinents
	4	L'acteur a la possibilité de filtrer et de trier la liste en fonction de différents critères, tels que le prix, la couleur, le type, etc.
	5	L'acteur peut sélectionner une voiture spécifique pour obtenir des détails plus approfondis.
Extensions	Branching scenario	
	Si aucun véhicule n'est disponible, le système affiche un message indiquant qu'aucune voiture n'est actuellement en stock	

Chapitr III- Conception de l'Application

III.1 Introduction

Dans le chapitre précédent, nous avons introduit une liste de fonctionnalités que nous comptons offrir dans notre application proposée... Ceci entraine l'implication des entités suivantes... Dans ce chapitre, nous utilisons des diagrammes UML de classes afin de clarifier notre modèle conceptuel et ses entités, leurs données considérées manipulées par l'application, leurs rôles dans l'application et leurs associations entre elles.

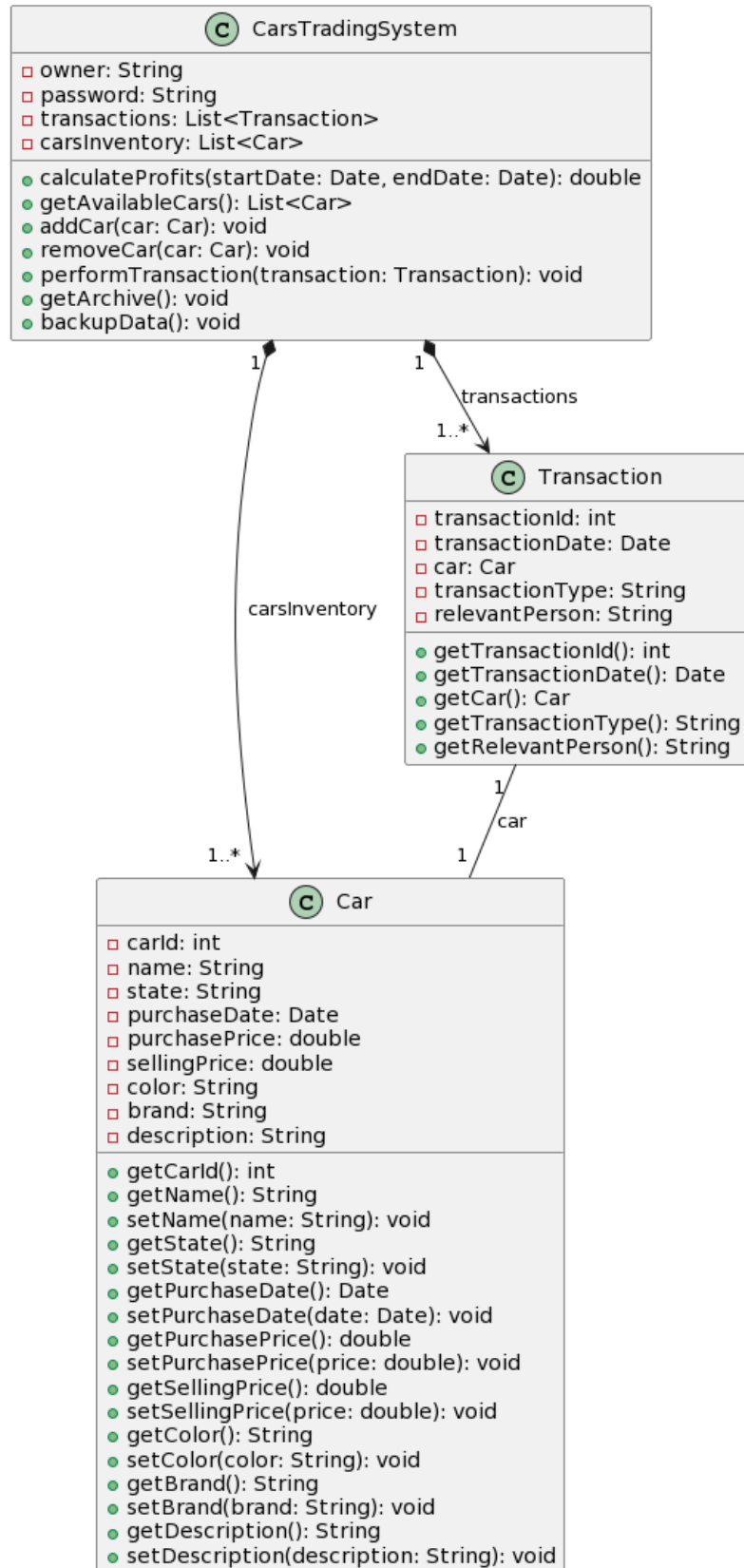
III.2 Diagramme de Classe

CarsTradingSystem représente la classe principale pour la gestion du système de trading. Elle contient des attributs tels que propriétaire (owner), transactions (transactions) et inventaire de voitures (carsInventory). Les méthodes incluent calculateProfits, getAvailableCars, addCar, removeCar, performTransaction et backupData.

Transaction représente une entité transaction avec des attributs tels que transactionId, transactionDate, car et transactionType. TransactionType définit le type de transaction (ACHAT ou VENTE).

Car représente les détails d'une voiture avec des attributs tels que carId, name, state, purchaseDate, purchasePrice, sellingPrice, color et engine.

Class Diagram



1. CarsTradingSystem (Système de Trading de Voitures) - Car (Voiture) :

- **Relation** : Composition

- **Explication** : Le système de trading de voitures contient une liste de voitures (carsInventory). Cela signifie que la durée de vie des objets de la classe Car est liée à celle du système de trading. Si le système est supprimé, toutes les voitures associées seront également supprimées.

2. CarsTradingSystem (Système de Trading de Voitures) - Transaction :

- **Relation** : Composition

- **Explication** : De manière similaire, le système de trading de voitures contient une liste de transactions (transactions). Cela implique que les transactions sont spécifiques à ce système, et si le système est supprimé, toutes les transactions associées seront également supprimées.

3. Transaction - Car (Transaction - Voiture) :

- **Relation** : Association

- **Explication** : Chaque transaction est associée à une voiture spécifique (car). Cela signifie qu'une transaction implique l'achat ou la vente d'une voiture particulière. Les objets Transaction et Car existent indépendamment l'un de l'autre, mais une transaction fait référence à une voiture particulière.

III.3 Details Techniques

Langage de Programmation :

Pour le développement de l'application logicielle côté serveur, Java est l'option privilégiée en raison de sa polyvalence et de sa portabilité. Java permettra la mise en œuvre d'une application qui peut être utilisée hors ligne sur un PC, avec la possibilité de se connecter en ligne tous les trois jours pour effectuer des sauvegardes de données.

Interface Utilisateur (GUI) :

Java offre des bibliothèques graphiques comme Swing ou JavaFX pour le développement de l'interface utilisateur hors ligne. Ces bibliothèques permettent de créer une interface utilisateur intuitive et conviviale pour le système de trading de voitures.

Base de Données :

MySQL est sélectionné comme système de gestion de base de données pour stocker et gérer les données locales du système de trading de voitures. Les données seront manipulées en mode hors ligne, avec la possibilité de synchroniser les données avec le cloud (par exemple, AWS) lors des sauvegardes régulières effectuées tous les trois jours.

Architecture Logicielle :

On a adopté une architecture MVC (Modèle-Vue-Contrôleur) pour la séparation des préoccupations et une gestion plus efficace des composants.

Sécurité :

Des mécanismes d'authentification sécurisés sont mis en œuvre pour garantir que seul le propriétaire du système peut accéder aux fonctionnalités, même en mode hors ligne. Les communications en ligne, lors des sauvegardes, seront sécurisées à l'aide de protocoles standard tels que HTTPS.

Sauvegardes et Récupération :

Configurez des routines de sauvegarde régulières pour assurer la sécurité des données. Pour cela, une connexion avec AWS Cloud est mise en œuvre.

Chapitre IV – Conclusion

En conclusion, le Cars Trading System représente une solution technologiquement avancée, répondant aux besoins du secteur du commerce automobile. Son adoption promet une gestion plus efficace, des transactions transparentes, et une prise de décision éclairée grâce à des analyses approfondies des données financières. La mise en œuvre réussie de ce projet peut conduire à une amélioration significative des opérations commerciales dans le domaine du trading automobile.