الجمهورية اليمنية

وزارة التعليم العالي والبحث العلمي

جامعة آزال للتنمية البشرية

كلية الهندسة وتقنية المعلومات

جامعة آزال للتنمية البشرية
Azal University for Human Development

Safer Application and Platform

تطبيق ومنصة سافر

إعداد الطلاب:

زايد حسن طه ناجي شايع	2022090123
زكريا يحيى علي احمد العجاج	2022110648
محمد إبراهيم طاهر عبد الجليل	2022110004
محمد ضيف الله مبخوت الفقيه	2022110738
محمد علي عبد الله زايد	2022110797

تحت إشراف: أ.م.د. مختار غيلان

تم إعداد هذا المشروع كجزء من متطلبات نيل شهادة البكالوريوس لقسم تقنية المعلومات جامعة آزال للتنمية البشرية

2025 -2026

آيات قرآنية

بسم الله الرحمن الرحيم

قال الله تعالى:

﴿وَقُل رَّبِّ زِدْنِي عِلْمًا﴾ (١)

وقال تعالى:

﴿وَقُلِ اعْمَلُوا فَسَيَرَى اللَّهُ عَمَلَكُمْ وَرَسُولُهُ وَالْمُؤْمِنُونَ وَسَتُرَدُّونَ إِلَىٰ عَالِمِ الْغَيْبِ وَالشَّهَادَةِ فَيُنَبِّئُكُم بِمَا كُنتُمْ تَعْمَلُونَ﴾ (٢)

وقال تعالى:

﴿قَالُواْ سُبْحَٰنَكَ لَا عِلْمَ لَنَآ إِلَّا مَا عَلَّمْتَنَآ إِنَّكَ أَنتَ ٱلْعَلِيمُ ٱلْحَكِيمُ﴾ (٣)

وقال تعالى:

﴿رَبَّنَا تَقَبَّلْ مِنَّاۖ إِنَّكَ أَنتَ السَّمِيعُ الْعَلِيمُ﴾ (٤)

---

١ .سورة طه (١١٤)

٢ .سورة التوبة (١٠٥)

٣ .سورة البقرة (٣٢)

٤ .سورة البقرة (١٢٧)

ملخص المشروع(*Summary*)

في ظل التطور التكنولوجي المتسارع والحاجة الملحة لتنظيم قطاع النقل في اليمن، يعد مشروع **"سافر "**منصة رقمية رائدة تهدف إلى إحداث تحول جذري في قطاع النقل البري بين المحافظات اليمنية عبر رقمنه العمليات التقليدية.

تعمل المنصة كحلقة وصل ذكية تجمع بين سائقي الرحلات والمسافرين، مستبدلةً آليات الحجز اليدوية بنظام تقني متكامل يضمن السهولة، الشفافية، والموثوقية في تقديم الخدمة.

تعتمد المنصة على بنية تقنية متطورة لتعزيز مستويات الأمان والكفاءة؛ حيث توفر ميزات التتبع اللحظي للمسارات، وتدعم خيارات الدفع الإلكتروني المواكبة للتحول المالي الرقمي، بالإضافة إلى نظام تقييم متبادل يضمن استمرارية جودة الخدمة والمنافسة العادلة بين مزودي الخدمة.

يهدف المشروع بشكل أساسي إلى إعادة هيكلة منظومة النقل من خلال رفع كفاءة استخدام المركبات وتقليل زمن الانتظار الضائع للمسافرين. ومن المتوقع أن يساهم "سافر" في خلق بيئة نقل آمنة ومنظمة، مما ينعكس إيجاباً على تجربة المستخدم النهائي ويساهم في دفع عجلة التحول الرقمي في البنية التحتية للخدمات في اليمن.

التفويض

نحـــــن نُخـــــوّل جامعـــــة آزال للتنميـــــة البشـــــرية – كليـــــة الهندســـــة وتقنيـــــة المعلومـــــات بأن تقوم بتزويد المكتبات أو الجهات أو الأفراد بنسخ من تقرير مشروع تخرجنا عند الطلب.

اسم الطالب	رقم الطالب	التوقيع
زايد حسن طه ناجي شايع	2022090123	
زكريا يحيى علي احمد العجاج	2022110648	
محمد إبراهيم طاهر عبد الجليل	2022110004	
محمد ضيف الله مبخوت الفقيه	2022110738	
محمد علي عبد الله زايد	2022110797	

التاريخ:...../.../ ...

إهداء

إلى من هو أحبّ إلينا من أنفسنا، سيّد الخلق وإمام المرسلين، محمد صلى الله عليه وسلم، الذي علّم الدنيا معنى الهداية والنور.

ثم إلى أمهاتنا الغاليات، من تعلّمنا في مدارس قلوبهنّ أبجدية الوفاء، وارتشفنا من تضحياتهنّ مناهج الصبر والعطاء.

معلماتنا الأولى وملهماتنا الدائمة، إلى من كنّ الصديقات الوفيات اللواتي لم يخذلننا يوماً، إلى حظنا الجميل وفوزنا الحقيقي في هذه الحياة، إلى من كنّ الداعم الأول في رحلة علمنا وتحقيق أحلامنا، وكان عوناً لنا وسنداً في كل خطوة، بكُنّ تعلمنا كيف نواجه الصعاب وبكُنّ استمددنا ثقتنا بأنفسنا . إلى القلوب الطيبة الرحيمة، إلى من كانت صلواتهنّ حارساً لنا في كل مكان

ونختم بآبائنا الكرام، الذين أرادوا لنا مستقبلاً أفضل من واقعهم فقالوا: "لا تكونوا مثلنا لنفخر بكم"، فاخترنا أن نقتدي بهم لنفخر بأننا أبناؤهم، ولنريهم أن منبتهم الطيب أنتج ثماراً تسرّ الناظرين.

شكر وعرفان

نتوجه أولاً بالشكر لله عز وجل الذي أعاننا على إتمام هذا المشروع، ثم نتوجه بجزيل

الشكر والتقدير لكل

من ساندونا في إنجاز هذا المشروع، أو تابعونا في كل مراحل بنائه وتنفيذه

وأخص بالشكر:

الدكتور/ مختار غيلان

المهندس/محمد شايع

الذي ما بخل في تقديم النصح والإرشاد في سبيل إنجاح المشروع ولا ننسى

فضل كل من

شارك في اتمام مسيرتنا التعلمية.

إقرار المشرف

أُقِرّ بأن إعداد هذا المشروع بعنوان

تطبيق ومنصة سافر(Safer Application and Platform)

قد أُعِدّ بواسطة:

اسم الطالب	رقم الطالب
زايد حسن طاهر ناجي شايع	2022090123
زكريا يحيى علي أحمد العجاج	2022110648
محمد إبراهيم طاهر عبد الجليل	2022110004
محمد ضيف الله مبخوت الفقيه	2022110738
محمد علي عبد الله زايد	2022110797

تم إعداد هذا المشروع تحت إشرافي في قسم تقنية المعلومات

وذلك استكمالاً لمتطلبات الحصول على درجة البكالوريوس في جامعة آزال للتنمية البشرية – كلية الهندسة.

اسم المشرف:

التوقيع:

التاريخ:

لجنة الحكم والمناقشة

عنوان المشروع: .....

المشرف

الاسم	المنصب	التوقيع

لجنة المناقشة

الاسم	المنصب	التوقيع

رئيس القسم

.....

جدول المحتويات

آيات قرآنية I

ملخص المشروع(Summary) II

التفويض III

إهداء IV

شكر وعرفان V

إقرار المشرف VI

لجنة الحكم والمناقشة VII

جدول المحتويات VIII

فهرس الجداول XIII

فهرس الاشكال XV

قائمة المختصرات XVII

1 الفصل الأول (Chapter1) 1

1.1 المقدمة (Introduction) 2

1.2 مشكلة البحث (Problem Statement) 2

1.3 الأهداف (Objectives) 3

1.4 حدود المشروع (Project Scope) 4

1.4.1 الحدود الوظيفية (Functional Scope) 4

1.4.2 الحدود التشغيلية:(Operational Scope) 5

1.4.3 الحدود المكانية(Spatial Scope / Geographic Boundaries) : 5

1.4.4 ما يخرج عن نطاق المشروع(Out of Project Scope) 5

1.5 المنهجية المستخدمة لإنجاز المشروع: 5

1.6 معايير القبول(Acceptance Criteria) 6

1.7 تعريف النظام (System Definition) 7

1.8 الغرض العام من التطبيق (Purpose) 7

1.9 أهمية المشروع (Significance) 7

1.10 القيود والمعوقات(Constraints & Limitations) 8
1.11 الفرضيات (Assumptions) 8
1.12 دورة حياة النظام(Life Cycle Model) 8
1.13 الخطة الزمنية (Time Plan) 9
2 الفصل الثاني(Chapter 2) 10
2.1 مكانة التطبيقات في تسهيل الخدمات: 11
2.2 الخلفية (Background) 11
2.3 الدراسات السابقة (Literature Review) 12
2.3.1 تطبيقUber 12
2.3.2 تطبيقCareem 12
2.3.3 تطبيقInDriver 12
2.3.4 تطبيقBlaBlaCar 13
2.3.5 تطبيقات النقل المحلية(Local Transportation Apps) 13
2.4 الفجوة البحثية(Research Gap) 15
2.5 النظام المقترح "سافر" (Suggested System - SAFER) 15
2.6 تحليل المقارنة(Comparative Analysis) 16
2.7 مستخدمي النظام وخصائصهم (Users) 17
2.8 دراسة الجدوى (Feasibility Study) 17
2.8.1 الجدوى التقنية(Technical Feasibility) 17
2.8.2 الجدوى الاقتصادية(Financial Feasibility) 19
2.8.3 الجدوى التشغيلية(Operational Feasibility) 21
2.8.4 الجدوى الزمنية(Scheduling Feasibility) 22
2.8.5 الجدوى القانونية(Legal Feasibility) 25
3 الفصل الثالث(Chapter 3) 26
3.1 المقدمة(Introduction) 27
3.2 متطلبات النظام(System Requirements) 27

3.3 نطاق التحليل(Scope of Analysis) 27
3.4 المتطلبات الوظيفية(Functional Requirements) 27
3.5 المتطلبات غير الوظيفية(Non-Functional Requirements) 28
3.6 لغة النمذجة الموحدة(UML) 28
3.6.1 حالات الاستخدام(Use Cases Descriptions) 29
3.6.2 مخطط التسلسل (Sequence Diagram) 32
3.6.3 مخطط الفئات (Class Diagram) 34
3.6.4 علاقات مخطط الفئات 35
3.6.5 مخطط التسلسل (سيناريو حجز رحلة من قبل الراكب) 36
3.6.6 مخطط التسلسل سيناريو إنشاء رحلة جديدة من قبل السائق 39
3.6.7 مخطط التسلسل سيناريو رحلة الراكب من البحث عن الرحلة إلى التقييم النهائي 41
3.6.8 مخطط النشاط لسيناريو: حجز رحلة والدفع وتأكيد الالتقاط 44
3.6.9 مخطط الكيانات والعلاقات(Entity Relationship Diagram - ERD) 47
3.6.10 نموذج الكيانات والعلاقات(Entity Relationship Model - ERM) 50
4 الفصل الربع(Chapter 4) 51
4.1 المقدمة(Introduction) 52
4.2 كيفية بناء النظام(System Overview) 52
4.3 ربط قاعدة البيانات بالنظام 52
4.4 بناء قواعد البيانات(Database Building) 53
4.5 قاموس البيانات(Data Dictionary) 54
5 الفصل الخامس(Chapter 5) 70
5.1 مقدمة(Introduction) 71
5.2 بيئة التنفيذ(Implementation Environment) 71
5.3 خطوات التنفيذ(Implementation Steps) 71
5.4 مرحلة الاختبار(Testing Phase) 71
5.5 قواعد العمل(Business Rules) 71

5.6 متطلبات الأمان(Security Requirements) 72
5.7 خطة الاختبار المبدئية(Initial Test Plan) 72
5.8 تحليل الاختبارات الأمنية والجدوى..... 72
5.9 حمية ضد ثغراتSQL Injection 73
5.10 اختبار حماية معدل الطلبات(Rate Limiting Test) 74
6 الفصل السادس(Chapter 6) 76
6.1 مخرجات النظام..... 77
6.1..... 77
6.1.1 واجهات التطبيق لأنشاء حساب سائق 77
6.1.2 واجهات التطبيق لأنشاء رحلة 78
6.1.3 واجهات التطبيق لحجز رحلة 79
6.1.4 واجهات التطبيق لتغذية المحفظة 80
6.1.5 واجهات التطبيق لإضافة مركبة..... 80
6.1.6 واجهات التطبيق تفاصيل إضافة مركبة..... 81
6.1.7 واجهة لوحة التحكم: 82
6.1.8 واجهة التحكم بالسائقين..... 83
6.1.9 وجهات التحكم بالرحلات 84
6.1.10 واجهة التحكم بطرق الدفع..... 85
6.1.11 واجهة التحكم بالمحافظ الالكترونية 86
6.1.12 واجهة التحكم بكشف الحساب..... 86
6.1.13 واجهة التحكم بالدعم والمساعدة 87
6.1.14 واجهة التحكم بالرسائل..... 87
6.1.15 واجهة التحكم بالإعدادات 88
6.1.16 واجهات التحكم بسجلات النظام 90
6.2 مناقشة النتائج:..... 91
7 الفصل السابع(Chapter 7) 92

7.1 الاستنتاجات(Conclusions) 93
7.2 التوصيات(Recommendations) 93
7.3 الخاتمة(Conclusion) 94
المراجع 95

فهرس الجداول

جدول 1-1 نموذج الإيرادات المقترح ..... **Error! Bookmark not defined.**
جدول 2-1 التكاليف الرأسمالية الأولية 19
جدول 3-1 التكاليف التشغيلية الشهرية 20
جدول 4-1 : مرحلة التهيئة 23
جدول 5-1 :مرحلة التحليل 23
جدول 6-1 :مرحلة التصميم 24
جدول 7-1: مرحلة التنفيذ 24
جدول 8-1مرحلة اختبار النظام 25
جدول 1-2:تحليل المقارنة 16
جدول 2-2: مستخدمين النظام 17
جدول 1-3: المتطلبات الوظيفية 27
جدول 2-3: المتطلبات الغير وظيفية 28
جدول 1-4: جدول المستخدمين 54
جدول 2-4: جدول الرحلات 55
جدول 3-4: الحجوزات 56
جدول 4-4: جدول المحفظة 57
جدول 5-4:حركات المحفظة 58
جدول 6-4: أنواع حركات المحفظة 59
جدول 7-4: طرق الدفع والشحن 59
جدول 8-4 :معاملات الحجوزات 60
جدول 9-4: المركبات 62
جدول 10-4: المحافظات 62
جدول 11-4 :الفرزات 63
جدول 12-4: الرسائل 64
جدول 13-4:العناوين 65

جدول 14-4: الاشعارات 66
جدول 15-4:وثائق السائق 67
جدول 16-4:التقيمات 68
جدول 1-5: الاختبارات المبدئية 72

فهرس الاشكال

شكل 1-1: منهجية البحث 6

شكل 1-2: الخطة الزمنية 9

شكل 1-3: مرحلة التهيئة 23

شكل 1-4 : مرحلة التحليل 23

شكل 1-5 : مرحلة التصميم 24

شكل 1-6 : مرحلة التنفيذ 24

شكل 1-7: اختبار النظام 25

شكل 3-1:مخطط حالة الاستخدام 29

شكل 3-2: مخطط التسلسل لعملية إنشاء وتوثيق حساب السائق 32

شكل 3-3 : مخطط الفئات 34

شكل 3-4: علاقات مخطط الفئات 35

شكل 3-5: مخطط التسلسل لسيناريو حجز رحلة من قبل الراكب 36

شكل 3-6 :سيناريو إنشاء رحلة جديدة من قبل السائق 39

شكل 3-7: مخطط التسلسل سيناريو رحلة الراكب من البحث عن الرحلة إلى التقييم النهائي 41

شكل 3-8 :مخطط النشاط حجز رحلة 44

شكل 3-9: مخطط الكيان والعلاقة(ERD) 47

شكل 3-10: نموذج الكيان والعلاقة 50

شكل 4-1: المستخدمين 54

شكل 4-2: الرحلات 55

شكل 4-3: الحجوزات 57

شكل 4-4: المحفظة 58

شكل 4-5: حركات المحفظة 58

شكل 4-6 : أنواع حركات المحفظة 59

شكل 4-7: طرف الدفع والشحن 60

شكل 4-8:معاملات الحجوزات 60

شكل 4-9: المركبات 62
شكل 4-10: المحافظات 63
شكل 4-11: الفرزات 64
شكل 4-12: الرسائل 64
شكل 4-13: العناوين 65
شكل 4-14 الاشعارات 66
شكل 4-15:وثائق السائق 68
شكل 4-16:التقيمات 69
شكل 6-1 إنشاء حساب سائق 77
شكل 6-2 إنشاء رحلة 78
شكل 6-3 حجز رحلة 79
شكل 6-4: تغذية محفظة 80
شكل 6-5 إضافة مركبة 80
شكل 6-6:خطوات إضافة مركبة 81
شكل 6-7 واجهة لوحة التحكم الرئيسية 82
شكل 6-8 : واجهة عرض وإضافة السائقين 83
شكل 6-9: لوحة التحكم بالرحلات 84
شكل 6-10: طرق الدفع 85
شكل 6-11: المحافظ الالكترونية 86
شكل 6-12: كشف حساب 86
شكل 6-13: الدعم والمساعدة 87
شكل 6-14: واجهة المحادثة 87
شكل 6-15: التحكم بالاعدادات 88
شكل 6-16: سجلات النظام 90

قائمة المختصرات

الاختصار	التعريف
API	Application Programming Interface
CAPEX	Capital Expenditures
ERD	Entity Relationship Diagram
ERM	Entity Relationship Model
FR	Functional Requirements
IDEs	Integrated Development Environments
iOS	iPhone Operating System
MySQL	My Structured Query Language
MVP	Minimum Viable Product
NFR	Non-Functional Requirements
OPEX	Operating Expenditures
REST APIs	Representational State Transfer APIs
ROI	Return on Investment
SDLC	Software Development Life Cycle
UI/UX	User Interface / User Experience

1 الفصل الأول (Chapter 1)

المقدمة (Introduction)

1.1 المقدمة (Introduction)

يشهد العالم اليوم تطورًا متسارعًا في تقنيات النقل الذكي، حيث أصبحت الحلول الرقمية عنصرًا أساسيًا لتسهيل حركة التنقل بين المدن وضمان أمان وكفاءة الخدمات المقدَّمة للمستخدمين. يهدف مفهوم النقل الذكي إلى توظيف التكنولوجيا لتحسين جودة الخدمات، تقليل الأخطاء البشرية، ومتابعة حركة المركبات بشكل لحظي.

انطلاقًا من هذه الرؤية، يأتي **مشروع" تطبيق ومنصة سافر "**لتصميم تطبيق رقمي مبتكر يربط بين المسافرين والسائقين، ويوفّر حلولًا فعّالة لمشكلات الحجز وتنظيم الرحلات، مع تعزيز مستوى الأمان والموثوقية. يرتكز التطبيق على واجهات مستخدم سهلة الاستخدام، وقواعد بيانات منظمة، ونظام إشعارات لحظية لضمان تجربة سلسة لكل من المسافرين والسائقين.

لا يُعد التطبيق مجرد منصة للحجز الإلكتروني، بل هو نظام متكامل لإدارة النقل بين المدن، يقدّم بيانات دقيقة ومحدّثة في الوقت الحقيقي، ويتيح لجميع الأطراف المشاركة—سواء المسافرين أو السائقين أو إدارات النقل—الوصول إلى المعلومات الضرورية لاتخاذ قرارات سريعة وفعّالة.

1.2 مشكلة البحث (Problem Statement)

يعاني قطاع النقل بين المحافظات اليمنية من عدة مشاكل أساسية تؤثر على كفاءة الخدمة ورضا المستخدمين، وبناءً على الواقع الحالي للنقل بين المحافظات اليمنية، تبرز المشاكل التالية:

1. **صعوبة الوصول للخدمة (مشكلة الوقت والمكان):**

اضطرار المسافرين للانتظار لساعات طويلة في الشوارع العامة أو الذهاب حصراً إلى "الفرزة" (مكان تجمع السيارات)، وانحصار الرحلات في أوقات محددة (غالباً في الصباح الباكر)، مما يجعل السفر في أوقات متأخرة من اليوم شبه مستحيل.

2. **انعدام الموثوقية وغياب الأمان (مشكلة الاحتيال والراحة):**

تعرض الركاب للاحتيال، مثل قيام السائق بإنزالهم في منتصف الطريق ونقلهم إلى سيارة أخرى قد تكون أقل راحة، مما يسبب القلق والتأخير، وعدم وجود آلية لضمان وصول المسافر إلى وجهته بنفس المركبة التي بدأ بها الرحلة.

3. **صعوبة الخدمة على فئات معينة (مشكلة العوائل والخصوصية):**

المشقة الكبيرة التي تواجهها العوائل في الانتقال إلى "الفرزة" بحقائبهم، وصعوبة الحصول على خدمة "التوصيل من الباب إلى الباب" (القدوم إلى المنزل) لعدم وجود نظام مركزي يتيح طلبها، إلا بالاعتماد على المعارف الشخصية.

4. **تدني الكفاءة التشغيلية للسائقين (مشكلة طوابير الانتظار):**

إضاعة السائقين لوقتهم في طوابير طويلة داخل "الفرزات" قد تمتد ليوم أو يومين للحصول على حمولة كاملة، وهدر كبير في الموارد بسبب عدم وجود آلية تضمن للسائق الحصول على ركاب لرحلة العودة (في حال كان يعتمد على هذه الوسيلة من اجل المعيشة، ويعمل على نفس الخط).

5. **غياب الشفافية (مشكلة التفاوض والسعر):**

استنزاف وقت وجهد الراكب والسائق في "التفاوض" على أجرة الراكب في كل مرة، لعدم وجود أسعار معيارية واضحة ومتاحة للجميع.

6. **انعدام وجود نظام تقني موحد:**

تفتقر اليمن لوجود نظام تقني متخصص في النقل التشاركي بين المدن، مع غياب نظام مركزي لإدارة بيانات الرحلات والسائقين، ونقص حاد في المعلومات الفورية حول الرحلات.

تلك المشاكل تجعل من الضروري تطوير نظام رقمي شامل قادر على معالجة هذه التحديات وتحسين تجربة المستخدم بشكل كبير.

7. **مشكلة الفراغ الرقابي (المتابعة) :**غياب سجل تاريخي للرحلات؛ في النظام التقليدي إذا فُقدت حقيبة أو حدثت مشكلة، لا يوجد مرجع للبيانات. (نظام "سافر" يحل هذا عبر سجل النشاط والرحلات المنتهية).

1.3 الأهداف (Objectives)

لمعالجة هذه المشاكل تحديداً، يهدف تطبيق "سافر" إلى تحقيق ما يلي:

1. **تمكين الحجز الفوري والمجدول (حل مشكلة الوقت والمكان).**

توفير منصة رقمية تتيح للمسافر حجز مقعده من موقعه الحالي وفي أي وقت، دون الحاجة للذهاب إلى "الفرزة" أو الانتظار في الشارع.

2. **بناء نظام موثوق وشفاف (حل مشكلة السعر والاحتيال).**

عرض أسعار الرحلات بشكل واضح ومحدد مسبقاً في التطبيق، مما يلغي الحاجة للتفاوض، تطبيق نظام تتبع للمسارات وتقييم متبادل يضمن التزام السائق بإيصال الراكب إلى وجهته النهائية، ومنع عمليات التبديل غير المصرح بها.

3. **توفير خيارات مريحة (حل مشكلة العوائل والخصوصية).**

إتاحة خيار للمسافر (خاصة العوائل) لطلب وصول السيارة إلى باب المنزل، مما يوفر عليهم عناء الانتقال إلى نقطة التجمع.

4. **رفع الكفاءة التشغيلية للسائقين (حل مشكلة طوابير الانتظار):**

استبدال طوابير "الفرزة" بنظام حجوزات إلكتروني فوري، مما يقلل وقت انتظار السائق إلى الصفر ويساعده على ملء "الحمولة الكاملة" بكفاءة.

5. **تمكين إدارة الرحلات (حل مشكلة رحلة العودة):**

توفير أداة للسائقين تمكنهم من جدولة رحلاتهم القادمة، بما في ذلك رحلات العودة، وفتح الحجز عليها مسبقاً لضمان الحصول على ركاب وتقليل الرحلات الفارغة.

6. **بناء نظام نقل تشاركي متكامل:**

بناء أول نظام نقل تشاركي حديث في اليمن يدعم الدفع الإلكتروني، ويوفر إشعارات فورية، ويضمن أمن وخصوصية بيانات المستخدمين.

7. **بناء قناة تواصل آمنة :**توفير نظام دردشة داخلي (In-app Chat) يغني السائق والراكب عن استخدام تطبيقات خارجية أو كشف أرقام هواتفهم في مراحل مبكرة، مما يحافظ على الخصوصية.

8. **تعزيز الثقة عبر "شارة التوثيق" :**إبراز السائقين "المعتمدين" بوضوح لضمان أعلى مستويات الأمان.

1.4 حدود المشروع (Project Scope)

تم تحديد النطاق الوظيفي والتقني لمنصة "سافر" لضمان تركيز الجهود على تقديم حل متكامل وعملي في المرحلة الحالية:

1.4.1 الحدود الوظيفية (Functional Scope)

- **نظام حجز وتخصيص المقاعد :**يتيح اختيار نقطة الانطلاق والوصول، تحديد نوع الرحلة (فورية/مجدولة)، واختيار المقاعد بدقة عبر واجهة تفاعلية.
- **إدارة رحلات السائقين الذكية :**لوحة تحكم تتيح للسائق إنشاء الرحلات، تتبع الحجوزات، وإدارة حالة الرحلة (قادمة، بدأت، مكتملة).
- **نظام المحادثة الفورية(Internal Chat) :** توفير قنوات اتصال مباشرة وآمنة داخل التطبيق بين السائق والركاب لتسهيل التنسيق دون الحاجة لتطبيقات خارجية.
- **إدارة المحفظة الرقمية(Wallet System) :** نظام مالي متكامل لإدارة الأرصدة، تتبع المعاملات، ومعالجة طلبات الشحن والسحب المالي بشكل موثق.
- **مركز الإشعارات الذكي :**نظام إشعارات لحظي يقوم بفلترة التنبيهات بناءً على دور المستخدم (سائق/راكب) لضمان عدم التشتت.
- **التوثيق المتقدم والرقابة :**نظام للتحقق من هويات السائقين ووثائق المركبات من قبل الإدارة لضمان أمان الرحلات.
- **دعم اللغات(Localization) :** دعم كامل للغتين العربية والإنجليزية لضمان سهولة الاستخدام لفئات أوسع.

1.4.2 الحدود التشغيلية:(Operational Scope)

- **النطاق الجغرافي :**يركز المشروع حصراً على النقل بين المحافظات والمدن اليمنية.(Inter-City)
- **نوع الخدمة :**استهداف نقل الركاب كخدمة أساسية، مع إمكانية التوسع المستقبلي.
- **التحكم الإداري المركزي :**توفير لوحة تحكم شاملة (Dashboard) للمشرفين لإدارة المستخدمين، الرحلات، والرقابة المالية.

1.4.3 الحدود المكانية(Spatial Scope / Geographic Boundaries) :

- **منطقة الإطلاق والتغطية :**يشمل النطاق الجغرافي للمشروع الجمهورية اليمنية، مع التركيز تحديداً على خطوط النقل الطويلة والحجوزات بين المحافظات والمدن الرئيسية (مثل: صنعاء، عدن، تعز، حضرموت، مأرب، إب، والحديدة).
- **بيئة التشغيل :**تقتصر الخدمة في مرحلتها الأولى على الطرق البرية الرابطة بين المدن اليمنية، ولا يشمل النطاق الجغرافي الرحلات الدولية أو النقل الداخلي (المشاوير الصغيرة) داخل المربع السكني للمدينة الواحدة.

1.4.4 ما يخرج عن نطاق المشروع(Out of Project Scope)

- **خرائط المسارات المخصصة :**الاعتماد على مزودي الخرائط العالميين مثل (Google Maps) بدلاً من بناء محرك خرائط خاص.
- **اللوجستيات الثقيلة :**عدم دعم نقل البضائع التجارية الضخمة أو الأثاث في المرحلة الحالية.

1.5 المنهجية المستخدمة لإنجاز المشروع:

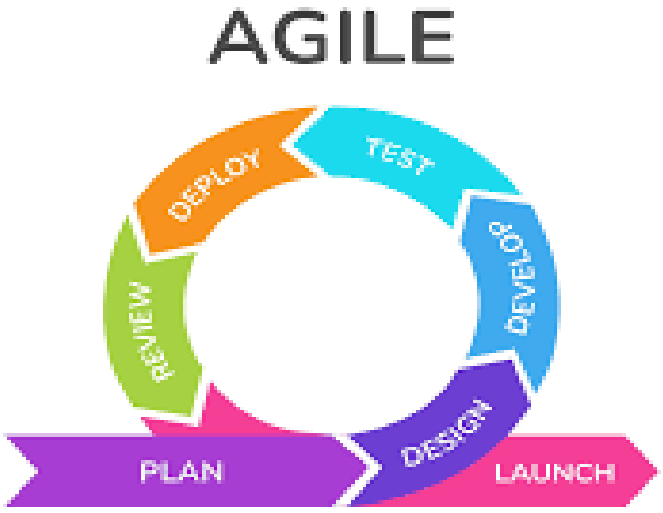
منهجية التطوير الرشيقة:(Agile Methodology)

- **لماذا؟** لأنها تعتمد على التطوير التكراري والمستمر؛ حيث تم تقسيم المشروع إلى "وحدات برمجية " (Modules)تشمل: (وحدة الحسابات والتوثيق، وحدة الرحلات والحجوزات، وحدة المحفظة المالية، وحدة الدردشة والإشعارات).

المراحل المنفذة:

- **التحليل :**دراسة احتياجات سوق النقل اليمني (مشاكل الفرزة، احتياجات العوائل، وتحديات السائقين).
- **التصميم :**هيكلة قواعد البيانات وبناء واجهات المستخدم (UI/UX) باستخدام Flutter لتكون بديهية وسهلة.
- **التدشين البرمجي :(Implementation)** استخدام Laravel لبناء محرك السيرفر (Backend) و Flutterلتطوير تطبيق الهاتف.
- **الاختبار :(Testing)** فحص كافة السيناريوهات البرمجية التي تضمن استقرار النظام ومنها:

- فحص دورة الحجز الكاملة من اختيار المقعد حتى تأكيد الرحلة.
- فحص دقة العمليات المالية في المحفظة (شحن، سحب، خصم العمولات).
- فحص وصول الإشعارات اللحظية ومزامنة رسائل الدردشة باستخدام تقنيةPusher.
- فحص منطق إلغاء الحجوزات والشروط المرتبطة بها لضمان حقوق الأطراف.
- فحص عملية اعتماد وثائق السائقين من لوحة تحكم المسؤول(Admin) .



شكل 1-1: منهجية البحث

1.6 معايير القبول (Acceptance Criteria)

تعد هذه المعايير بمثابة المؤشرات النهائية لنجاح المشروع واستيفائه للمتطلبات الوظيفية والتقنية:

1. **الموثوقية والتحقق :**قدرة النظام على التعامل بصرامة مع حسابات السائقين، بحيث لا يُسمح ببدء الرحلات إلا بعد استكمال الوثائق الثبوتية واعتمادها إدارياً.
2. **كفاءة البحث والحجز :**استجابة النظام الفورية لعروض الرحلات المتاحة عند البحث بالمحافظة أو التاريخ، وإتمام عملية حجز المقاعد في زمن لا يتجاوز ثوانٍ معدودة.
3. **دقة المنظومة المالية :**نجاح تنفيذ العمليات المالية عبر المحفظة الرقمية (خصم، إيداع، عمولات) بدقة متناهية وبدون أخطاء محاسبية.
4. **تزامن البيانات اللحظي :**ضمان وصول الإشعارات الفورية ورسائل الدردشة في الوقت الحقيقي(Real-time)لإبقاء المستخدم على اطلاع دائم.
5. **سهولة الإدارة والتحكم :**تمكين السائق من إدارة دورة حياة الرحلة كاملة (إنشاء، تحديث الحالة، إنهاء) من خلال واجهة مستخدم واضحة وبسيطة.

6. **استقرار الأداء :**قدرة التطبيق على العمل بشكل مستقر ودون انهيار (Crashes) على مختلف إصدارات أنظمة تشغيل الهواتف الذكية.(Android & iOS)

7. **تكامل البيانات :**قدرة لوحة التحكم الإدارية (Admin Panel) على رصد ومراقبة كافة العمليات والرحلات في النظام لضمان الرقابة والجودة.

1.7 تعريف النظام (System Definition)

"ســـافر" هو منظومة رقمية متكاملة تتكون من تطبيقات هواتف ذكية للركاب والســـائقين، مرتبطة بمركز بيانات سحابي ولوحة تحكم إدارية متطورة. تهدف المنصـة إلى إعادة تنظيم قطاع النقل بين المدن اليمنية من خلال ربط مباشر وفعال بين طرفي الخدمة.

يعتمد النظام على أتمتة عمليات الحجز عبر نظام ذكي يراعي الخصوصية، ويوفر بيئة آمنة للمدفوعات عبر المحفظة الإلكترونية، مع دمج قنوات تواصل لحظية وتنبيهات تفاعلية. وبذلك، يعمل "سافر" كمنصة وسيطة تضـــمن الأمان، الشـــفافية في الأســـعار، ورفع الكفاءة التشـــغيلية لمركبات النقل بما يواكب التحول الرقمي الحديث.

1.8 الغرض العام من التطبيق (Purpose)

يتمثـل الغـرض الجـوهري مـن مشـروع "سـافر" فـي رقمنـه وحوكمـة قطـاع النقـل البينـي بـين المحافظـات اليمنيـة، مـن خـلال ابتكـار وسـيط رقمـي ينهـي الأسـاليب التقليديـة المجهـدة. يهـدف المشـروع إلـى خلـق بيئـة تفاعليـة تضـمن للمسـافر حقوقـه فـي الأمـان والشـفافية وتلغـي معاناتـه مـع الانتظـار، وفـي الوقـت ذاتـه، تمـنح السـائق أدوات تشـغيلية احترافيـة ترفـع مـن كفـاءة عملـه واسـتقراره المالي.

1.9 أهمية المشروع (Significance)

تتجلى أهمية المشروع في كونه حلاً متعدد الأبعاد للتحديات اللوجستية الراهنة، ويمكن تلخيص أهميته في النقاط التالية:

- **الأهمية الاجتماعية والإنسانية :**تعزيز أمن المواطن عبر توفير وسيلة نقل موثقة وخاضعة للرقابة، وحماية المسافرين من الاحتيال أو سوء المعاملة، مع توفير سبل الراحة للعوائل والخصوصية التي يفتقدها النظام الحالي.

- **الأهمية الاقتصادية والتشغيلية :**رفع إنتاجية السائقين من خلال القضاء على هدر الوقت في انتظار "الطوابير"، وتمكينهم من جدولة رحلات (الذهاب والعودة) مسبقاً، مما يضمن تدفقاً مالياً مستقراً ويقلل من استهلاك الوقود والموارد في الرحلات الفارغة.

- **الأهمية التقنية والمستقبلية :**يمثل "سافر" خطوة سباقة نحو رقمنه الخدمات الأساسية في اليمن، حيث يوفر بنية تحتية لبيانات النقل، ويدعم توجه الدولة والمجتمع نحو تبني "المدفوعات الإلكترونية" والخدمات الذكية.

1.10 القيود والمعوقات(Constraints & Limitations)

- **الاعتماد على الإنترنت :**يعتمد الأداء الكامل للنظام (خاصة التتبع الآني والحجز) بشكل مطلق على استمرارية الاتصال بالإنترنت وجودة الشبكة.
- **تحديات البنية التحتية :**ضعف تغطية الشبكة في بعض الطرق الداخلية وعدم دقة بيانات الخرائط لبعض المناطق، مما يؤثر على دقة تحديد المواقع.
- **تبني الخدمة :**التحدي المتمثل في تحويل السائقين من النمط التقليدي (الفرزة) إلى النمط الرقمي، مما يتطلب واجهات بسيطة جداً(User Friendly) .
- **التكامل المالي :**الحاجة إلى توافقات برمجية مع بوابات الدفع المحلية المتعددة في اليمن لضمان سلاسة العمليات المالية.

1.11 الفرضيات (Assumptions)

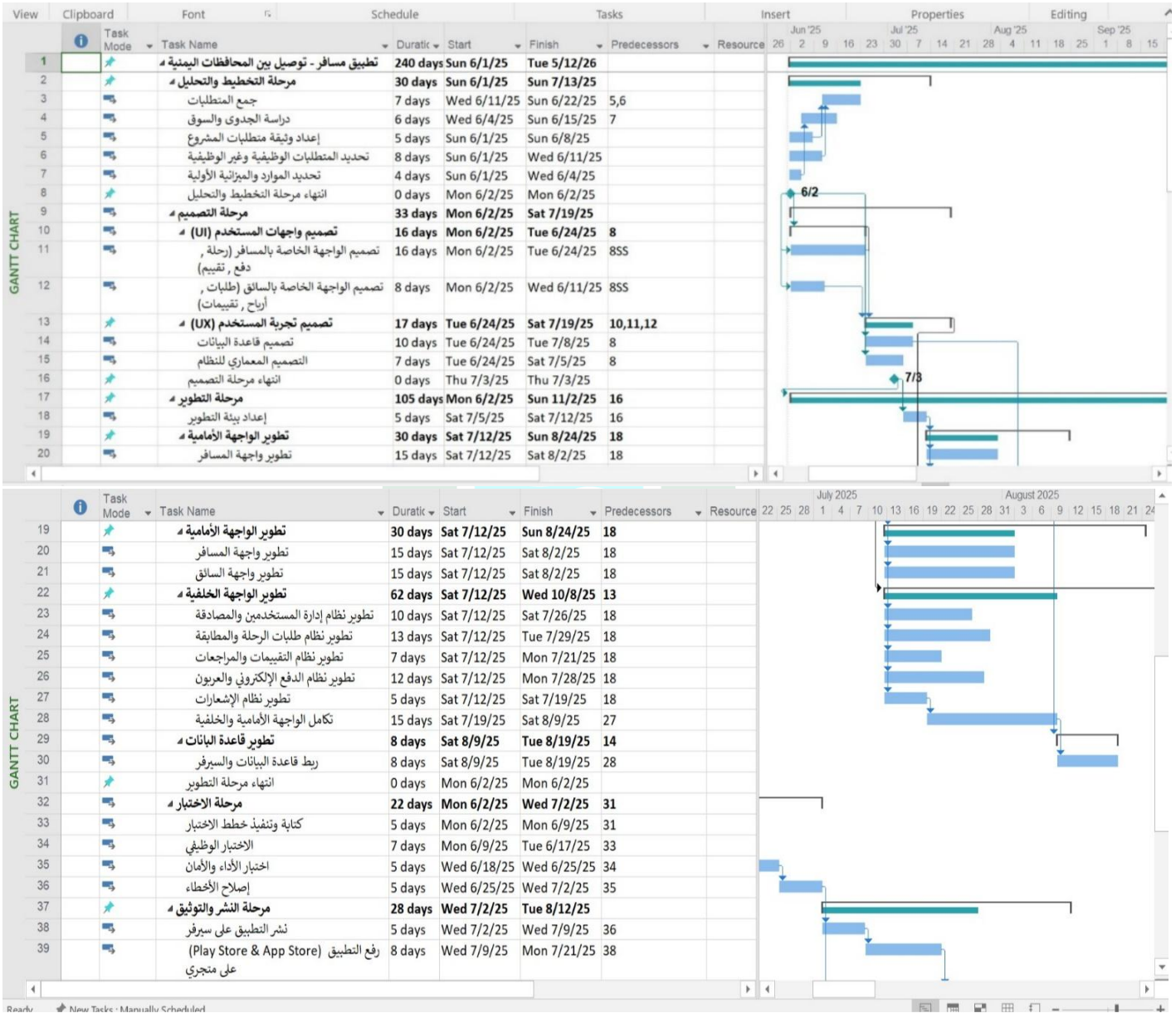
- **الوعي التقني :**افتراض امتلاك الفئات المستهدفة لهواتف ذكية ومعرفة أساسية في التعامل مع التطبيقات.
- **الاستقرار السعري** :قبول المستخدمين لسياسة التسعير الثابتة والشفافة كبديل لعمليات التفاوض التقليدية.
- **الموثوقية المتبادلة :**افتراض التزام المستخدمين بنظام التقييم والتعليقات لتعزيز مصداقية المنصة.

1.12 دورة حياة النظام(Life Cycle Model)

يعتمد النظام على نموذج التطوير المرن(Agile) ، الذي يمثل دورة حياة تطوير تكرارية متسارعة. تضمن هذه المنهجية تحقيق أعلى جودة وفعالية للتطبيق عبر تقسيم عملية التطوير إلى دورات قصيرة(Sprints) ، تشمل كل منها التخطيط، التحليل، التصميم، التنفيذ، والاختبار المستمر.

1.13 الخطة الزمنية (Time Plan)

يوضح هذا الجدول المراحل الزمنية التي تم اتباعها في تنفيذ مشروع "Safer Application and Platform"، ابتداءً من مرحلة جمع المتطلبات وحتى مرحلة الاختبار والتقييم، مع تحديد المدة الزمنية لكل مرحلة لضمان تنفيذ المشروع ضمن الإطار الزمني المحدد.



	Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource
1		تطبيق مسافر - توصيل بين المحافظات اليمنية	240 days	Sun 6/1/25	Tue 5/12/26		
2		مرحلة التخطيط والتحليل	30 days	Sun 6/1/25	Sun 7/13/25		
3		جمع المتطلبات	7 days	Wed 6/11/25	Sun 6/22/25	5,6	
4		دراسة الجدوى والسوق	6 days	Wed 6/4/25	Sun 6/15/25	7	
5		إعداد وثيقة متطلبات المشروع	5 days	Sun 6/1/25	Sun 6/8/25		
6		تحديد المتطلبات الوظيفية وغير الوظيفية	8 days	Sun 6/1/25	Wed 6/11/25		
7		تحديد الموارد والميزانية الأولية	4 days	Sun 6/1/25	Wed 6/4/25		
8		انتهاء مرحلة التخطيط والتحليل	0 days	Mon 6/2/25	Mon 6/2/25		
9		مرحلة التصميم	33 days	Mon 6/2/25	Sat 7/19/25		
10		تصميم واجهات المستخدم (UI)	16 days	Mon 6/2/25	Tue 6/24/25	8	
11		تصميم الواجهة الخاصة بالمسافر (رحلة , دفع , تقييم)	16 days	Mon 6/2/25	Tue 6/24/25	8SS	
12		تصميم الواجهة الخاصة بالسائق (طلبات , أرباح , تقييمات)	8 days	Mon 6/2/25	Wed 6/11/25	8SS	
13		تصميم تجربة المستخدم (UX)	17 days	Tue 6/24/25	Sat 7/19/25	10,11,12	
14		تصميم قاعدة البيانات	10 days	Tue 6/24/25	Tue 7/8/25	8	
15		التصميم المعماري للنظام	7 days	Tue 6/24/25	Sat 7/5/25	8	
16		انتهاء مرحلة التصميم	0 days	Thu 7/3/25	Thu 7/3/25		
17		مرحلة التطوير	105 days	Mon 6/2/25	Sun 11/2/25	16	
18		إعداد بيئة التطوير	5 days	Sat 7/5/25	Sat 7/12/25	16	
19		تطوير الواجهة الأمامية	30 days	Sat 7/12/25	Sun 8/24/25	18	
20		تطوير واجهة المسافر	15 days	Sat 7/12/25	Sat 8/2/25	18	
21		تطوير واجهة السائق	15 days	Sat 7/12/25	Sat 8/2/25	18	
22		تطوير الواجهة الخلفية	62 days	Sat 7/12/25	Wed 10/8/25	13	
23		تطوير نظام إدارة المستخدمين والمصادقة	10 days	Sat 7/12/25	Sat 7/26/25	18	
24		تطوير نظام طلبات الرحلة والمطابقة	13 days	Sat 7/12/25	Tue 7/29/25	18	
25		تطوير نظام التقييمات والمراجعات	7 days	Sat 7/12/25	Mon 7/21/25	18	
26		تطوير نظام الدفع الإلكتروني والعربون	12 days	Sat 7/12/25	Mon 7/28/25	18	
27		تطوير نظام الإشعارات	5 days	Sat 7/12/25	Sat 7/19/25	18	
28		تكامل الواجهة الأمامية والخلفية	15 days	Sat 7/19/25	Sat 8/9/25	27	
29		تطوير قاعدة البيانات	8 days	Sat 8/9/25	Tue 8/19/25	14	
30		ربط قاعدة البيانات والسيرفر	8 days	Sat 8/9/25	Tue 8/19/25	28	
31		انتهاء مرحلة التطوير	0 days	Mon 6/2/25	Mon 6/2/25		
32		مرحلة الاختبار	22 days	Mon 6/2/25	Wed 7/2/25	31	
33		كتابة وتنفيذ خطط الاختبار	5 days	Mon 6/2/25	Mon 6/9/25	31	
34		الاختبار الوظيفي	7 days	Mon 6/9/25	Tue 6/17/25	33	
35		اختبار الأداء والأمان	5 days	Wed 6/18/25	Wed 6/25/25	34	
36		إصلاح الأخطاء	5 days	Wed 6/25/25	Wed 7/2/25	35	
37		مرحلة النشر والتوثيق	28 days	Wed 7/2/25	Tue 8/12/25		
38		نشر التطبيق على سيرفر	5 days	Wed 7/2/25	Wed 7/9/25	36	
39		رفع التطبيق (Play Store & App Store) على متجري	8 days	Wed 7/9/25	Mon 7/21/25	38	

شكل 2-1: الخطة الزمنية

2 الفصل الثاني(Chapter 2)

الخلفية والدراسات السابقة

(Background and Literature Review)

2.1 مكانة التطبيقات في تسهيل الخدمات:

شهدت العقود الأخيرة تحولاً جذرياً في أنماط تقديم الخدمات، حيث أضحت التطبيقات الذكية هي المحرك الرئيس للتغيير الهيكلي في الاقتصادات الحديثة. لم تعد هذه التطبيقات مجرد أدوات تكميلية، بل غدت منصات مركزية تهدف إلى رقمنه العمليات التقليدية المعقدة وتحويلها إلى خطوات رقمية فورية وتفاعلية. ففي قطاع النقل، مكنت هذه التقنيات من تجاوز معضلات الانتظار العشوائي والتفاوض غير المنظم، عبر تقديم حلول الحجز المسبق والتتبع اللحظي(Real-time Tracking) ، مما عزز من كفاءة التشغيل ورفع معايير الموثوقية والأمان. إن هذا الانتقال نحو "الخدمات المعتمدة على البيانات" وضع معياراً جديداً يقوم على السرعة والشفافية، مما يجعله عنصراً جوهرياً لأي مشروع يهدف إلى إعادة تنظيم القطاعات الخدمية الحيوية.

2.2 الخلفية (Background)

يأتي مشروع "سافر" كاستجابة تقنية مباشرة للتحديات الهيكلية التي تواجه قطاع النقل البري بين المحافظات اليمنية. يعتمد هذا القطاع تقليدياً على الآليات التقليدية والمحدودة تتمحور حول "الفرزات" (محطات التجمع التقليدية)، وهو ما يخلق فجوة كبيرة بين متطلبات المسافر الحديث وبين واقع الخدمة المقدمة، وتتجلى مظاهر هذه الفجوة في واقع الخدمة التقليدية:

- **التحديات التي تواجه المسافرين :**الارتهان للانتظار الطويل في الشوارع أو التوجه القسري لنقاط التجمع، مع تقيد الرحلات بأوقات محددة (غالباً في الصباح الباكر)، مما يجعل السفر في أوقات متأخرة أمراً مرهقاً أو مستحيلاً.
- **انعدام الأمان والثقة :**تنتشر ظواهر سلبية مثل غياب الرقابة والموثوقية، حيث يتم تبديل المركبات في منتصف الطريق، بالإضافة إلى غياب الشفافية في الأسعار وإضاعة الوقت في التفاوض المتكرر على الأجرة.
- **هدر الموارد للسائقين :**يواجه السائقون تحدياً كبيراً في إضاعة وقتهم وأيام عملهم داخل طوابير الفرزات لضمان حمولة كاملة، كما يصعب عليهم تنظيم رحلات العودة، مما يؤدي إلى انخفاض كفاءة تشغيل المركبات وهدر اقتصادي.

لذلك، يسعى "سافر" لتقديم حل جذري يؤسس لنموذج نقل ذكي يرتكز على (الوضوح، الأمان، والكفاءة).

الضرورة التقنية:

على عكس التطور التقني العالمي في خدمات النقل الذكي، تفتقر اليمن إلى نظام رقمي موحد وموثوق يربط بين المسافرين والسائقين بكفاءة وأمان، ويدعم وسائل الدفع الحديثة كالمحافظ الإلكترونية.

لذلك، يهدف مشروع "سافر" إلى تقديم حل تقني جذري يستخدم التكنولوجيا الحديثة لتجاوز هذه التحديات التقليدية، ويؤسس لنموذج جديد للنقل بين المدن يركز على الشفافية، الأمان، والكفاءة التشغيلية.

2.3 الدراسات السابقة (Literature Review)

شهد مجال النقل الذكي تطورًا كبيرًا في السنوات الأخيرة، حيث ظهرت العديد من التطبيقات والمنصات التي تهدف إلى تنظيم عملية النقل البري وتحسين تجربة المستخدمين من حيث السرعة، الراحة، والأمان. تعد مراجعة الأنظمة العالمية والإقليمية خطوة مفصلية لاستنباط أفضل الممارسات وتلافي القصور في الأنظمة القائمة:

2.3.1 تطبيقUber

يُعَدّ Uber من أبرز وأوائل تطبيقات النقل الذكي عالميًا، حيث يوفّر منصة رقمية تربط بين السائقين والركاب عبر الهواتف الذكية. يُمكِّن التطبيق المستخدمين من حجز الرحلات بسهولة، وتتبع السائقين في الوقت الفعلي، وتقييم جودة الخدمة بعد إتمام الرحلة، كما يعتمد على خوارزميات تحديد المسار الأفضل وتقدير الأجرة تلقائيًا.

- **نطاق العمل الجغرافي :**يعمل على مستوى **عالمي واسع جداً**، حيث يغطي أكثر من 70 دولة وما يقارب 10,000 مدينة حول العالم (يشمل أمريكا الشمالية، أوروبا، أجزاء واسعة من آسيا، وأمريكا اللاتينية، بالإضافة إلى بعض العواصم الكبرى في الشرق الأوسط).
- **نقاط القوة :**سهولة الاستخدام وسرعة الاستجابة، الاعتماد على أنظمة GPS دقيقة، ونظام تقييم متطور يعزز الثقة.
- **نقاط الضعف :**ارتفاع الأسعار في أوقات الذروة، ومحدودية الدعم الفني في بعض المناطق.

2.3.2 تطبيق Careem

يُعتبر Careem المنافس الإقليمي الأبرز لـ Uber في الشرق الأوسط. يتميز بتكيّفه مع بيئات الدول العربية، وإضافة ميزات محلية مثل الدفع النقدي أو عبر المحافظ الإلكترونية. كما يتيح للمستخدمين جدولة الرحلات مسبقًا ومتابعتها لحظيًا.

- **نطاق العمل الجغرافي :**يركز بشكل أساسي على منطقة الشرق الأوسط وشمال أفريقيا، وباكستان، حيث ينشط بقوة في دول الخليج العربي (السعودية، الإمارات، قطر، إلخ)، مصر، الأردن، العراق، والمغرب العربي.
- **نقاط القوة :**دعم لغات متعددة منها العربية، ومرونة في خيارات الدفع (نقد/إلكتروني)، ونظام ولاء ومكافآت للمستخدمين الدائمين.
- **نقاط الضعف :**يحتاج إلى اتصال إنترنت مستقر، ويعتمد على توفر السائقين النشطين في نفس المنطقة.

2.3.3 تطبيقInDriver

يختلف InDriver عن التطبيقات السابقة في أنه يسمح للراكب بتحديد الأجرة التي يرغب بدفعها، بينما يختار السائقون قبول الرحلة أو رفضها. هذه الميزة تمنح المستخدم حرية أكبر وتقلل من ارتفاع الأسعار المفاجئ.

- **نطاق العمل الجغرافي :**يعمل بنطاق **عالمي وإقليمي مرن**، وينتشر بقوة في أسواق أمريكا اللاتينية، روسيا ودول آسيا الوسطى، وتوسع بشكل كبير جداً مؤخراً في دول الشرق الأوسط وشمال أفريقيا (مثل مصر، السعودية، وبعض المدن المغاربية).

- **نقاط القوة :**حرية التفاوض بين السائق والمسافر، وسهولة واجهة الاستخدام، وسرعة في التواصل بين الطرفين.
- **نقاط الضعف :**ضعف نظام التقييم مقارنةً بـ Uber وCareem، وغياب الرقابة المركزية أحيانًا يؤدي إلى مشكلات في الأمان

2.3.4 تطبيق BlaBlaCar

يُعد BlaBlaCar منصة رائدة عالمياً في مفهوم "مشاركة الرحلات الطويلة(Carpooling) "، حيث يربط بين السائقين الذين يخططون لرحلات بين المدن وبين الركاب الذين يتجهون إلى نفس الوجهة. يتميز هذا التطبيق بتركيزه على الرحلات المجدولة مسبقاً والطويلة، ويتيح للسائق تغطية جزء من تكاليف وقوده بينما يوفر للراكب وسيلة نقل بتكلفة أقل.

- **نطاق العمل الجغرافي :**ينشط بقوة في قارة أوروبا كمركز رئيسي له (فرنسا، إسبانيا، ألمانيا، وغيرها)، بالإضافة إلى تواجده في بعض أسواق أمريكا اللاتينية (مثل البرازيل والمكسيك)، ودول مثل الهند وتركيا وأوكرانيا، وهو النموذج الأقرب لطبيعة مشروع "سافر" من حيث السفر بين المدن.
- **نقاط القوة :**التخصص في النقل بين المدن، التسعير المنخفض للركاب، والقدرة على تلبية متطلبات الرحلات الطويلة المجدولة.
- **نقاط الضعف :**صعوبة إيجاد رحلات فورية (يعتمد على الجدولة المسبقة)، وعدم وجود نظام تتبع دقيق للسائقين في الوقت الحقيقي.

2.3.5 تطبيقات النقل المحلية(Local Transportation Apps)

شهد السوق اليمني في الآونة الأخيرة محاولات جادة لرقمنه قطاع النقل، إلا أن معظم هذه المحاولات ظلت محصورة في أطر عمل ضيقة أو واجهت تحديات حالت دون انتشارها كمنصة وطنية شاملة للنقل البيني. يمكن تصنيف الجهود المحلية الحالية إلى ثلاثة أنماط رئيسة:

1. **المنصات غير الرسمية(Social Media-Based Networks) :** تعتبر مجموعات "فيسبوك" وقنوات "واتساب" هي الوسيلة المحلية الأكثر انتشاراً حالياً لتنسيق الرحلات بين المدن.

- **القصور :**تفتقر هذه الوسائل لأي نوع من أنواع الأمان أو التتبع، ولا توجد ضمانات لحقوق الراكب أو السائق، بالإضافة إلى عشوائية الأسعار وصعوبة البحث عن رحلات مجدولة بدقة.

2. **تطبيقات النقل داخل المدن(Intra-city Startups) :** ظهرت تطبيقات محلية (مثل تطبيق "مشوار "أو راحتك أو محاولات لشركات مكاتب التاكسي) ركزت بشكل أساسي على طلب سيارات الأجرة داخل المدن الرئيسية ومنها:

2.3.5.1 تطبيق خط وصل (Kat wasl)

يُعتبر تطبيق "الوصل" من المبادرات المحلية التي تهدف إلى تنظيم خدمات النقل البري داخل اليمن عبر ربط الركاب بالسائقين من خلال منصة إلكترونية. يركز التطبيق على تسهيل عملية الحجز والتواصل بين الطرفين، وتقديم خدمات نقل أكثر تنظيمًا مقارنة بالطرق التقليدية.

- **نقاط القوة**: سهولة الوصول إلى السائقين، تقليل الاعتماد على التنسيق التقليدي عبر وسائل التواصل، وتوفير واجهة رقمية للحجز.
- **نقاط الضعف**: محدودية الانتشار الجغرافي، وقلة المعلومات الرسمية والتقنية المتاحة حول النظام، بالإضافة إلى ضعف خصائص التتبع والأمان مقارنة بالأنظمة العالمية .

2.3.5.2 تطبيق "يمن تكسي" (Yemen Taxi)

من أوائل التطبيقات التي استهدفت تسهيل طلب سيارات الأجرة داخل المدن اليمنية، وتحديداً في العاصمة صنعاء.

- **نقاط القوة:** تتبع المركبة عبر الخريطة، وتوفير معلومات السائق لزيادة الثقة.
- **نقاط الضعف:** عدم دعمه للرحلات الطويلة بين المحافظات، والاعتماد الكبير على استقرار الإنترنت.

2.3.5.3 تطبيق "راحتك" (Rahtak)

يقدم خدمات النقل والتوصيل داخل المدن ويوفر للمستخدمين إمكانية طلب الوسيلة بطريقة مبسطة.

- **نقاط القوة:** واجهة استخدام سهلة تلائم المستخدم المحلي.
- **نقاط الضعف:** نقص في أنظمة التقييم الأمني المتقدمة، وعدم التخصص في النقل البيني.

2.3.5.4 تطبيق "مشواري" (Mishwar) [8]

منصة تقدم خدمات نقل متعددة (تكاسي، باصات عائلية) مع ميزات اختيار نوع المركبة.

- **نقاط القوة:** تنوع الوسائل ودعم التتبع المباشر للرحلة.
- **نقاط الضعف:** التركيز التشغيلي محصور داخل المدن الرئيسية دون الخطوط الطويلة.

- **القصور :**هذه التطبيقات صُممت لخدمة "المشاوير القصيرة"، ولا تمتلك البنية البرمجية أو اللوجستية التي تراعي خصوصية "السفر الطويل" بين المحافظات، مثل (حجز مقاعد محددة، اختيار نوع المركبة للدور).

3. **أنظمة شركات النقل الجماعي(Fleet Management Systems) :** تمتلك بعض شركات النقل الكبرى (مثل الرويشان أو بن معمر) أنظمة حجز خاصة لأسطول حافلاتها.

- **القصور :** هذه الأنظمة منغلقة على أسطول الشركة فقط، ولا تخدم "السائق الفردي" أو مالكي السيارات الخاصة (7 ركاب) الذين يمثلون الشريحة الأكبر في قطاع النقل البيني اليمني. كما أن هذه الأنظمة غالباً ما تفتقر لميزات التتبع اللحظي للركاب أو الدردشة المباشرة مع السائق.

2.4 الفجوة البحثية(Research Gap)

من خلال المراجعة التحليلية المتعمقة للدراسات والأنظمة السابقة (العالمية والمحلية)، تم تحديد فجوة بحثية وتقنية مركبة يمكن تلخيصها في المحاور التالية:

1. **الفجوة الجغرافية والسياقية(Contextual Gap) :** تركز معظم المنصات العالمية الرائدة (مثل Uber وCareem) على "النقل الحضري (Intra-city) "داخل المدن ذات البنية التحتية المستقرة. وهناك نقص حاد في الأبحاث والتطبيقات الموجهة خصيصاً للنقل البيني (Inter-city) في البيئات النامية التي تواجه تحديات لوجستية وأمنية خاصة، كالحالة اليمنية، حيث يختلف نمط السفر والاحتياجات التنظيمية كلياً عن النقل داخل المدن.
2. **الفجوة الوظيفية والتنظيمية(Functional Gap) :** تعاني الأنظمة القائمة من انقسام حاد؛ فإما أنها أنظمة "حوكمة كبرى" منغلقة على أساطين النقل الجماعي (شركات الحافلات)، أو أنها شبكات "تواصل اجتماعي عشوائية" تفتقر لأدنى معايير الأمان والتنظيم. وتبرز الفجوة هنا في غياب منصة "وسيطة (Intermediary Platform)" قادرة على دمج السائقين الأفراد في منظومة رقمية منظمة توفر ميزات (حجز المقاعد، الدردشة التنسيقية، وتتبع المسار الطويل) في آن واحد.
3. **فجوة الموثوقية والتحقق الرقمي(Trust & Verification Gap) :** بالرغم من اعتماد المسافرين حالياً على "المعرفة الشخصية" كأساس للأمان، إلا أن هذا النموذج يفتقر للقابلية للتوسع (Scalability) والشفافية. وتتمثل الفجوة البحثية في كيفية بناء "نظام ثقة رقمي (Digital Trust System) "يجمع بين التوثيق الإداري اليدوي وبين نظام التقييم التشاركي، لضمان أمن المسافرين في الرحلات الطويلة التي تتسم بمخاطر تزيد عن الرحلات القصيرة.
4. **فجوة التكامل المالي المحلي(Financial Integration Gap) :** تعتمد التطبيقات العالمية على بوابات دفع دولية (Visa/MasterCard) لا تتوفر لشريحة واسعة في السوق المحلي، بينما تفتقر المحاولات المحلية لنظام محفظة رقمية مدمج (Integrated Wallet) يدير دورة المال بين السائق والراكب والمنصة بشكل آلي وموثق، مما يخلق عائقاً أمام التحول الرقمي الكامل في تحصيل الرسوم.

2.5 النظام المقترح "سافر" (Suggested System - SAFER)

بناءً على تحليل القصور في الأنظمة الحالية وتحديد الفجوة البحثية، يُقترح نظام "سافر"(SAFER) ؛ وهو منظومة تقنية متكاملة تهدف إلى رقمنه قطاع النقل البيني (بين المحافظات) وتطويره وفق معايير تكنولوجية حديثة. يمثل النظام المقترح نموذجاً هجيناً (Hybrid Model) يجمع بين مرونة تطبيقات النقل الذكي العالمية وبين الخصوصية اللوجستية والأمنية للبيئة المحلية.

وحدات ومكونات النظام المقترح:

يتألف النظام من ثلاث ركائز أساسية تعمل بشكل متزامن لضمان استمرارية وكفاءة الخدمة:

1. **تطبيق الهاتف المحمول(Mobile Application) :** واجهة برمجية موحدة تدعم نظامي، موجهة لكل من المسافر والسائق بآلية "تبديل الأدوار" أو التسجيل المتخصص. يركز التطبيق على تجربة مستخدم (UX) بسيطة تتيح الحجز اللحظي، إدارة الرحلات، التتبع، والدردشة المباشرة.
2. **المنصة الإدارية والخلفية البرمجية(Backend & Admin Dashboard) :** تمثل "عصب النظام" حيث تدير كافة العمليات المعقدة، مثل معالجة الحجوزات، إدارة قواعد البيانات المشفرة، وتوثيق السائقين يدوياً عبر لوحة تحكم ذكية تضمن الرقابة الكاملة على المنظومة.
3. **نظام الخدمات اللحظية(Real-time Cloud Services):** توظيف تقنيات الدفع اللحظي (WebSockets) وخدمات التنبيهات السحابية ومصادر الخرائط لضمان تحديث البيانات في أجزاء من الثانية.

القيمة الابتكارية للنظام(System Contribution) :

يأتي نظام "سافر" لردم الفجوة البحثية عبر تقديم عدد من الحلول المبتكرة، أهمها:

- توطين التكنولوجيا(Localization) : تصميم النظام ليتوافق مع تحديات الشبكة وتوفر الإنترنت في اليمن، مع دعم المحافظ الإلكترونية المحلية كوسيلة دفع أساسية.
- حوكمة الأمان :الانتقال من مفهوم "الأمان العشوائي" إلى "الأمان المؤسسي" عبر بوابة توثيق السائقين ووثائق المركبات إدارياً قبل تفعيل الحسابات.
- تفعيل النقل التشاركي المنظم :السماح للسائقين الأفراد بجدولة رحلاتهم مسبقاً، مما يرفع الكفاءة التشغيلية للمركبات ويقلل من هدر الوقت والموارد في نظام "الفرزة" التقليدي.

خلاصة القول؛ إن النظام المقترح لا يقدم مجرد "أداة حجز"، بل يؤسس لبيئة نقل رقمية متكاملة تتجاوز القصور التقني والاجتماعي في الحلول القائمة، وتضع معياراً جديداً لجودة خدمات النقل في اليمن والدول النامية المشابهة.

2.6 تحليل المقارنة(Comparative Analysis)

جدول 1-2:تحليل المقارنة

المعيار	In Driver	Careem	Uber	Safer
تحديد المواقع(GPS)	يدعم	يدعم	يدعم	يدعم
الإشعارات اللحظية	يدعم	يدعم	يدعم	يدعم
التقييم بعد الرحلة	لا يدعم	يدعم	يدعم	يدعم
التفاوض على السعر	يدعم	لا يدعم	لا يدعم	يدعم
دعم اللغة العربية	جزئي	يدعم	لا يدعم	يدعم
العمل بين المدن	يدعم	محدود	محدود	يدعم
الأمان والموثوقية	متوسط	عالي	عالي	عالي
سهولة الاستخدام	متوسط	عالي	عالي	عالي

2.7 مستخدمي النظام وخصائصهم (Users)

جدول 2-2: مستخدمين النظام

م	نوع المستخدم	الخصائص والاحتياجات
1	المسافر	✓ يحتاج إلى حجز سريع وسهل. ✓ يحتاج إلى متابعة دقيقة للرحلة. ✓ يعتمد على الإشعارات اللحظية لتفادي التأخير أو التغييرات المفاجئة. ✓ البحث عن رحلة وتصفية النتائج. ✓ عرض تفاصيل الرحلة وإمكانية تتبع مسارها. ✓ حجز مقعد واختيار نوع الدفع (كاش أو إلكتروني). ✓ إمكانية إلغاء الحجز. ✓ التواصل مع السائق عبر التطبيق. ✓ تقييم السائق بعد انتهاء الرحلة.
2	السائق	✓ يحتاج إلى تنظيم وإدارة رحلاته بكفاءة. ✓ متابعة الطلبات وحالة الحجز من المستخدمين. ✓ إنشاء حساب وتوثيقه عبر رفع الوثائق (رخصة، وثيقة سيارة، صورة شخصية). ✓ إنشاء رحلات جديدة وتحديد تفاصيلها. ✓ إدارة الرحلات وتأكيد طلبات الحجز. ✓ بدء الرحلة ومشاركة الموقع. ✓ عرض المحفظة لمراجعة الأرباح المالية. ✓ تأكيد انتهاء الرحلة وإمكانية تقييم الراكب.
3	فريق الدعم الفني	✓ إدارة المستخدمين (إدارة حسابات الركاب والسائقين). ✓ الموافقة على السائقين الجدد وتوثيق حساباتهم. ✓ مراقبة الرحلات الجارية والمباشرة في الوقت الفعلي. ✓ توليد تقارير عامة وإحصائيات عن أداء التطبيق. ✓ إدارة الدعم الفني والرد على شكاوى واستفسارات المستخدمين.

2.8 دراسة الجدوى (Feasibility Study)

تهدف دراسة الجدوى إلى تقييم إمكانية تنفيذ مشروع **Safer Application and Platform** من جميع الجوانب، لضمان نجاحه واستمراريته قبل الانتقال إلى مراحل التطوير الفعلية. وتشمل الدراسة خمسة محاور رئيسية كما يلي:

2.8.1 الجدوى التقنية(Technical Feasibility)

يُعد مشروع "سافر" ممكناً تقنياً، حيث يعتمد على تقنيات حديثة ومتوفرة ومُجربة عالمياً. يتمثل التحليل التقني للنظام في التالي:

- **المعدات الصلبة (Hardware):**

1. **المعدات الصلبة الموجودة (بيئة التطوير):**

- ✓ أجهزة الحاسوب: عدد (5) أجهزة حاسوب محمولة (لابتوب) خاصة بفريق التطوير، بمواصفات مناسبة لتشغيل بيئات التطوير (IDEs) ومحاكات الهواتف (Emulators).
- ✓ هواتف ذكية (Testing Phones): هواتف تعمل بنظام Android لاختبار التطبيق بشكل أولي.
- ✓ تجهيزات الشبكة: مودم (Router) للاتصال بالإنترنت لرفع ومزامنة الأكواد واختبار الاتصال بالـ API.

2. **المعدات المطلوبة (بيئة التشغيل والنشر):**

- سيرفر خاص بقواعد البيانات (Database Server).
- سيرفر خاص بواجهات النظام (Application Server): سيرفر مخصص لاستضافة الواجهة الخلفية (Backend) وإدارة منطق الأعمال، والتعامل مع طلبات الاتصال من تطبيقات الهواتف الذكية.
- سيرفر خاص بالنسخ الاحتياطي (Backup Server).

- **البرمجيات (Software):**

- **تطوير الجانب الأمامي(Frontend) :** إطار عمل Flutterلضمان سرعة التطوير وتوفير نسخة موحدة لنظامي Android وiOS.
- **تطوير الجانب الخلفي(Backend)** : إطار عمل Laravel (PHP)نظراً لقوته في إدارة العمليات المنطقية والأمان.
- **قاعدة البيانات:** نظام MySQLلإدارة البيانات العلائقية بفعالية.
- **الواجهات البرمجية (APIs) والخدمات السحابية:**
  - **Google Maps API**: لتحديد المواقع وحساب المسارات.
  - **Pusher & Firebase (FCM):** لإدارة التنبيهات اللحظية ومزامنة الدردشة فورياً.
  - **بوابات الدفع :**الربط مع واجهات المحافظ الإلكترونية المحلية (مثل كريمي، أم فلوس، ونهضة).

هل النظام تطبيقي أم لا؟ (Is it practical or Not?)

نعم، النظام عملي وتطبيقي. التقنيات المختارة (Flutter, MySQL, Laravel) هي تقنيات ناضجة ومستخدمة على نطاق واسع.

الاحتياج إلى خبراء (Needed Experts)

- خبير تصميم تجربة المستخدم (UI/UX): لضمان تصميم واجهات سهلة وبديهية، خصوصاً للفئات التي قد تكون غير متعلمة تقنياً.
- خبير تكامل أنظمة الدفع (Payment Integration): متخصص في ربط وتأمين الاتصال مع واجهات برمجية (APIs) للمحافظ الإلكترونية المحلية.

- خبير أمن بيانات (Data Security): لضمان تطبيق أفضل الممارسات في تشفير بيانات المستخدمين وحماية النظام من الاختراقات.

2.8.2 الجدوى الاقتصادية(Financial Feasibility)

تعتبر الجدوى المالية (Financial Feasibility) حجر الزاوية في أي مشروع، حيث تحدد مدى قدرة المشروع على تحقيق عائد استثماري (ROI) وتغطية تكاليفه. بناءً على قرار استخدام الخوادم والبنية التحتية المحلية(On-Premise)، تكون التكاليف الرأسمالية الأولية مرتفعة، وتتطلب ميزانية ثابتة للصيانة والتشغيل.

1. نموذج الإيرادات المقترح(Revenue Model) :

المصدر الرئيسي لدخل مشروع "سافر" سيكون عبر فرض رسوم خدمة (عمولة) على كل عملية حجز تتم بنجاح عبر المنصة:

جدول 3-2:نموذج الإيرادات المقترح

المصدر	الوصف	النسبة التقديرية
العمولة على الرحلات	فرض نسبة مئوية من قيمة كل رحلة يتم حجزها وتأكيدها عبر التطبيق، يتم تحصيلها من السائق أو تقسيمها بين الطرفين.	**5% - 10%**من قيمة الرحلة.
رسوم الإعلانات (لاحقاً)	إتاحة مساحات إعلانية محدودة لشركات النقل أو الخدمات ذات الصلة ضمن التطبيق.	غير محدد (يتم إضافته في المراحل المتقدمة).
خدمات مضافة	فرض رسوم بسيطة على خدمات مميزة مثل اختيار مقعد معين أو خدمة التوصيل "من الباب إلى الباب" للعوائل.	رسوم رمزية ثابتة.

2. التكاليف التقديرية(Estimated Costs) :

تنقسم التكاليف إلى شقين رئيسيين: رأسمالية (CAPEX) تُدفع لمرة واحدة، وتشغيلية (OPEX) تُدفع بشكل دوري.

أ. التكاليف الرأسمالية الأولية(Capital Expenditures - CAPEX) :

تشمل تكلفة شراء وتجهيز البنية التحتية للخوادم المحلية (On-Premise) والتراخيص لمرة واحدة:

جدول 2-4 :التكاليف الرأسمالية الأولية

الرقم	أجهزة الحاسوب المستخدمة	الوصف	التكلفة التقديرية (بالدولار الأمريكي)
1	خوادم مركز البيانات	حواسيب محمولة لتطوير وتشغيل النظام	$500
2	أجهزة الشبكة والأمان	شراء خادم فيزيائي واحد مدمج (تطبيق وقواعد بيانات) عالي الكفاءة مع وحدة تخزين احتياطية.	$2,500
3	تراخيص البرمجيات	شراء جدار ناري(Firewall) ، روتر، ومفتاح شبكة مصغر.	**$1,000**(الحد الأعلى للتكلفة)

4	التصميم الجرافيكي	التصميم الجرافيكي	$300
5	تطوير النظام (برمجة)	تصميم الواجهات والشعار والرسومات الترويجية الأولية.	$400
6	أجهزة الحاسوب المستخدمة	تكلفة رواتب فريق التطوير لمدة 3 أشهر (لإنشاءMVP).	**4,500دولار** (1,500دولار شهرياً × 3 أشهر)
الإجمالي التقريبي			$9,200

ب. التكاليف التشغيلية الشهرية(Operating Expenditures - OPEX)

تشمل هذه التكاليف النفقات المتكررة اللازمة لضمان استمرارية عمل النظام بعد الإطلاق:

جدول 5-2 التكاليف التشغيلية الشهرية

البند	الوصف	التكلفة التقديرية (بالدولار الأمريكي)
رواتب فريق التشغيل	رواتب شهرية للمطورين، مهندس صيانة الخوادم، وموظف دعم فني.	$2500– $ 4000
صيانة الخوادم والتجهيزات	تكلفة قطع الغيار، صيانة التبريد، وتحديث البرامج.	$400-$800
اشتراك الإنترنت عالي السرعة	ضمان اتصال إنترنت مستقر وعالي السرعة للخوادم.	$150-$300
خدمات الطرف الثالث(APIs)	رسوم استخدام Google Maps API وغيرها من واجهات APIs الدفع.	$150-$300تعتمد على حسب الاستخدام
تكاليف التسويق والترويج	حملات تسويقية لضم السائقين والركاب للتطبيق.	$500-$1000
التكلفة الإجمالية الشهرية	3,650دولارًا أمريكيًا – 6,400 دولارًا أمريكيًا	

تتطلب الجدوى المالية لمشروع "سافر" التزاماً بـ تمويل أولي كبير (لشراء الخوادم المحلية) وتدفق نقدي شهري ثابت (لتغطية الرواتب والصيانة).

- **نقطة التعادل(Break-Even Point) :** للوصول إلى نقطة التعادل، يجب على المشروع تحقيق إيرادات شهرية تغطي التكلفة التشغيلية (OPEX) الشهرية البالغة تقريباً **5,000$**.

المشروع مجدٍ اقتصادياً إذا تمكن من جذب قاعدة واسعة من السائقين والمسافرين، مما يضمن تحقيق عدد كافٍ من الرحلات اليومية لـ توليد العمولة المطلوبة .على سبيل المثال، إذا كان متوسط أجرة الرحلة 30$ والعمولة 10% (3$ للرحلة)، يجب إكمال ما يقارب 1670رحلة شهرياً (أي حوالي 56 رحلة يومياً) لتغطية الحد الأدنى من التكاليف التشغيلية.

2.8.3 الجدوى التشغيلية(Operational Feasibility)

الأداء(performance):

- **عدد العمليات (Throughput) :**نظراً لطبيعة المشروع (حجوزات بين المحافظات تحدث بوتيرة أقل من سيارات الأجرة داخل المدينة)، يجب أن يكون النظام قادراً على معالجة **100**عملية حجز/توثيق/تتبع في الثانية الواحدة كحد أدنى. (يمكن زيادتها في بيئة التشغيل الفعلية).
- **زمن الاستجابة(Response Time):** يجب أن يكون زمن الاستجابة لعمليات البحث وعرض تفاصيل الرحلة أقل من 0.5 ثانية (500 مللي ثانية) .عمليات التتبع المباشر (الموقع الحالي للسائق) يجب أن تُحدث كل 3-5ثوانٍ لضمان دقة التتبع.

المعلومات(Information) :

- **الادخال(Input) :** سيتم إدخال بيانات دقيقة لـ السائقين (الرخصة، أوراق السيارة، الموقع الحالي) والركاب (بيانات الحجز، تفاصيل الدفع). يتم تصميم واجهات الإدخال للتحقق من صحة البيانات (مثل تنسيق رقم الهاتف وتواريخ صلاحية الوثائق) لتقليل المدخلات الخاطئة.
- **تخزين البيانات(Data Stored)** : يتم تخزين بيانات المستخدمين مشفرة (مثل كلمات المرور وتفاصيل الدفع)، مع ضمان عدم التعارض في سجلات الحجز (مقعد واحد لا يمكن حجزه مرتين) وفي سجلات التوثيق (توثيق كل سائق فريد). يتم استخدام قواعد بيانات منظمة (SQL) لضمان سلامة التعاملات المالية والحجوزات.
- **الإخراج(Output):** سيقوم النظام بإخراج بيانات الحجز (التذاكر)، إشعارات حالة الرحلة (بدأت/انتهت/ألغيت)، وعرض نتائج البحث بشكل صحيح ومفكوك التشفير (عرض اسم السائق ونوع السيارة فقط بعد التأكيد)، مع عرض البيانات بوقت قياسي (زمن الاستجابة).

التحكم (control):

- **القدرة على أداء كل المتطلبات(Too Low) :**
- سيقوم النظام بـ أداء كل المتطلبات بدقة، خصوصاً فيما يتعلق بالدفع الإلكتروني المسبق (لضمان حق السائق) ووظيفة التتبع المباشر (لضمان أمان الراكب).
- **الأمنية والصلاحيات(Too High) :**

توثيق السائقين :لا يمكن للسائق بدء رحلة إلا بعد موافقة المدير على وثائقه.

البيانات :سيتم تشفير بيانات الدفع وتحديد الصلاحيات بدقة بحيث لا يمكن للراكب رؤية معلومات السائق الشخصية (والعكس)، ولا يمكن لأي طرف حذف أو تعديل حجز مؤكد إلا من خلال إجراءات الإلغاء المحددة.

الكفاءة(Efficiency) :

- **ضياع الوقت(Waste Time) :**
- سيتم ضمان عدم تكرار البيانات (كل مقعد يُحجز مرة واحدة فقط). بالنسبة للعمليات، ستتكرر فقط عند تكرار طلب البحث أو الحجز من قبل المستخدم. نظام الإشعارات يضمن وصول معلومات الحجز فوراً دون تأخير يدوي.
- **ضياع المواد(Waste Materials) :**

في حال ضمان عدم حصول اي مشاكل في الكهرباء او الحرائق فإنه لن يكون هناك أي خسارة للتجهيزات

- **الجهد(Effort) :**

سيتم بذل جهد مضاعف لبناء النظام لضمان أفضل تجربة مستخدم .(UX) أما فيما يتعلق بالاستخدام، فإن النظام مصمم لتقليل الجهد على الراكب في البحث والحجز، وعلى السائق في إدارة حجوزاته ومقاعده.

الخدمات(Service) :

- **عدم الدقة أو عدم التناسق:** تم بناء النظام لحل مشكلة عدم الموثوقية. سيوفر النظام :الموثوقية في مطابقة الرحلات، الدقة في النتائج المعروضة (المقاعد المتاحة والسعر)، وسرعة الاستجابة، مما يقضي على مشكلات عدم التناسق بين ما هو معلن وما هو متاح.
- **سهولة التعلم/الاستخدام:** النظام يمتاز بسهولة الاستخدام بسبب تصميم واجهات متفاعلة ومبسطة للسائق والراكب، تشابه تطبيقات النقل العالمية المعروفة، مما يقلل من منحنى تعلم المستخدم.
- **عدم المرونة(Inflexible) :**

سيكون النظام قادر على التعامل مع أية مستجدات (إضافة مميزات جديدة أو تحسين أحد الوظائف).

- **عدم التوافق(Incompatible) :**

التطبيق مصمم ليكون متوافقاً مع منصات التشغيل السائدة(iOS/Android) ، ويتوافق مع الأنظمة الخارجية الضرورية) بوابات الدفع، واجهات الخرائط (API باستخدام معايير اتصال قياسية) مثل(REST APIs).

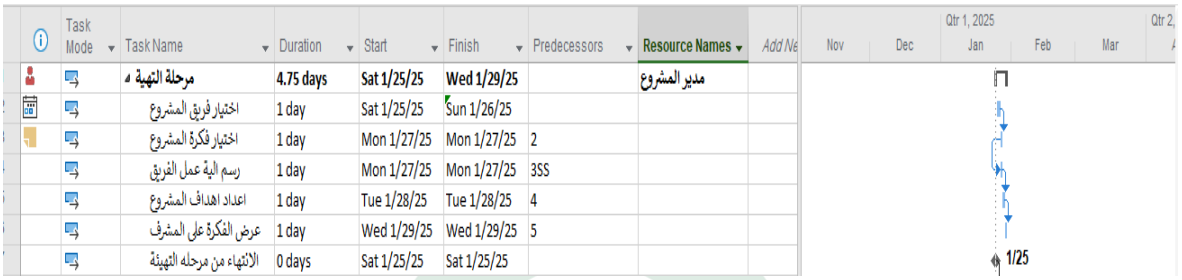
2.8.4 الجدوى الزمنية(Scheduling Feasibility)

اعتماداً على ضوء المنهجية المتبعة في هذا النظام فقد تم تقسيم الفترة الزمنية إلى عدة مراحل كالتالي:

- **مرحلة التهيئة (5 أيام):**

جدول 2-6 : مرحلة التهيئة

م	المهمة	الفترة	
		بدء المهمة	انتهاء المهمة
1	اختيار فريق المشروع	2025/1/25	2025/1/26
2	اختيار فكرة المشروع	2025/1/27	2025/1/29
3	رسم الية عمل المشروع	2025/1/29	2025/1/30
4	اعداد اهداف المشروع	2025/1/30	2025/1/30
5	عرض الفكرة على المشرف	2025/2/1	2025/2/1



شكل 2-1: مرحلة التهيئة

- **مرحلة التحليل (26 أيام):**

جدول 2-7 :مرحلة التحليل

م	المهمة	الفترة	
		بدء المهمة	انتهاء المهمة
1	جمع البيانات	2025/1/26	2025/1/28
2	تحليل البيانات	2025/1/28	2025/1/29
3	جمع المتطلبات	2025/1/29	2025/2/2
4	تحليل المتطلبات	2025/1/29	2025/1/29
5	تسليم الجزء الاول من التحليل	2025/2/2	2025/2/3
6	نمذجة عمليات النظام	2025/2/2	2025/3/3

شكل 2-2 : مرحلة التحليل

- **مرحلة التصميم (23 أيام):**

جدول 8-2 :مرحلة التصميم

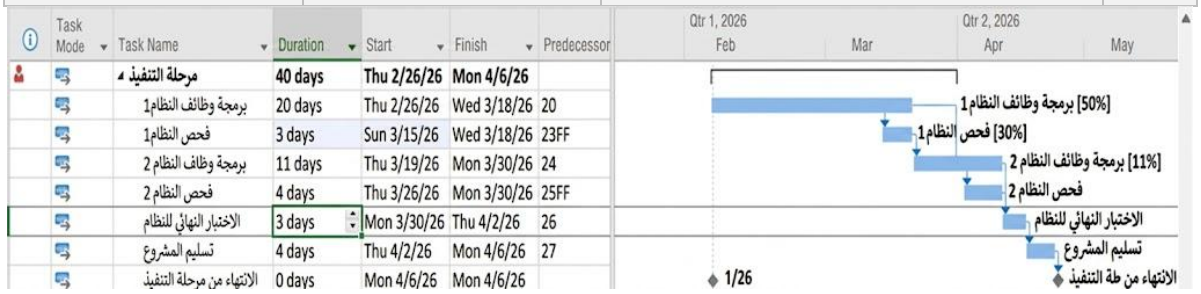
م	المهمة	الفترة	
		بدء المهمة	انتهاء المهمة
1	**تصميم واجهات المستخدم(UI)**	2025/6/2	2025/6/24
2	**تصميم الواجهة الخاصة بالمسافر (رحلة، دفع، تقييم)**	2025/6/2	2025/6/24
3	**تصميم الواجهة الخاصة بالسائق (طلبات، أرباح، تقييمات)**	2025/6/2	2025/6/11
4	**تصميم تجربة المستخدم(UX)**	2025/6/24	2025/7/19
5	**تصميم قاعدة البيانات**	2025/6/24	2025/7/8
6	**التصميم المعماري للنظام**	2025/6/24	2025/7/5

شكل 3-2 : مرحلة التصميم

- **مرحلة التنفيذ (32 أيام):**

جدول 9-2: مرحلة التنفيذ

م	المهمة	الفترة	
		بدء المهمة	انتهاء المهمة
1	برمجة وظائف النظام1	2026/2/26	2026/3/18
2	فحص النظام1	2026/3/15	2026/3/18
3	برمجة وظائف النظام 2	2026/3/19	2026/3/30
4	فحص النظام 2	2026/3/26	2026/3/30
5	الاختبار النهائي للنظام	2026/3/30	2026/4/2
6	تسليم المشروع	2026/4/2	2026/4/6

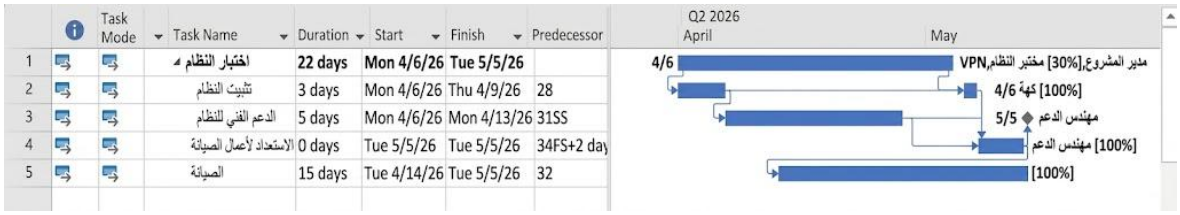


شكل 4-2 : مرحلة التنفيذ

- **مرحلة اختبار النظام (23 أيام):**

جدول 10-2مرحلة اختبار النظام

م	المهمة	الفترة	
		بدء المهمة	انتهاء المهمة
1	تثبيت النظام	2026/4/6	2026/4/9
2	الدعم الفني للنظام	2026/4/6	2026/4/13
3	الصيانة	2026/4/14	2026/5/5



شكل 5-2: اختبار النظام

2.8.5 الجدوى القانونية(Legal Feasibility)

المشروع لا يتعارض مع أي قوانين محلية أو دولية متعلقة باستخدام البيانات أو خدمات النقل .يتم الالتزام بمعايير حماية الخصوصية من خلال تشفير بيانات المستخدمين وحماية معلوماتهم الشخصية وفقًا للسياسات العامة لاستخدام التطبيقات الذكية كما يمكن تطوير التطبيق مستقبلًا بما يتماشى مع القوانين المنظمة لخدمات النقل الذكي في الدولة.

3 الفصل الثالث (Chapter 3)

تحليل المتطلبات والنمذجة

(Requirements Analysis and Modeling)

3.1 المقدمة(Introduction)

يهدف هذا الفصل إلى تقديم تحليل شامل ومتكامل لمتطلبات نظام "سافر"، وذلك لتحديد كافة الجوانب الوظيفية وغير الوظيفية المنوطة بالمنصة. يمثل هذا التحليل حجر الزاوية في دورة حياة تطوير النظام(SDLC) ؛ حيث يتم تحويل احتياجات المستخدمين إلى مواصفات فنية دقيقة ونماذج تحليلية (Diagrams) يستند إليها المصمم والمبرمج في مراحل التنفيذ، لضمان بناء نظام متماسك يخلو من الفجوات البرمجية ويحقق أهداف المشروع بكفاءة.

3.2 متطلبات النظام(System Requirements)

تُعرف متطلبات النظام بأنها المواصفات والشروط الفنية والوظيفية التي يجب أن يلبيها النظام لضمان تحقيق أهدافه المنشودة بكفاءة عالية. وهي تشكل "العقد" التقني بين المستخدم والمطور.

3.3 نطاق التحليل(Scope of Analysis)

يشمل التحليل كافة العمليات التشغيلية التي تغطيها منصة "سافر" لربط المسافرين والسائقين في النقل البيني، وهي:

- **إدارة الهوية :**التسجيل، الدخول، وتوثيق وثائق السائقين إدارياً.
- **إدارة الرحلات :**إنشاء الرحلات، البحث المتقدم، وتتبع حالات الرحلة (حالية/منتهية).
- **إدارة التواصل :**نظام الدردشة الفورية المدمج.(In-app Chat)
- **إدارة المحفظة الرقمية :**معالجة العمليات المالية، العمولات، وطلبات السحب والشحن.
- **إدارة المستخدمين :**لوحة تحكم إدارية (Admin Dashboard) للرقابة الشاملة والدعم الفني.

3.4 المتطلبات الوظيفية(Functional Requirements)

المتطلبات الوظيفية تصف الوظائف والمهام التي يجب أن يؤديها النظام لتلبية أهدافه. يوضح الجدول التالي أهم المتطلبات الوظيفية لنظام "سافر":

جدول 3-1: المتطلبات الوظيفية

رقم	المتطلب الوظيفي	الوصف(Description)
FR-01	تسجيل مستخدم جديد(User Registration)	يستطيع المستخدم (مسافر/سائق) إنشاء حساب جديد بإدخال الاسم ورقم الهاتف وكلمة المرور.
FR-02	تسجيل الدخول(User Login)	التحقق من بيانات المستخدم وتسجيل الدخول الآمن.
FR-03	إدارة الملف الشخصي(Profile Management)	تعديل بيانات المستخدم الشخصية (الاسم، الصورة، رقم الهاتف...).
FR-04	البحث عن الرحلات(Trip Search)	البحث عن رحلات متوفرة بين مدينتين مع تحديد التاريخ والوقت.
FR-05	حجز الرحلة(Trip Booking)	تنفيذ عملية حجز إلكتروني وحجز المقاعد المتاحة.
FR-06	إدارة الرحلات للسائق(Driver Trip Management)	تمكين السائق من إنشاء الرحلات وتعديلها أو إلغائها.

FR-07	نظام الإشعارات(Notification System)	إرسال إشعارات فورية عند تأكيد الحجز أو تغييره.
FR-08	التقييم(Rating System)	السماح للمسافر بتقييم السائق بعد الرحلة.
FR-09	لوحة تحكم المشرف(Admin Dashboard)	إدارة المستخدمين، الرحلات، التقييمات والتقارير.
FR-10	التقارير(Reporting Module)	توليد تقارير عن عدد الرحلات والحجوزات الإيرادات.

3.5 المتطلبات غير الوظيفية(Non-Functional Requirements)

تحدد المتطلبات غير الوظيفية معايير جودة النظام، مثل الأداء(Performance) ، الأمان(Security) ، القابلية للتوسع (Scalability) وغيرها.

جدول 2-3: المتطلبات الغير وظيفية

رقم	المتطلب غير الوظيفية	الوصف(Description)
NFR-01	الأداء(Performance)	زمن الاستجابة للعمليات الأساسية لا يتجاوز 2 ثانية.
NFR-02	الأمان(Security)	حماية البيانات عبر HTTPS وتشفير كلمات المرور باستخدام.crypt
NFR-03	الاعتمادية(Reliability)	استمرار النظام بالعمل بنسبة لا تقل عن 99% شهرياً لضمان جاهزية الخدمة.
NFR-04	قابلية الاستخدام(Usability)	واجهة بسيطة وواضحة للمستخدمين من مختلف المستويات.
NFR-05	التوافق(Compatibility)	يعمل على أجهزة Android الحديثة ويُدعم التحديث المستقبلي لـiOS.
NFR-06	القابلية للصيانة(Maintainability)	بنية كودية منظمة ودليل تطوير موثق للمبرمجين.
NFR-07	الخصوصية(Privacy)	احترام خصوصية المستخدمين وعدم مشاركة بياناتهم دون إذن.
NFR-08	التوسع(Scalability)	إمكانية إضافة خوادم جديدة بسهولة عند زيادة عدد المستخدمين.

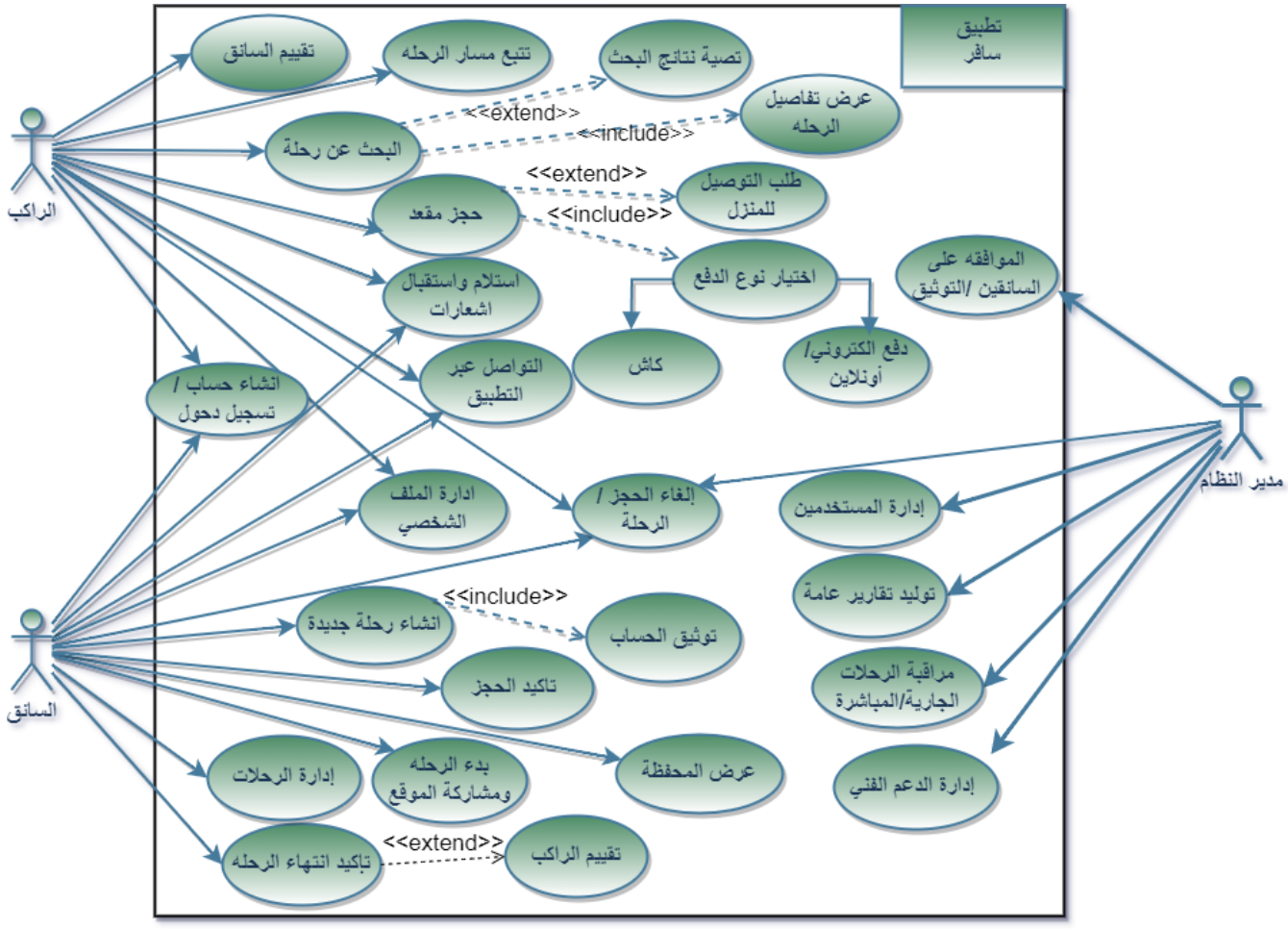
3.6 لغة النمذجة الموحدة(UML)

هي لغة رسومية قياسية تُستخدم لوصف وتصميم وتوثيق أنظمة البرمجيات. تهدف إلى توفير معيار مرئي موحد يساعد المطورين ومحللي النظم على فهم وتحليل هيكل وسلوك النظام قبل البدء ببرمجته، وذلك من خلال مجموعة من المخططات (مثل مخططات حالة الاستخدام، والفئات، والأنشطة) التي توضح تفاصيل النظام وكيفية تفاعل المستخدمين معه.

أهم مخططات لغة النمذجة الموحدة (UML) المستخدمة في تحليل وتصميم الأنظمة هي:

- **مخطط حالة الاستخدام(Use Case Diagram)** : يوضح وظائف النظام الأساسية وكيفية تفاعل المستخدمين (Actors)معها.
- **مخطط النشاط(Activity Diagram)** : يمثل سير العمل (Workflow) والخطوات المتسلسلة لتنفيذ عملية معينة داخل النظام.
- **مخطط الفئات(Class Diagram)** : يعرض الهيكل البرمجي وقواعد البيانات من خلال توضيح الفئات (Classes)والعلاقات بينها.
- **مخطط التسلسل(Sequence Diagram)** : يوضح كيفية انتقال البيانات والرسائل بين أجزاء النظام المختلفة عبر الزمن لتنفيذ مهمة محددة.
- **مخطط الحالة(State Machine Diagram)** : يتتبع التغيرات التي تطرأ على حالة كائن معين (مثل: طلب رخصة "قيد المراجعة" ثم "مقبول").

3.6.1 حالات الاستخدام(Use Cases Descriptions)



شكل 3-1:مخطط حالة الاستخدام

في هذا القسم يتم شرح أهم حالات الاستخدام التي تمثل السيناريوهات الفعلية لعمل النظام.

شرح مخطط حالات الاستخدام

1. **النظام(System)**

النظام هو "تطبيق سافر"، والممثل بحدود المربع الكبير.

2. **الممثلون(Actors)**

هناك ثلاثة ممثلين (مستخدمين) يتفاعلون مع النظام:

1. **الراكب(Passenger) :** الشخص الذي يبحث عن رحلة ويحجزها.
2. **السائق(Driver) :** الشخص الذي يقدم خدمة النقل.
3. **مدير النظام(System Administrator) :** الشخص الذي يشرف على النظام ويديره.

3. **حالات الاستخدام (Use Cases) والعلاقات**

أ. وظائف الراكب(Passenger)

يتفاعل "الراكب" مع النظام للقيام بالآتي:

- **إنشاء حساب / تسجيل دخول :**(مشترك مع السائق) لإنشاء ملف شخصي أو الوصول إليه.
- **إدارة الملف الشخصي :**(مشترك مع السائق ومدير النظام) لتعديل بياناته الشخصية.
- **البحث عن رحلة :**للبحث عن الرحلات المتاحة.
  - هذه الحالة يمكن أن **تُوسَّع (<<extend>>)** بـ **"تصفية نتائج البحث"**، مما يعني أن تصفية النتائج هي وظيفة اختيارية بعد البحث.
- **عرض تفاصيل الرحلة :**لرؤية معلومات عن رحلة معينة.
  - هذه الحالة يمكن أن **تُوسَّع (<<extend>>)** بـ **"تتبع مسار الرحلة"**، أي أن الراكب يمكنه اختيار تتبع الرحلة على الخريطة بعد عرض تفاصيلها.
- **حجز مقعد :**لحجز مكان في الرحلة.
  - هذه الحالة **تتضمن (<<include>>)** بالضرورة **"اختيار نوع الدفع** ."لا يمكن الحجز بدون تحديد طريقة الدفع.
  - "اختيار نوع الدفع" ينقسم إلى حالتين" :**كاش** "أو **"دفع الكتروني/اونلاين."**
  - حالة "حجز مقعد" يمكن أن تُوسَّع **(<<extend>>)** بـ **"طلب التوصيل للمنزل"**، وهي ميزة اختيارية.
- **إلغاء الحجز / الرحلة :**(مشترك مع السائق) لإلغاء رحلة محجوزة.
- **التواصل عبر التطبيق :**(مشترك مع السائق) للتحدث مع السائق.
- **استلام واستقبال اشعارات :**(مشترك مع مدير النظام) لتلقي تنبيهات من النظام.
- **تقييم السائق :**لتقييم السائق بعد انتهاء الرحلة.

ب. وظائف السائق(Driver)

يتفاعل "السائق" مع النظام للقيام بالآتي:

- **إنشاء حساب / تسجيل دخول:**(مشترك مع الراكب).
- **إدارة الملف الشخصي:**(مشترك مع الراكب ومدير النظام).
- **إنشاء رحلة جديدة :**لعرض رحلة جديدة للركاب.
- **إدارة الرحلات :**لمتابعة رحلاته المجدولة.
- **تأكيد الحجز :**لقبول طلبات الحجز من الركاب.

- هذه الحالة **تتضمن (<<include>>)** بالضرورة **"توثيق الحساب"**، مما يعني أن السائق يجب أن يكون حسابه موثقاً ليتمكن من تأكيد الحجز.

- **إلغاء الحجز / الرحلة :**(مشترك مع الراكب).
- **بدء الرحلة ومشاركة الموقع :**لبدء الرحلة الفعلية ومشاركة موقعه.
- **تأكيد انتهاء الرحلة :**للإعلان عن اكتمال الرحلة.

- هذه الحالة يمكن أن **تُوسَّع (<<extend>>)** بـ **"تقييم الراكب"**، أي أن السائق يمكنه اختيار تقييم الراكب بعد إنهاء الرحلة.

- **عرض المحفظة :**لمراجعة أرباحه المالية.
- **التواصل عبر التطبيق :**(مشترك مع الراكب) للتحدث مع الراكب.

ج. وظائف مدير النظام(System Administrator)

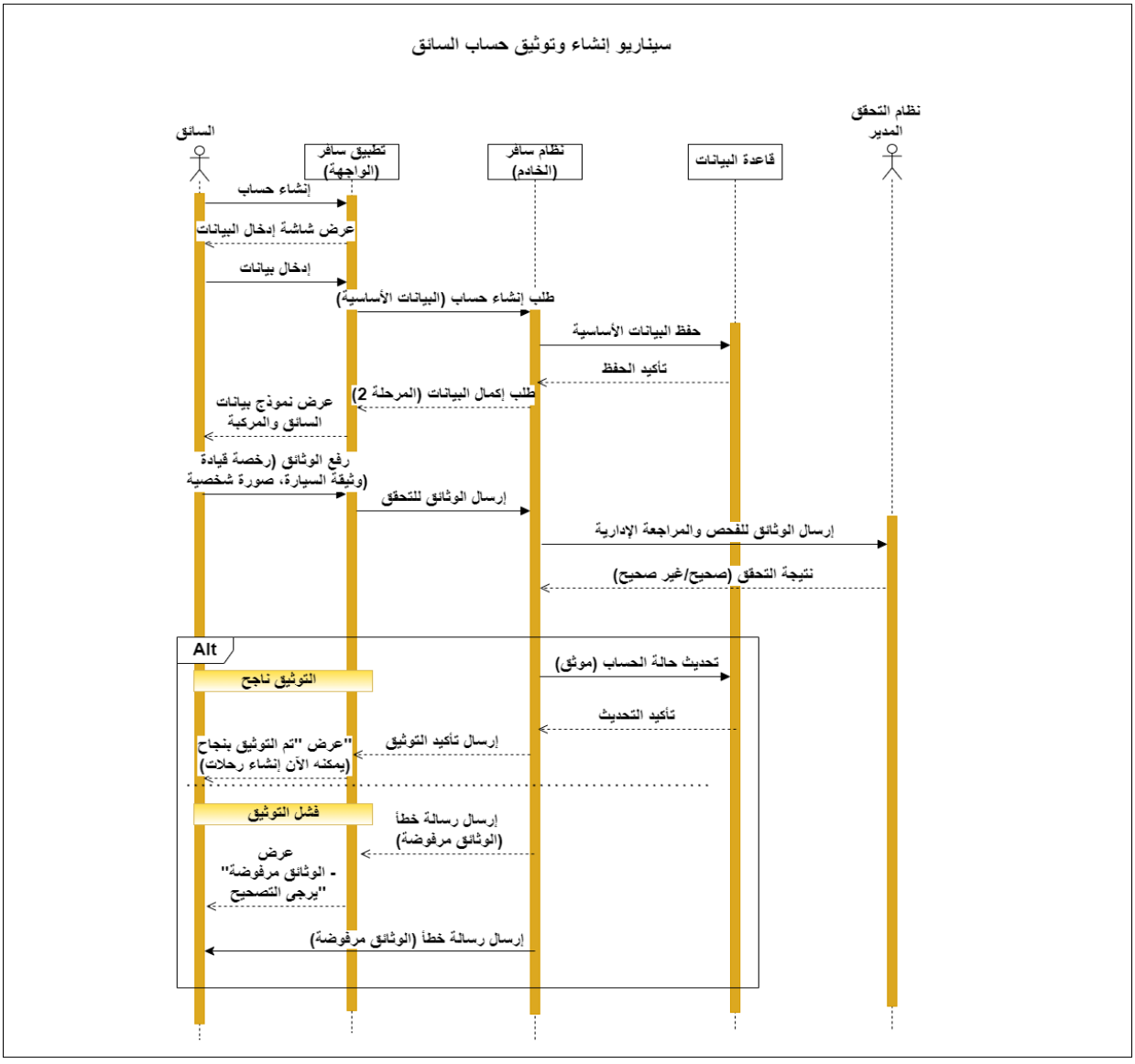
يتفاعل "مدير النظام" مع النظام للقيام بالآتي:

- **إدارة المستخدمين :**لإدارة حسابات الركاب والسائقين.
- **الموافقة على السائقين/التوثيق :**لمراجعة طلبات السائقين الجدد والموافقة عليها.
- **توثيق الحساب :**(تُستخدم أيضاً من قبل "تأكيد الحجز" للسائق) لتوثيق الحسابات.
- **مراقبة الرحلات الجارية/المباشرة :**لمتابعة الرحلات الحالية في الوقت الفعلي.
- **توليد تقارير عامة :**لاستخراج إحصائيات وتقارير عن أداء التطبيق.
- **إدارة الدعم الفني :**للرد على شكاوى واستفسارات المستخدمين.
- **إدارة الملف الشخصي:(**مشترك مع الراكب والسائق).
- **استلام واستقبال اشعارات:(** مشترك مع الراكب).

3.6.2 مخطط التسلسل (Sequence Diagram)

مخطط التسلسل (Sequence Diagram) يوضح تسلسل العمليات التي يقوم بها الفاعلون في النظام.

مخطط التسلسل لعملية إنشاء وتوثيق حساب السائق:



شكل 2-3: مخطط التسلسل لعملية إنشاء وتوثيق حساب السائق

توضيح سيناريو إنشاء وتوثيق حساب السائق

يوضح بالتفصيل الخطوات والتفاعلات بين المكونات المختلفة لإتمام عملية "سيناريو إنشاء وتوثيق حساب السائق" في "تطبيق سافر."

المكونات (المشاركون) في المخطط:

1. **السائق :**المستخدم (Actor) الذي يقوم بإنشاء الحساب.
2. **تطبيق سافر (الواجهة) :**واجهة المستخدم (UI) التي يتفاعل معها السائق.

3. **تطبيق سائق (الخادم) :**الخادم الخلفي (Backend) الذي يعالج الطلبات والمنطق.
4. **قاعدة البيانات :**المكان الذي يتم فيه تخزين المعلومات.
5. **نظام التحقق/المدير :**النظام أو الشخص المسؤول عن مراجعة الوثائق والموافقة عليها.

تسلسل الخطوات الموضحة:

المرحلة الأولى: إنشاء الحساب الأساسي

1. **السائق** يضغط على "إنشاء حساب" في **الواجهة**.
2. **الواجهة** تعرض له "شاشة إدخال البيانات."
3. **السائق** يقوم بـ "إدخال بيانات" (مثل الاسم، البريد الإلكتروني، كلمة المرور).
4. **الواجهة** ترسل "طلب إنشاء حساب (البيانات الأساسية)" إلى **الخادم**.
5. **الخادم** يقوم بـ "حفظ البيانات الأساسية" في **قاعدة البيانات**.
6. **قاعدة البيانات** ترد بـ "تأكيد الحفظ" إلى **الخادم**.

المرحلة الثانية: طلب التوثيق (إكمال البيانات).

7. **الخادم** يرسل للـ **واجهة** "طلب إكمال البيانات (المرحلة 2).
8. **الواجهة** تعرض لـ **السائق** "نموذج بيانات السائق والمركبة ."
9. **السائق** يقوم بـ "رفع الوثائق" (مثل رخصة القيادة، وثيقة السيارة، صورة شخصية) .
10. **الواجهة** ترسل "الوثائق للتحقق" إلى **الخادم**.

المرحلة الثالثة: المراجعة الإدارية.

11. **الخادم** يرسل "الوثائق للفحص والمراجعة الإدارية" إلى **نظام التحقق/المدير** .
12. بعد المراجعة، يقوم **نظام التحقق/المدير** بإرسال "نتيجة التحقق (صحيح/غير صحيح)" إلى **الخادم**.

المرحلة الرابعة: نتائج التحقق (كتلة – "Alt" البدائل) يوجد هنا سيناريوهان بناءً على النتيجة:

- **في حال "التوثيق ناجح:"**
    1. **الخادم** يرسل "تحديث حالة الحساب (موثق)" إلى **قاعدة البيانات**.
    2. **قاعدة البيانات** ترد بـ "تأكيد التحديث."
    3. **الخادم** يرسل "تأكيد التوثيق" إلى **الواجهة**.
    4. **الواجهة** تعرض للسائق رسالة: "تم التوثيق بنجاح، يمكنه الآن إنشاء رحلات."
- **في حال "فشل التوثيق:"**
    1. **الخادم** يرسل "رسالة خطأ (الوثائق مرفوضة)" إلى **الواجهة**.

2. **الواجهة** تعرض للسائق رسالة: "الوثائق مرفوضة، يرجى التصحيح."

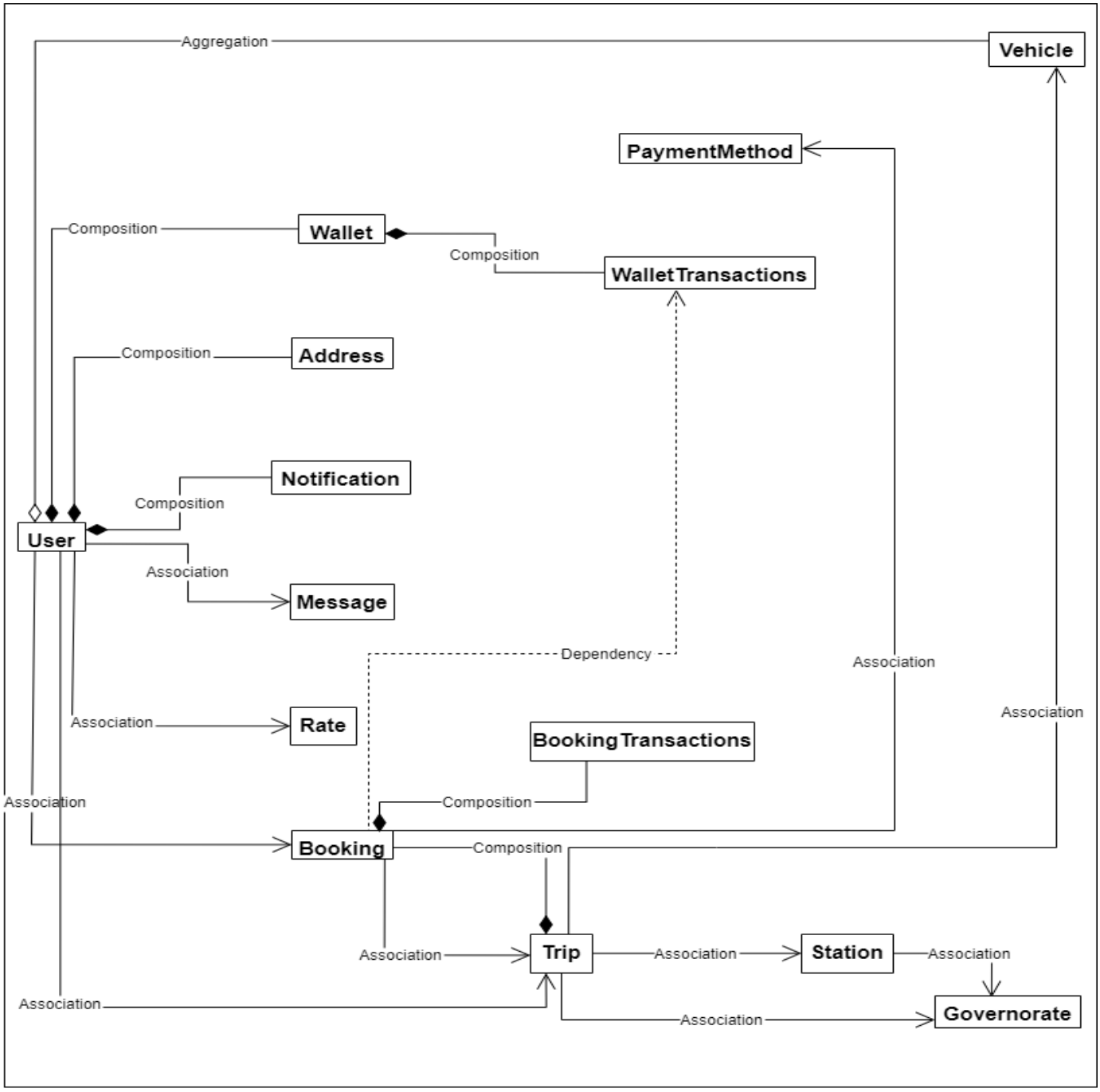
3.6.3 مخطط الفئات (Class Diagram)

- يوضح الشكل (3-3) مخطط الفئات (Class Diagram):

شكل 3-3 : مخطط الفئات

3.6.4 علاقات مخطط الفئات

- يوضح الشكل (3-4) مخطط الفئات (Class Diagram):



شكل 4-3: علاقات مخطط الفئات

شرح العلاقات بين الفئات :تم استخدام أربعة أنواع من العلاقات في مخطط الفئات. فيما يلي شرح لكل نوع من العلاقات المستخدمة:

1- **جدول المستخدم(User)** : المستخدم مرتبط بالحجز بالعلاقة (Association) ارتباط: حيث أن المستخدم (الراكب) يقوم بتقديم طلب حجز للرحلة في النظام.

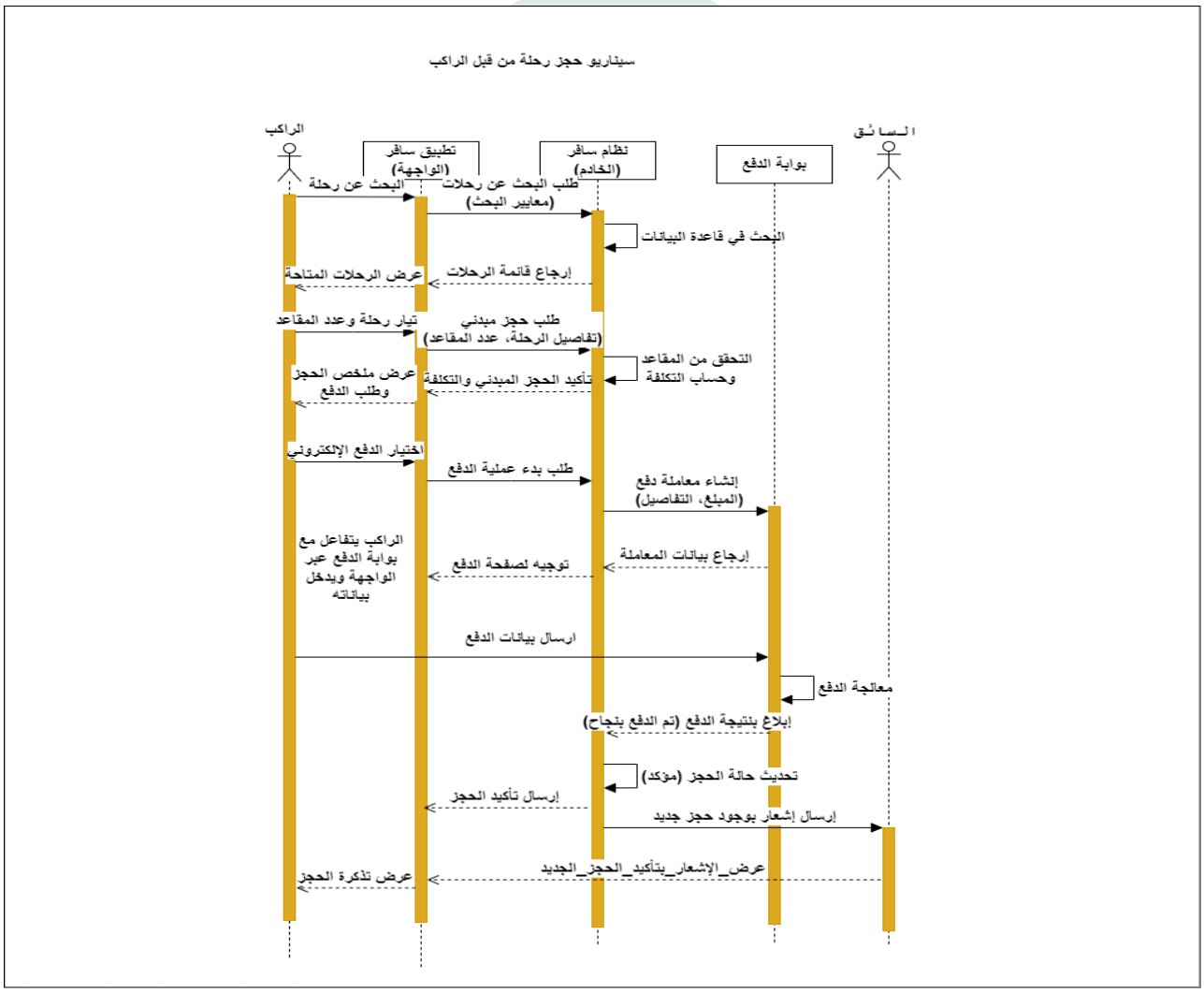
2- **جدول الرحلة(Trip) :** الرحلة مرتبطة بالحجوزات التابعة لها بالعلاقة (Composition) تركيب: حيث أن هذه الحجوزات جزء لا يتجزأ من الرحلة ولا يمكن أن تتواجد في النظام بدون وجود رحلة قائمة تتبع لها.

3- **جدول الحجز(Booking)** : الحجز مرتبط بسندات الدفع وعمليات المحفظة بالعلاقة (Dependency)اعتمادية: حيث يعتمد تأكيد الحجز وإصدار التذكرة على استخدام معلومات العملية المالية (Transactions) للتحقق من الدفع.

4- **جدول المركبة(Vehicle) :** جدول المركبة مرتبط بالسائق (User) بعلاقة (Aggregation) تجميع: حيث أن المركبة كيان مستقل عن السائق ويمكن أن تتواجد في النظام بدون الارتباط بسائق معين، أو يمكن إعادة تعيينها لسائق آخر.

3.6.5 مخطط التسلسل (سيناريو حجز رحلة من قبل الراكب)

- يوضح الشكل (5-3) سيناريو حجز رحلة من قبل الراكب:



شكل 5-3: مخطط التسلسل لسيناريو حجز رحلة من قبل الراكب

توضيح سيناريو حجز رحلة من قبل الراكب

يوضح الخطوات التقنية والتفاعلات بين مختلف أجزاء النظام لإتمام عملية الحجز والدفع.

المكونات (المشاركون) في المخطط:

1. **الراكب:** المستخدم (Actor) الذي يقوم بالحجز.
2. **السائق:** المستخدم (Actor) الذي يستقبل إشعار الحجز.
3. **تطبيق سافر (الواجهة):** واجهة المستخدم (UI) التي يتفاعل معها الراكب.
4. **نظام سافر (الخادم):** الخادم الخلفي (Backend) الذي يعالج المنطق.
5. **قاعدة البيانات:** نظام تخزين البيانات.
6. **بوابة الدفع:** النظام الخارجي المسؤول عن معالجة الدفع الإلكتروني.

تسلسل الخطوات الموضحة:

المرحلة الأولى: البحث واختيار الرحلة

1. **الراكب** يرسل "طلب البحث عن رحلة (معايير البحث)" إلى **الواجهة**.
2. **الواجهة** تمرر الطلب إلى **الخادم**.
3. **الخادم** يبحث في **قاعدة البيانات**.
4. **قاعدة البيانات** ترجع "قائمة الرحلات" إلى **الخادم**.
5. **الخادم** يرسل "عرض الرحلات المتاحة" إلى **الواجهة** ليعرضها للراكب.
6. **الراكب** يختار رحلة ويحدد عدد المقاعد ويرسلها لـ **واجهة**.

المرحلة الثانية: الحجز المبدئي والدفع

7. الواجهة ترسل "طلب حجز مبدئي" (بتفاصيل الرحلة وعدد المقاعد) إلى الخادم.
8. الخادم يتحقق من توفر المقاعد ويحسب التكلفة بالتواصل مع قاعدة البيانات.
9. قاعدة البيانات تؤكد الحجز المبدئي والتكلفة.
10. الخادم يرسل "عرض ملخص الحجز وطلب الدفع" إلى الواجهة.
11. الراكب يختار "الدفع الإلكتروني".
12. الواجهة تطلب من الخادم "بدء عملية الدفع".

المرحلة الثالثة: معالجة الدفع الإلكتروني

13. الخادم ينشئ "معاملة دفع" (بالمبلغ والتفاصيل) ويرسلها إلى بوابة الدفع.

14. بوابة الدفع ترد بـ "بيانات المعاملة".

15. الخادم يرسل "توجيه لصفحة الدفع" إلى الواجهة.

16. الراكب يتفاعل مع صفحة الدفع (التي تظهر عبر الواجهة) ويدخل بياناته.

17. الواجهة ترسل "بيانات الدفع" (التي أدخلها الراكب) إلى بوابة الدفع.

18. بوابة الدفع تقوم بـ "معالجة الدفع".

المرحلة الرابعة: تأكيد الحجز

19. بوابة الدفع تبلغ الخادم بنتيجة الدفع (نفترض "تم الدفع بنجاح").

20. الخادم يرسل "تحديث حالة الحجز (مؤكد)" إلى قاعدة البيانات.

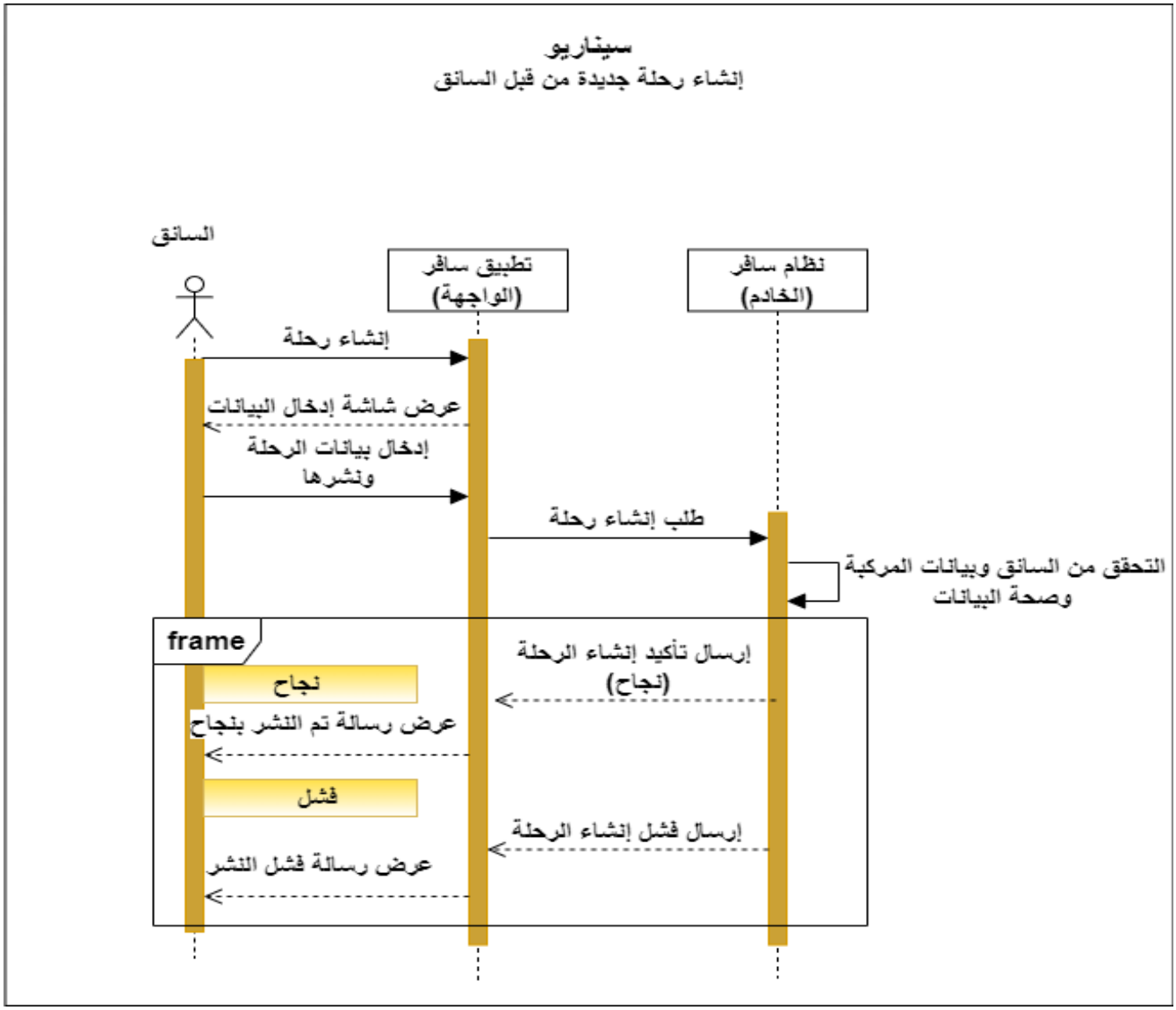
21. الخادم يرسل "إرسال تأكيد الحجز" إلى الواجهة.

22. في نفس الوقت، الخادم يرسل "إشعار بوجود حجز جديد" إلى السائق.

23. الواجهة تعرض للـ راكب "إشعار بتأكيد الحجز" و"تذكرة الحجز".

3.6.6 مخطط التسلسل سيناريو إنشاء رحلة جديدة من قبل السائق

- يوضح الشكل (3-6) سيناريو حجز رحلة من قبل الراكب:



شكل 3-6 :سيناريو إنشاء رحلة جديدة من قبل السائق

سيناريو إنشاء رحلة جديدة من قبل السائق

ويصف الخطوات التي تتم بين السائق والتطبيق لإضافة رحلة جديدة للنظام.

المكونات (المشاركون) في المخطط:

1. **السائق:** المستخدم (Actor) الذي ينشئ الرحلة.
2. **تطبيق سافر (الواجهة):** واجهة المستخدم (UI) التي يتفاعل معها السائق.
3. **نظام سافر (الخادم):** الخادم الخلفي (Backend) الذي يعالج الطلب.

تسلسل الخطوات الموضحة:

1. **السائق** يضغط على "إنشاء رحلة" في **الواجهة**.
2. **الواجهة** تعرض لـ **السائق** "شاشة إدخال البيانات".

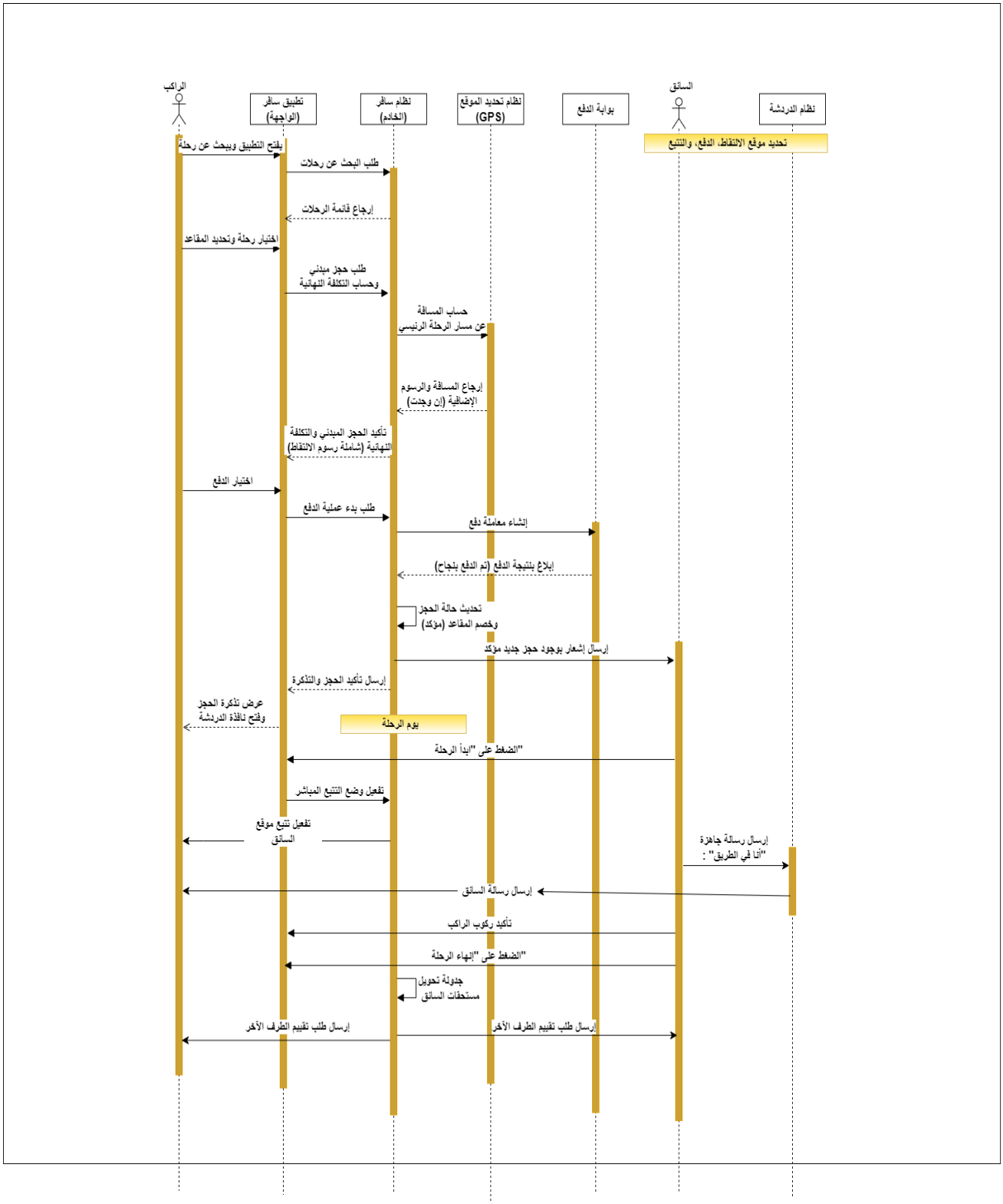
3. **السائق** يقوم بـ "إدخال بيانات الرحلة ونشرها" (مثل الوجهة، الوقت، السعر).
4. **الواجهة** ترسل "طلب إنشاء رحلة" إلى **الخادم**.
5. **الخادم** يقوم بعملية تحقق داخلية: "التحقق من السائق وبيانات المركبة وصحة البيانات".

بعد ذلك، يدخل المخطط في إطار (frame) يوضح نتيجتين محتملتين:

- **في حال "النجاح" (النجاح):**
  1. **الخادم** يرسل "تأكيد إنشاء الرحلة (نجاح)" إلى **الواجهة**.
  2. **الواجهة** تعرض لـ **السائق** رسالة: "تم النشر بنجاح".
- **في حال "فشل" (Failure):**
  1. **الخادم** يرسل "فشل إنشاء الرحلة" إلى **الواجهة** (بسبب بيانات غير صحيحة مثلاً).
  2. **الواجهة** تعرض لـ **السائق** رسالة: "فشل النشر".

3.6.7 مخطط التسلسل سيناريو رحلة الراكب من البحث عن الرحلة إلى التقييم النهائي

- يوضح الشكل (7-3) سيناريو حجز رحلة من قبل الراكب:



شكل 7-3: مخطط التسلسل سيناريو رحلة الراكب من البحث عن الرحلة إلى التقييم النهائي

سيناريو رحلة الراكب من البحث عن الرحلة إلى التقييم النهائي

هذا المخطط هو أشمل مخطط في المجموعة، حيث يغطي مراحل الحجز، والدفع، وتتبع الرحلة، وإنهاء الرحلة.

المكونات (المشاركون) في المخطط:

1. **الراكب:** المستخدم الذي يطلب الرحلة ويقوم بتقييمها.
2. **السائق:** المستخدم الذي يقود الرحلة ويقوم بتقييم الراكب.
3. **تطبيق سافر (الواجهة):** واجهة المستخدم (UI) لدى الراكب.
4. **نظام سافر (الخادم):** الخادم الخلفي الذي يدير منطق التطبيق.
5. **نظام تحديد المواقع (GPS):** نظام خارجي أو خدمة خاصة بالتطبيق لتحديد وتتبع الموقع.
6. **بوابة الدفع:** النظام الخارجي لمعالجة المعاملات المالية.
7. **نظام الدردشة:** نظام خارجي أو خاص بالتطبيق للتواصل بين الراكب والسائق.

تسلسل الخطوات الموضحة:

المرحلة الأولى: البحث والحجز والدفع

1. **الراكب** يبدأ بـ "إدخال التفاصيل والبحث عن رحلة".
2. **الواجهة** ترسل "طلب بحث عن رحلات" إلى **الخادم**.
3. **الخادم** يرجع "قائمة الرحلات" لـ **واجهة**.
4. **الراكب** يختار رحلة ويحدد مقاعده.
5. **الواجهة** ترسل "طلب حجز مبدئي" إلى **الخادم**.
6. **الخادم** يقوم بـ "حساب المسافة والتكلفة التقديرية" بالاستعانة بـ **نظام GPS**.
7. **الخادم** يرجع "التكلفة والرسوم" لـ **واجهة**.
8. **الخادم** يرسل "طلب تأكيد الحجز (التكلفة النهائية، نوع الدفع)" إلى **قاعدة البيانات** (ضمنياً).
9. **الراكب** يختار الدفع.
10. **الواجهة** ترسل "طلب بدء عملية دفع" إلى **الخادم**.
11. **الخادم** يقوم بـ "إنشاء معاملة دفع" مع **بوابة الدفع**.
12. **بوابة الدفع** تبلغ **الخادم** بـ "نتيجة الدفع (تم الدفع بنجاح)".
13. **الخادم** يقوم بـ "تحديث حالة الحجز (مؤكد)" في **قاعدة البيانات**.
14. **الخادم** يرسل "إشعار بوجود حجز جديد" إلى **السائق**.

15. **الخادم** يرسل "تأكيد الحجز وتذكرة" لـ **واجهة**.

16. **الواجهة** تعرض لـ **راكب** "تذكرة الحجز ورقم تذكرة الدردشة".

المرحلة الثانية: بدء وتتبع الرحلة

17. **السائق** يرسل رسالة "أنا في الطريق" عبر **نظام الدردشة** للراكب.

18. **السائق** يضغط على "البدء بالرحلة" (ويبدأ تفعيل تتبع الموقع).

19. **نظام GPS** يرسل "الموقع الحالي" لـ **خادم**.

20. **الخادم** يرسل "تفعيل وضع التتبع المباشر" لـ **واجهة**.

21. **الواجهة** تعرض لـ **الراكب** "تفعيل تتبع موقع السائق".

22. **السائق** يرسل "رسالة استقبال" (مثل "وصلت، انتظرني").

23. **الخادم** يرسل "تأكيد ركوب الراكب" (بعد أن يقوم السائق بتأكيد صعود الراكب).

المرحلة الثالثة: إنهاء الرحلة والتقييم

24. **السائق** يضغط على "إنهاء الرحلة".

25. **الخادم** يقوم بـ "جدولة تحويل مستحقات السائق".

26. **الخادم** يرسل "طلب تقييم الطرف الآخر" إلى **الواجهة** (لتقييم السائق).

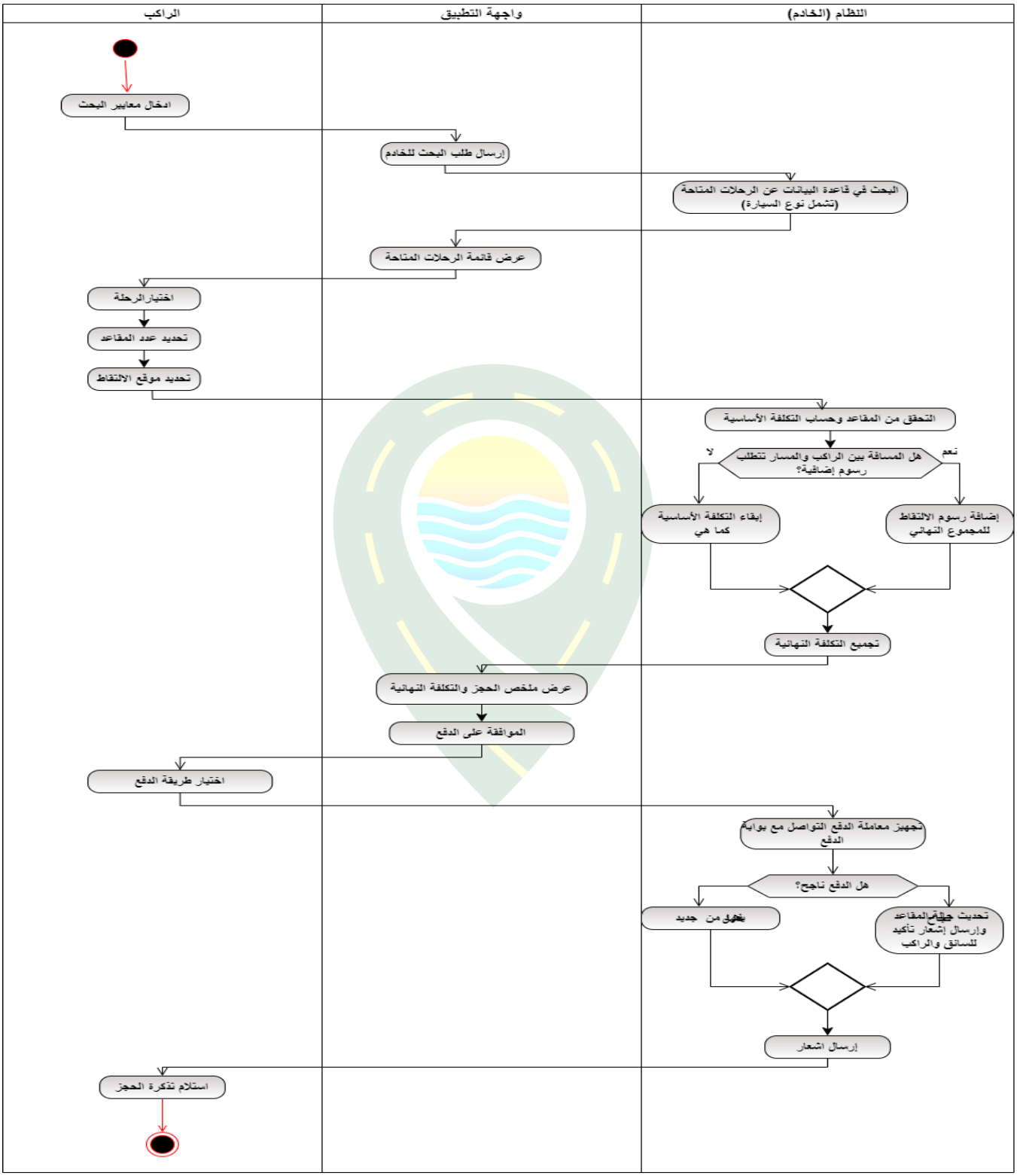
27. **الواجهة** تعرض لـ **الراكب** "طلب تقييم السائق".

28. **الخادم** يرسل "طلب تقييم الطرف الآخر" إلى **السائق** (لتقييم الراكب).

29. **السائق** يقوم بالتقييم (هذه الخطوة ضمناً).

3.6.8 مخطط النشاط لسيناريو: حجز رحلة والدفع وتأكيد الالتقاط

- يوضح الشكل (3-8) مخطط النشاط لسيناريو: حجز رحلة والدفع وتأكيد الالتقاط



شكل 3-8 :مخطط النشاط حجز رحلة

هذا المخطط هو "مخطط النشاط" (Activity Diagram):

إنه يوضح تدفق خطوات العمل (النشاطات) اللازمة لإتمام عملية "البحث، الحجز، والدفع" في تطبيق الرحلات، حيث يركز على سير العمل والقرارات بدلاً من التفاعل الزمني بين المكونات.

المكونات الأساسية (Swimlanes):

1. **الراكب:** المستخدم الذي يتخذ القرارات ويقوم بالإدخالات.
2. **واجهة التطبيق:** الشاشة التي يتفاعل معها الراكب.
3. **النظام (الخادم):** الخادم الخلفي الذي يقوم بالمنطق والحسابات.

تسلسل النشاطات:

المرحلة الأولى: البحث واختيار الرحلة

1. **الراكب** يبدأ بإدخال "معايير البحث" (نقطة البداية).
2. **واجهة التطبيق** ترسل "رسال طلب البحث" إلى **النظام (الخادم)**.
3. **النظام (الخادم)** يبحث في قاعدة البيانات عن الرحلات المتاحة (مع تضمين معايير مثل نوع السيارة).
4. **النظام (الخادم)** يعرض "قائمة الرحلات المتاحة" في **واجهة التطبيق**.
5. **الراكب** يقوم بـ:
   - "اختيار الرحلة".
   - "تحديد عدد المقاعد".
   - "تحديد موقع الالتقاط".

المرحلة الثانية: حساب التكلفة

6. **النظام (الخادم)** يقوم بـ "التحقق من المقاعد وحساب التكلفة الأساسية".
7. **النظام (الخادم)** يطرح سؤالاً (قرار - Diamond): "هل المسافة بين الراكب والمسار تتطلب رسوم إضافية؟"
   - **إذا كانت الإجابة "لا":** يتم الإبقاء على "التكلفة الأساسية كما هي".
   - **إذا كانت الإجابة "نعم":** يتم "إضافة رسوم الانعطاف للوصول" (لتحديد الموقع) للتجميع النهائي.
8. يتم "تجميع التكلفة النهائية" في **النظام (الخادم)**.
9. **واجهة التطبيق** تعرض "ملخص الحجز والتكلفة النهائية".

المرحلة الثالثة: الدفع وتأكيد الحجز

10. **الراكب** يقوم بـ "الموافقة على الدفع".

11. **الراكب** يقوم بـ "اختيار طريقة الدفع".

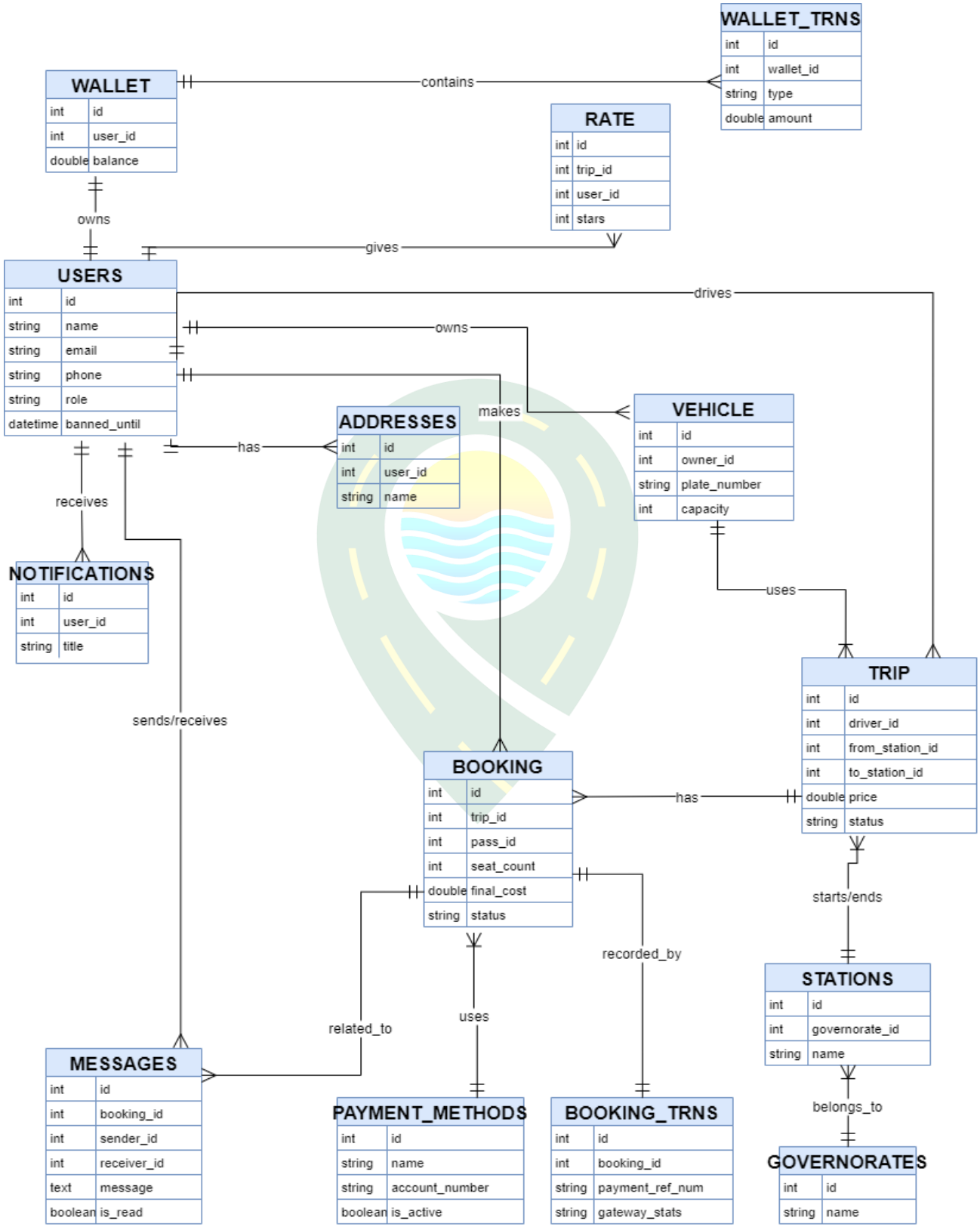
12. **النظام (الخادم)** يقوم بـ "تجهيز معاملة الدفع للتواصل مع بوابة الدفع".

13. **النظام (الخادم)** يطرح سؤالاً (قرار - Diamond): "هل الدفع ناجح؟"

- **إذا كانت الإجابة "نعم" (الدفع ناجح):**
  - **النظام (الخادم)** يقوم بـ "تحديث حالة المقاعد وإرسال تأكيد للسائق والراكب".
  - **النظام (الخادم)** يرسل "إرسال إشعار" (إشعار تأكيد).
  - **الراكب** يقوم بـ "استلام تذكرة الحجز" (نهاية العملية).
- **إذا كانت الإجابة "لا" (الدفع فاشل):** (يظهر تدفق سهم واحد بدون تفاصيل، مما يعني أن العملية ستتوقف أو تعود إلى خطوة سابقة لم يتم رسمها هنا).

3.6.9 مخطط الكيانات والعلاقات(Entity Relationship Diagram - ERD)

- يوضح الشكل (3-9) مخطط الكيان والعلاقة لسيناريو: حجز رحلة والدفع وتأكيد الالتقاط



شكل 9-3: مخطط الكيان والعلاقة(**ERD**)

شرح مخطط ERD

يمثل العلاقات بين الجداول الأساسية داخل قاعدة البيانات.

1. نواة النظام (المستخدمون والكيانات التابعة)

- **USERS (المستخدمون):** هو الجدول المركزي؛ كل مستخدم له دور **(Role)** محدد، سواء كان راكباً أو سائقاً.
- **WALLET (المحفظة):** ترتبط بالمستخدم بعلاقة (1:1) ؛ حيث يمتلك كل مستخدم محفظة مالية واحدة فقط لتخزين رصيده وإدارة عملياته.
- **VEHICLE (المركبات):** ترتبط بالسائق بعلاقة (N:1) ؛ حيث يمكن للسائق تسجيل أكثر من مركبة، بينما تتبع المركبة سائقاً واحداً في الرحلة الواحدة.

2. دورة حياة الرحلة والحجز

- **TRIP (الرحلة):** الكيان الرابط للنظام؛ يحددها السائق، وتُستخدم فيها مركبة معينة، وتنطلق من محطة (Station) وصولاً إلى أخرى.
- **BOOKING (الحجز):** يمثل حلقة الوصل بين الراكب والرحلة؛ حيث يقوم الراكب بحجز مقاعد في رحلة معينة مع تحديد حالة الحجز (مؤكد، ملغي، إلخ).
- **PAYMENT_METHODS (طرق الدفع):** تتيح للمستخدم اختيار وسيلة الدفع المناسبة (المحفظة، الحوالة، إلخ) عند إتمام الحجز.

3. العمليات المالية والتوثيق

- **WALLET_TRNS (عمليات المحفظة):** سجل تفصيلي لكل حركة مالية داخل المحفظة (إيداع أو سحب).
- **BOOKING_TRNS (سجلات الدفع):** تربط الحجز بالعملية المالية، وتوثق رقم المرجع (**Reference Number)** وحالة الدفع لضمان الشفافية والموثوقية.

4. الجغرافيا والمواقع

- **GOVERNORATES (المحافظات):** جدول مرجعي يحتوي على أسماء المحافظات.
- **STATIONS (المحطات):** تتبع كل محطة محافظة معينة، وتُستخدم لتحديد نقاط الانطلاق والوصول الدقيقة للرحلات.
- **ADDRESSES (العناوين):** تتيح للركاب حفظ مواقعهم المفضلة (المنزل، العمل) لتسهيل عملية طلب الرحلات.

5. التواصل والتفاعل

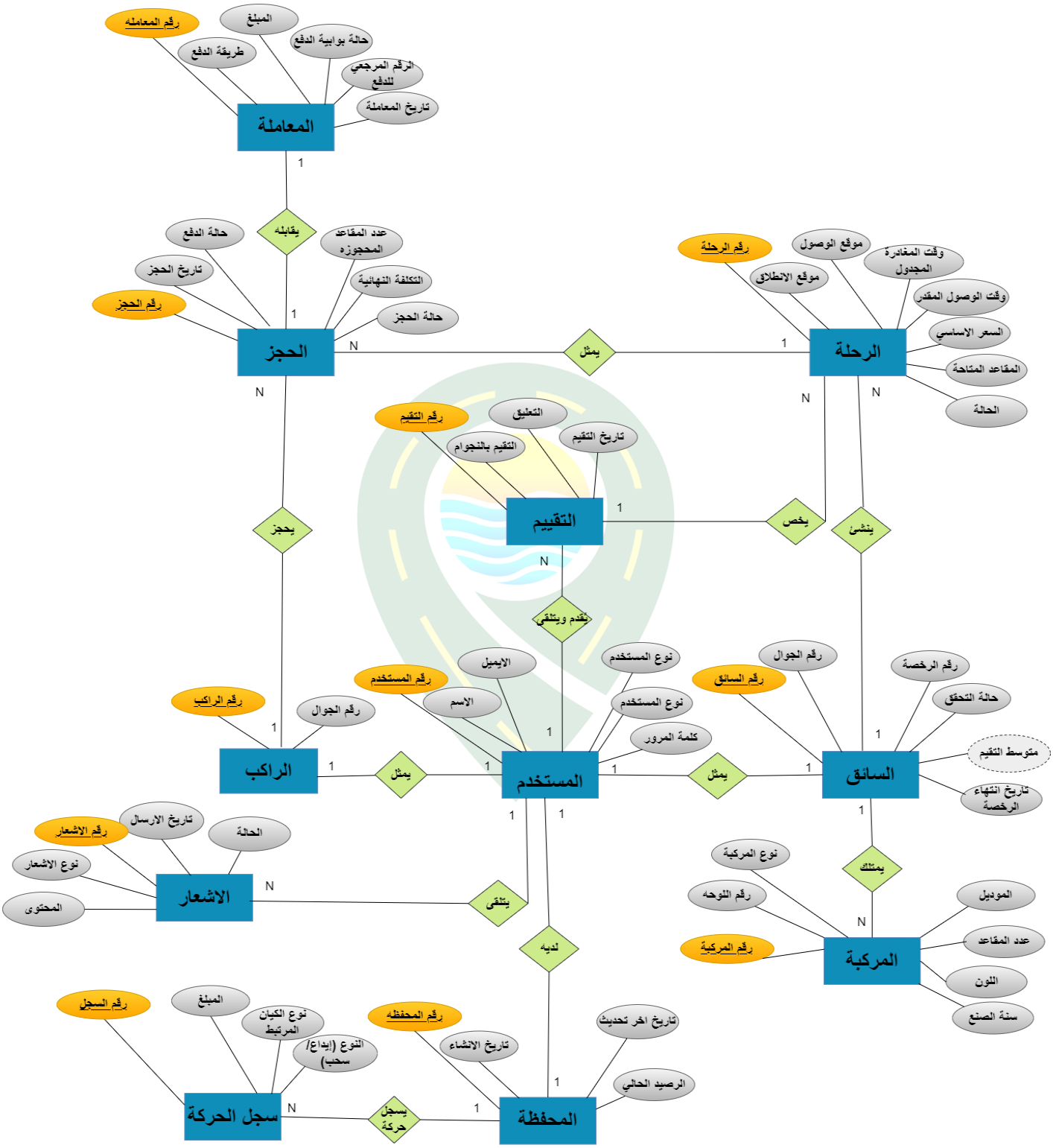
- **MESSAGES (الدردشة):** توفر قناة تواصل مباشر بين الراكب والسائق مرتبطة بحجز معين لتنسيق تفاصيل الرحلة.
- **NOTIFICATIONS (الإشعارات):** سجل للتنبيهات المرسلة للمستخدمين (تأكيد الحجز، بدء الرحلة، التحديثات).
- **RATE (التقييمات):** نظام لتقييم جودة الخدمة، يتيح للركاب تقييم الرحلة والسائق بعد الانتهاء.

ملخص العلاقات (Relationships Summary)

- **علاقة N:1 (واحد إلى متعدد):** مثل العلاقة بين المستخدم والإشعارات (مستخدم واحد لديه إشعارات كثيرة).
- **علاقة N:M (متعدد إلى متعدد):** تظهر بشكل غير مباشر بين المستخدمين والرحلات من خلال الجدول الوسيط (**Booking**).
- **علاقة الاعتمادية (Dependency):** مثل "الحجز" الذي يعتمد على وجود رحلة ومستخدم، و"الرسائل" التي تعتمد تقنياً على وجود حجز قائم.

3.6.10 نموذج الكيانات والعلاقات(Entity Relationship Model - ERM)

- يوضح الشكل (9-3) نموذج الكيان والعلاقات: حجز رحلة والدفع وتأكيد الالتقاط



شكل 10-3: نموذج الكيان والعلاقة

4 الفصل الرابع(Chapter 4)

تصميم النظام(System Design)

4.1 المقدمة(Introduction)

يُمثل هذا الفصل المرحلة التطبيقية المقترحة التي تلي تحليل المتطلبات، حيث يتم فيه تحويل المواصفات والمتطلبات التي تم تحديدها في الفصل السابق إلى هيكل تقني مُفصّل وقابل للتنفيذ. يركز التصميم على وضع الأسس المعمارية للنظام، وتحديد مكوناته الرئيسية، وواجهات التفاعل بينها، وقواعد البيانات، وواجهات المستخدم.

يهدف هذا الفصل إلى وضع مخططٍ شاملٍ ودليلٍ واضحٍ للمطورين، يضمن أن عملية البناء التقني ستسير وفق رؤية موحدة ومتكاملة. كما يحرص على أن يكون النظام مرنًا وقابلًا للتطوير والصيانة، وآمنًا بما يتناسب مع حساسية البيانات التي سيتعامل معها، مما يُسهّل عملية تحويل فكرة المشروع إلى واقع ملموس.

4.2 كيفية بناء النظام(System Overview)

تمت عملية بناء مشروع "سافر" وفق دراسة تحليلية تهدف إلى تسهيل عملية تنظيم الرحلات بين المحافظات وتسهيل حجز المقاعد للمسافرين وتوفير وسيلة تواصل فعالة بين السائق والراكب، بالإضافة إلى إدارة المحافظ المالية والتقارير المرتبطة بها.

وقد تم الاعتماد على منهجية) النموذج التكراري Iterative Model) - لما تتميز به من مرونة عالية وقدرتها على بناء النظام خطوة بخطوة، مما يتيح تجربة المكونات وتطويرها باستمرار. أما بالنسبة للتقنيات المستخدمة، فقد تم بناء النظام كالتالي:

- **الجانب الخلفي**(Backend) : تم استخدام إطار عمل Laravel (PHP)لما يتميز به من قوة في إدارة قواعد البيانات والأمان وسهولة بناء الـAPIs .
- **تطبيق الهواتف الذكية**(Mobile App) **:** تم استخدام إطار عمل Flutter (Dart)لإنتاج تطبيق يعمل على نظامي أندرويد و iOSبكفاءة عالية وواجهة مستخدم عصرية.
- **قاعدة البيانات :**تم استخدام MySQLلتخزين البيانات وضمان تكاملها.
- **التقنيات المساعدة :**تم استخدام Pusher & Laravel Echoلخدمات الإشعارات والدردشة الفورية، و Google Maps APIلتحديد المواقع والمسارات.

وفي هذا الفصل سيتم عرض عملية تنفيذ النظام، وقاموس البيانات، والمخططات الهيكلية التي توضح ترابط الأجزاء.

4.3 ربط قاعدة البيانات بالنظام

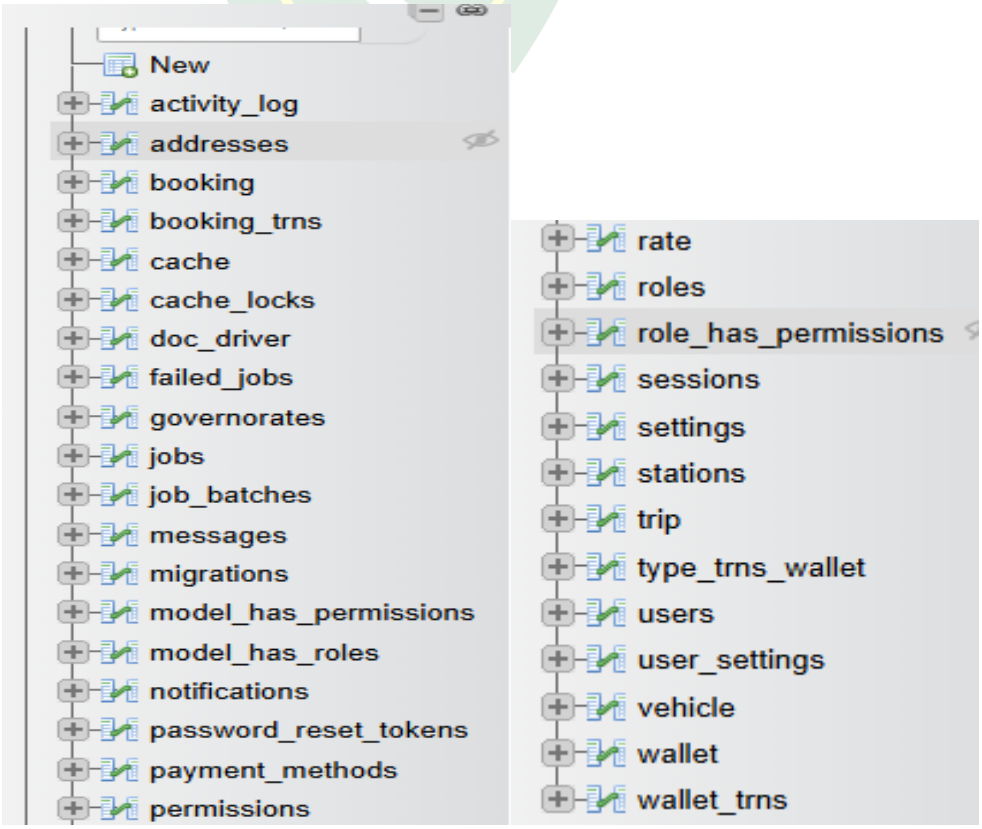
يعتمد النظام على معمارية(Client-Server) ، حيث يعمل تطبيق Flutter كواجهة للمستخدم (Client) يتواصل مع خادم Laravel عبر بروتوكول HTTP من خلال واجهات برمجة التطبيقات(RESTful APIs) . يتم الربط من خلال ملف الإعدادات (.env) في Laravel الذي يحتوي على بيانات الاتصال بقاعدة البيانات، بينما يقوم نظام Eloquent ORMبإدارة العمليات على الجداول بمرونة عالية.

4.4 بناء قواعد البيانات(Database Building)

يتلخص بناء قاعدة البيانات في مجموعة من الجداول المترابطة التي تضمن سير العمليات. فيما يلي قاموس البيانات لأهم الجداول الأساسية في النظام:

```
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=safer
DB_USERNAME=root
DB_PASSWORD=
```

```
PS E:\safer1\backend> php artisan migrate --seed
  0001_01_01_000000_create_users_table ..... 374.44ms DONE
  0001_01_01_000001_create_cache_table ..... 74.51ms DONE
  0001_01_01_000002_create_jobs_table ..... 122.15ms DONE
  2026_01_26_194219_create_payment_methods_table ..... 16.93ms DONE
  2026_02_13_000001_create_governorates_table ..... 21.71ms DONE
  2026_02_13_000003_create_stations_table ..... 87.30ms DONE
  2026_02_19_230804_create_wallet_table ..... 88.89ms DONE
  2026_02_19_230908_create_wallet_trns_table ..... 285.65ms DONE
  2026_02_20_223742_add_document_driver ..... 17.43ms DONE
  2026_02_20_224735_create_vehicle_table ..... 86.60ms DONE
  2026_02_20_225234_create_trip_table ..... 903.41ms DONE
  2026_02_20_230104_create_booking_table ..... 572.30ms DONE
  2026_02_20_232224_create_rate_table ..... 556.85ms DONE
  2026_02_28_005858_create_personal_access_tokens_table ..... 144.16ms DONE
  2026_03_14_031817_create_addresses_table ..... 84.83ms DONE
```



4.5 قاموس البيانات(Data Dictionary)

يُعد قاموس البيانات المرجع الأساسي لوصف بنية الجداول في قاعدة بيانات "سافر"، حيث يوضح أسماء الحقول، أنواعها، أحجامها، والقيود المفروضة عليها.

- في الجدول (4.1) توضيح لحقول جدول المستخدم.

جدول 4-1: جدول المستخدمين

Related Table	Required	Key	Size	Data Type	Description	Field Name
	Yes	PK	20	BigInt	المفتاح الأساسي	id
	Yes	-	255	Varchar	الاسم الكامل	name
	Yes	Unique	255	Varchar	البريد الإلكتروني	email
	Yes	-	255	Varchar	كلمة المرور المشفرة	password
	Yes	Unique	255	Varchar	رقم الهاتف	phone
	No	-	255	Varchar	رابط صورة المستخدم	img
	No	FK	20	BigInt	معرف المسؤول المؤكد	confirmed_by_id
	Yes	Index	-	Enum	نوع المستخدم (admin, driver, pass)	type
	Yes	Index	11	Int	حالة الحساب (نشط/غير نشط)	status
	Yes	-	11	Int	حالة السائق (متاح/مشغول)	status_driver
	No	-	-	Timestamp	تاريخ انتهاء الحظر	banned_until
	Yes	-	11	Int	عدد مرات الإلغاء المتتالية	consecutive_cancellations
	No	Index	-	Timestamp	تاريخ ووقت الإنشاء	created_at
	No	-	-	Timestamp	تاريخ ووقت التحديث	updated_at

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
2	name	varchar(255)	utf8mb4_unicode_ci		No	None		
3	email	varchar(255)	utf8mb4_unicode_ci		No	None		
4	email_verified_at	timestamp			Yes	NULL		
5	password	varchar(255)	utf8mb4_unicode_ci		No	None		
6	phone	varchar(255)	utf8mb4_unicode_ci		No	None		
7	img	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
8	confirmed_by_id	bigint(20)		UNSIGNED	Yes	NULL		
9	type	enum('admin', 'driver', 'pass')	utf8mb4_unicode_ci		No	None		
10	status	int(11)			No	1		
11	status_driver	int(11)			No	0		
12	banned_until	timestamp			Yes	NULL		
13	consecutive_cancellations	int(11)			No	0		
14	remember_token	varchar(100)	utf8mb4_unicode_ci		Yes	NULL		
15	deleted_at	timestamp			Yes	NULL		
16	created_at	timestamp			Yes	NULL		
17	updated_at	timestamp			Yes	NULL		

شكل 4-1: المستخدمين

- في الجدول (4.2) توضيح لحقول جدول الرحلة.

جدول 2-4: جدول الرحلات

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	Yes	PK	20	BigInt	المفتاح الأساسي للرحلة	id
Users	Yes	FK	20	BigInt	معرف السائق	driver_id
Governorates	No	FK	20	BigInt	معرف محافظة الانطلاق	from_governorate_id
Governorates	No	FK	20	BigInt	معرف محافظة الوصول	to_governorate_id
Vehicles	Yes	FK	20	BigInt	معرف المركبة المستخدمة	vehicle_id
-	Yes	-	255	Varchar	نقطة الانطلاق (نصي)	start_loc
-	Yes	-	255	Varchar	نقطة الوصول (نصي)	end_loc
-	Yes	Index	-	DateTime	وقت وتاريخ الانطلاق	start_trip_time
-	No	-	-	DateTime	وقت وتاريخ الوصول	arriavd_trip_time
-	No	-	11	Int	عدد المقاعد الإجمالي	seat_count
-	No	-	-	Double	سعر المقعد الواحد	price
-	No	-	255	Varchar	رمز الرحلة الفريد	code
-	No	-	-	LongText	بيانات المحطات الوسيطة	stations
-	No	-	-	LongText	المقاعد المحظورة أو المحجوزة	blocked_seats
-	No	-	255	Varchar	الموقع الحالي للسائق	driver_location
-	No	-	-	Text	سبب إلغاء الرحلة	reason_cancel
-	No	-	-	LongText	تصميم توزيع المقاعد	seat_layout
-	Yes	Index	-	Enum	حالة الرحلة (new, complete, cancel, ongoing)	status

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
2	driver_id	bigint(20)		UNSIGNED	No	None		
3	from_governorate_id	bigint(20)		UNSIGNED	Yes	NULL		
4	to_governorate_id	bigint(20)		UNSIGNED	Yes	NULL		
5	vehicle_id	bigint(20)		UNSIGNED	No	None		
6	start_loc	varchar(255)	utf8mb4_unicode_ci		No	None		
7	end_loc	varchar(255)	utf8mb4_unicode_ci		No	None		
8	start_trip_time	datetime			No	None		
9	arriavd_trip_time	datetime			No	None		
10	seat_count	int(11)			Yes	NULL		
11	price	double			Yes	NULL		
12	code	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
13	stations	longtext	utf8mb4_bin		Yes	NULL		
14	blocked_seats	longtext	utf8mb4_bin		Yes	NULL		
15	driver_location	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
16	reason_cancel	text	utf8mb4_unicode_ci		Yes	NULL		
17	seat_layout	longtext	utf8mb4_unicode_ci		Yes	NULL		
18	status	enum('new', 'complete', 'cancel', 'ongoing')	utf8mb4_unicode_ci		No	None		

شكل 2-4: الرحلات

- في الجدول (4.3) توضيح لحقول جدول الحجوزات.

جدول 3-4: الحجوزات

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	Yes	PK	20	BigInt	المفتاح الأساسي للحجز	id
Trips	Yes	FK	20	BigInt	معرف الرحلة المرتبطة	trip_id
Users	Yes	FK	20	BigInt	معرف الراكب	pass_id
-	Yes	-	-	Date	تاريخ الحجز	date
-	Yes	-	11	Int	عدد المقاعد المحجوزة	seat_count
-	Yes	-	-	Double	التكلفة النهائية	final_cost
-	No	-	255	Varchar	كود الحجز (التذكرة)	code
-	No	-	-	LongText	تفاصيل المقاعد المحجوزة	seats
-	No	-	-	Text	سبب إلغاء الحجز	reason_cancel
-	Yes	-	-	Enum	نوع الدفع (cash, in_app, wallet)	type_payment
-	Yes	Index	-	Enum	حالة الحجز (new, cancel, confirm...)	status
-	Yes	-	-	Enum	حالة الدفع (paided, not_paided...)	paid_status
-	No	-	-	Timestamp	تاريخ الحذف (Soft Delete)	deleted_at
-	No	Index	-	Timestamp	تاريخ الإنشاء	created_at
-	No	-	-	Timestamp	تاريخ التحديث	updated_at
-	No	-	255	Varchar	عنوان الركوب التفصيلي	pickup_address
-	No	-	255	Varchar	خط العرض لموقع الركوب	pickup_lat
-	No	-	255	Varchar	خط الطول لموقع الركوب	pickup_lng
-	No	-	255	Varchar	عنوان النزول التفصيلي	dropoff_address
-	No	-	255	Varchar	خط العرض لموقع النزول	dropoff_lat
-	No	-	255	Varchar	خط الطول لموقع النزول	dropoff_lng
-	No	-	8,2	Decimal	مسافة الركوب الإضافية	extra_pickup_distance
-	No	-	8,2	Decimal	مسافة النزول الإضافية	extra_dropoff_distance
-	No	PK	10,2	Decimal	سعر المسافة الإضافية للركوب	extra_pickup_price
				Decimal	سعر المسافة الإضافية للنزول	extra_dropoff_price

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
2	trip_id	bigint(20)		UNSIGNED	No	None		
3	pass_id	bigint(20)		UNSIGNED	No	None		
4	date	date			No	None		
5	seat_count	int(11)			No	None		
6	final_cost	double			No	None		
7	code	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
8	seats	longtext	utf8mb4_bin		Yes	NULL		
9	reason_cancel	text	utf8mb4_unicode_ci		Yes	NULL		
10	type_payment	enum('cash', 'in_app', 'wallet')	utf8mb4_unicode_ci		No	None		
11	status	enum('new', 'cancel', 'confirm', 'pending', 'compl...	utf8mb4_unicode_ci		No	None		
12	paid_status	enum('paided', 'not_paided', 'refunded')	utf8mb4_unicode_ci		No	None		
13	deleted_at	timestamp			Yes	NULL		
14	created_at	timestamp			Yes	NULL		
15	updated_at	timestamp			Yes	NULL		
16	pickup_address	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
17	pickup_lat	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
18	pickup_lng	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
19	dropoff_address	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
20	dropoff_lat	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
21	dropoff_lng	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
22	extra_pickup_distance	decimal(8,2)			Yes	NULL		
23	extra_dropoff_distance	decimal(8,2)			Yes	NULL		
24	extra_pickup_price	decimal(10,2)			Yes	NULL		
25	extra_dropoff_price	decimal(10,2)			Yes	NULL		

شكل 3-4: الحجوزات

ملاحظة مهمة لهذا الجدول:

- **الحقول الجغرافية :**الحقول من (16 إلى 21) مخصصة لتخزين إحداثيات GPS لضمان دقة مكان الركوب والنزول.
- **التسعير الديناميكي :**الحقول من (22 إلى 25) تُستخدم في حال طلب الراكب التوصيل من مكان بعيد عن "الفرزة" أو المحطة الرئيسية، حيث يحسب النظام تكلفة إضافية بناءً على المسافة.
- **Enums**: لاحظ أن حقول type_paymentو statusو paid_statusلا تقبل إلا قيماً محددة لضمان أمان العمليات المالية.

- في الجدول (4.4) توضيح لحقول جدول المحفظة.

جدول 4-4: جدول المحفظة

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	Yes	PK	20	BigInt	المفتاح الأساسي للمحفظة	id
Users	Yes	FK	20	BigInt	معرف صاحب المحفظة	user_id
-	Yes	-	12,2	Decimal	الرصيد المالي الحالي	balance
-	No	-	-	Timestamp	تاريخ الإنشاء	created_at
-	No	-	-	Timestamp	تاريخ التحديث	updated_at

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
2	user_id	bigint(20)		UNSIGNED	No	None		
3	amount	double			No	0		
4	deleted_at	timestamp			Yes	NULL		
5	created_at	timestamp			Yes	NULL		
6	updated_at	timestamp			Yes	NULL		

شكل 4-4: المحفظة

- في الجدول (4.5) توضيح لحقول جدول حركات المحفظة.

جدول 4-5:حركات المحفظة

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	Yes	Yes	20	BigInt	المفتاح الأساسي للعملية	id
Wallet	Yes	Yes	20	BigInt	معرف المحفظة المرتبطة	wallet_id
Type_trns_wallet	Yes	Yes	20	BigInt	معرف نوع الحركة (شحن، دفع...)	type_trns_wallet_id
-	No	Yes	-	Enum	اتجاه الحركة (credit, debit)	type
-	No	Yes	-	Double	مبلغ العملية المالي	amount
Payment_methods		No	20	BigInt	معرف طريقة الشحن المستخدمة	topup_method_id
-		No	255	Varchar	رابط صورة سند الشحن	topup_proof_img
-		No	255	Varchar	رقم حساب المستخدم المحول منه	user_account_number
-		Yes	-	Enum	حالة العملية (مقبولة، مرفوضة...)	status
-		No	-	Timestamp	تاريخ الحذف (Soft Delete)	deleted_at
-		No	-	Timestamp	تاريخ الإنشاء والوقت	created_at
-		No	-	Timestamp	تاريخ التحديث والوقت	updated_at

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
2	wallet_id	bigint(20)		UNSIGNED	No	None		
3	type_trns_wallet_id	bigint(20)		UNSIGNED	No	None		
4	type	enum('credit', 'debit')	utf8mb4_unicode_ci		No	None		
5	amount	double			No	None		
6	topup_method_id	bigint(20)		UNSIGNED	Yes	NULL		
7	topup_proof_img	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
8	user_account_number	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
9	status	enum('pending', 'approved', 'rejected')	utf8mb4_unicode_ci		No	approved		
10	deleted_at	timestamp			Yes	NULL		
11	created_at	timestamp			Yes	NULL		
12	updated_at	timestamp			Yes	NULL		

شكل 4-5: حركات المحفظة

- في الجدول (4.6) توضيح لحقول جدول أنواع حركات المحفظة(Type_trns_wallet)

هذا الجدول هو "القاموس" الذي يحدد ماذا تعني كل حركة (مثلاً: شحن رصيد، دفع قيمة رحلة، استرجاع مبلغ).

جدول 4-6: أنواع حركات المحفظة

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	Yes	PK	20	BigInt	المفتاح الأساسي للنوع	id
-	Yes	-	255	Varchar	اسم نوع الحركة (مثلاً: شحن رصيد)	name
-	No	-	-	Timestamp	تاريخ الحذف(Soft Delete)	deleted_at
-	No	-	-	Timestamp	تاريخ الإنشاء	created_at
-	No	-	-	Timestamp	تاريخ التحديث	updated_at

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
2	name	varchar(255)	utf8mb4_unicode_ci		No	None		
3	deleted_at	timestamp			Yes	NULL		
4	created_at	timestamp			Yes	NULL		
5	updated_at	timestamp			Yes	NULL		

شكل 4-6 : أنواع حركات المحفظة

لماذا تم إنشاء هذا الجدول؟ (الأسباب التقنية):

وجود هذا الجدول في مشروعك يعكس **احترافية في التصميم** (Normalization)، وهذه هي الأسباب الأربعة الرئيسية التي يمكنك ذكرها في بحثك أو عند المناقشة:

1. **تصنيف البيانات(Categorization) :** بدلاً من كتابة كلمة "شحن" أو "دفع" يدوياً في كل حركة مالية، نستخدم معرفاً رقمياً (ID) يشير إلى هذا الجدول. هذا يمنع الأخطاء الإملائية ويجعل البيانات منظمة جداً.
2. **المرونة والقابلية للتوسع(Extensibility) :** إذا قررت مستقبلاً إضافة نوع جديد من الحركات (مثل: "مكافأة" أو "غرامة إلغاء")، فلن تحتاج لتغيير الكود البرمجي أو هيكل قاعدة البيانات؛ كل ما عليك فعله هو إضافة سطر جديد في هذا الجدول فقط.

- في الجدول (4.7) توضيح لحقول جدول طرق الدفع والشحن**(Payment_methods)**

الجدول الذي تظهر بياناته للمستخدم عندما يريد شحن محفظته (مثل: كريمي، فلوسك، جيب).

جدول 4-7: طرق الدفع والشحن

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	Yes	PK	20	BigInt	المفتاح الأساسي	id
-	Yes	-	255	Varchar	اسم الوسيلة (مثلاً: بنك الكريمي)	name
-	No	-	255	Varchar	رقم الحساب الخاص بالشركة	account_number
-	No	-	255	Varchar	أيقونة أو شعار الوسيلة	image
-	Yes	-	-	Boolean	هل الوسيلة مفعلة حالياً؟	is_active

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
2	name	varchar(255)	utf8mb4_unicode_ci		No	None		
3	image	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
4	account_number	varchar(255)	utf8mb4_unicode_ci		No	None		
5	is_active	tinyint(1)			No	1		
6	created_at	timestamp			Yes	NULL		
7	updated_at	timestamp			Yes	NULL		
8	deleted_at	timestamp			Yes	NULL		

شكل 7-4: طرف الدفع والشحن

- في الجدول (4.8) توضيح لحقول جدول الربط بين الحجز والمالية(Booking_trns)

هذا الجدول مهم جداً لأنه "الجسر" الذي يربط عملية الحجز بعملية الخصم من المحفظة.

جدول 4-8 :معاملات الحجوزات

Related Table	Required	Key	Size	Data Type	Description	Field Name
20	PK	Yes	-	BigInt	المفتاح الأساسي للسند	id
20	FK	Yes	**Bookings**	BigInt	معرف الحجز المرتبط بالسند	booking_id
20	FK	No	**Wallet_trns**	BigInt	معرف حركة المحفظة (إن وُجد)	wallet_trns_id
-	-	Yes	-	Timestamp	تاريخ العملية وتوقيتها التلقائي	date
255	-	No	-	Varchar	نوع العملية (مثلاً: خصم، استرجاع)	type
255	-	No	-	Varchar	رقم المرجع (من بوابة الدفع)	payment_ref_num
-	-	Yes	-	Enum	حالة البوابة(success, failure)	gateway_stats
-	-	No	-	Timestamp	تاريخ الحذف(Soft Delete)	deleted_at
-	-	No	-	Timestamp	تاريخ الإنشاء	created_at
-	-	No	-	Timestamp	تاريخ التحديث	updated_at

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
2	booking_id	bigint(20)		UNSIGNED	No	None		
3	wallet_trns_id	bigint(20)		UNSIGNED	Yes	NULL		
4	date	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()
5	type	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
6	payment_ref_num	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
7	gateway_stats	enum('success', 'failure')	utf8mb4_unicode_ci		No	None		
8	deleted_at	timestamp			Yes	NULL		
9	created_at	timestamp			Yes	NULL		
10	updated_at	timestamp			Yes	NULL		

شكل 8-4:معاملات الحجوزات

توضيح أهمية هذا الجدول

يعمل هذا الجدول كـ **"مدقق مالي "**للربط بين العمليات الإدارية (الحجز) والعمليات المالية (المحفظة)، وفوائده هي:

1. **التوثيق المزدوج :**يسمح لك بمعرفة أي "حركة محفظة" بالضبط تم خصمها مقابل هذا "الحجز" عبر حقلwallet_trns_id.
2. **أتمتة الحالة :**حقل gateway_statsيخبر النظام تلقائياً: "هل العملية المالية نجحت؟"، فإذا كانت successيتم تأكيد الحجز، وإذا كانت failureيتم إلغاء الحجز أو إبقاؤه معلقاً.
3. **الشفافية المالية :**عبر حقل payment_ref_num(رقم المرجع)، يمكن للشركة المطابقة مع التقارير البنكية أو بوابات الدفع الخارجية للتأكد من المبالغ المستلمة.

ملخص تعريفي موجز لكل جدول من الجداول المتعلقة بالجانب المالي والمحفظة في مشروعك، مصاغ بأسلوب احترافي يناسب البحث التقني:

3. جدول المحفظة(Wallet) :

يُعد المستودع الأساسي لرصيد المستخدم؛ حيث يقوم بتخزين القيمة المالية الحالية المتاحة لكل مستخدم (سواء كان راكباً أو سائقاً)، ويرتبط بالمستخدم بعلاقة (واحد إلى واحد) لضمان دقة الرصيد.

2. جدول حركات المحفظة(Wallet_trns) :

هو "السجل المالي" التفصيلي للنظام؛ حيث يسجل كل حركة دخول أو خروج للأموال (إيداع، سحب، خصم قيمة حجز). كما يحتوي على حقول للتحقق من عمليات الشحن اليدوي عبر رفع صور السندات ومراجعتها من قبل الإدارة.

2. جدول أنواع حركات المحفظة(Type_trns_wallet) :

جدول تعريفي (Lookup Table) يُستخدم لتصنيف العمليات المالية داخل النظام. وظيفته الأساسية هي تمييز أنواع الحركات (مثلاً: "شحن رصيد"، "دفع رحلة"، "استرداد مبلغ")، مما يسهل عملية البرمجة واستخراج التقارير المالية التفصيلية.

3. جدول سندات الحجز(Booking_trns) :

يعمل كجسر ربط تقني ومالي بين "عملية الحجز" و "حركة المحفظة". وظيفته هي توثيق نجاح أو فشل عملية الدفع عبر بوابات الدفع الإلكترونية، وربط رقم مرجع العملية الخارجية بسجل الحجز الداخلي لضمان الشفافية والمطابقة.

4. جدول طرق الدفع(Payment_methods) :

يخزن قائمة الوسائل المالية المتاحة في النظام (مثل: بنك الكريمي، محفظة جوال، حوالة). يُستخدم لعرض الخيارات المتاحة للمستخدم عند رغبته في شحن رصيده، ويحتوي على بيانات الحسابات الخاصة بالشركة لضمان وصول التحويلات بشكل صحيح.

- في الجدول (4.9) توضيح لحقول جدول المركبات(Vehicle)

جدول 4-9: المركبات

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	PK	PK	20	BigInt	المفتاح الأساسي للمركبة	id
Users	FK	-	20	BigInt	معرف السائق المرتبط بالمركبة	driver_id
-	-	-	255	Varchar	رقم لوحة السيارة	plate_num
-	-	-	255	Varchar	نوع المركبة (باص، سيارة صغيره..)	type
-	-	-	11	Int	موديل المركبة (مثلاً: 2024)	model
-	-		11	Int	عدد مقاعد المركبة	seat_count
-	-		255	Varchar	لون المركبة	color
-	-		255	Varchar	رابط صورة المركبة	img
-	-		-	Text	تصميم توزيع المقاعد	seat_layout
-	No	-	-	Timestamp	تاريخ الحذف(Soft Delete)	deleted_at
-	No	-	-		تاريخ الإضافة	created_at
-	No	-	-		تاريخ التحديث	updated_at

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
2	driver_id	bigint(20)		UNSIGNED	Yes	NULL		
3	plate_num	varchar(255)	utf8mb4_unicode_ci		No	None		
4	type	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
5	model	int(11)			Yes	NULL		
6	seat_count	int(11)			Yes	NULL		
7	color	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
8	img	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
9	seat_layout	text	utf8mb4_unicode_ci		Yes	NULL		
10	deleted_at	timestamp			Yes	NULL		
11	created_at	timestamp			Yes	NULL		
12	updated_at	timestamp			Yes	NULL		

شكل 4-9: المركبات

- في الجدول (4.10) توضيح لحقول جدول المحافظات(Governorates)

جدول 4-10: المحافظات

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	Yes	PK	20	BigInt	المفتاح الأساسي للمحافظة	**id**
-	Yes	-	255	Varchar	اسم المحافظة بالعربي	name
-	No	-	255	Varchar	اسم المحافظة بالإنجليزي	name_en
-	No	-	255	Varchar	رابط صورة تعبيرية للمحافظة	image
-	Yes	-	1	TinyInt	هل المحافظة نشطة؟(0/1)	is_active
-	Yes	-	11	Int	ترتيب الظهور في القائمة	sort_order
-	No	-	-	Timestamp	تاريخ الإضافة والوقت	created_at
-	No	-	-	Timestamp	تاريخ التحديث والوقت	updated_at
-	No	-	-	Timestamp	تاريخ الحذف(Soft Delete)	deleted_at

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
☐	1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
☐	2	name	varchar(255)	utf8mb4_unicode_ci		No	None		
☐	3	name_en	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
☐	4	image	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
☐	5	is_active	tinyint(1)			No	1		
☐	6	sort_order	int(11)			No	0		
☐	7	created_at	timestamp			Yes	NULL		
☐	8	updated_at	timestamp			Yes	NULL		
☐	9	deleted_at	timestamp			Yes	NULL		

شكل 10-4: المحافظات

توضيح بعض الحقول:

1. **دعم اللغات :**وجود حقل name_enبجانب nameيعني أن النظام يدعم الواجهات المتعددة اللغات (العربية والإنجليزية).
2. **التفعيل والتعطيل :**حقل is_activeيسمح لمدير النظام بإخفاء محافظة معينة من قائمة الوجهات مؤقتاً دون حذفها.
3. **ترتيب العرض :**حقل sort_orderيُستخدم لترتيب المحافظات في القائمة المنسدلة داخل التطبيق (مثلاً لجعل المحافظات الأكثر طلباً تظهر في الأعلى).

- في الجدول (4.11) توضيح لحقول جدول المحطات/الفرزات(Stations)

جدول 11-4 :الفرزات

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	Yes	PK	20	BigInt	المفتاح الأساسي للمحطة	**id**
Governorates	Yes	FK	20	BigInt	معرف المحافظة التابعة لها	governorate_id
-	Yes	-	255	Varchar	اسم المحطة (الفرزة)	name
-	No	-	255	Varchar	خط العرض الجغرافي	location_lat
-	No	-	255	Varchar	خط الطول الجغرافي	location_long
-	Yes	-	1	TinyInt	هل المحطة مفعلة؟(0/1)	is_active
-	No	-	-	Timestamp	تاريخ الإضافة	created_at
-	No	-	-	Timestamp	تاريخ التحديث	updated_at
-	No	-	-	Timestamp	تاريخ الحذف(Soft Delete)	deleted_at

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
2	governorate_id	bigint(20)		UNSIGNED	No	None		
3	name	varchar(255)	utf8mb4_unicode_ci		No	None		
4	location_lat	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
5	location_long	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
6	is_active	tinyint(1)			No	1		
7	created_at	timestamp			Yes	NULL		
8	updated_at	timestamp			Yes	NULL		
9	deleted_at	timestamp			Yes	NULL		

شكل 4-11: الفرزات

- في الجدول (4.12) توضيح لحقول جدول المراسلات(Messages)

جدول 4-12: الرسائل

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	Yes	PK	20	BigInt	المفتاح الأساسي للرسالة	**id**
Bookings	No	FK	20	BigInt	معرف الحجز المرتبط (فارغ في رسائل الدعم)	booking_id
Users	Yes	FK	20	BigInt	معرف مرسل الرسالة	sender_id
Users	Yes	FK	20	BigInt	معرف مستقبل الرسالة	receiver_id
-	Yes	-	-	Text	نص الرسالة الكامل	message
-	Yes	Index	1	TinyInt	حالة قراءة الرسالة (0: غير مقروءة، 1: مقروءة)	is_read
-	No	-	-	Timestamp	تاريخ الحذف(Soft Delete)	deleted_at
-	No	-	-	Timestamp	تاريخ الإرسال	created_at
-	No	-	-	Timestamp	تاريخ التحديث	updated_at

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
2	booking_id	bigint(20)		UNSIGNED	Yes	NULL		
3	sender_id	bigint(20)		UNSIGNED	No	None		
4	receiver_id	bigint(20)		UNSIGNED	No	None		
5	message	text	utf8mb4_unicode_ci		No	None		
6	is_read	tinyint(1)			No	0		
7	deleted_at	timestamp			Yes	NULL		
8	created_at	timestamp			Yes	NULL		
9	updated_at	timestamp			Yes	NULL		

شكل 4-12: الرسائل

لقد تم تصميم جدول الرسائل بمرونة عالية، حيث تمكن التطبيق من استخدام نفس الجدول لخدمة غرضين مختلفين (الدردشة بين السائق والراكب، ورسائل الدعم الفني)، وذلك من خلال المنطق البرمجي التالي:

1. الدردشة بين السائق والراكب(Contextual Chat) :

- يتم الربط هنا من خلال حقل**booking_id**.
- عندما تكون الرسالة مرتبطة برقم حجز، يفهم النظام أنها "دردشة رحلة" بين سائق وراكب بخصوص موعد أو موقع معين.

2. رسائل الدعم الفني(Support Messaging) :

- فعندما يرسل المستخدم رسالة دعم، يترك النظام حقل booking_idفارغاً(Null) .
- يكون receiver_id(المستقبل) هو معرف أحد المسؤولين(Admin) .
- بهذه الطريقة، يتم فرز الرسائل برمجياً: "أي رسالة بدون booking_idوموجهة لـ Admin تُعتبر تذكرة دعم فني"(Support Ticket) ؟
  - في الجدول (4.13) توضيح لحقول جدول عناوين المستخدمين(Addresses)

جدول 13-4:العناوين

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	Yes	PK	20	BigInt	المفتاح الأساسي للعنوان	id
Users	Yes	FK	20	BigInt	معرف المستخدم صاحب العنوان	user_id
-	No	-	255	Varchar	اسم العنوان (منزل، عمل، إلخ)	name
-	No	-	-	Text	وصف العنوان الكامل نصياً	full_address
-	No	-	-	Double	خط العرض الجغرافي(Latitude)	latitude
-	No	-	-	Double	خط الطول الجغرافي(Longitude)	longitude
-	Yes	-	1	TinyInt	هل هو العنوان الافتراضي؟(0/1)	is_default
-	No	-	-	Timestamp	تاريخ الإضافة	created_at
-	No	-	-	Timestamp	تاريخ التحديث	updated_at
-	No	-	-	Timestamp	تاريخ الحذف(Soft Delete)	deleted_at

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
☐	1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
☐	2	user_id	bigint(20)		UNSIGNED	No	None		
☐	3	name	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
☐	4	full_address	text	utf8mb4_unicode_ci		Yes	NULL		
☐	5	latitude	double			Yes	NULL		
☐	6	longitude	double			Yes	NULL		
☐	7	is_default	tinyint(1)			No	0		
☐	8	created_at	timestamp			Yes	NULL		
☐	9	updated_at	timestamp			Yes	NULL		
☐	10	deleted_at	timestamp			Yes	NULL		

شكل 13-4: العناوين

كيف يخدم هذا الجدول

1. **سرعة الحجز(Quick Booking) :** بدلاً من أن يقوم الراكب بالبحث عن موقعه على الخريطة في كل مرة، يتيح له النظام اختيار عنوان محفوظ مسبقاً (مثل "المنزل")، مما يجعل عملية الحجز تتم بضغطة زر واحدة.
2. **دقة الحساب المالي(Price Calculation) :** يستخدم النظام حقول latitude وlongitudeالمخزنة هنا لمقارنتها بموقع "الفرزة". وبناءً على هذه الإحداثيات، يحسب النظام المسافة الإضافية بدقة ويضيفها إلى سعر الرحلة (خدمة التوصيل من الباب).
3. **العناوين الافتراضية:** حقل is_defaultيسمح للتطبيق بفتح الخريطة تلقائياً على عنوان المستخدم المفضل بمجرد فتح شاشة الحجز، مما يحسن تجربة المستخدم.(UX)
4. **تُستخدم حقول latitude وlongitude**لتفعيل ميزة "تحديد موقعي الآن "داخل التطبيق من أجل تحديد العنوان.

- في الجدول (4.14) توضيح لحقول جدول الإشعارات(Notifications)

جدول 14-4: الاشعارات

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	Yes	PK	36	Char	المفتاح الأساسي(معرف فريد) UUID	**id**
-	Yes	-	255	Varchar	نوع الإشعار البرمجي	type
Users	Yes	Index	255	Varchar	نوع الكيان المستلم(Polymorphic)	notifiable_type
Users	Yes	Index	20	BigInt	معرف المستخدم المستلم للإشعار	notifiable_id
-	Yes	-	-	Text	بيانات الإشعار (مخزنة بصيغة) JSON	data
-	No	Index	-	Timestamp	تاريخ ووقت قراءة الإشعار	read_at
-	No	-	-	Timestamp	تاريخ الإرسال	created_at
-	No	-	-	Timestamp	تاريخ التحديث	updated_at
-	No	-	-	Timestamp	تاريخ الحذف(Soft Delete)	deleted_at

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
☐	1	**id**	char(36)	utf8mb4_unicode_ci		No	*None*		
☐	2	**type**	varchar(255)	utf8mb4_unicode_ci		No	*None*		
☐	3	**notifiable_type**	varchar(255)	utf8mb4_unicode_ci		No	*None*		
☐	4	**notifiable_id**	bigint(20)		UNSIGNED	No	*None*		
☐	5	**data**	text	utf8mb4_unicode_ci		No	*None*		
☐	6	**read_at**	**timestamp**			Yes	*NULL*		
☐	7	**created_at**	timestamp			Yes	*NULL*		
☐	8	**updated_at**	timestamp			Yes	*NULL*		
☐	9	**deleted_at**	timestamp			Yes	*NULL*		

شكل 14-4 الاشعارات

توضيح تقني لمميزات هذا الجدول:

1. **استخدام الـــUUID :** تلاحظ أن الحقل id هو من نوع Char (36) وليس BigInt. هذا يعني أن النظام يستخدم معرفات فريدة عالمياً (UUID) بدلاً من الأرقام التسلسلية، وهي ميزة أمان قوية تمنع التنبؤ بأرقام الإشعارات وتزيد من خصوصية البيانات.
2. **الـــعـــلاقـــة الـــمـــتـــعـــددة(Polymorphic Relationship) :** استـــخـــدام حقول notifiable_type وnotifiable_id يعني أن النظام مرن جداً؛ حيث يمكنه إرسال إشعارات لأي نوع من المستخدمين (ركاب، سائقين، أو حتى مدراء) باستخدام نفس الجدول وبكفاءة عالية.
3. **تخزين البيانات المرن(JSON Data) :** حقل data يقوم بتخزين محتوى الإشـــعار (العنوان، النص، ورابط الانتقال) في حقل واحد بصيغة نصية، مما يسمح بإرسال أنواع مختلفة من التنبيهات دون الحاجة لتغيير هيكل الجدول في كل مرة.
4. **تتبع القراءة: حقل read_at** يسمح للتطبيق بإظهار "نقطة حمراء" على الإشعارات الجديدة فقط، ويقوم بتغيير الحالة بمجرد فتح المستخدم للإشعار.
5. **حقل :morphs('notifiable')**

هذا السطر البرمجي الواحد يقوم بإنشاء حقلين في قاعدة البيانات:

- ✓ **notifiable_id**: وهو معرف المستخدم (User ID) الذي سيتلقى الإشعار.
- ✓ **notifiable_type**: وهو نوع الكيان (مثلاً: App\Models\User).
- ✓ **الفائدة** :يُسمى هذا الربط(Polymorphic Relation) ، وهو يسمح للنظام بإرسال إشعارات لأي نوع من الجداول مستقبلاً (ركاب، سائقين، مدراء) باستخدام نفس الجدول.

6. **حقل type:** هـذا الحقل يخزن "اســم الصـــنف (Class Name) "الخـاص بـالإشـــعـار برمجيـاً (مثلاًBookingNotification :). فائدته هي إخبار تطبيق Flutter بكيفية عرض هذا الإشـــعار (هل يفتح شاشة الدردشة أم شاشة الرحلات؟).

- في الجدول (4.15) توضيح لحقول جدول الوثائق (doc_driver)

جدول 15-4:وثائق السائق

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	Yes	PK	20	BigInt	المفتاح الأساسي للوثيقة	**id**
Users	No	FK	20	BigInt	معرف المستخدم (السائق) صاحب الوثيقة	user_id
Vehicle	No	FK	20	BigInt	معرف المركبة المرتبطة بالوثيقة (إن وُجدت)	vehicle_id
-	Yes	-	255	Varchar	نوع الوثيقة (رخصة، بطاقة شخصية، ملكية)	type
-	Yes	-	255	Varchar	مسار الصورة المخزنة على الخادم	path
-	No	-	255	Varchar	رقم الوثيقة (مثلاً رقم الرخصة)	lic_num
-	No	-	-	Date	تاريخ انتهاء صلاحية الوثيقة	exp_date_lic
-	No	-	-	Timestamp	تاريخ الحذف(Soft Delete)	deleted_at

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
☐	1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
☐	2	user_id	bigint(20)		UNSIGNED	Yes	NULL		
☐	3	vehicle_id	bigint(20)		UNSIGNED	Yes	NULL		
☐	4	type	varchar(255)	utf8mb4_unicode_ci		No	None		
☐	5	path	varchar(255)	utf8mb4_unicode_ci		No	None		
☐	6	lic_num	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
☐	7	exp_date_lic	date			Yes	NULL		
☐	8	deleted_at	timestamp			Yes	NULL		

شكل 15-4:وثائق السائق

الأهمية الوظيفية لهذا الجدول:

1. **نظام التحقق(Verification System) :** لا يتم تفعيل حساب السائق في التطبيق إلا بعد قيامه برفع وثائقه في هذا الجدول. يقوم الآدمن بمراجعة هذه الصور (عبر حقل path)ومطابقة البيانات (lic_num) قبل الموافقة على السائق.
2. **الأمان والموثوقية :**تخزين تاريخ الانتهاء (exp_date_lic) يسمح للنظام بإرسال تنبيهات تلقائية للسائق قبل انتهاء رخصته، أو حتى إيقاف حسابه مؤقتاً إذا انتهت صلاحية وثائقه لضمان قانونية الرحلات.
3. **المرونة في الربط :**لاحظ أن الجدول يرتبط بـ المستخدم (لتوثيق هويته الشخصية ورخصته) وبـ المركبة (لتوثيق ملكية السيارة وتأمينها)، مما يجعله جدولاً مركزياً لإدارة الملفات القانونية للمشروع.
4. **تخزين الملفات :**حقل pathلا يخزن الصورة نفسها، بل يخزن "رابطها" فقط، مما يحافظ على سرعة وأداء قاعدة البيانات.

- في الجدول (4.16) توضيح لحقول جدول التقييمات(Rates)

جدول 16-4:التقيمات

Related Table	Required	Key	Size	Data Type	Description	Field Name
-	Yes	PK	20	BigInt	المفتاح الأساسي للتقييم	id
Trips	Yes	FK	20	BigInt	معرف الرحلة التي تم تقييمها	trip_id
Users	Yes	FK	20	BigInt	الشخص الذي تم تقييمه(Rated)	rated_id
Users	Yes	-	20	BigInt	الشخص الذي قام بالتقييم(Rater)	rater_id
-	Yes	-	-	Double	درجة التقييم (من 1 إلى 5)	rating
-	No	-	255	Varchar	تعليق أو ملاحظة مكتوبة	comment
-	No	-	-	Timestamp	تاريخ الحذف(Soft Delete)	deleted_at
-	No	-	-	Timestamp	تاريخ الإنشاء	created_at
-	No	-	-	Timestamp	تاريخ التحديث	updated_at

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
☐	1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT
☐	2	trip_id	bigint(20)		UNSIGNED	No	None		
☐	3	rated_id	bigint(20)		UNSIGNED	No	None		
☐	4	rater_id	bigint(20)		UNSIGNED	No	None		
☐	5	rating	double			No	5		
☐	6	comment	varchar(255)	utf8mb4_unicode_ci		Yes	NULL		
☐	7	deleted_at	timestamp			Yes	NULL		
☐	8	created_at	timestamp			Yes	NULL		
☐	9	updated_at	timestamp			Yes	NULL		

شكل 16-4:التقيمات

توضيح ميزات جدول التقييمات:

1. **نظام التقييم المتبادل(Bidirectional Rating) :** استخدام حقول rated_id وrater_idيوضح أن النظام يسمح للراكب بتقييم السائق، **وأيضاً** يسمح للسائق بتقييم الراكب. هذا يضمن جودة التعامل من الطرفين.
2. **الارتباط بالرحلة(Contextual Rating) :** ارتباط التقييم بـ trip_idيمنع المستخدمين من تقييم بعضهم بشكل عشوائي؛ حيث لا يمكن وضع تقييم إلا إذا كانت هناك رحلة فعلية تمت بينهما في النظام.
3. **حساب التقييم المتوسط :**حقل ratingمن نوع Doubleيسمح بحساب المتوسطات الحسابية بدقة (مثلاً 4.7 من 5)، ويتم عرض هذا المتوسط في الملف الشخصي للسائق لزيادة ثقة الركاب.
4. **التغذية الراجعة(Feedback) :** حقل commentيعطي الإدارة فكرة عن جودة الخدمة؛ ففي حال وجود تقييم منخفض مع تعليق سلبي، يمكن للآدمن التدخل لمراجعة سلوك السائق أو الراكب.

5 الفصل الخامس(Chapter 5)

التنفيذ والاختبار

(Implementation and Test)

5.1 مقدمة(Introduction)

يتناول هذا الفصل المرحلة العملية لتحويل التصاميم والمخططات السابقة إلى كود برمجي ونظام واقعي. كما يستعرض البيئة البرمجية والمعدات المستخدمة في التطوير، بالإضافة إلى استراتيجية الاختبار المتبعة لضمان خلو النظام من الأخطاء.

5.2 بيئة التنفيذ(Implementation Environment)

- **المعدات(Hardware):** تم التطوير على أجهزة حاسوب بمواصفات عالية لمعالجة بيئة الـ Android Emulator.
- **البرمجيات(Software) :**
  - **VS Code / Android Studio:** لكتابة كود Flutter .
  - **PHPStorm / VS Code:** لتطوير الـ Backend باستخدام Laravel .
  - **Postman:** لاختبار وتوثيق الـ APIs قبل ربطها بالتطبيق.
  - **XAMPP / MySQL:** لإدارة قاعدة البيانات محلياً.

5.3 خطوات التنفيذ(Implementation Steps)

1. **بناء القاعدة(Backend) :** تم إعداد خادم Laravel ، إنشاء الجداول(Migrations) ، وبرمجة الـ Controllers لإدارة الرحلات والمالية.
2. **بناء واجهات البرمجيات (API):** تم تطوير RESTful APIs لتوفير البيانات لتطبيق Flutter بشكل آمن.
3. **تطوير التطبيق (Mobile App):** بناء واجهات Flutter وربطها بالـAPIs ، وتفعيل ميزات الخرائط والإشعارات.

5.4 مرحلة الاختبار(Testing Phase)

- **Unit Testing:** اختبار الدوال البرمجية الفردية (مثل دالة حساب سعر الرحلة).
- **Integration Testing:** التأكد من نجاح عملية التواصل بين التطبيق وقاعدة البيانات.
- **User Acceptance Test (UAT):** تجربة التطبيق من قبل مستخدمين حقيقيين لمحاكاة عملية حجز رحلة كاملة من البداية للنهاية.

5.5 قواعد العمل(Business Rules)

1. لا يمكن حجز رحلة إذا كانت المقاعد ممتلئة.
2. يمكن للسائق تعديل الرحلة فقط قبل موعدها بـ 24 ساعة.
3. لا يسمح للمستخدم بتقييم السائق إلا بعد اكتمال الرحلة.

4. يمكن للمشرف حذف الحسابات المخالفة أو حظر المستخدمين.
5. يجب حفظ كل العمليات في سجل النشاط (Activity Log) لأغراض المراقبة.

5.6 متطلبات الأمان(Security Requirements)

1. تشفير البيانات الحساسة باستخدام خوارزمية قوية.
2. كل الاتصالات عبر بروتوكولHTTPS .
3. استخدام رموز JWT استخدمنا (sanctum)للتحقق من الجلسات.
4. تقييد صلاحيات الوصول حسب الدور(Role-Based Access) .
5. حماية واجهات البرمجة (API) من الهجمات الشائعة مثل SQL Injection وXSS.

5.7 خطة الاختبار المبدئية(Initial Test Plan)

جدول 1-5: الاختبارات المبدئية

رقم الحالة	وصف الاختبار	الإدخال	النتيجة المتوقعة
TC-01	اختبار تسجيل مستخدم جديد	بيانات مستخدم جديدة	إنشاء حساب بنجاح
TC-02	تسجيل دخول ناجح	بيانات صحيحة	الدخول إلى النظام
TC-03	حجز رحلة	اختيار رحلة متاحة	إنشاء حجز جديد
TC-04	حجز رحلة ممتلئة	اختيار رحلة ممتلئة	رسالة رفض الحجز
TC-05	إنشاء رحلة من السائق	بيانات صحيحة	إضافة الرحلة في الجدول
TC-06	إرسال إشعار	عملية حجز ناجحة	وصول إشعار للسائق والمسافر

5.8 تحليل الاختبارات الأمنية والجدوى

تم إخضاع نظام **"سافر (Safer)** "لسلسلة من الاختبارات التقنية لضمان كفاءته التشغيلية وأمنه المعلوماتي. شملت هذه الاختبارات تقييم **الجدوى التقنية(Technical Feasibility)** ، حيث تم التحقق من مواءمة الأدوات البرمجية المستخدمة مثل إطار العمل (Laravel) ولغة (Dart/Flutter) مع المتطلبات الوظيفية للنظام. كما أظهرت نتائج اختبارات الحماية قدرة النظام على التصدي لهجمات **التخمين المتكرر(Brute Force Attacks)** ؛ حيث تم تنفيذ آلية **تحديد معدل الطلبات (Rate Limiting)** التي تعمل على حظر الوصول مؤقتاً عند تجاوز سقف المحاولات المسموح به، مما يعزز من موثوقية النظام وحماية بيانات المستخدمين والسائقين. بالإضافة إلى ذلك، تم التركيز على دقة معالجة البيانات ومدخلات النظام لضمان جودة الأداء البرمجي وتقليل الأخطاء القواعدية أثناء التنفيذ، مما ينعكس إيجاباً على استقرار النظام النهائي".

5.9 حمية ضد ثغرات SQL Injection

التحقق مما إذا كانت الـ API الخاصة بتطبيقك (نقطة النهاية /api/login) محمية ضد ثغرات SQL Injection.
الهدف من استخدام جملة ' OR 1=1 -- هو محاولة خداع قاعدة البيانات لتسجيل الدخول دون الحاجة لمعرفة رقم الهاتف الحقيقي، حيث إن 1=1 هي جملة شرطية صحيحة دائماً.

تحليل المدخلات(The Payload)

- **الحقل الموجه :** حقل الـ phone.
- **القيمة المستخدمة" ' OR 1=1 -- " :**
  - تُستخدم لإغلاق حقل النص في استعلام SQL .
  - **OR 1=1:** تجعل الشرط محققاً دائماً.(True)
  - **--:** تقوم بتحويل ما تبقى من استعلام SQL الأصلي إلى "تعليق (Comment) "لكي يتم تجاهله.

النتيجة(The Result)

النتيجة التي ظهرت لك في Postman هي:

Status: 401 Unauthorized

"status": False

"Message:رقم الهاتف أو كلمة المرور غير صحيحة"

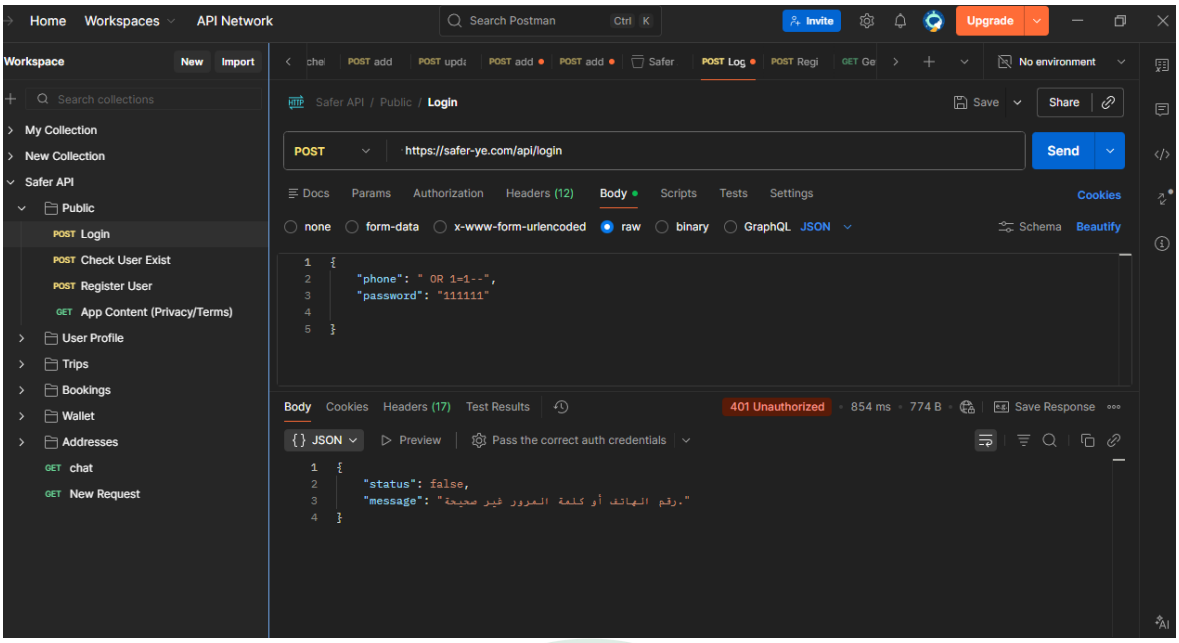
التفسير التقني للنجاح(Analysis)

هذه النتيجة **ممتازة جداً** وتعتبر نجاحاً برمجياً لك لعدة أسباب:

- **الأمان :** النظام لم يقع في فخ الاختراق، وقام بمعاملة الجملة المحقونة كنص عادي (String) وليس كأمر برمجي.

  هذا يعني أنك تستخدم غالباً **Prepared Statements** أو **ORM**(مثل Eloquent في Laravel) وهو ما يقوم بعمل "Sanitization" تلقائي للمدخلات.

- **الخصوصية :** الرسالة الموحدة "رقم الهاتف أو كلمة المرور غير صحيحة" هي ممارسة أمنية فضلى (Best Practice)، لأنها لا تخبر المهاجم أي الحقلين هو الخاطئ تحديداً.



5.10 اختبار حماية معدل الطلبات(Rate Limiting Test)

لتعزيز أمن الواجهة البرمجية (API) في مشروع **"سافر"**، تم الاعتماد على تقنية **Rate Limiting** المدمجة في إطار العمل (Laravel) للتحكم في تدفق البيانات ومنع سوء الاستخدام. تعمل هذه الآلية من خلال تعريف محددات وصول (Rate Limiters) تفرض قيوداً زمنية على الطلبات البرمجية؛ حيث تم ضبط النظام للسماح بمعدل محدد من الطلبات لكل دقيقة بناءً على عنوان البروتوكول (IP Address) الخاص بالمستخدم. تهدف هذه البنية التقنية إلى حماية موارد الخادم وقاعدة البيانات من الاستهلاك المفرط الناتج عن الطلبات الوهمية أو هجمات حجب الخدمة(DoS Attacks) ، فضلاً عن تأمين نقاط الدخول الحساسة مثل واجهة تسجيل الدخول، لضمان تجربة مستخدم آمنة وسريعة في بيئة العمل المحلية

```
RateLimiter::for('api', function (Request $request) {
    return Limit::perMinute(60)->by($request->ip());
});
```

الشرح السطري للكود

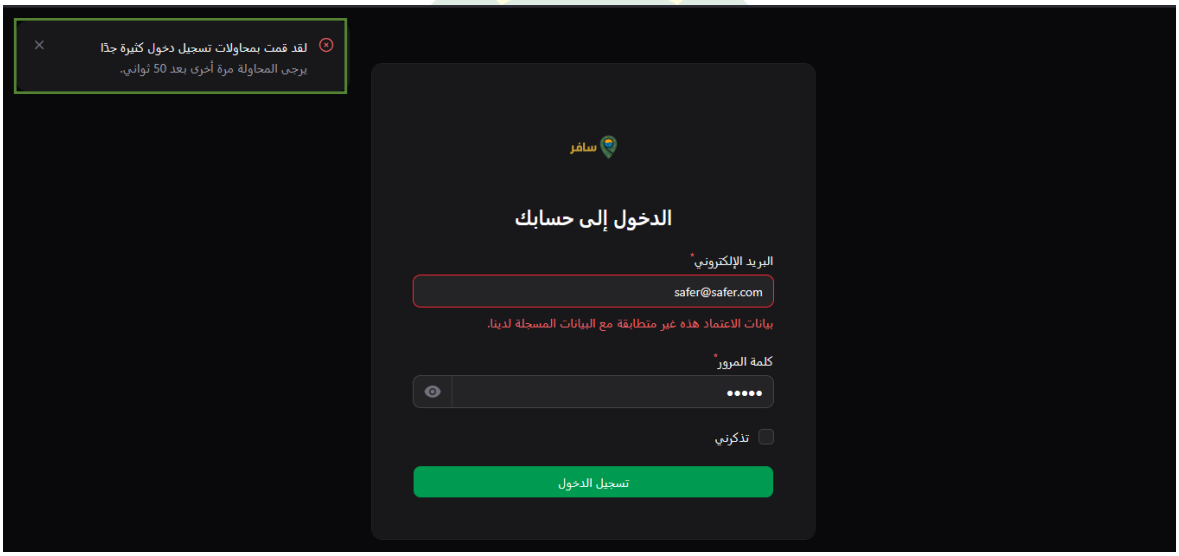
- **RateLimiter::for('api', ...)**: أنت هنا تقوم بتعريف "محدد معدل" خاص بـ الـ **API** أي أن أي طلب يأتي عبر مسارات api.php سيخضع لهذه القاعدة.
- **Limit::perMinute(60)**: هذا هو "سقف السماح"؛ حيث يسمح النظام بـ **60طلباً فقط في الدقيقة الواحدة**. إذا حاول المستخدم (أو المهاجم) إرسال الطلب رقم 61 في نفس الدقيقة، سيقوم النظام بحظره تلقائياً وإرجاع خطأ 429 Too Many Requests.

- **(->by($request->ip()**: هذا هو "مفتاح التعريف"؛ حيث يتم تطبيق الحظر بناءً على **عنوان الـ IP** الخاص بجهاز المستخدم. هذا يضمن أنك لا تحظر جميع المستخدمين، بل فقط الشخص الذي يقوم بمحاولات مكثفة من جهاز واحد.

أهمية هذا الكود لمشروع "سافر"

بما أن تطبيقك يتعامل مع حجوزات وسائقين ومبالغ مالية، فإن هذا الكود يوفر لك الحماية التالية:

- **منع هجمات الـBrute Force** : يمنع المهاجمين من تجربة آلاف كلمات المرور في ثوانٍ معدودة على صفحة تسجيل الدخول التي صممتها.
- **حماية موارد السيرفر** :يمنع استهلاك موارد قاعدة البيانات (MySQL) نتيجة طلبات وهمية مكثفة، مما يحافظ على سرعة التطبيق لبقية المستخدمين.
- **تجنب الـDOS Attacks** : يحمي الـ API من الغرق بطلبات عشوائية قد تؤدي لتوقف الخدمة في اليمن، خصوصاً مع محدودية موارد الاستضافة أحياناً.



(Chapter 6)الفصل السادس 6

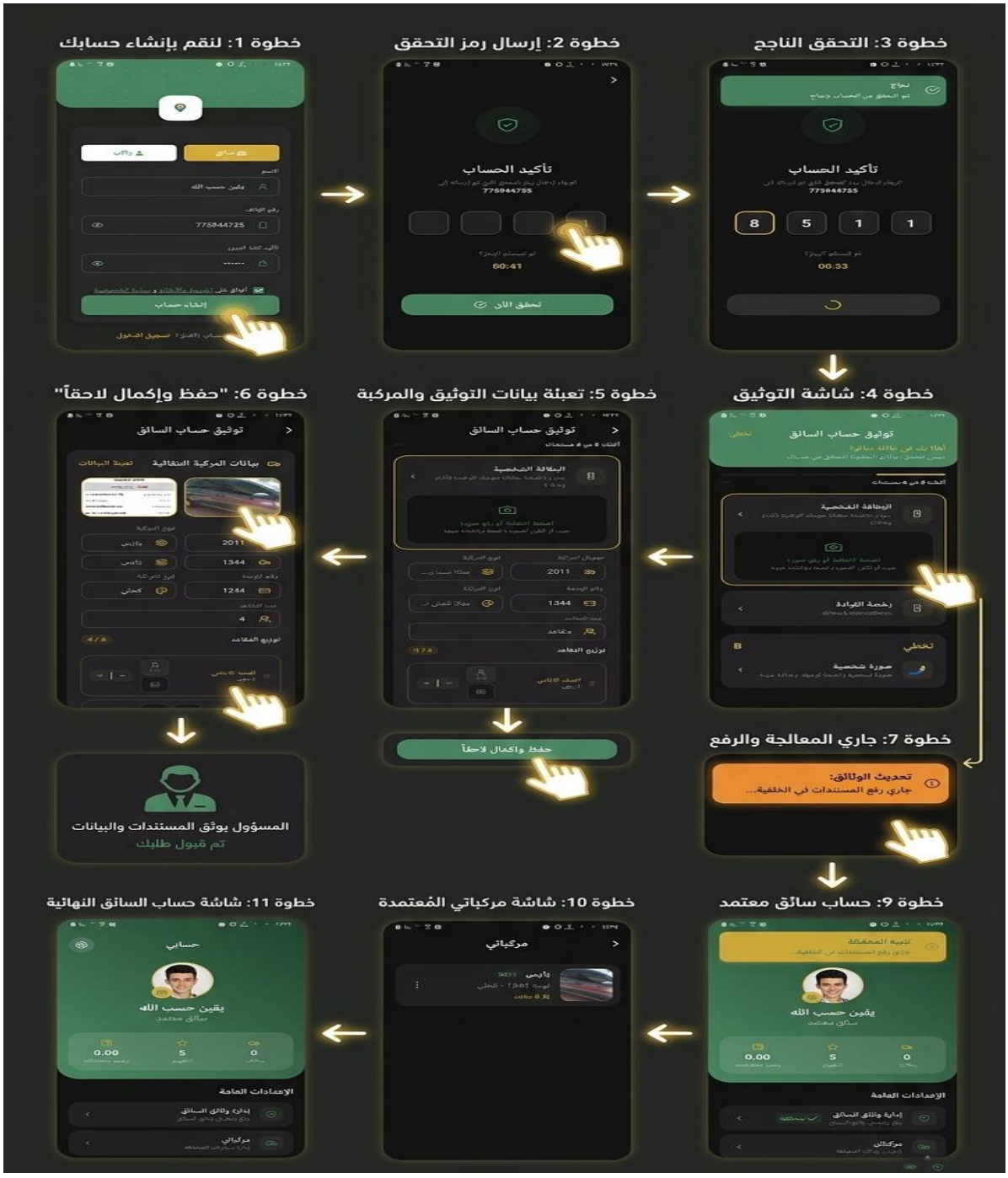
(Results and Discussions)النتائج والمناقشة

6.1 مخرجات النظام

مخرجات نظام "سافر" هي النتائج النهائية والخدمات والواجهات التي يقدمها للمستخدمين ومنها ما يلي:

6.1.1 واجهات التطبيق لأنشاء حساب سائق

يوضح الشكل (1-6) سيناريو انشاء حساب سائق في تطبيق سافر:



شكل 1-6 إنشاء حساب سائق

6.1.2 واجهات التطبيق لأنشاء رحلة

يوضح الشكل (2-6) سيناريو انشاء رحلة من قبل السائق في تطبيق سافر:

شكل 2-6 إنشاء رحلة

6.1.3 واجهات التطبيق لحجز رحلة

- **يوضح الشكل (3-6) سيناريو انشاء رحلة للمسافر في تطبيق سافر:**

شكل 3-6 حجز رحلة

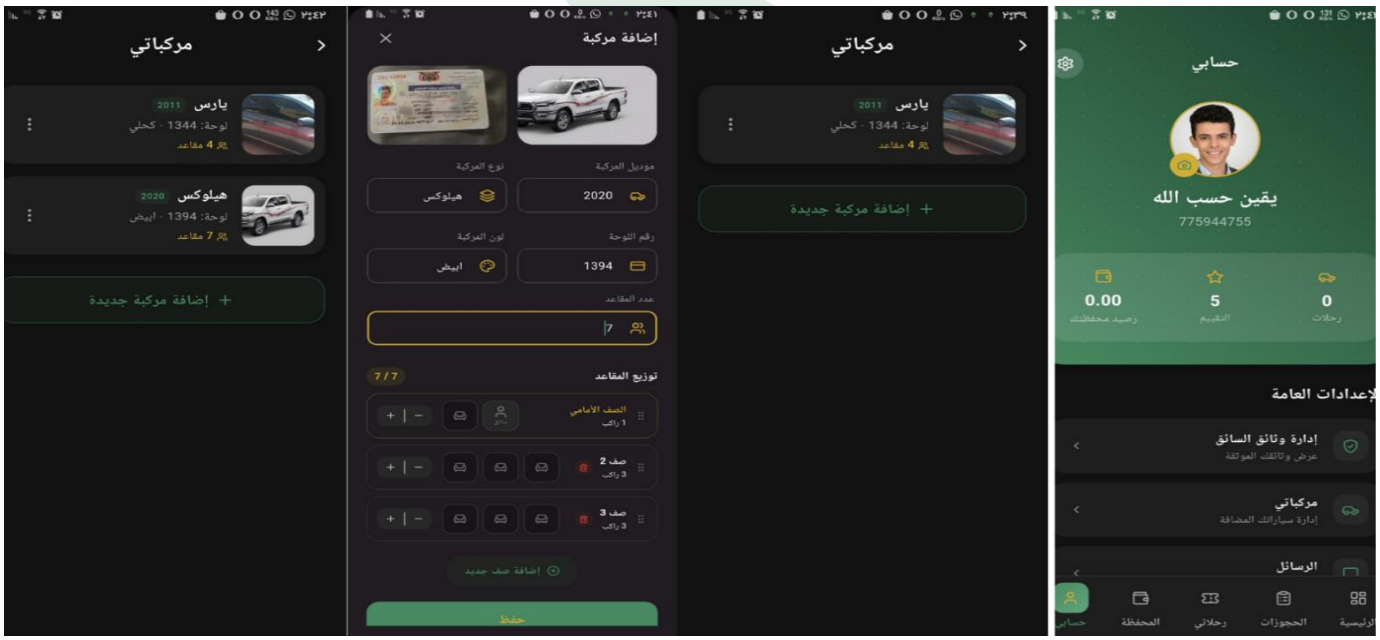
6.1.4 واجهات التطبيق لتغذية المحفظة

يوضح الشكل (4-6) سيناريو تغذية المحفظة الخاصة بالراكب في منصة سافر

شكل 4-6: تغذية محفظة

6.1.5 واجهات التطبيق لإضافة مركبة

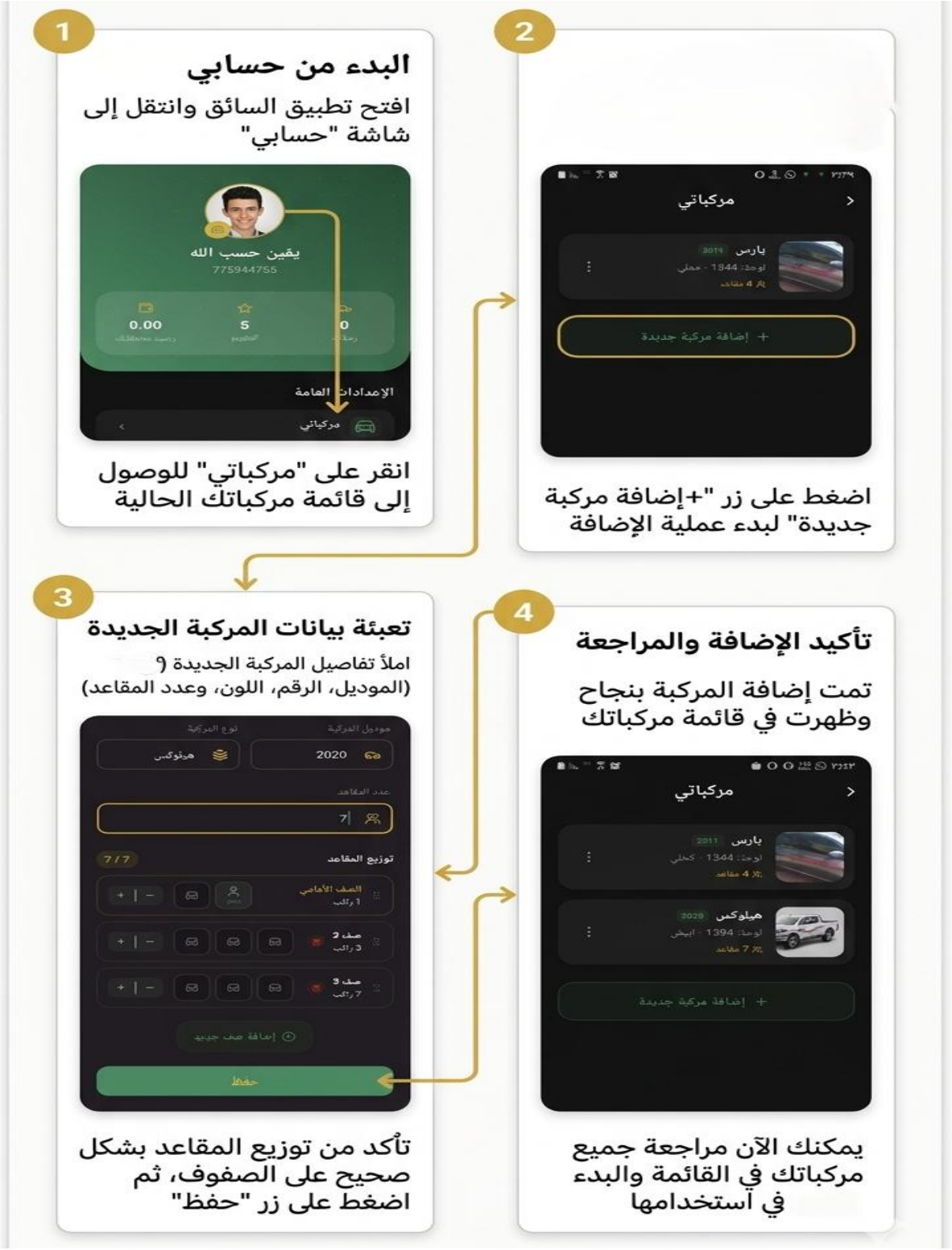
- يوضح الشكل (5-6) واجهات إضافة مركبة جديدة في تطبيق سافر:



شكل 5-6 إضافة مركبة

6.1.6 واجهات التطبيق تفاصيل إضافة مركبة

يوضح الشكل (6-6) خطوات إضافة مركبة في تطبيق سافر:



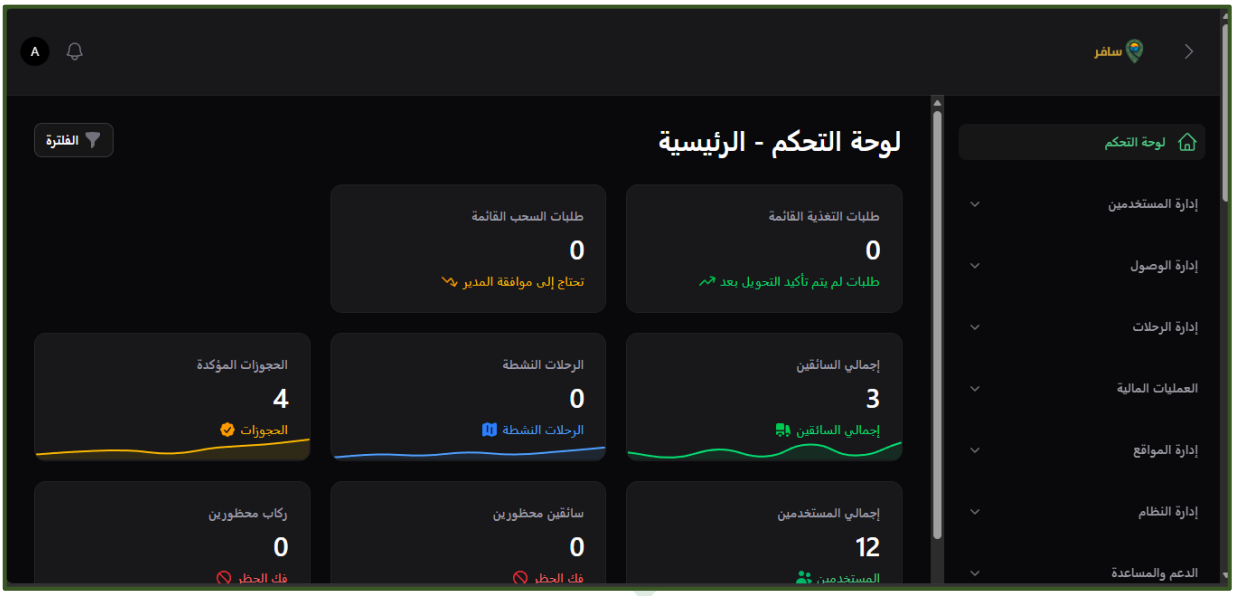
شكل 6-6:خطوات إضافة مركبة

6.1.7 واجهة لوحة التحكم:

من هنا يمكن لمدير النظام إدارة كلا من:

- ✓ **إدارة المستخدمين :**سائقين، ركاب، موظفين.
- ✓ **إدارة الوصول :**لإدارة الصلاحيات والأدوار.
- ✓ **إدارة الرحلات :**لمتابعة الجداول والمركبات والفرزات.
- ✓ **العمليات المالية :**لمتابعة المحافظ، طرق الدفع، والتقارير المالية.
- ✓ **إدارة المواقع :**لإدارة المحافظات والمدن ونقاط الانطلاق (الفرزات) داخل اليمن.
- ✓ **إدارة النظام :**للإعدادات العامة والرسائل والتنبيهات.
- ✓ **الدعم والمساعدة :**قسم خاص لحل مشاكل المستخدمين.

يوضح الشكل (7-6) واجهة لوحة التحكم الرئيسية في منصة سافر

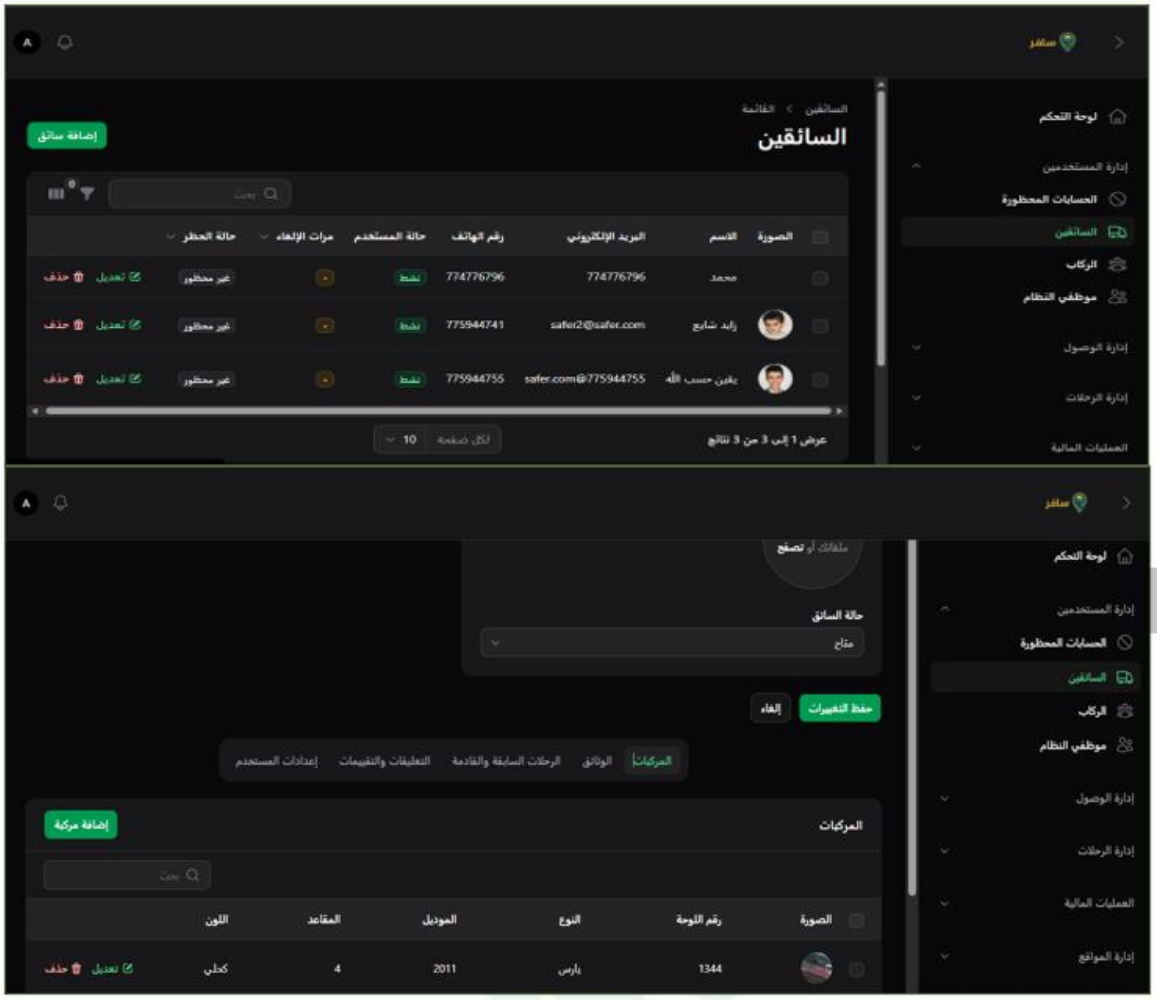


شكل 7-6 واجهة لوحة التحكم الرئيسية

تُعد هذه الواجهة **"قلب النظام "**أو **غرفة القيادة المركزية** لمنصة **سافر**، وهي مصممة خصيصاً لتمكين مدير النظام (Admin)من مراقبة وتوجيه كافة الأنشطة الحيوية للتطبيق في شاشة واحدة.

6.1.8 واجهة التحكم بالسائقين

يوضح الشكل (7-6) واجهات التحكم بالسائقين من إضافة وحذف وعرض السائقين في منصة سافر

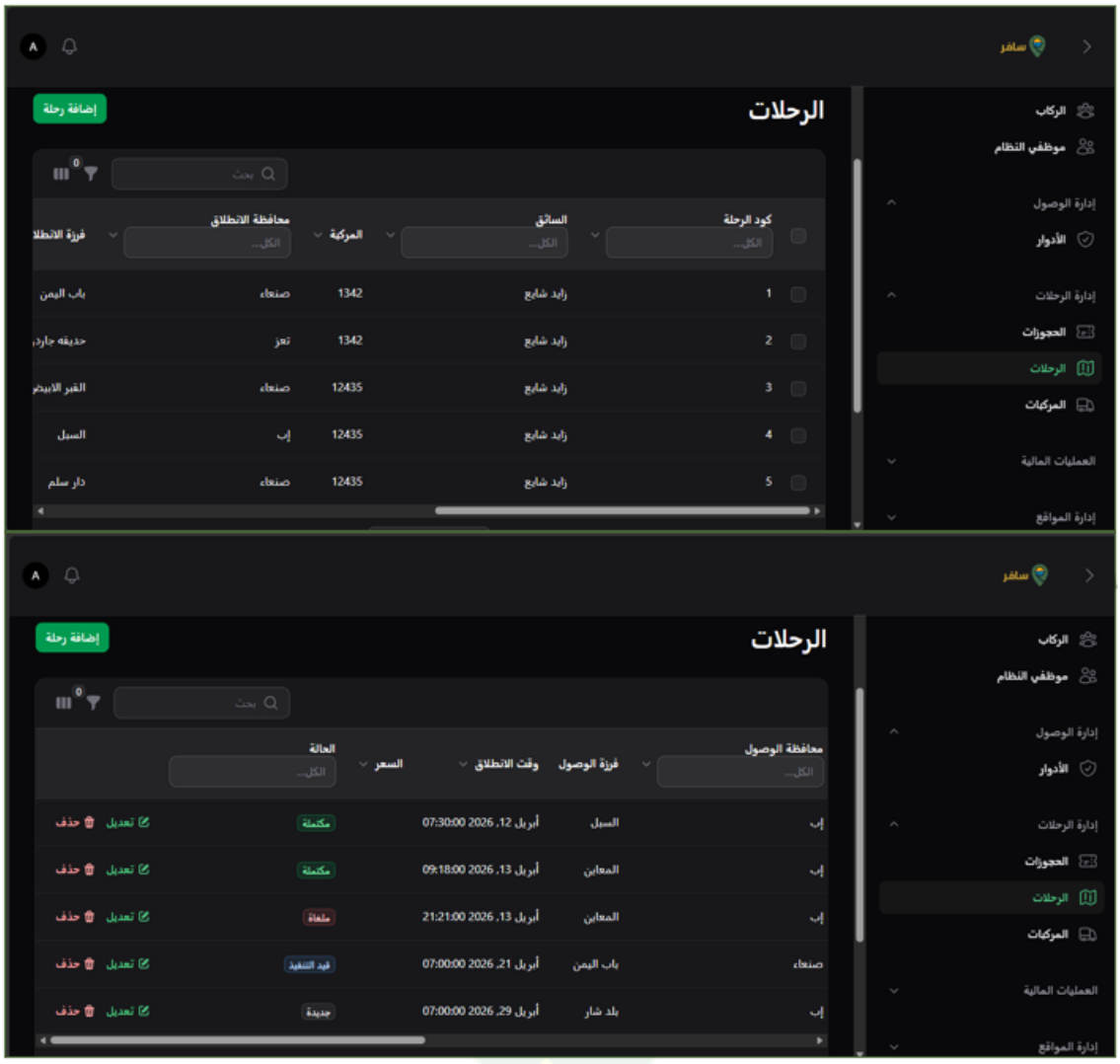


شكل 8-6 : واجهة عرض وإضافة السائقين

هذه الواجهة هي **"قائمة السائقين" (Drivers Management)**، وهي الجزء المسؤول عن إدارة قاعدة بيانات السائقين المسجلين في نظام "سافر". من هنا، يستطيع المسؤول (Admin) التحكم في كافة تفاصيل الحسابات ومراقبة سلوك السائقين وأيضا توثيق حسابات السائقين.

6.1.9 وجهات التحكم بالرحلات

يوضح الشكل (9-6) واجهات التحكم بالرحلات في منصة سافر



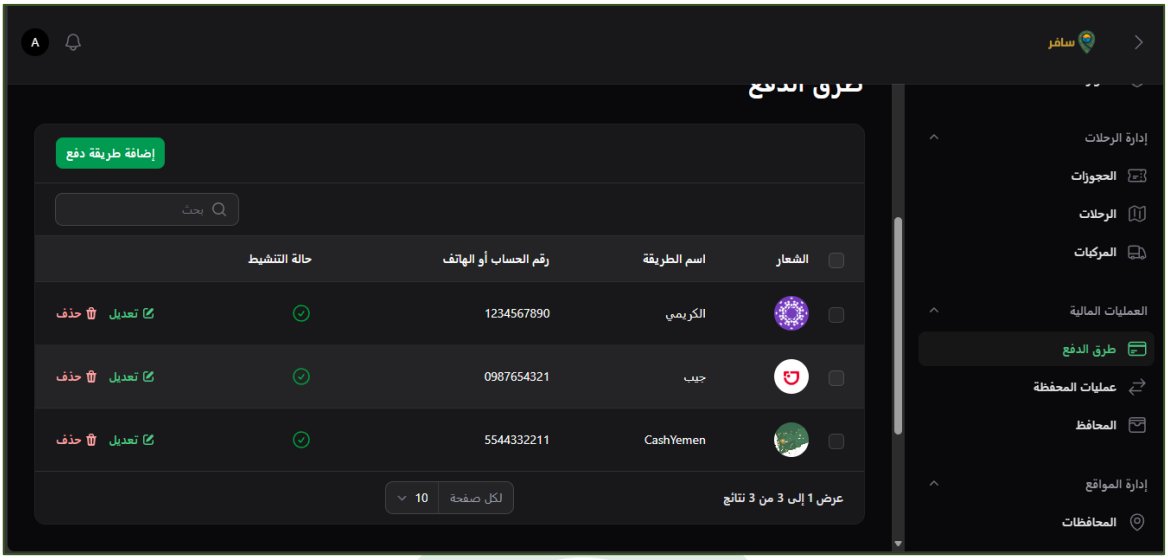
شكل 9-6: لوحة التحكم بالرحلات

الجدول يعرض السجلات البرمجية لكل رحلة، وتتوزع البيانات وفق الحقول التالية:

- **كود الرحلة (Primary Key):** المعرف الفريد لكل رحلة (مثل 1، 2، 3...). أكاديمياً، هذا هو المفتاح الأساسي الذي يمنع تكرار البيانات ويسمح بالوصول السريع للسجل.
- **السائق (Foreign Key - Driver):** ربط الرحلة بموظف محدد (يظهر اسم "زايد شايع" كقيمة متكررة)،
- **المركبة (Foreign Key - Vehicle):** يظهر رقم اللوحة أو المعرف الخاص بالسيارة (مثل 1342، 12435)، لضمان تتبع الأصول.
- **نقطة الانطلاق (Location Data):** مقسمة إلى مستويين (المحافظة والفرزة) لتحقيق دقة في البيانات الجغرافية (مثلاً: صنعاء <- باب اليمن) .

6.1.10 واجهة التحكم بطرق الدفع

يوضح الشكل (10-6) واجهة التحكم بطرق الدفع في منصة سافر



شكل 10-6: طرق الدفع

تُظهر هذه الواجهة وحدة **"طرق الدفع (Payment Methods Management)"** ضمن النظام:

- **الشعار :عنصر** مرئي يساعد في التعرف السريع على مقدم الخدمة) .
- **اسم الطريقة :**يمثل اسم المحفظة أو البنك المستخدم لتحويل الأموال.
- **رقم الحساب أو الهاتف :**المعرف الرقمي الذي تتم عليه العمليات المالية، وهو حقل حساس يتطلب تدقيقاً عند الإدخال.
- **حالة التنشيط :**أيقونة (Checkmark) تشير إلى ما إذا كانت هذه الطريقة مفعلة للاستخدام في النظام أم لا

تطبق هذه الواجهة نموذج **CRUD**الكامل (إنشاء، قراءة، تحديث، حذف):

- **الإضافة :(Create)** عبر زر **"إضافة طريقة دفع"**، والذي يفتح نموذجاً لإدخال بيانات وسيلة دفع جديدة.
- **التحديث والحذف :(Update & Delete)** يظهر في يسار كل سجل خياران:
  - **تعديل :**لتصحيح رقم الحساب أو تغيير الحالة.
  - **حذف :**لإزالة وسيلة الدفع من النظام (وعادة ما يكون حذفاً منطقياً *Soft Delete*للحفاظ على سجلات العمليات المالية المرتبطة بها).

6.1.11 واجهة التحكم بالمحافظ الالكترونية

يوضح الشكل (6-11) واجهة التحكم بالمحافظ الالكترونية

شكل 6-11: المحافظ الالكترونية

6.1.12 واجهة التحكم بكشف الحساب

يوضح الشكل (6-12) واجهة التحكم بكشف الحساب الخاص بالمستخدمين في منصة سافر

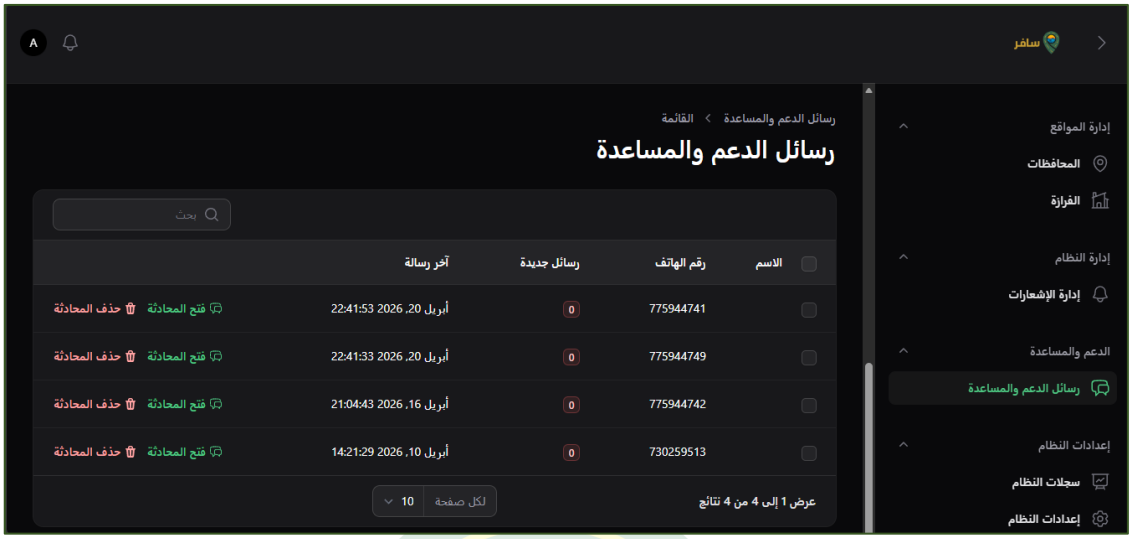
شكل 6-12: كشف حساب

تعمل هذه الشاشة على إدارة الأرصدة الرقمية للمستخدمين (سائقين أو ركاب)

- **المستخدم(Account Holder) :** كائن فريد برمجياً (UUID) يمنع اختلاط الحسابات عند تشابه الأسماء.
- **الرصيد(Balance) :** القيمة المالية المخزنة بدقة رقمية (Decimal) لتفادي أخطاء الحساب.
- **كشف الحساب :(Statement)** سجل تتبع (Audit Trail) لضمان الشفافية والرقابة المالية.
- **التعديل(Adjustment) :** صلاحية إدارية عليا لإدارة الأرصدة يدوياً مع التوثيق(Logging) .

6.1.13 واجهة التحكم بالدعم والمساعدة

يوضح الشكل (13-6) واجهة التحكم برسائل الدعم والمساعدة في منصة سافر



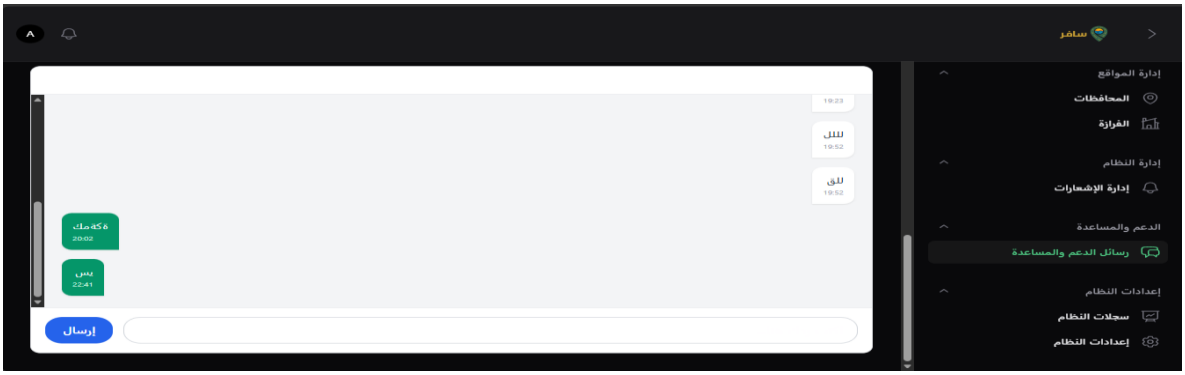
شكل 13-6: الدعم والمساعدة

واجهة قائمة الرسائل (السجل العام)

- **وظيفة النظام :**تعمل كمركز رئيسي لاستقبال جميع طلبات المساعدة من المستخدمين وتنظيمها.
- **هوية المستخدم :**يعتمد النظام على "رقم الهاتف" لربط كل رسالة بصاحبها بشكل دقيق.
- **ترتيب الأولويات :**يساعد عمود "رسائل جديدة" مدير النظام على معرفة الطلبات التي لم يتم الرد عليها بعد.
- **التوثيق الزمني :**يظهر تاريخ ووقت آخر رسالة لضمان متابعة سرعة الاستجابة وتقديم الخدمة في وقتها.
- **التحكم بالبيانات :**تتيح الواجهة خيارات مباشرة لفتح المحادثة أو حذفها لتنظيم السجلات.

6.1.14 واجهة التحكم بالرسائل

يوضح الشكل (14-6) واجهة الدردشة المباشرة بين الراكب والسائق في منصة سافر



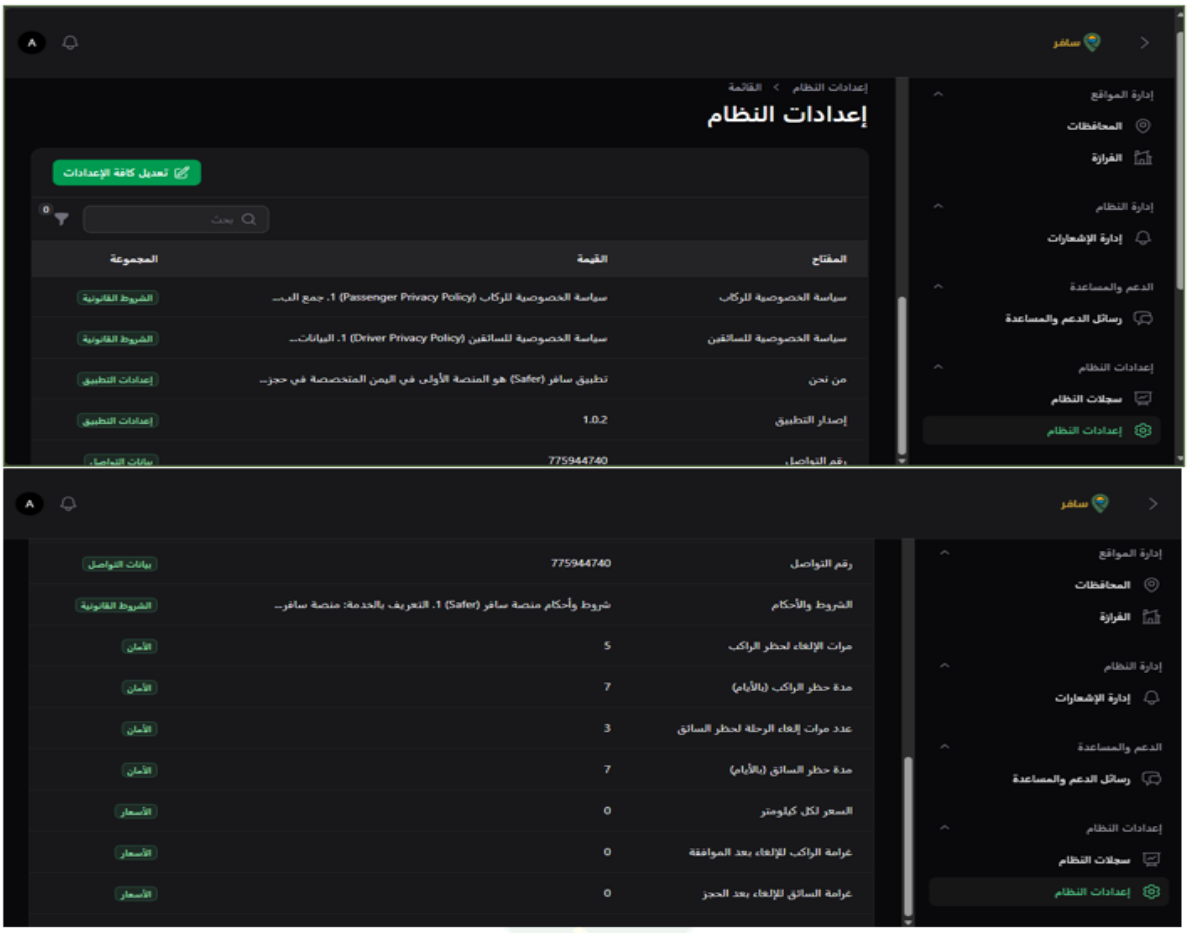
شكل 14-6: واجهة المحادثة

واجهة المحادثة (الدردشة المباشرة)

- **أسلوب التواصل :**تعتمد على التراسل الفوري لضمان حل المشكلات بشكل سريع ومباشر.
- **التمييز البصري:**
  - **جهة اليمين :**تظهر رسائل المستخدم لتوضيح طلبه.
  - **جهة اليسار :**تظهر ردود الإدارة لتمييزها عن كلام المستخدم.
- **تحديث البيانات :**تظهر الرسائل فور إرسالها واستقبالها دون الحاجة لإعادة تحميل الصفحة.
- **سياق الحوار :**يتم عرض الرسائل بشكل متسلسل زمنياً، مما يسهل على الموظف مراجعة أصل المشكلة وتفاصيلها السابقة.
- **سهولة الاستخدام :**يوفر التصميم حقل إدخال بسيط وزر إرسال واضح لضمان سلاسة التعامل مع الطلبات.

6.1.15 واجهة التحكم بالإعدادات

يوضح الشكل (15-6) وجهات التحكم بالإعدادات في منصة سافر



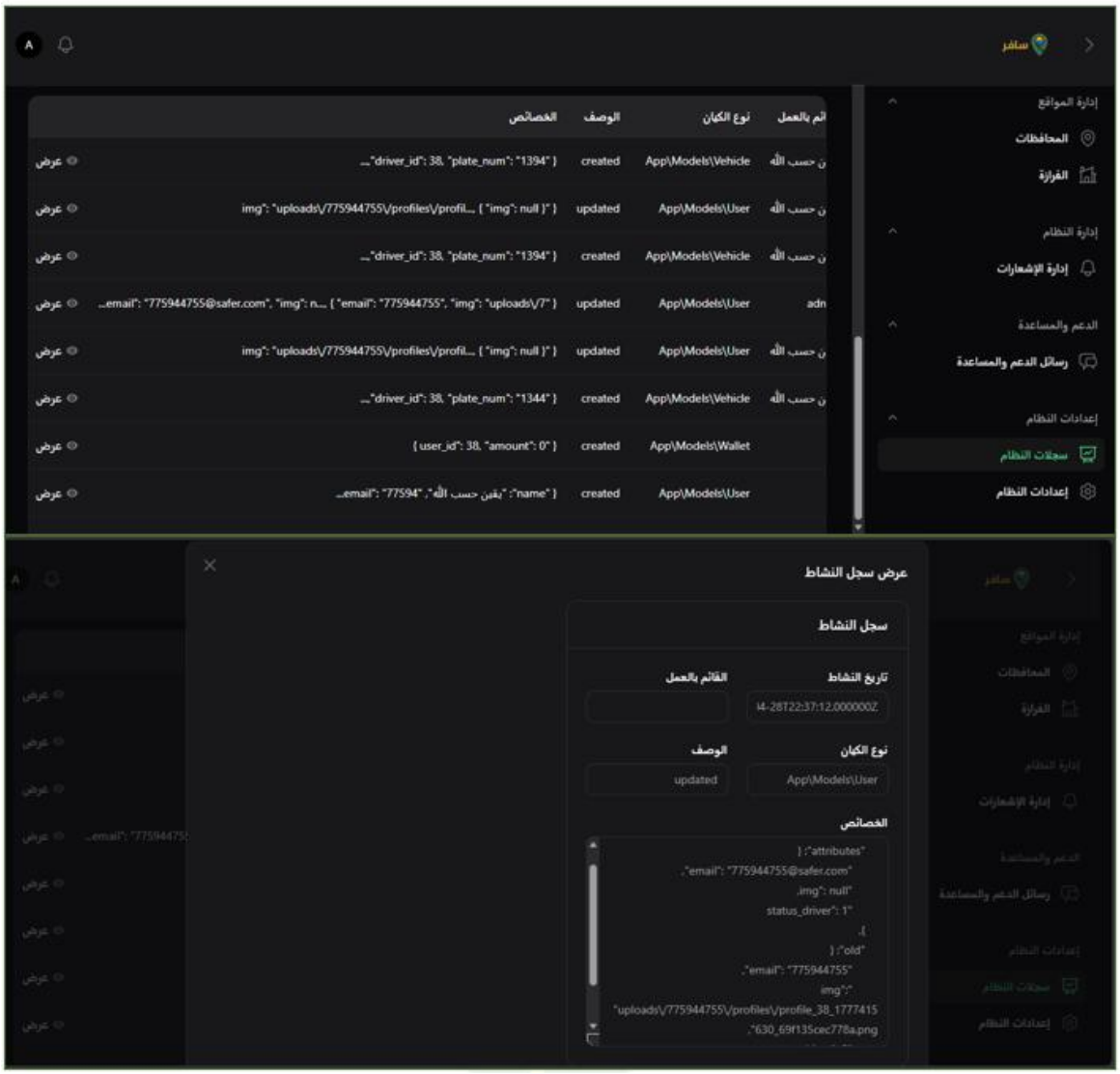
شكل 15-6: التحكم بالاعدادات

إعدادات النظام (لوحة التحكم المركزية)

- **هيكلية الإعدادات (المفتاح والقيمة):**
  - يعتمد النظام على نموذج "المفتاح والقيمة"، حيث يمثل **المفتاح** اسم الإعداد (مثل: رقم التواصل)، و**القيمة** هي المعطى الفعلي (مثل: 775944740) هذا يسهل تحديث النظام دون تعديل الكود البرمجي.
- **تصنيف المجموعات:(Categorization)**
  - يتم تنظيم الإعدادات في مجموعات ملونة لتسهيل الوصول إليها، مثل: (الشروط القانونية، إعدادات التطبيق، الأمان، الأسعار).
- **إدارة السياسات والقوانين:**
  - تسمح الواجهة بالتحكم في "سياسة الخصوصية" و"الشروط والأحكام" لكل من الركاب والسائقين، مما يضمن الجانب القانوني للمنصة.
- **قواعد الأمان والحظر:(Security Logic)**
  - تتضمن إعدادات دقيقة لمنع سوء الاستخدام، مثل: تحديد عدد مرات إلغاء الرحلة المسموح بها قبل حظر الركاب أو السائقين، وتحديد مدة الحظر بالأيام.
- **الإعدادات المالية:(Pricing Rules)**
  - تسمح بضبط "السعر لكل كيلومتر" وتحديد "غرامات الإلغاء" سواء بعد الموافقة أو بعد الحجز، مما يجعل النظام مرناً في التعامل مع التغيرات الاقتصادية.
- **إصدار التطبيق والتعريف:**
  - تحتوي على معلومات الإصدار الحالي (مثل: 1.0.2) ونصوص "من نحن" التي تظهر للمستخدمين في واجهة التطبيق.
- **المرونة الإدارية:**
  - يوفر زر "تعديل كافة الإعدادات" إمكانية تحديث شاملة لجميع معايير العمل بضغطة واحدة، مما يعطي الإدارة سيطرة كاملة ولحظية على سير العمل.

6.1.16 واجهات التحكم بسجلات النظام

يوضح الشكل (16-6) واجهة التحكم بسجلات النظام في منصة سافر



شكل 16-6: سجلات النظام

سجلات النظام (مراقب النشاط)

- **الوظيفة التقنية (التتبع):**
  - تقوم هذه الواجهة بتسجيل كل حركة تحدث داخل النظام، سواء كانت إضافة بيانات جديدة، أو تعديل بيانات موجودة، مما يضمن وجود مرجع كامل لكل ما يفعله الموظفون أو المستخدمون.
- **بيانات السجل (تفاصيل الحركة):**
  - **القائم بالعمل :**يوضح اسم الشخص الذي قام بالإجراء، مما يسهل المساءلة الإدارية.
  - **نوع الكيان :**يحدد الجزء الذي تم التأثير عليه (مثل: المستخدمين، المركبات، أو المحافظ المالية).

- **الوصف :**يصف طبيعة الحركة (مثل: "إنشاء" لحركة جديدة، أو "تحديث" لتعديل بيانات).

- **نافذة تفاصيل النشاط (عرض الخصائص):**

  - عند الضغط على "عرض"، تظهر نافذة تعرض البيانات الخام قبل وبعد التعديل.
  - **البيانات القديمة :**تظهر المعلومات السابقة قبل إجراء التغيير.
  - **البيانات الجديدة :**تظهر المعلومات الحالية بعد التعديل، مما يسمح بمقارنة الفروق واكتشاف أي أخطاء.

- **التوثيق الزمني الدقيق:**

  - يتم تسجيل الوقت بالتاريخ والثانية والملي ثانية، وهو أمر ضروري جداً في التحقيقات التقنية أو عند حدوث تداخل في العمليات المالية.

- **الشفافية والأمان:**

  - تعمل هذه السجلات كـ "صندوق أسود" للنظام؛ حيث لا يمكن إنكار أي فعل تم القيام به، مما يرفع من مستوى الأمان ويمنع التلاعب بالبيانات الحساسة مثل أرصدة المحافظ أو بيانات الرحلات.

6.2 مناقشة النتائج:

أثبت النظام قدرة عالية على التعامل مع الحجوزات المتزامنة، كما أظهرت النتائج دقة كبيرة في حساب التكلفة بناءً على الموقع الجغرافي. تمت مناقشة سهولة الاستخدام مع مجموعة من المستخدمين وأبدوا إعجابهم بسرعة الاستجابة ووضوح الخطوات.

7 الفصل السابع(Chapter 7)

الاستنتاجات والتوصيات

(Conclusions and Recommendations)

7.1 الاستنتاجات(Conclusions)

- **تطوير منصة نقل ذكية ومتكاملة :**تم بنجاح بناء نظام "سافر" كحل تقني شامل لإدارة وتنظيم الرحلات الجماعية بين المحافظات، يربط بكفاءة بين المسافرين والسائقين ومكاتب النقل (الفرزات).
- **تفعيل الخدمات المعتمدة على الموقع(LBS) :** نجح المشروع في دمج خرائط جوجل (Google Maps) ليس فقط للعرض، بل كأداة برمجية لحساب المسافات، تحديد مواقع الانطلاق، وتوفير نظام ملاحة دقيق يسهل وصول الأطراف لبعضها البعض.
- **نظام مالي رقمي آمن :**تم تطوير "محفظة إلكترونية" متكاملة داخل التطبيق، تتيح دورة مالية كاملة تبدأ من شحن الرصيد وتوثيق السندات وصولاً إلى الدفع الآلي للحجوزات، مما يقلل الاعتماد على النقد ويزيد من موثوقية العمليات.
- **تعزيز الأمان والتحقق التقني :**تم بناء نظام صارم للتحقق من هوية السائقين وصلاحية مركباتهم عبر وحدة "إدارة الوثائق"، مما يضمن أن جميع الرحلات المسجلة في المنصة تتم عبر سائقين مرخصين ومركبات آمنة.
- **مرونة التواصل والدعم :**يوفر النظام قنوات اتصال فورية ثنائية الغرض (دردشة للرحلات ودعم فني للمنصة) باستخدام بنية تحتية برمجية موحدة، مما يعزز من تجربة المستخدم وسرعة استجابة الإدارة.
- **استخدام تقنيات حديثة وعصرية :**أثبت استخدام إطار العمل (Laravel) في الخلفية و (Flutter) في التطبيق كفاءة عالية في سرعة الأداء، سهولة الصيانة، والقدرة على إصدار التطبيق لنظامي Android و iOSبنفس الجودة.

7.2 التوصيات(Recommendations)

- **التوسع في بوابات الدفع الإلكتروني :**نوصي بربط النظام بشكل مباشر مع واجهات البرمجة (APIs) للمصارف والمحافظ الإلكترونية اليمنية (مثل الكريمي، فلوسك، وكاش) لأتمتة عمليات الشحن الفوري دون الحاجة للمراجعة اليدوية.
- **نظام التتبع اللحظي للرحلات(Live Tracking) :** تطوير ميزة تتبع الحافلة مباشرة على الخريطة أثناء سير الرحلة، ليتمكن ذوي المسافرين من معرفة موقعهم الحالي وموعد وصولهم المتوقع لزيادة الطمأنينة والأمان.
- **دعم الذكاء الاصطناعي في التسعير :**استخدام خوارزميات تعلم الآلة للتنبؤ بأسعار الرحلات بناءً على العرض والطلب والمواسم، وتقديم "عروض ذكية" للمسافرين الدائمين.
- **توسيع نطاق الخدمات :**إمكانية تطوير النظام ليشمل خدمات شحن الطرود بين المحافظات (Safer Express) أو خدمات استئجار السيارات الخاصة.
- **تحليل البيانات الضخمة(Big Data) :** تطوير لوحات تحكم متقدمة لمتخذي القرار تعرض إحصائيات حول أكثر الخطوط طلباً وأوقات الذروة، للمساعدة في تنظيم حركة النقل العام بشكل أفضل.

7.3 الخاتمة(Conclusion)

تعد منصة "سافر" خطوة طموحة نحو رقمنه قطاع النقل في اليمن، وتوفير حل تكنولوجي مبتكر لمشكلة عانى منها المسافرون لسنوات طويلة. فمن خلال تبسيط إجراءات الحجز، وتقليل أوقات الانتظار، وتوفير معلومات شاملة حول مسارات السفر والأسعار، تهدف المنصة إلى تحسين جودة الحياة وتسهيل حركة النقل في البلاد.

ولكن لا تقتصر أهداف المنصة على ذلك، بل تمتد لتشمل بناء بيئة نقل آمنة وشفافة تعتمد على البيانات الجغرافية والمالية الدقيقة. فمن خلال توفير معلومات حول حالة الطرق وظروف الطقس، تساهم المنصة في تعزيز سلامة المسافرين وتقليل حوادث المرور. ومن خلال توفير معلومات حول أسعار التذاكر والرسوم، تساهم المنصة في تعزيز الشفافية في قطاع النقل وحماية حقوق المسافرين.

نأمل أن تكون هذه المنصة هي النواة الأولى لنظام نقل وطني موحد يربط جميع مدن الوطن، ويساهم في دفع عجلة التحول الرقمي المنشود، لخدمة المواطن وتسهيل سبل حياته اليومية. فمن خلال دمج جميع وسائل النقل في نظام واحد، يمكن توفير تجربة سفر سلسة ومتكاملة للمسافرين. ومن خلال استخدام التكنولوجيا الحديثة، يمكن تحسين كفاءة قطاع النقل وتقليل تكاليف التشغيل.

تعد منصة "سافر" مثالاً ملموساً على كيف يمكن للتكنولوجيا أن تساهم في تحسين حياة الناس وتعزيز التنمية المستدامة. فمن خلال توفير حلول مبتكرة لمشاكل قديمة، تساهم المنصة في بناء مستقبل أفضل لليمن. ومن خلال تعزيز التحول الرقمي في قطاع النقل، تساهم المنصة في دمج البلاد في الاقتصاد العالمي وتعزيز القدرة التنافسية للشركات اليمنية.

المراجع

تضم هذه القائمة كافة المصادر والمراجع التي تم الاستناد إليها في إعداد مشروع (سافر)، والتي شملت الدراسات السابقة، والوثائق التقنية التطبيقات (المواقع الرسمية):

جامعة آزال للتنمية البشرية(2025) . دليل إعداد مشاريع التخرج - كلية الهندسة وتقنية المعلومات، صنعاء، اليمن.

وزارة النقل - الجمهورية اليمنية(2026) . شبكة النقل البري وخطوط السير الدولية والبينية، متاح على الموقع الرسمي للوزارة :https://mot.gov.ye [تاريخ الزيارة: 15 مايو 2026].

منصة الوصل للنقل البري(2025) . تطبيق "خط وصل" لربط المسافرين والسائقين في اليمن، متاح على متجر تطبيقاتGoogle Play عبر
الرابط:https://play.google.com/store/apps/details?id=com.waslapps&hl=ar

شركة يمن تكسي(2025) . الموقع الرسمي والخدمات الرقمية لنظام "اليمن تكسي" و"الفخامة"، متاح على الرابطhttp://yementaxi.com ، أو عبر متجر التطبيقات :
https://apps.apple.com/eg/app/id6478607554 [تاريخ الزيارة: 10 مايو 2026]

تطبيق مشواري(2025) Mashwari . نظام خدمات النقل الذكي وتتبع الرحلات في المدن اليمنية، متاح على متجر تطبيقات Google Play عبر الرابط :
https://play.google.com/store/apps/details?id=osys_solution.masheri
[تاريخ الزيارة: 10 مايو 2026]

BlaBlaCar Official Website (2026). How Carpooling Works for Inter-city Trips. Available at: https://www.blablacar.com [Accessed: 12 April 2026]

Careem Help Center (2026). Payment Methods and Wallet Services in MENA Region. Available at: https://help.careem.com [Accessed: 5 March 2026].

Flutter Documentation (2026). *Build expressive and-flexible UI with Flutter*. Available at**:** **https://docs.flutter.dev** [Accessed: 20 February 2026].

InDrive (2026). *The Real-time Deal Model for Ride-hailing*. Available at: https://indrive.com [Accessed: 15 March 2026].

Laravel Framework (2026). Laravel Documentation - The PHP Framework for Web Artisans. Available at**:** **https://laravel.com/docs** [Accessed: 18 January 2026].

Uber Technologies Inc. (2026). Global Operational Scope and Dynamic Pricing Algorithms. Available at: https://www.uber.com [Accessed: 10 April 2026].