الجمهورية العربية السورية الجامعة السورية الخاصة
كلية الهندسة المعلوماتية – قسم هندسة البرمجيات

نظام إدارة فندق

تقرير نهائي لمشروع التخرج الثاني

قدم استكمالاً لمتطلبات الحصول على درجة البكالوريوس في هندسة المعلوماتية

هندسة البرمجيات ونظم المعلومات

إعداد:

فاروق جميل شحاده

بإشراف: المهندسة ثراء اصلان

العام الدراسي 2025/2024

إقرار المشرف

أشهد بإن إعداد هذا المشروع الموسوم نظام إدارة فندق

والمعد من قبل الطالب فاروق جميل شحاده

قد تم تحت إشرافي في قسم هندسة البرمجيات ونظم المعلومات كلية هندسة الحاسوب والمعلوماتية والاتصالات.

وهو جزء من متطلبات نيل شهادة الإجازة في هندسة البرمجيات ونظم المعلومات

المرتبة العلمية مدرّسة الاسم ثراء أصلان التوقيع

التاريخ 2025/2/25

شكر وتقدير

إلى أساتذتي
كل الشكر والتقدير أتوجه به لحضراتكم لما قمتم بفعله حتى أصل لأعلى المستويات وأفضل مكانة، فشكرًا جزيلًا على دعمكم المستمر.
صاحب الفضل الأعظم وملهمي الأول الدكتور مهيب النقري.
الى من كانت معلمتي وموجهتي وراعية أفكاري ... محفزتي الرئيسية والداعمة الأولى الأستاذة المهندسة ثراء اصلان.
وإلى الأساتذة الفضلاء صاحب الدور العظيم الدكتور أكرم مسوح، الدكتور وسيم احمد
الدكتورة كريستين زينية، الأستاذ شادي بليدي، الأستاذة روان الكردي، الأستاذ أمجد الحجازي
الأستاذ ماهر الصارم .

إهداء

هداء إلى أمّي

إلى نبع الحنان، وملاذ الروح، إلى التي سهرت الليالي واحتملت لأجل أن أبلغ هذه اللحظة،
يا أمي، أنتِ المعنى الحقيقي للتضحية، وكل ما أنجزته هو امتداد لعطائك.
لكِ حبي وامتناني ما حييت.

إهداء إلى أبي

إلى من كان سندًا لا يميل، إلى من غرس في قلبي العزم والإصرار،
يا أبي، أنتَ البوصلة التي دلّتني على طريق النجاح، وصدى صوتك كان يدفعني دائمًا إلى الأمام.
لك كل الشكر والتقدير.

إهداء إلى خطيبتي

إلى من كانت النور في عتمتي، والحافز في ضعفي،
إلى من آمنت بي في كل المراحل، ورافقتني قلبًا ودعاءً،
وجودك في حياتي هو أجمل ما حدث لي، وهذه الفرحة لا تكتمل إلا بكِ.

إهداء إلى إخوتي مصطفى و عمر

إلى من شاركوني اللحظات بكل تفاصيلها... فرحًا وحزنًا، تعبًا ونجاحًا،
إلى إخوتي الذين كانوا الرفقة في الطريق، والسند في الشدائد،
أنتم العائلة التي أفتخر بها دومًا، وجزء لا يتجزأ من هذا الإنجاز.
كل حرف في هذا العمل يحمل دعاءً لكم، ومكانة في قلبي لا تزول..

إهداء إلى أصدقاء العمر
إلى من شاركوني الضحكة الصافية، والنصيحة الصادقة، والسند في كل مراحل الطريق،
أنتم لستم فقط أصدقاء، بل أنتم إخوة اخترتهم لي الأيام.
هذا الإنجاز يحمل في طياته أثر دعمكم وذكريات لا تُنسى... شكرًا لأنكم كنتم دائمًا هناك.

قاسم , خالد , هارون ,رائد , يزن ,محمد عمار, احمد ,محمد امين ,نوار , رويد

كرم , بشار , محمد عوض ,عمار , ناصر .

الملخص:

نظراً لتطور التكنولوجيا واحتلالها جزءا كبيراً من حياتنا اليومية، تمت أتمتة اغلب النشاطات اليومية، فقامت الفنادق بالبحث عن طرق جديدة لتواكب احتياجات السوق. حيث تطورت تطبيقات المحمول لتكون أداة قيمة لبدء المشاريع والاعمال. فالعملاء مهتمون بالحصول على خدمتهم بطريقة بسيطة وسهلة الوصول، هذا الذي يمكن ان توفره التكنولوجيا (التطبيقات) بضغطة واحدة، فإنها تؤمن تفاعلاً سريعاً وفعالاً وهذا ما يشكل أحد المتطلبات التي يرغب بها الطرفان (مدير الخدمة والعميل).

ومنه انطلقت فكرة مشروعنا الذي يهدف لتأمين نظام برمجي متكامل لإدارة وحجز فندق، يتم من خلاله التواصل بين الفندق والزبون حيث يؤمن جميع الخدمات.

كما انه يؤمن تقارير تحليلية ويساعده في التسويق بشكل أفضل حسب طلب الزبون.

الفهرس

العنوان	الصفحة
الفصل الأول المقدمة	1
1.1 المقدمة	2
2.1 الأهمية	2
3.1 الهدف من النظام	3
4.1 منهجية التطوير	3
5.1 التخطيط	4
1.5.1 دراسة مبررات النظام	4
2.5.1 مخططات العمل	5
3.5.1 ضبط وتوجيه المشروع	6
الفصل الثاني جمع المتطلبات	7
1.2 المعنيين بالنظام	8
2.2 الطرائق المتبعة في جمع المتطلبات	8
1.2.2 الدراسة المرجعية للأنظمة المشابهة	9
2.2.2 الاستبيان	10
3.2.2 المقابلة	11
4.2.2 المتطلبات الوظيفية	12
5.2.2 مصفوفة تتبع المتطلبات	14
6.2.2 المتطلبات الغير وظيفية	16
الفصل الثالث مرحلة التحليل	17
1.3 مستخدمي النظام	18
2.3 تصنيف حالات الاستخدام	18
3.3 مخطط حالات الاستخدام وتوصيفه	19
4.3 التوصيف	20
5.3 مخطط Activity Diagram	27
6.3 مخطط علاقات البيانات	30
الفصل الرابع مرحلة التصميم	31

1.4 تصميم النظام الفيزيائي	32
2.4 التصميم المعماري	32
3.4 الأدوات المستخدمة في تطوير عناصر النظام	33
4.4 هيكل تفصيلي لكل جدول	34
5.4 مخطط Class	35
الفصل الخامس التطبيق والاختبار	36
1.5 مرحلة التطبيق	37
1.1.5 قاعدة البيانات	37
2.1.5 واجهات رئيسية في النظام	38
3.1.5 تطبيق الموبايل	42
4.1.5 واجهة قابلة للبرمجة API	48
2.5 مرحلة الاختبار	51
3.5 حالات الاختبار	53
الفصل السادس النشر	60
المراجع	62
الملاحق	63
ملحق عن PHP	63
ملحق عن Flutter	64
ملحق عن mysql	65

فهرس الجداول

الجدول	الصفحة
المدراء الذين تم مقابلتهم	11
مصفوفة تتبع المتطلبات	14
مستخدمي النظام	18
تصنيف حالات الاستخدام	18
جدول حالات الاختبار 1-1	53
جدول حالات الاختبار 1-2	53

جدول حالات الاختبار 1-3	54
جدول حالات الاختبار 2-1	54
جدول حالات الاختبار 2-2	55
جدول حالات الاختبار 2-3	55
جدول حالات الاختبار 2-4	55
جدول حالات الاختبار 2-5	56
جدول حالات الاختبار 3-1	56
جدول حالات الاختبار 3-2	56
جدول حالات الاختبار 3-3	57
جدول حالات الاختبار 3-4	57
جدول حالات الاختبار 4-1	57
جدول حالات الاختبار 4-2	58
جدول حالات الاختبار 4-3	58
جدول حالات الاختبار 4-4	59
جدول حالات الاختبار 4-5	59

فهرس الأشكال

الشكل	الصفحة
أهمية المشروع	2
منهجية الشلال	3
مرحلة التخطيط	4
مخطط Gantt للمشروع الفصلي	5
مخطط Gantt للمشروع التخرج1	5

مخطط Gantt للمشروع التخرج2	5
استراتيجيات جمع المتطلبات	8
الاستبيان	10
مخطط حالات الاستخدام	19
مخطط نشاط نظام إدارة فندق	27
مخطط نشاط نظام إدارة فنادق لجانب مسؤول الفندق	28
مخطط نشاط نظام إدارة فنادق لجانب الموظفين	29
ERD	30
هيكل تفصيلي	34
مخطط كلاس	35
بيانات قاعدة البيانات	37
واجهة تسجيل الدخول	38
الواجهة الرئيسية	38
واجهة العملاء	39
واجهة الحجوزات	39
واجهة العروض	40
واجهة الغرف	40
ADD Class	41
Accept Class	41
Main Class	42
Splach class	43
متحكم الغرف	43
Rooms Model	44
FoodRepo class	44
واجهات المستخدم	45
واجهات الغرف	46
واجهات الطلبات	46
واجهات تسجيل الدخول	47
conf.php	48

db.class.php	48
get_offers.php	49
log_in.php	50
اختبار get_offers	51
اختبار log_in	52

جدول المصطلحات

Term	Definition
API	Application programming interface: it is a way to connect the website/application with the database and GET/Post for it
Dashboard	Manage control
Gmail	Google mailing service
Gantt	One of the most popular and useful ways of showing activities
Flutter	is an open-source framework by Google
DB	Data Base
iOS	iPhone Operating System
PHP	Hypertext Preprocessor
MySQL	My Structured Query Language.

الفصل الأول

مقدمة

1.1 المقدمة:

بحثا عن طرق لتسويق وإدارة الفنادق يقوم نظامنا المنتج بتنظيم عملية الطلب عبر الانترنت وتوجيهها حسب اهتمامات المستخدم وجعلها أكثر اماناً وتجنبه عمليات الخداع وتضمن حقوقه وفق سياسة التطبيق ويضمن للمستخدم حرية انشاء حجز ويضمن للمدير بكل عمليات إدارة التي تتعلق بالفندق.

2.1 الأهمية:

سيقوم نظامنا بتشكيل قيمة فعلية من حيث:

1. توفير عدد كبير من المعاملات الورقية.
2. توفير جهد للموظف وللزبون.
3. المرونة.
4. سرعة عملية الحجز.
5. تقديم تقارير دقيقة والتخفيف من الخطأ قدر الامكان.
6. زيادة كفاءة الخدمات الفندقية.

حيث النظام الجديد سيقوم بمعالجة جميع المشاكل التي ابتلي بها النظام الحالي. سيقوم هذا النظام بتحليل وتخزين المعلومات أيضًا بشكل تلقائي أو تفاعلي. حيث سيتم تحليل البيانات لإنشاء التقارير.



الشكل (1) أهمية المشروع

3.1 الهدف من النظام:

لقد أصبحت التكنلوجيا شي أساسي في حياتنا اليومية، فأصبح من الضروري استثمارها من اجل راحتنا

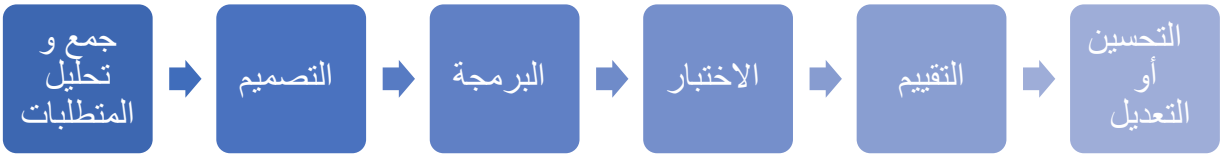
لذلك سيقوم نظامنا بجميع العمليات التي نحتاجها في الفنادق:

1. تقليل المعاملات الورقية.
2. تقليل الوقت الضائع في معالجة البيانات مع توفير بيانات دقيقة وتوقع للبيانات في بعض الاحيان.
3. الاحتفاظ بسجل دقيق لعمليات الحجز.
4. مساعدة اصحاب الفنادق في تطوير منتجاتهم عن طريق جمع تقييمات من العملاء.
5. انشاء منصة حجز عبر الانترنت.

من المؤكد ان هذا النظام مقارنة بنظام اليدوي، يوفر الكثير من العمل الورقي للفواتير ويؤمن معلومات غنية وأكثر دقة ويقلل من نسبة الخطأ التي من الممكن ان تحصل بسبب سوء تفاهم بين الفندق والزبون، حيث ان كل التفاصيل المنتجات موضحة وبدقة في تطبيق المستخدم والموقع.

4.1 منهجية التطوير:

طريقة العملية المتبعة في تطوير هذا النظام، هي منهجية التزايدية التكرارية الذي يتم فيه بناء النظام تدريجياً من خلال تكرارات (Iterations)، وكل تكرار ينتج جزءاً يعمل من النظام (Increment)، يتم تحسينه وإضافته في التكرارات اللاحقة حتى يكتمل النظام بالكامل.



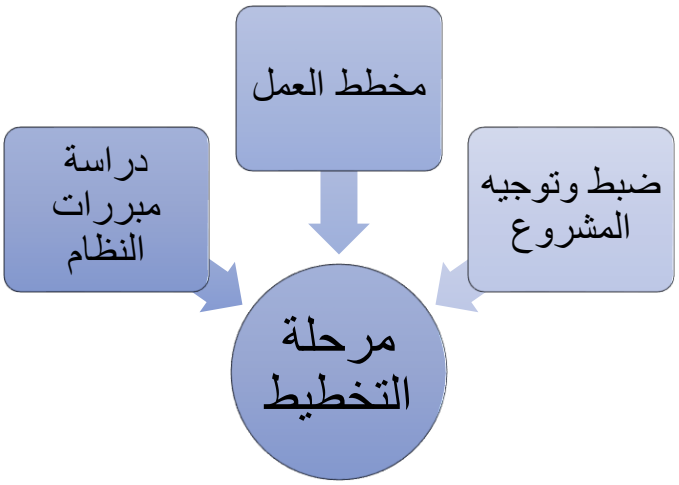
الشكل (2) منهجية الشلال

وأيضا منهجية ال clint server هي thick client هو نوع من البنية المعمارية للبرمجيات حيث :

الجزء الأكبر من منطق المعالجة و التطبيق يتم على جهاز المستخدم وليس على الخادم.

حيث سيتضمن هذا التقرير فصلاً لشرح كل مرحلة من المراحل السابقة وعرض مخرجاتها التي تعالج في المرحلة التالية. بالإضافة لمرحلة التخطيط التي تسبق جميع المراحل السابقة.

5.1 التخطيط:



الشكل (3) مرحلة التخطيط

1.5.1 دراسة مبررات النظام:

وثيقة طلب النظام ودراسة الجدوى:

1. الجدوى التقنية: التكنولوجيا اللازمة لتحقيق هذا النظام المحوسب متوفرة في السوق المحلي ويمكن شراؤها واستخدامها. وهي ستفيد في تسريع انجاز العمليات بسرعة ودقة ووثوقيه عالية.
2. الجدوى الاقتصادية: الزمن المحدد كافي لإنجاز هذا النظام وفق مخطط Ganttكما سيقوم هذا النظام المقترح بتوفير الوقت والجهد والمال بالعروض والحسومات المقدمة.
3. الجدوى التشغيلية: الكادر البشري المتوفر لإدارة وتشغيل النظام المقترح سيتمكن من اكتساب الخبرة اللازمة للتعامل مع النظام خلال فترة زمنية قصيرة عن طريق الأدوات التي سيتم تقديمها للمساعدة على فهم التعامل مع النظام مثل: مقطع مصور قصير يوضح طريقة الاستخدام لجميع مكونات النظام، كما انه سيوفر مجموعة التعليمات والتحذيرات والقيود الامر الذي سيمكن جميع أنواع المستفيدين من النظام من مستخدمين ومدراء وعمال توصيل من فهم التعامل مع النظام.
4. فرصة النجاح: يملك النظام فرصة عالية للنجاح والانتشار في السوق حيث أنه ازداد الطلب على هذا النوع من الخدمات مؤخراً، كما ان النظام قائم على دراسة الأنظمة المشابهة وإيجاد نقاط ضعفها وتحويلها لميزات إضافية في نظامنا مما سيجعله متميزا عن الأنظمة المشابهة الموجودة في السوق.

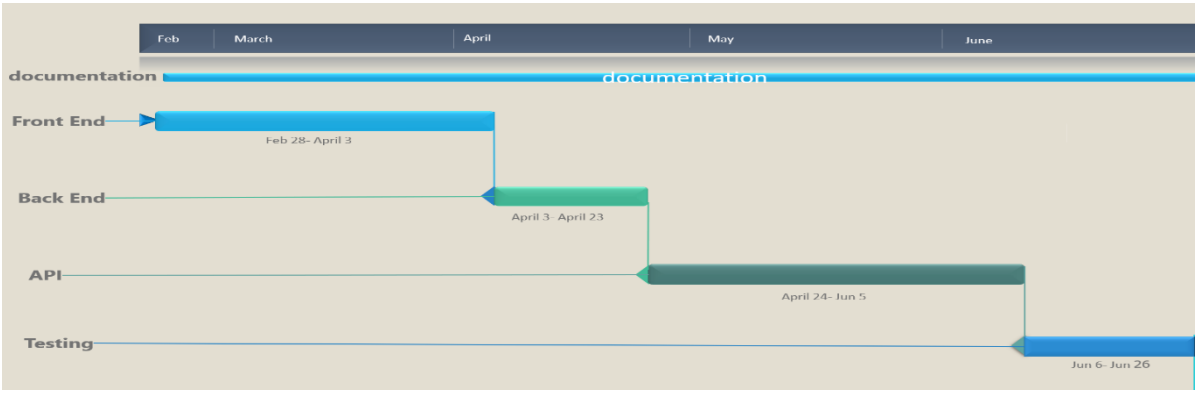
2.5.1 مخطط العمل:

حيث تم تحديد ما يجب إنجازه في المشروع الفصلي وفقا للمخطط التالي:



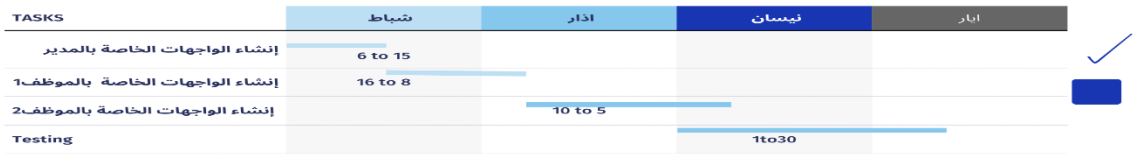
الشكل (4) مخطط Gantt فصلي

وكما تم اعتماد مخطط العمل التالي اثناء تنفيذ مشروع التخرج الأول:



الشكل (5) مخطط Gantt للتخرج 1

وكما تم اعتماد مخطط العمل التالي اثناء تنفيذ مشروع التخرج الثاني



الشكل (6) مخطط Gantt للتخرج 2

3.5.1 ضبط وتوجيه المشروع:

المعايير المتبعة في إنجاز المشروع:

1. انجاز العمل يتم وفق مهمات مقسمة بحيث كل مهمة لا تتجاوز الأسبوع.
2. أي إضافة او تعديل على المشروع يتطلب موافقة 2 من أعضاء الفريق على الأقل.
3. في حال كانت المهمة مخصصة لشخص ما من الفريق واستعصى عليه شيء، حينها نقوم بعقد اجتماع صغير للتوصل لحل يدعم المشروع، حيث ان وظيفتنا تعاونية –وليست تنافسية– فيما بخدم صالح المشروع.

تقييم المخاطر:

المخاطر المحتملة بشكل رئيسي هي مخاطر تنظيمية أو مخاطر متعلقة بالوقت لذلك حاولنا بشكل عام الالتزام بالمخطط الموضوع في مرحلة التخطيط مع تعديله وفقا للمدة الفعلية لكل مرحلة.

الفصل الثاني

جمع المتطلبات

1.2المعنيين بالنظام:

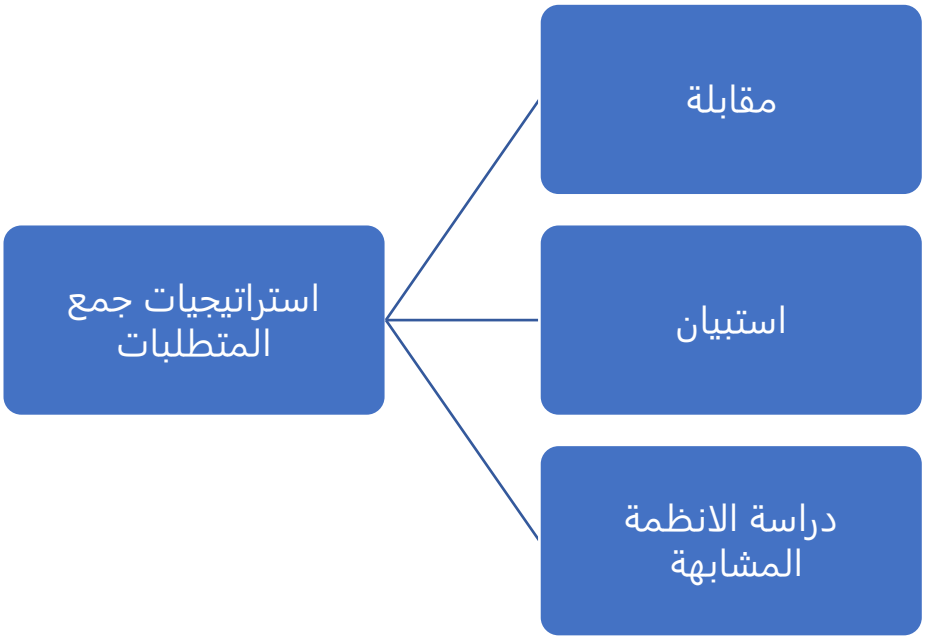
يمكن تقسيم مستخدمين النظام الى فئتين:

- الزبون: هو أي شخص يرغب في حجز غرفة في فندق.
- إدارة الفندق.

2.2 الطرائق المتبعة في جمع المتطلبات:

تم اجراء دراسة متعمقة لمعلومات المستخدم النهائي لإنتاج المتطلبات الوظيفية للنظام المقترح. حيث تم جمع البيانات حول نظام الطلب الحالي من خلال العديد من عمليات تقصي الحقائق ونتيجة لذلك تم تحديد متطلبات المستخدم للنظام المقترح في نهاية هذه المرحلة. كما انه تم إجراء دراسة على النظام الحالي بناءً على البيانات التي تم جمعها، حيث انها نقاط معينة قمنا بمناقشتها مع المستخدمين النهائيين لتقرير حالتها اي إذا كانت سلبيات تؤثر فعليا على خدمة المستخدم.

وكانت استراتيجيات جمع المتطلبات الاولية كالتالي:



الشكل (6) استراتيجيات جمع المتطلبات

1.2.2 الدراسة المرجعية للأنظمة المشابهة:

من اجل الحصول على أفضل تطبيق وتجنب الاخطاء الشائعة قمنا بدراسة عدة تطبيقات مثل

Trivago:

يتيح لك الموقع بحجز غرفة في الفنادق المشتركة معه:

- الايجابيات:
1. التطبيق لا يقتصر على الفنادق انما يتيح لك حجز شقق وفيلات
2. السرعة والمرونة
3. سهولة استخدام
4. توفر العروض والحسومات
- السلبيات:

غير مدعوم في سوريا

وأيضا قمنا ب دراسة مرجعية لموقع Booking:

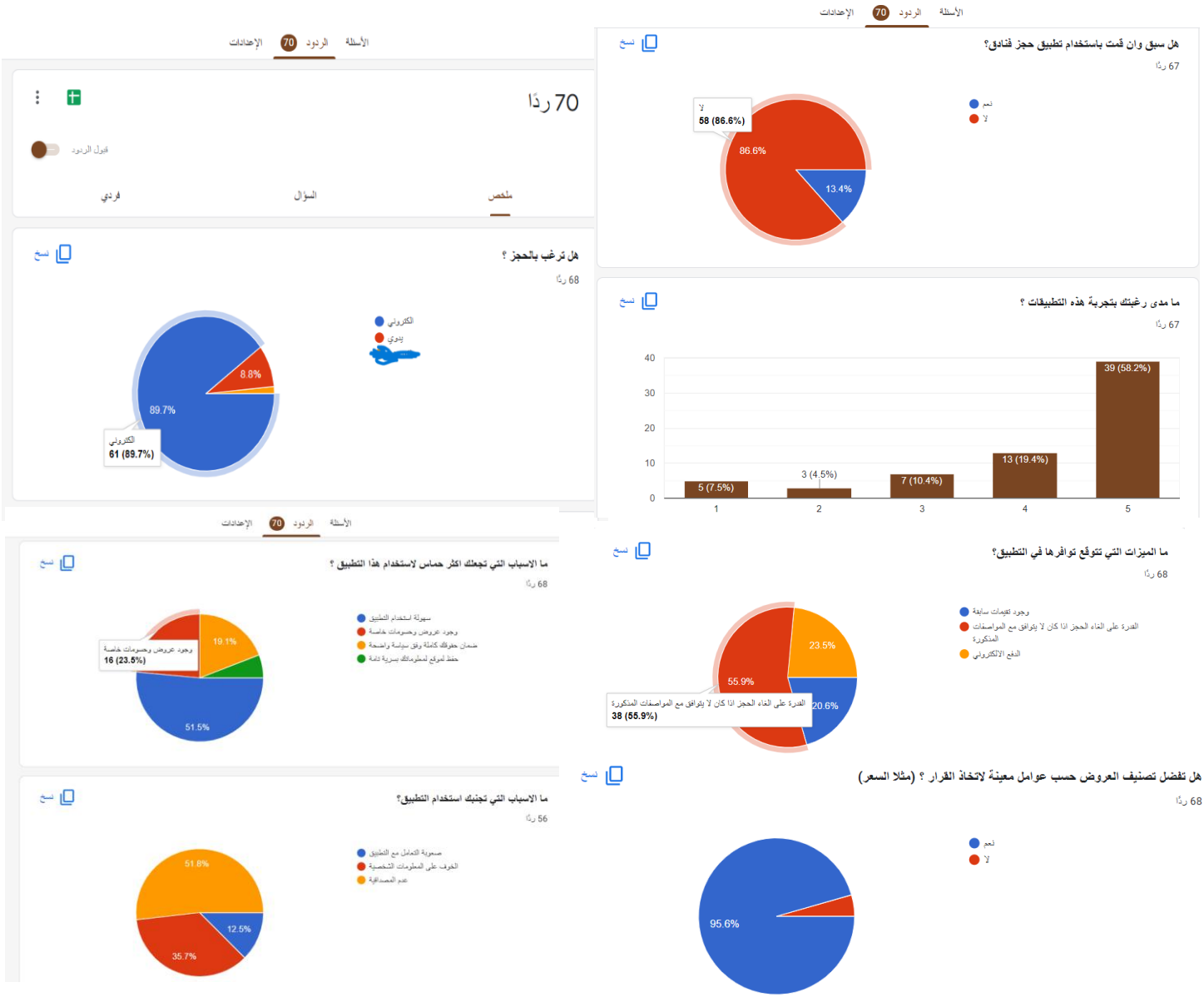
هو موقع رائع يسمح لك بالحجز في أي فندق:

- الإيجابيات
1. سهولة التعامل مع الموقع
2. السرعة
3. الأمان على المعلومات
4. وجود العروض الجيدة والتي تناسب الجميع

اعتمادا على هذه الدراسة قمنا بدراسة العوامل السابقة لإضافة الإيجابيات على نظامنا وإيجاد حلول للتعامل مع السلبيات التي ستشكل قيمة مضافة الى المشروع.

2.2.2 الاستبيان:

من نتائج الدراسة السابقة انشأنا استبياناً للمستخدمين الحاليين يدرس احتياجاتهم ويمكننا من اجراء التعديلات وتطوير المقترحات اللازمة عليه واستبياناً اخر للمستخدمين الجدد المقبلين على استخدام النظام لمعرفة ما الذي يتوقعون الحصول عليه وايجاده في نظامنا الجديد والأسباب التي من الممكن ان تجعل التعامل مع النظام صعبا بالنسبة لهم.



الشكل (7) الاستبيان

3.2.2 المقابلة:

استخدمنا المقابلة لاستخلاص المتطلبات الأكثر استخداماً، تم تحديد الأشخاص الذي ستتم مقابلتهم بناءً على المنصب الإداري الذي يغطيه نظامنا، حيث تم الحصول على حقائق موثوقة بناءً على الأسئلة التي تم طرحها على المتقابلين والتي ساعدت في بدء العمل كما ساعدت ايضا في مجال عرض الحلول من التصميم

الاسم	الموقع	
انوار محمد	مدير فندق فينيسيا	القدرة على تصور الخدمات التي يرغب المدير باستيفائها ومعرفة الية عمل النظام الحالي ومشاكله.
محمد علي	موظف استقبال فندق البستان	القدرة على تصور الخدمات التي يرغب موظف الاستقبال والمشاكل التي تواجهه.

الجدول (1) الأشخاص الذين تم مقابلتهم

ملخص المقابلة:

يرغب المدير توافر الميزات التالية في النظام الذي سيفضل استخدامه فندقه:

1. إضافة وحذف موظف.
2. الاطلاع على عمليات الحجز.
3. ان يرسل الموقع تقارير يومية.
4. تقارير مالية شاملة لكل يوم.
5. رؤية التقييم.

اما موظف الاستقبال:

1. تثبيت الحجز.
2. مراقبة حالة الغرف.
3. شكاوى النزلاء.
4. الإصلاحات المطلوبة.
5. انشاء حساب للعميل يكون متكامل وبسيط.
6. انشاء قائمة أسعار للغرف وفقا لنوع المصدر.
7. تغير الغرفة لزبون.
8. مراقبة مخزون المطعم، وادراج الوجبات ضمن واجهة المطعم.
9. تسويات النزلاء: تختلف وفقا للوضع كالدفع النقدي، وبطاقات الائتمان، كذلك نزلاء الفندق حسب رقم الغرفة وعرض الاسم الشاشة لطباعتها على الفاتورة.

3.2 المتطلبات الوظيفية:

ومن تحليل بيانات المقابلات السابقة تبين انه من **المتطلبات الوظيفية** التي يجب تأمينها في DASHBOARDالمدراء:

مدير الفندق:

1) القدرة على قبول الطلبات
2) القدرة على إضافة موظفين او التعديل على الموظفين الموجودين مسبقاً
3) القدرة على إضافة العروض والحسومات بشكل مباشر
4) استلام تقارير من النظام توضح:
   - نسبة الأرباح
   - نسبة توزع الطلبات حسب الأماكن لتوجيه العروض
   - أوقات ذروة الطلبات لتجهيز كوادر إضافية في هذه الأوقات
   - تقييمات الفندق من المستخدمين
5) القدرة على طباعة التقارير مباشرة

موظف الاستقبال:

1. القدرة على طرد المسيئين من التطبيق
2. رؤية العملاء الحاليين داخل الفندق
3. اضافة الغرف داخل الفندق
4. تأكيد حجوزات العملاء
5. يقوم بارسال الاشعارات للموظفين

موظف خدمة الغرف:

1. يقوم بإضافة المطاعم والمتاجر المتواجدة داخل الفندق
2. مسؤول عن استقبال طلبات تنظيف الغرف والطعام
3. يقوم بإضافة الخدمات التي يقدمها الفندق

الزبون:

1) تصفح البروفايل
2) تسجيل الدخول
3) مشاهدة العروض والغرف لمتاحة
4) حجز والغاء غرفة
5) تقيم الفندق
6) طلب طعام

سيكون للنظام المقترح أيضًا بعض الميزات الأخرى مثل:

1. الدقة في معالجة البيانات
2. سيتم تقليل حجم العمل الورقي بشكل كبير.
3. سرعة عمليات وتسليم في الوقت المناسب.
4. المرونة (أي يمكن الوصول إليها في أي وقت)
5. طريقة سهلة لإجراء نسخ احتياطي أو نسخ البيانات الموجودة على الأقراص المضغوطة في حالة فقد البيانات
6. تخزين أفضل ونظام استرجاع أسرع
7. سيتم التقليل إلى حد كبير من الأخطاء في التقارير

حيث أن الغرض من النظام الجديد هو معالجة جميع المشاكل التي ابتلي بها النظام الحالي. سيقوم هذا النظام بتحليل وتخزين المعلومات أيضًا بشكل تلقائي أو تفاعلي. حيث سيتم تحليل البيانات لإنشاء التقارير المطلوبة بحيث تتوافق مع المعلومات المحددة التي تحتاجها

4.2 مصفوفة تتبع المتطلبات:

ID	وصف المتطلب	الدور المسؤول	الكود	التصميم	الاختبار	ملاحظات
REQ-001	نظام قبول الطلبات	مدير الفندق	#copo	FF1	S1	يجب دمج نظام الدفع
REQ-002	إضافة وتعديل الموظفين	مدير الفندق	#copo	FF1	S12	يتطلب نظام صلاحيات متكامل
REQ-003	نظام العروض والحسومات	مدير الفندق	#copo	FF1	S13	مرتبط بنظام التقارير
REQ-004	تقرير نسبة الأرباح	مدير الفندق	#copo	FF1	S13	يتطلب تحليل بيانات مالي
REQ-005	توزيع الطلبات حسب الموقع	مدير الفندق	#copo	FF1	S1	مرتبط مع نظام العروض
REQ-006	تحليل أوقات الذروة	مدير الفندق	#copo	FF1	S1	يتطلب تحليل البيانات التاريخية
REQ-007	نظام تقييمات العملاء	مدير الفندق	#copo	FF1	S1	مرتبط مع نظام الحجوزات
REQ-008	طباعة التقارير	مدير الفندق	#copo	FF1	S1	يتطلب تنسيقات مخصصة
REQ-009	إدارة المسيئين	موظف الاستقبال	#copo	FF2	S2	يتطلب مراجعة إدارية
REQ-010	عرض العملاء الحاليين	موظف الاستقبال	#copo	FF2	S2	مرتبط بنظام الحجوزات
REQ-011	إضافة وتحديث الغرف	موظف الاستقبال	#copo	FF2	S2	يتطلب تحديث الأسعار
REQ-012	تأكيد الحجوزات	موظف الاستقبال	#copo	FF2	S2	يتطلب تأكيد الدفع

REQ-013	نظام الإشعارات	موظف الاستقبال	#copo	FF2	S2	يتطلب دمج البريد الإلكتروني والرسائل النصية
REQ-014	إدارة المطاعم والمتاجر	خدمة الغرف	#copo	FF2	S3	يتطلب قاعدة بيانات منفصلة
REQ-015	نظام طلبات التنظيف	خدمة الغرف	#copo	FF2	S3	يتطلب جدولة زمنية
REQ-016	نظام الطلبات الغذائية	خدمة الغرف	#copo	FF2	S3	يتطلب دمج نظام الطهي
REQ-017	إضافة خدمات جديدة	خدمة الغرف	#copo	FF2	S3	مرتبط بنظام العروض
REQ-018	نظام تسجيل الدخول	العميل	#copo	FF3	S4	يتطلب تأمين البيانات
REQ-019	عرض العروض والغرف المتاحة	العميل	#copo	FF3	S4	يتطلب تحديث فوري للبيانات
REQ-020	نظام الحجز والإلغاء	العميل	#copo	FF3	S4	يتطلب سياسات الإلغاء
REQ-021	نظام التقييم	العميل	#copo	FF3	S4	يتطلب التحقق من صحة التقييمات
REQ-022	نظام الطلبات الغذائية	العميل	#copo	FF3	S4	يتطلب دمج قوائم الطعام

5.2 المتطلبات الغير وظيفية :

التشغيلية:

- تطبيق الخاص بالزبون وتطبيق المستخدم النهائي يعمل على Android .
- نظام الفندق ينبغي ان يكون قادرا على العمل في أي متصفح ويب.

الأداء:

- التفاعل بين المستخدم والنظام ينبغي الا يتجاوز 10 ثانية.
- يتم حفظ أي تعديل يتم على البيانات في ال DB خلال مدة أقصاها 15 ثوان باسوا حالة.
-

الامن:

- الحفاظ على سرية بيانات المستخدمين المدخلة.
- لا يمكن وصول عامل توصيل الى معلومات عامل توصيل آخر.
- فقط مدير النظام ا له الحق بالموافقة على الغاءاتعيين طلب الزبون.

الثقافية والسياسية:

- يستطيع النظام التعرف على عملات مختلفة.
- المعلومات الشخصية محمية وفقا لـ "قانون حماية البيانات".
- النظام يتوافق مع قانون الدولة لحماية المستهلك.

الفصل الثالث

مرحلة التحليل

تم في هذه المرحلة البدء في اتمتة إجرائية العمل حيث تم تحديد المستفيدين من النظام بصورة ادق وتمثيل المتطلبات المستخرجة من المرحلة السابقة في مخطط حالات الاستخدام Use case.

1.3مستخدمي النظام:

مدير الفندق
موظف الاستقبال
موظف خدمة الغرف
العميل

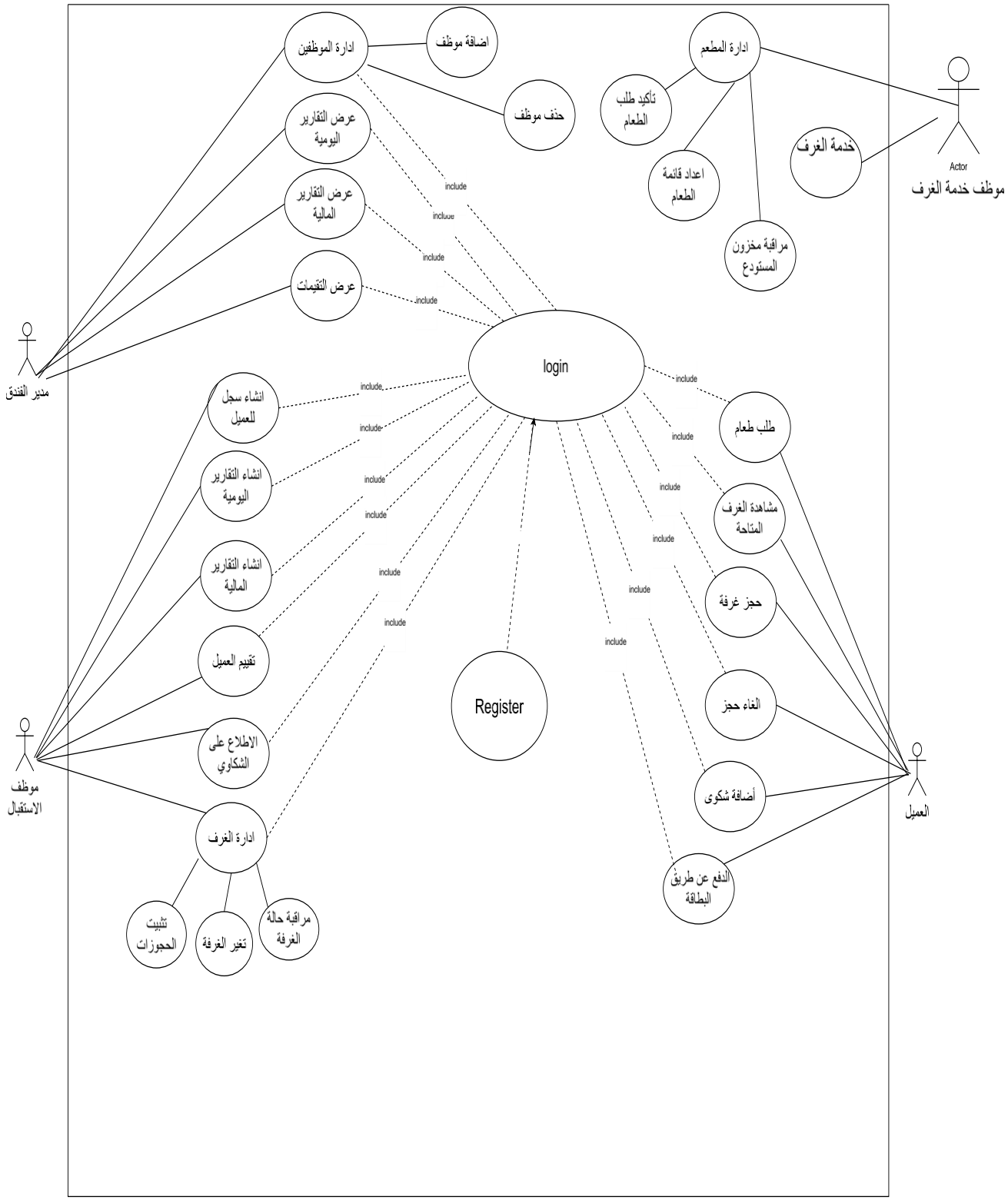
الجدول (2) مستخدمي النظام

2.3 تصنيف حالات الاستخدام

حالة الاستخدام	الممثلون	الهدف
إضافة موظفين	مدير الفندق	إعطاء الموظفين صلاحيات
طلب غرفة	الزبون	تكوين طلب من اجل الحصول على غرفة
إضافة عروض جديدة	مدير الفندق	إضافة عروض وتعديل على مواصفاتها واسعارها
تأكيد الطلبات	موظف الاستقبال	قبول الطلب

الجدول (3) تصنيف حالات الاستخدام

3.3 مخطط حالات الاستخدام وتوصيفه:



الشكل (8) مخطط حالات الاستخدام

4.3 التوصيف:

حالات استخدام مدير الفندق:

Use case name: قبول الطلبات
ID: REQ-001
الوصف: نظام متكامل لإدارة حجوزات العملاء
الممثل الرئيسي: مدير الفندق
الشروط المسبقة: يجب ان يكون المدير مسجل مسبقا
الملاحظات: يتطلب التحقق من توفر الغرف والأسعار وتأكيد الحجز مع العميل
الوضع: نشط

Use case name: إدارة الموظفين
ID: REQ-002
الوصف: نظام لإدارة بيانات وتعيينات الموظفين
الممثل الرئيسي: مدير الفندق
الشروط المسبقة: يجب ان يكون المدير مسجل مسبقا
الملاحظات: يتطلب تحديد صلاحيات الموظف وتخصيص دورته
الوضع: نشط

Use case name: إضافة العروض والحسومات
ID: REQ-003
الوصف: نظام لإدارة وتطبيق العروض الترويجية
الممثل الرئيسي: مدير الفندق
الشروط المسبقة: يجب ان يكون المدير مسجل مسبقا
الملاحظات: يتطلب تحديد شروط العرض وأسعار الحسومات
الوضع: نشط

Use case name : استلام التقارير
ID: REQ-003
الوصف: نظام متكامل لتوليد وتحليل التقارير الإدارية
الممثل الرئيسي: مدير الفندق
الشروط المسبقة: يجب ان يكون المدير مسجل مسبقا
الملاحظات: يتطلب تصفية حسب الفترة الزمنية والفئات
الوضع: نشط

Use case name :استلام التقارير
ID: REQ-008
الوصف: نظام لطباعة التقارير المطلوبة
الممثل الرئيسي: مدير الفندق
الشروط المسبقة: يجب ان يكون المدير مسجل مسبقا
الملاحظات: يتطلب خيارات تنسيق متعددة
الوضع: نشط

Use case name :استلام التقارير
ID: REQ-008
الوصف: نظام لطباعة التقارير المطلوبة
الممثل الرئيسي: مدير الفندق
الشروط المسبقة: يجب ان يكون المدير مسجل مسبقا
الملاحظات: يتطلب خيارات تنسيق متعددة
الوضع: نشط

حالات استخدام موظف الاستقبال:

Use case name : إدارة المسيئين
ID: REQ-009
الوصف: نظام للتحكم في وصول العملاء غير المرغوب فيهم
الممثل الرئيسي: موظف الاستقبال
الشروط المسبقة: يجب ان يكون موظف الاستقبال مسجل مسبقا
الملاحظات: يتطلب توثيق الأسباب والتواريخ
الوضع: نشط

Use case name : عرض العملاء الحاليين
ID: REQ-010
الوصف: نظام تتبع وجود العملاء في الفندق
الممثل الرئيسي: موظف الاستقبال
الشروط المسبقة: يجب ان يكون موظف الاستقبال مسجل مسبقا
الملاحظات: يتطلب تصفية حسب حالة الإقامة
الوضع: نشط

Use case name : إضافة الغرف
ID: REQ-011
الوصف: نظام لإدارة بيانات الغرف المتاحة
الممثل الرئيسي: موظف الاستقبال
الشروط المسبقة: يجب ان يكون موظف الاستقبال مسجل مسبقا
الملاحظات: يتطلب تحديث الأسعار والوصف
الوضع: نشط

Use case name : تأكيد الحجوزات
ID: REQ-012
الوصف: نظام لتأكيد حجوزات العملاء
الممثل الرئيسي: موظف الاستقبال
الشروط المسبقة: يجب ان يكون موظف الاستقبال مسجل مسبقا
الملاحظات: يتطلب تحديث حالة الغرفة
الوضع: نشط

Use case name : إرسال الإشعارات
ID: REQ-013
الوصف: نظام لإرسال التنبيهات والإشعارات للموظفين
الممثل الرئيسي: موظف الاستقبال
الشروط المسبقة: يجب ان يكون موظف الاستقبال مسجل مسبقا
الملاحظات: يتطلب اختيار المستلمين والمحتوى
الوضع: نشط

حالات استخدام موظف خدمة الغرف:

Use case name : إضافة المطاعم والمتاجر
ID: REQ-014
الوصف: نظام لإدارة الموردين والخدمات داخل الفندق
الممثل الرئيسي: موظف خدمة الغرف
الشروط المسبقة: يجب ان يكون موظف خدمة الغرف مسجل مسبقا
الملاحظات: يتطلب تحديث قوائم الخدمات
الوضع: نشط

Use case name : طلبات التنظيف
ID: REQ-015
الوصف: نظام لتتبع طلبات تنظيف الغرف
الممثل الرئيسي: موظف خدمة الغرف
الشروط المسبقة: يجب ان يكون موظف خدمة الغرف مسجل مسبقا
الملاحظات: يتطلب جدولة المواعيد
الوضع: نشط

Use case name : طلبات الطعام
ID: REQ-016
الوصف: نظام لتتبع طلبات الطعام
الممثل الرئيسي: موظف خدمة الغرف
الشروط المسبقة: يجب ان يكون موظف خدمة الغرف مسجل مسبقا
الملاحظات: يتطلب تحديد وقت التوصيل
الوضع: نشط

Use case name : إضافة الخدمات
ID: REQ-017
الوصف: نظام لإدارة الخدمات المقدمة
الممثل الرئيسي: موظف خدمة الغرف
الشروط المسبقة: يجب ان يكون موظف خدمة الغرف مسجل مسبقا
الملاحظات: يتطلب تحديث الأسعار والوصف
الوضع: نشط

حالات استخدام الزبون:

Use case name : تصفح البروفايل
ID: REQ-018
الوصف: نظام لعرض وتحديث معلومات الحساب الشخصي
الممثل الرئيسي: الزبون
الشروط المسبقة: يجب ان يكون الزبون مسجل مسبقا
الملاحظات: يتطلب تسجيل الدخول
الوضع: نشط

Use case name : تسجيل الدخول
ID: REQ-018
الوصف: نظام للتحقق من هوية الزبون
الممثل الرئيسي: الزبون
الشروط المسبقة: يجب ان يكون الزبون مسجل مسبقا
الملاحظات: يتطلب التحقق من البيانات
الوضع: نشط

Use case name : عرض العروض والغرف
ID: REQ-019
الوصف: نظام لعرض العروض والغرف المتاحة
الممثل الرئيسي: الزبون
الشروط المسبقة: يجب ان يكون الزبون مسجل مسبقا
الملاحظات: يتطلب تصفية حسب التفضيلات
الوضع: نشط

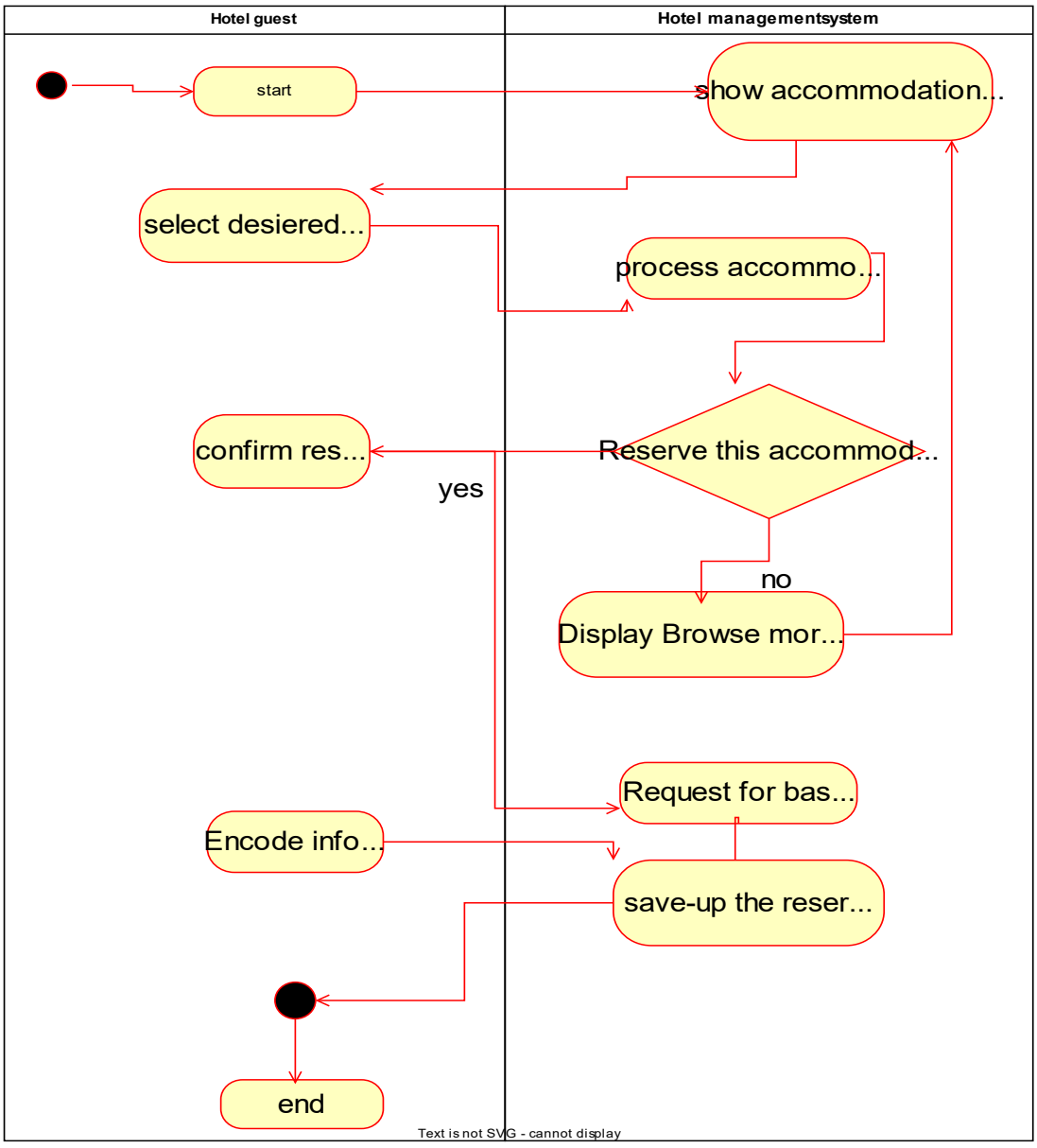
Use case name : حجز الغرفة
ID: REQ-020
الوصف: نظام لحجز الغرف وإلغائها
الممثل الرئيسي: الزبون
الشروط المسبقة: يجب ان يكون الزبون مسجل مسبقا
الملاحظات: يتطلب تأكيد الدفع
الوضع: نشط

Use case name : تقييم الفندق
ID: REQ-021
الوصف: نظام لتقييم تجربة الزبون
الممثل الرئيسي: الزبون
الشروط المسبقة: يجب ان يكون الزبون مسجل مسبقا
الملاحظات: يتطلب حجز سابق
الوضع: نشط

Use case name : طلب الطعام
ID: REQ-022
الوصف: نظام لطلب وتتبع الطعام
الممثل الرئيسي: الزبون
الشروط المسبقة: يجب ان يكون الزبون مسجل مسبقا
الملاحظات: يتطلب تحديد الوقت والكمية
الوضع: نشط

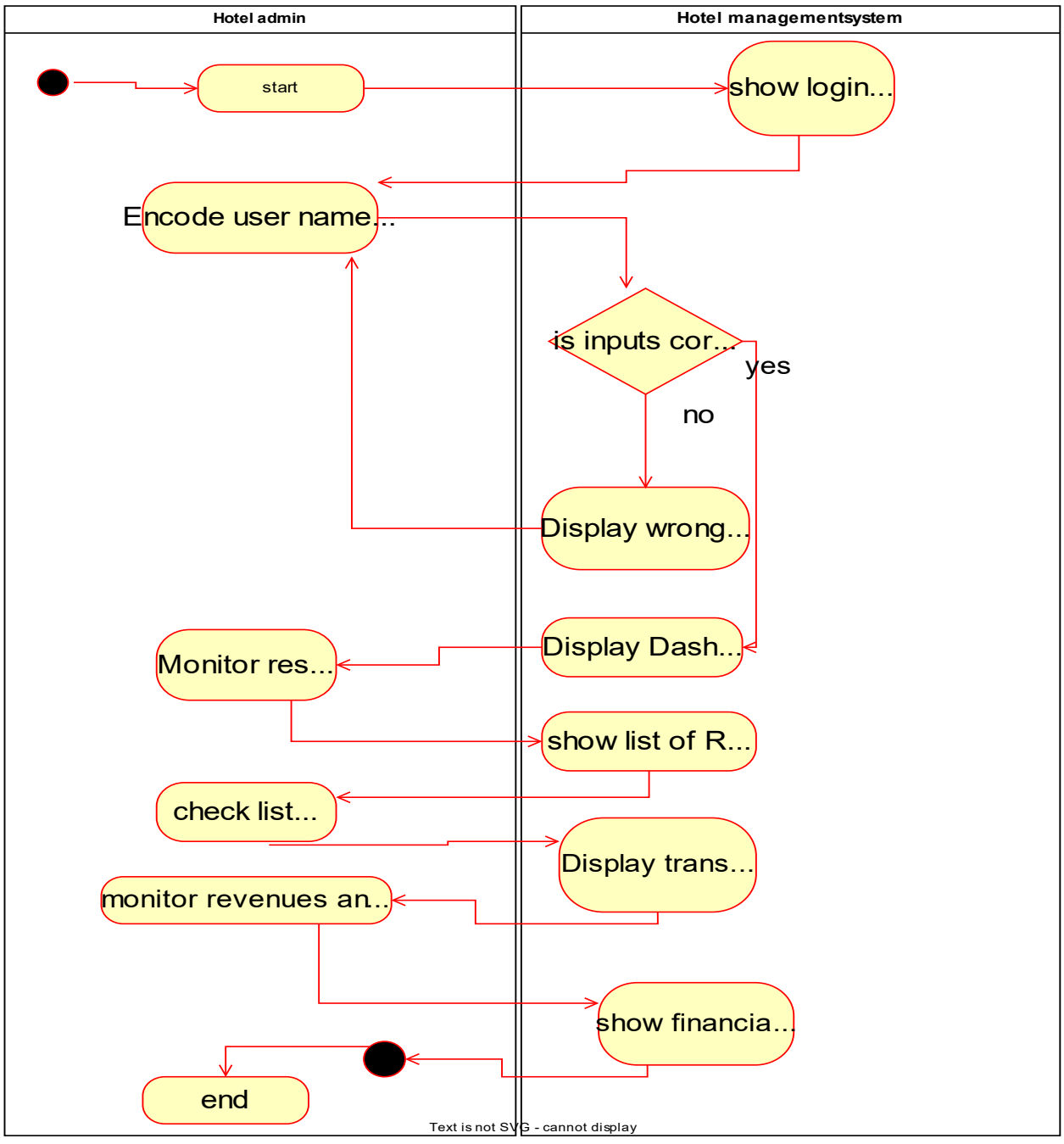
5.3 مخطط Activity Diagram:

مخطط نشاط نظام إدارة الفندق:



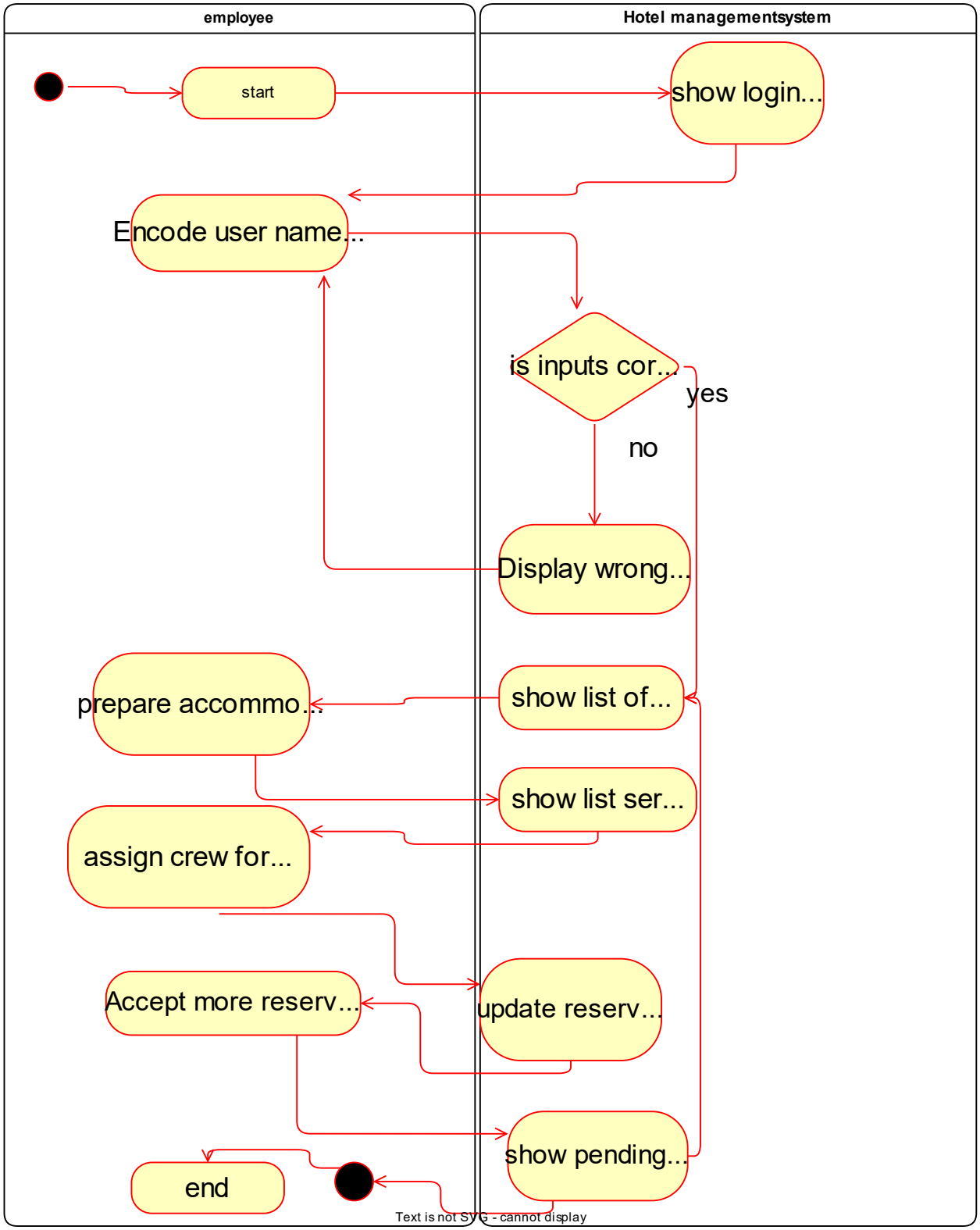
الشكل (9) مخطط نشاط نظام إدارة الفندق

مخطط نشاط نظام إدارة الفنادق لجانب مسؤول الفندق:



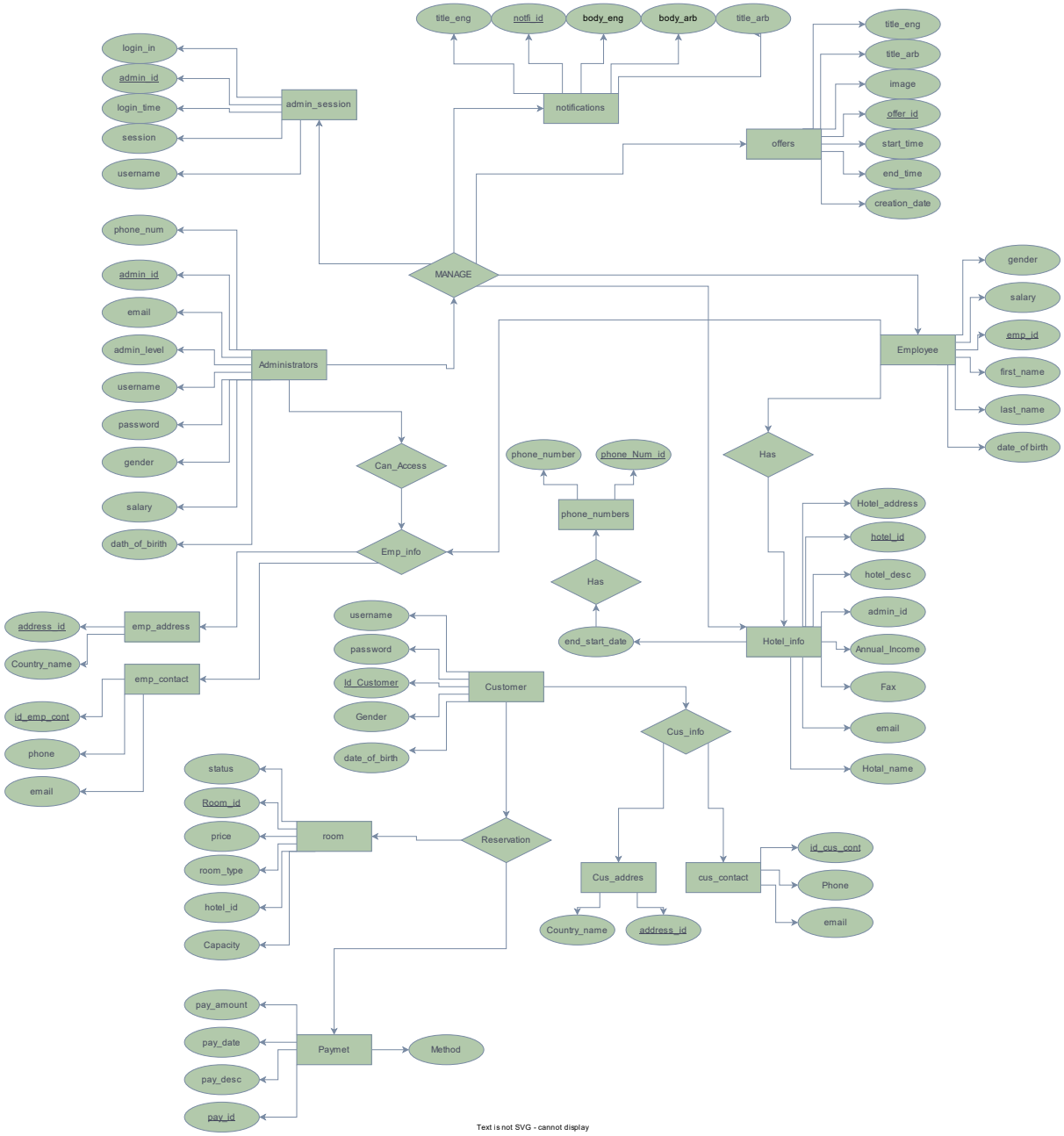
الشكل (10) مخطط نشاط نظام إدارة الفنادق لجانب مسؤول الفندق

مخطط نشاط نظام إدارة الفنادق لجانب الموظفين:



الشكل (11) مخطط نشاط نظام إدارة الفنادق لجانب الموظفين

6.3 مخطط العلاقات بين الكيانات Entity Relationship Diagram :



الشكل (12) ERD

الفصل الرابع

مرحلة التصميم

1.4 تصميم النظام الفيزيائي:

حيث تمت مناقشة مكونات النظام الأساسية وطريقة تطويرها وتم التوصل إلى أنه سيتم تطوير كافة عناصر النظام عدا:

- API الخاص بالبنك المستخدم للتأكد من صحة البطاقة البنكية.

2.4 التصميم المعماري:

سيتألف النظام المنشأ من:

لوحة تحكم (Dashboard): تمكن المدراء من الوصول ل صلاحياتهم المحددة وتعديل المعلومات وفقا ل رغباتهم

تطبيق: تطبيق للعميل يتم من خلاله عمليات استعراض الغرف والحجز الالكتروني

صفحة الويب: يتم من خلالها عرض معلومات وتعريف بنظامنا للمستهلك والمنشآت، كما يتم توفير آخر تحديثات التطبيقات للتحميل

قاعدة بيانات: توضع على سيرفر لتقوم بالربط بين جميع مكونات النظام وتحوي جميع البيانات المتناقلة عبر نظامنا.

3.4 الأدوات المستخدمة في تطوير عناصر النظام:

تمت مناقشة أساليب التطوير وتحديد خيارات الأدوات لاختيار الادوات الأكثر كفاءة ومرونة ليتم استخدامها حيث تم وضع العديد من لغات البرمجة والادوات في الاعتبار في تصميم هذا النظام.

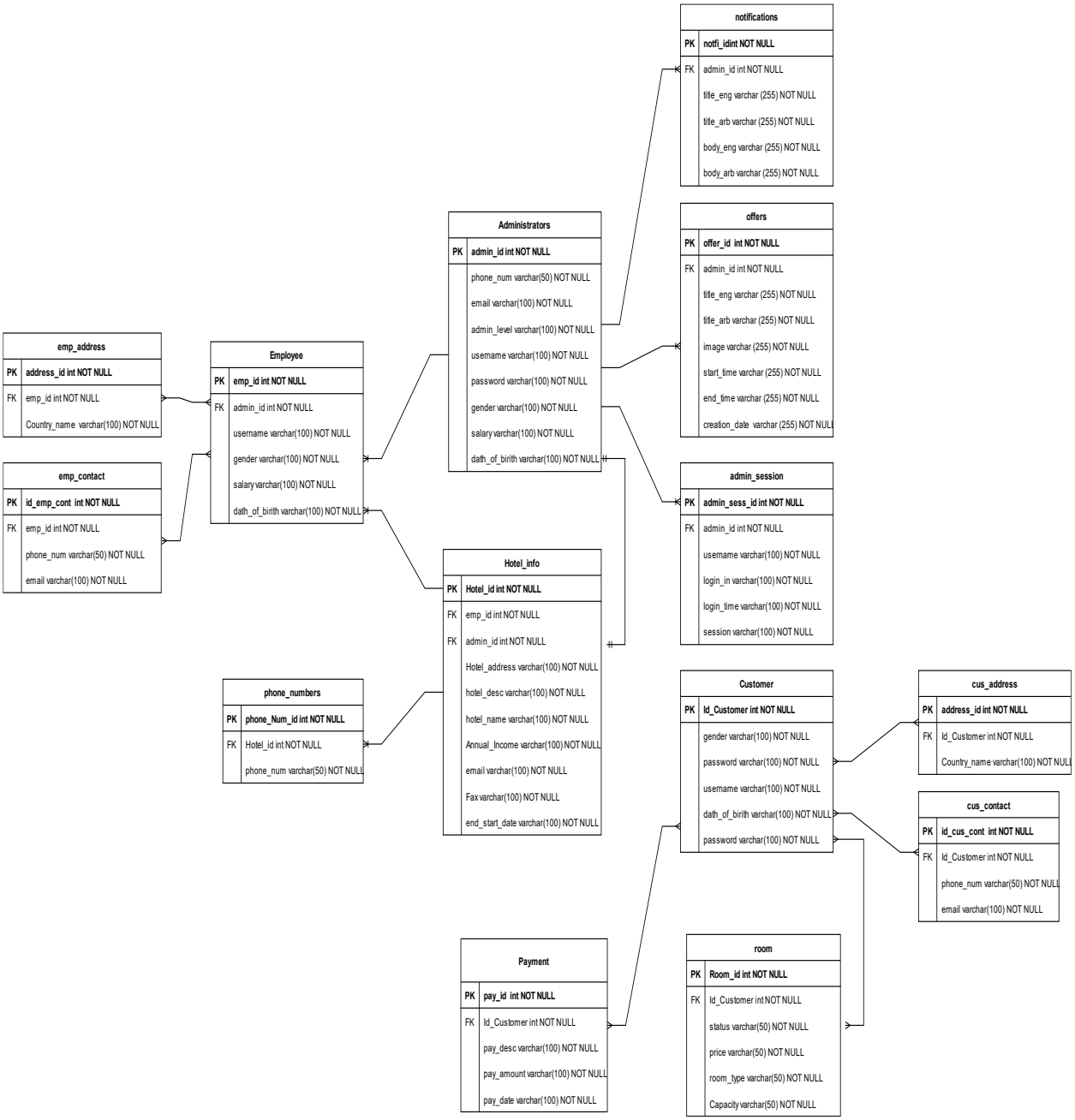
تم أيضًا أخذ الكثير من العوامل والتي تشمل الوصول إلى قاعدة البيانات عبر الإنترنت، ونقل البيانات عبر الشبكات، واسترجاع قاعدة البيانات عبر الإنترنت،

التقاط البيانات عبر الإنترنت، وأمان قاعدة بيانات الوصول إلى شبكة متعددة المستخدمين..

وبناءً على ذلك تم اختيار الادوات كالتالي:

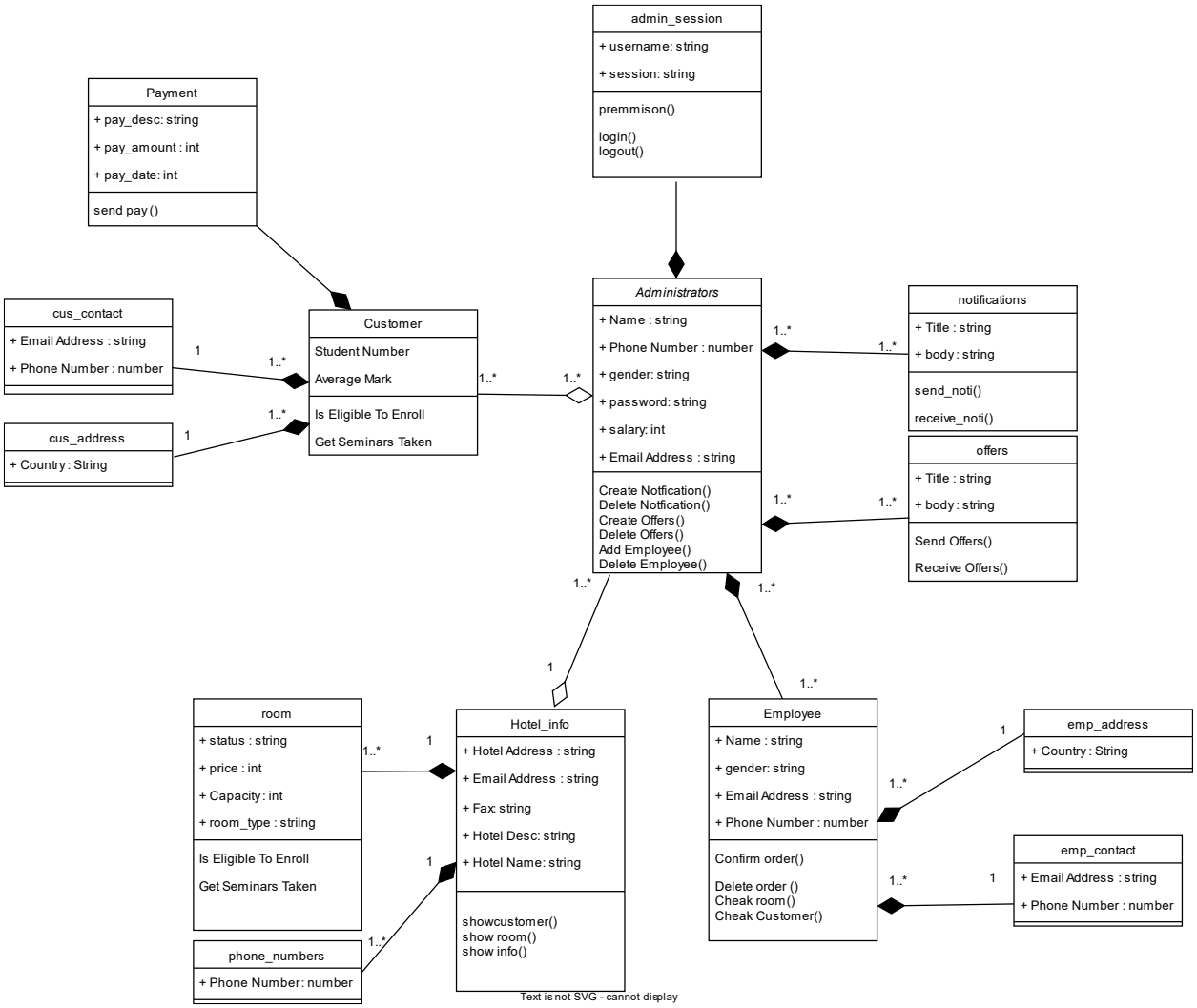
1. نظام قاعدة البيانات المستخدم لهذا النظام هو MySQL.
    1) قاعدة بيانات MySQL هي قاعدة بيانات قوية ي
    2) كما أنها تضمن سلامة قاعدة البيانات والبيانات.
    3) وتضمن حماية واستيعاب قاعدة بيانات كبيرة.
    4) تم الوصول إلى النظام من خلال (phpMyAdmin).
    5) يعد phpMyAdmin سهل الاستخدام ويمكن تعديله برمجيًا.
2. وقد تم الوصول إلى أن استخدام Bootstrap،PHP ، JavaScript يعد الخيار أمثل لسهولة التعامل معهم ولخبرة فريق العمل بهم.
3. ولبناء تطبيق الموبايل تم اعتماد إطار العمل Flutter لملائمته لطبيعة المشروع وسهولة استخدامه وخبرة فريق العمل به.

4.4 الهيكل التفصيلي لكل جدول في قاعدة البيانات:



الشكل (13) الهيكل التفصيلي

5.4 مخطط Class Diagram:



الشكل (14) مخطط كلاس

الفصل الخامس

التطبيق والاختبار

1.5 مرحلة التطبيق:

1.1.5 قاعدة البيانات:

تم اعتماد الأدوات المقترحة من مرحلة التصميم وتم البدء بإنشاء قاعدة البيانات عن طريق "MYSQL" وفقاً للمخطط العلائقي من مرحلة التصميم.

وتم حقنها ببيانات مؤقتة للاختبار.

Containing the word:

	Table	Action	Rows	Type	Collation	Size	Overhead
☐	administrators	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	admin_sessions	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	customer	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	info	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	notifications	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	offers	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
	6 tables	Sum	7	InnoDB	utf8mb4_general_ci	96.0 KiB	0 B

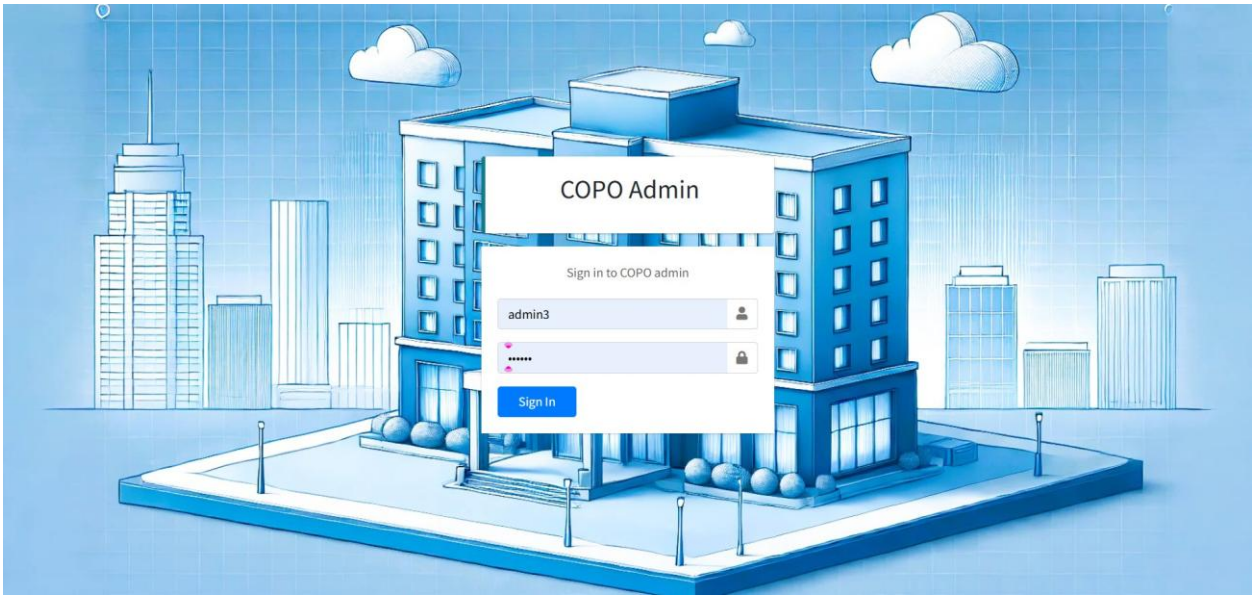
الشكل (15) البيانات في القاعدة

وبعدها تم تصنيف الواجهات المتاحة للمدير الرئيسي ل ملفات ب أسمائها وتم وضع التفاعلات المتاحة لكل أدمن مع هذه الواجهة ضمن الملف وبذلك نحقق إعادة الاستخدام لبعض المكونات المتشابهة ضمن الواجهات

كما تم ضمان حماية المعلومات عن طريق تشفير كلمات السر والمعلومات المهمة قبل حقنها في قاعدة البيانات

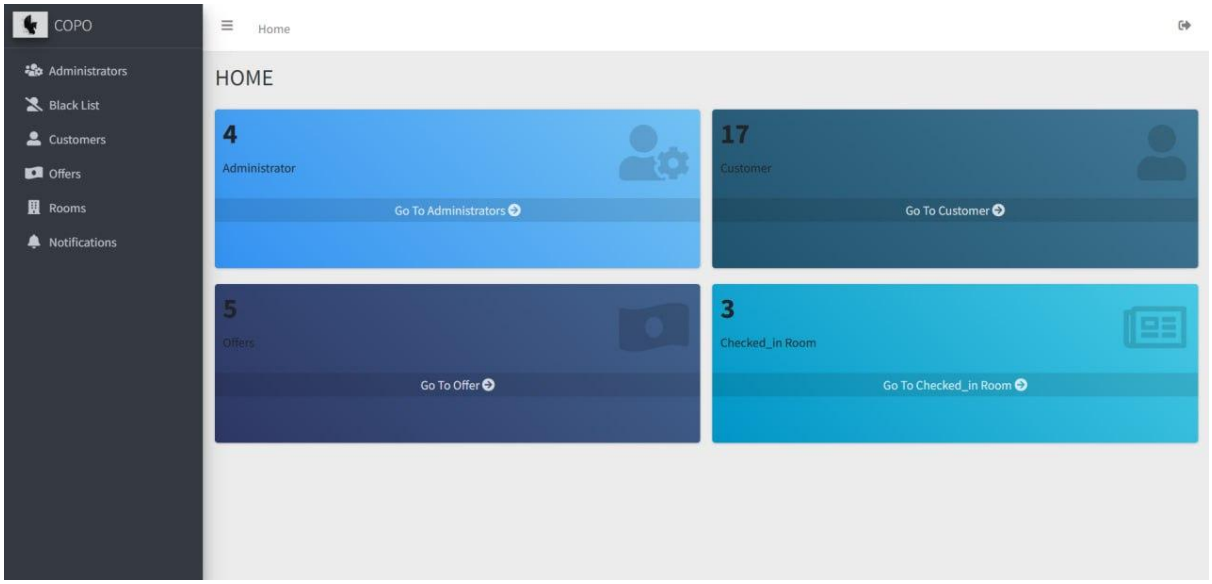
2.1.5 الواجهات الرئيسية في النظام:

تم اتباع منهجية Material Design system من غوغل حيث تقدم العديد من الأدوات التي تساعد في عملية التصميم بالإضافة إلى تقديم إرشادات حول اختبار نوع وحجم الخطوط والألوان المستخدمة من أجل تقديم أفضل تجربة للمستخدم.



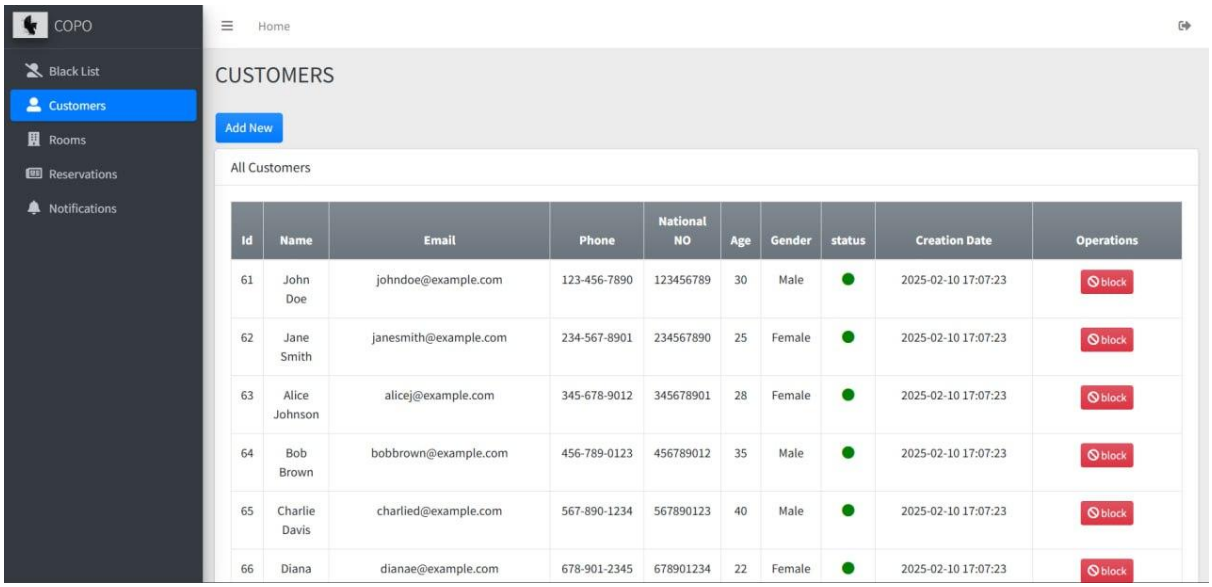
الشكل (16) واجهة تسجيل دخول الأدمن

واجهة مدير الفندق الرئيسية التي تمكنه من مشاهدة العملاء والموظفين والعروض الحالية داخل الفندق :



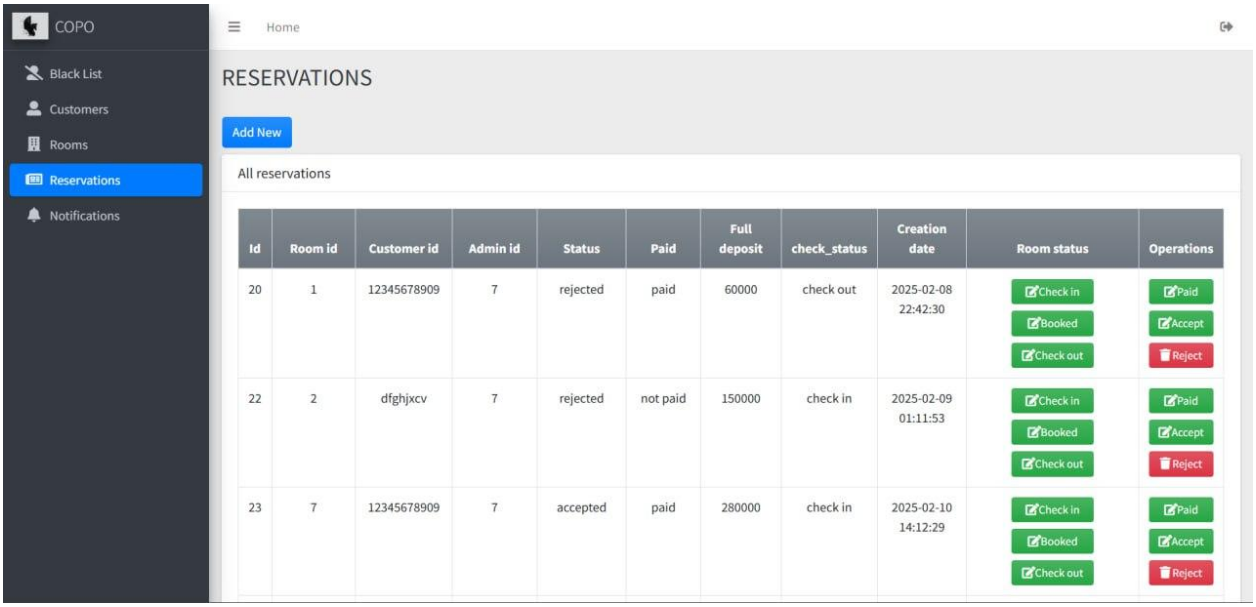
الشكل (17) واجهة الواجهة الرئيسية

واجهة العملاء الحاليين داخل الفندق ومعلوماتهم الشخصية وكشف العميل المسيء منهم :



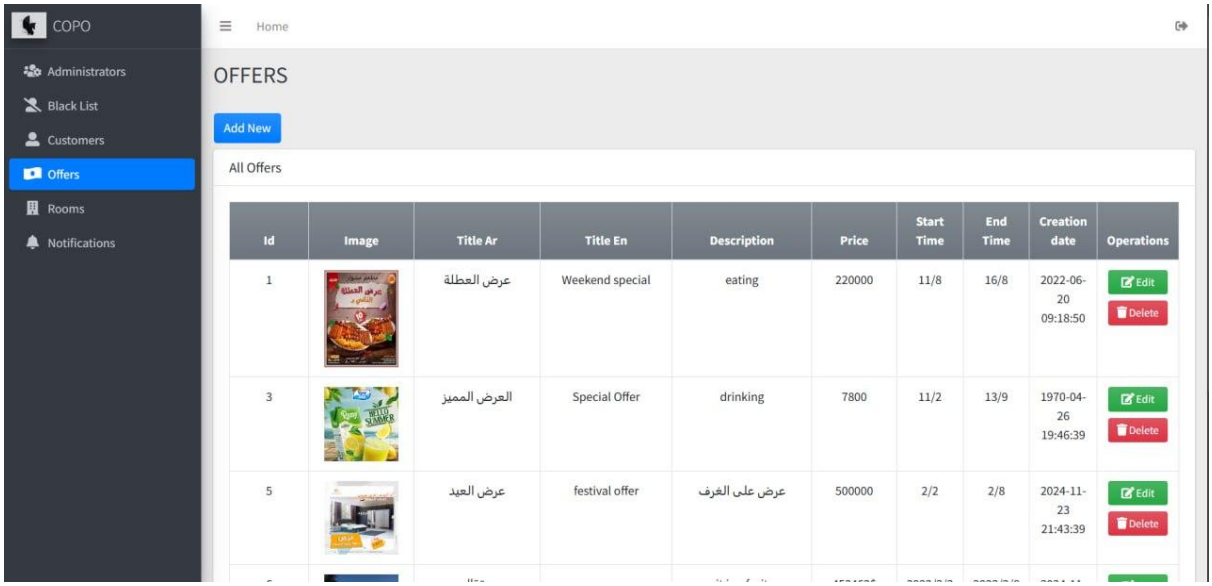
الشكل (18) واجهة العملاء

واجهة الحجوزات ووقت دخول الزبون وخروجه وقبول طلبه ورفضه :



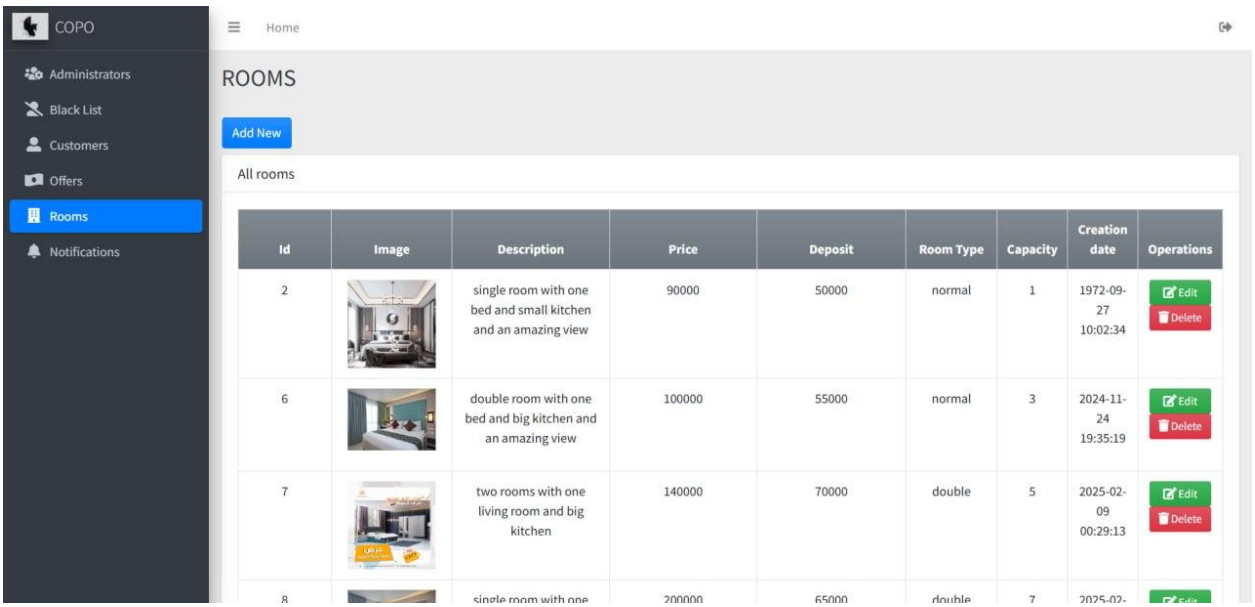
الشكل (19) واجهة الحجوزات

واجهة العروض التي تمكن المدير من إضافة العروض المميزة للعملاء داخل التطبيق :

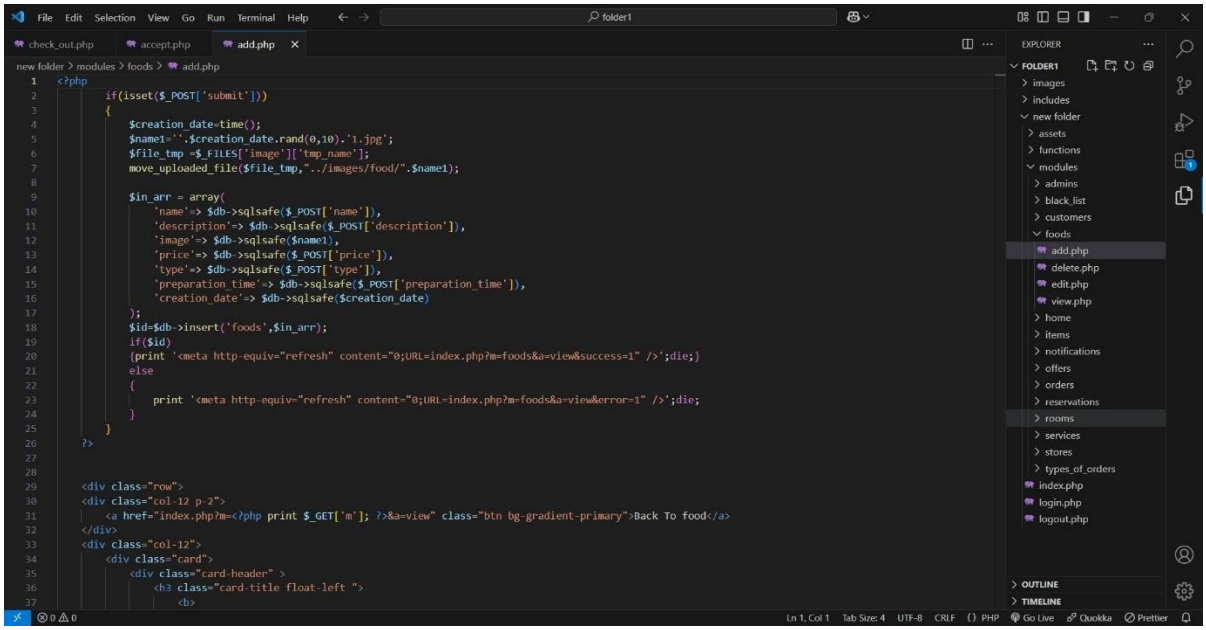


الشكل (20) واجهة offers

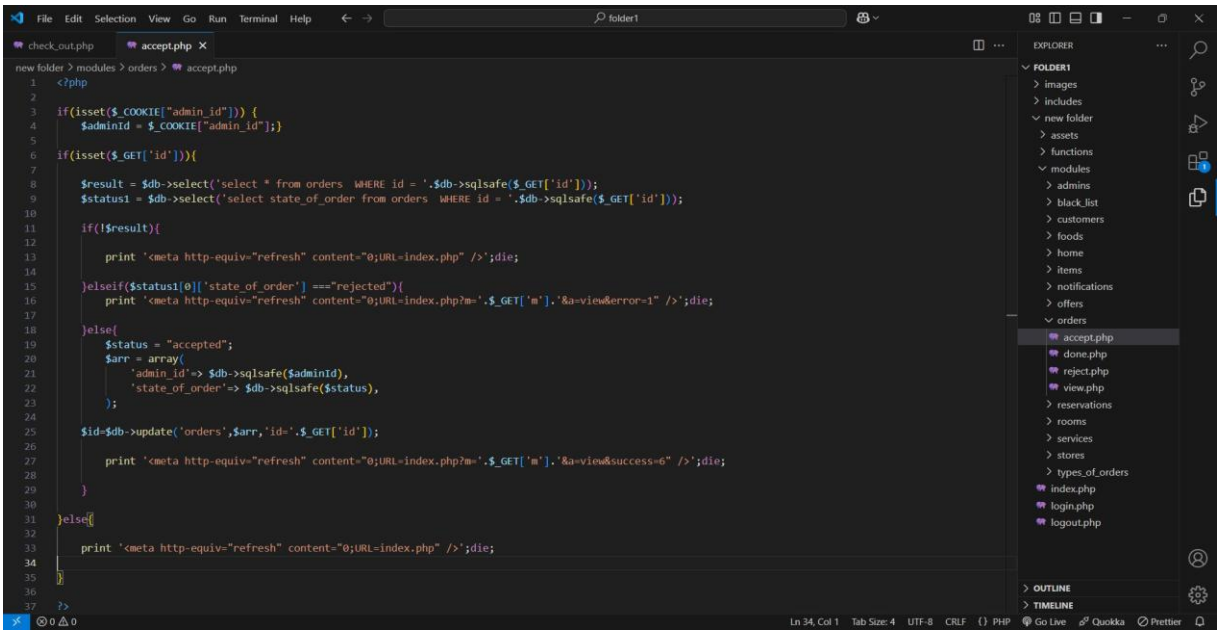
واجهة الغرف التي تمكن المدير من إضافة غرف جديدة في الفندق وحذفها في حال خرجت الغرفة عن الخدمة:



الشكل (21) واجهة ROOMS لإضافة الغرف داخل التطبيق



الشكل (22) add لإضافة الاكل يلي بمطعم الفندق



الشكل (23) accept لقبول طلب تنظيف او أكل او شراء غراض

مستودع البيانات(warehouse):

تحتوي مستودعات البيانات عادة بيانات تاريخية تم اشتقاقها واستخراجها من البيانات الموجودة في قواعد البيانات العادية المستخدمة في التطبيقات والتي تحدث عليها الكثير من عمليات الإدخال والتحديث، كما ويمكن أن تحتوي مستودعات البيانات على بيانات من مصادر أخرى كالملفات النصية والوثائق الأخرى.

تطبيق الموبايل3.1.5 :

تم بناء ال Main class مع اعتماد تطبيق البرمجة الغرضية وذلك عن طريق انشاء صف منفصل لكل واجهة أو مكون داخلها واستدعائه عبر البناء الخاص به مما يحقق التغليف والحماية وسرعة الأداء ويخفف عبء كتابة الأسطر البرمجية.

وبذلك ف إنه يستدعي صف Splash screen الذي يحوي واجهة التحميل بداية التطبيق

```dart
import ...

Future<void> main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await dep.init();
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({Key? key}) : super(key: key);
  @override
  Widget build(BuildContext context) {
    return const GetMaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'Pegasus',
      home: SplachScreen(),
    );  // GetMaterialApp
  }
}
```

الشكل (24) Main class

والذي بدوره يقوم بتوجيه التطبيق بعد التحميل لمدة 4 ثوان واستدعاء متحكمي النماذج بشكل متواز ليقوموا بدورهم بإحضار البيانات عن طريق الأنترنت للواجهة الرئيسية مما يسرع الوصول للمستخدم ويحد من فترة الانتظار

```dart
class _SplachScreenState extends State<SplachScreen> with TickerProviderStateMixin{
  late Animation<double> animation;
  late AnimationController controller;
  Future<void>_loadREsource()async{
    await Get.find<PopularProducController>().getPopularProductList();
    await Get.find<RecommendedProducController>().getRecommendedProductList();
  }
  @override
  void initState(){
    super.initState();
    _loadREsource();
    controller= AnimationController(
        vsync: this,
        duration:const Duration(seconds: 2)
    )..forward();  // AnimationController
    animation= CurvedAnimation(
        parent: controller,
        curve: Curves.linear
    );
    Timer(
      const Duration(seconds: 3),
          (){Navigator.push(
            context,
            MaterialPageRoute(builder: (context) => HomePage()),
          );}
    );  // Timer
  }
```

الشكل (25) Splash screen class

وكان البناء للمتحكمات على الشكل التالي والمثال هنا عن متحكم الغرف الذي يتم عن طريقه توليد قائمة الغرف التي ستعرض للمستخدم.

```dart
class RoomsController extends GetxController{
  final RoomsRepo roomsRepo;
  RoomsController({required this.roomsRepo});
  List<Room> _roomsList = [];
  List<Room> get roomsList => _roomsList;
  bool _isLoaded = false;
  bool get isLoaded => _isLoaded;

  Future<void> getRoomsList() async {
    Response response = await roomsRepo.getOffers();
    if (response.statusCode == 200) {
      _roomsList = [];
      List<Room> p = Rooms.fromJson(response.body).rooms;
      _roomsList.addAll(p);
      print(_roomsList);
      _isLoaded = true;
```

الشكل (26) متحكم الغرف

عند استعادة أي من البيانات من القاعدة فإننا نقوم بالتحكم فيها عن طريق وضعا في صف يقوم ب نمذجتها من لغة Json إلى متغيرات سهلة الاستخدام ضمن dart بعد استدعاء ال API المسئول عنها في صف منفصل والذي يحقق امتداد ل GetxService مما يسهل تشغيل ال URI وطلب البيانات.

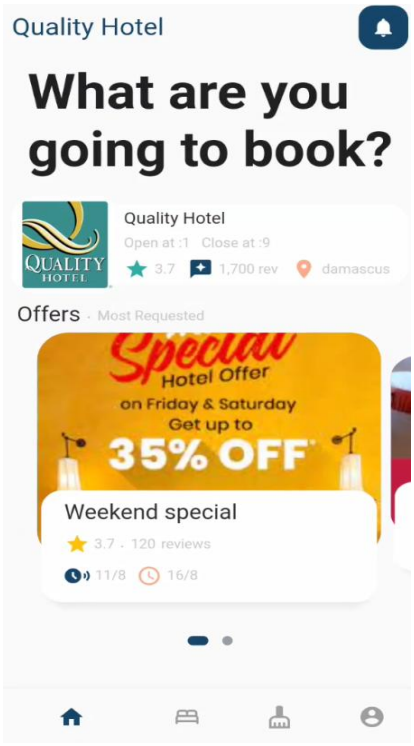
```dart
class Room {
  String ?id ;
  String ?image;
  String ?price;
  String ?descr;
  String ?room_type;
  String ?capacity;
  String ?creationDate;
  Room(
      {this.id,
        this.image,
        this.price,
        this.descr,
        this.room_type,
        this.capacity,
        this.creationDate});
  Room.fromJson(Map<String, dynamic> json) {
    id = json['id'];
    image = json['image'];
    price = json['price'];
    descr = json['descr'];
    room_type = json['room_type'];
    capacity = json['capacity'];
```

الشكل (27) Rooms Model

```dart
class FoodsRepo extends GetxService {
  final ApiClient apiclient;
  FoodsRepo({required this.apiclient});
  Future<Response> getFoods() async {
    return await apiclient.getData(
        'https://undeclared-cannon.000webhostapp.com/Del/folder/API/view_food.php');
  }
}
```

الشكل (28) FoodRepo class

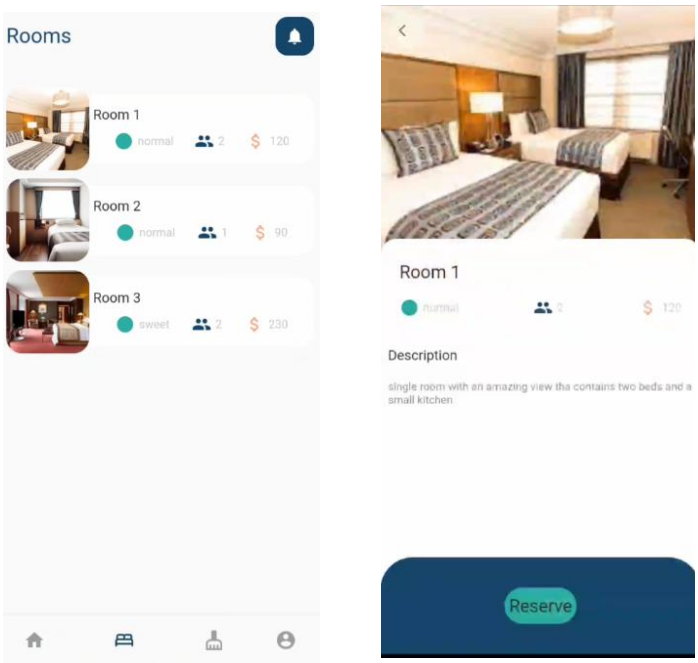
والان بعد التعامل مع البيانات تم بناء الواجهات لعرض هذه البيانات.



الشكل (29) واجهات المستخدم

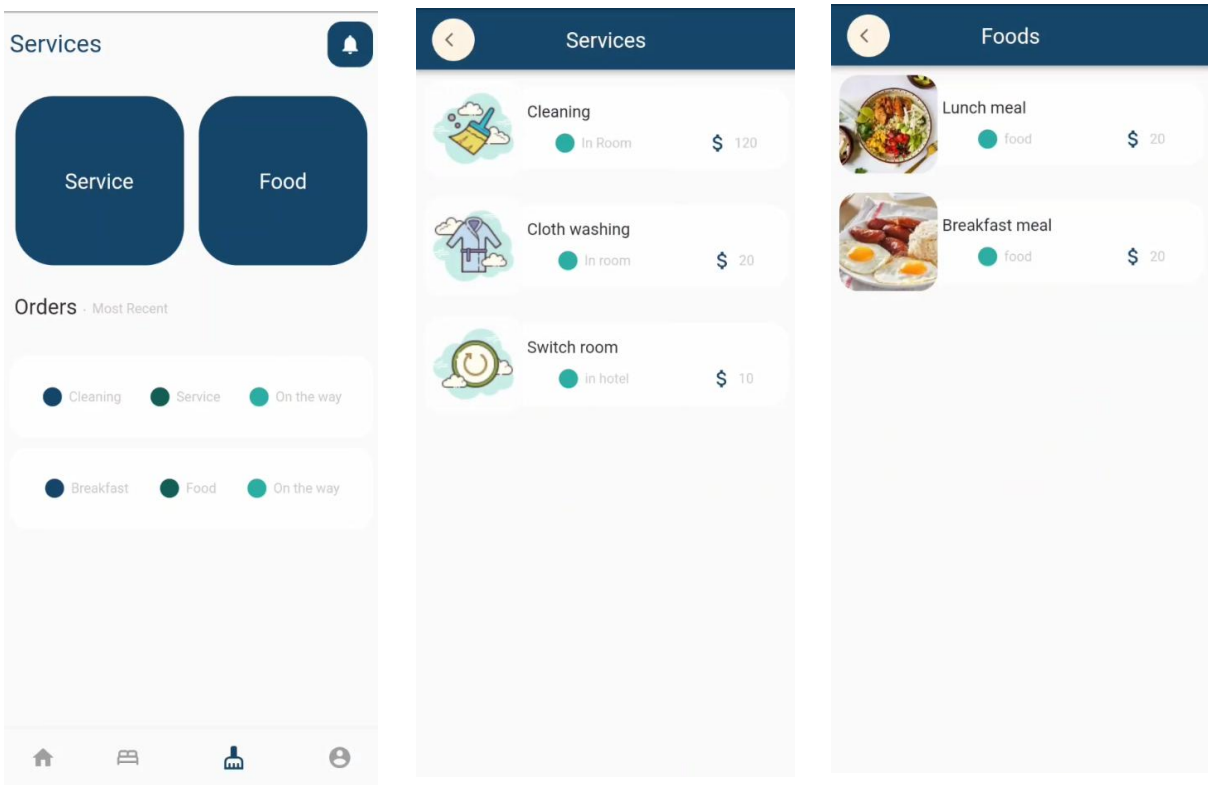
واجهة رئيسية تقوم بعرض المنتجات الأكثر رواجاً عبر التطبيق كما تقوم بعرض المقترحات الخاصة بالزبون كما تعرض أيضا أربع ازرار انتقال سريع لتساعد المستخدم على الوصول إلى كل الخصائص التي يتيحها تطبيقنا بشكل سهل وبسيط.

حيث تعرض أيضا معلومات الفندق الخاصة وتقييمه وساعات الدوام كما تسمح للمستخدم بمعرفة ما إذا كان لديه إشعارات قادمة من موظف الاستعلام.



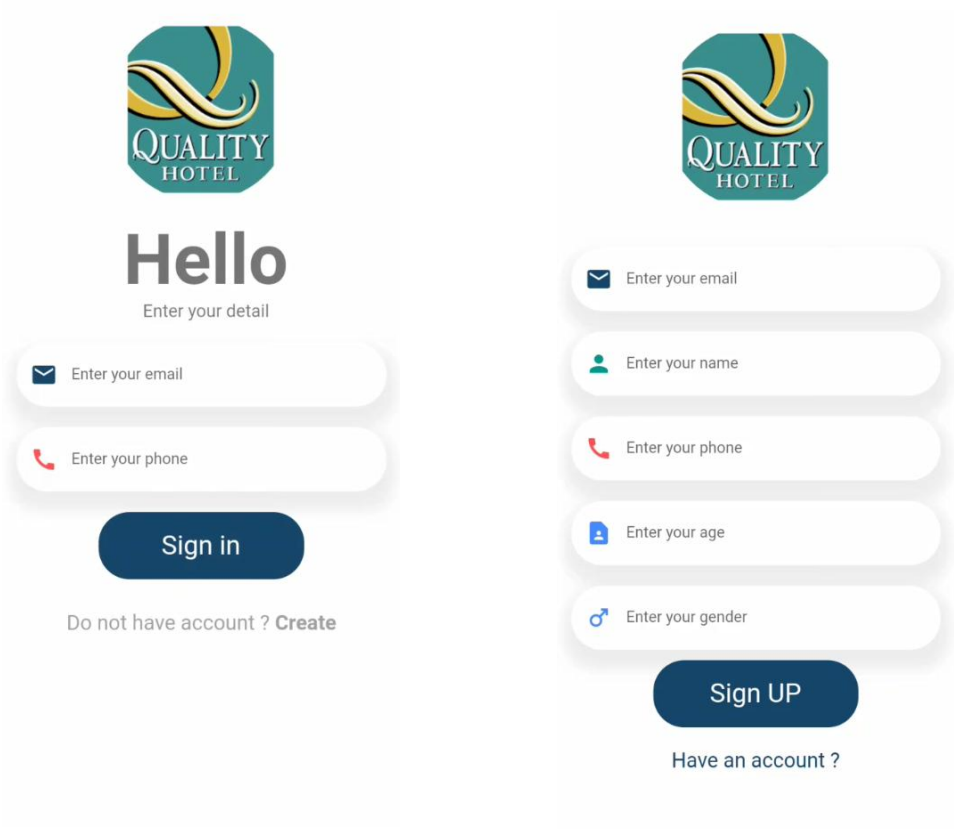
الشكل (30) واجهات الغرف

وبنفس المنهجية تم بناء واجهة الطلبات التي تحوي كل من خيار الطعام أو طلب خدمة



الشكل (31) واجهات الطلبات

كما تم بناء واجهة خاصة للتسجيل للمرة الأول كما أنا تمنح خيار للمستخدمين السابقين بالدخول لحساباتهم.



الشكل (32) واجهات تسجيل الدخول

حيث يتم التحقق من وجود المستخدم في اللائحة السوداء قبل السماح له بالتقدم و ذلك عن طريق ال API مباشرة .

مع إمكانية تمكين ميزة تسجيل الدخول عبر إحدى حسابات مواقع التواصل الاجتماعي و اعتماديها ك ميزة أساسية ستضاف للنظام لاحقا بعد التعامل مع API كل منهم.

4.1.5 واجهة التطبيقات البرمجية API:

تم بناء القسم الأساسي منها والذي يتعلق ب استرداد البيانات من قاعدة البيانات أو الإضافة إليها. حيث تم اعتماد PHP لكتابتهم مع الاعتماد على db.class.php الذي تم بنائه مع لوحة التحكم وتم اعتماد conf.php الذي يعطي المعلومات الأساسية عن الاتصال وطبيعته.

```php
<?php
ini_set('memory_limit', '500M');
date_default_timezone_set("Asia/Damascus");
require("db.class.php");
$full_site_url = '';
$user = 'root';
$password = '';
$server = 'localhost';
```

الشكل (33) conf.php

```php
<?php
class db{
    var $dbConnect;
    function connect($db , $server = "localhost" , $user ="root" ,
        $pass ="" ){
    }
    function select($query){
    }
    function select_where($query , $where = 1){
    }
    function update($table , $values , $where = 1){
    }
    function insert($table , $values ){
    }
    function delete($table , $where){
    }
```

الشكل(34)db.class.php

وكمثال API يقوم ب إعادة معلومات العروض من قاعدة البيانات بعد تحويلها إلى لغة JSON لكي يتم التعامل معها في جهة Flutter وتحويلها إلى أغراض لكي يتم عرضها للمستخدم تباعاً.

إذ يعتمد نفس التابع select المستخدم من قبل لوحة التحكم للولوج إلى قاعدة البيانات.

```php
<?php
    require ("../includes/conf.php");
    $headers = apache_request_headers();
    header('Content-Type: application/json');
    $final=array();
    $res=$db->select('SELECT * FROM `offers`');
    if($res)
    {
        $final['data']=$res;
    }

    echo json_encode($final, JSON_UNESCAPED_UNICODE);
?>
```

الشكل (35) get_offers.php

ومثال على API من نوع POST فنذكر الخاص ب تسجيل الدخول.

حيث يقوم ومباشرة فور استدعائه بتفقد رقم الهاتف الخاص بالعميل إذا كان متواجد مسبقا في قاعدة النظام ويتتبع حالته حيث تدل الحالة 3 أن العميل تم حظره مسبقا بسبب فعل مخالف وبذلك يقوم بمنع عملية الدخول أما في حال كان المستخدم سليما ف إنه يعيد كافة معلوماته الموجودة على قاعدة البيانات لكي يتم استخدامها داخل التطبيق

```php
<?php
require ("../includes/conf.php");
$headers = apache_request_headers();
header('Content-Type: application/json');
$final=array();
$phone=$_POST['email'];
$phone=$_POST['phone'];
$check_if_block = $db->select('SELECT * FROM `customer` WHERE status="Blocked" AND `phone`='.$db->sqlsafe($phone).
if($check_if_block){
    $final['status']='blocked';
}else{
    $res=$db->select('SELECT * FROM `customer1`WHERE `phone`='.$db->sqlsafe($phone).'');
    if($res){

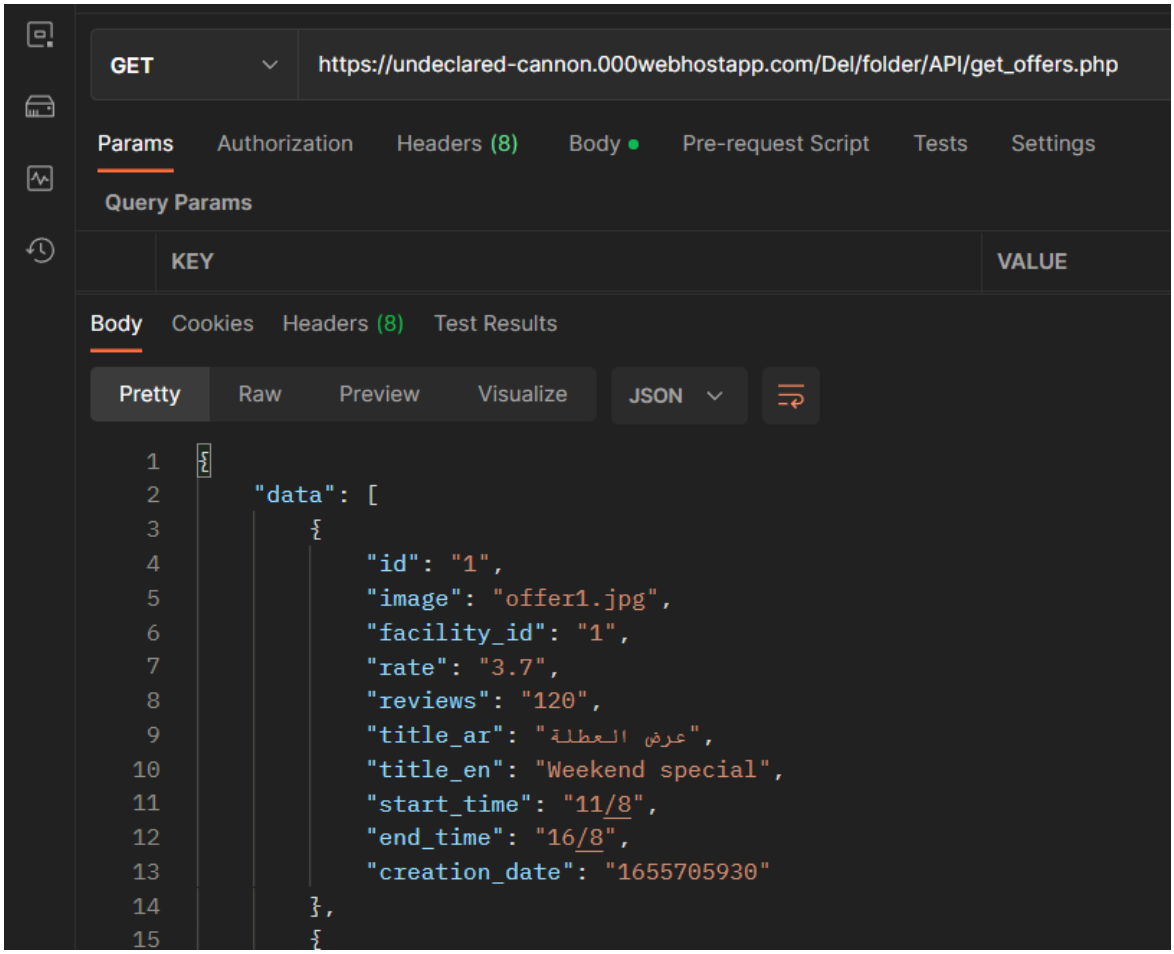
            $final=$res;
    }else{
            $final['status']='not found';
    }
}
echo json_encode( $final, JSON_UNESCAPED_UNICODE);

?>
```

الشكل (36) log_in.php

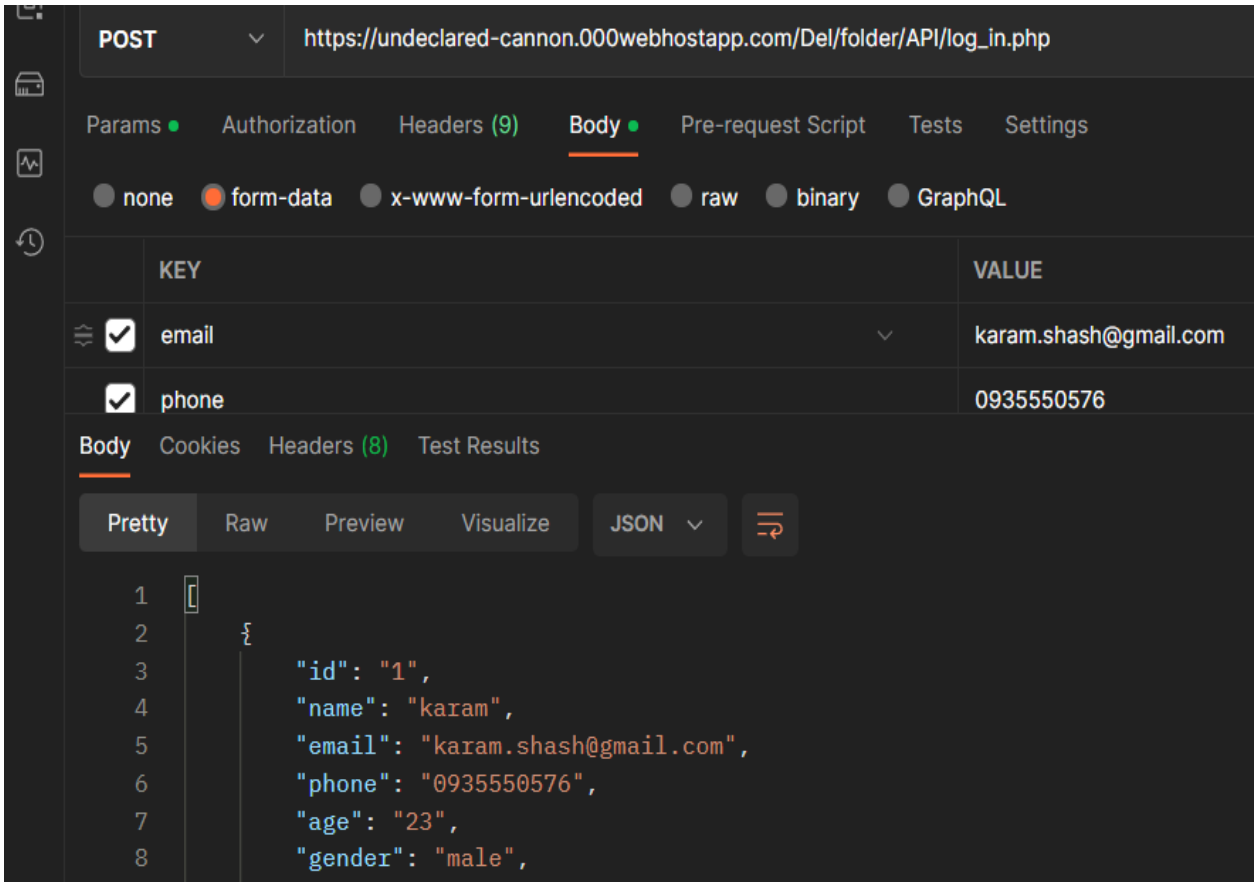
2.5 مرحلة الاختبار:

حيث اختبرنا عمل الAPI عن طريق تطبيقPostman والذي يتيح لنا تشغيل API ومعرفة أخطاءه وبذلك قمنا ب اختبار الAPI الذين تم ذكرهم سابقا مع إعطاء النتائج التالية.



الشكل (37) اختبار get_offers

كما تم اختبار الAPI التي تحتاج إلى معلومات مرسلة من طرف المستخدم ومثال على ذلك الخاص ب تسجيل الدخول حيث يأخذ من المستخدم عنوان البريد ورقم الهاتف



الشكل (38) اختبار log_in

3.5 حالات الاختبار :

حالات اختبار مدير الفندق:

ID	عنوان الحالة	شرح	نتيجة متوقعة
REQ-001	قبول طلب جديد	إدخال معلومات الطلب وإضافة تفاصيل العميل	يتم حفظ الطلب بنجاح مع رقم تعريفي فريد
REQ-001	رفض طلب غير صالح	محاولة إدخال بيانات غير صالحة	عرض رسالة خطأ واضحة وتحديد الحقول المعيبة
REQ-001	تعديل طلب موجود	تغيير تفاصيل طلب نشط	يتم حفظ التعديلات مع سجل للتغييرات

جدول حالات الاختبار 1-1

ID	عنوان الحالة	شرح	نتيجة متوقعة
REQ-002	إضافة موظف جديد	إنشاء حساب موظف مع صلاحياته	يتم إنشاء الحساب بنجاح مع إرسال بريد إلكتروني للتأكيد
REQ-002	تعديل بيانات الموظف	تحديث معلومات وحقوق الموظف	يتم تحديث البيانات مع حفظ سجل التغييرات
REQ-002	إزالة موظف	تعطيل حساب الموظف مع الاحتفاظ بسجله	يتم تعطيل الحساب مع نقل مهامه المتبقية

جدول حالات الاختبار 1-2

نتيجة متوقعة	شرح	عنوان الحالة	ID
يتم نشر العرض ويصبح متاحاً للعملاء	إنشاء عرض مع شروط وشروط انتهاء صلاحية	إضافة عرض جديد	REQ-003
يتم تطبيق الخصم وتحديث تكلفة الحجز	اختيار وعمل خصم على حجز فعال	تطبيق خصم على حجز	REQ-003
يتم إلغاء العرض مع إخطار العملاء المتأثرين	إلغاء عرض قبل انتهاء صلاحيته	إلغاء عرض	REQ-003

جدول حالات الاختبار 1-3

حالات اختبار موظف الاستقبال:

نتيجة متوقعة	شرح	عنوان الحالة	ID
يتم إضافة العميل مع تخزين سبب الإضافة	تقديم سبب واضح للإضافة	إضافة عميل إلى قائمة المسيئين	REQ-009
يتم إزالة العميل مع حذف تاريخ المخالفات	إعادة تقييم حالة العميل	إزالة عميل من القائمة	REQ-009

جدول حالات الاختبار 2-1

ID	عنوان الحالة	شرح	نتيجة متوقعة
REQ-010	عرض قائمة العملاء النشطين	تصفية حسب نوع الغرفة أو تاريخ الوصول	عرض البيانات المطلوبة مع خيارات الترشيح
REQ-010	بحث عن عميل محدد	البحث باستخدام اسم أو رقم الحجز	عرض معلومات العميل ومعلومات إقامته

جدول حالات الاختبار 2-2

ID	عنوان الحالة	شرح	نتيجة متوقعة
REQ-011	إضافة غرفة جديدة	تحديد نوع الغرفة وسعرها وخياراتها	يتم إضافة الغرفة للقائمة المتاحة
REQ-011	تحديث حالة الغرفة	تغيير حالة الغرفة (مستعدة/مشغولة/ تحت الصيانة)	يتم تحديث حالة الغرفة فورياً

جدول حالات الاختبار 2-3

ID	عنوان الحالة	شرح	نتيجة متوقعة
REQ-012	تأكيد حجز جديد	إضافة معلومات العميل وتاريخ الإقامة	يتم إنشاء الحجز وإرسال تأكيد للعميل
REQ-012	إلغاء حجز	إلغاء حجز مع تحديث حالة الغرفة	يتم إلغاء الحجز وإرجاع الرسوم المدفوعة حسب السياسة

جدول حالات الاختبار 2-4

ID	عنوان الحالة	شرح	نتيجة متوقعة
REQ-013	إرسال إشعارات للعملاء	إرسال رسائل متنوعة (مواعيد/عروض/تنبيهات)	يتم إرسال الإشعارات بنجاح مع تتبع التسليم
REQ-013	إرسال تنبيهات الموظفين	إرسال مهام وتذكيرات للموظفين	يتم إرسال التنبيهات مع تتبع القراءة

جدول حالات الاختبار 2-5

حالات اختبار خدمة الغرف:

ID	عنوان الحالة	شرح	نتيجة متوقعة
REQ-014	إضافة مورد جديد	إضافة مطعم أو متجر مع تفاصيله وقائمة خدماته	يتم إضافة المورد للقائمة المتاحة
REQ-014	تحديث قوائم الطعام	تعديل قائمة الطعام وأسعارها	يتم تحديث القائمة فورياً للعملاء

جدول حالات الاختبار 3-1

ID	عنوان الحالة	شرح	نتيجة متوقعة
REQ-015	إضافة طلب تنظيف	إرسال طلب تنظيف لغرفة مع تحديد الوقت المفضل	يتم تسجيل الطلب وجدولة المهام
REQ-015	إلغاء طلب تنظيف	إلغاء طلب تنظيف قبل تنفيذه	يتم إلغاء الطلب وإعلام فريق التنظيف

جدول حالات الاختبار 3-2

نتيجة متوقعة	شرح	عنوان الحالة	ID
يتم تسجيل الطلب وجدولة التوصيل	إرسال طلب طعام مع اختيار الوقت والتوقيت	إضافة طلب طعام	REQ-016
يتم تحديث الطلب مع إخطار المطعم	تغيير طلب طعام قبل إعداده	تعديل طلب طعام	REQ-016

جدول حالات الاختبار 3-3

نتيجة متوقعة	شرح	عنوان الحالة	ID
يتم إضافة الخدمة للقائمة المتاحة	إضافة خدمة مع تفاصيلها وأسعارها	إضافة خدمة جديدة	REQ-017
يتم تحديث الخدمة فورياً	تعديل تفاصيل وأسعار الخدمة	تحديث تفاصيل الخدمة	REQ-017

جدول حالات الاختبار 3-4

حالات اختبار العميل:

نتيجة متوقعة	شرح	عنوان الحالة	ID
يتم تسجيل الدخول بنجاح مع توجيه مناسب	دخول البيانات الصحيحة للحساب	تسجيل دخول ناجح	REQ-018
يتم إرسال رابط إعادة التعيين بنجاح	طلب استعادة كلمة المرور وإعادة تعيينها	استعادة كلمة المرور	REQ-018

جدول حالات الاختبار 4-1

ID	عنوان الحالة	شرح	نتيجة متوقعة
REQ-019	عرض العروض المتاحة	عرض جميع العروض والخصومات الحالية	عرض العروض مع شروطها وفترات صلاحيتها
REQ-019	عرض الغرف المتاحة	عرض الغرف المتاحة مع تفاصيلها وأسعارها	عرض الغرف مع خيارات الترشيح والتصفية

جدول حالات الاختبار 4-2

ID	عنوان الحالة	شرح	نتيجة متوقعة
REQ-020	حجز غرفة	اختيار غرفة وتأكيد الحجز والدفع	يتم تأكيد الحجز وإرسال إيصال الدفع
REQ-020	إلغاء حجز	إلغاء حجز مع تطبيق سياسة الإلغاء	يتم إلغاء الحجز مع إرجاع الرسوم حسب السياسة

جدول حالات الاختبار 4-3

ID	عنوان الحالة	شرح	نتيجة متوقعة
REQ-021	إضافة تقييم	تقديم تقييم مع تعليقات اختيارية	يتم حفظ التقييم وتحديث متوسط التقييمات
REQ-021	تعديل تقييم سابق	تغيير تقييم وتعليقات سابقة	يتم تحديث التقييم مع حفظ التاريخ السابق

جدول حالات الاختبار 4-4

ID	عنوان الحالة	شرح	نتيجة متوقعة
REQ-022	إضافة طلب طعام	اختيار الأطباق والكميات والتوقيت المفضل	يتم تسجيل الطلب وجدولة التوصيل
REQ-022	تعديل طلب طعام	تغيير طلب قبل إعداده	يتم تحديث الطلب مع إخطار المطعم

جدول حالات الاختبار 4-5

الفصل السادس

النشر

سيتم النشر وفق مراحل منفصلة:

المرحلة الأولى الاستضافة

ويتم فيها استضافة لوحة التحكم وقاعدة البيانات على مخدم يعطينا رابط عبرها لكي نتمكن من الولوج ل اللوحة والموقع عن طريق الويب.

المرحلة الثانية

تخريج التطبيقين على صيغة قابلة للتحميل لكي يتمكن المستخدمين من تثبيته على اجهزتهم.

المرحلة الثالثة

وهي مرحلة مراقبة الأداء ويتم فيها اختبار النظام مع مستخدمين فعليين وضغط واقع عمل حقيقي مع تامين صيانة دورية له اعتمادا على التغذية الراجعة من المستخدمين.

المرحلة الرابعة

بعد إتمام المراحل السابقة، تتمثل الخطوة التالية في توفير الدعم والصيانة للبرنامج، والتأكد من أنه يعمل بسلاسة. إذا صادف العميل والمستخدمون أخطاء / عيوب / مشاكل أثناء الاستخدام، فإن إصلاحها هو الغرض الرئيسي من هذه المرحلة .

المراجع

Fazle Rahman,Sayed(2014).Jump Start Bootstrap

Moila,Alberto(2020). Flutter Complete Reference

Murach,Joel(2012). Murach's MySQL

Payne,Rap(2019). Beginning App Development with Flutter

الملاحق:

ملحق عن لغة PHP

لغة PHP هي لغة برمجة مفتوحة المصدر، تم إنشائها لبرمجة وتطوير مواقع الويب، و قد تم إنشاء لغة PHP في البداية والتي هي اختصار ل Personal Home Pages وهي عبارة عن مجموعة من التطبيقات التي كتبها مبرمج يدعى راسموس بلغة برمجة بيرل، ثم توالى التطوير للغة PHP واتسع مفهومها عبر الزمن، وتمت صياغتها بلغات أخرى وتوالت التعديلات وتوسعت الدوال التي تشملها لغة برمجة PHP وقد تم إنشاء عدة إصدارات من اللغة، وتأتي طفرة التحول من PHP3 حيث في هذا الإصدار اختلفت اللغة كليا من حيث العمل والمفهوم وأصبحت أكثر تطورا فشملت البرمجة كائنية التوجه، وأصبح PHP هو اختصارا لمفهوم جديد وهو Hypertext Preprocessor وقد تم إطلاق هذا المفهوم في عام 1998 بعد شهور طويلة من العمل على تطوير اللغة في المختبرات واستمر التطوير بعد ذلك فلم يكن هذا الإصدار الأخير للغة PHP فيستمر العمل على تطوير PHP باستمرار

مميزات لغةPHP

مميزات لغة PHP عديدة ومنها:

1. أنها مفتوحة المصدر، فيمكنك الاطلاع على الكود المصدري وذلك بشكل مجاني تماما، كما يمكنك التعديل والتطوير باستمرار نظرا لأن اللغة مفتوحة المصدر وستظل كذلك.
2. لغة PHP لغة برمجة سهلة التعلم وسلسلة، فتعد من لغات البرمجة التي تتميز بالبساطة من حيث الأكواد والمعاملات .
3. كما أنها لغة برمجة متعددة المنصات فتعمل على أنظمة مختلفة مثل: ويندوز ولينوكس .
4. تدمج أكواد اللغة داخل أكواد لغة HTML مما يجعل التعامل في المواقع سهل للغاية وسريع .
5. كما أنها تتميز بالأمان التام ومرونة الاستخدام .
6. وتتميز كذلك بأن لها استخدامات عدة فتستخدم في تطوير البرامج ذات الواجهة الرسومية، وأيضا في تنفيذ برامج إدارية عديدة، وبشكل واسع في مجال الويب وكتابة وتطوير المواقع الإلكترونية كما يمكن استخدامها في إنشاء مواقع ويب تفاعلية .
7. كما تحظى بدعم فني كامل عبر الموقع الرسمي للغةPHP

ملحق عن إطار العمل Flutter

منذ مدة ليست بطويلة اشتهر إطار عمل جديد لصناعة و برمجة تطبيقات الأندرويد و IOS، إطار Flutter من ضمن مشاريع شركة جوجل مفتوحة المصدر، مبني بلغة Dart والتي بدورها مفتوحة المصدر من نفس الشركة، تسعى جوجل بهذا الإطار لتوفير طرق أكثر سلاسة لصناعة تطبيقات الهواتف الذكية و تسهيلها على المطورين، فكما نعلم أن صناعة تطبيقات الهواتف حاليا تتطلب معرفة بلغة JAVA او Kotlin للأندرويد ، الشيء الذي ساهم في انقسام مجتمع مطوري تطبيقات الهواتف و تخصص المطورين في أحد الأنظمة عن الأخر، بهذه الفكرة بدأ مشروع إطار عمل Flutter لِلَم شمل المطورين و صناعة تطبيقات لكل المنصات بشكل (Native) كأن التطبيق مبرمج باللغة الأصلية للنظام.

أهم ما جاء إطار فلاتر لتقديمه لمجتمع صانعي تطبيقات الهواتف هو إمكانية صناعة تطبيقات أندرويد و تطبيقات IOS في نفس الوقت و بكود مصدري مماثل، كما أنه يستخدم لغة برمجة مجمعة تسمى Dart ، والتي تم ترجمتها "في وقت مبكر" ('ahead of time 'AOT) إلى رمز أصلي لكل من iOS و Android. كما أن الإطار ليس مثل سابقيه التي تعتمد على JavaScript كجسر بين المتصفح ونظام التشغيل، بالإضافة لتسهيلها نمطية كتابة الأكواد وإنشاء واجهات احترافية بأقل كود ممكن ومقروء.

و قد أصدرت جوجل قبل أيام النسخة الأولى الثابتة من Flutter التي حملت اسم Flutter 1.0، الذي ستكون عبارة عن منصة متداخلة (Cross-Platform) من أجل تطوير تطبيقات لكل الأنظمة و لكل الأجهزة أيضا، ليبدأ بشكل رسمي عصر الـ Flutter .

من مميزات Flutter التي جعلته أكثر سهولة لتطوير تطبيقات الهواتف هو اعتماده على نظام الحاجيات (Widgets) فكل عنصر من Flutter يعتبر Widget وقد تم استنباط هذا النظام من React JSX، وهذه تعتبر ميزة للمطورين اللذين سبق لهم التعامل مع React، فقد يجدون أن صناعة التطبيقات باستخدام Flutter أكثر سهولة. نظام Widgets في إطار Flutter جعل من السهل إنشاء مكونات (Components) جديدة وقوة لغة Dart ساهمت أيضا بكونها لغة كائنية التوجه في هذا الأمر بحيث يمكن أخذ Widget رئيسي وإضافة عليه خصائص جديدة من تأثيرات وحركات (Animation) بشكل بسيط وبدون تعقيد. هذا الأمر مَكَّنَ من إنشاء مكونات جديدة داخل الإطار وإدراج Material Design في المكونات التي تشتغل على الأندرويد و Cupertino على IOS. واحدة من مميزات إطار Flutter هي دعمه بشكل كامل لواجهات ال Material Design فعلى نظام الأندرويد تأتي المكونات من Text و Button و Switch افتراضيا على تنسيق الماتريال ديزاين هذا ما سيوفر عليك عناء التعديل على الديزاين كما الحال على الأندرويد بالجافا

ملحق عن MySQL

هو نظام إدارة قواعد البيانات علائقي يعتمد التعامل معه على لغة إس كيو إل. وسمي بهذا الاسم تبعا لابنة مبرمجه الأصلي Michael Widenius، والتي اسمها My. ماي إس كيو إل هو من المنتجات مفتوحة المصدر ينشر كوده المصدري تحت رخصة جنو العمومية بالإضافة إلى بعض الاتفاقيات الاحتكارية. كانت تملكه وترعاه الشركة الربحية السويدية MySQL AB، لكن تملكه الآن صن ميكروسيستمز (والتي هي حاليا فرع من أوراكل).

يتميز خادم ماي إس كيو إل بسرعته الكبيرة؛ لأنه خادم قواعد بيانات متعدد المسالك (بالإنجليزية: multi-threaded) مما جعل إمكانية الاستعلام من قاعدة البيانات سريعة جدا؛ ويتميز بسهولة ربط جداوله بواجهة المستخدم التي تصمم بلغات البرمجة، فمثلا يمكنك ربطها بالبرامج المكتوبة بلغة فيجوال بيسك من خلال واجهة ODBC الخاص بها، كذلك له مكتبة خاصة به للتعامل معه من خلال أدو دوت نت.

الواجهة الرسومية (GUI) هي عبارة عن واجهه تمكن المستخدم من التفاعل مع التطبيق من خلال ايقونات رسومية ومؤشرات على عكس الطرفية التي يتفاعل معها المستخدم من خلال كتابة الاوامر بشكل مباشر. تتوفر واجهات رسومية مجانية تمكن المستخدمين من التعامل مع بنية قاعدة البيانات والبيانات بشكل مرئي.

Abstract:

Due to the development of technology and its occupation of a large part of our daily lives, most of the daily activities were automated, so hotels searched for new ways to keep pace with the needs of the market. Where mobile applications have evolved to be a valuable tool for starting projects and businesses. Customers are interested in obtaining their service in a simple and easy way, which can be provided by technology (applications) with a single click, as it provides fast and effective interaction, and this is one of the requirements desired by both parties (service manager and customer).

Hence, the idea of our project was launched, which aims to provide an integrated software system for managing and booking a hotel, through which communication takes place between the hotel and the customer, as it provides all services.

It also provides analytical reports and helps him in better marketing according to customer demand.

Syrian Arabic Republic Syrian Private University
Faculty of Information Engineering Software Engineering

Hotel Management System

Senior-2 Project-Completed the requirements for obtaining a bachelor's degree in Informatics Engineering - Software Engineering and Information Systems

Prepared By:

Farok jamil shihade

Supervised by: Eng.Tharaa Aslaan

2025/2024 Year