بِسْمِ اللهِ الرَّحْمنِ الرَّحِيمِ

جامعة بوليتكنك فلسطين

كلية تكنولوجيا المعلومات وهندسة الحاسوب

بوابة العيادة الطبية

(موقع الكتروني لإدارة الخدمات الطبية)

فريق العمل:

قيس أحمد يوسف عمرو يزن عطية محمد خمايسة عمر هشام محمد شنك

اشراف:

د. محمد أبو طه

يقدم هذا المشروع استكمالاً لمتطلبات التخرج والحصول على درجة البكالوريوس في علم الحاسوب وتكنولوجيا المعلومات بجامعة بوليتكنك فلسطين

الاهـــــداء

﴿وَقُلِ اعْمَلُوا فَسَيَرَى اللَّهُ عَمَلَكُمْ وَرَسُولُهُ وَالْمُؤْمِنُونَ﴾ صدق الله العظيم

إلى أمهاتنا الغاليات، من اختصهن الله بأن جعل الجنة تحت أقدامهن، اللواتي كنَّ الحاضن الأول لأحلامنا، والشموع التي أضاءت دروبنا بنور الحنان والتضحيات.

إلى آبائنا العظماء، السند والعزوة، الذين بذلوا أعمارهم بكل سخاء، وصاغوا لنا سلالم المجد بعرقهم وصبرهم.

إلى إخوتنا وأخواتنا، الذين كانوا لنا العائلة والدعم، العيون التي نظرت إلينا بمحبة صافية، والقلوب التي لم تبخل بالعطاء يومًا.

إلى معلمينا الأفاضل، مشاعل العلم والمعرفة، الذين خطّوا في قلوبنا دروس الحكمة، وزرعوا فينا شغف التعلم وحب المعرفة.

إلى زملائنا الأعزاء، رفاق الدرب، الذين كانت صداقتهم عزاءً وسندًا في محطات هذه الرحلة.

إلى أنفسنا، الذين تحمّلنا أعباء الطريق بثبات، وتخطينا العقبات بإيمان لا يتزعزع بأن الله لا يضيع أجر من أحسن عملًا.

إلى فلسطين الحبيبة، جرح الأمة النازف الذي لا يندمل، رمز العزة والكرامة والصمود.

إلى أرواح الشهداء الطاهرة، الذين ارتقوا دفاعًا عن الحق والأرض، وإلى الأسرى الذين تحملوا ظلمات السجون وهم يضيئون الأمل في قلوبنا.

إلى غزة العزة، التي وقفت بوجه المحن بإرادةٍ لا تُقهر، لتثبت أن الكبرياء لا يُحاصر.

إلى المسجد الأقصى المبارك، الذي نسأل الله أن يحرره من دنس الاحتلال، ويعيده طاهرًا شامخًا إلى حوزة المسلمين.

وإلى روح زميلنا العزيز **عماد الهريني**، الذي فارقنا في حادث أليم، لكنه لم يفارق قلوبنا . إلى ذكراه العطرة التي ستظل خالدة بيننا، ولابتسامته التي كانت تبعث الأمل في نفوسنا . رحمك الله يا عماد وأسكنك فسيح جناته، وجعل عملنا هذا نورًا يُهدى لروحك الطاهرة.

اللهم اجعل هذا العمل شاهدًا لنا يوم نلقاك، ووفقنا لأن نكون نورًا لمن حولنا، واملأ قلوبنا بالإيمان واليقين.

"فلسطينُ في القلب، وستبقى حُلم التحرير قريبًا بإذن الله".

الشكر والتقدير

بقلوب ملؤها الامتنان، وأرواح يغمرها الفخر والاعتزاز، نقف اليوم على أعتاب نهاية رحلتنا الجامعية، نطوي صفحاتٍ من الكفاح والمثابرة، ونفتح أبواب مستقبل نأمل أن يحمل الخير لنا ولأمتنا.

نرفع أسمى آيات الشكر والتقدير لأولئك الذين كان لهم الفضل الأكبر في رحلتنا العلمية، أساتذتنا الأفاضل، مشاعل النور الذين لم يبخلوا بعلمهم وجهودهم ليصوغوا منّا بذور الأمل التي ستنهض بالأمة من جديد.

إلى كل أستاذٍ حمل أقدس رسالة في الحياة، ووقف ليكون منارةً للعلم والمعرفة، إلى كل من شجّعنا، وأرشدنا، ووجهنا على درب الحياة.

"كُن عالماً، فإن لم تستطع فكن متعلماً، فإن لم تستطع فأحب العلماء، فإن لم تستطع فلا تبغضهم".

وبخالص التقدير نخص بالشكر:

مشرف مشروعنا، **الدكتور محمد أبو طه**، الذي كان لنا العون والنصح في كل خطوة، والذي كان دعمه مصدر إلهامٍ لنا طوال هذه الرحلة. نقول لك بشراك قول رسول الله صلى الله عليه وسلم:

"إن الحوت في البحر والطير في السماء ليصلون على معلم الناس الخير".

كما نتوجه بخالص الشكر لكل من كان له أثرٌ في تحقيق هذا المشروع، لكل يدٍ امتدت بالدعم، ولكل كلمةٍ ألهمتنا، ولكل جهدٍ ساهم في أن يصبح هذا المشروع واقعاً نفاخر به.

اللهم اجعل هذا العمل خالصًا لوجهك الكريم، وبارك في كل من علّمنا، وأرشدنا، وساندنا، فبفضلهم وبإرادتنا وصلنا إلى ما نحن عليه اليوم.

الملخص

مع التطور الرقمي المتسارع، أصبحت الحاجة إلى حلول تقنية تُسهّل التواصل بين المرضى والأطباء أكثر إلحاحًا، خاصة في العيادات الخارجية. يواجه المرضى صعوبة في حجز المواعيد وإدارتها، كما يفتقر الأطباء إلى نظام فعّال لتنظيم جداولهم والوصول إلى معلومات مرضاهم بسلاسة. بالإضافة إلى ذلك، تعاني العيادات من تحديات في إدارة بيانات المرضى، الوصفات الطبية، والفحوصات المخبرية، مما يؤثر على جودة الخدمات الطبية المقدمة.

يهدف مشروعنا إلى تطوير نظام إلكتروني شامل يُبسّط إدارة الخدمات الطبية ويحسّن تجربة المستخدمين. يتيح للمرضى البحث عن الأطباء والعيادات، حجز المواعيد أو تعديلها، وإدخال تاريخهم الطبي لضمان رعاية دقيقة. كما يُوفر للأطباء منصة متكاملة لإدارة جداولهم، الاطلاع على بيانات المرضى، إصدار الوصفات الطبية، وإدارة الفواتير. يدعم النظام أيضاً مسؤولي العيادات في إدارة الحسابات، إضافة عيادات جديدة، وتحليل البيانات بطرق مرئية. علاوة على ذلك، يُسهّل على المختبرات إرسال واستقبال الفحوصات الطبية إلكترونيًا. بذلك، يساهم المشروع في تحسين كفاءة الخدمات الصحية وتقديم تجربة أكثر تنظيماً وراحةً لجميع الأطراف.

Abstract

With the rapid advancement of digital technology, the need for innovative solutions to facilitate communication between patients and doctors has become more urgent, especially in outpatient clinics. Patients often face difficulties in booking and managing appointments, while doctors lack an efficient system to organize their schedules and access patient information seamlessly. Additionally, clinics struggle with managing patient data, medical prescriptions, and laboratory tests, which affects the quality of medical services provided.

Our project aims to develop a comprehensive electronic system that simplifies medical service management and enhances user experience. It allows patients to search for doctors and clinics, book or modify appointments, and enter their medical history to ensure accurate care. The system also provides doctors with an integrated platform to manage their schedules, view patient records, issue prescriptions, and handle billing. Moreover, it supports clinic administrators in managing user accounts, adding new clinics, and analyzing data through visual reports. Additionally, it facilitates laboratory operations by enabling the electronic exchange of medical test results. This project contributes to improving healthcare service efficiency and offering a more organized and convenient experience for all stakeholders.

جدول المحتويات

جدول المحتويات 6
قائمة الأشكال 7
قائمة الجداول 8
الفصل الأول: المقدمة 9
1.1 مقدمة 10
1.2 فكرة المشروع 10
1.3 المشكلة 10
1.4 الدوافع والمحفزات 11
1.5 الأهداف 11
1.6 نظرة عامة على المشروع 12
1.7 نطاق المشروع 12
1.8 منهجية المشروع 12
1.9 أهمية المشروع 12
1.10 البدائل العامة 13
1.11 النظام المقترح 14
1.12 Context diagram 14
1.13 جدولة المهام والمخطط الزمني للمهام 15
1.14 ملخص الفصل 15
الفصل الثاني: تحليل المتطلبات 16
2.1 مقدمة 17
2.2 ممثلي النظام 17
2.3 المتطلبات الوظيفية للنظام 17
2.4 المتطلبات غير الوظيفية للنظام 18
2.5 نموذج الاستخدام (use case diagram) 19
2.6 وصف المتطلبات الوظيفية للنظام 21
الفصل الثالث: تصميم قاعدة البيانات 29
3.1 مقدمة 30
3.2 Entity Relationships Diagram (ERD) 30
3.3 Database Mapping 31
3.4 وصف جداول قاعدة البيانات 32
الفصل الرابع: تصميم النظام 39
4.1 المقدمة 40
4.2 المصادر البرمجية لبناء النظام 40
4.3 نموذج معمارية النظام 41
4.3.1 Model 42
4.3.2 View 42
4.3.3 Controller 42
4.4 واجهات النظام (Interfaces) 42
4.5 الاكواد البرمجية 49
4.6 ملخص الفصل 51

الفصل الخامس: فحص النظام 52
5.1 مقدمة: 53
5.2 فحص النظام 53
5.2.1 فحص وحدات النظام 53
5.2.2 فحص تكامل النظام 56
5.3 ملخص الفصل 62
الفصل السادس: الخاتمة والخطط المستقبلية 63
6.1 الخاتمة 64
6.2 الخطط المستقبلية 64
المراجع 65

قائمة الأشكال

الشكل 1.1: مخطط السياق (context diagram) 14
الشكل 1.2: المخطط الزمني لتوزيع المهام على الأسابيع (Gantt Chart) 15
الشكل 2.1: مخطط الاستخدام (use case diagram) 20
الشكل 2.2: حجز المواعيد (Sequence diagram for) 21
الشكل 2.3: (إرسال نتائج الفحوصات الطبية)Sequence diagram for 28
الشكل 3.1: مخطط الكيانات -العلاقات (EERD Diagram) 30
الشكل 3.2: Database Mapping 31
الشكل 3.3: نموذج معمارية MVC 38
الشكل 4.1: شكل برنامج VSCode 40
الشكل 4.2: شكل برنامج phpMyAdmin 40
الشكل 4.3: ايقونة إطار العمل larval 41
الشكل 4.4: نموذج معمارية النظام 42
الشكل 4.5: واجهة تسجيل الدخول 43
الشكل 4.6: واجهة لوحة تحكم مسؤول النظام 43
الشكل 4.7: واجهة الطبيب لمسؤول النظام 44
الشكل 4.8: واجهة التسجيل في النظام (اختيار الشخصية) 44
الشكل 4.9: واجهة تسجيل المريض 45
الشكل 4.10: واجهة الصفحة الرئيسية للمريض 45
الشكل 4.11: واجهة صفحة الطبيب 46
الشكل 4.12: واجهة الملف الشخصي للطبيب 46
الشكل 4.13: واجهة صفحة حجز المواعيد 47
الشكل 4.14: واجهة صفحة التسجيل للطبيب 47
الشكل 4.15: الواجهة الرئيسية للطبيب 48
الشكل 4.16: كود صفحة تسجيل الدخول لمسؤول النظام 49
الشكل 4.17: كود معالجة طلب تسجيل الدخول لمسؤول النظام 50
الشكل 4.18: كود مسارات تسجيل الدخول لمسؤول النظام 50
الشكل 5.1: واجهة حالة الطلب عند الطبيب بعد التسجيل في النظام 53
الشكل 5.2: واجهة طلبات الانضمام عند مسؤول النظام 54
الشكل 5.3: واجهة تسجيل الدخول 54
الشكل 5.4: واجهة تسجيل الدخول لمسؤول النظام 55
الشكل 5.5: واجهة ادخال خاطئ لاسم مستخدم او كلمة مرور 55
الشكل 5.6: واجهة طلب كلمة مرور جديدة 56
الشكل 5.7: واجهة حجز المواعيد الخاصة بالمريض 56
الشكل 5.8: واجهة اشعارات الطبيب 57

الشكل 5.9: واجهة المحادثات بين الطبيب والمريض 57
الشكل 5.10: واجهة إدارة مواعيد العيادة 58
الشكل5.11 : واجهة إدارة مواعيد المرضى 58
الشكل 5.12: واجهة انشاء الوصفات الطبية 59
الشكل 5.13: واجهة الوصفات الطبية عند المريض 59
الشكل 5.14: واجهة انشاء طلب فحص طبي عند الطبيب 60
الشكل 5.15: واجهة طلبات الفحوصات الطبية عند الطبيب 60
الشكل 5.16: واجهة انشاء فحوصات طبية عند فني المختبر 61
الشكل 5.17: واجهة نتائج الفحوصات عند فني المختبر 61
الشكل 5.18: واجهة نتائج الفحوصات غير المكتملة عند الطبيب 62
الشكل 5.19: واجهة نتائج الفحوصات المكتملة عند الطبيب 62

قائمة الجداول

الجدول 1.1: تقسيم المهام 15
الجدول 2.1: حجز المواعيد 21
الجدول 2.2: الوصول إلى معلومات الطبيب والعيادة 22
الجدول 2.3: الوصول إلى سجلات الصحية الشخصية 22
الجدول 2.4: التواصل بين المريض والطبيب 23
الجدول 2.5: الوصول إلى سجلات الصحية الشخصية للمرضى 23
الجدول 2.6: إدارة المواعيد وجداول المرضى 24
الجدول 2.7: كتابة الوصفات الطبية 24
الجدول 2.8: طلب اجراء فحوصات طبية 25
الجدول 2.9: التواصل بين الطبيب والمريض 25
الجدول 2.10: إدارة بيانات العيادة 26
الجدول 2.11: قبول او رفض انضمام عيادة جديدة للنظام 26
الجدول 2.12: ادارة حسابات المستخدمين 27
الجدول 2.13: إرسال نتائج الفحوصات الطبية 28
الجدول 3.1: وصف جداول قاعدة البيانات 32
الجدول 3.2: وصف حقول جدول الطبيب 33
الجدول 3.3: وصف حقول جدول مدير النظام 33
الجدول 3.4: وصف حقول جدول المريض 34
الجدول 3.5: وصف حقول جدول السجلات الطبية 34
الجدول 3.6: وصف حقول جدول فني المختبر 35
الجدول 3.7: وصف حقول جدول المواعيد 35
الجدول 3.8: وصف حقول جدول الوصفات الطبية 36
الجدول 3.9: وصف حقول جدول الاختبار 36
الجدول 3.10: وصف حقول جدول التقييمات 37
الجدول 3.11: وصف حقول جدول العيادة 37
الجدول 3.12: وصف حقول جدول مواعيد العيادة 37
الجدول 3.13: وصف حقول جدول الاشعارات 38
الجدول 3.14: وصف حقول جدول الرسائل 38

الفصل الأول

المقدمة

المحتويات:

1.1 مقدمة

1.2 فكرة المشروع

1.3 المشكلة

1.4 الدوافع والمحفزات

1.5 الأهداف

1.6 نظرة عامة على المشروع

1.7 نطاق المشروع

1.8 منهجية المشروع

1.9 أهمية المشروع

1.10 البدائل العامة

1.11 النظام المقترح

1.12 Context diagram

1.13 جدولة المهام والمخطط الزمني للمهام

1.14ملخص الفصل

1.1 مقدمة

يقدم هذا الفصل نظرة شاملة على المشروع، بدءًا من فكرته والمشكلة التي يعالجها، مرورًا بالدوافع والأهداف الرئيسية. كما يحدد نطاق المشروع ويوضح النظام المقترح مقارنة بالبدائل المتاحة. بالإضافة إلى ذلك، يستعرض الفصل المنهجية المتبعة، المخطط السياقي لتدفق البيانات، والجدولة الزمنية للمهام، وينتهي بملخص لأهم النقاط.

1.2 فكرة المشروع

في عصرنا الحالي، أصبحت التكنولوجيا جزءاً لا يتجزأ من حياتنا اليومية، حيث تشهد تطوراً سريعاً ومستمراً يؤثر على جميع القطاعات، بما في ذلك الصحة والتعليم والصناعة والخدمات. تطورت التقنيات الحديثة لتشمل الذكاء الاصطناعي، وإنترنت الأشياء، والحوسبة السحابية، مما ساهم في توفير حلول مبتكرة تُحسن من الكفاءة وتُسهل العمليات المختلفة.

في مجال الرعاية الصحية، تلعب التكنولوجيا دوراً محورياً في تطوير أدوات تشخيصية دقيقة، وتحسين إدارة البيانات الصحية، وتسهيل تواصل المرضى مع الأطباء من خلال التطبيقات والمنصات الرقمية. ومع هذا التقدم، أصبح توفير أنظمة متكاملة ضرورة لمواكبة هذه التطورات وتلبية احتياجات الأفراد والمؤسسات بشكل أفضل.

من هذا المنطلق، جاءت فكرة مشروعنا الذي يهدف إلى الاستفادة من هذه التقنيات الحديثة لحل التحديات التي تواجه قطاع العيادات الطبية، وتقديم تجربة متطورة تُسهم في رفع مستوى الرعاية الصحية في فلسطين.

1.3 المشكلة

في ظل التعقيد الذي تواجهه عملية حجز المواعيد في العيادات الخارجية، يعتمد الكثيرون على الاتصالات الهاتفية، حيث يتطلب الأمر التواصل مع موظف لتحديد نوع الخدمة المطلوبة، واختيار العيادة أو الطبيب، أو حتى التحدث مباشرة مع الطبيب. هذه الطريقة غالبًا ما تكون بطيئة وغير فعّالة، وقد لا تلبي احتياجات المريض بالشكل المطلوب.

أما من جهة الطبيب، فالتحديات لا تقل تعقيدًا، حيث يفتقر إلى معلومات كافية عن المريض أثناء الحجز، مما يضطره لجمع البيانات بشكل يدوي. إضافةً إلى ذلك، عند الحاجة لتغيير جدول المواعيد أو إلغاء أحدها، يتوجب على الطبيب التنسيق مع المرضى بشكل مباشر، مما يستهلك وقتًا وجهدًا إضافيين.

لحل هذه المشكلات، قمنا بتصميم موقع إلكتروني مبتكر يُحدث نقلة نوعية في طريقة إدارة المواعيد. يُمكن الموقع المرضى من حجز مواعيدهم بسهولة، مع إمكانية اختيار العيادة والطبيب المناسب، والوصول إلى خدماتهم بشكل مباشر. في المقابل، يوفر للأطباء أداة فعالة لإدارة جداولهم، والوصول إلى معلومات المرضى بسهولة، مما يضمن عملية أكثر تنظيمًا وسلاسة للطرفين.

1.4 الدوافع والمحفزات

في ظل الحاجة إلى تصميم أنظمة مبتكرة تسهم في تحسين الكفاءة وتلبية احتياجات المستخدمين. جاء مشروعنا استجابة لهذه المتطلبات. وفيما يلي أهم الدوافع لأنشاء بوابة العيادة الطبية:

1. تحسين الوصول للرعاية الصحية: يمكن للمرضى حجز مواعيدهم بسهولة عبر الإنترنت، مما يقلل من الانتظار ويحسن الوصول للخدمات الطبية.
2. تبسيط عملية التواصل: يمكن للمرضى التواصل مع الأطباء وعرض نتائج الفحوصات والاستشارات بسهولة، مما يعزز التواصل المباشر وفهم أفضل لحالتهم الصحية.
3. تقديم خدمات متكاملة: توفير خدمات مثل عرض المواعيد والوصول إلى السجلات الطبية وتحديثها، وإدارة المواعيد والإشعارات، وإمكانية طلب وتوصيل الوصفات الطبية.
4. توفير الكلفة: تقليل التكاليف المرتبطة بالزيارات الطبية الدورية والتنقلات، مما يعود بالفائدة على المرضى والمؤسسات الطبية على حد سواء.
5. زيادة التواجد الرقمي: يمكن للعيادات الخارجية زيادة تواجدها الرقمي عبر الإنترنت، مما يعزز من فرص جذب المرضى وتقديم خدماتها بشكل أوسع.

1.5 الأهداف

يهدف مشروعنا إلى تقديم حل رقمي مبتكر يعالج التحديات التي تواجه العيادات الخارجية، من خلال تحسين تجربة المرضى وتبسيط العمليات اليومية للأطباء والإداريين. يعتمد النظام على الاستفادة من التكنولوجيا الحديثة لتقديم خدمات أكثر كفاءة ومرونة. تتلخص أهداف المشروع فيما يلي:

1. تسهيل الحجز: يمكن للمرضى حجز مواعيدهم بسهولة من خلال النظام، مما يوفر الوقت والجهد لهم وللموظفين.
2. تحسين الاتصال: يسمح النظام بتواصل أفضل بين المرضى والأطباء، مما يعزز تجربة الرعاية الصحية.
3. توفير معلومات دقيقة: يساعد النظام في توفير معلومات دقيقة ومحدثة حول مواعيد العيادات وتاريخ العلاج والوصفات الطبية.
4. زيادة كفاءة العمل: يمكن للأطباء إدارة جداولهم بشكل أفضل وتحسين تنظيمها من خلال النظام، مما يؤدي إلى زيادة الكفاءة والإنتاجية.
5. تحسين رضا المرضى: من خلال توفير تجربة سلسة ومريحة، يمكن للنظام زيادة رضا المرضى وتعزيز سمعة المستشفى.
6. تقليل تكاليف الملفات الورقية، والتعامل معها
7. تسهيل البحث في السجل الطبي للمريض، وتقليل الوقت الضائع بسبب البحث في الملفات الورقية
8. تسهيل استرجاع المعلومات الضرورية والأساسية للمرضى في حالات الطوارئ.

1.6 نظرة عامة على المشروع

في هذا المشروع، سنقوم بتطوير نظام إلكتروني متكامل لإدارة العيادات الطبية. سيتم تحليل العمليات والإجراءات المستخدمة حاليًا في إدارة العيادات الطبية، ثم سنعمل على تطوير نظام إلكتروني يسهل ويحسن عملية تقديم الخدمات الصحية وتفاعل الأطباء والمرضى .

1.7 نطاق المشروع

يستهدف المشروع في بدايته مجموعة من سكان مدينة الخليل، حيث سيقوم فريق المشروع بتطبيق النظام في بعض العيادات الخاصة والمستشفيات داخل الخليل، وبعد نجاح المشروع سوف يتم تطبيقه في عده مستشفيات حتى يصبح المشروع في كافة عيادات ومستشفيات فلسطين.

1.8 منهجية المشروع

سوف يتبع فريق المشروع منهجية الشلال (Waterfall Model) لتحقيق المخرجات المتوقعة من النظام. تبدأ هذه المنهجية بمرحلة التخطيط للنظام، ومن ثم تحليل المتطلبات، يليها تصميم النظام، ثم تطويره وتشغيله، ثم فحصه، وتنتهي بتطبيق النظام وصيانته. يمكن الرجوع إلى أي مرحلة من هذه المراحل وإجراء التعديلات عليها في أي وقت خلال تنفيذ المشروع.

في هذه المنهجية، يتم التركيز بداية على جمع أكبر عدد من المعلومات والمشكلات التي تواجه المؤسسات الطبية في حفظ بيانات المرضى، والتعرف على مشكلات الأنظمة التقليدية من خلال زيارة المؤسسات الطبية مثل المستشفيات والعيادات الخاصة، والاهتمام بالمشكلات التي تواجه الأنظمة التقليدية لتداركها في النظام الإلكتروني لبناء نظام إلكتروني متكامل.

1.9 أهمية المشروع

تلعب بوابة العيادة الطبية دورًا حيويًا في تحسين جودة الخدمات الصحية وتسهيل العمليات الإدارية والطبية على حد سواء. يساهم النظام في معالجة التحديات التي تواجه المرضى والأطباء وإدارة العيادات، ويُحدث نقلة نوعية في تجربة المرضى من خلال تسهيل حجز المواعيد، متابعة توافر الأطباء والعيادات، وتعزيز الثقة في الخدمة. كما يُمكّن الأطباء من التواصل الفعّال مع المرضى، الوصول السريع إلى بياناتهم، وإدارة جداولهم بشكل أكثر كفاءة، مما يُحسن من جودة القرارات الطبية. النظام أيضًا يقلل من الإجراءات اليدوية المرهقة ويُبسط إدارة المواعيد، مما يوفر الوقت والجهد للطرفين.

1.10 البدائل العامة

بعد دراسة الأنظمة الحالية في العيادات، تبين أن المرضى يواجهون صعوبة في حجز المواعيد وإدارة سجلاتهم الورقية، كما يجد الموظفون صعوبة في تنظيم بيانات المرضى والمواعيد والتقارير الطبية. لذا، اقترح فريق المشروع تطوير نظام شامل لحجز المواعيد إلكترونيًا، وإدارة السجلات الطبية، وتعزيز التواصل بين المرضى والأطباء، مما يواكب التطور التكنولوجي ويسهل الخدمات الطبية.

تتمثل البدائل العامة في:

1) **نظام إلكتروني (Website)**

- **ميزات هذا الموقع:**

- سهولة الاستخدام من قبل المستخدمين.

- الوصول إلى الموقع من أي نوع من الأجهزة (أجهزة الكمبيوتر والأجهزة الذكية).

- سهولة التحديث في أي وقت.

- متوافق مع جميع أنظمة التشغيل.

- يوفر واجهة متجاوبة (Responsive) عند الوصول إليه من الهواتف المحمولة.

- **عيوب هذا الموقع:**

- الحاجة الدائمة للاتصال بالأنترنت.

2) **تطبيق للهواتف المحمولة**

- **ميزات هذا التطبيق:**

- سهولة الوصول إلى التطبيق دائمًا من خلال الهواتف المحمولة.

- يتيح تجربة مستخدم مخصصة ومحسّنة على الهاتف.

- **عيوب هذا التطبيق:**

- الحاجة الدائمة للاتصال بالأنترنت.

- لا يمكن الوصول اليه من أي نوع من الأجهزة، فقط باستخدام الهواتف المحمولة.

- يتطلب مساحة تخزين على الأجهزة المحمولة لتحميل التطبيق.

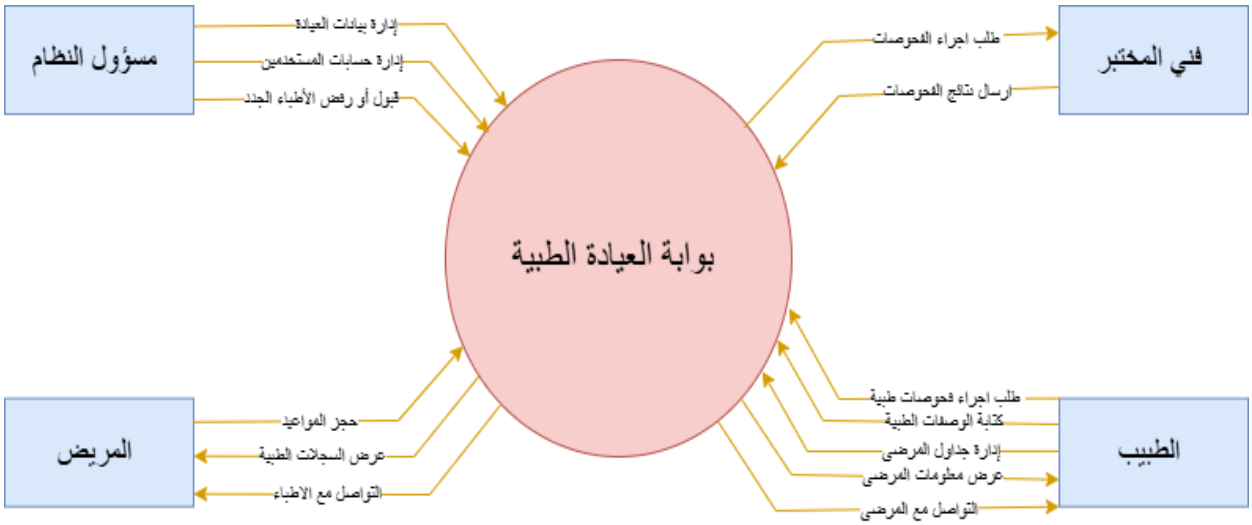
بعد النظر إلى البدائل المقترحة، تم اختيار النظام الإلكتروني (Website) لبناء النظام بسبب مزاياه التي تشمل التوافق مع جميع الأجهزة وسهولة التحديث، مما يجعله أكثر مرونة وتوسعًا. أما تطبيق الهاتف المحمول فيُعد خيارًا مستقبليًا في حال الحاجة إليه بناءً على متطلبات المستخدمين.

1.11 النظام المقترح

بعد مراجعة الحلول البديلة وطرحها على بعض العاملين في المجال الطبي، تم اختيار الموقع الإلكتروني ليكون النظام المقترح، وذلك لعدة أسباب منها إمكانية الوصول وتصفحه من خلال أي نوع من الأجهزة بالإضافة للهواتف المحمولة، بالإضافة إلى أن المستخدم الذي يعمل في المجال الطبي يفضل استخدام الحاسوب لتصفح الموقع والعمل عليه. وسيقوم فريق المشروع بالعمل على تنفيذ النظام المقترح.

Context diagram 1.12

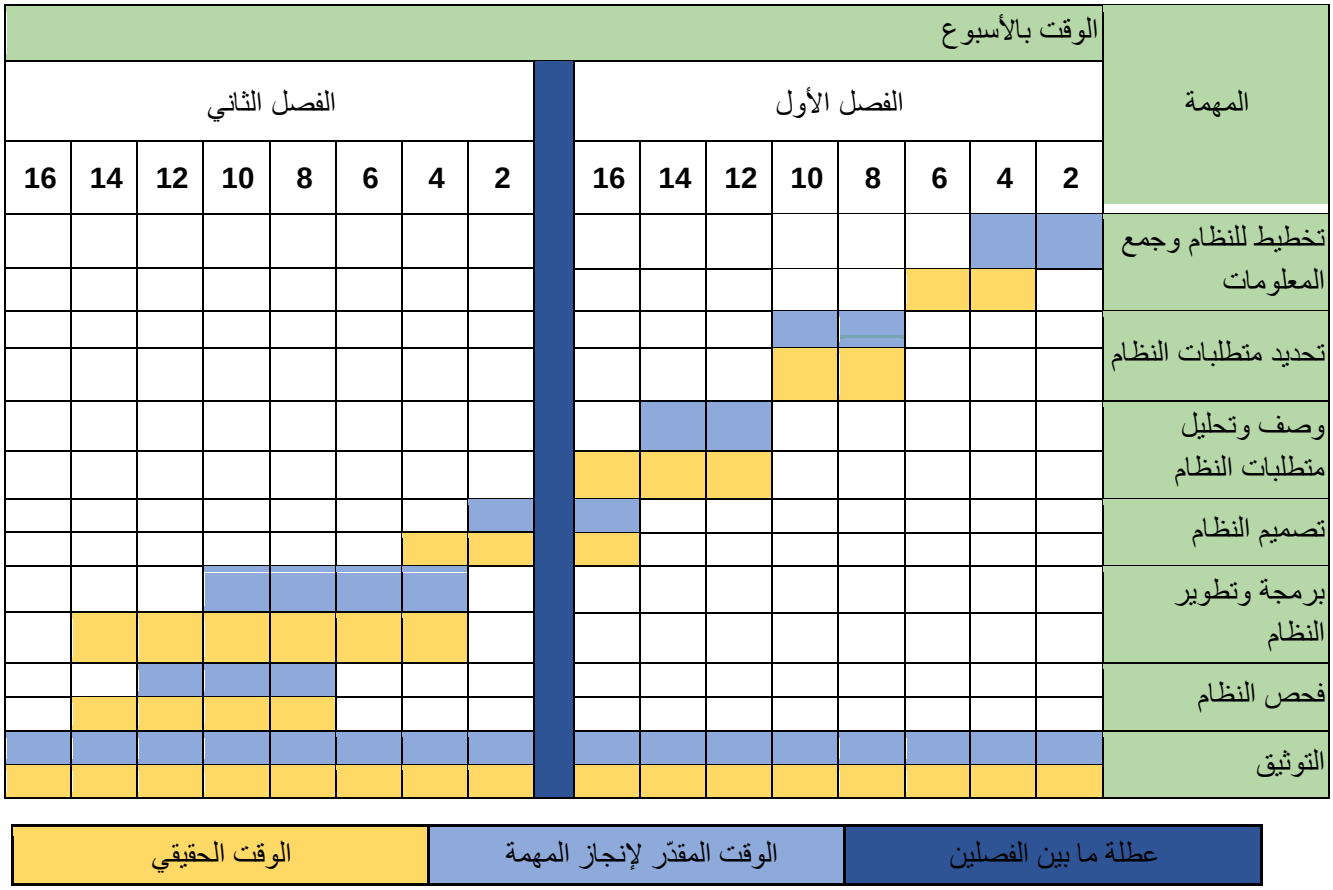
يوضح الشكل (1) context diagram الخاص بالمشروع، حيث يبيّن أهم الاعمال والعناصر المحيطة بالنظام أو الأفراد وآلية تفاعلها مع النظام، وهم بشكل أساسي: المريض، فني المختبر، مسؤول العيادة، الطبيب



الشكل (1.1): مخطط السياق (context diagram)

1.13 جدولة المهام والمخطط الزمني للمهام

يوضح الشكل (2) توزيع مراحل المشروع على مدار الفصلين الدراسيين، مع ذكر المواعيد الفعلية والمقدّرة لإنجاز المهام:



الشكل (1.2): المخطط الزمني لتوزيع المهام على الأسابيع(Gantt Chart)

الجدول (1.1): تقسيم المهام

رقم المهمة	اسم المهمة	الوقت بالأسابيع
1	تخطيط للنظام وجمع المعلومات	2
2	تحديد متطلبات النظام	2
3	وصف وتحليل متطلبات النظام	3
4	تصميم النظام	3
5	برمجة وتطوير النظام	6
6	فحص النظام	4
7	التوثيق	طول فترة العمل على المشروع

1.14 ملخص الفصل

قمنا في هذا الفصل عرض الأنظمة المستخدمة حالياً ومشاكلها ووضعنا بدائل لتشمل نظام الكتروني، أهداف المشروع، نطاق النظام، بالإضافة الى المنهجية المتبعة في بناء النظام وأيضا توزيع الأدوار وعرض المخطط الزمني للمشروع.

الفصل الثاني

تحليل المتطلبات

المحتويات:

2.1 مقدمة

2.2 متطلبات النظام

2.3 ممثلي النظام والمتطلبات الوظيفية للنظام

2.4 المتطلبات غير الوظيفية للنظام

2.5 Use Case Diagram

2.6 وصف المتطلبات الوظيفية للنظام

2.1 مقدمة

في هذا الفصل سيتم توضيح ممثلي النظام والمتطلبات الوظيفية الخاصة بهم، والمتطلبات غير الوظيفية التي نوفرها في مشروعنا، بالإضافة إلى نموذج الاستخدام (use case diagram) مع وصف للمتطلبات.

2.2 ممثلي النظام

1. المريض
2. الطبيب
3. مسؤول النظام
4. فني المختبر

2.3 المتطلبات الوظيفية للنظام

تم تقسيم المتطلبات الوظيفية وتصنيفها اعتماداً على وظائف مستخدمي النظام وهي على الشكل الاتي:

المتطلبات الوظيفية خاصة بالمريض:

1. حجز المواعيد.
2. الوصول إلى معلومات الطبيب والعيادة.
3. الوصول إلى سجلات الصحية الشخصية.
4. التواصل بين المريض والطبيب.

المتطلبات الوظيفية خاصة بالطبيب:

3. الوصول إلى سجلات الصحية الشخصية للمرضى.
4. إدارة المواعيد وجداول المرضى.
5. كتابة الوصفات الطبية.
6. طلب اجراء الفحوصات الطبية.
7. التواصل بين الطبيب والمريض.

المتطلبات الوظيفية خاصة بمسؤول النظام:

1. ادارة بيانات العيادة.
2. قبول او رفض انضمام عيادة جديدة للنظام.
3. ادارة حسابات المستخدمين.

المتطلبات الوظيفية خاصة بفني المختبر:

1. ارسال نتائج الفحوصات الطبية الى النظام.

2.4 المتطلبات غير الوظيفية للنظام

مجموعة من المعايير المتعارف عليها والتي تجعل النظام أكثر مرونة وسهولة في الاستخدام.

❖ **سهولة التعامل مع النظام:**

استنادا إلى التوجيهات والتعليمات المتعلقة في واجهة تصميم النظام، يجب عند قيام فريق المشروع بإنشاء النظام مراعاة ما يلي:

1. أن تكون الواجهة سهلة التعامل والاستخدام لمستخدمي النظام، والوصول إلى جميع مميزات الواجهة دون عوائق، مثل التصميم المعقد، البطء في الاستجابة، وعدم التوافق مع الأجهزة المختلفة. لضمان تجربة مستخدم سلسة وواضحة.
2. أن تكون الألوان التي تستخدم في النظام مريحة للعين.
3. يجب أن يوافق التصميم فكرة النظام والتركيز على الفئات المستهدفة ومراعاة المستويات المختلفة، من خلال تبسيط الواجهة للمستخدمين العاديين مثل المرضى، وتوفير واجهات متقدمة للأطباء والموظفين. وأيضاً تصميمها بما يناسب الأجهزة التي يستخدمها النظام.

❖ **الأمان وسرية البيانات:**

يجب حماية النظام من المخاطر التي تواجهه من سرقة بيانات النظام وهجمات إلكترونية وغيرها، وذلك من خلال تطبيق هذه الأساليب عند إنشاء الموقع الإلكتروني:

1. عدم السماح بدخول النظام لغير المصرح لهم بذلك.
2. عند القيام بتقديم طلب الانضمام للنظام من قبل العاملين في المجال الطبي يُطلب منهم إدخال بيانات شخصية ومجموعة من الوثائق لإثبات هويتهم، مثل صورة عن شهادة مزاولة المهنة وغيرها من البيانات، ثم يقوم العاملين في قسم المتابعة والإشراف بتأكد من هذه المعلومات.
3. حماية النظام من الهجمات مثل هجمات SQL injection وغيرها.

❖ **الكفاءة العالية للنظام وسرعة الاستجابة:**

تتضح كفاءة النظام عندما يستطيع التعامل مع العديد من الطلبات في وقت واحد، مثل الانضمام والتسجيل لأكثر من شخص، وإنشاء سجلات طبية متعددة في نفس الوقت. يعتمد النظام على تقنيات فعّالة لإدارة البيانات والعمليات المتزامنة، مما يضمن تلبيته لجميع الأوامر بسرعة ودقة. على سبيل المثال، يمكن للنظام معالجة طلبات التسجيل وتحديث البيانات في الخلفية دون التأثير على أداء واجهة المستخدم، مما يساهم في تقديم تجربة سلسة للمستخدمين دون تأخير أو مشاكل.

❖ **القابلية لتطوير النظام وصيانته:**

يجب أن يكون النظام مصمماً لإجراء صيانة دورية من خلال بناء نظام يمتلك مرونة عالية في التعامل مع تطويره والتعديل عليه لاحقاً.

2.5 نموذج الاستخدام (use case diagram)

فيما يلي تم توضيح تصورات مستخدمي النظام لكل طرف من الأطراف التي يتعامل معها النظام، مع ذِكر الصلاحيات التي يمكنهم الوصول إليها من خلال الموقع والخدمات التي تتوفر لكلٍ منهم:

المريض:

بصفتي مستخدم يمكنني الدخول إلى نظام العيادة باستخدام رابط صالح، وتسجيل الدخول إلى حسابي وعرض الصفحة الرئيسية، بمجرد تسجيل الدخول سوف أتمتع بمزايا الويب التي تتيح لي قائمة تتضمن خانة العيادات والسجلات الصحية الشخصية بالإضافة إلى خانة الفحوصات الطبية.
من خلال هذه القائمة يمكنني الوصول لخانة العيادات التي من خلالها اتمكن من حجز المواعيد والتواصل مع الاطباء

الطبيب:

بصفتي الطبيب، يجب أن أقوم بتسجيل الدخول كطبيب لعرض جدول المواعيد وادارتها، كما وان صلاحياتي تسمح لي بالوصول الى سجلات المريض الشخصية، بالإضافة الى كتابة الوصفات الطبية وطلب اجراء الفحوصات الطبية، والتواصل مع المرضى.

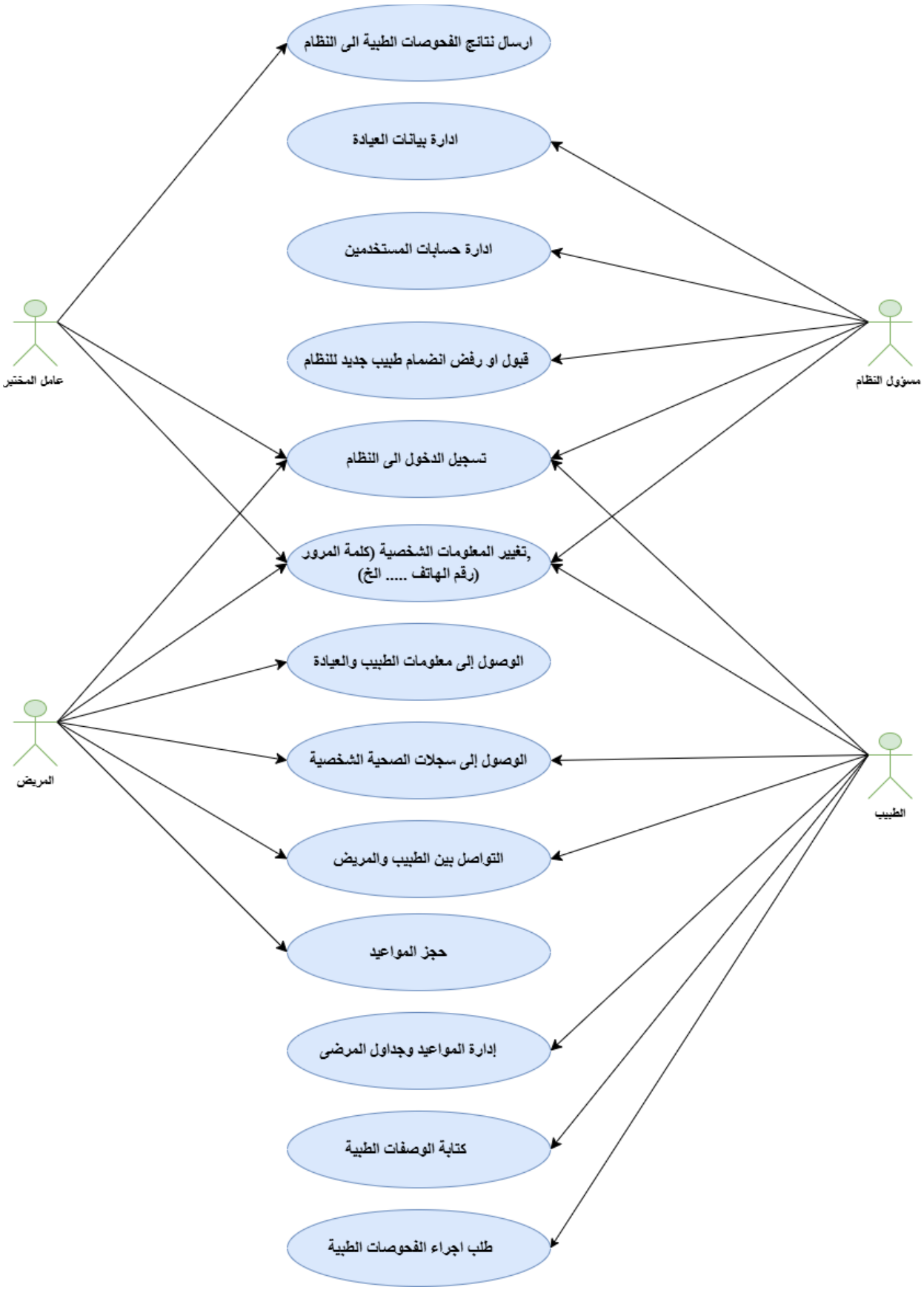
مسؤول النظام :

بصفتي المسؤول، فأنا أملك لوحة تحكم كاملة بالموقع، يجب أن أقوم بتسجيل الدخول باستخدام حساب المسؤول لعرض جميع بيانات العيادة وحسابات المستخدمين، كما وان من صلاحياتي تسمح لي بإدارة بيانات العيادة وحسابات المستخدمين وقبول او رفض انضمام العيادات الى النظام حسب شروط معينة.

فني المختبر :

بصفتي فني المختبر، يجب أن أقوم بتسجيل الدخول فتي المختبر لأستطيع القيام بأرسال نتائج الفحوصات الطبية الى النظام.

يوضح الشكل (3) نموذج الاستخدام العام لنظام:



الشكل (2.1): مخطط الاستخدام (use case diagram)

2.6 وصف المتطلبات الوظيفية للنظام

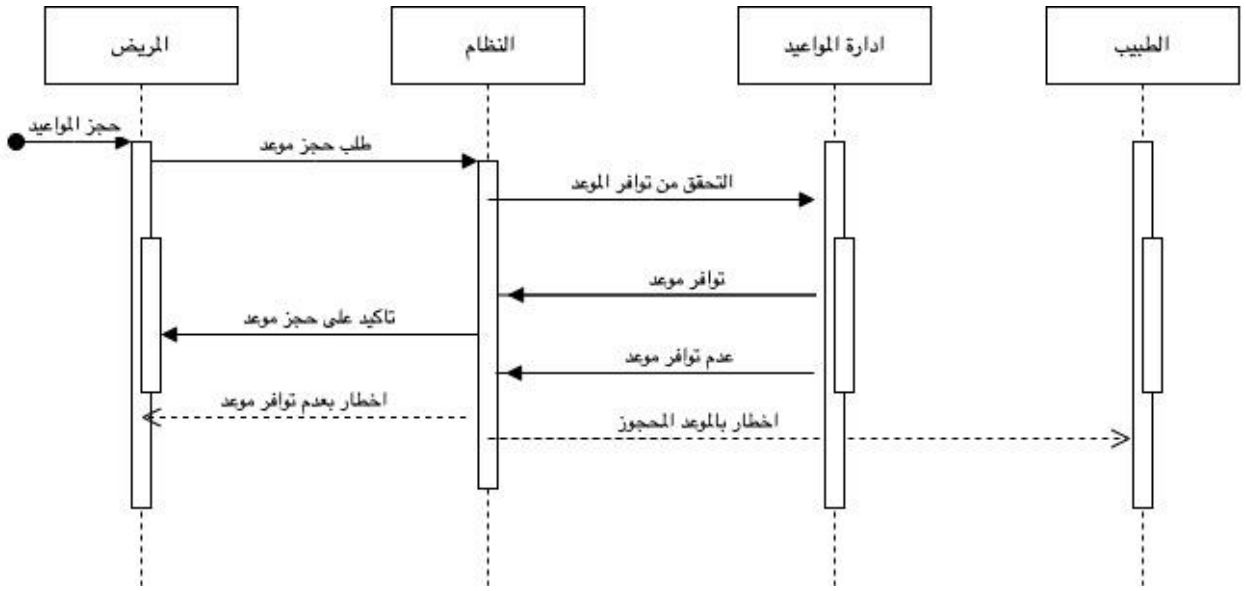
في هذا القسم، تم توضيح التفاصيل الخاصة لكل متطلب وظيفي تم ذكره في نموذج الاستخدام، لتتكون صورة واضحة عن كيفية بناء كل متطلب من المتطلبات المذكورة.

- **<u>وصف متطلبات النظام الخاصة بالمريض</u>**

1. حجز المواعيد

الجدول (2.1): حجز المواعيد

اسم النظام	حجز المواعيد
المستخدم الرئيسي	المريض
الهدف	إتاحة حجز موعد طبي في العيادة
الشروط	• يجب على المريض أن يكون مسجلاً في النظام • يجب أن تكون هناك مواعيد متاحة للحجز
الاجراءات	• المريض يقوم بتسجيل الدخول إلى النظام • المريض يختار التاريخ والوقت المناسبين له • الطبيب يؤكد الحجز ويعرض تأكيد الحجز للمريض
الاستثناءات	إذا لم تكن هناك مواعيد متاحة، يظهر رسالة تفيد بعدم توفر مواعيد حاليًا
حل الاستثناءات	يمكن للمريض اختيار موعد آخر متاح أو العودة لاحقًا للحجز



الشكل (2.2): (حجز المواعيد) Sequence diagram for

2. الوصول إلى معلومات الطبيب والعيادة

الجدول (2.2): الوصول إلى معلومات الطبيب والعيادة

اسم النظام	الوصول إلى معلومات الطبيب والعيادة
المستخدم الرئيسي	المريض
الهدف	عرض معلومات الطبيب والعيادة للمريض
الشروط	• يجب على المريض أن يكون مسجلاً في النظام • يجب أن يكون للمريض صلاحيات الوصول إلى معلومات الطبيب والعيادة
الاجراءات	• المريض يقوم بتسجيل الدخول إلى النظام • المريض يقوم بالبحث عن اسم الطبيب أو اسم العيادة • النظام يعرض معلومات الطبيب والعيادة للمريض
الاستثناءات	إذا لم يتم العثور على معلومات الطبيب أو العيادة المطلوبة
حل الاستثناءات	يمكن للمريض تحديد معايير البحث الأخرى أو العودة للبحث لاحقًا

3. الوصول إلى سجلات الصحية الشخصية

الجدول (2.3): الوصول إلى سجلات الصحية الشخصية

اسم النظام	الوصول إلى سجلات الصحية الشخصية
المستخدم الرئيسي	المريض
الهدف	السماح للمريض بالوصول إلى سجلاته الصحية الشخصية
الشروط	• يجب أن يكون المريض مسجلاً في النظام • يجب أن يكون للمريض صلاحيات الوصول إلى سجلاته الصحية الشخصية
الاجراءات	• المريض يقوم بتسجيل الدخول إلى النظام • المريض ينتقل إلى قسم سجلات الصحية الشخصية • النظام يعرض سجلات الصحية الشخصية للمريض
الاستثناءات	إذا لم تكن هناك سجلات صحية شخصية متاحة للمريض في النظام
حل الاستثناءات	يمكن للمريض التواصل مع الجهة المسؤولة لتوفير سجلاته الصحية الشخصية

4. التواصل بين المريض والطبيب

الجدول (2.4): التواصل بين المريض والطبيب

اسم النظام	التواصل بين المريض والطبيب
المستخدم الرئيسي	المريض
الهدف	تيسير عملية التواصل بين المريض والطبيب وتبادل المعلومات الصحية اللازمة
الشروط	• يجب أن يكون المريض مسجلاً في النظام • يجب أن يكون للمريض صلاحيات التواصل مع الطبيب
الاجراءات	• المريض يقوم بتسجيل الدخول إلى النظام • المريض يختار الطبيب الذي يرغب في التواصل معه • المريض يكتب رسالته أو استفساره ويقوم بإرسالها للطبيب • الطبيب يستقبل الرسالة ويقوم بالرد على المريض • يمكن للمريض الرجوع إلى سجل الرسائل لمتابعة المحادثة
الاستثناءات	في حالة عدم تواجد الطبيب في النظام أو عدم استجابته لفترة طويلة
حل الاستثناءات	يمكن للمريض التواصل مع مسؤول النظام للمساعدة في التواصل مع الطبيب

- **<u>وصف متطلبات النظام الخاصة بالطبيب</u>**

1. الوصول إلى سجلات الصحية الشخصية للمرضى

الجدول (2.5): الوصول إلى سجلات الصحية الشخصية للمرضى

اسم النظام	الوصول إلى سجلات الصحية الشخصية للمرضى
المستخدم الرئيسي	الطبيب
الهدف	توفير وصول آمن وفعال إلى سجلات الصحية الشخصية للمرضى
الشروط	• يجب أن يكون الطبيب مسجلاً في النظام • يجب أن يكون للطبيب صلاحيات الوصول إلى سجلات الصحية للمرضى
الاجراءات	• الطبيب يقوم بتسجيل الدخول إلى النظام • الطبيب يبحث عن ملف المريض الذي يرغب في الوصول إليه • الطبيب يعرض سجلات الصحة الشخصية للمريض، مثل الفحوصات والتشخيصات السابقة • يمكن للطبيب إضافة ملاحظات أو تحديثات إلى سجل الصحة للمريض • يمكن للطبيب طباعة نسخة من سجل الصحة للمريض إذا لزم الأمر
الاستثناءات	في حالة عدم وجود ملف للمريض أو عدم توافر البيانات اللازمة في الملف
حل الاستثناءات	يمكن للطبيب الاتصال بمسؤول النظام لمساعدته في حل المشكلة

2. إدارة المواعيد وجداول المرضى

الجدول (2.6): إدارة المواعيد وجداول المرضى

اسم النظام	إدارة المواعيد وجداول المرضى
المستخدم الرئيسي	الطبيب
الهدف	تنظيم مواعيد المرضى وجداولهم بشكل فعال
الشروط	• يجب أن يكون الطبيب مسجلاً في النظام • يجب أن يكون للطبيب صلاحيات الوصول لإدارة المواعيد وجداول المرضى
الاجراءات	• الطبيب يقوم بتسجيل الدخول إلى النظام • الطبيب يقوم بفتح قائمة المواعيد وجداول المرضى • الطبيب يقوم بإضافة موعد جديد لمريض أو تعديل موعد حالي • الطبيب يمكنه عرض جدول يومي أو أسبوعي لمواعيد المرضى • يمكن للطبيب إلغاء مواعيد المرضى إذا لزم الأمر • الطبيب يستطيع تأكيد مواعيد المرضى عند حضورهم للعيادة
الاستثناءات	في حالة عدم توافر موعد في الوقت المحدد أو تضارب في المواعيد
حل الاستثناءات	يمكن للطبيب إعادة جدولة الموعد أو توجيه المريض لموعد آخر

3. كتابة الوصفات الطبية

الجدول (2.7): كتابة الوصفات الطبية

اسم النظام	كتابة الوصفات الطبية
المستخدم الرئيسي	الطبيب
الهدف	تسهيل عملية كتابة وصفات طبية دقيقة وفعالة للمرضى.
الشروط	• يجب أن يكون الطبيب مسجلاً في النظام • يجب أن يكون للطبيب صلاحيات الوصول لكتابة الوصفات الطبية
الاجراءات	• الطبيب يقوم بتسجيل الدخول إلى النظام • الطبيب يختار مريضًا لكتابة وصفة طبية له • الطبيب يقوم بإضافة الأدوية والجرعات المناسبة للوصفة الطبية • الطبيب يمكنه إضافة تعليمات خاصة بالوصفة إذا لزم الأمر • النظام يقوم بتوليد وصفة طبية مكتملة وجاهزة للطباعة أو الإرسال الإلكتروني
الاستثناءات	في حالة تعارض الأدوية أو الجرعات مع حالة المريض أو تاريخه الطبي السابق
حل الاستثناءات	ينبغي للنظام تحذير الطبيب وعرض رسالة توضح التعارضات أو المشكلات المحتملة

4. طلب اجراء الفحوصات الطبية

الجدول (2.8): طلب اجراء الفحوصات الطبية

اسم النظام	طلب اجراء الفحوصات الطبية
المستخدم الرئيسي	الطبيب
الهدف	تسهيل عملية طلب الفحوصات الطبية المناسبة للمرضى بدقة وسهولة.
الشروط	• يجب أن يكون الطبيب مسجلاً في النظام. • يجب أن يكون للطبيب صلاحيات الوصول لطلب الفحوصات الطبية.
الاجراءات	• الطبيب يقوم بتسجيل الدخول إلى النظام • الطبيب يختار مريضًا لتقديم طلب فحوصات طبية له • الطبيب يحدد الفحوصات الطبية المناسبة بناءً على حالة المريض • الطبيب يحدد الأولويات إذا كان هناك أكثر من فحص مطلوب • الطبيب يمكنه إضافة ملاحظات خاصة على الفحوصات إذا لزم الأمر • النظام يقوم بتوليد طلب الفحوصات الطبية بشكل مكتمل وجاهز للطباعة أو الإرسال إلى المعامل الطبية
الاستثناءات	في حالة وجود تضارب بين الفحوصات المطلوبة وحالة المريض الصحية أو تاريخه الطبي.
حل الاستثناءات	ينبغي للنظام تحذير الطبيب وعرض رسالة توضح أي تضارب أو مشاكل محتملة مع الفحوصات المختارة.

5. التواصل بين الطبيب والمريض

الجدول (2.9): التواصل بين الطبيب والمريض

اسم النظام	التواصل بين الطبيب والمريض
المستخدم الرئيسي	الطبيب
الهدف	تمكين التواصل الفعال والآمن بين الطبيب والمريض
الشروط	• يجب أن يكون الطبيب والمريض مسجلين في النظام • يجب أن يكون للطبيب والمريض صلاحيات الوصول للتواصل مع بعضهما البعض
الاجراءات	• الطبيب يقوم بتسجيل الدخول إلى النظام • الطبيب يختار المريض الذي يرغب في التواصل معه • الطبيب يمكنه إرسال رسائل نصية أو صور أو ملفات للمريض • المريض يمكنه الرد على الرسائل من الطبيب وطلب مواعيد أو طلبات أخرى • يتم تسجيل كل رسالة ومحادثة بين الطبيب والمريض في سجل تواصل خاص
الاستثناءات	في حالة تأخر الرد من الطبيب على رسائل المريض
حل الاستثناءات	ينبغي للنظام تذكير الطبيب بوجود رسالة جديدة من المريض وتحفيزه على الرد

- **<u>وصف متطلبات النظام الخاصة بمسؤول النظام</u>**

1. **ادارة بيانات العيادة**

الجدول (2.10): إدارة بيانات العيادة.

اسم النظام	إدارة بيانات العيادة
المستخدم الرئيسي	مسؤول النظام
الهدف	تمكين مسؤول النظام من إدارة بيانات العيادة وضبط الإعدادات الرئيسية للنظام
الشروط	• يجب أن يكون مسؤول النظام مسجل في النظام • يجب أن يحصل مسؤول النظام على صلاحيات الوصول الكاملة لإدارة البيانات والإعدادات
الاجراءات	• مسؤول النظام يقوم بتسجيل الدخول إلى النظام • يمكن لمسؤول النظام إضافة وتحرير وحذف بيانات العيادة والموظفين والأطباء والمرضى • يمكن لمسؤول النظام ضبط الإعدادات الرئيسية للنظام مثل ساعات العمل والإعدادات المالية • يمكن لمسؤول النظام إنشاء حسابات جديدة للمستخدمين الآخرين في النظام • يمكن لمسؤول النظام توليد تقارير عن أداء العيادة ومعلومات أخرى ذات صلة
الاستثناءات	في حالة حدوث خطأ أثناء تحرير بيانات العيادة.
حل الاستثناءات	يجب على النظام توفير نظام تأكيد وتحقق لتجنب فقدان البيانات أو تكرار البيانات

2. **قبول او رفض انضمام عيادة جديدة للنظام**

الجدول (2.11): قبول او رفض انضمام عيادة جديدة للنظام

اسم النظام	قبول او رفض انضمام عيادة جديدة للنظام
المستخدم الرئيسي	مسؤول النظام
الهدف	تمكين مسؤول النظام من قبول أو رفض انضمام عيادة جديدة للنظام
الشروط	يجب أن يكون مسؤول النظام مسجل في النظام يجب أن يحصل مسؤول النظام على صلاحيات الوصول الكاملة لإدارة البيانات والإعدادات
الاجراءات	• مسؤول النظام يقوم بتسجيل الدخول إلى النظام • يمكن لمسؤول النظام الوصول إلى قائمة العيادات الجديدة المطلوب قبولها أو رفضها • مسؤول النظام يقوم بمراجعة معلومات العيادة الجديدة والتأكد من صحتها واكتمالها • يمكن لمسؤول النظام قبول أو رفض انضمام العيادة الجديدة بناءً على مراجعته للمعلومات
الاستثناءات	في حالة وجود معلومات ناقصة أو غير دقيقة في طلب انضمام العيادة
حل الاستثناءات	يجب على النظام إرسال إشعار إلى العيادة بطلب تحديث أو تعديل المعلومات لإكمال العملية

3. ادارة حسابات المستخدمين

الجدول (2.12): ادارة حسابات المستخدمين

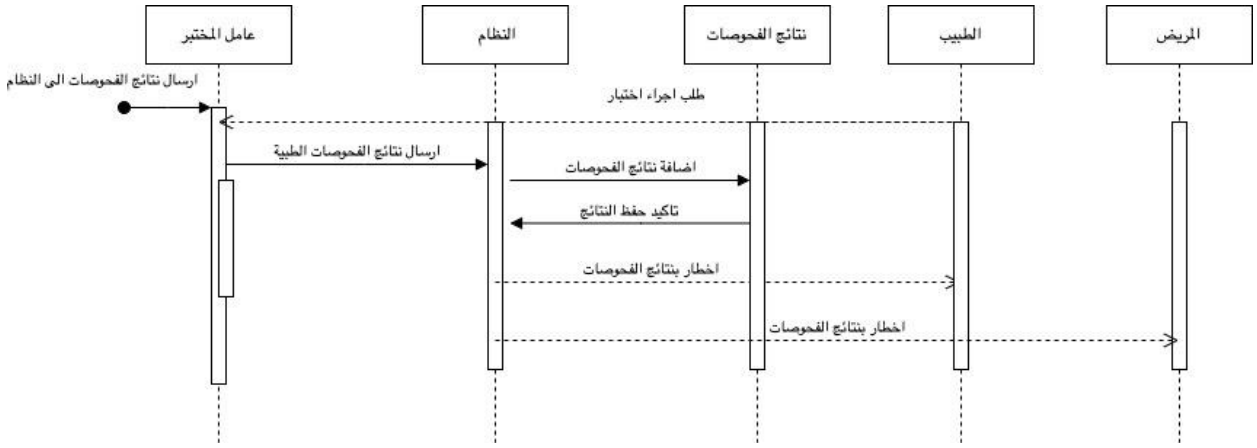
اسم النظام	ادارة حسابات المستخدمين
المستخدم الرئيسي	مسؤول النظام
الهدف	إدارة حسابات المستخدمين في النظام
الشروط	• يجب أن يكون لدى مسؤول النظام صلاحيات خاصة لإدارة حسابات المستخدمين. • يجب أن تكون العمليات سهلة وواضحة للمسؤول.
الاجراءات	• تسجيل دخول مسؤول النظام إلى النظام. • الوصول إلى صفحة إدارة حسابات المستخدمين. • تعديل معلومات المستخدمين. • حذف حسابات المستخدمين.
الاستثناءات	إذا كان هناك خطأ في عملية التعديل أو الحذف، يجب على النظام إظهار رسالة خطأ وعدم تنفيذ العملية.
حل الاستثناءات	تقديم خيار لإلغاء العملية في حالة وجود خطأ.

- **<u>وصف متطلبات النظام الخاصة بفني المختبر</u>**

1. ارسال نتائج الفحوصات الطبية الى النظام.

الجدول (2.13): إرسال نتائج الفحوصات الطبية

اسم النظام	إرسال نتائج الفحوصات الطبية
المستخدم الرئيسي	فني المختبر
الهدف	إرسال نتائج الفحوصات الطبية إلى النظام لتخزينها واستخدامها لاحقًا من قبل الأطباء والمرضى
الشروط	• يجب أن يكون فني المختبر صلاحيات خاصة لإرسال النتائج الطبية إلى النظام. • يجب أن تكون عملية إرسال النتائج سهلة وفعالة.
الاجراءات	• تسجيل دخول فني المختبر إلى النظام. • اختيار الفحص الطبي الذي تم إجراؤه وإدخال نتائجه. • تأكيد إرسال النتيجة وتسجيلها في النظام.
الاستثناءات	إذا فشلت عملية إرسال النتيجة بسبب مشكلة تقنية، يجب على النظام إظهار رسالة خطأ وتوفير وسيلة لإعادة المحاولة.
حل الاستثناءات	-تقديم خيار لإعادة المحاولة في حالة فشل العملية بسبب مشكلة تقنية.



الشكل (2.3): (إرسال نتائج الفحوصات الطبية) Sequence diagram for

الفصل الثالث

تصميم قاعدة البيانات

المحتويات:

3.1 مقدمة

3.2 EERD Diagram

3.3 Mapping

3.4 وصف جداول قاعدة البيانات

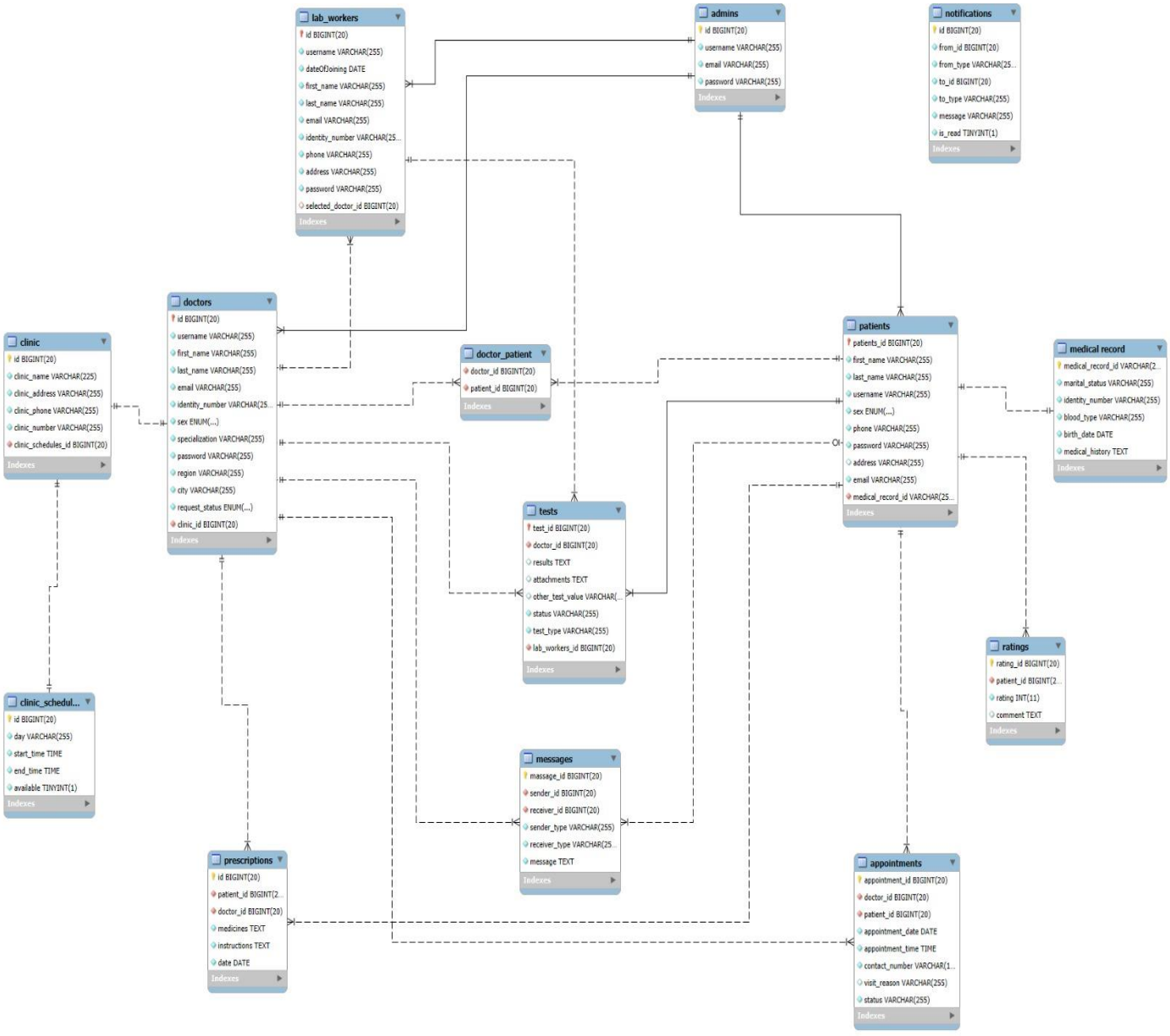
3.5 ملخص الفصل

3.1 مقدمة

يهدف هذا الفصل إلى تصميم قاعدة بيانات متكاملة تدعم وظائف النظام بكفاءة. يبدأ بعرض مخطط العلاقات الكيانية (ERD) لتوضيح الكيانات والعلاقات، يليه نمذجة قاعدة البيانات (Database Mapping) التي تُترجم الكيانات إلى جداول فعلية مع تحديد المفاتيح الأساسية والأجنبية. أخيرًا، يتم وصف جداول قاعدة البيانات، موضحًا تفاصيل الحقول وأنواع البيانات لضمان إدارة سليمة وفعالة للمعلومات.

3.2 Entity Relationships Diagram (ERD)

الشكل (6) يوضح نموذج الكيان-العلاقة (EERD Diagram) للمشروع حيث تظهر العناصر الأساسية بالموقع وعلاقتها ببعض، بالإضافة إلى الحقول الخاصة بكل عنصر.

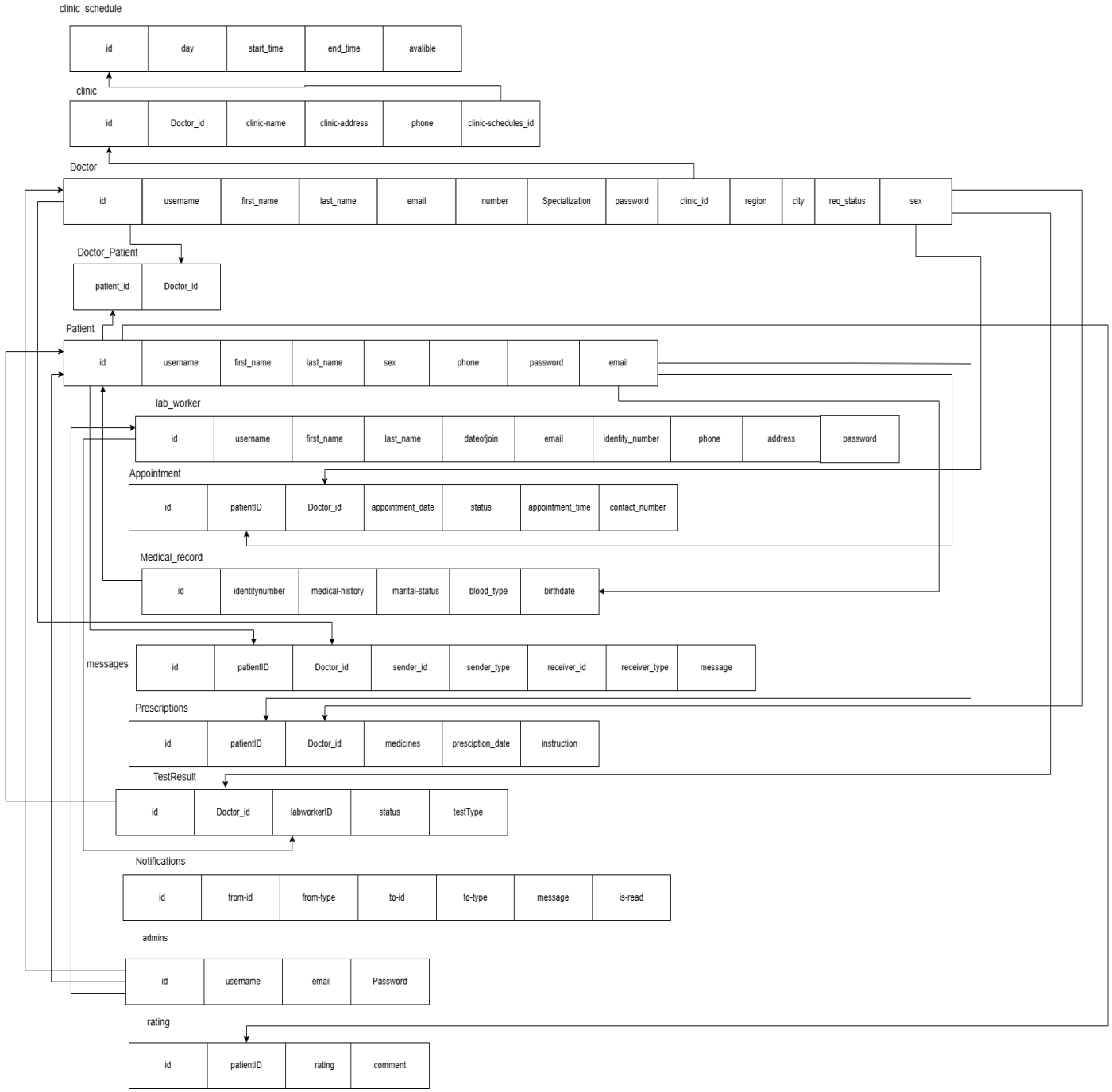


الشكل (3.1): مخطط الكيانات -العلاقات (ERD Diagram)

Database Mapping 3.3

في هذا القسم، تم توضيح عملية الربط (Mapping) لقاعدة البيانات الخاصة بمشروع بوابة العيادة الطبية. يظهر في المخطط أعلاه كيفية ترابط الجداول المختلفة في قاعدة البيانات، مثل جداول المستخدمين، الأطباء، المرضى، المواعيد، السجلات الطبية، والوصفات الطبية. الهدف من هذا الربط هو توضيح العلاقات بين الكيانات المختلفة في النظام، مما يسهل عملية تصميم وتنفيذ قاعدة البيانات بشكل فعّال.

كل هذه الجداول مترابطة بشكل واضح من خلال المفاتيح الخارجية (Foreign Keys) لتوفير تكامل البيانات، مما يضمن تنفيذ العمليات المختلفة مثل حجز المواعيد، إدارة السجلات الطبية، وتسجيل المرضى بطريقة سلسة وفعّالة.



الشكل (3.2): Database Mapping

3.4 وصف جداول قاعدة البيانات

في مشروعنا هذا سوف يقوم فريق العمل باستخدام قاعدة بيانات mysql، لأنه مستقر، سهل الاستخدام، مجاني، مفتوح المصدر، يوفر أداءً عاليًا وأمانًا قويًا، ويتكامل جيدًا مع PHP ومعظم التقنيات الأخرى.

الجدول (3.1): وصف جداول قاعدة البيانات

الجدول	اسم الجدول في قاعدة البيانات	وصف الجدول
مدير النظام	**Admin**	يُعد جدول admins أداة أساسية لإدارة النظام وضمان أمانه وفعاليته.
الطبيب	**Doctor**	يُعد جدول Doctor أداة أساسية لإدارة معلومات الأطباء وتحسين رعاية المرضى.
المريض	**Patient**	يُعد جدول Patient أداة أساسية لإدارة معلومات المرضى.
فني المختبر	**lab_worker**	يُعد جدول lab_worker أداة أساسية لإدارة معلومات فنيي المختبر وتحسين العمليات المختبرية.
العيادة	**Clinic**	يحتوي على تفاصيل العيادات. يمكن أيضًا أن يكون هناك ارتباط مع جداول الجداول الزمنية الخاصة
المواعيد	**Appointments**	يُعد جدول Appointments أداة أساسية لإدارة مواعيد المرضى.
السجلات الطبية	**medical_records**	يخزن معلومات حول السجلات الطبية للمرضى. يلعب هذا الجدول دورًا حاسماً في توثيق تاريخ المريض الطبي.
مواعيد العيادة	**Clinic_schedule**	يخزن المواعيد الزمنية الخاصة بالعيادات (اليوم، وقت البدء، ووقت الانتهاء). يرتبط بجدول العيادات لتحديد الجداول الزمنية لكل عيادة.
الوصفات الطبية	**prescriptions**	يُعد جدول prescriptions أداة أساسية لإدارة الوصفات الطبية وتحسين سلامة المرضى ورعاية صحية أفضل.
الاختبارات	**Tests**	يخزن معلومات حول نتائج الفحوصات الطبية التي يتم إجراؤها على المرضى. يلعب هذا الجدول دورًا حاسماً في تتبع صحة المرضى.
علاقة الطبيب بالمريض	**doctor_patient**	جدول وسيط يحدد العلاقة بين المرضى والأطباء.
الرسائل	**messages**	يحتفظ بسجل الرسائل بين المرضى والأطباء. يتضمن نوع الرسالة والمعلومات ذات الصلة.
الإشعارات	**notifications**	يحتوي على إشعارات مرسلة إلى المستخدمين مثل المرضى أو الأطباء، ويحدد المصدر والوجهة ونوع الإشعار.
التقييمات	**ratings**	يحتوي على تقييمات المرضى للأطباء. يتضمن التقييم الرقمي، التعليقات، والارتباط بالمريض والطبيب.

1. **جدول الطبيب (doctors):** يعد جدول الطبيب عنصرًا أساسيًا في بوابة العيادة الطبية، مما يتيح رعاية فعالة للمرضى، واتصالات مبسطة، وإدارة فعالة.

الجدول (3.2): وصف حقول جدول الطبيب

اسم الحقل	Null	الطول	نوع الحقل	مفتاح فريد	الوصف
Doctor_id (PK)	No	20	BIGINT	Yes	معرف فريد لكل طبيب.
Username	NO	255	VARCHAR	No	اسم المستخدم لتسجيل الدخول.
First_name	No	255	VARCHAR	No	الاسم الأول لطبيب
Last_name	No	255	VARCHAR	No	الاسم الاخير لطبيب
Email	No	255	VARCHAR	No	عنوان البريد الإلكتروني لطبيب.
Identity_number	No	255	VARCHAR	No	رقم الهوية الخاص بالطبيب
Specialization	No	255	VARCHAR	No	التخصص الطبي للطبيب (مثل أمراض القلب والأمراض الجلدية).
Password	No	255	VARCHAR	No	كلمة المرور لتسجيل الدخول (مشفرة).
region	Yes	255	VARCHAR	No	المنطقة السكنية للطبيب.
city	Yes	255	VARCHAR	No	المدينة التي يقيم بها الطبيب.
Clinic_id {FK}	No	20	BIGINT	No	مفتاح أجنبي يربط جدول الطبيب مع جدول العيادة.

2. **جدول مدير النظام (admins):** يخدم جدول admins دورًا حاسمًا في الحفاظ على أمان النظام، مما يضمن أن الأفراد المصرح لهم فقط هم من يمكنهم إدارة مكونات النظام المهمة وتنفيذ المهام الإدارية.

الجدول (3.3): وصف حقول جدول المستخدم

اسم الحقل	Null	الطول	نوع الحقل	مفتاح فريد	الوصف
Admin_id	No	20	BIGINT	Yes	اسم المستخدم لتسجيل الدخول.
Password	No	255	VARCHAR	No	كلمة المرور لتسجيل الدخول (مشفرة).
Email	No	255	VARCHAR	No	البريد الالكتروني الخاص بمسؤول النظام
Username	No	25	VARCHAR	No	اسم المستخدم لتسجيل الدخول.

3. **جدول المريض (patients):** يعد جدول المريض بمثابة الأساس لإدارة فعالة للمرضى وتنظيم المعلومات داخل بوابة العيادة الطبية.

الجدول (3.4) : وصف حقول جدول المريض

اسم الحقل	Null	الطول	نوع الحقل	مفتاح فريد	الوصف
Patient_id {PK}	No	-	INT	Yes	معرف فريد لكل مريض.
FirstName	No	255	VARCHAR	No	الاسم الأول للمريض.
LastName	No	255	VARCHAR	No	الاسم الأخير للمريض.
Username	No	255	VARCHAR	No	الاسم المستخدم الخاص بالمريض.
sex	No	-	ENUM	No	جنس المريض (ذكر/ انثى)
Address	Yes	255	VARCHAR	No	عنوان سكن المريض.
Phone	No	20	VARCHAR	No	رقم الهاتف الخاص بالمريض.
Password	No	255	VACHAR	No	كلمة المرور الخاصة بالمريض.
Email	No	255	VARCHAR	No	عنوان البريد الإلكتروني للمريض
medical_record_id {FK}	Yes	20	BIGINT	No	مفتاح أجنبي يربط جدول الطبيب بجدول السجلات الطبية.
Test_id {FK}	Yes	20	BIGINT	No	مفتاح أجنبي يربط جدول الطبيب بجدول الفحوصات.
Admins_id {FK}	No	20	BIGINT	No	مفتاح أجنبي يربط جدول الطبيب مع جدول المسؤول.

4. **جدول السجلات الطبية (medical_record):** ومن خلال الاستخدام الفعال لجدول "السجلات الطبية" وإدارته، يمكن لأنظمة العيادة الطبية تعزيز رعاية المرضى وتحسين الاتصال وتبسيط العمليات الإدارية.

الجدول (3.5): وصف حقول جدول السجلات الطبية

اسم الحقل	Null	الطول	نوع الحقل	مفتاح فريد	الوصف
Medical_records_id (PK)	No	20	BIGINT	Yes	معرف فريد للسجل الطبي لكل مريض.
marital_status	No	255	VARCHAR	No	الحالة الاجتماعية للمريض.
identity_number	No	255	VARCHAR	No	رقم الهوية الوطنية للمريض.
Blood_type	No	255	VARCHAR	No	فصيلة الدم للمريض
Birth_date	No	-	DATE	No	تاريخ ميلاد المريض
Medical_history	No	-	TEXT	No	التاريخ الطبي للحالة

5. **جدول فني المختبر (lab_workers):** يعد جدول فني المختبر بمثابة أداة قيمة لإدارة موظفي المختبر، وضمان سير العمل بكفاءة، والحفاظ على القوى العاملة المؤهلة داخل بوابة العيادة الطبية.

الجدول (3.6): وصف حقول جدول فني المختبر

اسم الحقل	Null	الطول	نوع الحقل	مفتاح فريد	الوصف
Lab_Worker_id {PK}	No	255	INT	YES	المعرف الفريد لفني المختبر.
Username	No	255	INT	NO	اسم المستخدم لفني المختبر.
DateOfJoining	No	-	DATE	NO	تاريخ انضمام فني المختبر إلى النظام.
FirstName	No	255	VARCHAR	NO	الاسم الأول لفني المختبر.
LastName	No	255	VARCHAR	NO	الاسم الأخير لفني المختبر.
Email	No	255	VARCHAR	No	عنوان البريد الإلكتروني لفني المختبر
Identity_number	No	255	VARCHAR	No	رقم الهوية الخاص بفني المختبر.
Address	Yes	255	VARCHAR	NO	عنوان السكن لفني المختبر.
Phone	No	20	VARCHAR	NO	رقم الهاتف الخاص بفني المختبر.
Password	No	255	VACHAR	No	كلمة المرور الخاصة بفني المختبر.
Admins_id {FK}	No	20	BIGINT	No	مفتاح أجنبي يربط جدول فني المختبر مع جدول المسؤول.

6. **جدول المواعيد (appointments):** يعد جدول "المواعيد" بمثابة عنصر أساسي لإدارة رعاية المرضى، وضمان جدولة منظمة، واتصال فعال، وتفاعلات المريض الموثقة جيدًا داخل بوابة العيادة الطبية.

الجدول (3.7): وصف حقول جدول المواعيد

اسم الحقل	Null	الطول	نوع الحقل	مفتاح فريد	الوصف
Appointment_id (PK)	No	20	BIGINT	Yes	معرف فريد لكل موعد.
Patient_id	No	20	BIGINT	No	يشير المفتاح الخارجي إلى معرف المريض في جدول المريض.
Doctor_id	No	20	BIGINT	No	يشير المفتاح الخارجي إلى Doctor_id في جدول Doctor.
Appointment_date	No	-	DATE	No	تاريخ الموعد المقرر.
Appointment_time	No	-	TIME	No	وقت الموعد المقرر
Contact_number	No	255	VATCHAR	No	رقم التواصل للمريض
Visit_reason	Yes	255	VATCHAR	No	سبب الزيارة
Status	NO	20	VARCHAR	No	الحالة الحالية للموعد، مثل "مجدول" أو "مؤكد" أو "معاد جدولته" ....

7. **جدول الوصفات الطبية (Prescriptions):** من خلال الاستخدام الفعال لجدول "الوصفات الطبية" وإدارته، يمكن لبوابة العيادة الطبية تعزيز سلامة الدواء، وتحسين التزام المريض بخطط العلاج، وتحسين عمليات إدارة الدواء.

الجدول (3.8): وصف حقول جدول الوصفات الطبية

اسم الحقل	Null	الطول	نوع الحقل	مفتاح فريد	الوصف
Prescription_id {PK}	NO	20	BIGINT	Yes	معرف فريد لكل سجل وصفة طبية.
Patient_Id	NO	20	BIGINT	No	يشير المفتاح الخارجي إلى معرف المريض في جدول المريض. ربط الوصفة الطبية بالمريض الذي وصفت له.
Doctor_Id	NO	-	BIGINT	No	مفتاح الخارجي يشير إلى DoctorIDفي جدول Doctor.يحدد الطبيب الذي أصدر الوصفة الطبية.
Medicine	NO	-	TEXT	No	قائمة الأدوية في الوصفة.
Date	No	-	DATE	No	تاريخ إصدار الوصفة الطبية.
instructions	NO	-	TEXT	No	تعليمات الاستخدام الخاصة بالوصفة.

8. **جدول الاختبارات (tests):** من خلال الاستخدام الفعال لجدول Tests وإدارته، يمكن لبوابة العيادة الطبية تعزيز رعاية المرضى وتحسين دقة التشخيص وتحسين نتائج العلاج.

الجدول (3.9): وصف حقول جدول الاختبار.

اسم الحقل	Null	الطول	نوع الحقل	مفتاح فريد	الوصف
Test_id {PK}	No	20	BIGINT	Yes	المعرف الفريد لكل اختبار.
doctor_id	No	20	BIGINT	No	يشير المفتاح الخارجي إلى معرف الطبيب في جدول الطبيب. ربط نتيجة الاختبار بالطبيب الذي طلب الاختبار.
Result	Yes	-	TEXT	No	النتائج التفصيلية للفحص الطبي. يمكن أن يشمل ذلك: * القيم العددية (على سبيل المثال، مستوى السكر في الدم، ومستوى الكوليسترول) * النتائج الوصفية (على سبيل المثال، وجود أو عدم وجود تشوهات).
attachments	Yes	-	TEXT	No	الملفات المرفقة المتعلقة بالفحص.
Other_test_value	Yes	-	TEXT	No	قيم أخرى للفحص (إن وجدت)
status	Yes	255	VARCHAR	No	حالة الفحص (تم/قيد التنفيذ).
TestType	No	255	VARCHAR	No	نوع الاختبار الطبي الذي يتم إجراؤه (على سبيل المثال، فحص الدم، الأشعة السينية، التصوير بالرنين المغناطيسي).
lab_workers_id	No	20	BIGINT	No	معرف فني المختبر الذي أجرى الفحص.

9. **جدول التقييمات (ratings):** يحتوي على تقييمات المرضى للأطباء. يتضمن التقييم الرقمي، التعليقات، والارتباط بالمريض والطبيب.

الجدول (3.10): وصف حقول جدول التقييمات

اسم الحقل	Null	الطول	نوع الحقل	مفتاح فريد	الوصف
Rating_id {PK}	No	20	BIGINT	Yes	معرف فريد لكل سجل وصفة طبية.
Patient_Id	No	20	BIGINT	No	يشير المفتاح الخارجي إلى معرف المريض في جدول المريض. ربط التقييم بالمريض الذي قام بالتقييم.
Rating	No	-	INT	No	تقييم المريض للطبيب أو الخدمة.
Comment	Yes	-	TEXT	No	التعليقات النصية المتعلقة بالتقييم.

10. **جدول العيادة (clinic):** يحتوي على تفاصيل العيادات. يمكن أيضًا أن يكون هناك ارتباط مع جداول الجداول الزمنية الخاصة.

الجدول (3.11): وصف حقول جدول العيادة

اسم الحقل	Null	الطول	نوع الحقل	مفتاح فريد	الوصف
clinic_id {PK}	No	20	BIGINT	Yes	معرف فريد لكل عيادة.
clinic_name	No	255	VARCHAR	No	اسم العيادة
clinic_address	No	255	VARCHAR	No	عنوان العيادة
Clinic_phone	No	255	VARCHAR	No	رقم هاتف العيادة
Clinic_number	Yes	255	VARCHAR	No	رقم العيادة داخل مؤسسة (إن وجد)
clinic_schedules_id	No	255	VARCHAR	No	معرف الجداول الزمنية للعيادة (مرتبط بـ clinic_schedules).

11. **جدول مواعيد العيادة (clinic schedules):** يخزن المواعيد الزمنية الخاصة بالعيادات (اليوم، وقت البدء، ووقت الانتهاء). يرتبط بجدول العيادات لتحديد الجداول الزمنية لكل عيادة.

الجدول (3.12): وصف حقول جدول مواعيد العيادة

اسم الحقل	Null	الطول	نوع الحقل	مفتاح فريد	الوصف
Clinic_schedules_id {PK}	No	20	BIGINT	Yes	معرف فريد لكل عيادة.
Day	No	255	VARCHAR	No	أيام العمل في الاسبوع
Start_time	No	-	TIME	No	وقت بداية الدوام
End_time	No	-	TIME	No	وقت انتهاء الدوام
Available	Yes	1	TINYINT	No	حالة توفر الدوام (1 = متاح, 0 = غير متاح)

12.جدول الاشعارات (notifications): يحتوي على إشعارات مرسلة إلى المستخدمين مثل المرضى أو الأطباء، ويحدد المصدر والوجهة ونوع الإشعار.

الجدول (3.13): وصف حقول جدول الاشعارات

اسم الحقل	Null	الطول	نوع الحقل	مفتاح فريد	الوصف
not_id {PK}	No	20	BIGINT	Yes	معرف فريد لكل إشعار.
From_id	No	20	BIGINT	No	معرف المستخدم الذي أرسل الإشعار.
From_type	No	255	VARCHAR	No	نوع المستخدم الذي أرسل الاشعار
To_id	No	20	BIGINT	No	معرف المستخدم المستلم للإشعار.
To_type	No	255	VARCHAR	No	نوع المستلم (مثل طبيب، مريض).
message	No	255	VARCHAR	No	نص الرسالة للإشعار
Read	No	1	TINYINT	No	حالة الإشعار (1 = مقروء، 0 = غير مقروء).

13. **جدول الرسائل (messages):** يحتفظ بسجل الرسائل بين المرضى والأطباء. يتضمن نوع الرسالة والمعلومات ذات الصلة.

الجدول (3.14): وصف حقول جدول الرسائل

اسم الحقل	Null	الطول	نوع الحقل	مفتاح فريد	الوصف
message_id {PK}	No	20	BIGINT	Yes	معرف فريد لكل إشعار.
sender_id	No	20	BIGINT	No	معرف المستخدم الذي أرسل الرسالة. مفتاح خارجي يشير الى (doctor_id & patient_id)
Receiver_id	No	20	BIGINT	No	معرف المستخدم المستلم للرسالة. مفتاح خارجي يشير الى (doctor_id & patient_id)
Sender_type	No	255	VARCHAR	No	نوع المستخدم الذي أرسل الرسالة (طبيب، مريض)..
Receiver_type	No	255	VARCHAR	No	نوع المستخدم المستلم للرسالة (طبيب، مريض).
message	No	-	TEXT	No	محتوى الرسالة.

الفصل الرابع

تصميم النظام

المحتويات:

4.1 المقدمة

4.2 المصادر البرمجية لبناء النظام

4.3 نموذج معمارية النظام

Model 4.3.1

View 4.3.2

Controller 4.3.3

4.4 واجهات النظام

4.5 الاكواد البرمجية

4.6 ملخص الفصل

4.1 المقدمة

في هذا الفصل، سيتم توضيح مراحل بناء النظام الالكتروني لبوابة العيادة الطبية، بالإضافة إلى الأدوات والبرامج التي تم استخدامها في عملية تطويره.
سنبدأ بتفصيل المراحل المختلفة لتطوير النظام، بدء ا باختيار البرمجيات اللازمة لمراحل تطوير النظام، وسنناقش الأدوات والبرامج التي تم استخدامها في بناء النظام. تشمل هذه الأدوات تقنيات البرمجة، قواعد البيانات، أدوات التصميم. وبعدها سيتم عرض واجهات المستخدم والأكواد الخاصة ببعض الصفحات.

4.2 المصادر البرمجية لبناء النظام

1. **Visual studio code**: هو محرر شفرة مصدر خفيف الوزن ومجان تم تطويره بواسطة مايكروسوفت .يتميز بتصميمه القوي والمرن، مما يجعله مناسب لمجموعة واسعة من لغات البرمجية وأطر العمل. بفضل واجهته القابلة للتخصيص ودعمه الكبير للملحقات، يوفر VS Code بيئة تطوير متكاملة تساعد المطورين على كتابة وتصحيح الشفرة بكفاءة عالية.

الشكل (4.1): ايقونة برنامج VSCode

2. **Microsoft Word**: هو برنامج لمعالجة النصوص تم استخدامه في توثيق سير عمل المشروع، كما ساعد في كتابة وتنسيق التقارير الفنية، ومتابعة تقدم العمل بشكل منظم

3. **phpMyAdmin**: أداة لإدارة قواعد البيانات MySQL عبر واجهة ويب، تُستخدم لإدارة البيانات وتنفيذ استعلامات SQL بسهولة.

الشكل (4.2): ايقونة برنامج phpMyAdmin

4. **Laravel:** هو إطار عمل (Framework) مفتوح المصدر لتطوير تطبيقات الويب باستخدام لغة PHP يتميز ببنية مرتبة تعتمد على معمارية MVC، مما يجعل تطوير التطبيقات أسرع وأسهل. يوفر Laravel أدوات قوية مثل التوجيه (Routing) إدارة قواعد البيانات (Eloquent ORM)، الحماية (Security) وقوالب العرض (Blade)، مما يساعد المطورين على بناء تطبيقات مرنة وقابلة للتطوير.

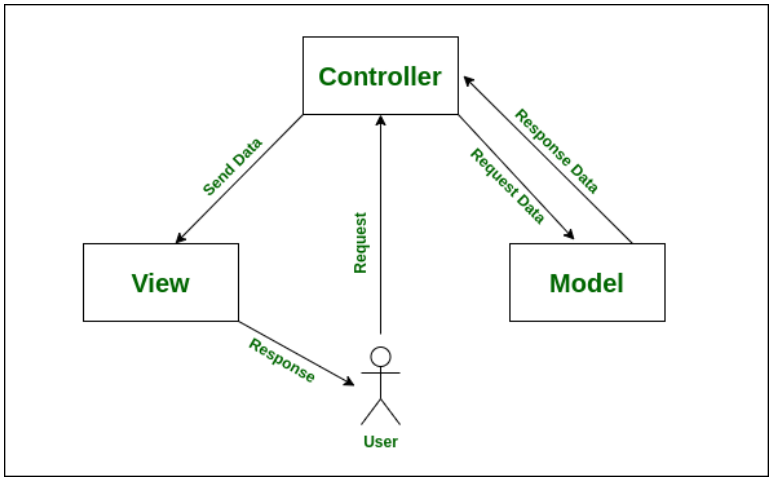
الشكل (4.3): ايقونة إطار العمل Laravel

5. **Canva:** هو أداة تصميم جرافيكي عبر الانترنت التي تم استخدامها لإنشاء بعض الشعارات (logo) وكذلك عمل فيديو العرض التقديمي.

6. **Bootstrap:** هو إطار عمل مفتوح المصدر لتصميم واجهات المستخدم باستخدام HTML, CSS وJavaScript. تم استخدام Bootstrap في المشروع لتطوير وتصميم صفحات الويب

4.3 نموذج معمارية النظام

سيتم اعتماد نموذج Model-View-Controller (MVC) لبناء النظام، الذي يفصل النظام إلى ثلاثة أجزاء أساسية: Model, View, Controller، حيث يقوم كل جزء منها بتنفيذ مهمة معينة في النظام. هذا النموذج يُستخدم على نطاق واسع في تطوير تطبيقات الويب الحديثة لأنه يساعد في تصميم تطبيقات قابلة للتطوير والتوسع بسهولة، ويخفف من تعقيد النظام من خلال فصل المهام.



الشكل (4.4): نموذج معمارية MVC

Model 4.3.1

يمثل هذا الجزء مسؤولية إدارة البيانات التي يتعامل معها النظام مثل: معلومات المرضى، الأطباء، المواعيد، والوصفات الطبية، وأي عمليات مرتبطة بهذه البيانات. يقوم الـ Model بتنفيذ العمليات المرتبطة بقاعدة البيانات مثل جلب البيانات، حذف البيانات، إضافة البيانات، وتحديثها ليتم استخدامها في النظام. إذا حدثت تغييرات في البيانات الموجودة في الـ Model، فإنه يقوم بإعلام الـ View لعرض التغييرات اللازمة، وأحياناً يُخبر الـ Controller حسب الحاجة.

View 4.3.2

يمثل View جميع عناصر واجهة المستخدم التي يتفاعل معها مستخدم النظام مثل: نماذج الحجز، الجداول الزمنية، المعلومات الطبية، وشاشات العرض المختلفة. هذا الجزء يتحكم في كيفية عرض البيانات للمستخدم ويضمن أن تكون واجهة المستخدم سهلة الاستخدام وواضحة. الـ View لا يتفاعل مع الـ Model بشكل مباشر، بل يتلقى البيانات والتعليمات من Controller لعرضها بشكل صحيح.

Controller 4.3.3

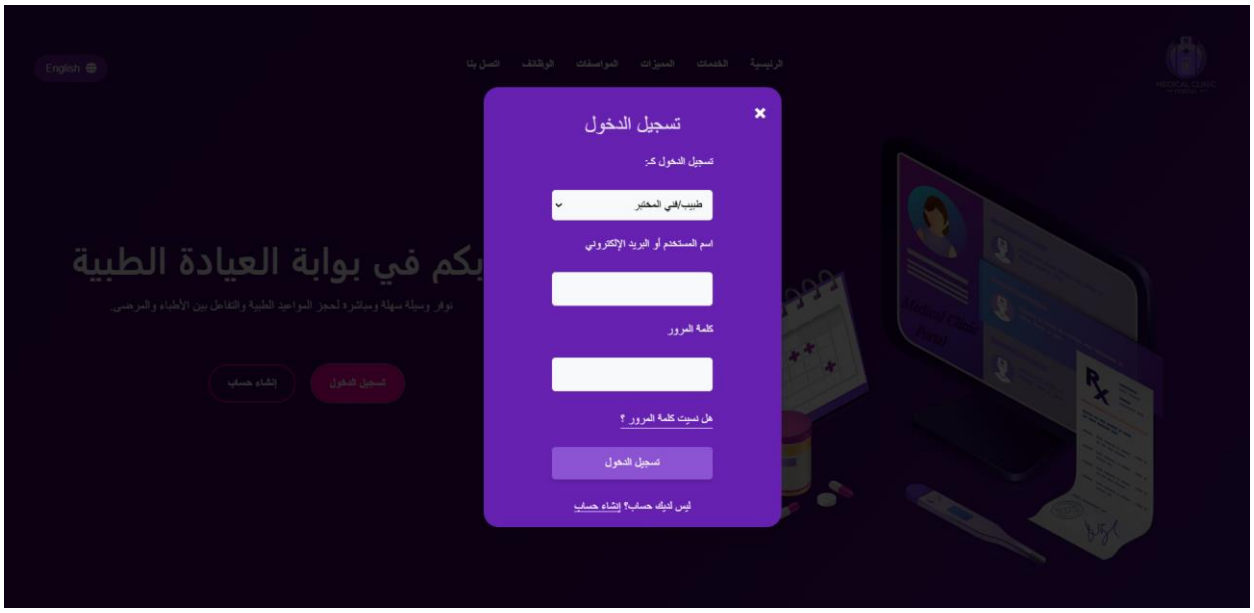
يعمل Controller كحلقة الوصل بين الـ Model والـ View. يقوم بمعالجة الطلبات الصادرة من المستخدم مثل حجز المواعيد أو تسجيل المرضى أو كتابة وصفات طبية جديدة. يتلقى الـ Controller الطلبات من الـ View، ويقوم بتنفيذ العمليات المطلوبة على البيانات من خلال الـ Model، ثم يحدد البيانات التي سيتم عرضها للمستخدم من خلال الـ View.

4.4 واجهات النظام (Interfaces)

- **الصفحات المشتركة**

1. واجهة تسجيل الدخول:

يجب على مستخدمي الموقع تسجيل الدخول للحصول على الخدمات المختلفة. سيدخلون بريدهم الإلكتروني او اسم المستخدم وكلمة المرور. إذا كانت صحيحة، فستوجههم هذه الشاشة إلى صفحاتهم. صفحة تسجيل الدخول مخصصة لجميع أنواع المستخدمين عدا مسؤولي النظام لوجود صفحة تسجيل دخول خاصة بهم.

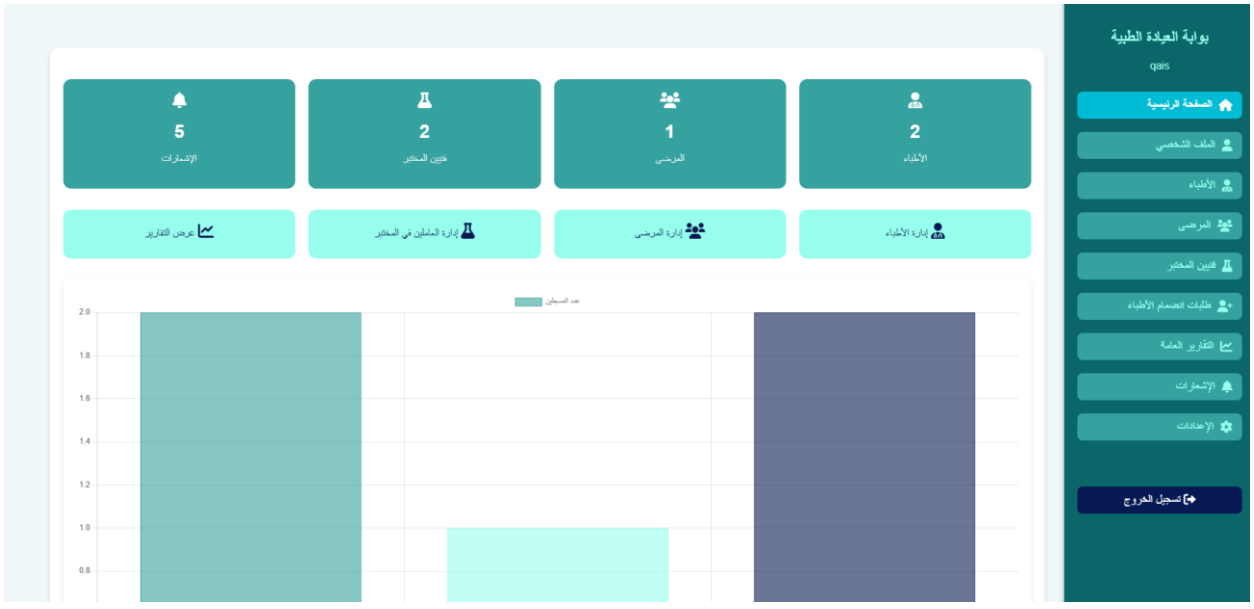


(4.5): شكل واجهة تسجيل الدخول

- **صفحات المسؤول(Admin)**

1. واجهة لوحة التحكم لمسؤول النظام:

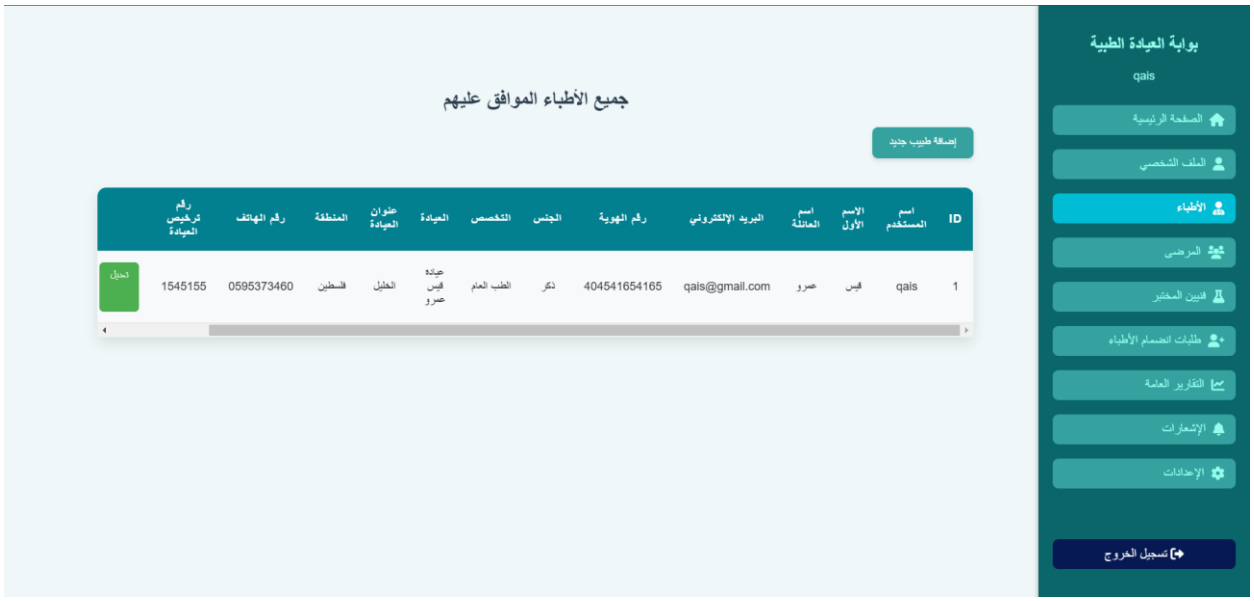
من هذه الصفحة، سيحصل المشرف على ملخص لحالة العيادات الخارجية. تُظهِر هذه الشاشة عدد الأطباء والمرضى والمواعيد في جميع العيادات. وتُظهِر شريطاً جانبياً يوفر تنقلًا سريعًا بين الأقسام المختلفة للنظام.



(4.6): شكل واجهة لوحة تحكم مسؤول النظام.

2. واجهة الطبيب لمسؤول النظام:

هذه واجهة تعرض قائمة بجميع الأطباء المسجلين في النظام، مع تفاصيلهم مثل الاسم، الجنس، التخصص، البريد الإلكتروني، رقم الهوية، العنوان، والمنطقة. تحتوي على زر لإضافة طبيب جديد، بالإضافة إلى شريط جانبي للتنقل بين الأقسام المختلفة للنظام.

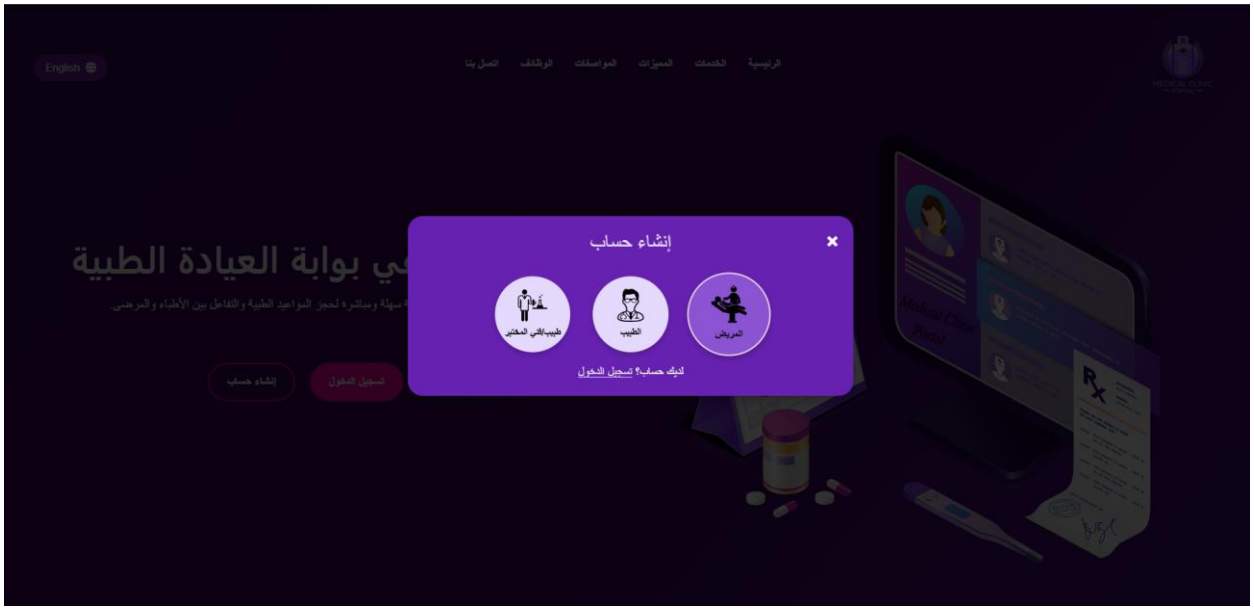


(4.7): شكل واجهة الأطباء لمسؤول النظام.

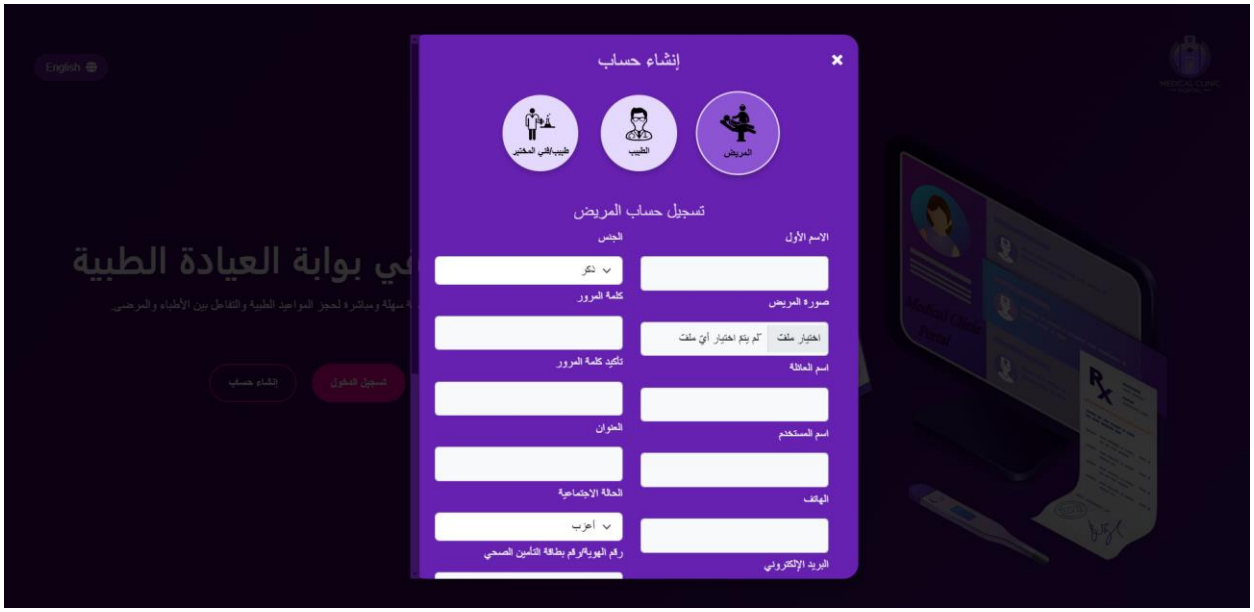
- **صفحات المريض**

1. واجهة التسجيل في النظام

عند الدخول للنظام للمرة الاولى، يقوم المستخدم بتحديد شخصيته ثم يقوم بإدخال البيانات المطلوبة للوصول الى النظام.



(4.8): شكل واجهة التسجيل في النظام (اختيار الشخصية)



(4.9): شكل واجهة تسجيل المريض

2. الواجهة الرئيسية:

تعرض هذه الشاشة ايقونات رئيسية تؤدي إلى شاشات أخرى. الايقونة الأولى تتيح له إمكانية عرض الأطباء. الايقونة الثانية تعرض له جميع العيادات الخارجية. الايقونة الثالثة تعرض له مواعيده والأخيرة لقائمة وصفاته الطبية.
كما تحتوي هذه الشاشة على قائمة تنقل. وهي قائمة بها عدة علامات تبويب، الأولى تعرض ملفه الشخصي. والثانية لعرض الاشعارات. البطاقة الثالثة تعرض له شاشة الرسائل والدردشات.

(4.10): شكل واجهة الصفحة الرئيسية للمريض.

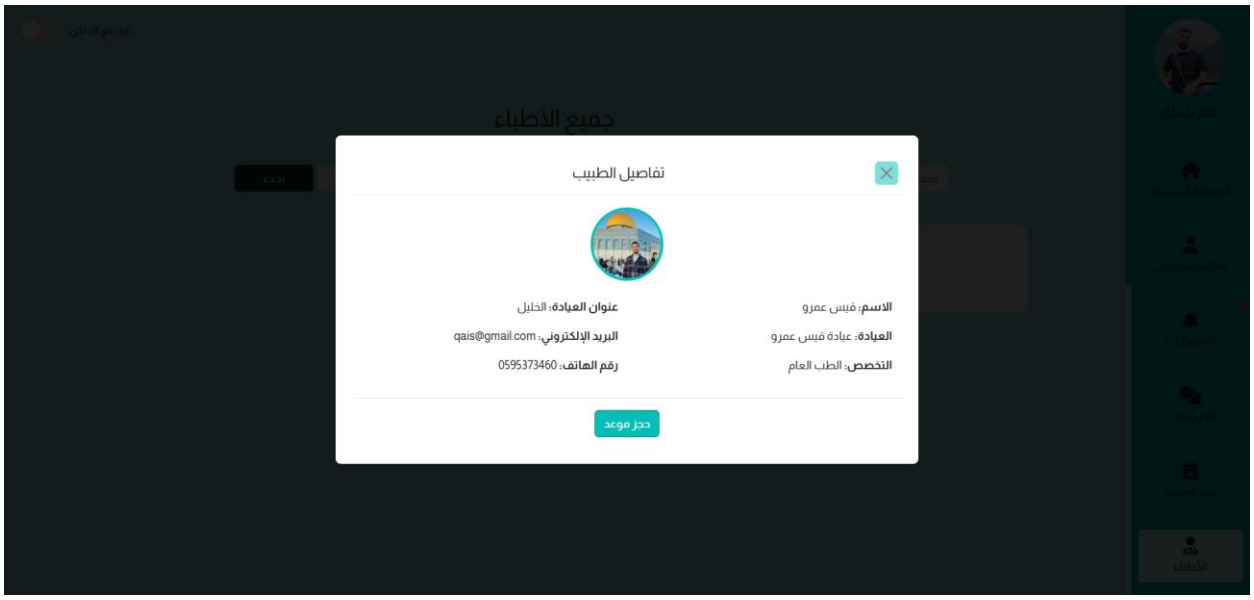
3. صفحة الاطباء:

عند فتح هذه الشاشة تعرض قائمة بجميع الأطباء مع إمكانية البحث عن طبيب معين من خلال شريط البحث. تُظهر كل بطاقة معلومات مختصرة عن الطبيب مثل اسمه وتخصصه. تحتوي الواجهة أيضًا على شريط جانبي للتنقل بين الأقسام المختلفة.

(4.11): شكل واجهة صفحة الطبيب

4. الصفحة الشخصية للطبيب:

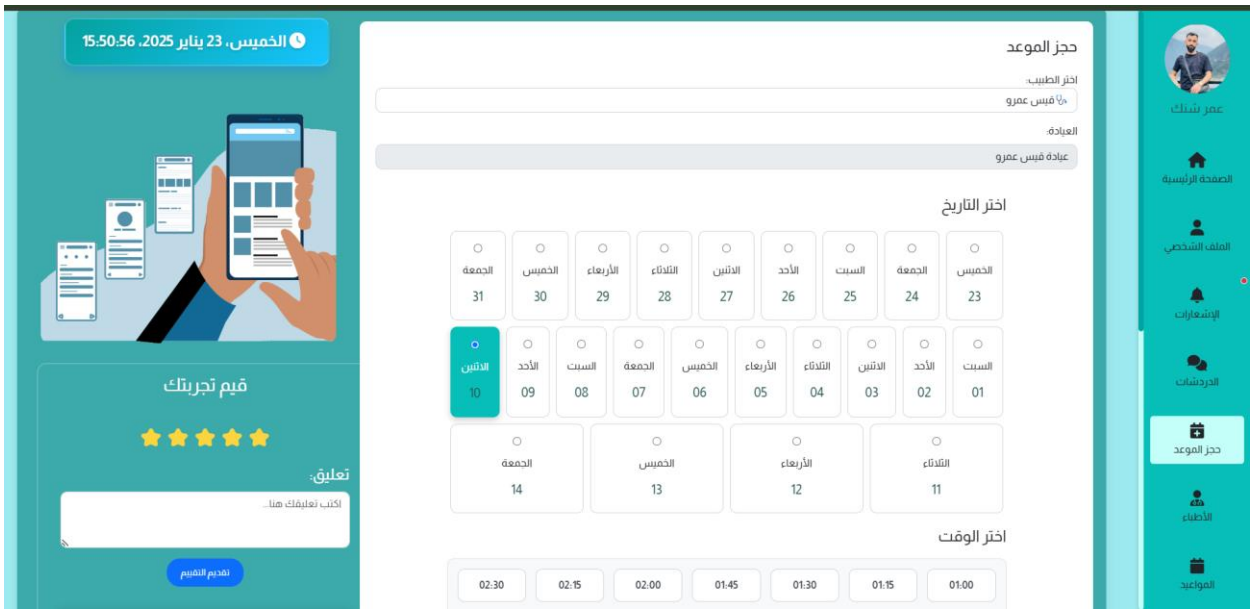
عند اختيار الطبيب من شاشة الأطباء أو بعد الضغط على عيادة معينة ستظهر شاشة الملف الشخصي للطبيب، من هذه الصفحة يمكنه اختيار موعد مع هذا الطبيب.



(4.12): شكل واجهة الملف الشخصي للطبيب

5. صفحة حجز المواعيد

عندما يريد المريض حجز موعد مع طبيب معين يقوم بالضغط على ملفه الشخصي واختيار تبويب حجز موعد ستظهر هذه الشاشة وتظهر جميع الأيام التي يتوفر بها موعد وإذا قام بالضغط على أحد هذه الأيام تظهر المواعيد المتوفرة مع أوقاتها ويمكنه اختيار الموعد وتأكيد الحجز.

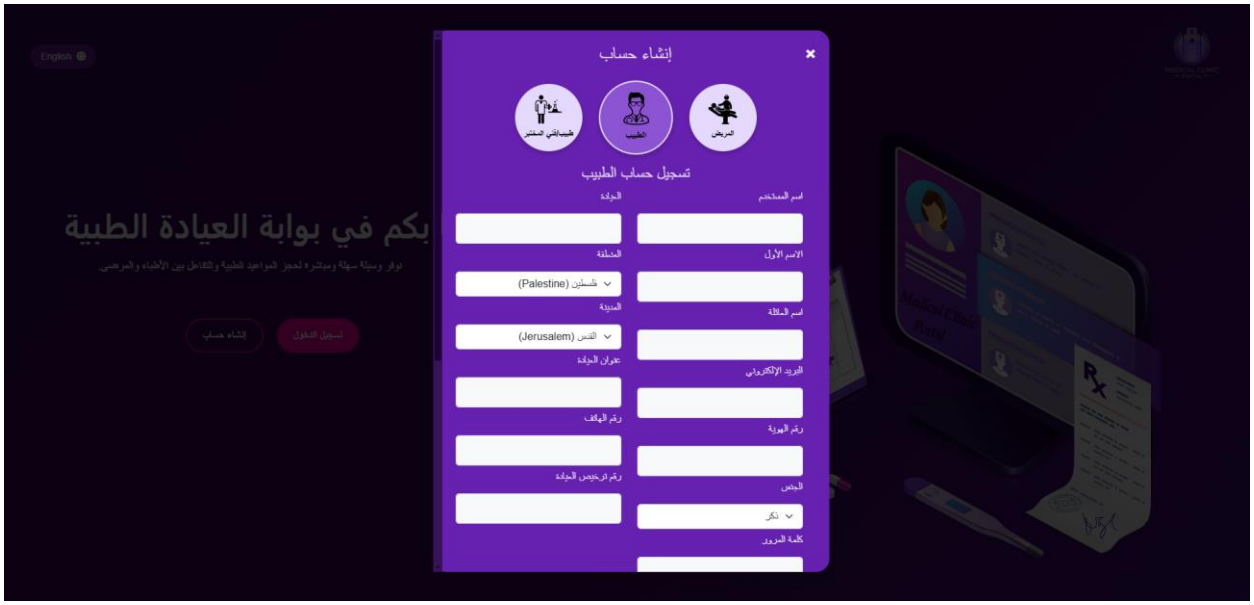


(4.13): شكل واجهة صفحة حجز المواعيد

- **صفحات الطبيب**

1. صفحة التسجيل بالنظام

بعد تحديد الشخصية كطبيب في الشكل (4.3) يجب على الطبيب ادخال جميع البيانات المطلوبة كما في الشكل (4.11)



(4.14): شكل واجهة التسجيل للطبيب

2. الواجهة الرئيسية للطبيب:

واجهة لوحة تحكم الطبيب تعرض ترحيبًا شخصيًا، أزرارًا لإجراءات سريعة (إضافة موعد أو وصفة)، أقسامًا رئيسية (الوصفات، المختبر، المرضى، والمواعيد)، وإحصائيات يومية. وتتضمن شريطًا جانبيًا للتنقل بين الصفحات.

(4.15): شكل الواجهة الرئيسية للطبيب.

4.5 الاكواد البرمجية

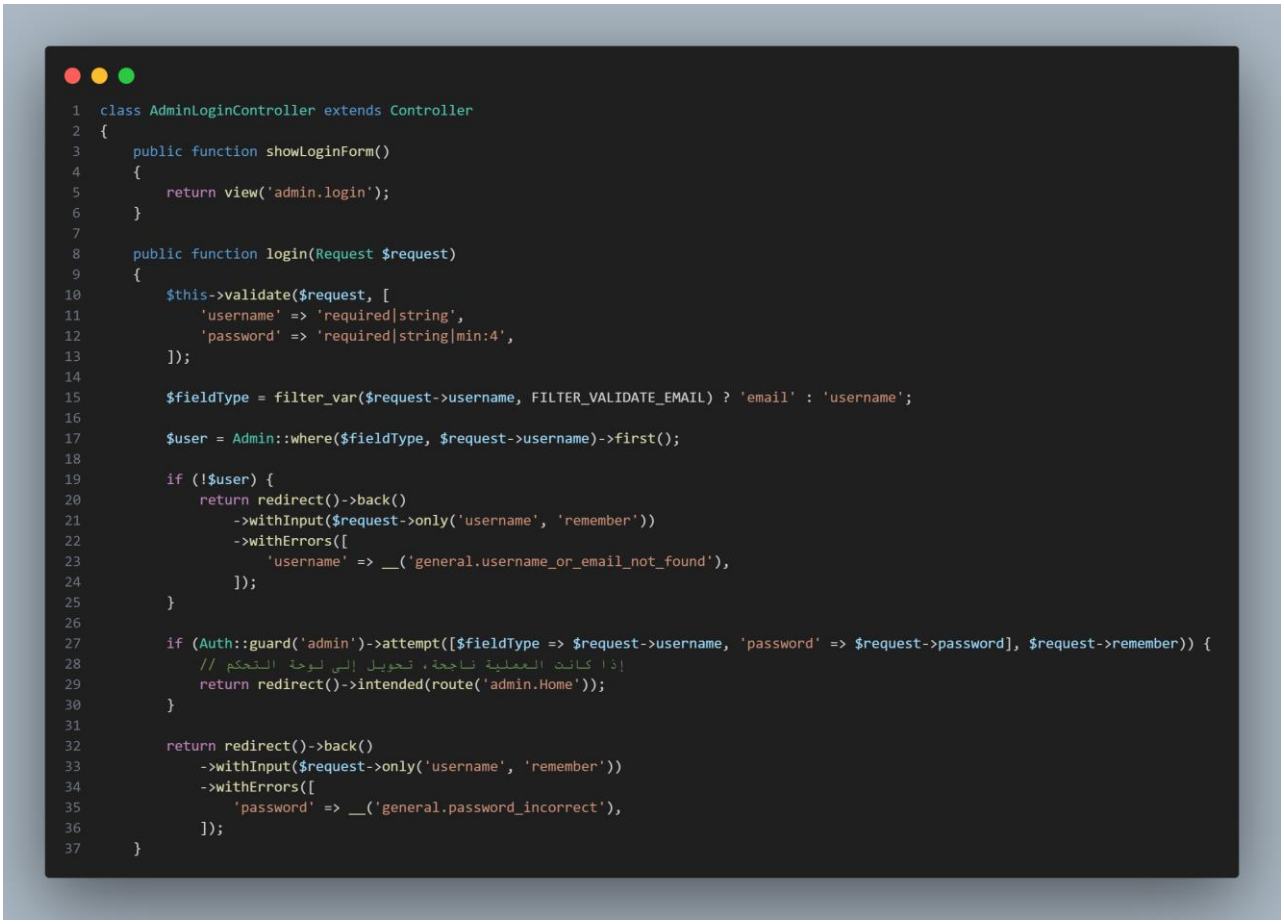
في هذا القسم، سيتم عرض الأكواد البرمجية المتعلقة ببناء الموقع الإلكتروني من داخل المحرر، مع التركيز على تنظيمها وفق وحدة التطبيق. وحدة التطبيق تعني تجميع الملفات البرمجية ذات الصلة بوظيفة معينة داخل النظام، مما يسهل فهمها وإدارتها.

- **واجهة تسجيل الدخول (login.blade.php)**: تحتوي على التصميم والحقول اللازمة لإدخال بيانات تسجيل الدخول. يتم عرضها باستخدام دالة showLoginForm في AdminLoginController.

```blade
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>{{ __('general.admin_login_title') }}</title>
    <link rel="stylesheet" href="{{ asset('css/style.css') }}">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css">
</head>
<body>
    <div class="login-container">
        <div class="login-box">
            <h2>{{ __('general.admin_login_title') }}</h2>

            @if ($errors->any())
                <div class="alert alert-danger">
                    <ul>
                        @foreach ($errors->all() as $error)
                            <li>{{ $error }}</li>
                        @endforeach
                    </ul>
                </div>
            @endif

            <form action="{{ route('admin.login') }}" method="POST">
                @csrf
                <div class="form-group">
                    <i class="fas fa-user"></i>
                    <input type="text" name="username" placeholder="{{ __('general.username_or_email') }}" value="{{ old('username') }}" required autofocus>

                </div>
                <div class="form-group">
                    <i class="fas fa-lock"></i>
                    <input type="password" name="password" id="password" placeholder="{{ __('general.password_placeholder') }}" required>
                    <i class="fas fa-eye password-toggle" style="margin-left: 20px;" onclick="togglePassword()"></i>

                </div>
                <div class="remember-me">
                    <input type="checkbox" name="remember" id="remember">
                    <label for="remember">{{ __('general.remember_me') }}</label>
                </div>
                <button type="submit" class="btn-login">{{ __('general.login_button') }}</button>
            </form>
        </div>
    </div>

```

الشكل (4.16): كود صفحة تسجيل الدخول لمسؤول النظام (login.blade.php)

- **معالجة طلب تسجيل الدخول (AdminLoginController):** يستقبل البيانات المُدخلة ويتحقق من صحتها. يستخدم نموذج المستخدم (User Model) لجلب بيانات الحساب. في حال النجاح، يتم توجيه المستخدم للوحة التحكم، وإلا فسيتم عرض رسالة خطأ.



```php
class AdminLoginController extends Controller
{
    public function showLoginForm()
    {
        return view('admin.login');
    }

    public function login(Request $request)
    {
        $this->validate($request, [
            'username' => 'required|string',
            'password' => 'required|string|min:4',
        ]);

        $fieldType = filter_var($request->username, FILTER_VALIDATE_EMAIL) ? 'email' : 'username';

        $user = Admin::where($fieldType, $request->username)->first();

        if (!$user) {
            return redirect()->back()
                ->withInput($request->only('username', 'remember'))
                ->withErrors([
                    'username' => __('general.username_or_email_not_found'),
                ]);
        }

        if (Auth::guard('admin')->attempt([$fieldType => $request->username, 'password' => $request->password], $request->remember)) {
            // إذا كانت العملية ناجحة، تحويل إلى لوحة التحكم
            return redirect()->intended(route('admin.Home'));
        }

        return redirect()->back()
            ->withInput($request->only('username', 'remember'))
            ->withErrors([
                'password' => __('general.password_incorrect'),
            ]);
    }
```

الشكل (4.17): كود معالجة طلب تسجيل الدخول لمسؤول النظام

- **ربط الواجهة والمعالجة (web.php):** يتم تعريف المسارات التي تربط بين الواجهة والعمليات الخلفية.

```php
// ------------------------
// مسارات المسؤول (Admin Routes)
// ------------------------
Route::prefix('admin')->group(function () {
    // تسجيل الدخول والخروج
    Route::get('/login', [AdminLoginController::class, 'showLoginForm'])->name('admin.login');
    Route::post('/login', [AdminLoginController::class, 'login']);
    Route::post('/logout', [AdminLoginController::class, 'logout'])->name('admin.logout');

    // لوحة التحكم
    Route::middleware('auth:admin')->group(function () {
        // مسار حذف إشعار فردي
    Route::delete('/notifications/{id}', [AdminDashboardController::class, 'deleteNotification'])->name('admin.notifications.delete');
```

الشكل (4.18): كود مسارات تسجيل الدخول والخروج لمسؤول لنظام

بهذا الشكل، يتم تحقيق تنظيم واضح في وحدة التطبيق الخاصة بتسجيل الدخول، مما يُسهل صيانته وتطويره مستقبلًا.

4.6 ملخص الفصل

تم استعراض عملية بناء النظام الالكتروني لبوابة العيادة الطبية والأدوات والبرامج المستخدمة في هذه العملية، بدأنا بشرح البرمجيات اللازمة لمراحل تطوير النظام، ثم انتقلنا إل تفصيل برمجة الجزأين الرئيسي للنظام: واجهات المستخدم وقاعدة البيانات. تم استخدامHTML ، CSS، وJavaScript لواجهات المستخدم، بينما تم استخدام Laravel لإدارة قاعدة البيانات. كما تم عرض بعض الشاشات الخاصة ببناء الموقع الالكتروني من داخل المحرر، مثل صفحة تسجيل الدخول، بالإضافة إلى عرض الكود المستخدم لبعض الصفحات وكيفية تنظيمها في المجلدات المناسبة

الفصل الخامس

فحص النظام

المحتويات:

5.1 مقدمة

5.2 مراحل فحص النظام

5.2.1 فحص وحدات النظام

5.2.2 فحص تكامل النظام

5.3 ملخص الفصل

5.1 مقدمة:

في هذا الفصل، سنقوم بفحص النظام واختبار وظائفه المختلفة للتأكد من أنه يعمل بشكل صحيح وفعال. سنقوم بعرض لقطات شاشة وشرح العمليات الأساسية التي يقوم بها النظام.

5.2 فحص النظام

عند القيام بفحص النظام الإلكتروني، يجب اتباع خطوات معينة ليكون الفحص متكاملاً وشاملاً لجميع أجزاء النظام، تشمل مراحل فحص النظام المراحل التالية:

- **فحص وحدات النظام (Unit Testing):** يتم اختبار كل وحدة من وحدات النظام بشكل منفصل لضمان أن كل جزء يعمل كما هو متوقع. يشمل هذا الفحص التحقق من وظائف الوحدات البرمجية الصغيرة، ومعالجة الأخطاء المحتملة.

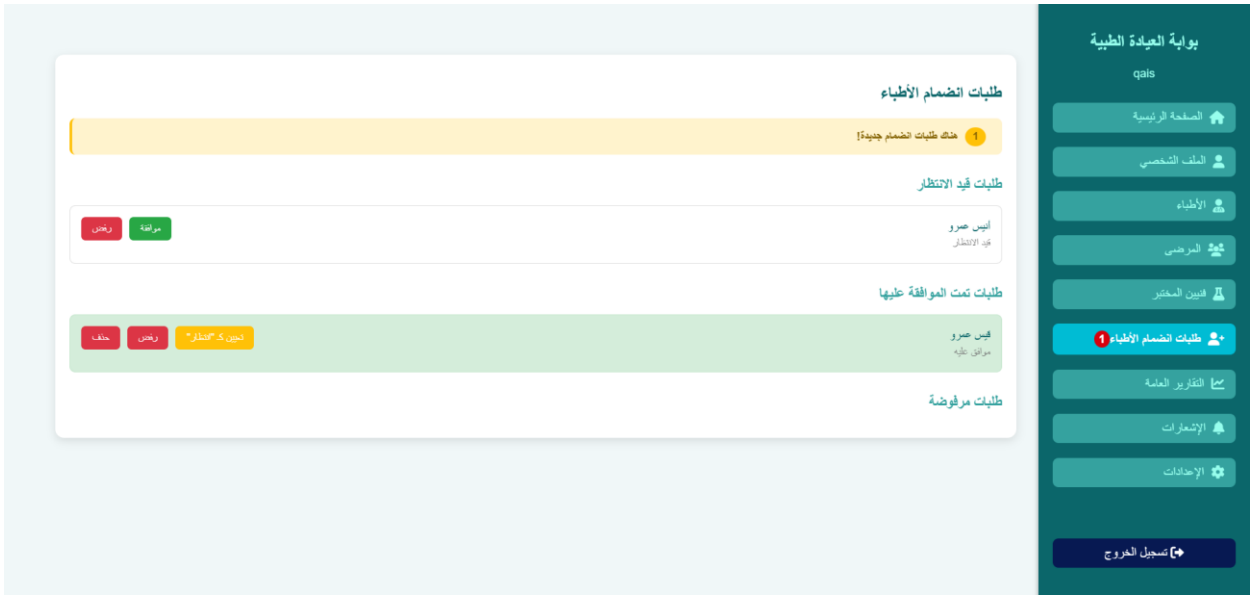
- **فحص تكامل النظام (Integration Testing):** يتم التحقق من تكامل الوحدات المختلفة مع بعضها البعض، لضمان أن التفاعلات بين الوحدات تتم بشكل صحيح ودون أي مشاكل. هذا يشمل التأكد من أن البيانات تنتقل بشكل صحيح وبين الوحدات المختلفة، وأن العمليات المشتركة تعمل بدون تعارض.

5.2.1 فحص وحدات النظام

- **فحص عملية قبول طلبات انضمام الأطباء لنظام:**

يُعد قبول طلبات الانضمام أحد العمليات الرئيسية التي يقوم بها مسؤول النظام. تتيح هذه العملية إضافة أطباء وعيادات جديدة الى النظام.

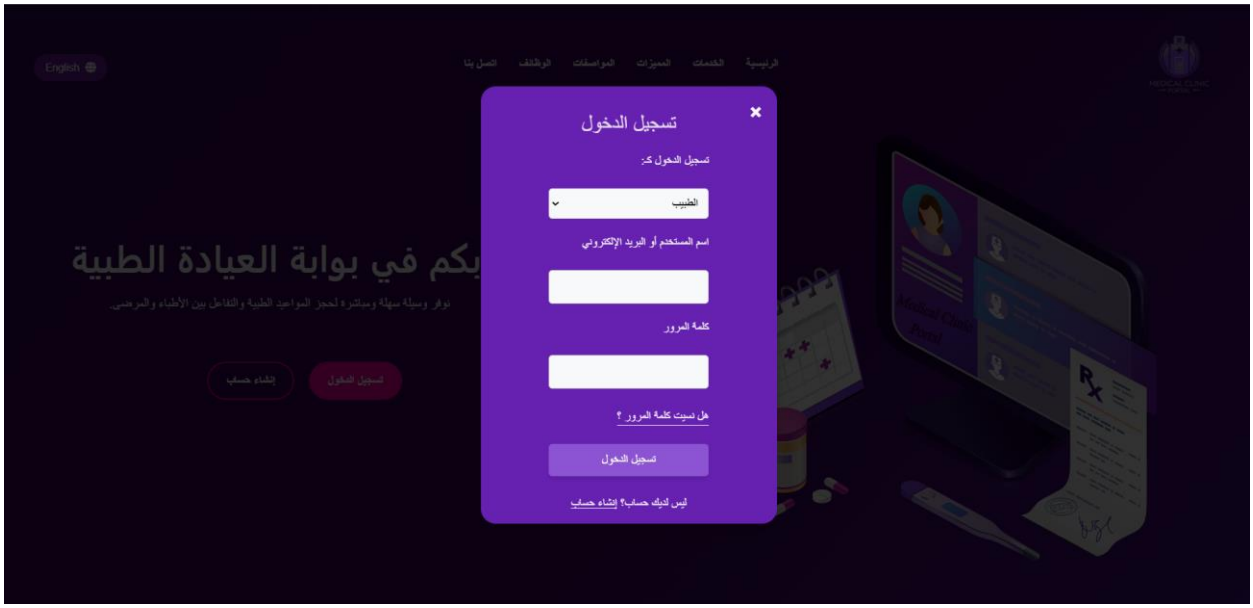
(5.1) شكل واجهة حالة الطلب عند الطبيب بعد التسجيل في النظام



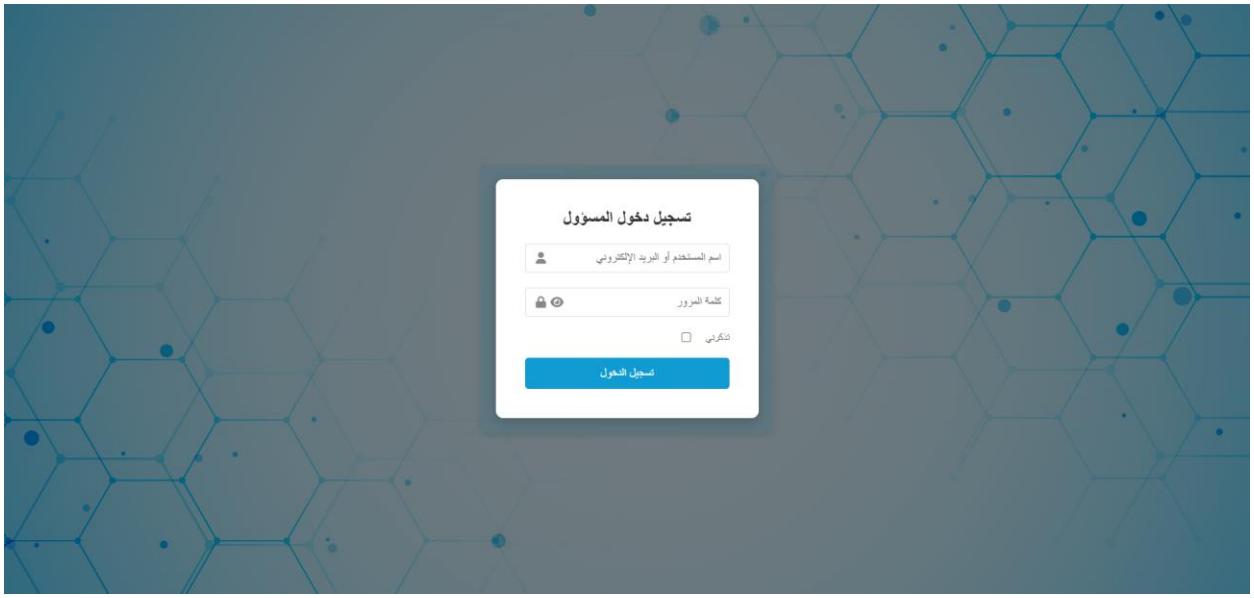
(5.2) شكل واجهة طلبات الانضمام عند مسؤول النظام

- **فحص عملية تسجيل الدخول إلى النظام:**

يوضح الشكل (5.3) واجهة تسجيل الدخول: يقوم المستخدم بإدخال اسم المستخدم وكلمة المرور للوصول إلى لوحة التحكم حسب نوع المستخدم (الشخصية). كما وتظهر هذه الواجهة لجميع مستخدمي النظام عدا مسؤولي النظام، لوجود صفحة تسجيل خاصة بهم لدخول الى النظام.



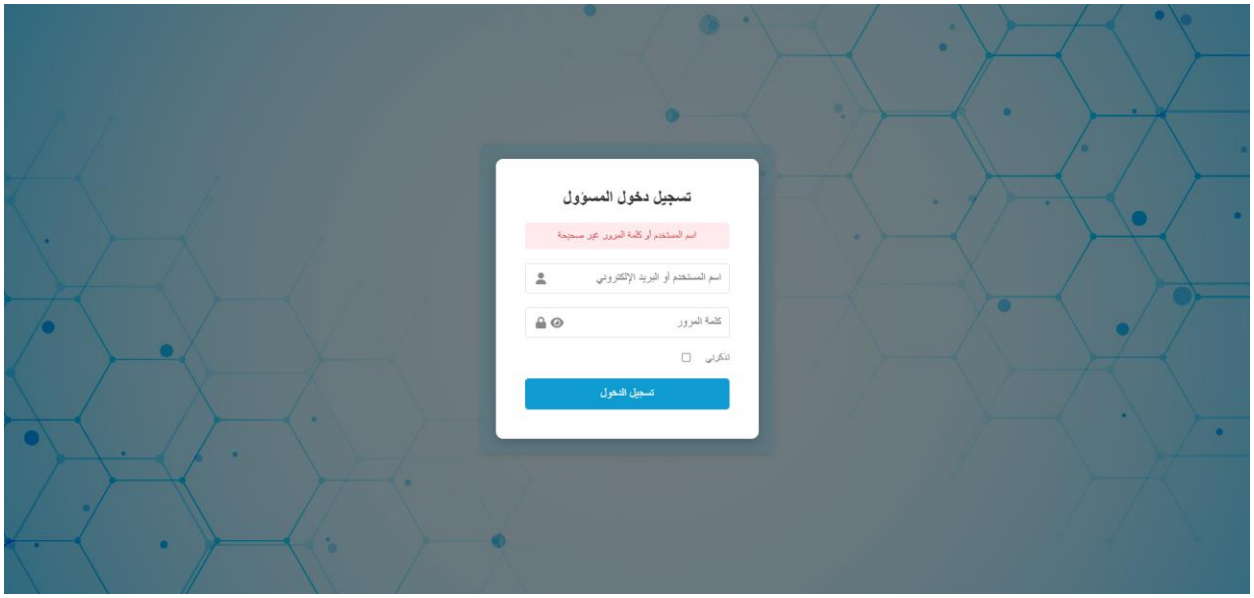
(5.3) شكل واجهة تسجيل الدخول



(5.4) شكل واجهة تسجيل الدخول لمسؤول النظام.

- **فحص الإدخال الخاطئ:**

في البداية، أخطأ مسؤول النظام في تعبئة النموذج بشكل صحيح. عند حدوث هذا الخطأ، قام النظام بتنبيه مسؤول النظام ولن تتم عملية تسجيل الدخول



(5.5) شكل واجهة ادخال خاطئ لاسم مستخدم او كلمة مرور.

5.2.2 فحص تكامل النظام

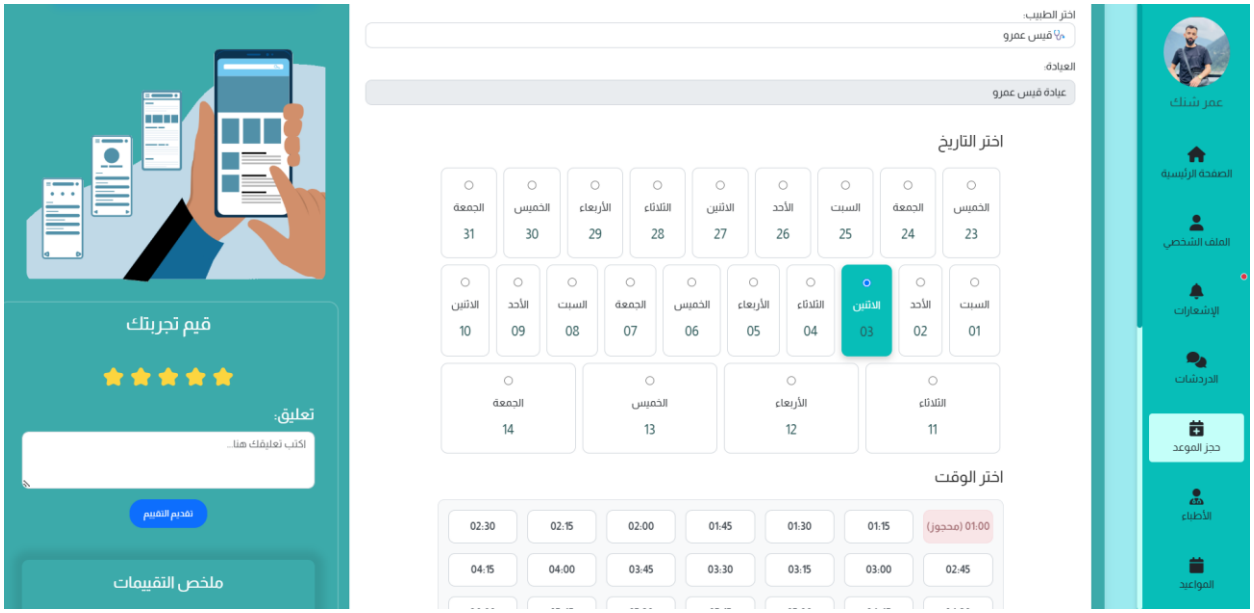
- **إعادة تعيين كلمة المرور:**

يقوم المستخدم بالضغط على زر اعادة تعيين كلمة المرور ثم يتم الوصول إلى لوحة إعادة تعيين كلمة المرور.

(5.6) شكل واجهة طلب كلمة مرور جديدة

- **حجز المواعيد:**

يقوم المريض بحجز من خلال واجهة حجز المواعيد ليصل بعد ذلك اشعار لطبيب بقيام المريض بحجز موعد جديد.



(5.7) شكل واجهة حجز المواعيد الخاصة بالمريض

(5.8) شكل واجهة اشعارات الطبيب

- **التواصل بين الطبيب و المريض:**

تُسهّل الواجهة (5.8) إدارة المحادثات بين الطبيب والمريض بشكل منظم، حيث تتيح إرسال واستقبال الرسائل مع عرض التوقيت، وتوفر خيارات تنقل واضحة للوصول إلى باقي أقسام النظام

(5.9) شكل واجهة المحادثات بين الطبيب والمريض

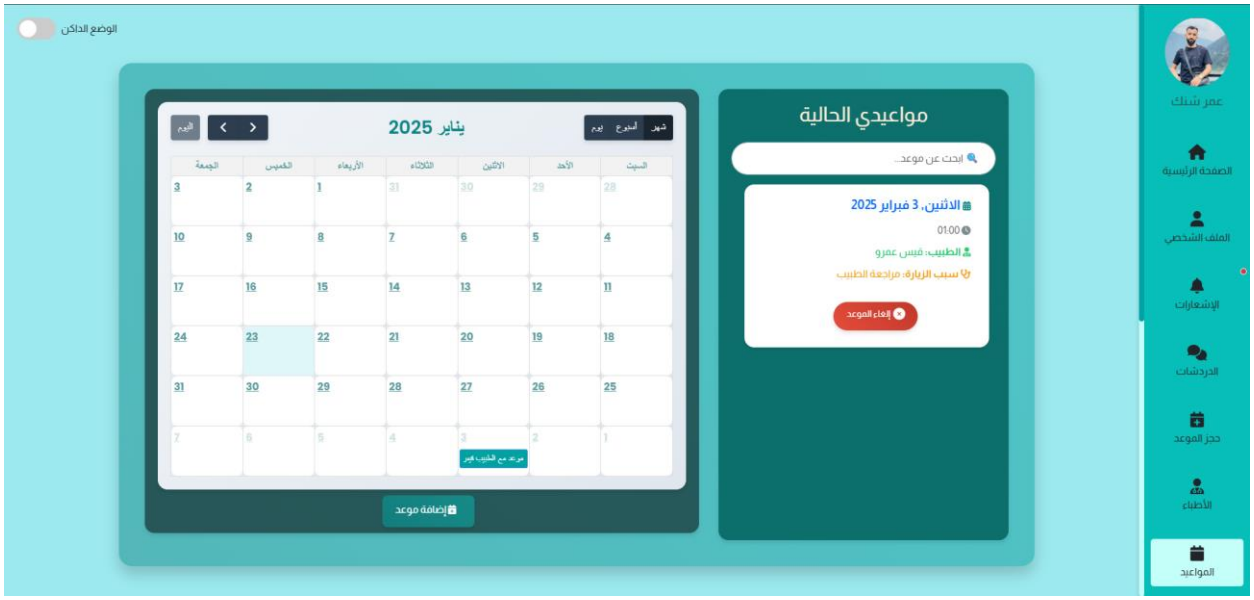
- **إدارة مواعيد العيادة:**

يقوم الطبيب بإضافة أيام تكون العيادة فيها غير متاحة وتكون ظاهرة للمرضى عند القيام بحجز موعد

(5.10) شكل واجهة إدارة مواعيد العيادة.

- **إدارة مواعيد المرضى:**

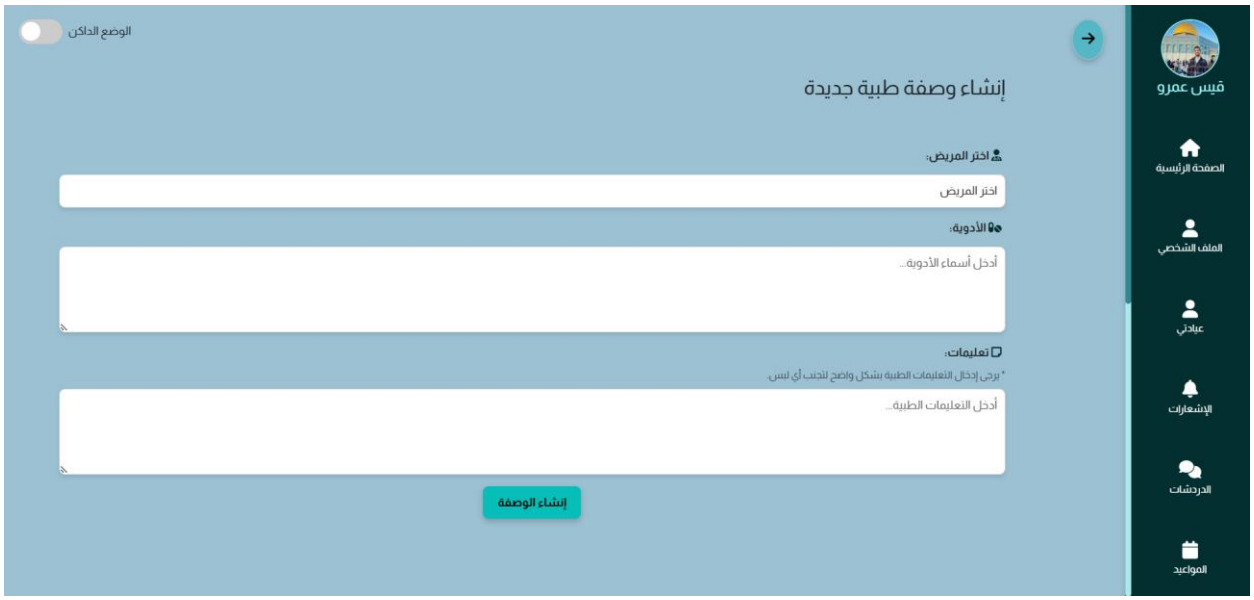
يستطيع الطبيب القيام بإضافة او حذف او تعديل على مواعيد المرضى من الخلال الواجهة التالية ليصل بعد ذلك اخطار للمريض بقيام الطبيب بتعديل على موعده.



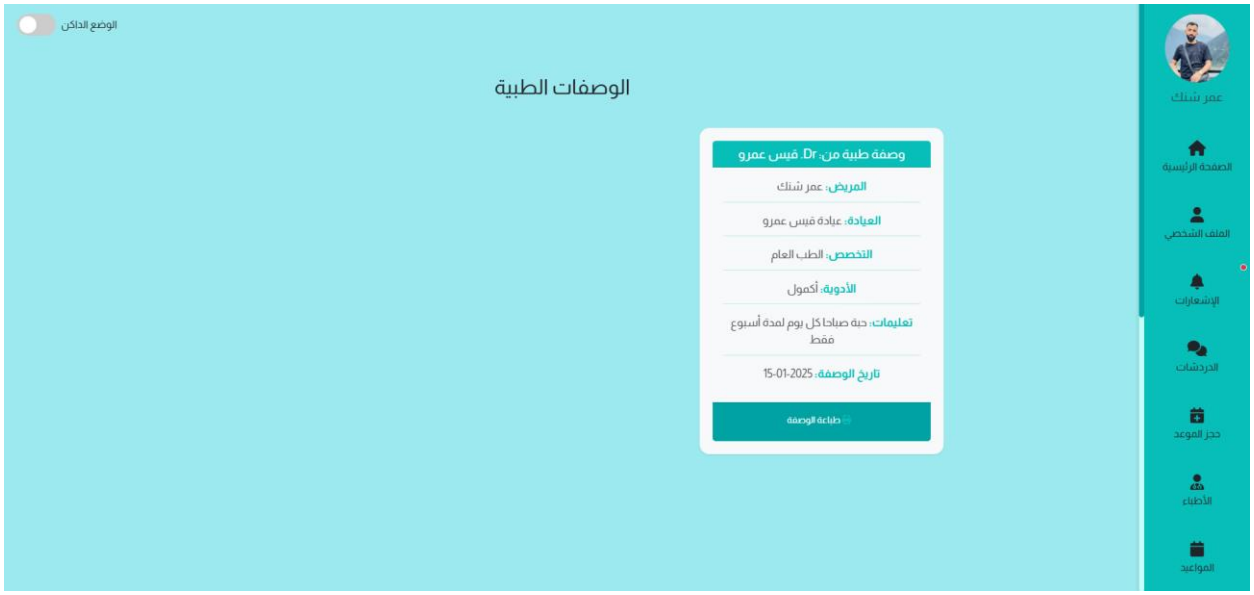
(5.11) شكل واجهة إدارة مواعيد المرضى.

- **كتابة الوصفات الطبية:**

يقوم الطبيب بإنشاء وصفة طبية للمريض بعد تعبئة كامل النموذج في الواجهة التالية ولا يقوم النظام بإنشاء الوصفة الا بعد تعبئة كامل النموذج.



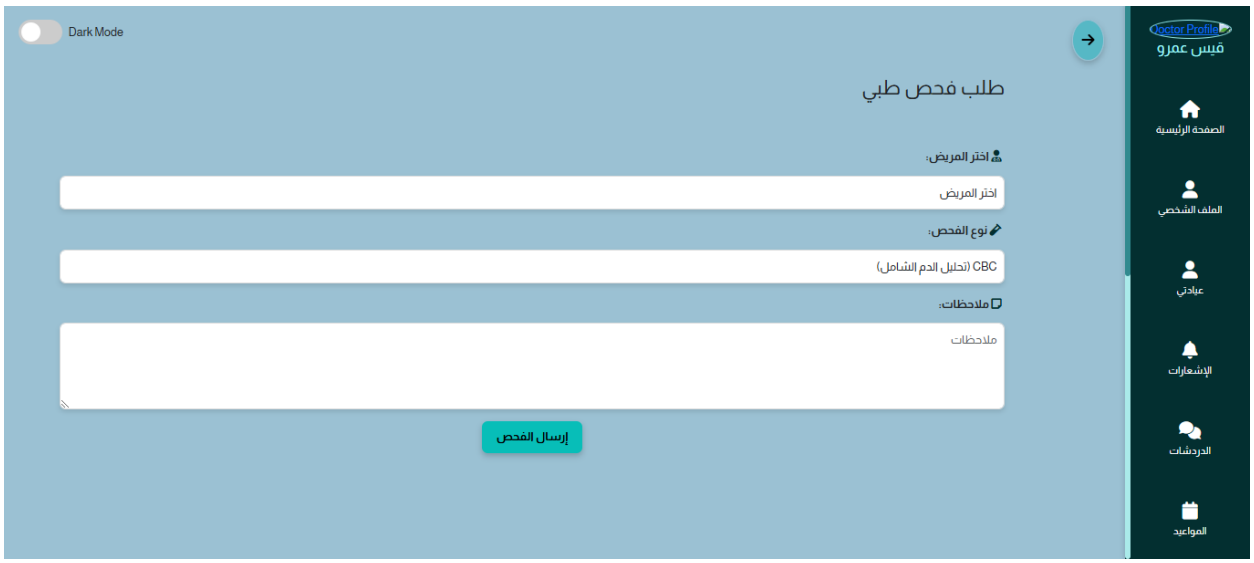
(5.12) شكل واجهة انشاء الوصفات الطبية.



(5.13) شكل واجهة الفحوصات الطبية عند المريض.

- **طلب اجراء الفحوصات الطبية:**

يمكن لطبيب طلب فحص طبي من خلال تحديد اسم المريض ونوع الفحص المطلوب، بعدها يقوم النظام بإرسال طلب الفحص لعامل المختبر.



(5.14) شكل واجهة إنشاء طلب الفحص الطبي عند الطبيب.

(5.15) شكل واجهة طلبات الفحوصات الطبية عند الطبيب.

- **انشاء الفحوصات الطبية:**

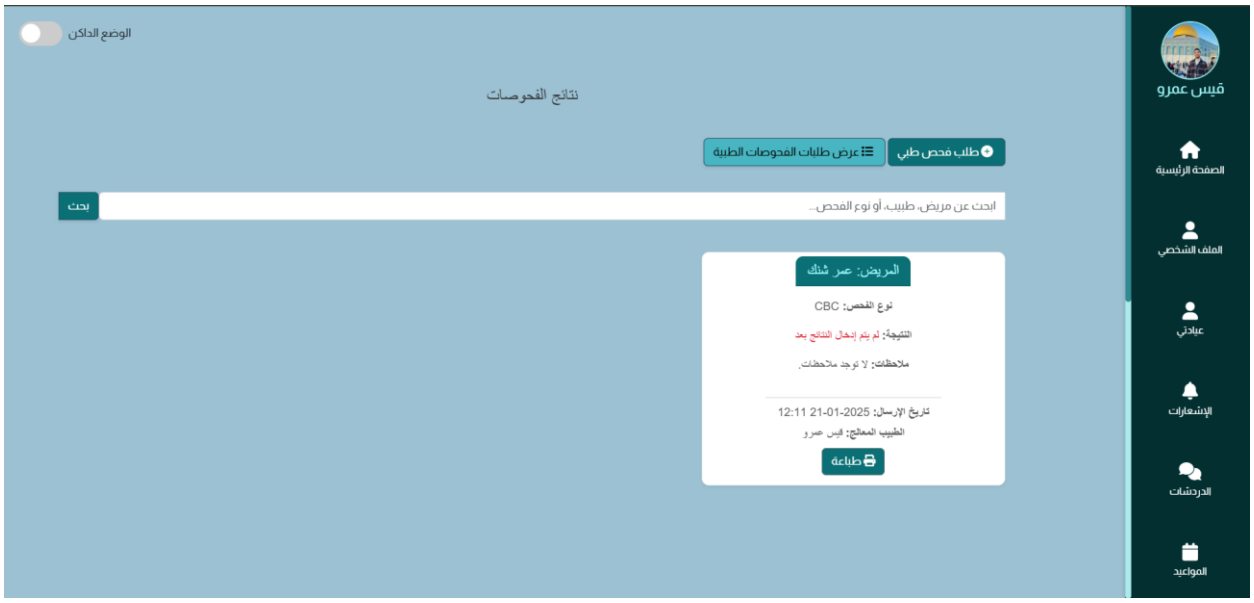
بعد قيام فني المختبر بتسجيل الدخول للنظام يستطيع كتابة الوصفات الطبية وارسالها للنظام بعد تعبئة كامل النموذج.

(5.16) شكل واجهة انشاء الفحوصات الطبية عند فني المختبر.

يمكن لفني المختبر ان يقوم بإنشاء وصفات طبية غير مكتملة بدون تعبئة النموذج بشكل سابق وذلك لحفظ خانة الفحص وتعديلها واكمالها بعد صدور نتيجة الفحص.

(5.17) شكل واجهة نتائج الفحوصات عند فني المختبر.

بعد قيام فني المختبر بالخطوات السابقة يقوم النظام بإرسال نتائج الفحوصات الطبية الى الطبيب. تظهر الواجهة التالية فحص غير مكتمل أي ان فني المختبر لم يقوم بإضافة النتيجة بعد.



(5.18) شكل واجهة نتائج الفحوصات غير المكتملة عند الطبيب.

بعد قيام فني المختبر بإضافة نتيجة الفحص تظهر نتيجة الفحص عند الطبيب مع اشعار لطبيب باكتمال الفحص الطبي.

(5.19) شكل واجهة الفحوصات المكتملة عند الطبيب.

5.3 ملخص الفصل

في هذا الفصل، تناولنا عملية فحص النظام الخاص بموقع بوابة العيادة الطبية. يشمل فحص النظام مرحلتين أساسيتين: فحص وحدات النظام وفحص تكامل النظام. الهدف من هذه الفحوصات هو ضمان أن نظام يعمل بكفاءة وفعالية ويحقق المتطلبات المحددة.

الفصل السادس

الخاتمة والخطط المستقبلية

المحتويات:

6.1 الخاتمة

6.2 الخطط المستقبلية

6.1 الخاتمة

يُعد مشروع بوابة العيادة الطبية نقلة نوعية في تقديم الخدمات الطبية الإلكترونية، حيث يهدف إلى تحسين تجربة المرضى وتسهيل إدارة الخدمات الطبية بشكل شامل. بفضل الوظائف المتنوعة التي يقدمها، مثل إدارة المواعيد، عرض وصفات طبية، وإتاحة منصة للأطباء لإدارة بيانات المرضى وسجلاتهم الطبية، يُحقق المشروع التكامل المطلوب بين المرضى، الأطباء، والمسؤولين.

تم تصميم النظام ليكون بسيطًا، عمليًا، وقابلًا للتوسع، مما يُعزز من كفاءة العمل الطبي ويُسهم في تقديم خدمات ذات جودة عالية. النتائج التي توصلنا إليها تُظهر أن النظام قادر على معالجة التحديات التقليدية في إدارة العيادات الطبية وتحسين تجربة المستخدمين.

6.2 الخطط المستقبلية

لضمان استمرارية المشروع وتوسيع نطاق تأثيره، نُخطط لإضافة الميزات التالية:

1. دمج الذكاء الاصطناعي لتحليل البيانات الصحية

تطوير ميزة تُساعد الأطباء في اقتراح تشخيصات أولية بناءً على البيانات التاريخية للمرضى.
وتقديم توصيات شخصية للمرضى لتحسين نمط حياتهم بناءً على الفحوصات..

2. تطبيق للهاتف المحمول

تطوير تطبيقات للهواتف الذكية (iOS & Android) لتسهيل الوصول إلى النظام في أي وقت ومن أي مكان.
توفير إشعارات فورية للمرضى عند تحديث المواعيد أو الوصفات.

3. توسيع التكامل مع الأجهزة الطبية

ربط النظام الإلكتروني في العيادات أو المستشفيات بأجهزة الفحص الطبية المختلفة، مثل أجهزة قياس الضغط، أجهزة فحص الدم، أو أجهزة التصوير الطبي (مثل الأشعة السينية، الموجات فوق الصوتية، إلخ). الهدف من هذا التكامل هو أن يتم تحديث بيانات المريض بشكل تلقائي وفوري في سجلاته الطبية داخل النظام بمجرد أن يتم إجراء الفحص. على سبيل المثال عند قياس ضغط الدم للمريض، يمكن أن تُرسل القراءة مباشرة إلى سجل المريض في النظام الإلكتروني دون الحاجة لإدخال البيانات يدويًا.

4. أتمتة الفواتير والمدفوعات

إضافة نظام مدفوعات إلكترونية يتيح للمرضى دفع تكاليف المواعيد والخدمات الطبية مباشرةً من خلال البوابة.

5. تحسين الأمان والخصوصية

تعزيز بروتوكولات التشفير وأمان البيانات لضمان الامتثال للوائح حماية البيانات (مثل GDPR).

المراجع

1. Laravel Documentation: [https://laravel.com/docs/11.x]

2. CS 410/510 - Software Engineering [https://cs.ccsu.edu/~stan/classes/CS410/Notes16/06-ArchitecturalDesign.html]

3. مستشفى دورا الحكومي.

4. وزارة الصحة الفلسطينية.