الجمهورية الجزائرية الديموقراطية الشعبية
وزارة التكوين والتعليم المهنيين
المعهد الوطني المتخصص في التكوين المهني
الشهيد داود صالح بالمسيلة

مذكرة تخرج مكملة لنيل شهادة تقني سام
اختصاص المعلوماتية خيار: قاعدة المعطيات

بعنوان:

تصميم وإنجاز منصة إلكترونية
تابعة للمؤسسة العمومية للصحة الجوارية - المسيلة

من إعداد المتربص:
شتاح نسيم

لجنة المناقشة

الإسم واللقب	الصفة	الرتبة
شريفي حمزة	رئيسا	مدير فرعي للدراسات
سريش سفيان	مناقشا	أستاذ
حداد محمد	مناقشا	أستاذ
بوجلال الحسين	مشرفا	أستاذ

دفعة مارس 2025

الجمهورية الجزائرية الديموقراطية الشعبية
وزارة التكوين والتعليم المهنيين
المعهد الوطني المتخصص في التكوين المهني
الشهيد داود صالح بالمسيلة

مذكرة تخرج مكملة لنيل شهادة تقني سام
اختصاص المعلوماتية خيار: قاعدة المعطيات

بعنوان:

تصميم وإنجاز منصة إلكترونية
تابعة للمؤسسة العمومية للصحة الجوارية - المسيلة

من إعداد المتربص:
شتاح نسيم

لجنة المناقشة

الإسم واللقب	الصفة	الرتبة
شريفي حمزة	رئيسا	مدير فرعي للدراسات
سريش سفيان	مناقشا	أستاذ
حداد محمد	مناقشا	أستاذ
بوجلال الحسين	مشرفا	أستاذ

دفعة مارس 2025

شكر وتقدير

شكر خاص لكل من ساهم في هذا العمل

المحتويات

مقدمة.....11
I الإشكالية.....11
الفصل الأول: الدراسة النظرية.....12
I تمهيد.....12
1 البرمجيات الحرة والمفتوحة المصدر.....12
2 لغات البرمجة الآمنة للذاكرة.....13
3 أفضل ممارسات التطوير المستدامة.....14
II قواعد المعطيات.....15
1 أنواع قواعد المعطيات.....15
2 أنظمة تسيير قواعد المعطيات.....16
III واجهة برمجة التطبيقات.....17
1 واجهات برمجة التطبيقات REST.....17
IV التطبيقات العابرة للمنصات والمبنية بتقنيات الWeb.....18
V السجل الطبي الإلكتروني.....19
1 فوائد السجلات الطبية الإلكترونية وأهميتها في رقمنة القطاع الصحي.....19
2 الفرق بين EMR وEHR.....20
3 مصطلحات ومرادفات.....21
3.ا المفهوم.....21
3.ب الملاحظة.....21
3.ج الزيارة.....21
3.د الطلب.....22
الفصل الثاني: الدراسة المسبقة.....23
I تقديم الدراسة.....23
1 التعريف بالمؤسسة.....23
2 بطاقة تقنية للمؤسسة.....23
2.ا الوضعية والتركيب.....23
2.ب التغطية الإدارية.....23
2.ج التنظيم الصحي للمؤسسة.....24
2.د التغطية السكانية للمؤسسة حسب كل بلدية.....24
3 مهام المؤسسة.....25
4 الهيكل التنظيمي للمؤسسة.....26

II دراسة الموجود.....27
1 تعريف دراسة الموجود.....27
2 تحديد مجال الدراسة.....27
3 الهيكل التنظيمي لمجال الدراسة.....27
4 دراسة مناصب العمل.....28
4.ا منصب رئيس مكتب تنظيم النشاطات الصحية ومتابعتها وتقييمها.....28
4.ب منصب رئيس مكتب القبول.....30
5 دراسة تدفق المعلومات.....32
5.ا مخطط تدفق المعلومات.....32
5.ب وصف التدفق.....32
6 دراسة الوثائق.....33
6.ا قائمة الوثائق المدروسة.....33
III قائمة وصف إجراءات العمل.....44
1 قائمة إجراءات العمل.....44
2 جدول وصف إجراءات العمل.....45
IV النقد والاقتراحات.....46
1 النقد.....46
2 الاقتراحات.....46
الفصل الثالث: الدراسة التصميمية.....47
I قاموس المعطيات.....47
1 قاموس المعطيات الصافي.....47
II النموذج التصوري للمعطيات.....50
1 مخطط MCD.....51
1.ا جدول وصف الأفراد.....52
1.ب جدول وصف العلاقات.....54
III النموذج المنطقي للمعطيات.....57
1 نموذج MLD.....57
الفصل الرابع: الدراسة التقنية.....58
I حزمة التطوير.....58
1 PostgreSQL.....58
2 Podman.....58
3 Axum & SQLx.....59
3.ا Axum.....59

3.ب SQLx.....59
4 Tauri.....60
5 Svelte & Tailwind.....60
5.ا Svelte.....60
5.ب Tailwind.....61
6 Github Actions.....61
الفصل الخامس: الإنجاز البرمجي.....62
I منصة الرازي.....62
1 أبو بكر محمد بن زكريا الرازي.....62
2 المميزات الرئيسية.....62
3 الرؤية التطلعية.....62
4 بنية المنصة.....63
4.ا قاعدة المعطيات.....63
4.ب واجهة برمجة التطبيقات.....63
4.ج تطبيق عابر للمنصات.....63
II استراتيجيات التطوير.....64
1 العزل بغرض الحماية.....64
2 تشفير الاتصال.....65
3 إدارة المستخدمين وصلاحياتهم.....65
4 دوال الاستعلام.....67
III دليل التشغيل.....70
1 الخادم.....70
1.ا المتطلبات.....70
1.ب الخطوات.....70
2 تطبيق RhazesEMR Client.....71
2.ا واجهة التطبيق.....72
IV النموذج الفيزيائي للمعطيات.....76
1 وصف النموذج الفيزيائي للمعطيات MPD.....76

قائمة الجداول

جدول 1: تطبيقات مبنية بـElectron أو/و React Native.....18
جدول 2: التغطية السكانية للمؤسسة حسب كل بلدية.....24
جدول 3: منصب العمل الأول - الوثائق المنجزة.....28
جدول 4: منصب العمل الأول - الوثائق الصادرة.....29
جدول 5: منصب العمل الأول - الوثائق الواردة.....29
جدول 6: منصب العمل الثاني - الوثائق المنجزة.....30
جدول 7: منصب العمل الثاني - الوثائق الصادرة.....31
جدول 8: منصب العمل الثاني - الوثائق الواردة.....31
جدول 9: قائمة الوثائق المدروسة.....33
جدول 10: بطاقة تحليلية لوثيقة القائمة الاسمية لعمال العيادة.....34
جدول 11: بطاقة تحليلية لوثيقة توزيع الأطباء على التخصصات.....35
جدول 12: بطاقة تحليلية لوثيقة جدول العيادات.....36
جدول 13: بطاقة تحليلية لوثيقة قائمة التخصصات.....37
جدول 14: قائمة اسمية للأطباء على مستوى العيادة + مساعد طبيب.....38
جدول 15: وصفة طبية.....39
جدول 16: تفاصيل الموارد البشرية للمرافق.....40
جدول 17: تعداد الأطباء في المرفق الخاص.....41
جدول 18: تعداد الأطباء في المرفق الخاص.....42
جدول 19: تعداد الأطباء في المرفق الخاص.....43
جدول 20: قائمة إجراءات العمل.....44
جدول 21: وصف إجراءات العمل.....45
جدول 22: وصف الأفراد.....52
جدول 23: وصف العلاقات.....54
جدول 24: وصف الصلاحيات.....66
جدول 25: توزيع الصلاحيات على الأدوار.....66
جدول 26: أمثلة على توظيف مكتبة SQLx في دوال الاستعلام.....67
جدول 27: MPD (person).....76
جدول 28: MPD (person_name).....77
جدول 29: MPD (person_address).....78
جدول 30: MPD (patient).....79
جدول 31: MPD (patient_identifier).....80

جدول 32: MPD (patient_identifier_type) 81
جدول 33: MPD (concept) 81
جدول 34: MPD (concept_class) 82
جدول 35: MPD (concept_datatype) 83
جدول 36: MPD (provider) 84
جدول 37: MPD (visit) 85
جدول 38: MPD (visit_type) 86
جدول 39: MPD (encounter_provider) 86
جدول 40: MPD (encounter) 87
جدول 41: MPD (obs) 88
جدول 42: MPD (drug) 89
جدول 43: MPD (orders) 90
جدول 44: MPD (order_type) 91
جدول 45: MPD (location) 92
جدول 46: MPD (program) 93
جدول 47: MPD (patient_program) 94

قائمة الأشكال

شكل 1: طريقة عمل واجهة برمجة التطبيقات.....17
شكل 2: طريقة عمل الـREST API.....17
شكل 3: رسم تخطيطي يوضح عمل الـEMR.....19
شكل 4: الفرق بين EMR وEHR.....20
شكل 5: الهيكل التنظيمي للمؤسسة.....26
شكل 6: الهيكل التنظيمي لمجال الدراسة.....27
شكل 7: مخطط تدفق المعلومات.....32
شكل 8: وصف التدفق.....32
شكل 9: مثال عن تصميم مخطط نموذج MCD.....50
شكل 10: مخطط MCD.....51
شكل 11: شعار PostgreSQL.....58
شكل 12: شعار podman.....58
شكل 13: شعار Tauri.....60
شكل 14: شعار Svelte.....60
شكل 15: شعار TailwindCSS.....61
شكل 16: شعار gh Actions.....61
شكل 17: Portrait of Rhazes.....62
شكل 18: رسم توضيحي لقاعدة معطيات معزولة عن الشبكة تتصدى لهجوم سيبراني.....64
شكل 19: رسم توضيحي لعمل الـReverse Proxy.....65
شكل 20: docs.rhazesemr.info.....71
شكل 21: rhazesemr.info/download.....71
شكل 22: الواجهة الرئيسية لتطبيق RhazesEMR (Desktop).....72
شكل 23: تصميم تطبيق RhazesEMR (Desktop).....73
شكل 24: الواجهة الرئيسية لتطبيق RhazesEMR (Mobile).....74
شكل 25: تصميم تطبيق RhazesEMR (Mobile).....75

مقدمة

شهد القطاع الصحي في العالم تحولًا جذريًا نحو الرقمنة استجابة للتطورات التكنولوجية المتسارعة. حيث أصبحت رقمنة القطاع الصحي ضرورة حتمية لمواكبة التحديات المعاصرة وتحسين جودة الخَدمات الصحية المقدمة. وتعد أنظمة السجلات الطبية الإلكترونية (EMR) إحدى أهم ركائز هذا التحول، والتي تطورت تدريجيًا لتلبية احتياجات المؤسسات الصحية في مختلف المجالات الطبية.

ونظرًا لأهمية هذا التحول الرقمي، لابد من الاستفادة القصوى من التطور التكنولوجي الحاصل في مجال المعلوماتية الصحية من خلال توظيف أنظمة السجلات الطبية الإلكترونية في معالجة وإنجاز العمليات مما يؤدي إلى ربح الوقت وتوفير الجهد وتحقيق تحكم مستقر في تسيير المعطيات الطبية بسهولة ودقة. ومن خلال معاينتنا لميدان الدراسة واطلاعنا على المهام المرتبطة بالسجلات الطبية وإدارتها وبعد دراسة مختلف الوثائق المتداولة وملاحظة عمل الطاقم الطبي والإداري.

تم اقتراح موضوع بعنوان تصميم وإنجاز منصة رقمية تابعة للمؤسسة العمومية للصحة الجوارية - المسيلة كمحاولة منا للمساهمة في رقمنة القطاع الصحي وتحسين التسيير والوصول السريع للمعلومات الطبية اللازمة وذلك بتوظيف معارفنا التي تلقيناها خلال فترة تكويننا.

فقد تبين لنا أن هناك عدة مشاكل تعيق سير العمل في المرافق الصحية، ومن أهمها:

- هدر الوقت في البحث عن السجلات الطبية السابقة.
- صعوبة مشاركة المعلومات بين مختلف الأقسام الطبية.
- تكرار الفحوصات والتحاليل الطبية الغير الضرورية.
- العقبات التي يواجهها الطاقم الطبي في إدارة مواعيد المرضى وملفاتهم الطبية.

وهذه نتيجة لغياب الرقمنة المتكاملة في عملية تسجيل وحفظ بيانات المرضى. مما دفعنا إلى طرح التساؤل التالي:

I الإشكالية

ما مدى فعالية إنجاز منصة رقمية لإدارة السجلات الطبية في إطار رقمنة القطاع الصحي بالمؤسسة العمومية للصحة الجوارية بالمسيلة؟

الفصل الأول: الدراسة النظرية

I تمهيد

يستند هذا المشروع على مجموعة من مبادئ التطوير الحديثة التي تضع الشفافية، الأمان وممارسات التطوير المستدامة في مقدمة أولوياتها. تم اختيار هذه المبادئ بعناية لضمان إنشاء نظام متين، قابل للصيانة ومواكب لمتطلبات البرمجة الحديثة. في حين أن نُهُج التطوير التقليدية قد أدت دورها في الماضي، إلا أن التطور التقني السريع يتطلب منظورًا جديدًا يتبنى أفضل الممارسات المعاصرة. تبدي الأقسام التالية المبادئ الأساسية التي وجهت قراراتنا التقنية والمسار العام للمشروع:

1 البرمجيات الحرة والمفتوحة المصدر

لا يخفى على أحد مدى أهمية البرمجيات الحرة والمفتوحة المصدر في التكنولوجيا المعاصرة. فمشاريع مثل نواة لينِكس، FFmpeg[1] وNGINX[2] هي أمثلة من قائمة طويلة من البرمجيات الحرة والمفتوحة المصدر التي تُشغّل الملايين، إن لم يكن المليارات، من الأجهزة حول العالم. في الواقع، بعض هذه التقنيات تجاوزت حدود كوكبنا! فعلى سبيل المثال، تستخدم مركبة Perseverance التابعة لـNASA على سطح المريخ برنامج FFmpeg لضغط الصور ومقاطع الفيديو قبل إرسالها إلى الأرض[1]. أما نظام التشغيل الذي يدير هذه الآلات المذهلة؟ إنه مبني، بالطبع، على نظام لينِكس.

إن التبني العالمي للبرمجيات الحرة والمفتوحة المصدر[2] يؤكد أهميتها الاستراتيجية في البنية التحتية الرقمية الحديثة. تُعد ألمانيا مثالًا بارزًا على ذلك؛ ففي عام 2020، تبنت قواتها المسلحة بروتوكول Matrix للاتصالات الداخلية، حيث طورت تطبيقها الخاص المبني على Element[3]. كما كانت مختلف الولايات والمدن الألمانية سباقةً في تبني البرمجيات الحرة والمفتوحة المصدر؛ مؤخرًا، أعلنت ولاية شلسفيغ هولشتاين عن خططها لترقية 30,000 كمبيوتر شخصي في الإدارات العامة إلى نظام لينكس وحلول برمجية حرة أخرى مثل LibreOffice وThunderbird[2]. أما هدفهم؟ فهو تحقيق الاستقلالية والشفافية الرقمية مع دعم اقتصادهم الرقمي المحلي. أتمنى رؤية بلدي ينضم إلى هذه الحركة؛ فهذا لن يجعلني مواطنًا فخورًا فحسب، بل سيتيح لي كمطور فرصة المساهمة المباشرة في التقدم التكنولوجي لوطني.

1 برنامج حر ومفتوح المصدر لمعالجة ملفات الوسائط المتعددة، بما في ذلك الفيديو والصوت.

2 برنامج خادم web حر ومفتوح المصدر يُستخدم لتقديم المحتوى، وتعزيز سرعة وأداء المواقع.

وعليه، يلتزم هذا المشروع بشكل صارم بمبادئ البرمجة الحرة والمفتوحة المصدر، وهو مرخص تحت رخصة غنو أفّيرو لعامة الشعب، الإصدار الثالث (AGPLv3)، مما يضمن نفس الحريات والشفافية التي نؤمن بها ونسعى لتحقيقها.

2 لغات البرمجة الآمنة للذاكرة

في عام 2007، كانت Jean Bookout تقود سيارتها Toyota Camry طراز 2005 عندما تسارعت السيارة فجأة وفقدت السيطرة عليها، مما أدى إلى حادث مأساوي نتج عنه إصابتها بجروح بالغة ووفاة صديقتها المقربة Barbara Schwarz. كشف التحقيق الشامل، الذي أُجري استجابة لأكثر من 6,200 شكوى مماثلة، عن خلل جوهري في نظام التحكم الإلكتروني بصمام الخانق (ETCS). وأظهرت التحليلات أن النظام كان يعاني من عيوب برمجية خطيرة متعلقة بإدارة الذاكرة[4][5]، مما يؤكد أن لغة البرمجة C، رغم قوتها وكفاءتها، تحمل مخاطر كامنة في التعامل مع الذاكرة. تشمل هذه المخاطر تجاوز سعة المخزن المؤقت[1]، وإلغاء مرجعية المؤشرات غير الصالحة[2]، وحالات التزامن غير المنضبط[3]؛ وهي عيوب يصعب اكتشافها وتصحيحها[6].

في عام 2023، دشنت وزارة الدفاع الأمريكية مشروع TRACTOR، وهي مبادرة طموحة تهدف إلى تحويل أكثر من 3 ملايين سطر برمجي من لغة C إلى لغة Rust[7][8]. يمثل هذا المشروع، الذي تم تطويره بالتعاون مع وحدة الابتكار الدفاعي (DIU)، نقلة نوعية في كيفية تعامل المؤسسات الكبرى مع أمان الذاكرة[7]. يستخدم المشروع تقنيات الذكاء الاصطناعي المتقدمة لتحليل الشفرة المصدرية وفهم منطقها وترجمتها تلقائيًا إلى Rust مع الحفاظ على وظائفها الأصلية. وقد حقق المشروع نجاحات ملموسة في تحويل عدد معتبر من الأنظمة، منها نظام إدارة الاتصالات العسكرية وأنظمة التحكم في المعدات، مما يثبت أن الانتقال إلى لغات البرمجة الآمنة للذاكرة ليس مجرد مفهوم نظري سليم فحسب، بل قابل للتحقيق عمليًا على نطاق واسع[9].

وعليه، يعتمد هذا المشروع على لغة Rust التي تتميز بنظام ملكية صارم وضوابط أمنية مدمجة في تصميمها. تضمن هذه الميزات القضاء على الثغرات المتعلقة بالذاكرة في مرحلة البناء، مثل تلك التي تسببت في حادث

1 أو Buffer Overflow، خطأ برمجي يحدث عندما يحاول البرنامج تخزين بيانات في منطقة من الذاكرة متجاوزا الحد المخصص.

2 أو Invalid Pointer Dereferencing، وهو محاولة فاشلة للوصول إلى موقع في الذاكرة عبر مؤشر.

3 أو Race Conditions، حالة تحدث في البرامج متعددة المسارات عندما يحاول مساران أو أكثر الوصول إلى نفس المورد.

Toyota، بدلًا من اكتشافها في مرحلة الإنتاج. كما يوفر نظام التحقق في وقت البناء[1] منعًا تلقائيًا لمشاكل تجاوز سعة المخزن المؤقت وإلغاء مرجعية المؤشرات غير الصالحة، مما يجعل البرامج أكثر أمانًا وموثوقيه.

3 أفضل ممارسات التطوير المستدامة

مواكبين للتطور التقني الهائل الذي شهدته مختلف مجالات تطوير وهندسة البرمجيات، يعتمد الجانب التطبيقي من هذا العمل على أحدث هذه التقنيات وأكثرها فعالية. أمثلة على هذه الأدوات نذكر:

- Version Control System (Git): استعمل لتتبع تطور وإصدارات المشروع.
- Github Actions: أَتْمَتَةٌ تامة لمختلف الاختبارات وعمليات البناء.
- Containerization Technology: وُظِفَت لضمان استقرار وسلاسة سير البرامج بِغَض النظر عن نظام التشغيل المعتمد
- استخدام قاعدة بيانات هجينة لتلبية مختلف احتياجات النظام دون الحاجة إلى التزام حدود تعيق تطوره.

1 أو Compile-time Checking، وهي آلية تكشف الأخطاء البرمجية أثناء تحويل الشفرة إلى برنامج تنفيذي.

II قواعد المعطيات

قواعد البيانات هي مجموعة منظمة من المعلومات أو البيانات[1] المهيكلة، يتم تخزينها والوصول إليها إلكترونيًا. اليوم، تتجاوز قواعد البيانات مجرد تخزين المعلومات فهي أنظمة ذكية تساعد في تنظيم وتحليل وإدارة المعلومات في الوقت الفعلي.[11]

1 أنواع قواعد المعطيات

تتعدد وتختلف أنوع قواعد المعطيات، نذكر منها:

- **قواعد المعطيات الهرمية:** تنظم البيانات في بنية تشبه الشجرة، مما يسمح بالوصول الى المعلومات عن طريق علاقات الأصل والتتابع[12].
- **قواعد المعطيات الشبكية:** تمثل البيانات بطريقة ديناميكية تتميز بكونها مخططًا تظهر فيه أنواع الأفراد كعقدة وأنواع العلاقات كأسهم[13].
- **Graph Databases:** مصممة للتعامل مع العلاقات المعقدة بين البيانات، مثالية لمنصات الشبكات الاجتماعية[14].
- **قواعد معطيات السلاسل الزمنية:** مصممة للتعامل مع السلاسل الزمنية بواسطة أزواج مكونة من الوقت والقيم، حيث يمثل الوقت المعرف الأساسي المعتمد في الأفراد والعلاقات[15].
- **قواعد المعطيات الشعاعية:** تمثل البيانات على أنها أشعة[2] وتنفذ عادة خوارزميات بحث النقطة الأقرب للوصول إلى أقرب النتائج في مقاربتها للاستعلام[16].
- **قواعد المعطيات الموزعة:** توزع البيانات على مختلف أجهزة التخزين، تستعمل عادة لتخزين البيانات الحساسة نظرا لقدرتها على ضمان سرية المعلومات وصحتها[17].

1 تمثيل للمعلومات بصيغة مناسبة للتخزين أو المعالجة أو النقل[10].

2 قائمة أرقام ذات طول ثابت إضافة إلى بيانات أخرى.

- **قواعد المعطيات المبنية على تقنيات سلسلة الكتل (Blockchain):** مزيج بين قواعد المعطيات التقليدية وقواعد المعطيات الموزعة، تدعمها طبقات متعددة من سلاسل الكتل ويتم مشاركة قاعدة المعطيات نفسها على شكل مِلَفّ مشفر وثابت مما يجعل المعلومة متاحة للجميع[18].
- **قواعد المعطيات الكائنية:** تمثل البيانات على أنها كائنات مماثلة لتلك المستعملة في OOP[1][19].
- **قواعد المعطيات العلائقية:** تمثل البيانات في جداول تربط بينها علاقات وتستغل لغة SQL لإجراء مختلف استعلامات CRUD.
- **قواعد المعطيات غير العلائقية (NoSQL):** مصممة للتعامل مع البيانات التي نمذجة بطرق غير علائقية[20].
- **قواعد المعطيات NewSQL:** تسعى إلى دمج مزايا قواعد البيانات العلائقية والغير علائقية مع الحفاظ على الأداء والتناسق بين المعطيات[21].

2 أنظمة تسيير قواعد المعطيات

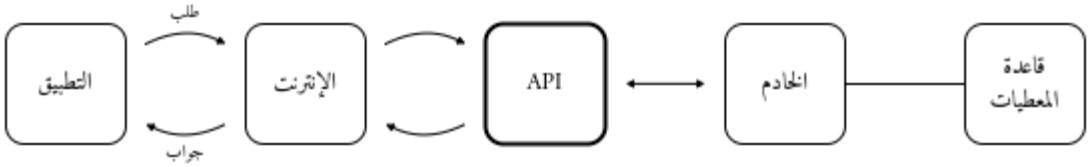
هي حلول برمجية مصممة لإدارة، تنظيم واستخراج البيانات بفعالية. تلعب هذه الأنظمة دورا حاسمًا في الحوسبة الحديثة، حيث تسمح لمختلف المستخدمين بتخزين والتلاعب ببياناتهم في العديد من التطبيقات، سواءًا في البرمجيات الصغيرة أو المنصات الضخمة[22]. ومن أبرز أنظمة تسيير قواعد المعطيات نذكر:

- PostgreSQL, MariaDB, SQLite, Firebird, Drizzle لتسيير قواعد المعطيات العلائقية.
- MongoDB, Apache Cassandra, Redis لتسيير قواعد المعطيات الغير علائقية.
- TimescaleDB لتسيير قواعد معطيات السلاسل الزمنية.
- BigchainDB لتسيير قواعد المعطيات المبنية على تقنيات سلسلة الكتل.

1 نمط برمجة متقدم، وفيه يقسم البرنامج إلى وحدات تسمى الكائنات.

III واجهة برمجة التطبيقات

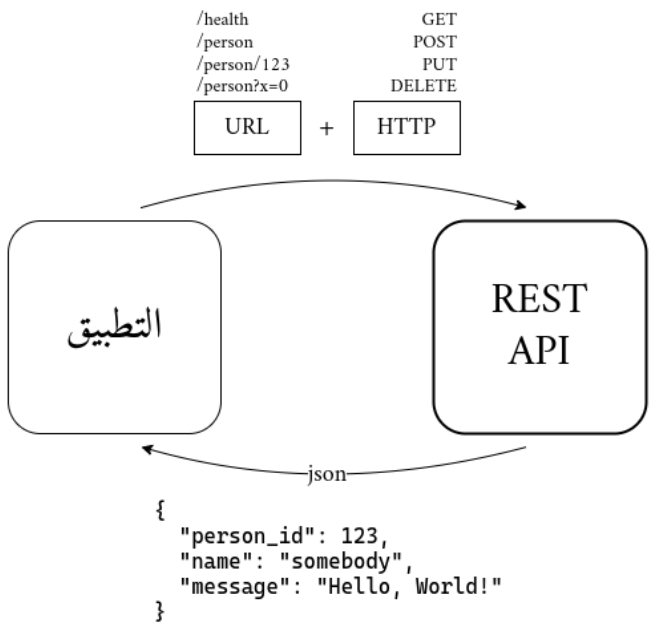
أو Application Programming Interface (API)، وهي آليات تمكن مكونات البرنامَج من الاتصال ببعضها البعض باستخدام مجموعة من الضوابط والبروتوكولات. حيث يُعَرِف مطور الAPI مجموعة من الكائنات والإجراءات التي تستغل لاحقا في مختلف الاستعمالات البرمجية، مثل تطبيقات مساعدة أو إضافات[1] للبرنامج الرئيسي[23].



شكل 1: طريقة عمل واجهة برمجة التطبيقات

1 واجهات برمجة التطبيقات REST

يحدد مجموعة من الوظائف، مثل GET (الحصول) وPUT (التنظيم) وDELETE (الحذف) وغيرها، يمكن للعملاء استخدامها في التواصل مع الخادم وتبادل البيانات باستخدام بروتوكول HTTP. ميزته الرئيسية هي عدم الحالة[2].تماثل طلبات العميل المرسلة إلى خادم الREST API عناوين URL التي تكتبها عند تصفح مواقع الـWeb. أما الجواب من الخادم يكون بيانات عادية (غالبا JSON) بدون عرض رسومي على واجهة الصفحة[23].



شكل 2: طريقة عمل الـ*REST API*

1 أو extension أو plugin وهي المكونات الإضافية التي توسع وظائف البرنامَج.

2 المقصود بعديمة الحالة أن الخوادم لا تحفظ بيانات العميل بين الطلبات.

IV التطبيقات العابرة للمنصات والمبنية بتقنيات الWeb

أو Web-based Cross-platform Applications. تستغل هذه التطبيقات أطر عمل لتحويل صفحات Web من ملفات HTML، JavaScript و CSS إلى حزم يمكن تثبيتها على مختلف الأجهزة. من أطر العمل هذه نذكر:

- **Electron:** إطار عمل حر ومفتوح المصدر يُضّمِن محرك متصفح Chromium داخل حزم التطبيق، الذي يقوم لاحقا بعرض صفحات الWeb.
- **React Native:** إطار عمل حر ومفتوح المصدر طورته شركة Meta. يستعمل هذا الإطار في عدد من تطبيقات التواصل الاجتماعي الرائجة، ويتميز على Electron بترجمته لصفحات الWeb إلى تطبيق محلي.

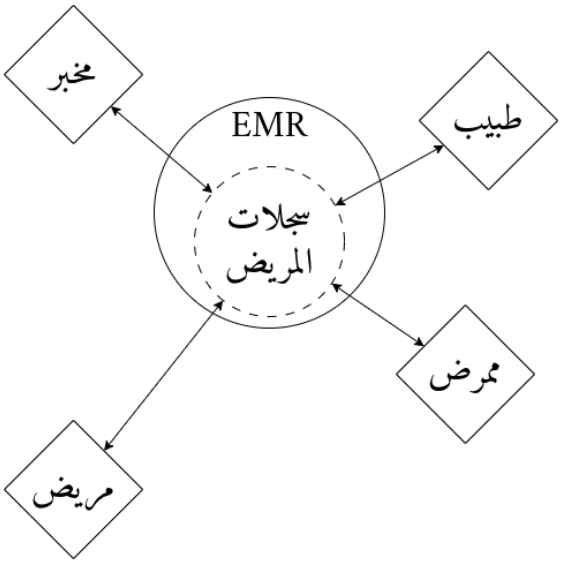
اليوم، تستعمل هذه التقنيات في تطوير عدد معتبر من التطبيقات الرائجة. أمثلة على هذه التطبيقات موضحة في الجدول التالي:

جدول 1: تطبيقات مبنية بـ*Electron* أو/و *React Native*

التطبيق	إطار العمل المستغل	إحصائيات
Pinterest	React Native	أكثر من مليار تحميل على Play Store
Facebook	React Native	أكثر من 10 مليار تحميل على Play Store
Instagram	React Native	أكثر من 5 مليار تحميل على Play Store
Shopify	React Native	أكثر من 10 مليون تحميل على Play Store
Tesla App	React Native	أكثر من مليون تحميل على Play Store
Slack	Electron	42 مليون مستخدم نشط سنة 2024[24]
Visual Studio Code	Electron	يستخدمه 73.6% من المطورين[25]
Spotify	Electron, React Native	675 مليون مستخدم نشط شهريا[26]
Discord	Electron, React Native	259.2 مليون مستخدم نشط شهريا[27]
Microsoft Teams	Electron, React Native	320 مليون مستخدم نشط يوميا[28]

V السجل الطبي الإلكتروني

أو Electronic Medical Record (EMR)، وهو المكافئ الرقمي للسجلات الورقية التي نجدها بمختلف المرفقات الصحية، من عيادات ومستشفيات وما إلى ذلك. المصطلح EMR لا يقتصرُ بِالدلالة على محتوى هذه السجلات وحسب، بل ويشمل النظام المحيط الذي يلتقط، يخزن ويدير جميع البيانات الطبية الخاصة بالمرضى. تحتوى السجلات الطبية الإلكترونية عادة على معلومات مثل تاريخ صحي شامل لكل مريض، تشخيصات، أدوية، حساسيات، نتائج المخابر ومختلف الخطط العلاجية للمرضى[29].



شكل 3: رسم تخطيطي يوضح عمل الـ*EMR*

1 فوائد السجلات الطبية الإلكترونية وأهميتها في ر قمنة القطاع الصحي

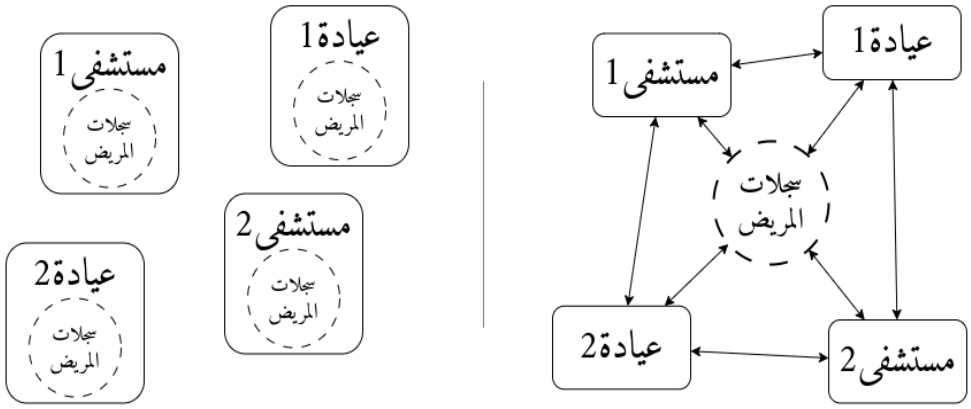
عادت رقمنة خدمات الرعاية الصحية وترقيتها من وضعها الورقي بفوائد عدة على منظمات الرعاية الصحية وأداءها. وتعد السجلات الطبية الإلكترونية من أهم العوامل التي جعلت هذا التحول ممكنًا[29]. من أهم ميزات استغلال الـEMRs نذكر:

- **مركزية معلومات المرضى:** بجمعها لمختلف بيانات المرضى في مكان واحد، يحصل مستغلوها من الموفرين[1] على نَظْرَة أشمل حول الحالة الصحية للمريض. ما يزيد من سداد قراراتهم ودقتها.
- **التنسيق بين المختصين:** يمكن لجميع الأطباء من مختلف التخصصات الوصول إلى السجل الخاص بكل مريض، مما يسمح بتنسيق إجراءات طبية أفضل وأكثر فعالية.
- **إدارة الوصفات الطبية:** يمكن للموفرين إرسال وصفات إلكترونية إلى الصيدليات، تخفيفًا من خطر الأخطاء وتعزيزا لعملية إدارة الوصفات الطبية.

كل ذلك إضافة إلى سرعة الوصول لسجلات رقمية، دقيقة وشاملة للمريض ما يعزز من فعالية وإنتاجية خدمات الرعاية من جانب، ويحقق توفير معتبر في المصاريف من جانب آخر.

2 الفرق بين EMR وEHR

السجل الصحي الإلكتروني أو Electronic Health Record هو أيضا مكافئ رقمي لسجلات المرضى، لكن الفرق الجوهري بينه وبين الـEMR هو تميزه بنطاق أوسع، حيث يمكن للموفرين المرخصين الوصول إلى سجلات المرضى حتى في حال عدم انتمائهم لنفس المفرقات الصحية.



شكل 4: الفرق بين *EMR* و*EHR*

1 يُقصد بهم مقدمو الرعاية الصحية مثل الأطباء والممرضين وغيرهم من المهنيين الصحيين الذين يوفرون الخَدمات الطبية.

3 مصطلحات ومرادفات

تستخدم السجلات الطبية الإلكترونية مصطلحات تقنية قد تبدو مبهمة للمستخدمين الجدد. وبما أن فهم هذه المصطلحات أمر ضروري للاستخدام الفعال لهذه الأنظمة ولتحقيق أقصى استفادة منها، فيمَا يلي شرح لأهم هذه المصطلحات:

3.ا المفهوم

أو Concept. يمكن أن يمثل مختلف المعلومات والمفاهيم الصحية، تُنظم المفاهيم عادةً في هيكل هرمي وتُعرف بمعرفات فريدة مرتبطة بمصطلحات قياسية مثل SNOMED CT أو ICD-11. أمثلة:

- ارتفاع ضغط الدم الأساسي (ICD11: BA00.Z)
- حساسية البنسلين (SNOMED CT: 294513009)
- فصح نسبة السكر في دم الصائم (LOINC: 1558-6)
- باراسيتامول 500 ملغ قرص (NDC: 0904-1982-61)

3.ب الملاحظة

أو Observation (OBS). وهي البيانات الطبية المسجلة عن المريض، مثل معدلات مؤشراته الحيوية، نتائج التحاليل المخبرية أو الأعراض المبلغ عنها. كل ملاحظة تربط بين مفهوم معين وقيمته. أمثلة:

- ضغط الدم، 130/85
- مستوى السكر في دم الصائم: 110 ملغ/ديسيلتر
- الوزن: 75 كغ
- الأعراض المبلغ عنها: سعال مستمر مع بلغم أصفر منذ 3 أيام

3.ج الزيارة

أو Visit. هي سجل يشمل جميع زيارات المريض للموفر، سواءا كانت زيارات شخصية، استشارة عن بعد أو إقامة في المستشفى. يتم توثيق كل لقاء بتاريخ ووقت ومكان ونوع الخدمة المقدمة. أمثلة:

- زيارة شخصية، قسم أمراض القلب، تاريخ: 05/01/2025، الموفر: د. أحمد
- استشارة طبية عن بعد، تاريخ: 10/01/2025، الموفر: د. سمير
- إقامة في المستشفى، قسم الجراحة، تاريخ الدخول: 15/01/2025، تاريخ الخروج: 20/01/2025، سبب الإدخال: استئصال الزائدة الدودية
- زيارة روتينية، عيادة السكري، تاريخ: 25/01/2025

3.د الطلب

أو Order. وهي تعليمات الموفرين للإجراءات الطبية، مثل طلب فحوصات مخبرية، أو أشعة، وصف أدوية أو تحويل المريض إلى متخصص آخر. تحتوي الإجراءات على معلومات مفصلة عن كيفية تنفيذها والغرض منها. أمثلة:

- طلب فحص صورة أشعة للصدر، الأولوية: عادية، الغرض: استبعاد الالتهاب الرئوي
- طلب فحص تعداد الدم الكامل، تاريخ الطلب: 12/01/2025، وقت سحب العينة: صباحًا
- وصفة طبية: أموكسيسيلين 500 ملغ، مرتين يوميًا لمدة 7 أيام، التعليمات: تناول مع الطعام
- تحويل إلى أخصائي أمراض الجهاز الهضمي، سبب التحويل: آلام بطنية مزمنة، الأولوية: متوسطة

الفصل الثاني: الدراسة المسبقة

I تقديم الدراسة

1 التعريف بالمؤسسة

أنشأت المؤسسة العمومية للصحة الجوارية منذ تاريخ يناير 2008، بحيث تتكون المؤسسة من مجموعة من العيادات المتعددة الخدمات وقاعات العلاج، أما المؤسسة العمومية للصحة الجوارية المسيلة فقد أنشئت بموجب المرسوم التنفيذي رقم 07/140 المؤرخ في 19 ماي 2007 المتضمن إنشاء المؤسسات العمومية الاستشفائية والمؤسسات العمومية للصحة الجوارية[30].

2 بطاقة تقنية للمؤسسة

ا.2 الوضعية والتركيب

تقع المؤسسة العمومية للصحة الجوارية المسيلة شمال الولاية، حيث تغطي مساحة جغرافية تقدر بـ 2389 كم^2 أي بمعدل 25% من مساحة الولاية.كما تغطي المؤسسة عدد سكان يقدر بـ359372 بمعدل 150 ساكن على كم^2.

ب.2 التغطية الإدارية

المؤسسة العمومية للصحة الجوارية بالمسيلة تغطي 08 بلديات مقسمة على 04 دوائر:

- دائرة المسيلة: بلدية المسيلة.
- دائرة أولاد دراج: وتتضمن كل من بلدية المطارفة، بلدية السوامع.
- دائرة حمام الضلعة: وتتضمن كل من بلدية حمام الضلعة، بلدية تارمونت وبلدية أولاد منصور.
- دائرة الشلال: تتضمن كل من بلدية الشلال وبلدية أولاد ماضي.

2.ج التنظيم الصحي للمؤسسة

تتكون المؤسسة العمومية للصحة الجوارية بالمسيلة من عدة مصالح وهي كالتالي:

- 11 عيادة متعددة خدمات.
- 44 قاعة علاج 12 مصلحة طب المدرسي.
- مصلحة الأوبئة والوقاية SEMEP.
- مصلحة مكافحة السل UCTMR.
- مصلحة طب العمل.

2.د التغطية السكانية للمؤسسة حسب كل بلدية

جدول 2: التغطية السكانية للمؤسسة حسب كل بلدية

الدائرة	البلدية	تعداد السكان	المجموع
المسيلة	المسيلة	250144	250144
حمام الضلعة	حمام الضلعة	49524	70064
	تارمونت	12320	
	أولاد منصور	8220	
أولاد دراج	المطارفة	11732	21846
	السوامع	10114	
الشلال	الشلال	6961	17318
	أولاد ماضي	10357	
المجموع			359372

3 مهام المؤسسة

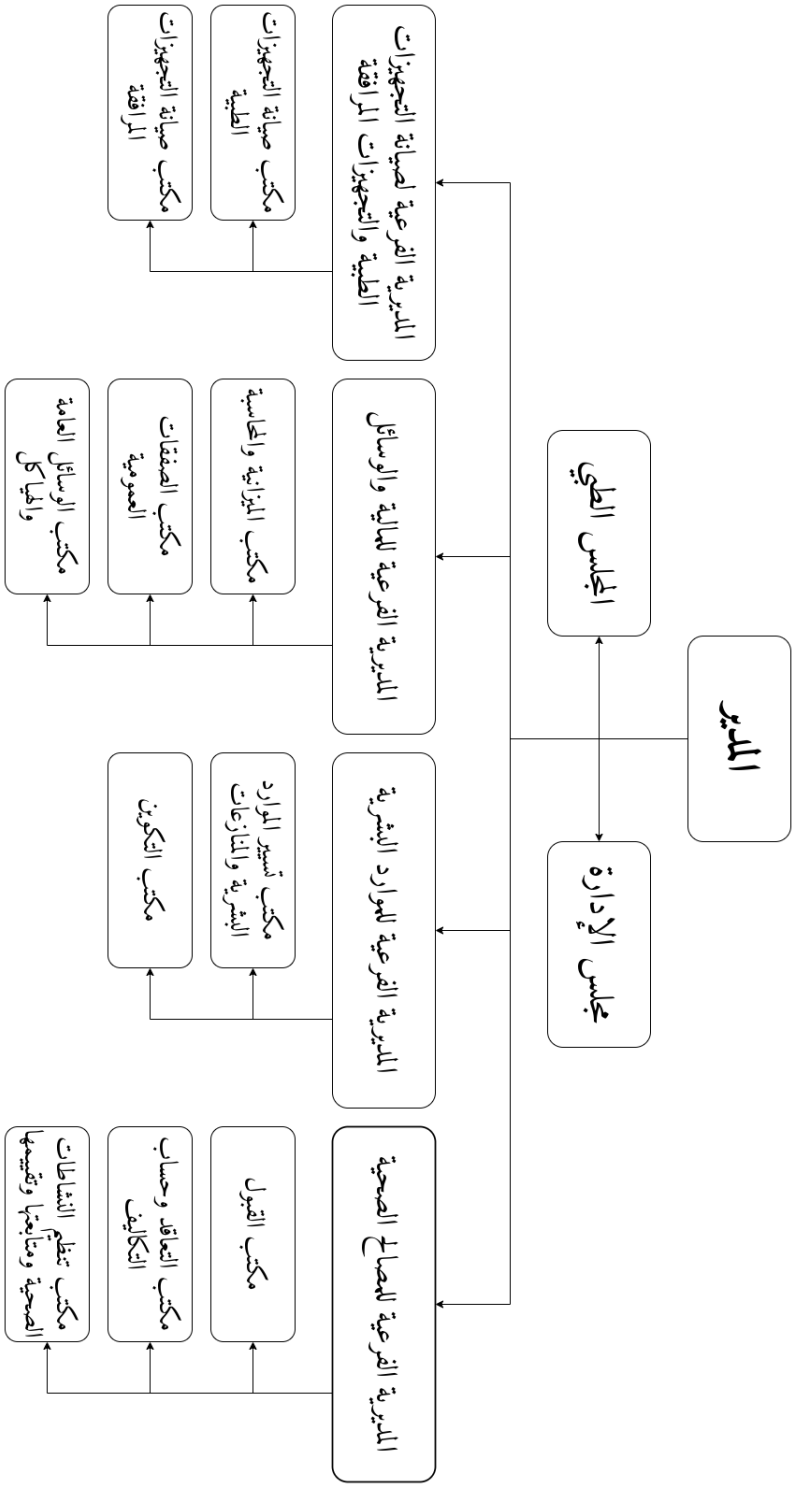
تتمثل مهام المؤسسة العمومية للصحة الجوارية في التكفل بصفة متكاملة ومتسلسلة فيما يأتي:

- الوقاية والعلاج القاعدي.
- تشخيص المرض.
- العلاج الجواري.
- الفحوص الخاصة بالطب العام والطب المتخصص القاعدي.
- الأنشطة المرتبطة بالصحة الإنجابية والتخطيط العائلي تنفيذ البرامج الوطنية للصحة والسكان.

وتكلف على الخصوص بما يأتي:

- المساهمة في ترقية وحماية البيئة في المجالات المرتبطة بحفظ الصحة والنقاوة ومكافحة الأضرار والآفات الاجتماعية.
- المساهمة في تحسين مستوى مستخدمي مصالح الصحة وتجديد معارفهم كما يمكن استخدام المؤسسة العمومية للصحة الجوارية ميدان التكوين شبه الطبي والتكوين في التسيير الاستشفائي على أساس اتفاقيات تبرم مع مؤسسات التكوين.

4 الهيكل التنظيمي للمؤسسة



شكل 5: الهيكل التنظيمي للمؤسسة

II دراسة الموجود

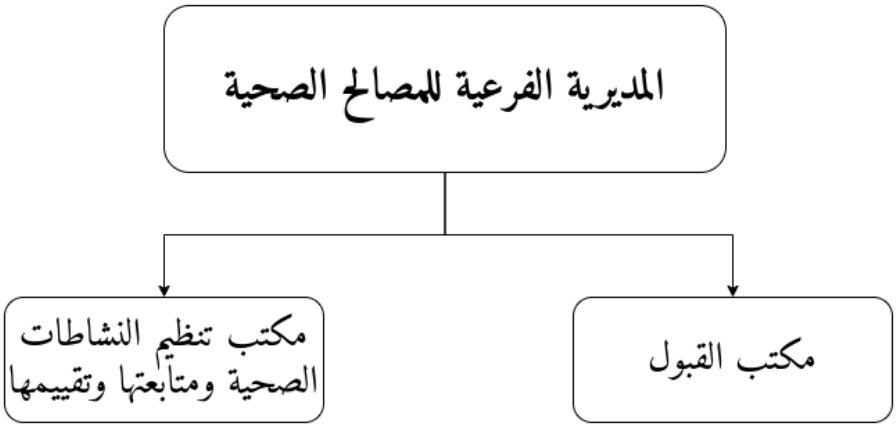
1 تعريف دراسة الموجود

هي الدراسة الشاملة والتحليلية لحقل الدراسة، تعتمد على جملة من الإجراءات مثل تحليل للوثائق المستعملة غاية إيجاد حلول واضحة وفعالة.[31]

2 تحديد مجال الدراسة

يتمثل مجال الدراسة في المديرية الفرعية للمصالح الصحية.

3 الهيكل التنظيمي لمجال الدراسة



شكل 6: الهيكل التنظيمي لمجال الدراسة

4 دراسة مناصب العمل

1.4 منصب رئيس مكتب تنظيم النشاطات الصحية ومتابعتها وتقييمها

بطاقة تحليلية لمنصب العمل

رمز المنصب: sys_admin

تعيين المنصب: رئيس مكتب تنظيم النشاطات الصحية ومتابعتها وتقييمها

رتبة المنصب: مساعد مهندس في الإعلام الآلي

تابع لـ: المديرية الفرعية للمصالح الصحية

عدد العمال: 01

الوسائل المستعملة: جهازي حاسوب (02)، طابعة، هاتف

المهام الموكلة:

- تحرير وتتبع مختلف الإحصائيات والتبليغ عنها.
- صيانة النظام وتقديم الدعم الفني اللازم.
- مسؤول عن تحرير كل الوثائق المستخدمة.

1 الوثائق المنجزة

جدول 3: منصب العمل الأول - الوثائق المنجزة

الرقم	تعيين الوثيقة	عدد النسخ	التواتر
02	تعداد الأطباء في المرفق الخاص	3	عشوائي
04	جدول العيادات	3	عشوائي
05	قائمة التخصصات	3	عشوائي
07	قائمة اسمية للأطباء على مستوى العيادة + مساعد طبيب	3	عشوائي
08	تفاصيل الموارد البشرية للمرافق	3	عشوائي

ب الوثائق الصادرة

جدول 4: منصب العمل الأول - الوثائق الصادرة

الرقم	تعيين الوثيقة	الوجهة	التواتر
02	تعداد الأطباء في المرفق الخاص	مكتب تسيير الموارد البشرية والمنازعات	عشوائي
04	جدول العيادات	مكتب الوسائل العامة والهياكل	عشوائي
05	قائمة التخصصات	مكتب الوسائل العامة والهياكل	عشوائي
07	قائمة اسمية للأطباء على مستوى العيادة + مساعد طبيب	مكتب تسيير الموارد البشرية والمنازعات	عشوائي

ج الوثائق الواردة

جدول 5: منصب العمل الأول - الوثائق الواردة

الرقم	تعيين الوثيقة	المصدر	التواتر
02	تفاصيل الموارد البشرية للمرافق	مكتب تسيير الموارد البشرية والمنازعات	عشوائي

4.ب منصب رئيس مكتب القبول

بطاقة تحليلية لمنصب العمل

رمز المنصب: sys_mod

تعيين المنصب: رئيس مكتب القبول

رتبة المنصب: تقني سامي في الإعلام الآلي

تابع لـ: المديرية الفرعية للمصالح الصحية

عدد العمال: 01

الوسائل المستعملة: جهاز حاسوب، طابعة، هاتف

المهام الموكلة:

- فهرسة الموظفين الجدد.
- المصادقة على عمليات التحويل.

ا الوثائق المنجزة

جدول 6: منصب العمل الثاني - الوثائق المنجزة

الرقم	تعيين الوثيقة	عدد النسخ	التواتر
01	بطاقة موفر	2	عشوائي
02	بطاقة مستفيد	2	عشوائي
03	توزيع الأطباء على التخصصات	3	عشوائي
04	تعداد الأطباء في المرفق الخاص	3	عشوائي

ب الوثائق الصادرة

جدول 7: منصب العمل الثاني - الوثائق الصادرة

الرقم	تعيين الوثيقة	الوجهة	التواتر
01	بطاقة موفر	موفر	عشوائي
02	بطاقة مستفيد	مستفيد	عشوائي
03	توزيع الأطباء على التخصصات	مكتب تسيير الموارد البشرية والمنازعات	عشوائي
04	تعداد الأطباء في المرفق الخاص	مكتب تنظيم النشاطات الصحية ومتابعتها وتقييمها	عشوائي

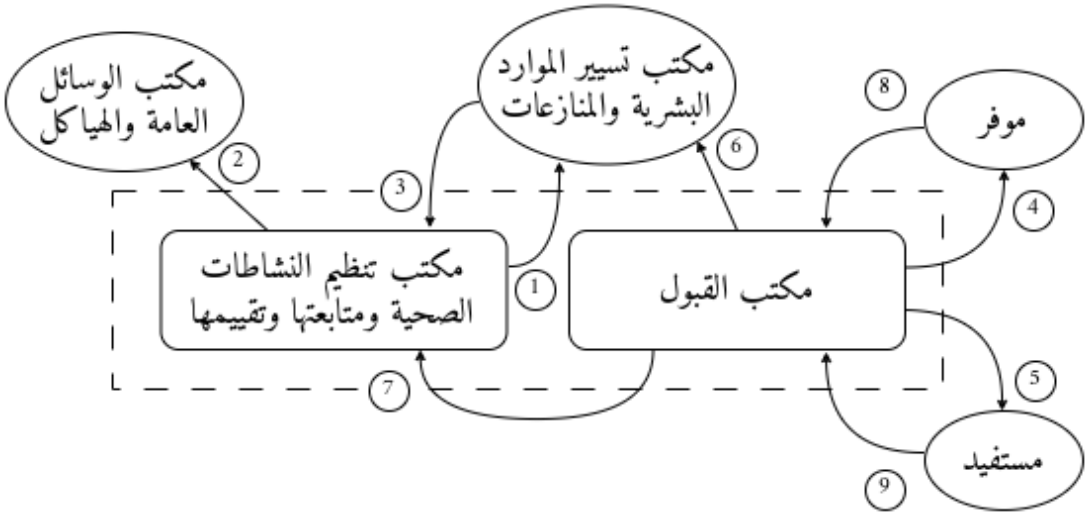
ج الوثائق الواردة

جدول 8: منصب العمل الثاني - الوثائق الواردة

الرقم	تعيين الوثيقة	المصدر	التواتر
01	بطاقة التعريف الوطني	موفر، مستفيد	عشوائي
02	وصفة طبية	موفر	عشوائي

5 دراسة تدفق المعلومات

1.5 مخطط تدفق المعلومات



شكل 7: مخطط تدفق المعلومات

2.5 ب وصف التدفق

شكل 8: وصف التدفق

رقم التدفق	التدفق
01	تعداد الأطباء في المرفق الخاص، قائمة اسمية للأطباء على مستوى العيادة + مساعد طبيب
02	جدول العيادات، قائمة التخصصات
03	تفاصيل الموارد البشرية للمرافق
04	بطاقة موفر
05	بطاقة مستفيد
06	توزيع الأطباء على التخصصات
07	تعداد الأطباء في المرفق الخاص
08	بطاقة التعريف الوطني، وصفة طبية
09	بطاقة التعريف الوطني

6 دراسة الوثائق

1.6 قائمة الوثائق المدروسة

جدول 9: قائمة الوثائق المدروسة

الرقم	تعيين الوثيقة	الرمز
01	القائمة الاسمية لعمال العيادة	D01
02	توزيع الأطباء على التخصصات	D02
03	جدول العيادات	D03
04	قائمة التخصصات	D04
05	قائمة اسمية للأطباء على مستوى العيادة + مساعد طبيب	D05
06	وصفة طبية	D06
08	تفاصيل الموارد البشرية للمرافق	D08
09	تعداد الأطباء في المرفق الخاص	D09
10	بطاقة موفر	D10
11	بطاقة مستفيد	D11

بطاقة تحليلية للوثيقة رقم: 01

رمز الوثيقة: D01

اسم الوثيقة: القائمة الاسمية لعمال العيادة

اللغة: عربية

المصدر: المديرية الفرعية للمصالح الصحية

الوجهة: مكتب تسيير الموارد البشرية والمنازعات

دور الوثيقة: إحصائيات

عدد النسخ: 3

الطبيعة: داخلية

الظهر: لا

الشكل: A4

اللون: أبيض

جدول 10: بطاقة تحليلية لوثيقة القائمة الاسمية لعمال العيادة

رمز المعلومة	المعلومة	النوع	الطول	الاستعمال	الملاحظة
clinic	العيادة	حرفي رقمي	64	نعم	
doc_num	رقم الوثيقة	حرفي رقمي	8	نعم	###/YYYY
emp_id	الرقم	رقمي	4	نعم	
emp_fullname	الاسم واللقب	حرفي	32	نعم	اللقب الاسم
emp_degree	الرتبة	حرفي رقمي	16	نعم	
emp_job	الوظيفة	حرفي	32	نعم	
emp_shift	توقيت العمل	حرفي رقمي	11	نعم	hh:mm-hh:mm
notes	الملاحظات	حرفي رقمي	32	نعم	#/# أو {أيام الراحة}

بطاقة تحليلية للوثيقة رقم: 02

رمز الوثيقة: D02

اسم الوثيقة: توزيع الأطباء على التخصصات

اللغة: عربية

المصدر: المديرية الفرعية للمصالح الصحية

الوجهة: مكتب تسيير الموارد البشرية والمنازعات

دور الوثيقة: إحصائيات

عدد النسخ: 3

الطبيعة: داخلية

الظهر: لا

الشكل: A4

اللون: أبيض

جدول 11: بطاقة تحليلية لوثيقة توزيع الأطباء على التخصصات

رمز المعلومة	المعلومة	النوع	الطول	الاستعمال	الملاحظة
doc_reg_num	رقم تسجيل الطبيب	رقمي	8	نعم	
doc_fullname	اسم ولقب الطبيب	حرفي	16	نعم	اللقب الاسم
doc_spec	تخصص الطبيب	حرفي	16	نعم	

بطاقة تحليلية للوثيقة رقم: 03

رمز الوثيقة: D03

اسم الوثيقة: جدول العيادات

اللغة: عربية

المصدر: المديرية الفرعية للمصالح الصحية

الوجهة: مكتب الوسائل العامة والهياكل

دور الوثيقة: إحصائيات

عدد النسخ: 3

الطبيعة: داخلية

الظهر: لا

الشكل: A4

اللون: أبيض

جدول 12: بطاقة تحليلية لوثيقة جدول العيادات

رمز المعلومة	المعلومة	النوع	الطول	الاستعمال	الملاحظة
clinic_num	رقم العيادة	رقمي	8	نعم	
clinic_name	اسم العيادة	حرفي رقمي	32	نعم	
clinic_address	عنوان العيادة	حرفي رقمي	32	نعم	
clinic_phone	رقم هاتف العيادة	رقمي	16	نعم	
clinic_email	عنوان بريد العيادة	حرفي رقمي	16	نعم	
clinic_spec	تخصص العيادة	حرفي	32	نعم	

بطاقة تحليلية للوثيقة رقم: 04

رمز الوثيقة: D04

اسم الوثيقة: قائمة التخصصات

اللغة: عربية

المصدر: المديرية الفرعية للمصالح الصحية

الوجهة: مكتب الوسائل العامة والهياكل

دور الوثيقة: إحصائيات

عدد النسخ: 3

الطبيعة: داخلية

الظهر: لا

الشكل: A4

اللون: أبيض

جدول 13: بطاقة تحليلية لوثيقة قائمة التخصصات

رمز المعلومة	المعلومة	النوع	الطول	الاستعمال	الملاحظة
spec_num	رقم التخصص	حرفي رقمي	8	نعم	
spec_name	اسم التخصص	حرفي	16	نعم	

بطاقة تحليلية للوثيقة رقم: 05

رمز الوثيقة: D05

اسم الوثيقة: قائمة اسمية للأطباء على مستوى العيادة + مساعد طبيب

اللغة: عربية

المصدر: المديرية الفرعية للمصالح الصحية

الوجهة: مكتب تسيير الموارد البشرية والمنازعات

دور الوثيقة: إحصائيات

عدد النسخ: 3

الطبيعة: داخلية

الظهر: لا

الشكل: A4

اللون: أبيض

جدول 14: قائمة اسمية للأطباء على مستوى العيادة + مساعد طبيب

رمز المعلومة	المعلومة	النوع	الطول	الاستعمال	الملاحظة
doc_register_num	رقم تسجيل الطبيب	رقمي	8	نعم	
doctor_first_name	اسم الطبيب	حرفي	16	نعم	
doctor_family_name	لقب الطبيب	حرفي	16	نعم	
address	عنوان	حرفي رقمي	32	نعم	
doctor_phone_number	رقم هاتف الطبيب	رقمي	16	نعم	
doctor_email_address	عنوان بريد الطبيب	حرفي رقمي	32	نعم	
doctor_specialty	تخصص الطبيب	حرفي	32	نعم	
clinic_id	رقم العيادة	رقمي	8	نعم	
clinic_address	عنوان العيادة	حرفي رقمي	32	نعم	

بطاقة تحليلية للوثيقة رقم: 06

رمز الوثيقة: D06

اسم الوثيقة: وصفة طبية

اللغة: فرنسية

المصدر: الموفر

الوجهة: المستفيد، المديرية الفرعية للمصالح الصحية

دور الوثيقة: إحصائيات

عدد النسخ: 3

الطبيعة: خارجية

الظهر: لا

الشكل: A4

اللون: أبيض

جدول 15: وصفة طبية

رمز المعلومة	المعلومة	النوع	الطول	الاستعمال	الملاحظة
patient_family_name	لقب المريض	حرفي	16	نعم	
patient_first_name	اسم المريض	حرفي	16	نعم	
patient_age	عمر المريض	رقمي	3	نعم	
prescription_date	تاريخ الوصفة	حرفي رقمي	10	نعم	YYYY/MM/DD
prescription_num	رقم الوصفة	رقمي	6	نعم	
doc_phone_number	رقم الهاتف	رقمي	10	نعم	
details	حيثيات	حرفي رقمي	128	نعم	

بطاقة تحليلية للوثيقة رقم: 07

رمز الوثيقة: D07

اسم الوثيقة: تفاصيل الموارد البشرية للمرافق

اللغة: عربية

المصدر: مكتب تسيير الموارد البشرية والمنازعات

الوجهة: المديرية الفرعية للمصالح الصحية

دور الوثيقة: إحصائيات

عدد النسخ: 3

الطبيعة: داخلية

الظهر: لا

الشكل: A4

اللون: أبيض

جدول 16: تفاصيل الموارد البشرية للمرافق

رمز المعلومة	المعلومة	النوع	الطول	الاستعمال	الملاحظة
commune	البلدية	حرفي	16	نعم	
structure	بنية	حرفي رقمي	64	نعم	
gp_count	طبيب عام	رقمي	2	نعم	
dentist_count	طبيب أسنان	رقمي	2	نعم	
pm_midwife_count	مساعدة طبية - قابلة	رقمي	2	نعم	
pm_general_health_count	مساعد طبي - مراعي الصحة العامة	رقمي	2	نعم	
pm_laboratory_count	مساعد طبي - مخبري	رقمي	2	نعم	
pm_radiology_count	مساعد طبي - فني أشعة	رقمي	2	نعم	
pm_nursing_assistants_count	مساعد ممرض	رقمي	2	نعم	

بطاقة تحليلية للوثيقة رقم: 08

رمز الوثيقة: D08

اسم الوثيقة: تعداد الأطباء في المرفق الخاصة

اللغة: عربية

المصدر: المديرية الفرعية للمصالح الصحية

الوجهة: مكتب تسيير الموارد البشرية والمنازعات

دور الوثيقة: إحصائيات

عدد النسخ: 3

الطبيعة: داخلية

الظهر: لا

الشكل: A4

اللون: أبيض

جدول 17: تعداد الأطباء في المرفق الخاص

رمز المعلومة	المعلومة	النوع	الطول	الاستعمال	الملاحظة
commune	البلدية	حرفي	16	نعم	
specialist_doctor_count	طبيب مختص	رقمي	2	نعم	
gp_count	طبيب عام	رقمي	2	نعم	
dentist_count	طبيب أسنان	رقمي	2	نعم	
pharmacists_count	صيدلي	رقمي	2	نعم	

بطاقة تحليلية للوثيقة رقم: 09

رمز الوثيقة: D09

اسم الوثيقة: بطاقة موفر

اللغة: عربية

المصدر: المديرية الفرعية للمصالح الصحية

الوجهة: موفر

دور الوثيقة: إثبات هوية

عدد النسخ: 3

الطبيعة: داخلية

الظهر: لا

الشكل: A4

اللون: أبيض

جدول 18: تعداد الأطباء في المرفق الخاص

رمز المعلومة	المعلومة	النوع	الطول	الاستعمال	الملاحظة
provider_uuid	UUID الموفر	حرفي رقمي	32	نعم	
provider_identifier	رمز الموفر	حرفي رقمي	16	نعم	
provider_full_name	اسم الموفر الكامل	حرفي	32	نعم	
provider_specialty	تخصص الموفر	حرفي	32	نعم	
provider_age	عمر الموفر	رقمي	2	نعم	
provider_location	مقر الموفر	حرفي رقمي	64	نعم	

بطاقة تحليلية للوثيقة رقم: 10

رمز الوثيقة: D10

اسم الوثيقة: بطاقة مستفيد

اللغة: عربية

المصدر: المديرية الفرعية للمصالح الصحية

الوجهة: مستفيد

دور الوثيقة: إثبات هوية

عدد النسخ: 3

الطبيعة: داخلية

الظهر: لا

الشكل: A4

اللون: أبيض

جدول 19: تعداد الأطباء في المرفق الخاص

رمز المعلومة	المعلومة	النوع	الطول	الاستعمال	الملاحظة
patient_uuid	UUID المستفيد	حرفي رقمي	32	نعم	
patient_full_name	اسم المستفيد الكامل	حرفي	32	نعم	
patient_age	عمر المستفيد	رقمي	3	نعم	
patient_residence	مقر إقامة المستفيد	حرفي رقمي	64	نعم	

III قائمة وصف إجراءات العمل

1 قائمة إجراءات العمل

جدول 20: قائمة إجراءات العمل

رقم الإجراء	اسم الإجراء
01	تسجيل المريض في المنصة
02	تسجيل طبيب، عيادة، تخصصات طبية
03	تسجيل الزيارة الطبية
04	توثيق التشخيص والعلاج
05	طلب وإدارة الفحوصات المخبرية
06	متابعة الحالات المزمنة
07	إدارة السجلات الطبية

2 جدول وصف إجراءات العمل

جدول 21: وصف إجراءات العمل

الرقم	الوصف
01	يقوم الموفر بالدخول إلى المنصة الإلكترونية وإنشاء حساب جديد للمريض عن طريق تعبئة نموذج التسجيل الذي يتضمن المعلومات الشخصية (الاسم، تاريخ الميلاد، العنوان...). بعد إدخال البيانات المطلوبة، يتم إنشاء مِلَفّ طبي إلكتروني فريد للمريض متضمناً رمز UUID خاص به يستخدم في جميع المعاملات الطبية اللاحقة.
02	يقوم مسؤول النظام بتسجيل الأطباء في المنصة بإدخال المعلومات الشخصية والمهنية (الاسم، التخصص...). كما يتم إضافة العيادات بإدخال المعلومات (اسم العيادة، العنوان، أرقام الاتصال، ساعات العمل). يتم ربط الأطباء بالعيادات والتخصصات المختلفة مع تحديد الأيام والأوقات المتاحة لكل طبيب. تُرفع الوثائق والشهادات المطلوبة إلكترونيا للمراجعة والتحقق، ويتم تفعيل الحساب بعد الموافقة الإدارية.
03	يقوم موظف الاستقبال بتسجيل زيارة المريض عند وصوله للعيادة من خلال البحث عن بيانات المريض باستخدام رقم الهوية أو UUID. يتم تحديد سبب الزيارة، مستوى الأولوية، والطبيب المعالج. يُسجل وقت الوصول ويتم إضافة المريض لقائمة الانتظار الإلكترونية. يتلقى المريض إشعاراً بالوقت التقريبي للمقابلة الطبية.
04	يقوم الطبيب بالدخول إلى ملف المريض الإلكتروني وتسجيل الأعراض والعلامات السريرية والتشخيص المبدئي. يتم توثيق خطة العلاج والأدوية الموصوفة مع الجرعات والتعليمات. يمكن للطبيب الوصول للتاريخ الطبي السابق، نتائج الفحوصات المخبرية والأشعة.
05	يقوم الطبيب بإنشاء طلبات الفحوصات المخبرية إلكترونيًا من خلال قائمة الفحوصات المتاحة. يتم طباعة نموذج الطلب وتحويل المريض للمختبر. يقوم فني المختبر بتسجيل استلام العينات وإدخال النتائج في النظام. يتم إشعار الطبيب المعالج فور اكتمال النتائج. يمكن مقارنة النتائج الحالية مع النتائج السابقة وعرضها بيانياً.
06	يوفر النظام خطط متابعة آلية للمرضى المزمنين (السكري، ضغط الدم، القلب). يتم جدولة فحوصات دورية وتذكير المرضى بها. يعرض النظام تطور المؤشرات الصحية للمريض بيانًا عبر الزمن (مستوى السكر، ضغط الدم). يتم إنشاء تنبيهات للطبيب في حال عدم التزام المريض بالمتابعة الدورية أو تدهور الحالة.
07	يوفر النظام آلية لحفظ وأرشفة الملفات الطبية وفق المعايير القانونية. يتم تصنيف الوثائق الطبية وفهرستها بطريقة تسهل استرجاعها. يوفر النظام سجل تدقيق يحفظ تاريخ الوصول والتعديلات على الملفات الطبية.

IV النقد والاقتراحات

1 النقد

بعد إجراء هذه الدراسة تبينت بعض النقائص التي تعرقل سير النظام وتهدد أمانه، منها:

- الحاجة إلى تكرار المهام بأكثر من برنامَج/موقع/منصة لغياب المركزية في عملية استقبال البيانات.
- الاعتماد على برامج وأنظمة تشغيل مدفوعة غالبا ما تفعل باستخدام Cracks تجازف بأمان واستقرار الأجهزة.
- نظام الـEMR المعتمد بمجال الدراسة حاليًا (نظام DEMdz) يعاني من عيوب كثيرة منها عملية التثبيت المعقدة بشكل غير مبرر والحاجة إلى استعمال محاكي عن طريق VMware ما يؤثر على موارد الخادم. هذا وغياب ترخيصات SSL/TLS ما يجعل النظام مرحبا لمختلف الهجمات السيبرانية (هجوم MITM[1] مثلا).

2 الاقتراحات

- إنجاز قاعدة معطيات مركزية يمكن لمختلف الأجهزة الوصول إليها عبر API آمن، مرن وقابل للتوسيع.
- توظيف أبدال حرة ومفتوحة المصدر لمختلف الاحتياجات البرمجية للنظام.
- استغلال تقنية الحاويات في توزيع النظام لضمان سلاسة في عملية التثبيت وسهولة الصيانة.
- ترقية وترخيص جميع الاتصالات لتكون عبر HTTPS.

1 اعتراض غير مصرح به للاتصال بين طرفين، يصبح أكثر خطورة عند غياب بروتوكولات التشفير التي تحمي البيانات المتبادلة.

الفصل الثالث: الدراسة التصميمية

I قاموس المعطيات

لائحة تحتوي على المعلومات الموجودة في الوثائق والملفات المدروسة، ويتم تحديد طبيعة وطول وعلاقات الحساب والإدماج لكل معلومة[31].

1 قاموس المعطيات الصافي

رمز المعلومة	المعلومة	النوع	الطول	الطبيعة		الملاحظة
				E/CO/CA	SiG/SiT/M	
person.person_id	معرف الشخص	N	10	E	SiG	
person.gender	جنس الشخص	A	1	E	SiG	”M”/“F“
person.birth_date	تاريخ ميلاد الشخص	AN	10	E	SiG	YYYY-MM-DD
person.birth_time	وقت ميلاد الشخص	AN	5	E	SiG	hh-mm
person.birth_date_estimated	تاريخ ميلاد الشخص تقريبي	B	1	E	SiG	
person.dead	الشخص متوفي	B	1	E	SiT	
person.death_date	تاريخ وفاة الشخص	AN	10	E	SiG	YYYY-MM-DD
person.death_time	وقت وفاة الشخص	AN	5	E	SiG	hh-mm
person.death_date_estimated	تاريخ وفاة الشخص تقريبي	B	1	E	SiG	
person.cause_of_death	سبب وفاة الشخص	N	10	E	SiG	
person.cause_of_death_non_coded	سبب وفاة الشخص الغير مدرج	AN	256	E	SiG	
person_name.person_name_id	معرف اسم الشخص	N	10	E	SiG	
person_name.preferred	اسم الشخص مفضل	B	1	E	SiT	
person_name.degree	درجة اسم الشخص	AN	64	E	M	
person_name.given_name	اسم الشخص الأول	A	256	E	SiG	
person_name.middle_name	اسم الشخص الثاني	A	256	E	SiG	
person_name.family_name	لقب الشخص	A	256	E	SiG	
person_name.given_name_ar	اسم الشخص الأول بالعربي	A	256	E	SiG	
person_name.middle_name_ar	اسم الشخص الثاني بالعربي	A	256	E	SiG	
person_name.family_name_ar	لقب الشخص بالعربي	A	256	E	SiG	
person_address.person_address_id	معرف عنوان الشخص	N	10	E	SiG	

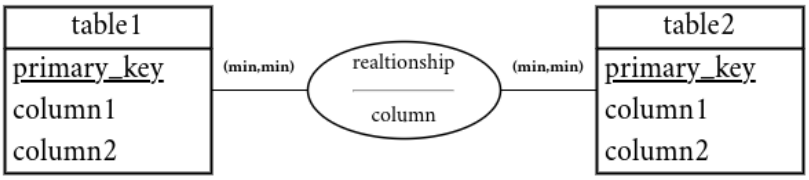
	SiT	E	1	B	عنوان الشخص مفضل	person_address.preferred
	SiG	E	256	AN	عنوان الشخص عنوان 1	person_address.address1
	SiG	E	256	AN	عنوان الشخص عنوان 2	person_address.address2
	SiG	E	128	A	عنوان الشخص مدينة	person_address.city_village
	SiT	E	2	N	عنوان الشخص ولاية	person_address.state_province
	SiG	E	64	N	عنوان الشخص الرمز البريدي	person_address.postal_code
	SiG	E	256	A	عنوان الشخص البلد	person_address.country
	SiG	E	10	N	معرف المريض	patient.patient_id
	SiG	E	10	N	معرف رمز المريض	patient_identifier.identifier_id
	SiG	E	256	AN	رمز المريض	patient_identifier.identifier
	SiG	E	1	B	رمز المريض مفضل	patient_identifier.preferred
	SiG	E	10	N	معرف نوع رمز المريض	patient_identifier_type.identifier_type_id
	SiG	E	128	AN	اسم نوع رمز المريض	patient_identifier_type.name
	SiG	E	/	AN	وصف نوع رمز المريض	patient_identifier_type.description
	SiG	E	256	AN	تنسيق نوع رمز المريض	patient_identifier_type.format
	SiG	E	1	B	رمز تحقق نوع رمز المريض	patient_identifier_type.check_digit
	SiG	E	10	N	معرف المفهوم	concept.concept_id
	SiG	E	128	AN	اسم المفهوم	concept.name
	SiG	E	/	AN	وصف المفهوم	concept.description
	SiG	E	1	B	المفهوم مدرج	concept.is_set
	SiG	E	10	N	معرف قسم المفهوم	concept_class.concept_class_id
	SiG	E	128	AN	اسم قسم المفهوم	concept_class.name
	SiG	E	/	AN	وصف قسم المفهوم	concept_class.description
	SiG	E	10	N	معرف نوع بيانات المفهوم	concept_datatype.concept_datatype_id
	SiG	E	128	AN	اسم نوع بيانات المفهوم	concept_datatype.name
	SiG	E	3	A	اختصار hl7 للمفهوم	concept_datatype.hl7_abbreviation
	SiG	E	/	AN	وصف نوع بيانات المفهوم	concept_datatype.description
	SiG	E	10	N	معرف الموفر	provider.provider_id
	SiG	E	256	AN	رمز الموفر	provider.identifier
	SiG	E	10	N	معرف الزيارة	visit.visit_id
YYYY-MM-DD	SiG	E	10	AN	تاريخ البدء	visit.date_started

YYYY-MM-DD	SiG	E	10	AN	تاريخ النهاية	visit.date_stopped
	SiG	E	10	N	معرف نوع الزيارة	visit_type.visit_type_id
	SiG	E	128	AN	اسم نوع الزيارة	visit_type.name
	SiG	E	/	AN	وصف نوع الزيارة	visit_type.description
	SiG	E	10	N	معرف موفر المقابلة	encounter_provider.encounter_provider_id
	SiG	E	10	N	معرف المقابلة	encounter.encounter_id
MM-DD hh:mm	SiG	E	10	AN	تاريخ المقابلة	encounter.encounter_datetime
	SiG	E	9	A	حالة المقابلة	encounter.status
	SiG	E	10	N	معرف نوع المقابلة	encounter_type.encounter_type_id
	SiG	E	128	AN	اسم نوع المقابلة	encounter_type.name
	SiG	E	/	AN	وصف نوع المقابلة	encounter_type.description
	SiG	E	10	N	معرف الملاحظة	obs.obs_id
MM-DD hh:mm	SiG	E	10	AN	تاريخ الملاحظة	obs.obs_datetime
	SiG	E	JSON	AN	قيمة الملاحظة	obs.value
	SiG	E	/	AN	تعليقات الملاحظة	obs.comments
	SiG	E	10	N	معرف الدواء	drug.drug_id
	SiG	E	128	AN	اسم الدواء	drug.name
	SiG	E	1	B	الدواء مركب	drug.combination
	SiG	E	10	N	وحدة جرعة الدواء	drug.dosage_form
	SiG	E	10	N	معرف الطلب	orders.order_id
	SiG	E	17	A	ضرورة الطلب	orders.urgency
	SiG	E	256	AN	رقم الطلب	orders.order_number
	SiG	E	11	A	فعل الطلب	orders.order_action
YYYY-MM-DD	SiG	E	10	AN	تاريخ تفعيل الطلب	orders.date_activated
YYYY-MM-DD	SiG	E	10	AN	تاريخ الانقضاء التلقائي للطلب	orders.auto_expire_date
YYYY-MM-DD	SiG	E	10	AN	تاريخ الانقضاء الطلب	orders.date_stopped
	SiG	E	10	N	معرف نوع الطلب	order_type.order_type_id
	SiG	E	128	AN	اسم الطلب	order_type.name
	SiG	E	/	AN	وصف الطلب	order_type.description
	SiG	E	10	N	معرف العنوان	location.location_id
	SiG	E	128	AN	اسم العنوان	location.name

	SiG	E	/	AN	وصف العنوان	location.description
	SiG	E	256	AN	عنوان 1	location.address1
	SiG	E	256	AN	عنوان 2	location.address2
	SiG	E	128	A	مدينة العنوان	location.city_village
	SiG	E	2	N	ولاية العنوان	location.state_province
	SiG	E	64	N	رمز العنوان البريدي	location.postal_code
	SiG	E	256	A	بلد العنوان	location.country
	SiG	E	10	N	معرف البرنامج	program.program_id
	SiG	E	128	AN	اسم البرنامج	program.name
	SiG	E	/	AN	وصف البرنامج	program.description
	SiG	E	10	N	معرف برنامج المريض	patient_program.patient_program_id
YYYY-MM-DD	SiG	E	10	AN	تاريخ البدء	patient_program.date_enrolled
YYYY-MM-DD	SiG	E	10	AN	تاريخ الإكمال	patient_program.date_completed

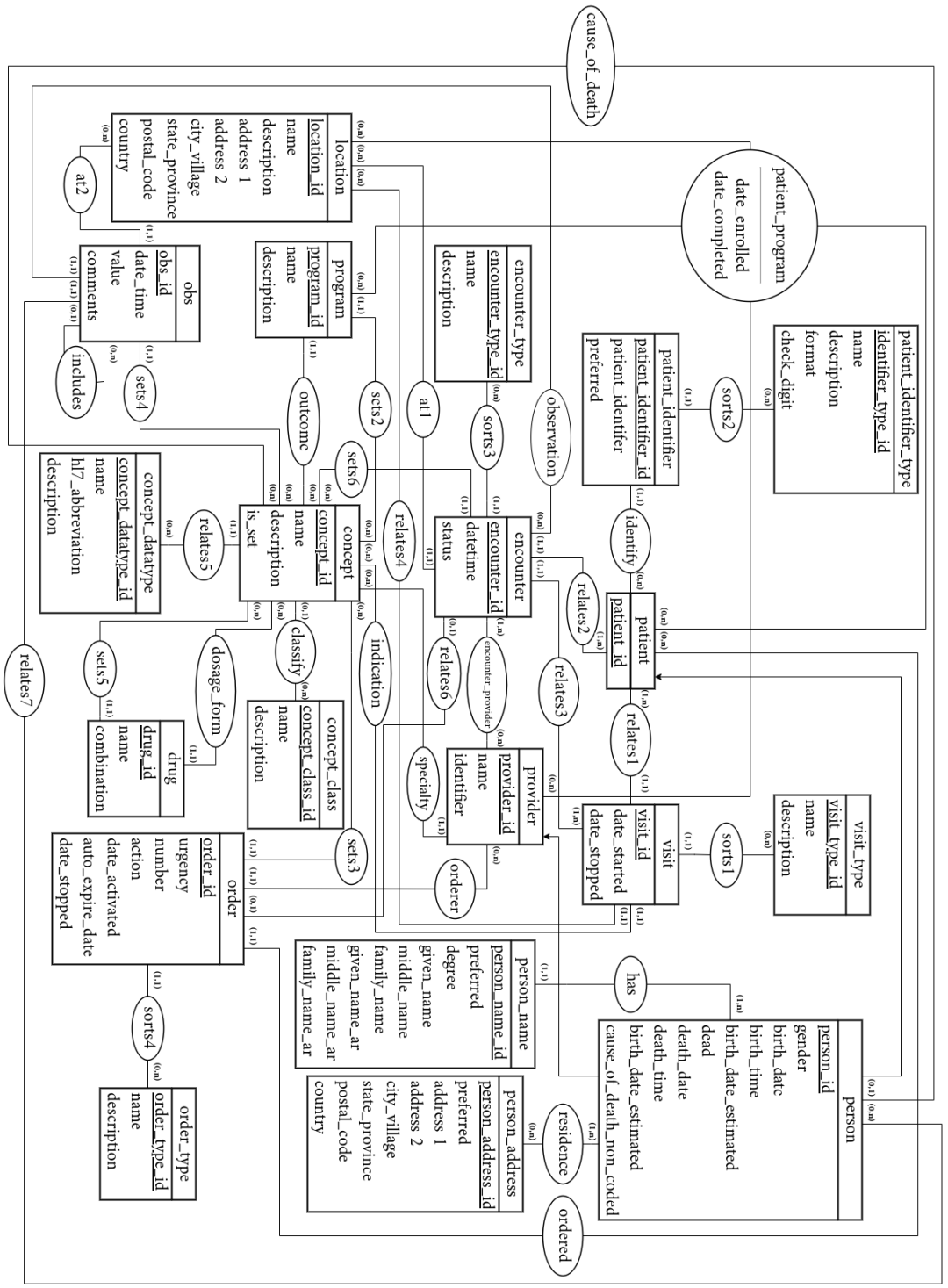
II النموذج التصوري للمعطيات

عرض شامل للمعطيات المتحصل عليها أثناء الدراسة في شكل أفراد تربط بينها علاقات مما يسمح بتنظيم النظام المعلوماتي.



شكل 9: مثال عن تصميم مخطط نموذج *MCD*

1 مخطط MCD



شكل 10: مخطط *MCD*

ا.1 جدول وصف الأفراد

جدول 22: وصف الأفراد

الفرد	المعرف	الخصائص	تعيين الخصائص
person	person_id	gender	جنس
		birth_date	تاريخ الميلاد
		birth_time	وقت الميلاد
		birth_date_estimated	تاريخ الميلاد تقريبي
		dead	متوفي
		death_date	تاريخ الوفاة
		death_time	وقت الوفاة
		death_date_estimated	تاريخ الوفاة تقريبي
		cause_of_death	سبب الوفاة
		cause_of_death_non_coded	سبب الوفاة الغير مدرج
person_name	person_name_id	preferred	مفضل
		degree	درجة
		given_name	اسم الأول
		middle_name	اسم الثاني
		family_name	لقب
		given_name_ar	اسم الأول بالعربي
		middle_name_ar	اسم الثاني بالعربي
		family_name_ar	لقب بالعربي
person_address	person_address_id	preferred	مفضل
		address1	عنوان 1
		address2	عنوان 2
		city_village	مدينة
		state_province	ولاية
		postal_code	الرمز البريدي
		country	البلد
patient	patient_id	/	/
patient_identifier	patient_identifier_id	identifier	رمز
		preferred	مفضل

إسم وصف تنسيق رمز التحقق	name description format check_digit	patient_identifier_type_id	patient_identifier_type
اسم وصف مدرج	name description is_set	concept_id	concept
اسم وصف	name description	concept_class_id	concept_class
اسم اختصار hl7 وصف	name hl7_abbreviation description	concept_datatype_id	concept_datatype
رمز	identifier	provider_id	provider
تاريخ البدء تاريخ النهاية	date_started date_stopped	visit_id	visit
اسم وصف	name description	visit_type_id	visit_type
/	/	encounter_provider_id	encounter_provider
تاريخ حالة	datetime status	encounter_id	encounter
اسم وصف	name description	encounter_type_id	encounter_type
تاريخ قيمة تعليقات	datetime value comment	obs_id	obs
اسم مركب وحدة الجرعة	name combination dosage_form	drug_id	drug

order	order_id	urgency ضرورة order_number رقم order_action فعل date_activated تاريخ التفعيل auto_expire_date تاريخ الانقضاء التلقائي date_stopped تاريخ الانقضاء	
order_type	order_type_id	name اسم description وصف	
location	location_id	name اسم description وصف address1 العنوان 1 address2 العنوان 2 city_village مدينة state_province ولاية postal_code رمز بريدي country بلد	
program	program_id	name اسم description وصف	

1.ب جدول وصف العلاقات

جدول 23: وصف العلاقات

العلاقة	تربط بين	الأصليات	الخصائص
relates1	patient visit	(1,n) (1,1)	/
relates2	patient encounter	(1,n) (1,1)	/
relates3	visit encounter	(1,n) (1,1)	/
relates4	visit location	(1,1) (0,n)	/
relates5	concept concept_datatype	(1,1) (0,n)	/

relates6	provider encounter	(0,1) (0,1)	/
relates7	person obs	(0,n) (1,1)	/
sets1	patient patient_identifier	(0,n) (1,1)	/
sets2	concept program	(0,n) (1,1)	/
sets3	order concept	(1,1) (0,n)	/
sets4	concept obs	(0,n) (1,1)	/
sets5	drug concept	(1,1) (0,n)	/
set6	encounter concept	(1,1) (0,n)	/
sorts1	visit visit_type	(1,1) (0,n)	/
sorts2	patient_identifier patient_identifier_type	(1,1) (0,n)	/
sorts3	encounter encounter_type	(1,1) (0,n)	/
sorts4	order order_type	(1,1) (0,n)	/
at1	encounter location	(1,1) (0,n)	/
at2	obs location	(1,1) (0,n)	/
residence	person person_address	(1,n) (0,n)	/

/	(1,1) (0,n)	provider concept	specialty
/	(0,n) (1,1)	encounter obs	observation
/	(1,n) (0,n)	person person_name	has
/	(0,1) (0,n)	concept concept_class	classify
/	(0,n) (1,1)	concept program	outcome
/	(0,n) (1,1)	obs obs	includes
/	(1,1) (0,n)	visit concept	indicator
/	(1,1) (0,n)	drug concept	dosage_form
/	(1,1) (0,n)	person concept	cause_of_death
/	(0,n) (1,n)	provider encounter	encounter_provider
/	(0,n) (1,1)	provider order	orderer
/	(0,n) (1,1)	patient order	ordered
date_enrolled date_completed	(0,n) (0,n) (0,n) (0,n)	provider patient program address	patient_program

III النموذج المنطقي للمعطيات

هو مجموعة من العلاقات يتم استنتاجها من النموذج التصوري للمعطيات، الهدف منه هو التعريف المنطقي للمعطيات[31].

1 نموذج MLD

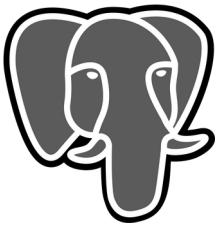
person(person_id, gender, birth_date, birth_time,
birth_date_estimated, dead, death_date, death_time,
death_date_estimated, cause_of_death*, cause_of_death_non_coded)
person_name(person_name_id, person_id*, preferred, degree, given_name,
middle_name, family_name, given_name_ar, middle_name_ar,
family_name_ar)
person_address(person_address_id, person_id*, preferred, address1,
address2, city_village, state_province, postal_code, country)
patient(patient_id*)
patient_identifier(identifier_id, identifier, preferred, patient_id*,
patient_identifier_type_id*)
patient_identifier_type(identifier_type_id, name, description, format,
check_digit)
concept(concept_id, name, description, is_set, concept_datatype_id*)
concept_class(concept_class_id, name, description)
concept_datatype(concept_datatype_id, name, hl7_abbreviation,
description)
provider(provider_id, person_id*, specialty, identifier)
visit(visit_id, date_started, date_stopped, visit_type_id*,
patient_id*, location_id*, indication_concept_id*)
visit_type(visit_type_id, name, description)
encounter_provider(encounter_provider_id, encounter_id*, provider_id*)
encounter(encounter_id, date_time, status, visit_id*, patient_id*,
encounter_type_id*, location_id*)
obs(obs_id, person_id*, value, comments, encounter_id*, location_id*,
obs_group_id*, concept_id*)
drug(drug_id, name, combination, dosage_form*, concept_id*)
order(order_id, urgency, order_number, order_action, date_activated,
auto_expire_date, date_stopped, patient_id*, orderer_id*,
encounter_id*, concept_id*)
order_type(order_type_id, name, description)
location(location_id, name, description, address1, address2,
city_village, state_province, postal_code, country)
program(program_id, name, description, concept_id*,
outcomes_concept_id*)
patient_program(patient_program_id, date_enrolled, date_completed,
patient_id*, program_id*, location_id*)

الفصل الرابع: الدراسة التقنية

I حزمة التطوير

أو Development Stack، وهو مجموعة الأدوات أو التقنيات المستعملة في الإنجاز البرمجي لمشروع معين، ويشمل المكاتب أو أطر العمل الأساسية التي استعملت في مرحلة التطوير. الـDevelopment Stack لهذا المشروع يشمل:

1 PostgreSQL



شكل 11: شعار *PostgreSQL*

نظام إدارة قواعد بيانات حر ومفتوح المصدر يدعم كلًا من قواعد البيانات العلائقية وغير العلائقية. يتميز بقدرته على معالجة مشاريع بمختلف الأحجام، من التطبيقات الصغيرة التي تعمل على جهاز واحد إلى تطبيقات المؤسسات الكبيرة ذات المعاملات المتزامنة العدّة ومخازن البيانات الضخمة. يتميز PostgreSQL بثلاث مزايا رئيسية تجعله خيارًا مثاليًا للمطورين:

- موثوقيه عالية مع ضمان تناسق البيانات مع معايير ACID[1]
- نظام قوي للأنواع المخصصة والامتدادات الإضافية
- مجتمع نشيط وتطوير مستمر مع دعم المعايير القياسية

2 Podman

شكل 12: شعار *podman*

أداة حرة ومفتوحة المصدر لإدارة الحاويات تعمل بدون الحاجة إلى امتيازات مستخدم root. يوفر Podman بديلًا آمنًا وفعالًا لـDocker مع الحفاظ على توافق واجهة سطر الأوامر. يسمح Podman بإنشاء وإدارة وتشغيل حاويات Linux والصور المرتبطة بها. و يتميز بثلاث مزايا رئيسية:

- نموذج أمان معزز بفضل العمل بدون خدمة daemon مركزية
- توافقية عالية مع أوامر Docker مما يسهل الانتقال بين الأداتين

1 أربع خصائص يجب توفرها في قواعد المعطيات، وهي الذرية، التناسق، العزل والدوام[32].

- القدرة على إدارة مجموعات الحاويات (pods) بطريقة مشابهة لـKubernetes

3 Axum & SQLx

3.ا Axum

إطار عمل مبني بِلُغَةِ Rust لتطوير خدمات الـWeb وواجهات برمجة التطبيقات. يتميز Axum بتركيزه على البنية النمطية والكفاءة العالية. تم تصميمه خصيصًا للاستفادة من نموذج البرمجة غير المتزامنة (asynchronous programming) في Rust باستخدام Tokio. يتميز Axum بثلاث مزايا رئيسية:

- أداء عالٍ مع استهلاك منخفض للموارد بفضل استخدام لغة Rust
- نظام توجيه مرن وسهل الاستخدام للطلبات

3.ب SQLx

مكتبة SQL لـRust تتيح التعامل مع قواعد البيانات بشكل آمن وفعال. تتميز بقدرتها على التحقق من صحة الاستعلامات SQL في وقت البناء دون الحاجة إلى تشغيل خادم قاعدة البيانات. تدعم SQLx عدة أنظمة لإدارة قواعد البيانات بما فيها PostgreSQL وMySQL وSQLite. أهم مميزات SQLx:

- تحقق من صحة الاستعلامات في وقت البناء مما يقلل من الأخطاء
- دعم العمليات غير المتزامنة (async/await) بشكل كامل
- واجهة برمجة بسيطة ومباشرة مع قواعد البيانات دون طبقات وسيطة معقدة

4 Tauri

شكل 13: شعار *Tauri*

إطار عمل حر ومفتوح المصدر لبناء تطبيقات صغيرة وسريعة تعمل على جميع المنصات الرئيسية للحواسيب والهواتف المحمولة. يمكن للمطورين دمج أي إطار عمل للواجهات الأمامية (frontend) قابل للتحويل إلى HTML وJavaScript وCSS لبناء تجرِبة المستخدم الخاصة بهم، مع الاستفادة من لغات البرمجة القوية مثل Rust وSwift وKotlin لتطوير المنطق البرمجي الخلفي (backend) عند الحاجة. يتميز Tauri بثلاث مزايا رئيسية يمكن للمطورين الاعتماد عليها:

- أساس آمن لبناء التطبيقات
- حجم حُزْمَة أصغر باستخدام الـnative webview الذي يوفره النظام
- مرونة تتيح للمطورين استخدام أي إطار عمل للواجهات الأمامية وروابط لمختلف اللغات الأخرى

5 Svelte & Tailwind

١.5 Svelte

شكل 14: شعار *Svelte*

إطار عمل حر ومفتوح المصدر JavaScript مبتكر لبناء واجهات المستخدم بنهج مختلف عن المكتبات التقليدية. يتميز Svelte بكونه إطار عمل تجميعي compile-time framework حيث يحول المكونات إلى شيفرة JavaScript مُحسنة أثناء مرحلة البناء بدلًا من تفسيرها في وقت التشغيل. هذا يؤدي إلى تطبيقات أصغر حجمًا وأسرع أداءً. مميزات Svelte الرئيسية:

- حجم حزمة صغير مقارنة بأطر العمل الأخرى مما يعزز سرعة التحميل
- أداء عالٍ جدًا بفضل تقليل العمليات الحسابية أثناء التشغيل
- بناء نحوي (syntax) بسيط وسلس يقلل من كمية الشفرة المطلوبة

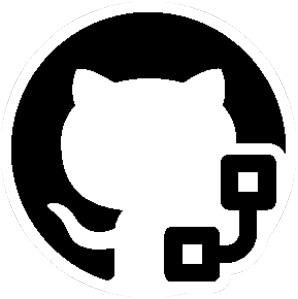
5.ب Tailwind

شكل 15: شعار *TailwindCSS*

إطار عمل CSS حر ومفتوح المصدر يعتمد على مفهوم الفئات المنفعية (utility classes). يوفر Tailwind مجموعة واسعة من الفئات الجاهزة التي يمكن تطبيقها مباشرة على عناصر HTML لتنسيقها، مما يقلل الحاجة إلى كتابة CSS مخصص. مميزات Tailwind CSS:

- سرعة في تطوير واجهات المستخدم دون مغادرة ملفات المكونات
- تخصيص عالي عبر ملف تكوين مركزي
- تكامل ممتاز مع Svelte مما يعزز تجربة وإنتاجية المطور
- حجم نهائي صغير للإنتاج بفضل إزالة الفئات غير المستخدمة

6 Github Actions



شكل 16: شعار *gh Actions*

نظام CI/CD مدمج مباشرة في منصة GitHub. يسمح بأتمتة سير أعمال مثل اختبار الشيفرة وبنائها ورفعها عند تحقق شروط محددة في مستودع GitHub. تتميز Github Actions بثلاث مزايا رئيسية:

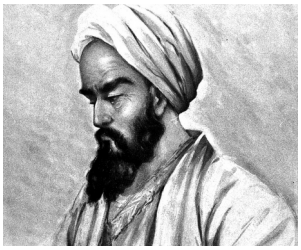
- تكامل سلس مع مستودعات GitHub
- نظام مرن لتعريف سير العمل باستخدام تنسيق YAML
- مكتبة واسعة من الإجراءات الجاهزة

الفصل الخامس: الإنجاز البرمجي

I منصة الرازي

منصة platform agnostic[1] حرة ومفتوحة المصدر توفر EMR متقدم مصمم لتبسيط إدارة المعلومات الطبية وتحسين رعاية المرضى. يأتي بواجهة سهلة الاستخدام تتيح للأطباء والممرضين والإداريين الوصول الآمن إلى سجلات المرضى وإدارتها بكفاءة.

1 أبو بكر محمد بن زكريا الرازي



سمية المنصة تيمنًا بأبو بكر محمد بن زكريا الرازي (854-925 م)، احد أعظم العقول العلمية في العصر الذهبي للحضارة الإسلامية. كان الرازي طبيبًا وكيميائيًا وفيلسوفًا، وقد ألف أكثر من 200 كتاب في مجالات متعددة، خاصة في الطب والصيدلة[33].

شكل 17: *Portrait of Rhazes*

2 المميزات الرئيسية

- **إدارة سجلات المرضى:** تخزين وتنظيم المعلومات الطبية للمرضى بطريقة آمنة وسهلة الوصول.
- **واجهة سهلة الاستخدام:** تصميم بسيط وبديهي يجعل العمل اليومي أكثر سلاسة.
- **حل متكامل:** يربط بين مختلف جوانب الرعاية الصحية في نظام واحد متكامل.
- **مفتوح المصدر:** برنامج حر ومفتوح المصدر متاح للجميع للاستخدام والمساهمة في التطوير.
- **معزز بالذكاء الاصطناعي:** يأتي مع LLM مدرب خصيصا على فهم مختلف اللهجات العربية المحلية.

3 الرؤية التطلعية

حاليًا، تقتصر منصة الرازي على تطبيق وحيد وهوRhazes EMR، لكن إحدى الأهداف الأساسية للمنصة هو توفير حل يغطي جميع جوانب الرعاية الصحية. وعليه، نخطط لتوسيع عائلة تطبيقات الرازي في المستقبل، مع إضافة حلول جديدة تلبي احتياجات مختلف جوانب الرعاية الصحية، مثل تطبيقات لحجز

1 يشير إلى البرمجيات أو التطبيقات التي تعمل على أنظمة تشغيل متعددة دون الحاجة إلى تعديلات كبيرة.

المواعيد الطبية online وأخرى لتتبع انتظام المرضى في تناول الأدوية الموصوفة. هدفنا هو بناء نظام بيئي متكامل يدعم المؤسسات الصحية من جميع الأحجام.

4 بنية المنصة

1.4 قاعدة المعطيات

يستخدم المشروع نظام تسيير قواعد المعطيات PostgreSQL، أما طريقة توزيعه فتكون عبر حاوية مكونة من قاعدة المعطيات إضافة إلى برنامَج migrations كتب باستخدام مكتبة SQLx. يقوم البرنامج بتصميم وهيكلة قاعدة المعطيات تلقائيًا فور تشغيل الحاوية.

2.4 واجهة برمجة التطبيقات

تم توظيف إطار العمل Axum لكتابة واجهة برمجة تطبيقات يمكن استغلالها لربط مختلف مكونات المنصة مع بعض، حيث يتم تثبيتها على جهاز الخادم باستخدام مِلَفّ docker-compose ينسخ أو يحمل من منظمة RhazesEMR على Github. يحتوى المِلَفّ الخطوات اللازمة لإنشاء وربط قاعدة المعطيات مع الـAPI، مع إبقاء قاعدة المعطيات في عزلة عن الشبكة وتكون الطريقة الوحيدة للتواصل معها عبر الـAPI. بذلك، نضاعف أمان البيانات وفرص مقاومتها لمختلف الهجمات الممكنة.

3.4 تطبيق عابر للمنصات

تطبيق طور بإطار العمل Tauri يمكن تثبيته على مختلف أنظمة التشغيل، مما يتيح للمستخدمين الوصول إلى Rhazes EMR بِغضّ النظر عن النظام المستخدم. تم تصميم واجهة المستخدم باستخدام Svelte وTailwindCSS، مما يوفر تجربة استخدام سلسة وسريعة الاستجابة.

II استراتيجيات التطوير

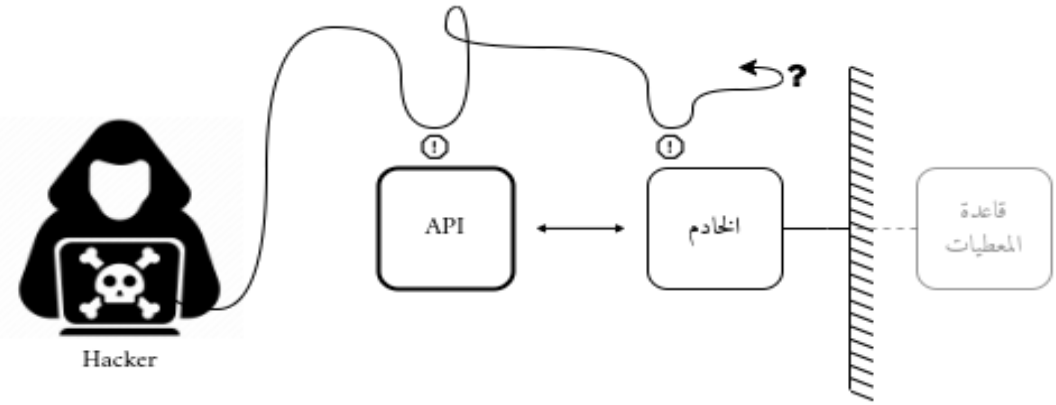
1 العزل بغرض الحماية

تم تصميم قاعدة المعطيات لتكون معزولة تماما عن الشبكة، سواءا WAN أو حتى LAN، مما يجعل واجهة برمجة التطبيقات المنفذ الوحيد للتواصل معها. هذا العزل يتم باستخدام ميزة الشبكات الداخلية بتقنية الحاويات:

```
networks:
	internal:
		internal: true
	external:

services:
	database:
		...
		networks:
			- internal
	api:
		...
		networks:
			- internal
			- external
```

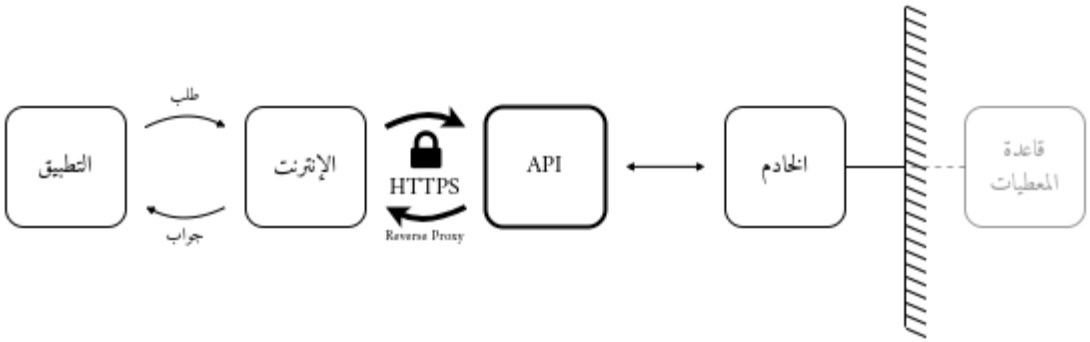
بإضافة الأسطر السابقة للملف `docker-compose.yml`، تنشئ أداة إدارة الحاويات (Podman في هذه الحالة) شبكتين إحداهما شبكة داخلية (للتواصل بين قاعدة المعطيات والـAPI) والأخرى شبكة مكشوفة (للتواصل بين الـAPI وباقي الأجهزة على نفس الشبكة).



شكل 18: رسم توضيحي لقاعدة معطيات معزولة عن الشبكة تتصدى لهجوم سيبراني

2 تشفير الاتصال

رغم أن تطبيق شهادات SSL/TLS على تطبيق محلي يعمل فقط على شبكة LAN يمكن أن يشكل تحديا، إلا أننا نستخدم ما يسمى بالreverse proxy، والذي يعمل كوسيط بين المستخدمين والتطبيق، حيث يتولى إدارة جميع جوانب التشفير والأمان، مما يسمح للتطبيق المحلي بالاستفادة من حماية SSL/TLS دون الحاجة إلى تعديل البنية الأساسية للتطبيق نفسه. هذا يبسط عملية التنفيذ بشكل كبير مع الحفاظ على مستوى عالٍ من الأمان.



شكل 19: رسم توضيحي لعمل الReverse Proxy

3 إدارة المستخدمين وصلاحياتهم

يتوافق تصميم قاعدة المعطيات وطريقة حفظنا لمعلومات المستخدمين مع بروتوكول OAuth2، وهو حل حديث لإدارة عمليات التحقق من هوية المستخدم ويسهل استغلال ميزات أمان مثل المصادقة الثنائية، رسائل التحقق عبر البريد الإلكتروني أو الرسائل القصيرة والتسجيل بنقرة واحدة عن طريق حسابات سابقة الإنشاء (Gmail أو Meta مثلا).

تقسم البيانات في حفظها على3 جداول وهي:

- User
- Role
- Permission

يحصل كل مستخدم على مجموعة من الصلاحيات بناءا على دوره، تمكنه هذه الصلاحيات من إجراء مختلف العمليات على قاعدة المعطيات كما هو موضح بالجدول التالي:

جدول 24: وصف الصلاحيات

رمز الصلاحية	وصفها
TC (Total Control)	تمكن الكتابة، الحذف، التعديل والتعديل إضافة إلى القراءة بالضرورة
W (Write)	تمكن الكتابة والتعديل إضافة إلى القراءة بالضرورة
RO (Read only)	تمكن القراءة فقط

أما الأدوار فتوزع على المستخدمين كالآتي:

جدول 25: توزيع الصلاحيات على الأدوار

الجدول	الدور		
	مستفيد	موفر	مدير النظام
person	W	W	TC
patient	RO	W	TC
concept	RO	RO	TC
provider	RO	RO	TC
visit	RO	W	TC
encounter	RO	W	TC
obs	RO	W	TC
drug	W	W	TC
order	RO	W	TC
location	RO	RO	TC
program	W	W	TC

4 دوال الاستعلام

الحوار بين واجهة برمجة التطبيقات وقاعدة المعطيات يتم عبر مكتبة SQLx، التي توفر مجموعة غنية من الميزات التي ساهمة بشكل حاسم في الإنجاز البرمجي لهذا المشروع. تبين الأمثلة التالية استغلال مكتبة SQLx في إجراء مختلف استعلامات CRUD على قاعدة المعطيات:

جدول 26: أمثلة على توظيف مكتبة *SQLx* في دوال الاستعلام

الاستعلام	دالته على جدول location

CREATE	`pub async fn create(`
	`  pool: &sqlx::PgPool,`
	`  new_location: CreateLocationDto,`
	`) → errors::Result<()> {`
	`  validate_create_person(&new_person)?`
	`  sqlx::query!(`
	`  r#"`
	`  INSERT INTO location (`
	`  name,`
	`  description,`
	`  address1,`
	`  address2,`
	`  city_village,`
	`  state_province,`
	`  postal_code,`
	`  country,`
	`  parent_location_id,`
	`  creator,`
	`  date_created,`
	`  voided,`
	`  uuid`
	`  ) VALUES (`
	`  $1, $2,`
	`  $3, $4,`
	`  $5, $6,`
	`  $7, $8,`
	`  $9, $10,`
	`  CURRENT_TIMESTAMP,`
	`  false,`
	`  gen_random_uuid()`
	`  )`
	`  "#,`
	`  new_location.name,`
	`  new_location.description,`
	`  new_location.address1,`
	`  new_location.address2,`
	`  new_location.city_village,`
	`  new_location.state_province,`
	`  new_location.postal_code,`
	`  new_location.country,`
	`  new_location.parent_location_id,`
	`  new_location.creator,`
	`  )`
	`  .execute(pool)`
	`  .await?;`
	`  Ok(())`
	`}`

`pub async fn find_by_id(`
`  pool: &sqlx::PgPool,`
`  uuid: Uuid,`
`) → errors::Result<LocationResponseDto> {`
`  let location = sqlx::query!(Location, "SELECT * FROM location WHERE uuid = $1", uuid)`
`  .fetch_optional(pool).await`
`  .map_err(errors::Error::Database)?;`
`  match location {`
`  Some(location) ⇒ Ok(LocationResponseDto {`
`  name: location.name,`
`  description: location.description,`
`  address1: location.address1,`
`  address2: location.address2,`
`  city_village: location.city_village,`
`  state_province: location.state_province,`
`  postal_code: location.postal_code,`
`  country: location.country,`
`  parent_location_id: location.parent_location_id,`
`  uuid: location.uuid,`
`  }),`
`  None ⇒ Err(errors::Error::NotFound),`
`  }`
`}`
	READ

`pub async fn update(`
`  pool: &sqlx::PgPool,`
`  uuid: Uuid,`
`  updated_location: UpdateLocationDto,`
`) → errors::Result<()> {`
`  sqlx::query!(`
`  r#"`
`  UPDATE location`
`  SET`
`  name = COALESCE($1, name),`
`  description = COALESCE($2, description),`
`  address1 = COALESCE($3, address1),`
`  address2 = COALESCE($4, address2),`
`  city_village = COALESCE($5, city_village),`
`  state_province = COALESCE($6, state_province),`
`  postal_code = COALESCE($7, postal_code),`
`  country = COALESCE($8, country),`
`  parent_location_id = COALESCE($9, parent_location_id),`
`  changed_by = COALESCE($10, changed_by),`
`  date_changed = CURRENT_TIMESTAMP,`
`  voided = COALESCE($11, voided),`
`  void_reason = COALESCE($12, void_reason)`
`  WHERE uuid = $13`
`  "#,`
`  updated_location.name,`
`  updated_location.description,`
`  updated_location.address1,`
`  updated_location.address2,`
`  updated_location.city_village,`
`  updated_location.state_province,`
`  updated_location.postal_code,`
`  updated_location.country,`
`  updated_location.parent_location_id,`
`  updated_location.changed_by,`
`  updated_location.voided,`
`  updated_location.void_reason,`
`  uuid,`
`  )`
`  .execute(pool).await?;`
`  Ok(())`
`}`
	UPDATE

`pub async fn delete(`
`  pool: &sqlx::PgPool,`
`  uuid: Uuid`
`) → errors::Result<()> {`
`  sqlx::query!(`
`  "UPDATE location SET voided = true, void_reason = 'Deleted' WHERE uuid = $1",`
`  uuid`
`  )`
`  .execute(pool)`
`  .await?;`
`  Ok(())`
`}`
	DELETE

يمكن ملاحظة توظيفنا لأحد مفاهيم برمجة OOP، وهو كائنات DTO[1]. مكنتنا هذه الكائنات من التعامل مع الجداول على أنها أفراد ثلاثية الوظائف، فكائن CreateLocationDto يستخدم في عملية الإنشاء، وكائن UpdateLocationDto يستخدم في عملية التعديل، أما كائن LocationResponseDto فيستخدم في مختلف عمليات القراءة. إضافة إلى ذلك، يتم استغلال ميزة Transaction من مكتبة SQLx عند الحاجة لإجراء مجموعة من التعليمات دفعة واحدة. ما يمنح النظام القدرة على التراجع في حال الخطأ بشكل سلس وفعال.

```
pub async fn create(pool: &PgPool, new_person: CreatePersonDto) → errors::Result<()> {
    validate_create_person(&new_person)?;

    let mut transaction = pool.begin().await?;

    let p = sqlx::query!(/* query on person table */, /* args */)
    .fetch_one(&mut *transaction)
    .await?;

    sqlx::query!(/* query on person_name table */, /* args */)
    .execute(&mut *transaction)
    .await?;

    sqlx::query!(/* query on person_address table */, /* args */)
    .execute(&mut *transaction)
    .await?;

    transaction.commit().await?;

    Ok(())
}
```

1 كائنات بسيطة تحتوي على خصائص عامة دون أي منطق، مما يسهل نقل البيانات بكفاءة عبر الشبكة أو بين مكونات النظام.

III دليل التشغيل

1 الخادم

1.ا المتطلبات

- Podman (أو Docker)

1.ب الخطوات

يحفظ الملف الآتي باسم `docker-compose.yml` في مسار فارغ:

```
services:
	rhazesemrdb:
		container_name: rhazesemrdb
		image: ghcr.io/rhazesemr/rhazesemr-database:latest
		environment:
			POSTGRES_PASSWORD: admin
			POSTGRES_DB: rhazesemrdb
			DATABASE_URL:
postgres://postgres:admin@127.0.0.1:5432/rhazesemrdb
		volumes:
			- data:/var/lib/postgresql/data
		networks:
			- internal
	rhazesemr-api:
		container_name: rhazesemr-api
		image: ghcr.io/rhazesemr/rhazesemr-api:latest
		environment:
			DATABASE_URL:
postgres://postgres:admin@rhazesemrdb:5432/rhazesemrdb
		ports:
			- "8808:3000"
		depends_on:
			rhazesemrdb:
		networks:
			- internal
			- external
volumes:
	data:
networks:
	internal:
		internal: true
	external:
```

بعد ذلك يتم تشغيل الحاويات باستخدام Podman عن طريق الأمر `podman-compose up -d`. يمكن أيضا استخدام برنامج الـmigrations المدمج بحاوية قاعدة المعطيات للتأكد من نجاح إنشائها وهيكلتها عن طريق الأمر `podman exec -it rhazesemrdb db-migrator`. يمكن الوصول إلى التوثيق الكامل للمنصة عبر الرابط التالي:

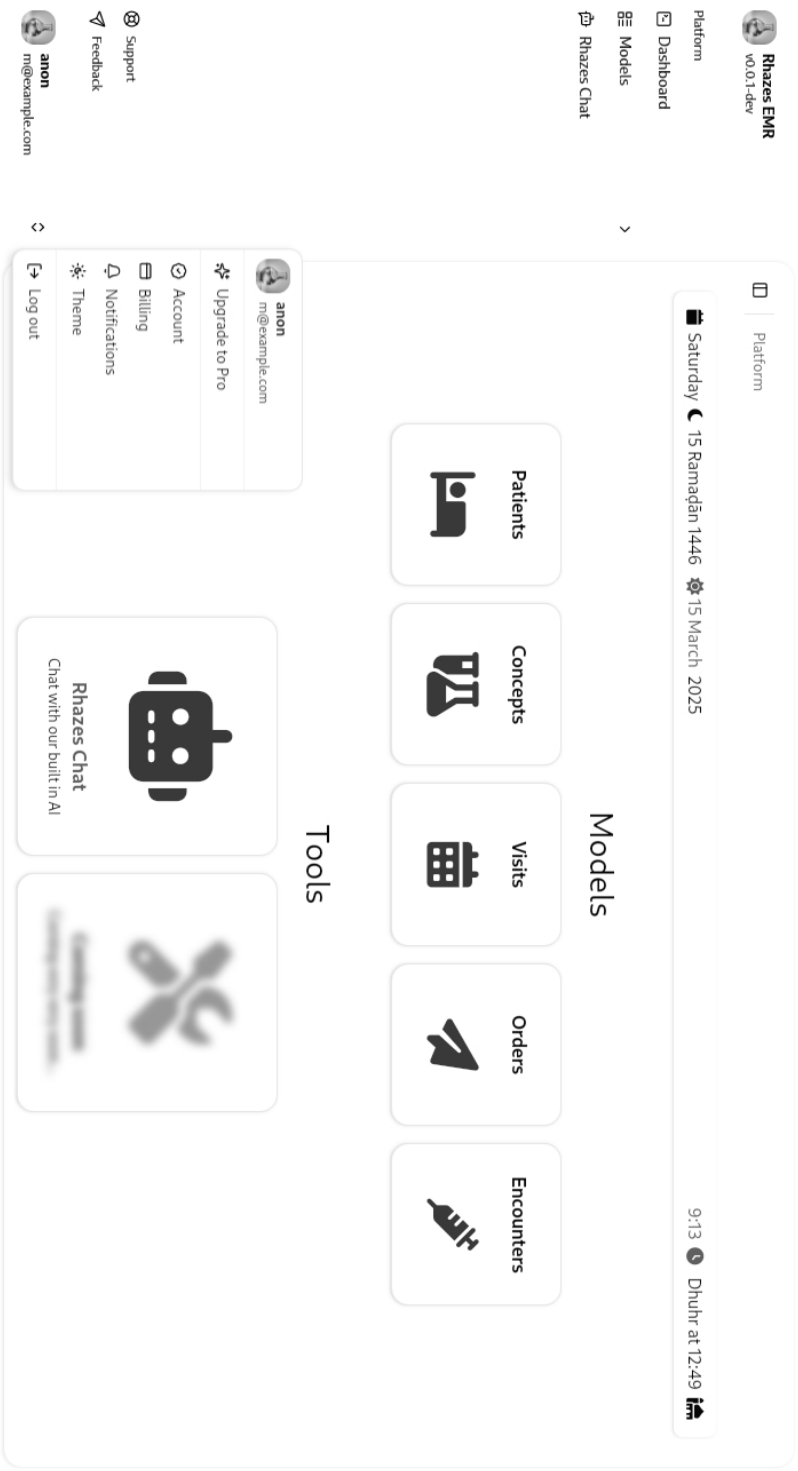
شكل 20: *docs.rhazesemr.info*

2 تطبيق RhazesEMR Client

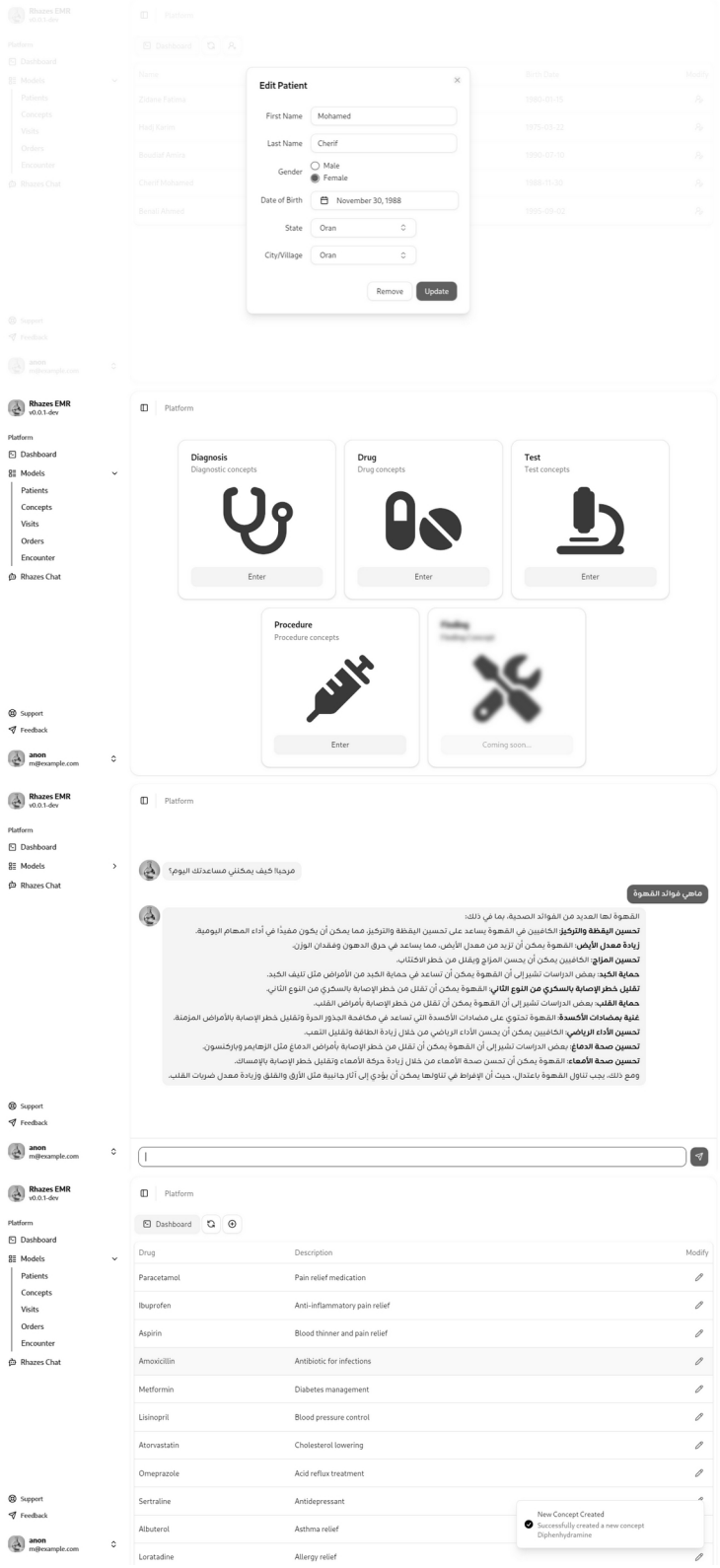
ترفع آخر إصدارات تطبيق RhazesEMR Client إلى Github Releases وSourceForge. للتحميل، يمكن زيارة الرابط التالي واختيار الإصدار المناسب حسب حاجة المستخدم:

شكل 21: *rhazesemr.info/download*

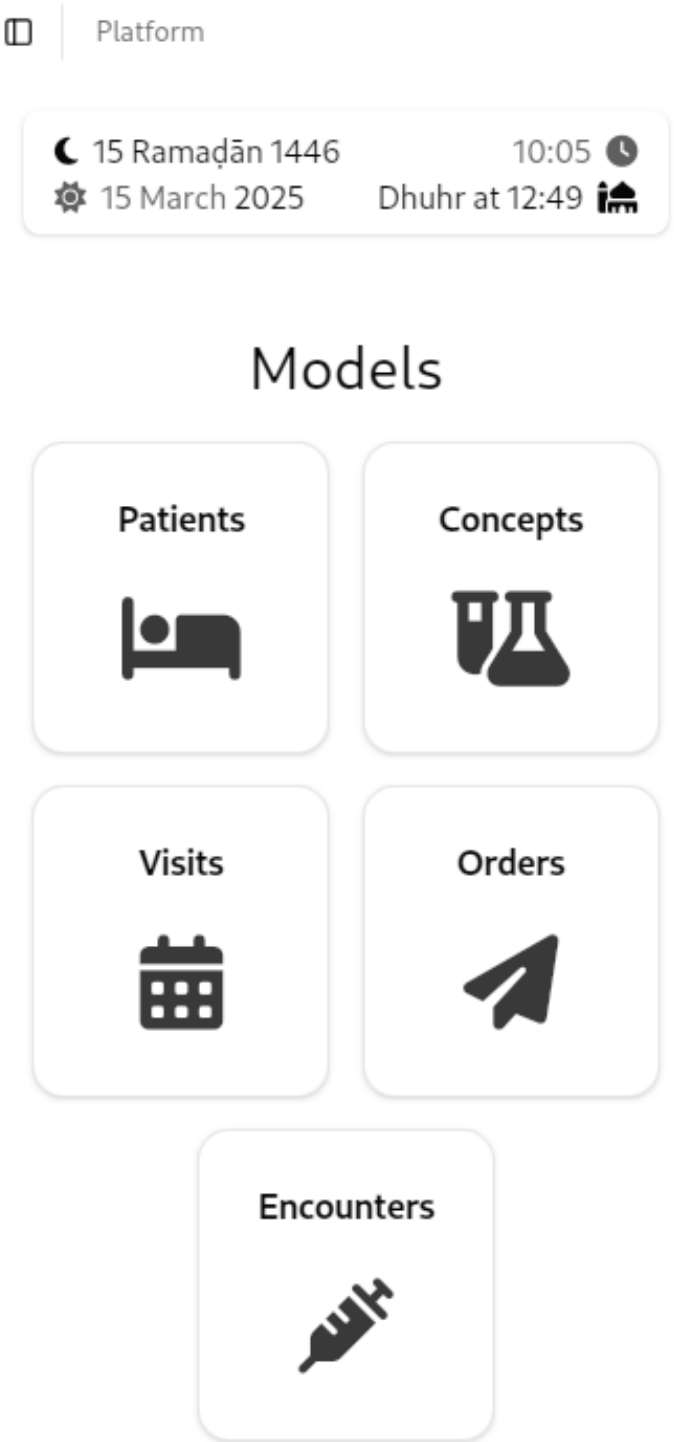
١.٢ واجهة التطبيق



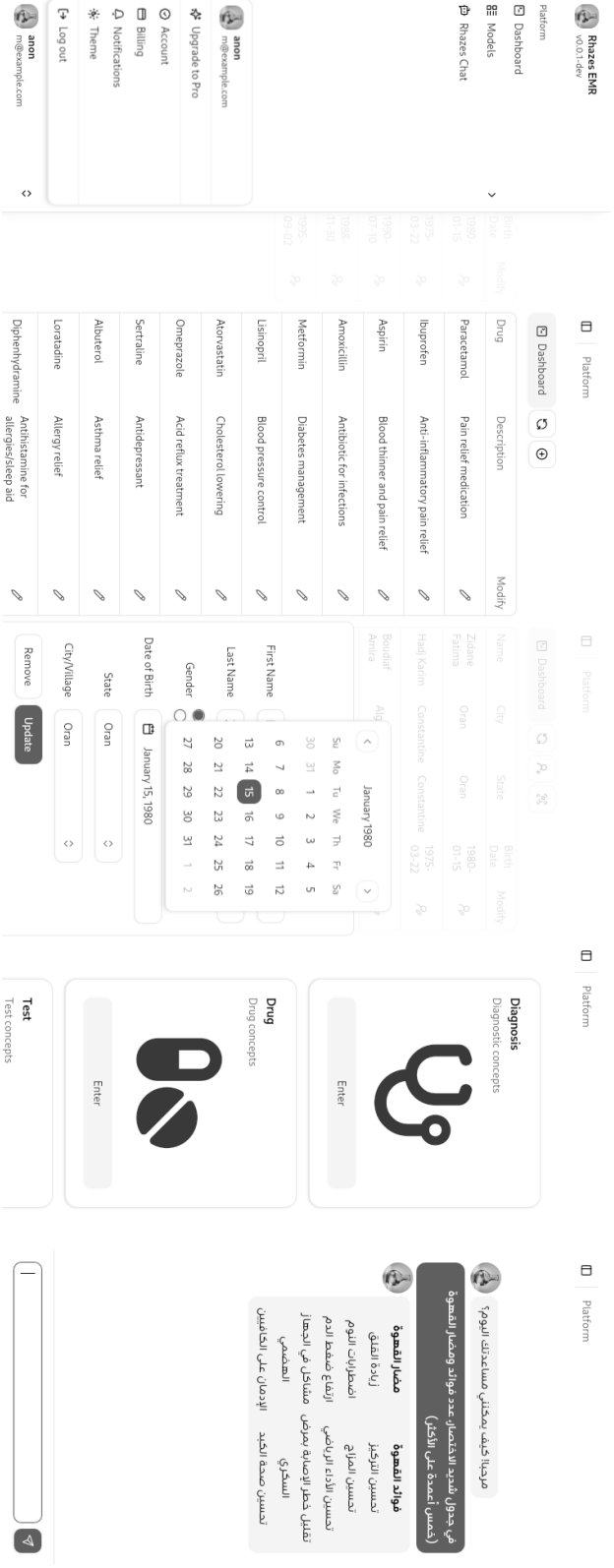
شكل 22: الواجهة الرئيسية لتطبيق *RhazesEMR* (*Desktop*)



شكل 23: تصميم تطبيق *RhazesEMR* (*Desktop*)



شكل 24: الواجهة الرئيسية لتطبيق *RhazesEMR* (*Mobile*)



شكل 25: تصميم تطبيق *RhazesEMR* (*Mobile*)

IV النموذج الفيزيائي للمعطيات

يعتبر مدخل للبرمجة حيث يأخذ بعين الاعتبار نظام تسيير قواعد المعطيات، ويسمح بوصف وإدخال مجموعة من المعطيات المستخرجة من النموذج المنطقي مع إبراز خصائص كل فرد على حدى[31].

1 وصف النموذج الفيزيائي للمعطيات MPD

جدول 27: *MPD* (*person*)

Table name: person	**Primary key:** person_id

Column	**Column name**	**Type**	**Length**	**Observation**
01	person_id	SERIAL	10	PK
02	gender	CHAR	1	NOT NULL
03	birth_date	DATE	10	DEFAULT NULL
04	birth_time	TIME	8	DEFAULT NULL
05	birth_date_estimated	BOOLEAN	1	DEFAULT FALSE
06	dead	BOOLEAN	1	DEFAULT FALSE
07	death_date	DATE	10	DEFAULT NULL
08	death_time	TIME	8	DEFAULT NULL
09	death_date_estimated	BOOLEAN	1	DEFAULT FALSE
10	cause_of_death	INT	10	DEFAULT NULL
11	cause_of_death_non_coded	VARCHAR	256	DEFAULT NULL
Total			316	

```
CREATE TABLE person (
  person_id SERIAL PRIMARY KEY,
  gender CHAR(1) NOT NULL,
  birth_date DATE DEFAULT NULL,
  birth_time TIME DEFAULT NULL,
  birth_date_estimated BOOLEAN NOT NULL DEFAULT FALSE,
  dead BOOLEAN NOT NULL DEFAULT FALSE,
  death_date DATE DEFAULT NULL,
  death_time TIME DEFAULT NULL,
  death_date_estimated BOOLEAN NOT NULL DEFAULT FALSE,
  cause_of_death INT DEFAULT NULL, -- FK to concept.concept_id
  cause_of_death_non_coded varchar(256) DEFAULT NULL,
  CONSTRAINT fk_person_concept_cause_of_death FOREIGN KEY (cause_of_death)
REFERENCES concept (concept_id),
  CONSTRAINT ch_gender CHECK (gender IN ('M', 'F')),
  CONSTRAINT ch_lifetime CHECK (
    dead IS FALSE
    OR (birth_date ≤ death_date)
  )
);
```

جدول 28: *MPD* (*person_name*)

Table name: person_name	**Primary key:** person_name_id

Column	Column name	Type	Length	Observation
01	person_name_id	SERIAL	10	PK
02	person_id	INT	10	NOT NULL
03	preferred	BOOLEAN	1	DEFAULT FALSE
04	degree	VARCHAR	64	DEFAULT NULL
05	given_name	VARCHAR	256	NOT NULL
06	middle_name	VARCHAR	256	DEFAULT NULL
07	family_name	VARCHAR	256	NOT NULL
08	given_name_ar	VARCHAR	256	DEFAULT NULL
09	middle_name_ar	VARCHAR	256	DEFAULT NULL
10	family_name_ar	VARCHAR	256	DEFAULT NULL
Total			1621	

```
CREATE TABLE person_name (
  person_name_id SERIAL PRIMARY KEY,
  person_id INT NOT NULL, -- FK to person(person_id)
  preferred BOOLEAN NOT NULL DEFAULT FALSE,
  degree VARCHAR(64) DEFAULT NULL,
  -- latin spelling
  given_name VARCHAR(256) NOT NULL,
  middle_name VARCHAR(256) DEFAULT NULL,
  family_name VARCHAR(256) NOT NULL,
  -- arabic spelling
  given_name_ar VARCHAR(256) DEFAULT NULL,
  middle_name_ar VARCHAR(256) DEFAULT NULL,
  family_name_ar VARCHAR(256) DEFAULT NULL,
  CONSTRAINT fk_person_name_creator FOREIGN KEY (creator) REFERENCES users
(user_id),
  CONSTRAINT fk_person_name_changed_by FOREIGN KEY (changed_by) REFERENCES
users (user_id),
  CONSTRAINT fk_person_name_voided_by FOREIGN KEY (voided_by) REFERENCES users
(user_id),
  CONSTRAINT fk_person_name FOREIGN KEY (person_id) REFERENCES person
(person_id)
);
```

جدول 29: *MPD* (*person_address*)

Table name: person_address	**Primary key:** person_address_id

Column	Column name	Type	Length	Observation
01	person_address_id	SERIAL	10	PK
02	person_id	INT	10	NOT NULL
03	preferred	BOOLEAN	1	DEFAULT FALSE
04	address1	VARCHAR	256	DEFAULT NULL
05	address2	VARCHAR	256	DEFAULT NULL
06	city_village	VARCHAR	256	NOT NULL
07	state_province	VARCHAR	2	NOT NULL
08	postal_code	VARCHAR	64	DEFAULT NULL
09	country	VARCHAR	256	DEFAULT 'Algeria'
Total			1111	

```
CREATE TABLE person_address (
  person_address_id SERIAL PRIMARY KEY,
  person_id INT NOT NULL, -- FK to person(person_id)
  preferred BOOLEAN NOT NULL DEFAULT FALSE,
  address1 VARCHAR(256) DEFAULT NULL,
  address2 VARCHAR(256) DEFAULT NULL,
  city_village VARCHAR(256) NOT NULL,
  state_province VARCHAR(2) NOT NULL,
  postal_code VARCHAR(64) DEFAULT NULL,
  country VARCHAR(256) NOT NULL DEFAULT 'Algeria',
  CONSTRAINT fk_person_address FOREIGN KEY (person_id) REFERENCES person
(person_id)
);
```

جدول 30: *MPD* (*patient*)

Table name: patient	**Primary key:** patient_id

Column	Column name	Type	Length	Observation
01	patient_id	SERIAL	10	PK
Total			10	

```
CREATE TABLE patient (
  patient_id INT PRIMARY KEY,
  CONSTRAINT fk_patient_person_self FOREIGN KEY (patient_id) REFERENCES person
(person_id)
);
```

جدول 31: *MPD* (*patient_identifier*)

Table name: patient_identifier	**Primary key:** patient_identifier_id

Column	**Column name**	**Type**	**Length**	**Observation**
01	patient_identifier_id	SERIAL	10	PK
02	patient_id	INT	10	NOT NULL
03	identifier_type_id	INT	10	NOT NULL
04	identifier	VARCHAR	256	UNIQUE
05	preferred	BOOLEAN	1	DEFAULT FALSE
Total			287	

```
CREATE TABLE person_address (
  person_address_id SERIAL PRIMARY KEY,
  person_id INT NOT NULL, -- FK to person(person_id)
  preferred BOOLEAN NOT NULL DEFAULT FALSE,
  address1 VARCHAR(256) DEFAULT NULL,
  address2 VARCHAR(256) DEFAULT NULL,
  city_village VARCHAR(256) NOT NULL,
  state_province VARCHAR(2) NOT NULL,
  postal_code VARCHAR(64) DEFAULT NULL,
  country VARCHAR(256) NOT NULL DEFAULT 'Algeria',
  CONSTRAINT fk_person_address FOREIGN KEY (person_id) REFERENCES person
(person_id)
);
```

جدول 32: *MPD* (*patient_identifier_type*)

Table name: patient_identifier_type	**Primary key:** patient_identifier_type_id

Column	Column name	Type	Length	Observation
01	patient_identifier_type_id	SERIAL	10	PK
02	name	VARCHAR	128	NOT NULL
03	description	TEXT	/	DEFAULT NULL
04	format	VARCHAR	256	DEFAULT NULL
05	check_digit	BOOLEAN	1	DEFAULT FALSE
Total			395	

```
CREATE TABLE patient_identifier_type (
  identifier_type_id SERIAL PRIMARY KEY,
  name VARCHAR(128) NOT NULL,
  description TEXT DEFAULT NULL,
  format VARCHAR(256) DEFAULT NULL,
  check_digit BOOLEAN NOT NULL DEFAULT FALSE
);
```

جدول 33: *MPD* (*concept*)

Table name: concept	**Primary key:** concept_id

Column	Column name	Type	Length	Observation
01	concept_id	SERIAL	10	PK
02	name	VARCHAR	128	NOT NULL
03	description	TEXT	/	DEFAULT NULL
04	is_set	BOOLEAN	1	DEFAULT FALSE
05	concept_datatype_id	INT	10	NOT NULL
Total			149	

```
CREATE TABLE concept (
  concept_id SERIAL PRIMARY KEY,
  name VARCHAR(128) NOT NULL,
  description TEXT DEFAULT NULL,
  datatype_id INT NOT NULL, -- FK to concept_datatype.concept_datatype_id
  class_id INT NOT NULL, -- FK to concept_class.concept_class_id
  is_set BOOLEAN NOT NULL DEFAULT FALSE,
  CONSTRAINT fk_concept_datatype FOREIGN KEY (datatype_id) REFERENCES
concept_datatype (concept_datatype_id),
  CONSTRAINT fk_concept_class FOREIGN KEY (class_id) REFERENCES concept_class
(concept_class_id)
);
```

جدول 34: *MPD* (*concept_class*)

Table name: concept_class	**Primary key:** concept_class_id

Column	Column name	Type	Length	Observation
01	concept_class_id	SERIAL	10	PK
02	name	VARCHAR	128	NOT NULL
03	description	TEXT	/	DEFAULT NULL
Total			138	

```
CREATE TABLE concept_class (
  concept_class_id SERIAL PRIMARY KEY,
  name VARCHAR(128) NOT NULL UNIQUE,
  description TEXT DEFAULT NULL
);
```

جدول 35: *MPD* (*concept_datatype*)

Table name: concept_datatype	**Primary key:** concept_datatype_id

Column	**Column name**	**Type**	**Length**	**Observation**
01	concept_datatype_id	SERIAL	10	PK
02	name	VARCHAR	128	NOT NULL
03	hl7_abbreviation	VARCHAR	3	DEFAULT NULL
04	description	TEXT	/	DEFAULT NULL
Total			141	

```
CREATE TABLE concept_datatype (
  concept_datatype_id SERIAL PRIMARY KEY,
  name VARCHAR(128) NOT NULL UNIQUE,
  hl7_abbreviation VARCHAR(3) DEFAULT NULL,
  description TEXT DEFAULT NULL,
  CONSTRAINT ch_hl7_abbreviation CHECK (
    hl7_abbreviation IN (
      'NM', -- Numeric
      'ST', -- String/Text
      'CWE', -- Coded
      'DT', -- Date
      'TM', -- Time
      'TS', -- Timestamp
      'BIT', -- Boolean
      'ED', -- Encapsulated Data
      'RP', -- Reference Pointer
      'SN' -- Structured Numeric
    )
  )
);
```

جدول 36: *MPD* (*provider*)

Table name: provider	**Primary key:** provider_id

Column	**Column name**	**Type**	**Length**	**Observation**
01	provider_id	SERIAL	10	PK
02	person_id	INT	10	NOT NULL
03	preferred	BOOLEAN	1	DEFAULT FALSE
04	address1	VARCHAR	256	DEFAULT NULL
05	address2	VARCHAR	256	DEFAULT NULL
06	city_village	VARCHAR	256	NOT NULL
07	state_province	VARCHAR	2	NOT NULL
08	postal_code	VARCHAR	64	DEFAULT NULL
09	country	VARCHAR	256	DEFAULT 'Algeria'
Total			1111	

```
CREATE TABLE person_address (
  person_address_id SERIAL PRIMARY KEY,
  person_id INT NOT NULL, -- FK to person(person_id)
  preferred BOOLEAN NOT NULL DEFAULT FALSE,
  address1 VARCHAR(256) DEFAULT NULL,
  address2 VARCHAR(256) DEFAULT NULL,
  city_village VARCHAR(256) NOT NULL,
  state_province VARCHAR(2) NOT NULL,
  postal_code VARCHAR(64) DEFAULT NULL,
  country VARCHAR(256) NOT NULL DEFAULT 'Algeria',
  CONSTRAINT fk_person_address FOREIGN KEY (person_id) REFERENCES person
(person_id)
);
```

جدول 37: *MPD* (*visit*)

Table name: visit	**Primary key:** visit_id

Column	Column name	Type	Length	Observation
01	visit_id	SERIAL	10	PK
02	date_started	TIMESTAMP	19	NOT NULL
03	date_stopped	TIMESTAMP	19	DEFAULT FALSE
04	visit_type_id	INT	10	DEFAULT NULL
05	patient_id	INT	10	DEFAULT NULL
06	location_id	INT	10	NOT NULL
07	indication_concept_id	INT	10	NOT NULL
Total			88	

```
CREATE TABLE visit (
  visit_id SERIAL PRIMARY KEY,
  patient_id INT NOT NULL, -- FK to patient(patient_id)
  visit_type_id INT NOT NULL, -- FK to visit_type.visit_type_id
  date_started TIMESTAMP NOT NULL,
  date_stopped TIMESTAMP DEFAULT NULL,
  indication_concept_id INT NOT NULL, -- FK to concept.concept_id
  location_id INT NOT NULL, -- FK to location(location_id)
  CONSTRAINT fk_visit_patient FOREIGN KEY (patient_id) REFERENCES patient
(patient_id),
  CONSTRAINT fk_visit_type FOREIGN KEY (visit_type_id) REFERENCES visit_type
(visit_type_id),
  CONSTRAINT fk_visit_concept FOREIGN KEY (indication_concept_id) REFERENCES
concept (concept_id),
  CONSTRAINT fk_visit_location FOREIGN KEY (location_id) REFERENCES location
(location_id),
  CONSTRAINT ch_date_stopped CHECK (
    date_stopped IS NULL
    OR date_stopped > date_started
  )
);
```

جدول 38: *MPD* (*visit_type*)

Table name: visit_type	**Primary key:** visit_type_id

Column	Column name	Type	Length	Observation
01	visit_type_id	SERIAL	10	PK
02	name	VARCHAR	128	NOT NULL
03	description	TEXT	/	DEFAULT FALSE
Total			138	

```
CREATE TABLE visit_type (
  visit_type_id SERIAL PRIMARY KEY,
  name VARCHAR(128) NOT NULL,
  description TEXT DEFAULT NULL
);
```

جدول 39: *MPD* (*encounter_provider*)

Table name: encounter_provider	**Primary key:** encounter_provider_id

Column	Column name	Type	Length	Observation
01	encounter_provider_id	SERIAL	10	PK
02	encounter_id	TIMESTAMP	19	NOT NULL
03	provider_id	TIMESTAMP	19	DEFAULT FALSE
Total			48	

```
CREATE TABLE encounter_provider (
  encounter_provider_id SERIAL PRIMARY KEY,
  encounter_id INT NOT NULL, -- FK to encounter.encounter_id
  provider_id INT NOT NULL, -- FK to provider.provider_id
  encounter_role_id INT NOT NULL, -- FK to encounter_role.encounter_role_id
  CONSTRAINT fk_encounter_provider_encounter FOREIGN KEY (encounter_id)
REFERENCES encounter (encounter_id),
  CONSTRAINT fk_encounter_provider_provider FOREIGN KEY (provider_id)
REFERENCES provider (provider_id),
```

```
  CONSTRAINT fk_encounter_provider_encounter_role FOREIGN KEY
(encounter_role_id) REFERENCES encounter_role (encounter_role_id),
  CONSTRAINT uq_encounter_provider UNIQUE (encounter_id, provider_id,
encounter_role_id)
);
```

جدول 40: *MPD* (*encounter*)

Table name: encounter	**Primary key:** encounter_id

Column	Column name	Type	Length	Observation
01	encounter_id	SERIAL	10	PK
02	datetime	TIMESTAMP	19	NOT NULL
03	status	VARCHAR	64	DEFAULT 'PLANNED'
04	visit_id	INT	10	NOT NULL
05	patient_id	INT	10	NOT NULL
06	encounter_type_id	INT	10	NOT NULL
07	location_id	INT	10	NOT NULL
Total			133	

```
CREATE TABLE encounter (
  encounter_id SERIAL PRIMARY KEY,
  encounter_type_id INT NOT NULL, -- FK to encounter_type.encounter_type_id
  patient_id INT NOT NULL, -- FK to patient.patient_id
  visit_id INT NOT NULL, -- FK to visit.visit_id
  location_id INT NOT NULL, -- FK to location.location_id
  encounter_datetime TIMESTAMP NOT NULL,
  status VARCHAR(64) DEFAULT 'PLANNED',
  CONSTRAINT fk_encounter_type FOREIGN KEY (encounter_type_id) REFERENCES
encounter_type (encounter_type_id),
  CONSTRAINT fk_encounter_patient FOREIGN KEY (patient_id) REFERENCES patient
(patient_id),
  CONSTRAINT fk_encounter_visit FOREIGN KEY (visit_id) REFERENCES visit
(visit_id),
  CONSTRAINT fk_encounter_location FOREIGN KEY (location_id) REFERENCES
location (location_id),
  CONSTRAINT ch_encounter_status CHECK (
    status IN (
      'PLANNED',
      'ARRIVED',
      'STARTED',
```

```
      'COMPLETED',
      'EXITED',
      'CANCELLED'
    )
  )
);
```

جدول 41: *MPD* (*obs*)

Table name: obs	**Primary key:** obs_id

Column	Column name	Type	Length	Observation
01	obs_id	SERIAL	10	PK
02	person_id	INT	10	NOT NULL
03	value	JSONB	/	NOT NULL
04	comments	TEXT	/	DEFAULT NULL
05	encounter_id	INT	10	DEFAULT NULL
06	location_id	INT	10	NOT NULL
07	obs_group_id	INT	10	DEFAULT NULL
08	concept_id	INT	10	NOT NULL
Total			50	

```
CREATE TABLE obs (
  obs_id SERIAL PRIMARY KEY,
  person_id INT NOT NULL, -- FK to person.person_id
  encounter_id INT DEFAULT NULL, -- FK to encounter.encounter_id
  concept_id INT NOT NULL, -- FK to concept.concept_id
  obs_datetime TIMESTAMP NOT NULL,
  location_id INT NOT NULL, -- FK to location.location_id
  obs_group_id INT DEFAULT NULL, -- FK to obs.obs_id
  value JSONB NOT NULL,
  comments TEXT DEFAULT NULL,
  CONSTRAINT fk_obs_person FOREIGN KEY (person_id) REFERENCES person
(person_id),
  CONSTRAINT fk_obs_encounter FOREIGN KEY (encounter_id) REFERENCES encounter
(encounter_id),
  CONSTRAINT fk_obs_concept FOREIGN KEY (concept_id) REFERENCES concept
(concept_id),
  CONSTRAINT fk_obs_order FOREIGN KEY (order_id) REFERENCES orders (order_id),
```

```
  CONSTRAINT fk_obs_location FOREIGN KEY (location_id) REFERENCES location
(location_id),
  CONSTRAINT fk_obs_group FOREIGN KEY (obs_group_id) REFERENCES obs (obs_id),
  CONSTRAINT fk_obs_value_group FOREIGN KEY (value_group_id) REFERENCES obs
(obs_id) );
```

جدول 42: *MPD* (*drug*)

Table name: drug	**Primary key:** drug_id

Column	Column name	Type	Length	Observation
01	drug_id	SERIAL	10	PK
02	name	VARCHAR	256	NOT NULL
03	combination	BOOLEAN	1	DEFAULT FALSE
04	dosage_form	INT	10	DEFAULT NULL
05	concept_id	BOOLEAN	10	NOT NULL
Total			287	

```
CREATE TABLE drug (
  drug_id SERIAL PRIMARY KEY,
  concept_id INT NOT NULL, -- FK to concept.concept_id
  name VARCHAR(256) NOT NULL,
  combination BOOLEAN NOT NULL DEFAULT FALSE,
  dosage_form INT DEFAULT NULL, -- FK to concept.concept_id
  CONSTRAINT fk_drug_concept_primary FOREIGN KEY (concept_id) REFERENCES
concept (concept_id),
  CONSTRAINT fk_drug_dosage_form FOREIGN KEY (dosage_form) REFERENCES concept
(concept_id)
);
```

جدول 43: *MPD* (*orders*)

Table name: orders	**Primary key:** order_id

Column	Column name	Type	Length	Observation
01	order_id	SERIAL	10	PK
02	urgency	INT	10	NOT NULL
03	order_number	VARCHAR	256	UNIQE
04	order_action	VARCHAR	64	NOT NULL
05	date_activated	TIMESTAMP	19	NOT NULL
06	auto_expire_date	TIMESTAMP	19	DEFAULT NULL
07	date_stopped	TIMESTAMP	19	DEFAULT NULL
08	patient_id	INT	10	NOT NULL
09	orderer_id	INT	10	NOT NULL
10	encounter_id	INT	10	NOT NULL
11	concept_id	INT	10	NOT NULL
Total			397	

```
CREATE TABLE orders (
  order_id SERIAL PRIMARY KEY,
  order_type_id INT NOT NULL, -- FK to order_type.order_type_id
  concept_id INT NOT NULL, -- FK to concept.concept_id
  orderer_id INT NOT NULL, -- FK to provider.provider_id
  encounter_id INT NOT NULL, -- FK to encounter.encounter_id
  patient_id INT NOT NULL, -- FK to patient.patient_id
  urgency VARCHAR(64) NOT NULL,
  order_number VARCHAR(256) NOT NULL UNIQUE,
  order_action VARCHAR(64) NOT NULL,
  date_activated TIMESTAMP NOT NULL,
  auto_expire_date TIMESTAMP DEFAULT NULL,
  date_stopped TIMESTAMP DEFAULT NULL,
  CONSTRAINT fk_order_type FOREIGN KEY (order_type_id) REFERENCES order_type
(order_type_id),
  CONSTRAINT fk_order_concept_reason FOREIGN KEY (concept_id) REFERENCES
concept (concept_id),
  CONSTRAINT fk_order_provider_orderer FOREIGN KEY (orderer_id) REFERENCES
provider (provider_id),
```

```
  CONSTRAINT fk_order_encounter_ordered FOREIGN KEY (encounter_id) REFERENCES
encounter (encounter_id),
  CONSTRAINT fk_order_patient FOREIGN KEY (patient_id) REFERENCES patient
(patient_id),
  CONSTRAINT ch_order_urgency CHECK (
    urgency IN ('ROUTINE', 'STAT', 'URGENT', 'ON_SCHEDULED_DATE')
  ),
  CONSTRAINT ch_order_action CHECK (
    order_action IN ('NEW', 'REVISE', 'DISCONTINUE', 'RENEW')
  ),
  -- For dates logic
  CONSTRAINT ch_order_dates CHECK (
    (
      (date_stopped IS NULL)
      OR (date_stopped > date_activated)
    )
    AND (
      (auto_expire_date IS NULL)
      OR (auto_expire_date > date_activated)
    )
  )
);
```

جدول 44: *MPD* (*order_type*)

Table name: order_type	**Primary key:** order_type_id

Column	Column name	Type	Length	Observation
01	order_type_id	SERIAL	10	PK
02	name	VARCHAR	128	NOT NULL
03	description	TEXT	256	UNIQE
Total			394	

```
CREATE TABLE order_type (
  order_type_id SERIAL PRIMARY KEY,
  name VARCHAR(128) NOT NULL,
  description TEXT DEFAULT NULL
);
```

جدول 45: *MPD* (*location*)

Table name: location	**Primary key:** location_id

Column	Column name	Type	Length	Observation
01	location_id	SERIAL	10	PK
02	name	VARCHAR	128	DEFAULT NULL
03	description	TEXT	/	DEFAULT NULL
04	address1	VARCHAR	256	DEFAULT NULL
05	address2	VARCHAR	256	DEFAULT NULL
06	city_village	VARCHAR	256	NOT NULL
07	state_province	VARCHAR	2	NOT NULL
08	postal_code	VARCHAR	64	DEFAULT NULL
09	country	VARCHAR	256	DEFAULT 'Algeria'
Total			1228	

```
CREATE TABLE location (
  location_id SERIAL PRIMARY KEY,
  name VARCHAR(128) DEFAULT NULL,
  description TEXT DEFAULT NULL,
  address1 VARCHAR(256) DEFAULT NULL,
  address2 VARCHAR(256) DEFAULT NULL,
  city_village VARCHAR(256) NOT NULL,
  state_province VARCHAR(2) NOT NULL,
  postal_code VARCHAR(64) DEFAULT NULL,
  country VARCHAR(256) NOT NULL DEFAULT 'Algeria',
  parent_location_id INT DEFAULT NULL, -- self-FK to location(location_id)
  CONSTRAINT fk_location_parent FOREIGN KEY (parent_location_id) REFERENCES
location (location_id),
  CONSTRAINT ch_no_self_reference CHECK (parent_location_id ≠ location_id)
);
```

جدول 46: *MPD* (*program*)

Table name: program	**Primary key:** program_id

Column	**Column name**	**Type**	**Length**	**Observation**
01	program_id	SERIAL	10	PK
02	name	VARCHAR	128	NOT NULL
03	description	TEXT	/	DEFAULT NULL
04	concept_id	INT	10	NOT NULL
05	outcomes_concept_id	INT	10	NOT NULL
Total			158	

```
CREATE TABLE program (
  program_id SERIAL PRIMARY KEY,
  concept_id INT NOT NULL, -- FK to concept.concept_id
  outcomes_concept_id INT NOT NULL, -- FK to concept.concept_id
  name VARCHAR(256) NOT NULL,
  description TEXT DEFAULT NULL,
  CONSTRAINT fk_program_concept FOREIGN KEY (concept_id) REFERENCES concept
(concept_id),
  CONSTRAINT fk_program_concept_outcomes FOREIGN KEY (outcomes_concept_id)
REFERENCES concept (concept_id)
);
```

جدول 47: *MPD* (*patient_program*)

Table name: patient_program	**Primary key:** patient_program_id

Column	**Column name**	**Type**	**Length**	**Observation**
01	patient_program_id	SERIAL	10	PK
02	date_enrolled	TIMESTAMP	19	NOT NULL
03	date_completed	TIMESTAMP	19	DEFAULT NULL
04	patient_id	INT	10	NOT NULL
05	program_id	INT	10	NOT NULL
06	location_id	INT	10	NOT NULL
Total			68	

```
CREATE TABLE patient_program (
  patient_program_id SERIAL PRIMARY KEY,
  patient_id INT NOT NULL, -- FK to patient(patient_id)
  program_id INT NOT NULL, -- FK to program.program_id
  provider_id INT NOT NULL, -- FK to provider(provider_id)
  date_enrolled TIMESTAMP NOT NULL,
  date_completed TIMESTAMP DEFAULT NULL,
  location_id INT NOT NULL, -- FK to location(location_id)
  CONSTRAINT fk_patient_program_patient FOREIGN KEY (patient_id) REFERENCES
patient (patient_id),
  CONSTRAINT fk_patient_program_program FOREIGN KEY (program_id) REFERENCES
program (program_id),
  CONSTRAINT fk_patient_program_location FOREIGN KEY (location_id) REFERENCES
location (location_id),
  CONSTRAINT ch_patient_program_dates CHECK (
    date_completed IS NULL
    OR date_completed > date_enrolled
  )
);
```

الخاتمة

في ختام هذا العمل الذي كانت بدايته متابعة ميدانية بالمؤسسة العمومية للصحة الجوارية بولاية المسيلة، وبالتحديد المديرية الفرعية للمصالح الصحية، وبعد الاطّلاع عليها فقد اكتسبنا الخبرة الميدانية فيها متبعين الخطوات والمراحل التي تم دراستها خلال مدّة التربص وتمكنا من الوصول إلى المرحلة النهائية، فصممننا وأنجزنا منصة رقمية تسمح بإدارة السجلات الطبية للمرضى والوصول إلى المعلومات ومعاينتها وحفظها بكل سهولة ودقة واستخراجها وقت الضرورة، مما يؤدي إلى تحسين جودة الخدمة الصحية وتنظيم وتسهيل تبادل المعلومات بين مختلف الأقسام الطبية وتطوير سير العمل في المؤسسة الصحية في إطار رقمنة القطاع الصحي.

مصادر عربية

10: الهيئة السعودية للبيانات والذكاء الاصطناعي و مجمع الملك سلمان العالمي للغة العربية، معجم البيانات والذكاء الاصطناعي، 2022

11: ويكيبيديا، قاعدة بيانات، 2024 https://ar.wikipedia.org/wiki/Database [تاريخ الاطلاع: 2025-01-05]

12: ويكيبيديا، نموذج قاعدة البيانات الهرمية، 2023 https://ar.wikipedia.org/wiki/Hierarchical_database_model [تاريخ الاطلاع: 2025-03-02]

13: ويكيبيديا، نموذج الشبكة، 2024 https://ar.wikipedia.org/wiki/Network_model [تاريخ الاطلاع: 2025-03-02]

17: ويكيبيديا، قاعدة بيانات موزعة، 2024 https://ar.wikipedia.org/wiki/Distributed_database [تاريخ الاطلاع: 2025-03-02]

18: Wikipedia contributors، Blockchain-based database، 2025 https://en.wikipedia.org/wiki/Blockchain-based_database [Accessed: 2025-03-02]

19: ويكيبيديا، قاعدة البيانات الكائنية، 2025 https://ar.wikipedia.org/wiki/Object_database [تاريخ الاطلاع: 2025-03-02]

20: ويكيبيديا، قاعدة البيانات غير العلائقية، 2024 https://ar.wikipedia.org/wiki/NoSQL [تاريخ الاطلاع: 2025-03-02]

23: Amazon، ما المقصود بواجهة برمجة التطبيقات؟، 2024 https://aws.amazon.com/ar/what-is/api/ [تاريخ الاطلاع: 2025-03-02]

30: الأمانة العامة للحكومة، الجريدة الرسمية للجمهورية الجزائرية الديمقراطية الشعبية، 2007 العدد 33، السنة الرابعة والأربعون، الموافق 3 جمادى الأولى عام 1428 هـ

31: بن عيسى، س. و سلامي، آ. نعيجي، و.، تصميم وإنجاز قاعدة معطيات لمتابعة الإعارة المكتبية، 2019

32: ويكيبيديا، أسيد (معلوماتية)، 2024 https://ar.wikipedia.org/wiki/ACID [تاريخ الاطلاع: 2025-03-08]

33: ويكيبيديا، أبو بكر الرازي، 2025 https://ar.wikipedia.org/wiki/Abu_Bakr_al-Razi [تاريخ الاطلاع: 16-03-2025]

مصادر أجنبية

1: Maki, J. N. and Gruel, D. and McKinney, C. and Ravine, M. A. and Morales, M. and Lee, D. and Willson, R. and Copley-Woods, D. and Valvo, M. and Goodsall, T. and McGuire, J. and Sellar, R. G. and Schaffner, J. A. and Caplinger, M. A. and Shamah, J. M. and Johnson, A. E. and Ansari, H. and Singh, K. and Litwin, T. and Deen, R. and Culver, A. and Ruoff, N. and Petrizzo, D. and Kessler, D. and Basset, C. and Estlin, T. and Alibay, F. and Nelessen, A. and Algermissen, S., The Mars 2020 Engineering Cameras and Microphone on the Perseverance Rover: A Next-Generation Imaging System for Mars Exploration, 2020 https://pmc.ncbi.nlm.nih.gov/articles/PMC7686239/

2: Wikipedia contributors, Adoption of free and open-source software by public institutions, 2024 https://en.wikipedia.org/wiki/Adoption_of_free_and_open-source_software_by_public_institutions [Accessed: 2025-01-11]

3: BWI GmbH, Open-Source: „Matrix“ ist einheitlicher Messenger-Standard für die Bundeswehr, 2020 https://www.bwi.de/magazin/artikel/open-source-matrix-ist-einheitlicher-messenger-standard-fuer-die-bundeswehr [Accessed: 2025-01-11]

4: Kane, S. and Liberman, E. and DiViesti, T. and Click, F. and MacDonald, M., An Examination of the NHTSA and NASA Engineering Safety Center Assessment of Toyota Electronic Throttle Control Systems and Unintended Acceleration, 2011 https://www.safetyresearch.net/Library/NHTSA-NASA_Response_Final_052311.pdf [Accessed: 2025-01-12]

5: Koopman, Phil, A Case Study of Toyota Unintended Acceleration and Software Safety, 2014 https://ptolemy.berkeley.edu/projects/chess/pubs/1081/koopman14_toyota_ua_slides.pdf

6: Dunn, M., Toyotas killer firmware: Bad design and its consequences, 2013 https://www.edn.com/toyotas-killer-firmware-bad-design-and-its-consequences/

7: DARPA, Eliminating Memory Safety Vulnerabilities Once and For All, 2024 https://www.darpa.mil/news/2024/memory-safety-vulnerabilities [Accessed: 2025-01-12]

8: Khaitan, Ashish, DARPA Unveils TRACTOR Initiative: Transforming Legacy C Code to Rust with AI, 2023 https://thecyberexpress.com/darpas-tractor/ [Accessed: 2025-01-12]

9: Shetty, M. and Jain, N. and Godbole, A. and Seshia, S. A. and Sen, K., Syzygy: Dual Code-Test C to (safe) Rust Translation using LLMs and Dynamic Analysis, 2024 [Accessed: 2025-01-12]

14: Wikipedia contributors, Graph database, 2025 https://en.wikipedia.org/wiki/Graph_database [Accessed: 2025-03-02]

15: Wikipedia contributors, Time series database, 2024 https://en.wikipedia.org/wiki/Time_series_database [Accessed: 2025-03-02]

16: Wikipedia contributors, Vector database, 2025 https://en.wikipedia.org/wiki/Vector_database [Accessed: 2025-03-02]

18: Wikipedia contributors, Blockchain-based database, 2025 https://en.wikipedia.org/wiki/Blockchain-based_database [Accessed: 2025-03-02]

21: Wikipedia contributors, NewSQL, 2025 https://en.wikipedia.org/wiki/NewSQL [Accessed: 2025-03-02]

22: Geeksforgeeks contributors, Introduction of DBMS (Database Management System), 2025 https://www.geeksforgeeks.org/introduction-of-dbms-database-management-system-set-1/ [Accessed: 2025-03-02]

24: , Slack Revenue and Usage Statistics (2025), https://www.businessofapps.com/data/slack-statistics/ [Accessed: 08-03-2025]

25: Stack Overflow, 2024 Developer Survey, 2024 https://survey.stackoverflow.co/2024 [Accessed: 2025-03-08]

26: Spotify, Q4 2024 Shareholder Deck, 2025 https://s29.q4cdn.com/175625835/files/doc_financials/2024/q4/Q4-2024-Shareholder-Deck-FINAL.pdf [Accessed: 2025-03-08]

27: Kumar, N., Discord Users & Market Share Statistics 2025 (Updated), 2025 https://www.demandsage.com/discord-statistics/ [Accessed: 2025-03-08]

28: , Microsoft Teams daily active users worldwide 2024, https://www.statista.com/statistics/1033742/worldwide-microsoft-teams-daily-and-monthly-users/ [Accessed: 08-03-2025]

29: usfhealthonline, What are Electronic Medical Records (EMRs)?, 2024 https://www.usfhealthonline.com/resources/health-informatics/what-are-electronic-medical-records-emr/ [Accessed: 2025-03-03]