الجمهورية العربية السورية
وزارة التعليم العالي والبحث العلمي
الجامعة الوطنية الخاصة
كلية الهندسة – قسم هندسة الحاسوب

مشروع تخرّج

# نبض — نظام السجلات الطبية الرقمية

## Nabd — Digital Medical Records System

منصّة صحّية قائمة على معيار FHIR HL7 مع دعم القرار السريري والذكاء الاصطناعي التنبؤي

**إعداد الطلاب:**

أمير المنصور  محمد خير الحوراني  ميزر عمري

**بإشراف:**

د. وسيم رمضان

**العام الدراسي ٢٠٢٥ – ٢٠٢٦**

# كلمة شكر

**نتقدّم بخالص الشكر والامتنان إلى كلّ من ساندنا وأسهم في إنجاز هذا المشروع...**

في البداية، نرفع شكرنا العميق إلى آبائنا وأمّهاتنا الذين لم يبخلوا علينا بدعائهم ودعمهم الدائم، فكانوا سندنا وملاذ آمالنا، ولا ننسى أصدقاءنا الأعزّاء الذين رافقونا في مسيرتنا الدراسية وكانوا خير عونٍ لنا في أوقات الجدّ والاجتهاد.

ونخصّ بالشكر والتقدير الدكتور **وسيم رمضان** الذي أشرف على مشروعنا برعايته الكريمة وأنار لنا الطريق بتوجيهاته القيّمة وملاحظاته الدقيقة طوال مراحل العمل، كما نعبّر عن شكرنا الكبير للدكتور **هاني الخطيب**، عميد كلية الهندسة المحترم، على دعمه المستمر وإسهامه في مسيرتنا، ولكلّ أساتذة قسم هندسة الحاسوب في الجامعة الوطنية الخاصة الذين لم يدّخروا جهداً في تزويدنا بالعلم والمعرفة.

نسأل الله أن يجعل هذا العمل لبنةً نافعة في خدمة القطاع الصحّي ووطننا الحبيب.

# الملخّص

تُعدّ السجلات الطبية حجر الأساس في تقديم رعاية صحّية آمنة وفعّالة؛ فهي الذاكرة التي تختزن تاريخ المريض من تشخيصاتٍ وأدويةٍ وفحوصٍ وإجراءات. غير أنّ هذه السجلات غالباً ما تكون موزّعة بين أنظمةٍ غير متوافقة لا يتبادل بعضها البيانات مع بعض، ما يضعف التنسيق بين مقدّمي الرعاية، ويؤدّي إلى تكرار الفحوص، وإلى قراراتٍ سريرية تُتّخذ أحياناً من دون صورةٍ كاملة عن حالة المريض. من هنا جاءت فكرة هذا المشروع: بناء نظامٍ حديث لإدارة السجلات الطبية الرقمية يعتمد على المعيار العالمي FHIR HL7 في إصداره الخامس (R5) لضمان قابلية التشغيل البيني (Interoperability) بين الأنظمة الصحية المختلفة.

يجمع النظام بين عدّة طبقاتٍ متكاملة: سجلٌّ طبي قائم على معيار FHIR لإدارة بيانات المرضى والتشخيصات والملاحظات والأدوية والإجراءات والمواعيد، مع حفظٍ كامل لتاريخ النسخ والتعديلات؛ ومحرّك دعم قرار سريري (Clinical Decision Support) قابل للتهيئة يُطلق تنبيهاتٍ للأطباء في موضع الرعاية وفق قواعد منطقية مرنة؛ وطبقة ذكاء اصطناعي تضمّ ثلاثة نماذج تنبّؤية تُقدّر مخاطر المريض انطلاقاً من بياناته الفعلية: خطر الإصابة بالسكري، وخطر أمراض القلب، وخطر إعادة الدخول إلى المشفى خلال ثلاثين يوماً. ويُضاف إلى ذلك سجلٌّ لربط نماذج ذكاء اصطناعي خارجية، وبوابة خدمة ذاتية للمريض، ودعم كامل للّغتين العربية والإنجليزية.

بُنيت الواجهات الخلفية باستخدام إطار العمل ASP.NET Core مع إطار Entity Framework Core وقاعدة بيانات PostgreSQL، في حين بُنيت الواجهة الأمامية باستخدام إطار Blazor WebAssembly. واعتمد النظام على خادم Keycloak لتوفير مصادقةٍ وتفويضٍ آمنين وفق بروتوكولَي OAuth2 وOpenID Connect مع نطاقات صلاحيات على غرار SMART on FHIR. أمّا نماذج الذكاء الاصطناعي فقد دُرّبت بلغة Python ثمّ صُدّرت إلى صيغة ONNX لتُستهلَك داخل الخادم بأداء استدلالٍ عالٍ.

نهدف من خلال هذا المشروع إلى تحسين جودة الرعاية الصحية ورفع كفاءة العمل الطبي عبر توحيد البيانات وفق معيارٍ عالمي، وتقديم دعمٍ ذكي للقرار السريري في موضع الرعاية، وتمكين المريض من الاطّلاع على سجلّه الصحّي بأمان.

# Abstract

Medical records are the foundation of safe and effective healthcare, yet they are often scattered across incompatible systems that do not exchange data — weakening coordination between care providers, causing repeated tests, and delaying clinical decisions made without a complete picture of the patient. This project addresses that challenge by building Nabd, a modern Digital Medical Records System, based on the international HL7 FHIR (Release 5) standard to guarantee interoperability between healthcare systems.

The system combines several integrated layers: a FHIR-based clinical record that manages patients, conditions, observations, medications, procedures and appointments with full version history; a configurable Clinical Decision Support (CDS) engine that alerts clinicians at the point of care through a flexible rule engine; and an artificial-intelligence layer of three predictive models that estimate patient risk from real FHIR data — diabetes risk, cardiovascular risk, and 30-day hospital readmission risk. It also adds a registry for external AI models, a patient self-service portal, and full Arabic/English localization.

The backend was built with ASP.NET Core, Entity Framework Core and PostgreSQL, while the frontend was built with Blazor WebAssembly. Authentication and authorization rely on Keycloak using OAuth2 / OpenID Connect with SMART-on-FHIR–style scopes. The AI models were trained in Python and exported to the ONNX format to be served inside the API with high inference performance.

Through this project we aim to improve the quality of healthcare and the efficiency of clinical work by unifying data under a global standard, providing intelligent decision support at the point of care, and empowering patients to securely access their own health records.

# الفهرس


كلمة شكر ..... 2
الملخّص ..... 3
Abstract ..... 4
الفهرس ..... 5
الفصل الأول – مقدمة عامة ..... 8
١-١ التحديات ..... 9
١-٢ الهدف ..... 9
١-٣ الفوائد (ممكن حذفها) ..... 10
١-٤ أهمّ ميزات المشروع ..... 10
١-٥ الجهات المستفيدة من المشروع ..... 11
١-٦ الدراسة المرجعية ..... 12
١-٧ المساهمات ..... 13
الفصل الثاني – الأدوات والتقنيات البرمجية ..... 15
٢-١ تقنيات البرمجة الخلفية Backend ..... 15
٢-١-١ لغة C# ..... 15
٢-١-٢ إطار العمل ASP.NET Core ..... 15
٢-١-٣ إطار العمل Entity Framework Core ..... 16
٢-١-٤ قاعدة البيانات PostgreSQL ..... 16
٢-١-٥ معيار Hl7Firely ومكتبة FHIR ..... 16
٢-١-٦ نظام المصادقة Keycloak ..... 17
٢-١-٧ أدوات مساندة (Serilog وData Protection وSwagger) ..... 17
٢-٢ تقنيات الواجهة الأمامية Frontend ..... 17
٢-٢-١ إطار العمل Blazor WebAssembly ..... 17
٢-٢-٢ لغتا HTML وCSS ..... 18
٢-٢-٣ إطار العمل Bootstrap ..... 18
٢-٢-٤ الترجمة وتعدّد اللغات (i18n) ..... 18
٢-٢-٥ بنية المكوّنات ونمط الموارد الموحّد ..... 19
٢-٣ أدوات الذكاء الاصطناعي ..... 19
٢-٣-١ لغة Python ومكتبة scikit-learn ..... 19
٢-٣-٢ صيغة ONNX وأداة skl2onnx ومحرّك ONNX Runtime ..... 19

٢-٣-٣ نماذج المخاطر الثلاثة بالتفصيل ..... 20
٢-٤ البنية العامة للنظام ..... 21
٢-٥ الخلاصة ..... 22
الفصل الثالث – متطلبات المشروع والمخططات ..... 23
٣-١ البنية التحتية ..... 23
٣-١-١ خادم الويب Web Server ..... 23
٣-١-٢ قاعدة بيانات مركزية ..... 23
٣-١-٣ خادم المصادقة Keycloak ..... 23
٣-١-٤ خدمة معلومات الأدوية ..... 23
٣-١-٥ شبكة اتصال محمية ..... 23
٣-٢ المتطلبات الوظيفية (للحذف) ..... 24
٣-٣ المتطلبات غير الوظيفية (للحذف) ..... 24
٣-٤ الأمان والصلاحيات ..... 25
٣-٥ المخططات وقاعدة البيانات ..... 26
٣-٣-١ الفاعلون Actors ..... 26
٣-٣-٢ مخططات حالات الاستخدام Use Case Diagram ..... 26
٣-٣-٣ مخطّط قاعدة البيانات Entity Relation Diagram ..... 30
٣-٣-٤ مخطّطات التسلسل لأهمّ الآليات ..... 32
٣-٦ واجهة النظام البرمجية (API) ..... 34
٣-٦-١ مبدأ التصميم ..... 34
٣-٦-٢ الواجهة الموحّدة لموارد FHIR ..... 35
٣-٦-٣ عائلات المسارات المتخصّصة ..... 35
٣-٦-٤ المصادقة والاصطلاحات ..... 36
٣-٦-٥ مثال تطبيقي ..... 37
٣-٦-٦ التوثيق التفاعلي والمرجع الكامل ..... 37
الفصل الرابع – التطبيق العملي ..... 39
٤-١ واجهات الزائر والمصادقة ..... 39
٤-٢ واجهات الطبيب/الممارس ..... 42
٤-٣ واجهات مدير المؤسسة ..... 51
٤-٤ واجهات مدير النظام ..... 54
٤-٥ بوابة المريض ..... 56

٦-٤ لوحة الرؤى الذكية AI Insights ..... 58
الفصل الخامس – الخاتمة والآفاق المستقبلية ..... 61
الآفاق المستقبلية ..... 61
إضافة منظومة اختبارات آلية ..... 61
النشر السحابي والحاويات ..... 61
ربط مصدر بيانات الأدوية الحيّ ..... 61
توسيع نماذج الذكاء الاصطناعي ..... 61
تطوير تطبيق موبايل ..... 62
تعزيز الأمان وحماية الخصوصية ..... 62
توسيع التشغيل البيني ..... 62
الفصل السادس – الملحق ..... 63
٦-١ روابط المشروع ..... 63
٦-٢ ملخّص حزمة التقنيات ..... 63
٦-٣ موارد FHIR المدعومة ..... 63
٦-٤ مسرد المصطلحات ..... 64
٦-٥ تشغيل النظام (مختصر) ..... 64
المراجع ..... 66

# الفصل الأول – مقدمة عامة

يشهد قطاع الرعاية الصحية تحوّلاً رقمياً متسارعاً يَعِد بتحسين جودة الخدمة ورفع كفاءة العمل الطبي، غير أنّ كثيراً من المؤسسات الصحية ما زالت تعتمد على أنظمةٍ منعزلة تخزّن بيانات المريض بصِيَغٍ خاصّة لا يفهمها سواها. ويؤدّي هذا التشتّت إلى صعوبة تبادل المعلومات بين المشافي والعيادات والمخابر، وإلى تكرار الفحوص وإهدار الموارد، وإلى قراراتٍ سريرية تُتّخذ أحياناً من دون صورةٍ كاملة عن تاريخ المريض. كما تبقى الرعاية في كثيرٍ من الأنظمة ذات طابعٍ تفاعلي (Reactive) أكثر منه استباقي (Proactive)؛ إذ يُكتشَف ارتفاع خطورة حالة المريض في وقتٍ متأخّر بسبب غياب أدواتٍ تنبّؤية تعمل في موضع الرعاية وتساعد الطبيب على اتخاذ القرار المناسب في الوقت المناسب.

ولمواجهة هذه التحديات، صمّمنا «نبض» — نظام السجلات الطبية الرقمية: منصّةٌ متكاملة تعتمد على المعيار العالمي FHIR HL7 في إصداره الخامس [١] لتوحيد تمثيل البيانات الصحية وضمان قابلية تبادلها بين الأنظمة المختلفة، وتضيف فوق هذا السجلّ محرّك دعمٍ للقرار السريري قابلاً للتهيئة وفق مواصفة CDS Hooks [١٣]، وطبقة ذكاء اصطناعي تُقدّر مخاطر المريض من بياناته الفعلية، وبوابةً يطّلع المريض من خلالها على سجلّه الصحّي. وقد روعيَ في التصميم أن يكون النظام قابلاً للتوسّع والصيانة، وأن يلتزم بمعايير الأمان وحماية الخصوصية المناسبة للبيانات الصحية الحسّاسة.

ويأتي اختيار هذا المجال انطلاقاً من أهمّيته البالغة؛ فالبيانات الصحية من أكثر البيانات حساسيةً وقيمة، وأيّ تحسينٍ في إدارتها ينعكس مباشرةً على جودة حياة الناس وسلامتهم. كما أنّ تبنّي معيارٍ عالمي مثل FHIR يجعل النظام جاهزاً للتكامل مع المنظومة الصحية الأوسع بدل أن يبقى جزيرةً معزولة، وهو ما يميّز هذا المشروع عن الأنظمة التقليدية المغلقة.

ويتميّز النظام كذلك بأنّه لا يكتفي بتخزين البيانات وعرضها، بل يضيف فوقها طبقاتٍ من الذكاء: محرّكُ قواعدَ يحوّل المعرفة الطبية إلى تنبيهاتٍ قابلة للتنفيذ، ونماذجُ تنبّؤية تقرأ سجلّ المريض وتقدّر مخاطره، وسجلٌّ يتيح ربط نماذج ذكاءٍ خارجية متخصّصة. وبذلك ينتقل النظام بالرعاية من الطابع التفاعلي إلى الطابع الاستباقي.

يتناول هذا الفصل عرضاً عامّاً للمشروع من حيث التحديات التي يعالجها وأهدافه وفوائده وأهمّ ميزاته والجهات المستفيدة منه، تمهيداً للفصول التالية التي تستعرض الأدوات والتقنيات المستخدمة، ثمّ متطلّبات المشروع ومخطّطاته، ثمّ الجانب العملي والواجهات، وصولاً إلى الخاتمة والآفاق المستقبلية.

## ١-١ التحديات

تنطلق فكرة المشروع من جملةٍ من التحديات الواقعية التي تواجه المنظومة الصحية، ويمكن تلخيص أبرزها بما يأتي:

تشتّت البيانات وانعدام التشغيل البيني: تُخزَّن بيانات المريض في أنظمةٍ متفرّقة وبصِيَغٍ مغلقة، فلا ينتقل سجلّ المريض معه بين مقدّمي الرعاية، ما يحرم الطبيب من رؤيةٍ شاملة ويؤدّي إلى تكرار الفحوص والإجراءات.

ضعف الدعم الاستباقي للقرار: تفتقر كثيرٌ من الأنظمة إلى أدواتٍ تنبّه الطبيب آنياً إلى المخاطر (كجرعةٍ دوائية زائدة أو تداخلٍ دوائي أو ارتفاع خطر إعادة الدخول إلى المشفى)، فتبقى الأخطاء الطبية احتمالاً قائماً.

صعوبة تتبّع تاريخ السجل: في غياب آليةٍ منضبطة لحفظ النسخ، يصعب معرفة من غيّر بيانات المريض ومتى وماذا كانت القيمة السابقة، وهو أمرٌ جوهري للمساءلة والتدقيق الطبي.

غياب مشاركة المريض: كثيراً ما يُحرَم المريض من الاطّلاع على سجلّه الصحّي أو متابعة مواعيده وأدويته بنفسه، ما يضعف انخراطه في رعايته الخاصة.

تحديات الأمان والخصوصية: تتطلّب البيانات الصحية مستوىً عالياً من الحماية في التخزين والنقل، وتحكّماً دقيقاً بالوصول بحسب دور كلّ مستخدم وصلاحياته.

## ١-٢ الهدف

يهدف المشروع إلى بناء نظامٍ حديثٍ ومتكامل لإدارة السجلات الطبية الرقمية يوحّد تمثيل البيانات وفق معيارٍ عالمي، ويوفّر دعماً ذكياً للقرار السريري في موضع الرعاية، ويرفع كفاءة العمل الطبي وجودته. ويمكن تفصيل هذا الهدف العام في الأهداف الفرعية الآتية:

١. بناء سجلٍّ طبي قائم على معيار FHIR R5 يدعم العمليات الأساسية من إنشاءٍ وقراءةٍ وتحديثٍ وحذفٍ وبحثٍ وحفظٍ لتاريخ النسخ والتحقّق من صحّة الموارد.
٢. توفير نظام مصادقةٍ وتفويضٍ آمن قائم على الأدوار ونطاقات الصلاحيات، يضمن أن يرى كلّ مستخدمٍ ما يخصّه فقط.
٣. تقديم دعمٍ ذكي للقرار السريري عبر محرّك قواعد مرنٍ يُطلق بطاقات تنبيهٍ مبنية على بيانات المريض ومعرفة الأدوية ومخاطره.
٤. دمج نماذج ذكاءٍ اصطناعي تنبّؤية تعمل على بيانات المريض الحقيقية لعرض تقديراتٍ للمخاطر تساعد الكادر الطبي، مع إتاحة ربط نماذج خارجية.
٥. تمكين المريض من الاطّلاع الآمن على سجلّه الصحّي عبر بوابة خدمةٍ ذاتية، ودعم الوصول بلغتين عبر واجهةٍ ثنائية اللغة (عربي/إنجليزي).

## ١-٣ الفوائد (ممكن حذفها)

- توحيد البيانات الصحية وفق معيارٍ عالمي يضمن قابلية تبادلها بين الأنظمة المختلفة.
- رفع جودة القرار السريري وتقليل الأخطاء الطبية عبر التنبيهات الاستباقية في موضع الرعاية.
- الكشف المبكر عن المرضى الأعلى خطورة بفضل نماذج المخاطر، ما يتيح التدخّل قبل تفاقم الحالة.
- تحسين الكفاءة التشغيلية وتوفير الوقت والجهد على الكادر الطبي والإداري عبر أتمتة العمل وتنظيمه.
- تعزيز الشفافية والمساءلة من خلال حفظ تاريخ النسخ الكامل لكلّ مورد طبي.
- تمكين المريض ورفع انخراطه في رعايته الصحية عبر بوابة الخدمة الذاتية.
- إرساء أساسٍ تقني قابلٍ للتوسّع نحو تطبيقاتٍ صحية أوسع مستقبلاً.

## ١-٤ أهمّ ميزات المشروع

يقدّم النظام مجموعةً غنية من الميزات التي تغطّي الجوانب السريرية والإدارية الذكية، نعرض أبرزها فيما يأتي:

سجلّ طبي قائم على معيار FHIR R5[١]: يدير النظام طيفاً واسعاً من الموارد السريرية وفق المعيار، منها: المريض (Patient)، والممارس (Practitioner) ودوره (PractitionerRole)، والمؤسسة (Organization)، والموقع (Location)، والزيارة (Encounter)، والموعد (Appointment)، والحالة المرضية (Condition)، والملاحظة (Observation)، والإجراء (Procedure)، وطلب الدواء (MedicationRequest)، وفرط الحساسية (AllergyIntolerance)، وطلب الخدمة (ServiceRequest).

حفظ النسخ وتاريخ التعديلات: يُحفَظ لكلّ مورد تاريخه الكامل من النسخ، مع دعم الحذف الناعم (Soft Delete)، بحيث يمكن استعراض أيّ نسخةٍ سابقة ومعرفة وقت تعديلها.

البحث والتحقّق وفق المعيار: يدعم النظام البحث بباراماترات قياسية عبر مفهرسات الموارد، والتحقّق من صحّة الموارد مقابل مواصفة FHIR R5 باستخدام مكتبة Firely [٢].

مصادقة وتفويض آمنان: يعتمد النظام خادم Keycloak [٨] لإصدار رموز الدخول (JWT) وإدارة المستخدمين والأدوار، مع تحكّمٍ بالوصول قائم على الأدوار وعلى نطاقات الصلاحيات (System/User/Patient) على غرار SMART on FHIR [١٩]، وفق بروتوكولَي OAuth2/OpenID Connect [١٨].

محرّك دعم القرار السريري (CDS): يوفّر مساراتٍ متوافقة مع CDS Hooks [١٣]، ومحرّك قواعد يعتمد منطق JSON-Logic [٢١]، وورشةً لتأليف القواعد (CDS Admin) مع قوالب جاهزة ومعاينةٍ وتحقّقٍ ونشرٍ مُصدَّر النسخ وتفعيل للقواعد، إضافةً إلى تغذيةٍ حيّة للتنبيهات.

نماذج ذكاء اصطناعي للتنبّؤ بالمخاطر: ثلاثة نماذج بصيغة ONNX [٩] تُقدّر المخاطر من سجلّ المريض: خطر السكري، وخطر أمراض القلب، وخطر إعادة الدخول إلى المشفى خلال ثلاثين يوماً، مع تعويض القيم الناقصة بالوسيط والإشارة إلى ذلك.

سجلّ نماذج الذكاء الاصطناعي الخارجية: يتيح للمدير تسجيل نماذج ذكاء اصطناعي مستضافة خارجياً عبر رابط آمن (HTTPS) مع حفظ أسرار المصادقة مشفّرةً عبر ASP.NET Data Protection [٢٥]، وتشغيلها على بيانات مريضٍ معيّن وقراءة قرارها.

معرفة الأدوية: خدمةٌ مستقلّة لمعلومات الأدوية (جرعات وسلامة) تُستهلَك من محرّك دعم القرار، مع إمكانية الاعتماد على مزوّدٍ مُجمَّع أو مصدرٍ حيّ مثل RxNorm [٢٢].

بوابة المريض ووثائقه: بوابة خدمةٍ ذاتية يتطلع المريض من خلالها على ملخّص سجلّه ومواعيده وأدويته ويدير ملفّه الشخصي، إضافةً إلى إدارة الوثائق السريرية المرفوعة لكلّ مريض.

لوحات معلوماتٍ وتحليلات: لوحة عملٍ رئيسية لكلّ دور، وقائمة مراقبةٍ تنبّؤية للمرضى الأعلى خطورة، ولوحة «الرؤى الذكية» (AI Insights) التي تقدّم نظرةً على مستوى المرضى وانتشار الحالات المرضية.

واجهة ثنائية اللغة (عربي/إنجليزي): دعمٌ كامل للترجمة في زمن التشغيل مع تبديلٍ فوري للّغة واتجاه الكتابة (RTL/LTR)، يغطّي مئات النصوص في الواجهة.

## ١-٥ الجهات المستفيدة من المشروع

المؤسسات الصحية (أطباء وإداريون): يرفع النظام كفاءة العمل السريري والإداري عبر توحيد البيانات وأتمتة المهام وتقليل الأخطاء البشرية، ويمنح المدراء رقابةً وإشرافاً أفضل عبر لوحات المعلومات والتحليلات.

المرضى: يعزّز ثقة المريض بالمنظومة الصحية عبر تمكينه من الاطّلاع على سجلّه ومتابعته، ويوسّع نطاق الوصول إلى الخدمة الصحية.

وزارات الصحّة ومنظّمات الرعاية: يوفّر أساساً موحَّداً للبيانات يدعم التخطيط واتخاذ القرار على مستوى المنظومة، ويسهّل تكامل الأنظمة بفضل الالتزام بمعيارٍ عالمي.

الباحثون والمطوّرون: يقدّم النظام بنيةً قياسية وقابلة للتوسّع يمكن البناء عليها في أبحاث المعلوماتية الصحية وتطوير خدماتٍ صحية جديدة.

## ١-٦ الدراسة المرجعية

لم يأتِ هذا المشروع في فراغ؛ فهناك أنظمةٌ قائمة لإدارة السجلات الطبية، منها المفتوح المصدر ومنها التجاري، ولكلٍّ منها فلسفته ونطاقه. ونستعرض فيما يأتي أبرز هذه الأنظمة قرباً من نظامنا، تمهيداً لمقارنةٍ تبيّن ما يشترك فيه معها وما يتميّز به عنها:

OpenMRS: نظام سجلات طبية مفتوح المصدر واسع الانتشار، وُجّه أساساً إلى المؤسسات الصحية في الدول النامية، ويقوم على بنيةٍ وحدوية (Modules) تتيح توسيعه. يعتمد نموذج بياناتٍ داخلياً خاصاً به، ويوفّر دعم معيار FHIR عبر وحدة ترجمةٍ مستقلّة تحوّل بين نموذجه الداخلي والمعيار، لا عبر تخزينٍ أصيل للموارد. ويقدّم تنبيهاتٍ وتذكيراتٍ لدعم القرار السريري [٣٠].

OpenEMR: من أوسع أنظمة السجلات الطبية المفتوحة انتشاراً، وحاصلٌ على اعتماد ONC الأمريكي، ويضمّ إدارةً للعيادة وفوترةً ووصفاتٍ إلكترونية وبوابة مريض. ويقدّم واحداً من أكمل تطبيقات FHIR R4 بين الأنظمة المفتوحة مع دعم SMART on FHIR وUS Core، غير أنّ تخزينه الداخلي ليس أصيلاً وفق المعيار، ودعمه للقرار السريري يقوم على قواعد وتذكيراتٍ خاصّة به لا على معيار CDS Hooks [٣١].

HAPI FHIR: التطبيق المرجعي المفتوح المصدر لمعيار FHIR بلغة Java، ويدعم إصداراته من DSTU2 حتى R5 مع تخزينٍ أصيل للموارد وواجهةٍ برمجية كاملة وتاريخ نسخ. لكنّه لبنةٌ تحتية لا نظامٌ سريري متكامل: لا يتضمّن واجهة مستخدمٍ مدمجة ولا دعماً للقرار السريري ولا نماذج ذكاءٍ اصطناعي، وتُبنى هذه الطبقات فوقه [٣٢].

Medplum: منصّةٌ حديثة مفتوحة المصدر (رخصة Apache 2.0) موجّهة للمطوّرين، تخزّن البيانات تخزيناً أصيلاً وفق FHIR R4 وتوفّر مصادقةً وتحكّماً بالوصول ومكوّنات واجهةٍ جاهزة، إضافةً إلى «Bots» لتنفيذ منطقٍ خلفي عند تغيّر الموارد. وهي أقرب إلى أساسٍ يُبنى عليه تطبيقٌ صحّي منها إلى نظامٍ سريري جاهز، ولا تقدّم دعم قرارٍ معيارياً ولا نماذج تنبّؤ جاهزة [٣٣].

Epic و Oracle Health (Cerner): النظامان التجاريان المهيمنان على سوق السجلات الصحية في العالم، ويقدّمان منظومةً شاملة تغطّي المستشفى بكامله، مع واجهات FHIR R4 ودعمٍ لمعيار CDS Hooks وبوّاباتٍ للمرضى (MyChart وHealtheLife) ونماذج تنبّؤية مدمجة مثل مؤشّر التدهور (Deterioration Index). غير أنّهما مغلقا المصدر وباهظا الكلفة، ويتطلّبان بنيةً تحتية وفرق تشغيلٍ كبيرة، ما يجعلهما بعيدَين عن متناول المؤسسات الصغيرة والمتوسطة [٣٤][٣٥].

ويلخّص الجدول الآتي مقارنةً بين نظام نبضوهذه الأنظمة وفق أبرز المعايير التي تهمّ مشروعنا:

| المعيار | نبض (نظامنا) | OpenMRS | OpenEMR | HAPI FHIR | Medplum | Epic / Cerner |
|---|---|---|---|---|---|---|
| الترخيص والكلفة | مفتوح – أكاديمي | مفتوح | مفتوح | مفتوح | مفتوح (Apache 2.0) | تجاري مغلق مرتفع |
| إصدار FHIR المدعوم | R5 | R4 | R4 | DSTU2 – R5 | R4 | R4 |
| تخزين أصيل لموارد FHIR | ✓ | ✗ | ✗ | ✓ | ✓ | ✗ |
| واجهة برمجية قياسية (API) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| تاريخ النسخ الكامل | ✓ | جزئي | جزئي | ✓ | ✓ | ✓ |
| دعم قرار سريري (تنبيهات) | ✓ | ✓ | ✓ | ✗ | جزئي (Bots) | ✓ |
| توافق معيار CDS Hooks | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ |
| ورشة تأليف قواعد رسومية | ✓ | جزئي | جزئي | ✗ | ✗ | جزئي |
| نماذج تنبّؤ بالمخاطر مدمجة | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ |
| تسجيل نماذج AI خارجية | ✓ | ✗ | ✗ | ✗ | جزئي | جزئي |
| بوابة المريض | ✓ | جزئي | ✓ | ✗ | ✓ (تُبنى) | ✓ |
| نشر محلي بلا كلفة ترخيص | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ |

*جدول ١-١: مقارنة بين نظام نبضوأبرز الأنظمة المشابهة.*

ولا يدّعي هذا العرض أنّ نظامنا يفوق الأنظمة الناضجة المذكورة؛ فهي تتفوّق عليه بنطاقٍ أوسع يشمل المخابر والفوترة والوصفات الإلكترونية، وباعتماداتٍ رسمية وسنواتِ تشغيلٍ بأحجامٍ ضخمة وفرق دعمٍ كبيرة، في حين أنّ «نبض»مشروعٌ أكاديمي في نطاقٍ محدود. غير أنّ ما يميّزه هو اجتماع عناصر لا تجتمع في أيٍّ من هذه الأنظمة ضمن نظامٍ واحد مفتوح: تخزينٌ أصيل وفق أحدث إصدارات المعيار (FHIR R5)، ومحرّك قرارٍ سريري متوافق مع CDS Hooks مع ورشةٍ رسومية لتأليف القواعد، ونماذج تنبّؤ بالمخاطر مدمجة مع إمكانية ربط نماذج خارجية، وبوابة مريض، وواجهةٌ معرَّبة بالكامل — وكلّ ذلك قابلٌ للنشر محلياً دون كلفة ترخيص.

## ١-٧ المساهمات

لا يقتصر أثر هذا المشروع على ما يقدّمه من ميزاتٍ تقنية، بل يمتدّ إلى ما يضيفه فعلياً إلى واقع العمل الصحّي والمستفيدين منه. ونوجز فيما يأتي أبرز مساهمات النظام:

سجلٌّ طبي موحَّد للمريض: يجمع النظام تاريخ المريض الصحّي في مكانٍ واحد بدل تشتّته بين الأنظمة المتفرّقة، فيصبح السجلّ متاحاً لمن يحتاجه من مقدّمي الرعاية الصحية أينما كان المريض.

واجهة برمجة التطبيقات (API): يوفّر النظام واجهةً برمجية مبنيّة على معيارٍ عالمي تتيح للأنظمة والتطبيقات الصحّية الأخرى الاتّصال به وتبادل البيانات معه، فلا يبقى النظام جزيرةً منعزلة بل يصبح أساساً يمكن للآخرين البناء عليه (يُفصَّل توثيقها في القسم ٣-٦).

سجلات الطبيب وأدوات عمله: يمنح النظام الطبيب مساحة عملٍ منظّمة يتابع فيها مرضاه ومواعيده ويسجّل تشخيصاته ووصفاته، ما يوفّر وقته ويقلّل الأخطاء الناتجة عن العمل اليدوي.

مساعدة الطبيب لحظة اتّخاذ القرار: يقرأ النظام سجلّ المريض وينبّه الطبيب إلى ما قد يفوته من مخاطر أثناء عمله لا بعده، فيرفع أمان الرعاية بدل أن يكتفي بتوثيقها.

تمكين المريض من سجلّه: يتيح النظام للمريض الاطّلاع على سجلّه الصحّي ومتابعة مواعيده وأدويته بنفسه، ما يعزّز ثقته بالمنظومة الصحّية وانخراطه في رعايته.

أساسٌ صحّي رقمي قابل للتوسّع: يقدّم المشروع نموذجاً عملياً يمكن البناء عليه وتوسيعه على مستوى المؤسسات الصحّية، بما يخدم التحوّل الرقمي في القطاع الصحّي محلياً.

# الفصل الثاني – الأدوات والتقنيات البرمجية

نستعرض في هذا الفصل الأدوات والتقنيات البرمجية التي اعتُمدت في تطوير نظام السجلات الطبية الرقمية، موضّحين دور كلٍّ منها وأبرز ميزاتها وكيفية استثمارها لتحقيق أهداف المشروع. ولأنّ النظام يتألّف من عدّة خدماتٍ متكاملة (واجهة برمجية رئيسية، وواجهة أمامية، وخدمة معلومات أدوية، وطبقة ذكاء اصطناعي)، فقد وُظّفت مجموعةٌ متنوّعة من التقنيات نقسّمها إلى تقنيات البرمجة الخلفية، وتقنيات الواجهة الأمامية، وأدوات الذكاء الاصطناعي، ثمّ نختم بنظرةٍ على البنية العامة للنظام.

## ٢-١ تقنيات البرمجة الخلفية Backend

### ٢-١-١ لغة C#

لغة C# لغةٌ برمجية حديثة كائنية التوجّه طوّرتها شركة Microsoft، تتميّز بأمان الأنماط (Type Safety) ودعمها الواسع للبرمجة غير المتزامنة (Asynchronous) التي ترفع كفاءة معالجة الطلبات المتزامنة، وتعمل ضمن منصّة .NET [٤][٢٧].

البرمجة الكائنية (OOP): تدعم اللغة الوراثة والتغليف وتعدّد الأشكال، ما يتيح كتابة شيفرةٍ منظّمة قابلة لإعادة الاستخدام ويُسهّل تطوير الأنظمة المعقّدة وصيانتها.

التكامل مع .NET.: تعمل ضمن منصّة .NET. التي توفّر مكتباتٍ غنيّة للتعامل مع البيانات والتشفير وإدارة الجلسات والشبكة.

القابلية للتوسعة: لغةٌ قوية وآمنة توفّر بيئةً مرنة لتوسيع النظام مستقبلاً وربطه بتطبيقاتٍ أخرى.

الفائدة في المشروع: كتابة منطق الخادم بكفاءةٍ ومرونة وتنظيم الشيفرة وفق مبادئ OOP ضمن إطار ASP.NET Core.

### ٢-١-٢ إطار العمل ASP.NET Core

ASP.NET Core إطار عملٍ مفتوح المصدر وعابر للمنصّات من Microsoft لبناء تطبيقات الويب وخدمات الويب (Web APIs) عالية الأداء [٣].

السرعة والكفاءة: من أسرع أطر العمل لتطوير تطبيقات الويب بفضل تحسيناته المستمرّة على الأداء وطرق المعالجة.

دعم RESTful APIs: أدواتٌ متكاملة لبناء واجهاتٍ برمجية تتبادل البيانات بطريقةٍ منظّمة وموثوقة؛ وقد بُنيت عليها مسارات FHIR في النظام.

الأمان والحقن: يدعم المصادقة والتفويض وحَقْن الاعتماديات (Dependency Injection) والوسطاء (Middleware) والحماية من الهجمات الشائعة.

الفائدة في المشروع: بناء الواجهات البرمجية التي تتبع نمط REST وتلتزم بمعيار FHIR في مساراتها واستجاباتها.

### ٢-١-٣ إطار العمل Entity Framework Core

Entity Framework Core إطار عملٍ لربط الكائنات بالعلاقات (ORM) يتيح التعامل مع قاعدة البيانات عبر كائناتٍ برمجية بدل كتابة استعلاماتٍ يدوية، ويدعم منهج «الشيفرة أولاً» (Code-First) [٥].

تحويل البيانات: يحوّل الجداول إلى كائناتٍ (Entities) ويبسّط عمليات القراءة والإضافة والتحديث والحذف.

الهجرات (Migrations): آليةٌ لإنشاء بنية قاعدة البيانات وتحديثها تلقائياً عند تغيّر نموذج البيانات، ما يسهّل التطوير والصيانة.

دعم LINQ: طريقةٌ مرنة وآمنة لاسترجاع البيانات باستخدام C# مباشرة.

الفائدة في المشروع: إدارة الكيانات والعلاقات وفهارس البحث وجداول دعم القرار بسهولةٍ ومن دون استعلاماتٍ يدوية.

### ٢-١-٤ قاعدة البيانات PostgreSQL

PostgreSQL نظام إدارة قواعد بياناتٍ علائقي مفتوح المصدر يتميّز بالموثوقية والالتزام بالمعايير ودعمه لأنواع بياناتٍ متقدّمة مثل JSONB [٦].

الموثوقية والمعاملات: دعمٌ كامل للمعاملات (ACID) يضمن سلامة البيانات الحسّاسة.

تخزين JSONB: تخزينٌ ثنائي فعّال لبنى JSON مع إمكانية الفهرسة، وقد استُثمر لتخزين موارد FHIR بصيغتها الأصلية.

الأداء والتوسّع: أداءٌ عالٍ مع دعمٍ للفهارس المتقدّمة والاستعلامات المعقّدة.

الفائدة في المشروع: تخزين موارد FHIR وتاريخ نسخها وفهارس البحث وقواعد دعم القرار بطريقةٍ منظّمة وآمنة.

### ٢-١-٥ معيار HL7 FHIR ومكتبة Firely

يُعدّ HL7 FHIR (Fast Healthcare Interoperability Resources) المعيار الحديث لتبادل البيانات الصحية، إذ يمثّل المفاهيم السريرية على شكل «موارد» (Resources) لكلٍّ منها بنيةٌ

محدّدة وواجهةٌ برمجية وفق نمط REST [١]. وقد اعتمدنا الإصدار الخامس R5 عبر مكتبة Firely .NET SDK التي توفّر نماذج الموارد وأدوات التحقّق والتسلسل [٢].

قابلية التشغيل البيني: تمثيلٌ موحَّد للبيانات يضمن تبادلها بين الأنظمة الصحية المختلفة.

موارد قياسية: نماذجٌ جاهزة لكلّ مفهومٍ سريري (مريض، حالة، ملاحظة، دواء...) مع حقولٍ وترميزاتٍ معيارية (SNOMED CT [٢٩]، LOINC [٢٠]...).

التحقّق والتسلسل: أدواتٌ للتحقّق من مطابقة الموارد للمواصفة وتحويلها من/إلى JSON بدقّة.

الفائدة في المشروع: توحيد تمثيل البيانات الصحية وضمان قابلية تبادلها، وهو حجر الأساس الذي بُني عليه سجلّ النظام.

### ٢-١-٦ نظام المصادقة Keycloak

Keycloak حلٌّ مفتوح المصدر لإدارة الهويّة والوصول، يدعم بروتوكولَي OAuth2 وOpenID Connect [٨][١٨].

إدارة المستخدمين والأدوار: إدارةٌ مركزية للحسابات والأدوار والمجالات (Realms) والعملاء (Clients).

رموز الدخول JWT: إصدار رموز دخولٍ موقّعة يتحقّق منها الخادم في كلّ طلب.

تدفّق آمن: دعم تدفّق Authorization Code مع PKCE المناسب لتطبيقات الواجهة الأمامية.

الفائدة في المشروع: تطبيق تحكّمٍ بالوصول قائم على الأدوار وعلى نطاقات الصلاحيات على غرار SMART on FHIR.

### ٢-١-٧ أدوات مساندة (Serilog وData Protection وSwagger)

اعتمد النظام أدواتٍ مساندة ترفع جودة التشغيل والأمان والتوثيق.

Serilog [٢٤]: تسجيلٌ مُهيكَل للأحداث يسهّل التشخيص والمتابعة.

ASP.NET Data Protection [٢٥]: تشفيرٌ للأسرار الحسّاسة عند التخزين، وقد استُخدم لحماية أسرار نماذج الذكاء الاصطناعي الخارجية.

Swagger / OpenAPI [٢٦]: توثيقٌ تفاعلي للواجهة البرمجية يسهّل اختبارها وتكاملها؛ ويُفصَّل توثيق واجهة النظام البرمجية في القسم ٣-٦-٦.

## ٢-٢ تقنيات الواجهة الأمامية Frontend

### ٢-٢-١ إطار العمل Blazor WebAssembly

Blazor WebAssembly إطار عملٍ من Microsoft يتيح بناء تطبيقات ويبٍ تفاعلية تعمل ضمن المتصفّح باستخدام لغة C# بدلاً من JavaScript، عبر تقنية WebAssembly [٧][٢٨].

توحيد اللغة: مشاركة النماذج والمنطق بين الخادم والواجهة بلغةٍ واحدة، ما يقلّل التكرار والأخطاء.

مكوّنات قابلة لإعادة الاستخدام: بناء الواجهة من مكوّنات (Components) منظّمة وقابلة للتركيب.

العمل ضمن المتصفّح: تنفيذٌ من جهة العميل يقلّل الحِمل على الخادم ويمنح تجربةً تفاعلية.

الفائدة في المشروع: بناء واجهةٍ سريرية موحّدة (مساحة العمل، ورشة دعم القرار، الرؤى الذكية، بوابة المريض) تتشارك النماذج مع الخادم.

### ٢-٢-٢ لغتا HTML وCSS

HTML لغة توصيف بنية صفحات الويب، وCSS لغة تنسيق مظهرها، وتُستخدمان أساساً لبناء عناصر الواجهة وتنسيقها ضمن مكوّنات Blazor.

هيكلة المحتوى: تنظيم عناصر الصفحة بشكلٍ واضح وسهل التصفّح.

التصميم المتجاوب: تنسيقاتٌ تتكيّف مع أحجام الشاشات المختلفة.

الدعم المرئي للاتجاهين: تنسيقاتٌ تدعم الكتابة من اليمين إلى اليسار (RTL) ومن اليسار إلى اليمين (LTR).

### ٢-٢-٣ إطار العمل Bootstrap

Bootstrap إطار عملٍ شائع لتصميم واجهاتٍ متجاوبة (Responsive) عبر نظامٍ شبكي ومجموعة مكوّناتٍ جاهزة [١٢].

نظام شبكي: تخطيطٌ مرن يتكيّف مع الأجهزة المختلفة.

مكوّنات جاهزة: أزرار ونوافذ وبطاقات وتنبيهات قابلة للتخصيص تسرّع التطوير.

الفائدة في المشروع: ضمان عرضٍ متّسق وأنيق للواجهة على مختلف أحجام الشاشات.

### ٢-٢-٤ الترجمة وتعدّد اللغات (i18n)

بُنيت في الواجهة طبقة ترجمةٍ في زمن التشغيل تعتمد ملفّات موارد بصيغة JSON للّغتين العربية والإنجليزية، مع خدمةٍ مركزية للترجمة ومكوّن تبديل لغة.

تبديل فوري للّغة: تبديل اللغة من دون إعادة تحميل الصفحة مع حفظ تفضيل المستخدم.

دعم الاتجاهين: تبديلٌ تلقائي لاتجاه الواجهة بين RTL وLTR بحسب اللغة المختارة.

تغطية واسعة: مئات النصوص المترجمة تغطّي عناصر التنقّل والنماذج والرسائل ولوحات المعلومات.

الفائدة في المشروع: إتاحة النظام لمستخدمين بالعربية والإنجليزية بتجربةٍ متّسقة وكاملة.

### ٢-٢-٥ بنية المكوّنات ونمط الموارد الموحّد

نُظِّمت الواجهة وفق بنية «ميزات» (Features) يضمّ كلٌّ منها مكوّناتها ونماذجها وخدماتها، مع نمطٍ موحّد لكلّ مورد FHIR يتكوّن من صفحات: قائمة (Index)، وإنشاء (Create)، وتعديل (Edit)، وتفاصيل (Details)، وتاريخ (History)، تُبنى فوق مكوّناتٍ مشتركة (مثل صفحة فهرس الموارد والترقيم وعارض JSON).

إعادة الاستخدام: مكوّناتٌ مشتركة تقلّل التكرار وتوحّد السلوك عبر جميع أنواع الموارد.

الاتّساق: تجربةٌ متطابقة في كلّ شاشات الموارد تسهّل تعلّم النظام.

قابلية التوسّع: إضافة موردٍ جديد تتبع النمط نفسه بأقلّ جهد.

الفائدة في المشروع: بناء واجهةٍ موحّدة وقابلة للصيانة تغطّي عشرات الشاشات بنمطٍ واحد.

## ٢-٣ أدوات الذكاء الاصطناعي

### ٢-٣-١ لغة Python ومكتبة scikit-learn

دُرّبت نماذج المخاطر بلغة Python باستخدام مكتبة scikit-learn، وهي مكتبةٌ واسعة الانتشار لتعلّم الآلة توفّر خوارزميات تصنيفٍ وانحدار جاهزة [١٠]، واعتُمدت خوارزمية الغابة العشوائية (Random Forest) لبناء النماذج الثلاثة.

مجموعات بيانات عامة: دُرّبت النماذج على بيانات Pima للسكري [١٤]، وأمراض القلب من UCI [١٥]، وإعادة الدخول «US hospitals-130» [١٦].

بيانات تركيبية للعرض: استُخدمت بيانات مرضى تركيبية مولّدة بأداة Synthea لتعبئة النظام لأغراض العرض [١٧].

مجموعة سمات مُختزَلة: اقتُصِر التدريب على السمات التي يمكن استخراجها فعلاً من سجلّ FHIR لضمان واقعية الاستدلال.

### ٢-٣-٢ صيغة ONNX وأداة skl2onnx ومحرّك ONNX Runtime

بعد التدريب تُصدَّر النماذج إلى صيغة ONNX المفتوحة عبر أداة skl2onnx [١١]، لتُحمَّل وتُشغَّل داخل الخادم باستخدام محرّك ONNX Runtime [٩].

فصل بيئتي التدريب والتشغيل: تدريبٌ بلغة Python واستدلالٌ داخل خادم C# من دون اعتماديةٍ على بيئة Python في الإنتاج.

أداء استدلال عالٍ: تشغيلٌ سريع وخفيف للنماذج عبر جلسة استدلال (InferenceSession).

تدهور لطيف: في حال غياب ملفّ نموذجٍ ما، يُعيد النظام نتيجةً فارغة بدل أن ينهار، فتُعرَض رسالة «غير متاح».

## ٢-٣-٣ نماذج المخاطر الثلاثة بالتفصيل

يقوم تصميم النماذج على مبدأ جوهري: لا يكون النموذج مفيداً في التطبيق إلا إذا أمكن استخلاص مدخلاته من سجلّ مريضٍ حقيقي؛ لذلك دُرّب كلّ نموذجٍ على «مجموعة سماتٍ مُختزَلة» تقتصر على الأعمدة التي يستطيع النظام استخراجها من بيانات FHIR، بدل المجموعة الكاملة في مجموعة البيانات البحثية، ما يجعل التدريب أميناً ويعلّم النموذج من النوع نفسه من البيانات التي سيراها أثناء الاستدلال. ويلخّص الجدول الآتي النماذج الثلاثة ومجموعات بياناتها ومؤشّرات أدائها:

| النموذج | ما يتنبّأ به | مجموعة البيانات | الدقّة | AUC |
|---|---|---|---|---|
| خطر السكري | احتمال الإصابة بالسكري من النمط الثاني | Pima Indians Diabetes | ٠,٧٣ | ٠,٨٢ |
| خطر القلب | خطر مرض القلب التاجي | UCI Heart Disease | ٠,٨١ | ٠,٩٣ |
| إعادة الدخول | إعادة الدخول إلى المشفى خلال ٣٠ يوماً | US hospitals-130 UCI | ٠,٦٥ | ٠,٦٣ |

*جدول ٢-١: نماذج المخاطر الثلاثة ومجموعات بياناتها ومؤشّرات أدائها.*

ويوضّح الجدول الآتي السمات التي يعتمدها كلّ نموذجٍ ومصدرها من موارد FHIR:

| النموذج | السمات (بترتيب التدريب) | مصدرها من FHIR |
|---|---|---|
| خطر السكري | الغلوكوز، الضغط الانبساطي، كتلة الجسم، العمر | ملاحظات (LOINC) + تاريخ ميلاد المريض |
| خطر القلب | العمر، الجنس، الضغط الانقباضي، الكوليسترول، أقصى نبض، سكّر صيامي | المريض + ملاحظات (LOINC) |
| إعادة الدخول | العمر، الجنس، عدد الحالات، عدد الأدوية الفعّالة، عدد الزيارات الحديثة، عدد الإجراءات | المريض + عدّ موارد الحالات والأدوية والزيارات والإجراءات |

*جدول ٢-٢: سمات كلّ نموذجٍ ومصدرها من موارد FHIR.*

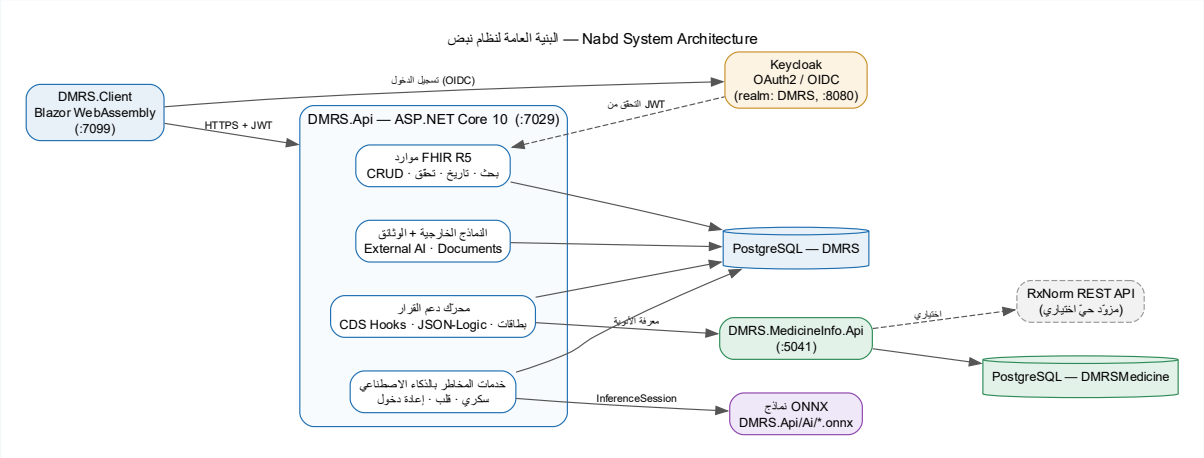
وعند الاستدلال تحمّل خدمة المخاطر المريض وموارده، وتبني متّجه السمات، وتعوّض أيّ سمةٍ ناقصة بوسيط مجموعة التدريب مع رفع إشارة «السمات غير مكتملة»، ثمّ تشغّل نموذج ONNX

وتحوّل الاحتمال إلى مستوى (منخفض دون ٠,٣٥، ومتوسط حتى ٠,٦٥، ومرتفع فوقها). وإذا غاب ملفّ نموذجٍ ما تُعيد الخدمة نتيجةً فارغة وتظهر «غير متاح» بدل أن ينهار النظام. ويُلاحَظ أنّ هذه النماذج دُرّبت على مجموعات بياناتٍ بحثية عامة لا على بيانات هذه المنصّة، فتقديراتها استرشادية للدعم لا للتشخيص، وتبقى أداة مساعدة للقرار السريري لا بديلاً عنه.

## ٢-٤ البنية العامة للنظام

يتألّف النظام من عدّة خدماتٍ متكاملة تتواصل فيما بينها عبر الشبكة وفق الأدوار الآتية:

- DMRS.Api: الواجهة البرمجية الرئيسية (ASP.NET Core) التي تتضمّن مسارات FHIR (إنشاء/قراءة/بحث/تاريخ/تحقّق)، ومحرّك دعم القرار، وخدمات المخاطر بالذكاء الاصطناعي، وإدارة الوثائق.
- DMRS.Client: الواجهة الأمامية (Blazor WebAssembly) التي تضمّ مساحة العمل السريرية وورشة دعم القرار والرؤى الذكية وبوابة المريض.
- DMRS.MedicineInfo.Api: خدمةٌ مستقلّة تقدّم معلومات الأدوية (البحث عبر RxCui) لمحرّك دعم القرار.
- DMRS.Shared: مكتبةٌ مشتركة تضمّ النماذج والثوابت المتبادلة.
- Keycloak: خادم الهويّة والمصادقة الذي يصدر رموز الدخول ويتحقّق منها.
- قاعدتا بيانات PostgreSQL: قاعدة DMRS للموارد والنسخ والفهارس وقواعد دعم القرار، وقاعدة DMRSMedicine لمعلومات الأدوية.
- نماذج ONNX: ملفّات النماذج الثلاثة التي يحمّلها الخادم لتقدير المخاطر.




**الوصف:** تُظهر الواجهة الأمامية (Blazor) وهي تتصل بخادم Keycloak لتسجيل الدخول (OIDC)، ثمّ تخاطب الواجهة البرمجية الرئيسية عبر HTTPS حاملةً رمز JWT. تستدعي الواجهة البرمجية قاعدة بيانات PostgreSQL لتخزين موارد FHIR، وتحمّل نماذج ONNX لتقدير

المخاطر، وتتصل بخدمة معلومات الأدوية (وبمصدر RxNorm [٢٢] اختيارياً) لإثراء تقييم دعم القرار. يُرسَم كلّ مكوّن في صندوقٍ مستقل وتربط بينها أسهمٌ تبيّن اتجاه الطلبات.

*الشكل ٢-١: نظرة عامة على معمارية نظام نبضوتدفّق الطلبات بين مكوّناته.*

## ٢-٥ الخلاصة

وفّرت هذه المجموعة من الأدوات أساساً متيناً لبناء نظامٍ موثوقٍ وقابل للتوسّع: منصّة .NET ولغة C# للخادم، وإطار EF Core وقاعدة PostgreSQL لإدارة البيانات، ومعيار FHIR ومكتبة Firely للتشغيل البيني، وإطار Blazor للواجهة مع Bootstrap وطبقة ترجمة، وخادم Keycloak للأمان، وسلسلة أدوات الذكاء الاصطناعي القائمة على Python وONNX. وقد جرى اختيار كلّ تقنيةٍ بعناية لتخدم هدف المنصّة في توحيد البيانات الصحية ودعم القرار السريري الذكي.

# الفصل الثالث – متطلبات المشروع والمخططات

يعرض هذا الفصل البنية التحتية اللازمة لتشغيل النظام، ثمّ المتطلّبات الوظيفية وغير الوظيفية، ثمّ الفاعلين الذين يتعاملون مع النظام ومخطّطات حالات الاستخدام الخاصة بهم مع وصفٍ تفصيلي لأهمّ الحالات، وصولاً إلى مخطّط قاعدة البيانات (ERD) وبعض مخطّطات التسلسل التي توضّح آليات العمل الداخلية.

## ٣-١ البنية التحتية

يتطلّب تشغيل النظام بنيةً تحتية تضمن التوافر والأمان وقابلية التوسّع، وتتكوّن من العناصر الآتية:

### ٣-١-١ خادم الويب Web Server

خادمٌ يستضيف الواجهة البرمجية المبنية على ASP.NET Core ويتولّى استقبال الطلبات ومعالجتها والتحقّق من الموارد وفق معيار FHIR. ويُفضَّل أن يدعم بيئة .NET (مثل Kestrel خلف وكيلٍ عكسي) وأن تتوفّر له موارد كافية من معالجٍ وذاكرةٍ وتخزين.

### ٣-١-٢ قاعدة بيانات مركزية

قاعدة بيانات PostgreSQL لتخزين موارد FHIR وتاريخ نسخها وفهارس البحث وقواعد دعم القرار، إضافةً إلى قاعدةٍ منفصلة لخدمة معلومات الأدوية. ويُشترَط أن تكون مستقرّةً وسريعة الاستجابة ومحميةً بنُسخٍ احتياطية دورية.

### ٣-١-٣ خادم المصادقة Keycloak

خادم Keycloak [٨] لإدارة الهويّات وإصدار رموز الدخول والتحقّق منها وفق بروتوكولَي OAuth2/OpenID Connect [١٨]، ويشكّل المرجعية الموحّدة للمصادقة والتفويض في النظام، ويُهيَّأ له مجالٌ (Realm) باسم DMRS وعميلٌ (Client) للواجهة الأمامية.

### ٣-١-٤ خدمة معلومات الأدوية

خدمةٌ مستقلّة (DMRS.MedicineInfo.Api) مع قاعدة بياناتها الخاصة، تزوّد محرّك دعم القرار بمعلومات الجرعات والسلامة الدوائية، مع إمكانية الاستعاضة عنها بمصدرٍ حيّ مثل RxNorm [٢٢].

### ٣-١-٥ شبكة اتصال محمية

قناة اتصالٍ مشفّرة (HTTPS) مع شهادات SSL بين الواجهة والخادم لحماية البيانات الصحية الحسّاسة أثناء النقل، ويُنصَح بجدار حمايةٍ وأنظمة كشف تسلّل لحماية الخوادم من الهجمات الإلكترونية.

## ٣-٢ المتطلبات الوظيفية (للحذف)

يجب أن يوفّر النظام الوظائف الآتية:

- إدارة موارد FHIR (إنشاء، قراءة، تحديث، حذف ناعم، بحث) لكلّ الأنواع المدعومة مع التحقّق من صحّتها.
- حفظ تاريخ النسخ لكلّ مورد وإتاحة استعراض النسخ السابقة.
- مصادقة المستخدمين عبر Keycloak وتطبيق تحكّمٍ بالوصول قائم على الأدوار ونطاقات الصلاحيات.
- عرض لوحة معلوماتٍ لكلّ دور تتضمّن المؤشّرات والمواعيد والتنبيهات وقائمة المراقبة التنبّؤية.
- تقدير مخاطر المريض عبر النماذج الثلاثة وعرضها على مخطّط المريض ولوحة الرؤى الذكية.
- تأليف قواعد دعم القرار والتحقّق منها ومعاينتها ونشرها وتفعيلها، وإطلاق بطاقات التنبيه عند تحقّق شروطها.
- تسجيل نماذج ذكاء اصطناعي خارجية وتشغيلها على بيانات المرضى.
- إدارة المؤسسات والكادر عبر آلية الدعوة والربط (Invite/Claim).
- إتاحة بوابة خدمةٍ ذاتية للمريض وإدارة الوثائق السريرية.
- دعم الواجهة باللغتين العربية والإنجليزية مع اتجاه الكتابة المناسب.

## ٣-٣ المتطلبات غير الوظيفية (للحذف)

الأمان والخصوصية: تشفير الاتصال، وحماية الأسرار عند التخزين، وتحكّمٌ دقيق بالوصول بحسب الدور والصلاحيات.

الأداء: زمن استجابةٍ منخفض للطلبات الشائعة، واستدلالٌ سريع للنماذج، وتجميعٌ للطلبات على جانب الخادم لتقليل عددها.

الموثوقية: حفظٌ كامل لتاريخ النسخ، وتدهورٌ لطيف عند غياب أحد المكوّنات (مثل غياب ملفّ نموذج).

قابلية التشغيل البيني: الالتزام بمعيار FHIR R5 في تمثيل الموارد ومساراتها.

قابلية الصيانة والتوسّع: بنيةٌ معيارية مقسّمة إلى ميزاتٍ وطبقات، تسهّل إضافة موارد أو نماذج جديدة.

سهولة الاستخدام: واجهةٌ متجاوبة وثنائية اللغة مع تنظيمٍ واضح للتنقّل بحسب الدور.

## ٣-٤ الأمان والصلاحيات

يعتمد النظام نموذج تحكّمٍ بالوصول قائم على الأدوار (RBAC) مدعوماً بنطاقات صلاحياتٍ على غرار SMART on FHIR [١٩]. فبعد مصادقة المستخدم عبر Keycloak [٨] وفق OAuth2/OpenID Connect [١٨] يحمل رمز الدخول (JWT) أدواره، ويتحقّق الخادم من كلّ طلبٍ ليقرّر ما إذا كان مسموحاً. ويميّز النظام بين ثلاثة أنواعٍ من النطاقات: نطاق النظام (System) للعمليات الإدارية على مستوى المنصّة، ونطاق المستخدم (User) للكادر الذي يصل إلى بيانات المرضى ضمن صلاحيته، ونطاق المريض (Patient) الذي يقيّد المريض ببياناته هو فقط. ويلخّص الجدول الآتي توزيع الصلاحيات على الأدوار:

| الدور | أبرز الصلاحيات |
|---|---|
| مدير النظام | الوصول الكامل: إدارة المؤسسات وقواعد دعم القرار ونماذج الذكاء الاصطناعي الخارجية، إضافةً إلى كلّ القدرات السريرية. |
| مدير المؤسسة | العمل السريري والجدولة ضمن مؤسسته، وإنشاء دعوات الكادر والمرضى، وتأليف قواعد دعم القرار. |
| الطبيب/الممارس | مساحة العمل السريرية: المرضى والمخطّطات والطلبات والمواعيد وبطاقات المخاطر والوثائق. |
| المريض | سجلّه هو فقط عبر بوابة المريض (نطاق Patient)، من دون أيّ وصولٍ لبيانات الآخرين. |

*جدول ٣-١٩: توزيع الصلاحيات على الأدوار.*

ويتبنّى النظام نموذج السجلّ الطبي الوطني (National Record): فبما أنّ المريض قد يُعالَج في أكثر من جهة، يُتاح لأيّ عضو كادرٍ طبي (طبيب/ممارس) الاطّلاع على سجلّ أيّ مريضٍ في المنصّة والمساهمة في بياناته السريرية عبر المؤسسات كافّة — أي إنشاء الحالات والملاحظات والإجراءات والأدوية وطلبات الخدمة والحساسيات والزيارات والمواعيد وتحديثها بصرف النظر عن المؤسسة المالكة للسجل. وبذلك تكتمل فائدة القراءة عبر المؤسسات: فلا يقتصر الطبيب على مطالعة سجلّ مريضٍ مُحال إليه من جهةٍ أخرى، بل يستطيع أيضاً وصف الدواء وتوثيق الرعاية له. أمّا تعديل البيانات التعريفية (الديموغرافية) للمريض، وعمليات الحذف، والبيانات الإدارية (المؤسسات والمواقع والكادر) فتبقى مقيَّدةً بمؤسسة الكادر، حفاظاً على ملكية كلّ مؤسسةٍ لبياناتها الإدارية.

وتُحمى البيانات الحسّاسة بإجراءاتٍ إضافية: تشفير الاتصال عبر HTTPS، وتخزين أسرار نماذج الذكاء الاصطناعي الخارجية مشفّرةً عبر ASP.NET Data Protection بحيث لا تُعاد إلى

العملاء، والتحقّق من كون روابط النماذج الخارجية آمنة (HTTPS) فقط، إضافةً إلى التحقّق من صحّة الموارد قبل تخزينها.

## ٣-٥ المخططات وقاعدة البيانات

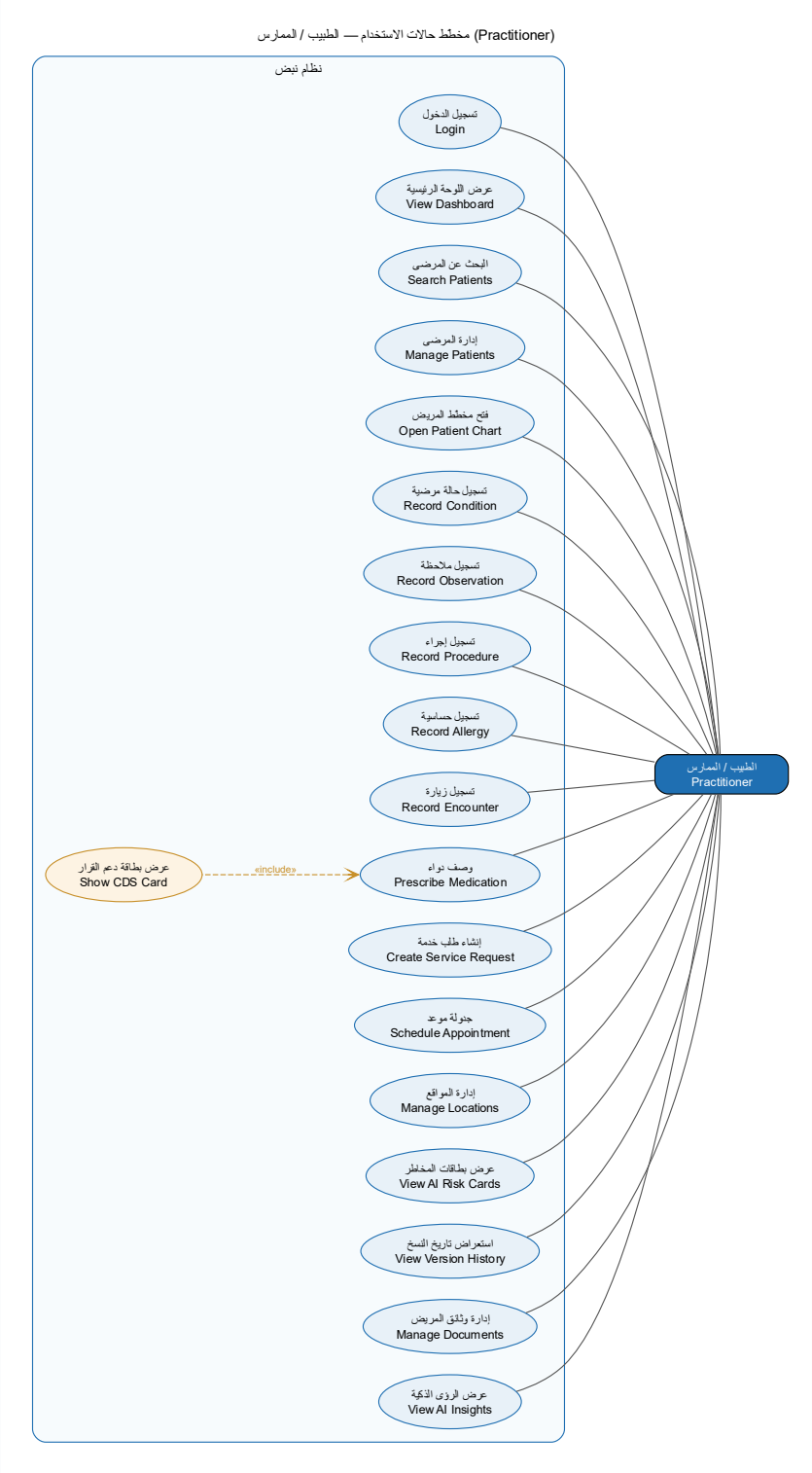
### ٣-٣-١ الفاعلون Actors

يتعامل مع النظام خمسة فاعلين رئيسيين، يحدّد دور كلٍّ منهم ما يراه وما يستطيع فعله:

| الفاعل | الوصف والصلاحيات |
|---|---|
| مدير النظام (System Admin) | أعلى صلاحية في المنصّة؛ يدير المؤسسات وقواعد دعم القرار ونماذج الذكاء الاصطناعي الخارجية، ويطّلع على كلّ الأقسام. |
| مدير المؤسسة (Org Admin) | يدير العمل السريري والجدولة ضمن مؤسسته، ويُنشئ دعوات الكادر والمرضى، ويؤلّف قواعد دعم القرار. |
| الطبيب/الممارس (Practitioner) | يدير سجلات المرضى والطلبات والمواعيد، ويطّلع على بطاقات المخاطر والرؤى الذكية، ويرفع الوثائق. |
| المريض (Patient) | يطّلع على سجلّه الصحّي ومواعيده وأدويته بعد ربط حسابه، ويدير ملفّه الشخصي. |
| الزائر (Unauthenticated) | مستخدمٌ غير مسجّل يطّلع على صفحة الترحيب والتعريف بالنظام ويبدأ تسجيل الدخول، من دون الوصول إلى أيّ بياناتٍ سريرية. |

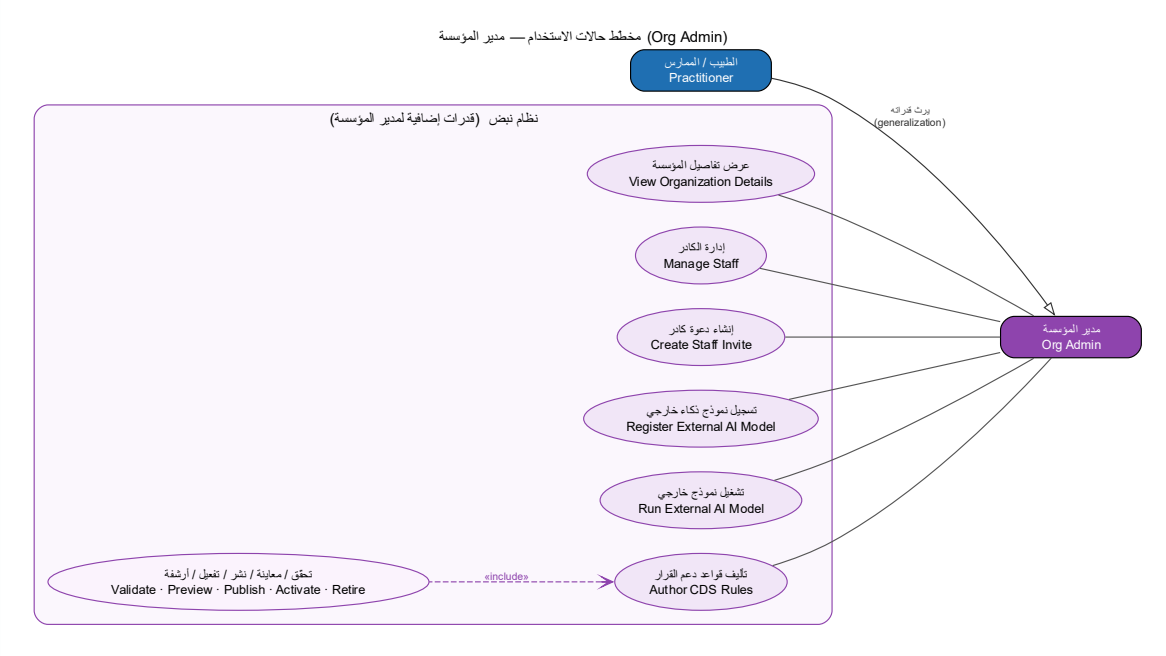
### ٣-٣-٢ مخططات حالات الاستخدام Use Case Diagram

تُستخدَم مخطّطات حالات الاستخدام في طور تحليل المشروع لتعريف المهمّات التي يقوم بها النظام وتجزئتها، وتعتمد على فصل النظام إلى قسمين رئيسيين هما الفاعلون (Actors) وحالات الاستخدام (Use Cases)؛ إذ تصف حالة الاستخدام سلوك النظام عندما يطلب أحد الفاعلين منه شيئاً. وفيما يأتي وصفٌ لمخطّط حالات الاستخدام لكلّ فاعل:



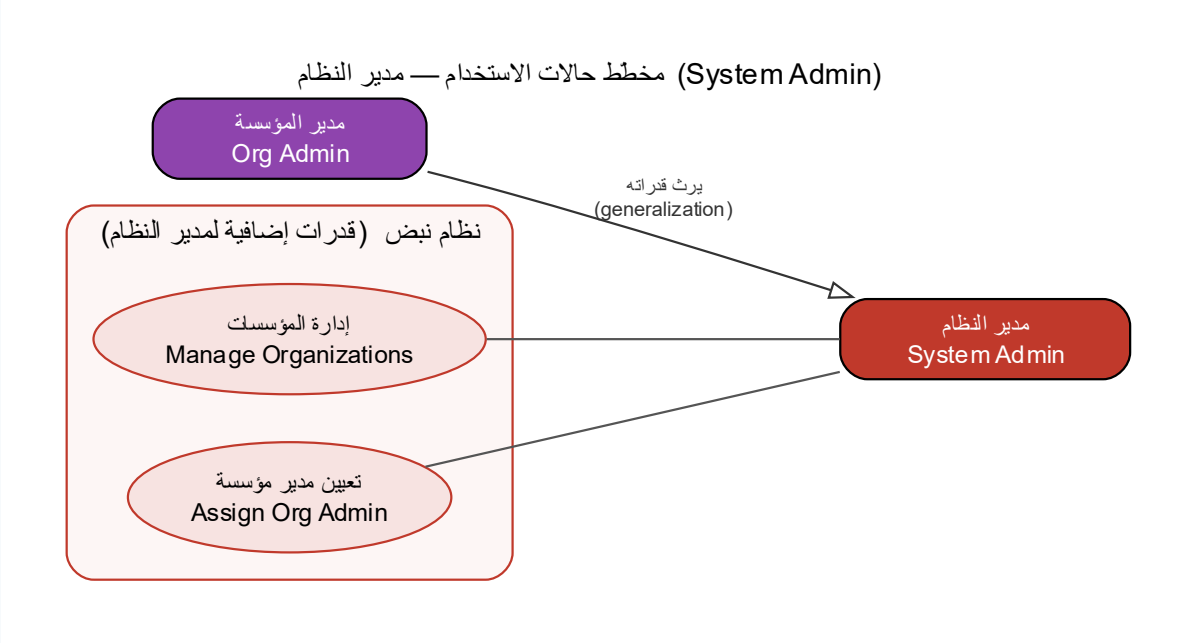

**الوصف:** يظهر الفاعل «الطبيب» على اليمين، وتتفرّع منه حالات الاستخدام داخل حدود النظام: إدارة المرضى (إنشاء/تعديل/بحث)، فتح مخطّط المريض، تسجيل البيانات السريرية (حالة، ملاحظة، إجراء، حساسية، زيارة)، وصف الأدوية، إنشاء طلبات الخدمة، جدولة المواعيد، الاطّلاع على بطاقات المخاطر، استعراض تاريخ النسخ، ورفع الوثائق. وترتبط حالة «وصف دواء» بحالة «عرض بطاقة دعم القرار» بعلاقة «يتضمّن» (include).

*الشكل ٣-١: حالات الاستخدام الخاصة بالطبيب/الممارس.*



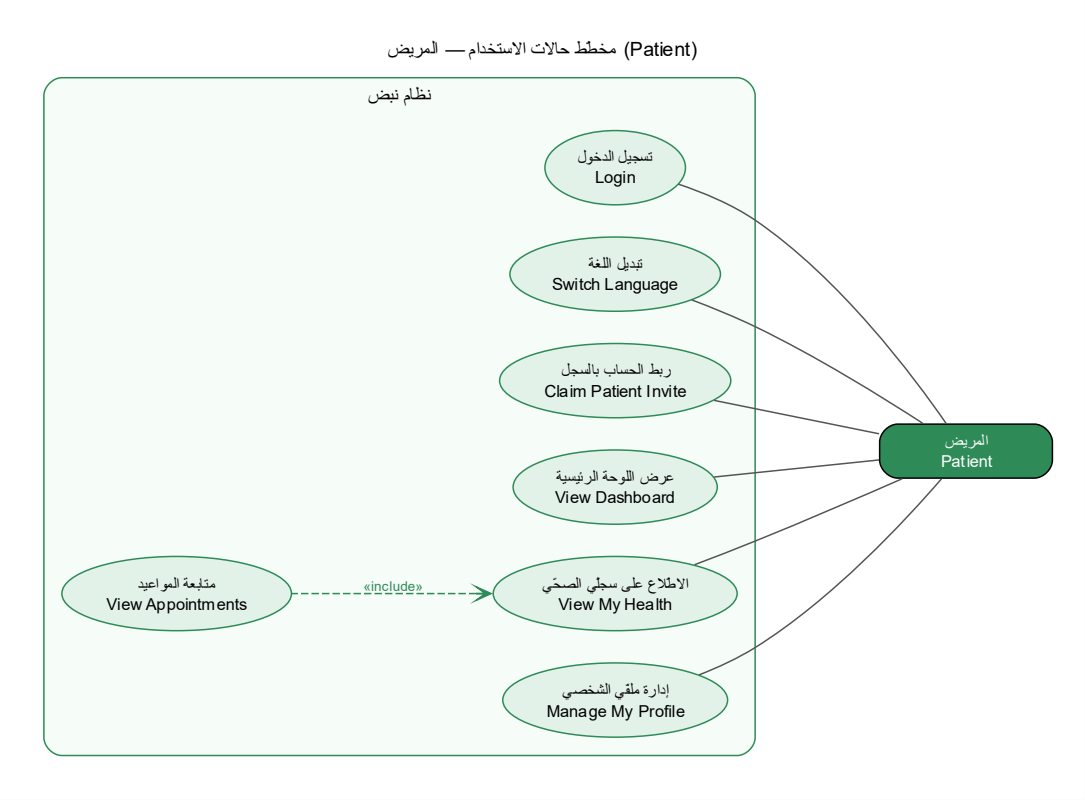

**الوصف:** يظهر الفاعل «مدير المؤسسة» وترتبط به حالات: إدارة المؤسسة وبياناتها، إنشاء دعوات الكادر والمرضى ومتابعتها، إدارة الكادر، إضافةً إلى تأليف قواعد دعم القرار والتحقّق منها ومعاينتها ونشرها وتفعيلها. كما يرث مدير المؤسسة قدرات الطبيب السريرية ضمن نطاق مؤسسته.

*الشكل ٣-٢: حالات الاستخدام الخاصة بمدير المؤسسة.*



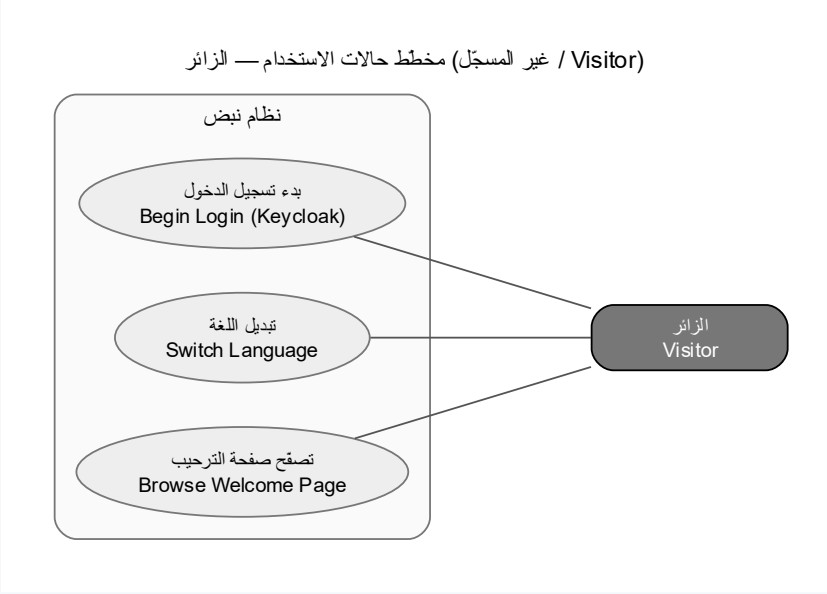

**الوصف:** يظهر الفاعل «مدير النظام» وترتبط به حالات: إدارة المؤسسات (إنشاء/تعديل/تعيين مدراء)، إدارة قواعد دعم القرار على مستوى المنصّة، وتسجيل نماذج الذكاء الاصطناعي الخارجية وإدارتها. ويملك صلاحيةً شاملة تغطّي كلّ أقسام النظام.

*الشكل ٣-٣: حالات الاستخدام الخاصة بمدير النظام.*




**الوصف:** يظهر الفاعل «المريض» وترتبط به حالات: ربط الحساب عبر رمز دعوة، الاطّلاع على ملخّص السجل الصحّي (الحالات والأدوية والملاحظات)، متابعة المواعيد، وإدارة الملف الشخصي. وتُقيَّد كلّ حالاته ضمن سجلّه هو فقط.

*الشكل ٣-٤: حالات الاستخدام الخاصة بالمريض.*




**الوصف:** يظهر الفاعل «الزائر» (غير المسجّل) وترتبط به حالتان فقط: تصفّح صفحة الترحيب والتعريف بالنظام، وبدء تسجيل الدخول عبر Keycloak، إضافةً إلى تبديل لغة الواجهة. ولا يملك الزائر أيّ وصولٍ إلى البيانات السريرية.

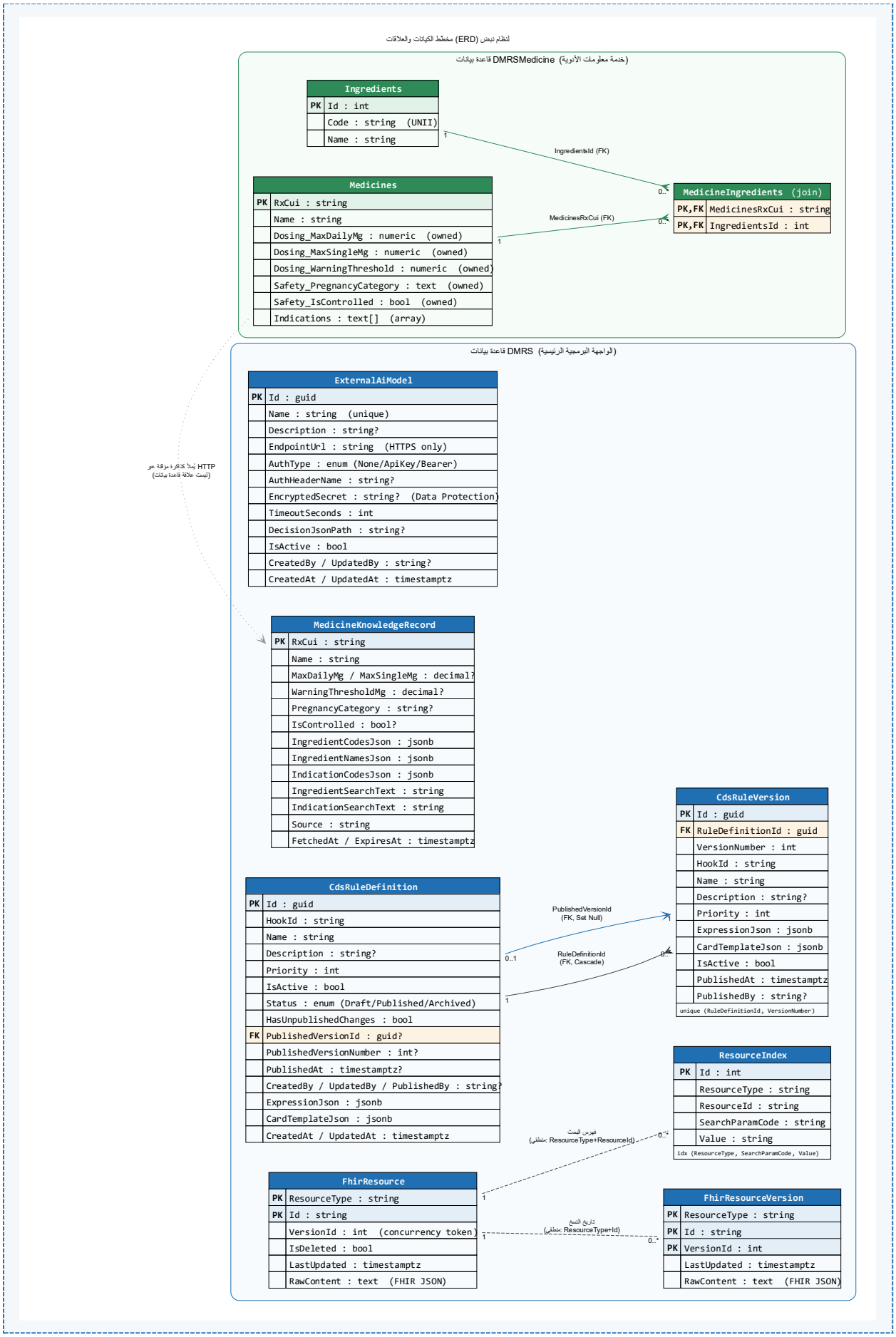
*الشكل ٣-٥: حالات الاستخدام الخاصة بالزائر.*

### ٣-٣-٣ مخطّط قاعدة البيانات Entity Relation Diagram

تعتمد قاعدة البيانات تصميماً «عامّاً» (Generic) يخزّن كلّ مورد FHIR بصيغته الأصلية (JSON عبر نوع JSONB في [٦] PostgreSQL) في جدول FhirResources، مع جدول FhirResourceVersions لحفظ تاريخ النسخ كاملاً، وجدول ResourceIndices لفهرسة الحقول القابلة للبحث؛ ما يمنح مرونةً في دعم أنواع موارد متعدّدة من دون تغيير المخطّط. وتُضاف جداول دعم القرار CdsRuleDefinitions وCdsRuleVersions، وجدول معرفة الأدوية MedicineKnowledgeRecords، وجدول نماذج الذكاء الاصطناعي الخارجية ExternalAiModels. أمّا خدمة الأدوية فلها قاعدتها الخاصة (DMRSMedicine) التي تضمّ جدولَي Medicine وIngredient بعلاقة متعدّد-إلى-متعدّد.

يلخّص الجدول الآتي أبرز الكيانات وأدوارها:

| الكيان (Entity) | الدور |
|---|---|
| FhirResource | النسخة الحالية لكلّ مورد FHIR مع نوعه ومعرّفه ورقم نسخته وحالة الحذف الناعم ومحتواه الخام (JSON). |
| FhirResourceVersion | أرشيف تاريخ النسخ لكلّ مورد (نوع + معرّف + رقم نسخة) مع وقت التحديث والمحتوى. |
| ResourceIndex | فهرس بحثٍ مُسطّح يربط (نوع المورد + معرّفه) بقيم بارامترات البحث القياسية. |
| CdsRuleDefinition | تعريف قاعدة دعم القرار: المعرّف، الخطاف (Hook)، الاسم، الحالة (مسودّة/منشورة/مؤرشَفة)، الأولوية، تعبير الإطلاق وقالب البطاقة (jsonb). |
| CdsRuleVersion | نسخةٌ غير قابلة للتعديل من القاعدة عند كلّ نشرٍ، مرتبطة بتعريفها مع رقم نسخةٍ ووقت نشر. |
| MedicineKnowledgeRecord | ذاكرةٌ مؤقّتة لمعرفة الأدوية (الجرعات والسلامة) مع وقتَي الجلب والانتهاء. |
| ExternalAiModel | سجلّ نموذج ذكاء اصطناعي خارجي: الاسم والرابط ونوع المصادقة والسرّ المشفّر ومهلة الاتصال ومسار القرار. |
| Medicine / Ingredient | في قاعدة الأدوية: الدواء (مع جرعاته وسلامته كأنواعٍ مملوكة) ومكوّناته بعلاقة متعدّد-إلى-متعدّد. |

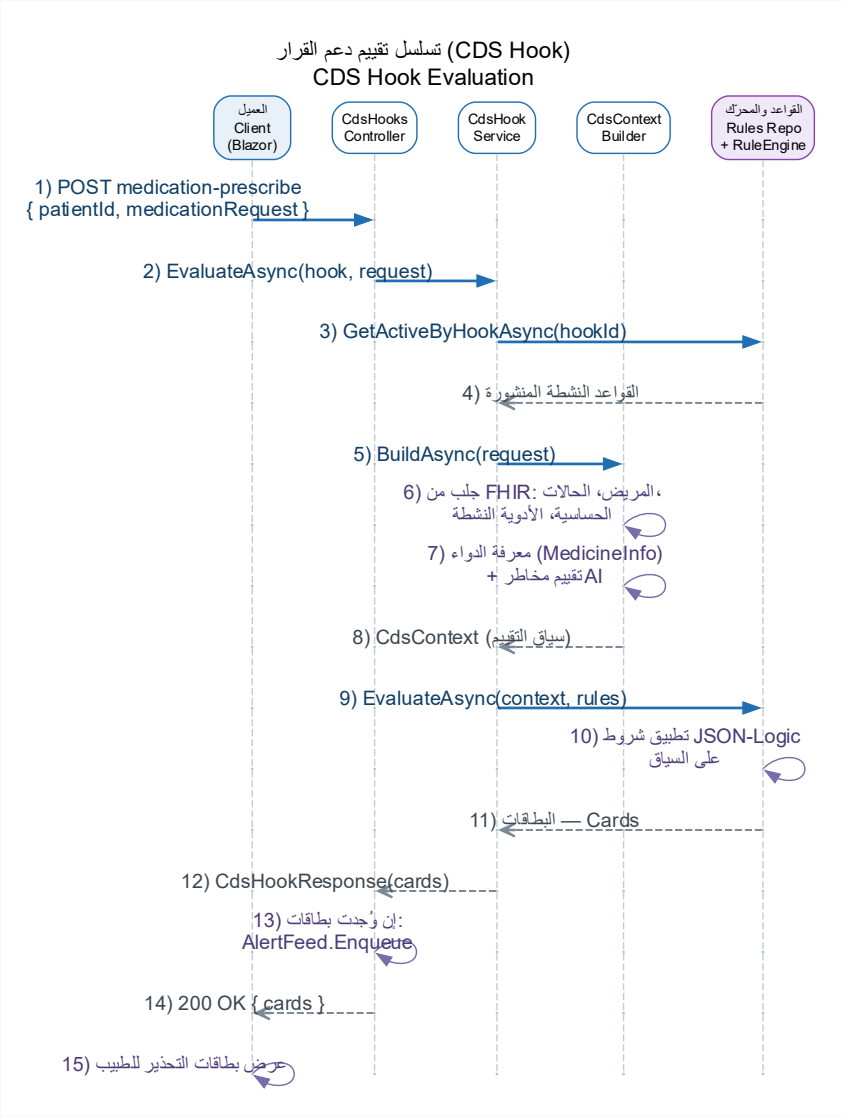

لنظام نبض (ERD) مخطط الكيانات والعلاقات
قاعدة بيانات DMRSMedicine (خدمة معلومات الأدوية)
Ingredients
PK Id : int
Code : string (UNII)
Name : string
IngredientsId (FK)
Medicines
PK RxCui : string
Name : string
Dosing_MaxDailyMg : numeric (owned)
Dosing_MaxSingleMg : numeric (owned)
Dosing_WarningThreshold : numeric (owned)
Safety_PregnancyCategory : text (owned)
Safety_IsControlled : bool (owned)
Indications : text[] (array)
MedicinesRxCui (FK)
MedicineIngredients (join)
PK,FK MedicinesRxCui : string
PK,FK IngredientsId : int
قاعدة بيانات DMRS (الواجهة البرمجية الرئيسية)
ExternalAiModel
PK Id : guid
Name : string (unique)
Description : string?
EndpointUrl : string (HTTPS only)
AuthType : enum (None/ApiKey/Bearer)
AuthHeaderName : string?
EncryptedSecret : string? (Data Protection)
TimeoutSeconds : int
DecisionJsonPath : string?
IsActive : bool
CreatedBy / UpdatedBy : string?
CreatedAt / UpdatedAt : timestamptz
HTTP يملأ كذاكرة مؤقتة عبر
(ليست علاقة قاعدة بيانات)
MedicineKnowledgeRecord
PK RxCui : string
Name : string
MaxDailyMg / MaxSingleMg : decimal?
WarningThresholdMg : decimal?
PregnancyCategory : string?
IsControlled : bool?
IngredientCodesJson : jsonb
IngredientNamesJson : jsonb
IndicationCodesJson : jsonb
IngredientSearchText : string
IndicationSearchText : string
Source : string
FetchedAt / ExpiresAt : timestamptz
CdsRuleVersion
PK Id : guid
FK RuleDefinitionId : guid
VersionNumber : int
HookId : string
Name : string
Description : string?
Priority : int
ExpressionJson : jsonb
CardTemplateJson : jsonb
IsActive : bool
PublishedAt : timestamptz
PublishedBy : string?
unique (RuleDefinitionId, VersionNumber)
CdsRuleDefinition
PK Id : guid
HookId : string
Name : string
Description : string?
Priority : int
IsActive : bool
Status : enum (Draft/Published/Archived)
HasUnpublishedChanges : bool
FK PublishedVersionId : guid?
PublishedVersionNumber : int?
PublishedAt : timestamptz?
CreatedBy / UpdatedBy / PublishedBy : string?
ExpressionJson : jsonb
CardTemplateJson : jsonb
CreatedAt / UpdatedAt : timestamptz
PublishedVersionId
(FK, Set Null)
RuleDefinitionId
(FK, Cascade)
ResourceIndex
PK Id : int
ResourceType : string
ResourceId : string
SearchParamCode : string
Value : string
idx (ResourceType, SearchParamCode, Value)
فهرس البحث
(منطقي: ResourceType+ResourceId)
FhirResource
PK ResourceType : string
PK Id : string
VersionId : int (concurrency token)
IsDeleted : bool
LastUpdated : timestamptz
RawContent : text (FHIR JSON)
تاريخ النسخ
(منطقي: ResourceType+Id)
FhirResourceVersion
PK ResourceType : string
PK Id : string
PK VersionId : int
LastUpdated : timestamptz
RawContent : text (FHIR JSON)

**الوصف:** يُظهر الكيان FhirResource في الوسط وترتبط به علاقتان منطقيتان: «واحد إلى متعدّد» مع FhirResourceVersion (تاريخ النسخ بحسب النوع والمعرّف)، و«واحد إلى متعدّد» مع ResourceIndex (فهرس البحث). وإلى جانبها يظهر CdsRuleDefinition مرتبطاً بـ CdsRuleVersion (نشر مُصدَّر النسخ)، وكيانا MedicineKnowledgeRecord وExternalAiModel كجدولين مستقلّين. وفي قاعدةٍ منفصلة يظهر Medicine وIngredient بعلاقة متعدّد-إلى-متعدّد.

*الشكل ٣-٦: مخطّط الكيانات والعلاقات لقاعدتَي بيانات النظام.*

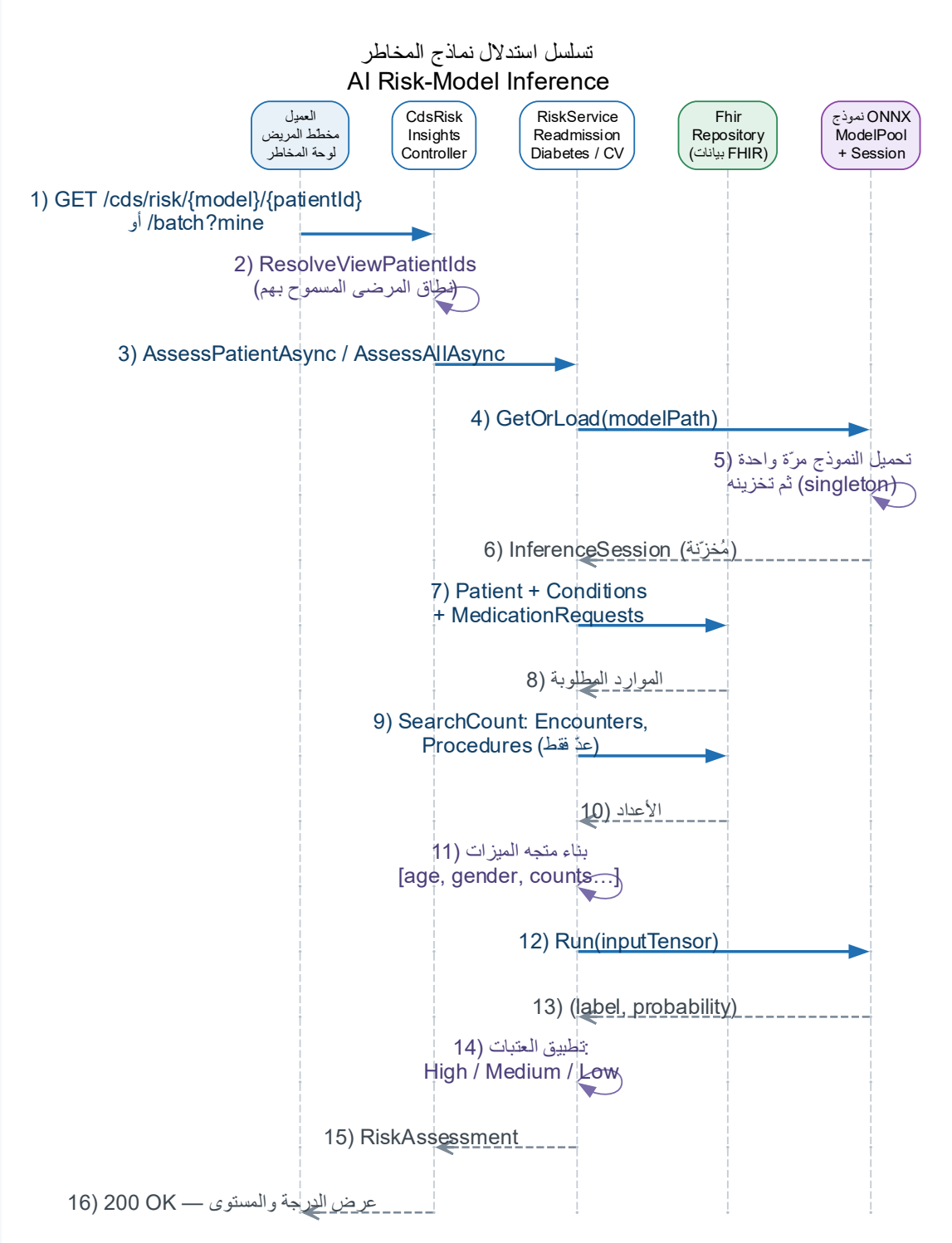
## ٣-٣-٤ مخطّطات التسلسل لأهمّ الآليات

نوضّح فيما يأتي ثلاث آليّات داخلية أساسية عبر وصفٍ لمخطّطات التسلسل الخاصة بها:



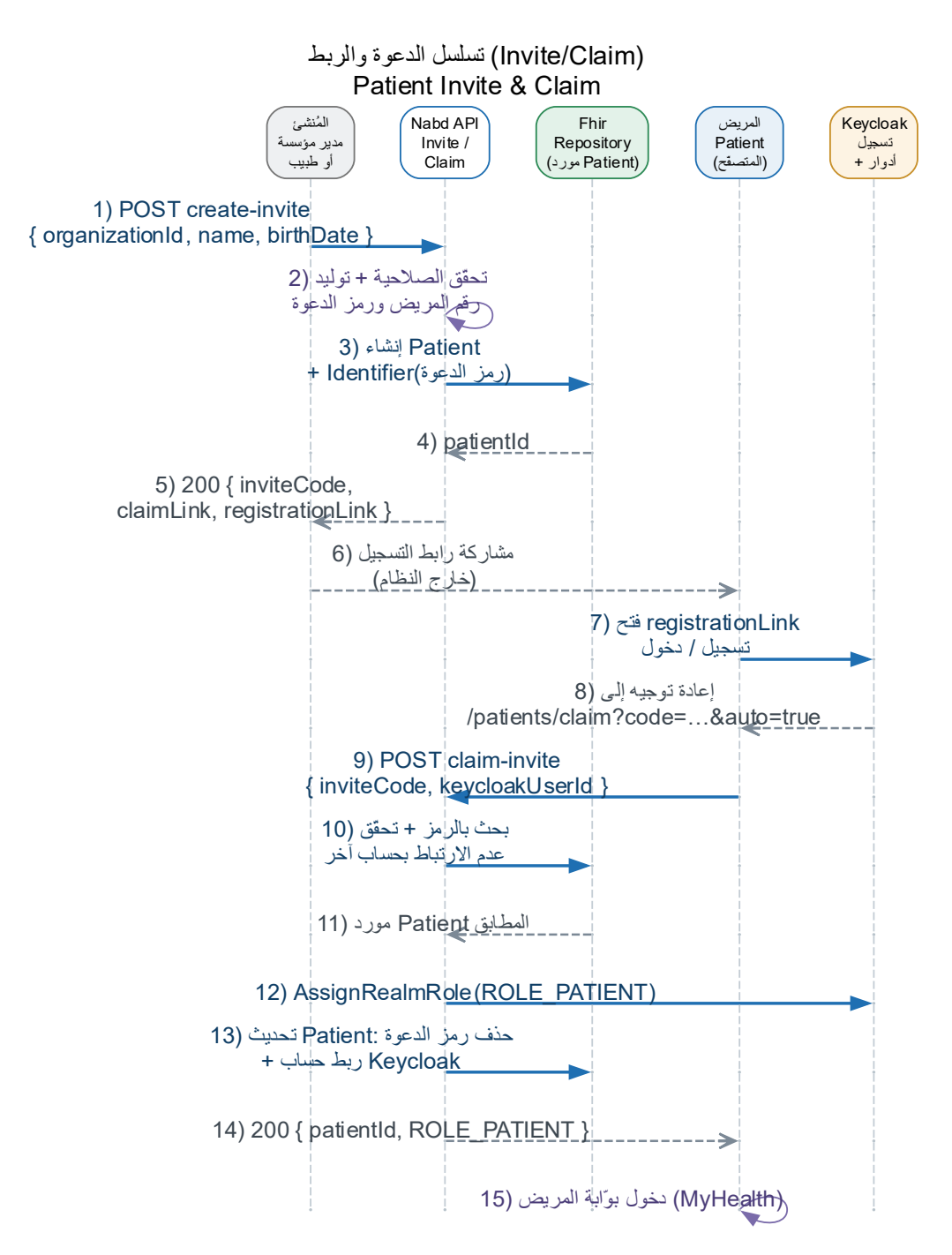

**الوصف:** يبدأ بإجراء العميل (مثل وصف دواء) فيرسل طلباً إلى متحكّم CDS Hooks [١٣] مع سياق المريض، فيبني «باني السياق» متغيّرات التقييم بجمع بيانات المريض السريرية من مستودع FHIR، واستدعاء خدمة المخاطر (متغيّرات ai.*)، واستعلام معرفة الأدوية. ثمّ يمرّر المتغيّرات إلى محرّك القواعد (JSON-Logic) الذي يقيّم القواعد المنشورة الفعّالة ويعيد القواعد المطابقة على شكل بطاقات تُعرَض للطبيب.

*الشكل ٣-٧: تسلسل تقييم قواعد دعم القرار وإطلاق البطاقات.*




**الوصف:** تحمّل خدمة المخاطر المريض وموارده، وتبني متّجه السمات عبر «مستخرِج سمات الملاحظات» (بحسب رموز LOINC)، ثمّ تعوّض السمات الناقصة بالوسيط وتؤشّر على ذلك، وتشغّل نموذج ONNX عبر جلسة استدلال، وتفسّر الخرج إلى احتمالٍ ومستوى (منخفض/متوسط/مرتفع) قبل إعادته إلى الواجهة.

*الشكل ٣-٨: تسلسل استخراج السمات وتشغيل نموذج المخاطر.*




**الوصف:** يُنشئ المدير دعوةً فيُنشَأ سجلّ FHIR ويُولَّد رمز دعوة. يسلّم الرمز للشخص المدعوّ الذي يسجّل دخوله عبر Keycloak ويُدخل الرمز، فيربط النظام حساب Keycloak بالسجل المناسب ويمنحه الدور الصحيح (مريض/طبيب/مدير مؤسسة).

*الشكل ٣-٩: تسلسل ربط حساب الدخول بالسجل عبر آلية الدعوة.*

## ٣-٦ واجهة النظام البرمجية (API)

### ٣-٦-١ مبدأ التصميم

تُبنى واجهة النظام البرمجية على عقدٍ (Contract) موحّدٍ يتّبع نمط REST ويلتزم بمعيار FHIR HL7 R5 [١] في مساراته وصيغه، فبدل أن يعرّف كلّ نوع مورد مساراته الخاصّة، تتشارك أنواع الموارد الطبية الخمسة عشر مجموعةً واحدة من ثماني عمليّاتٍ قياسية) بحث، وقراءة، وقراءة نسخة، وتاريخ، وإنشاء، وتحديث، وتعديل جزئي، وحذف .(ونتيجة ذلك أنّ الواجهة تضمّ نحو مئةٍ وعشرين مساراً لموارد FHIR تنتظم جميعها في جدولٍ واحد، إضافةً إلى نحو ثمانٍ وخمسين نقطة نهايةٍ متخصّصة تخدم دعم القرار والتحليلات والذكاء الاصطناعي والدعوات والخدمة الذاتية . ويوثّق هذا القسم شكل العقد ومبادئه لا سرد النقاط كافّةً؛ إذ يتوافر السرد المرجعي الكامل لكلّ نقطة نهايةٍ على مستودع المشروع (انظر القسم ٦-١) وعبر التوثيق التفاعلي (القسم ٣-٦-٦).

## ٣-٦-٢ الواجهة الموحّدة لموارد FHIR

تُخدَم جميع أنواع موارد FHIR عبر متحكّمٍ عامٍّ واحد على المسار /fhir/{Type} ، بالعمليّات القياسية الآتية:

*جدول ٣-٢٠: العمليّات القياسية على موارد FHIR.*

| العملية | الطريقة والمسار | الوصف |
|---|---|---|
| بحث(Search) | GET /fhir/{Type}?param=value | يعيد حزمة FHIR (Bundle) بنتائج البحث؛ والوسائط هي وسائط بحث FHIR القياسية. |
| قراءة(Read) | GET /fhir/{Type}/{id} | النسخة الحالية من المورد. |
| قراءة نسخة(vread) | GET /fhir/{Type}/{id}/_history/{vid} | نسخةٌ تاريخية محدّدة من المورد. |
| تاريخ(History) | GET /fhir/{Type}/{id}/_history | جميع نسخ المورد. |
| إنشاء(Create) | POST /fhir/{Type} | جسم الطلب موردFHIR ؛ ويعيد 201 مع ترويسة.Location |
| تحديث(Update) | PUT /fhir/{Type}/{id} | استبدالٌ كامل؛ ويجب أن يطابق معرّف الجسم معرّف المسار. |
| تعديل جزئي(Patch) | PATCH /fhir/{Type}/{id} | تحديثٌ جزئي للمورد. |
| حذف(Delete) | DELETE /fhir/{Type}/{id} | حذفٌ ناعم يحفظ التاريخ. |

وتشمل أنواع الموارد المدعومة :المريض، والممارس، ودور الممارس، والمؤسسة، والموقع، والزيارة، وفرط الحساسية، والحالة المرضية، والملاحظة، والإجراء، وطلب الدواء، والموعد، وطلب الخدمة، والحزمة، والإسناد (Provenance) — وتفصيلها مذكورٌ في القسم 3-6. ويجري التحقّق من صحّة كلّ مورد يُنشَأ أو يُحدَّث وفق مواصفة FHIR R5 قبل تخزينه، فإن كان الجسم غير صالحٍ رُدّت استجابة 400 مصحوبةً بمورد OperationOutcome يوضّح سبب الرفض.

## ٣-٦-٣ عائلات المسارات المتخصّصة

إلى جانب الواجهة الموحّدة، تعرّف الواجهة عائلاتٍ من المسارات المتخصّصة، تخدم كلٌّ منها وظيفةً بعينها .ويلخّص الجدول الآتي هذه العائلات وعدد نقاط النهاية في كلٍّ منها وا لصلاحية المطلوبة (وفق نموذج الصلاحيات المفصّل في القسم ٣-٤):

*جدول ٣-٢١: عائلات المسارات المتخصّصة في الواجهة البرمجية.*

| بادئة المسار | الوظيفة | عدد النقاط | الصلاحية المطلوبة |
|---|---|---|---|
| /cds-services | خدمات CDS Hooks: الاكتشاف وتنفيذ الخدمة. | 2 | نطاقFHIR |
| /cds/rules | إدارة قواعد دعم القرار (إنشاء، ونشر، وتفعيل، ونسخ، ومعاينة، وقوالب). | 16 | مدير المؤسسة / النظام |
| /cds/risk | تقديرات المخاطر بالذكاء الاصطناعي (فردية ودُفعية). | 6 | نطاقFHIR |
| /cds/medications | معرفة الأدوية المخزّنة (بحث، وتفصيل، وتحديث). | 3 | نطاقFHIR |
| /cds/alerts | سجلّ أحدث بطاقات التنبيه المُطلَقة. | 1 | نطاقFHIR |
| /external-ai | سجلّ نماذج الذكاء الاصطناعي الخارجية وتشغيلها. | 7 | مدير (للسجل) / نطاق FHIR (للتشغيل) |
| /analytics | مؤشّرات لوحات المعلومات وانتشار الحالات. | 2 | نطاقFHIR |
| /api/patients | دعوات المرضى وربطها ووثائق المريض. | 7 | مستخدم مُصادَق |
| /api/staff | دعوات الكادر وربطها ودليل الكادر وتزويد الحسابات. | 4 | مستخدم مُصادَق |
| /api/me | سياق المستخدم الحالي وخدمة المريض الذاتية. | 3 | مستخدم مُصادَق |
| /dev | أدوات التطوير وتحميل البيان ات التجريبية (بيئة التطوير فقط). | 7 | بيئة التطوير |

## ٣-٦-٤ المصادقة والاصطلاحات

تتطلّب جميع نقاط النهاية الرئيسة رمز دخولٍ من نوع JWT صادراً عن خادم Keycloak لنطاق DMRS، يُرسَل في ترويسةAuthorization: Bearer ، ويخضع أغلبها لسياسة FhirScope مع تقييدٍ إضافي لكلّ مريضٍ ومؤسسةٍ عبر خدمة التخويل على غر ار SMART (انظر القسم ٣-٤). ويلخّص الجدول الآتي أبرز الاصطلاحات:

*جدول ٣-٢٢: اصطلاحات الواجهة البرمجية.*

| الاصطلاح | القيمة |
|---|---|
| المصادقة | رمز Bearer JWT من Keycloak (نطاق DMRS). |
| صيغة مواردFHIR | application/fhir+json |
| صيغة النقاط غير التابعة لـFHIR | application/json |

| الاصطلاح | القيمة |
|---|---|
| صيغة الخطأ في مساراتFHIR | مورد.OperationOutcome |

## ٣-٦-٥ مثال تطبيقي

يوضّح المثال الآتي نداءً على خدمة CDS Hooks عند وصف دواءٍ لمريض، واستجابة النظام ببطاقة تنبيهٍ سريري:

*الشكل ٣-١٠: نموذج نداءٍ واستجابةٍ لخدمة دعم القرار CDS Hooks.*

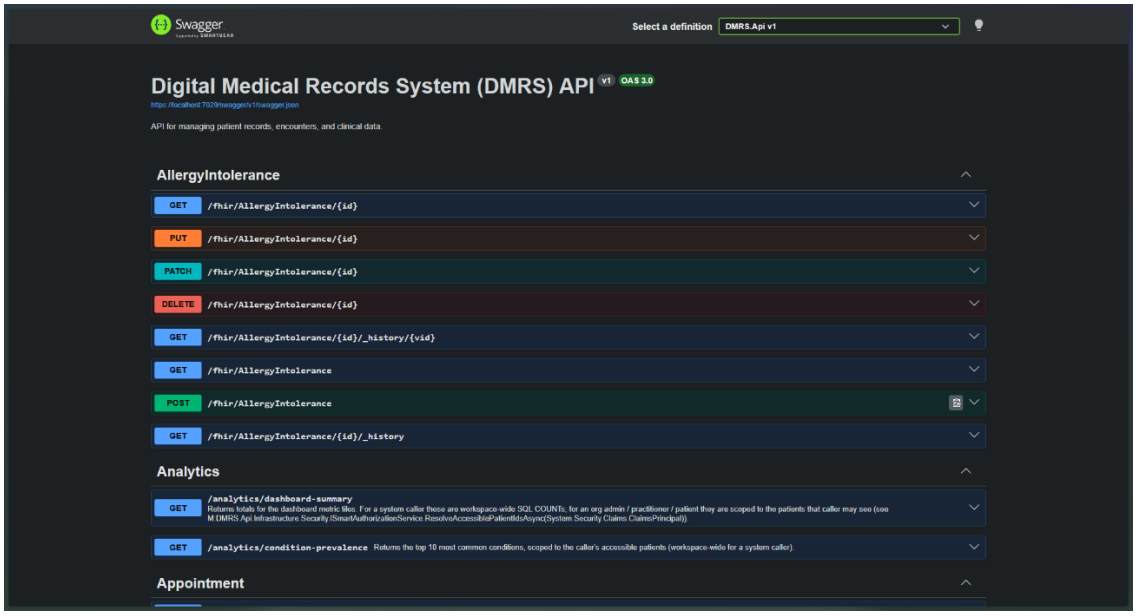
```
# الطلب
POST /cds-services/medication-prescribe
Authorization: Bearer <JWT>
Content-Type: application/json

{ "hook": "medication-prescribe",
  "context": { "patientId": "Patient/HM12345",
               "medications": [ { "reference": "MedicationRequest/new" } ] } }

# الاستجابة (200 OK)
{ "cards": [
    { "summary":   "تنبيه: تداخل دوائي محتمل",
      "indicator": "warning",
      "detail":    "قد يتفاعل الدواء الموصوف مع دواءٍ حالي للمريض.",
      "source":    { "label": "Nabd CDS" } } ] }
```

## ٣-٦-٦ التوثيق التفاعلي والمرجع الكامل

يوفّر النظام توثيقاً تفاعلياً للواجهة عبرOpenAPI / Swagger [٢٦]، يتيح استعراض نقاط النهاية واختبارها مباشرةً من المتصفّح على المسار (swagger/ متاحٌ في بيئة التطوير .(ويعرض الشكل الآتي هذه الواجهة:




*الشكل ٣-١١: واجهة التوثيق التفاعلي Swagger UI لواجهة النظام البرمجية.*

أمّا المرجع الكامل المفصّل لكلّ نقطة نهاية — بطرائقها ووسائطها وصيغ أجسامها — فمتوافرٌ ضمن الشيفرة المصدرية للمشروع على مستودع GitHub (انظر القسم ٦-١) في الملف docs/api-reference.md، على الرابط:

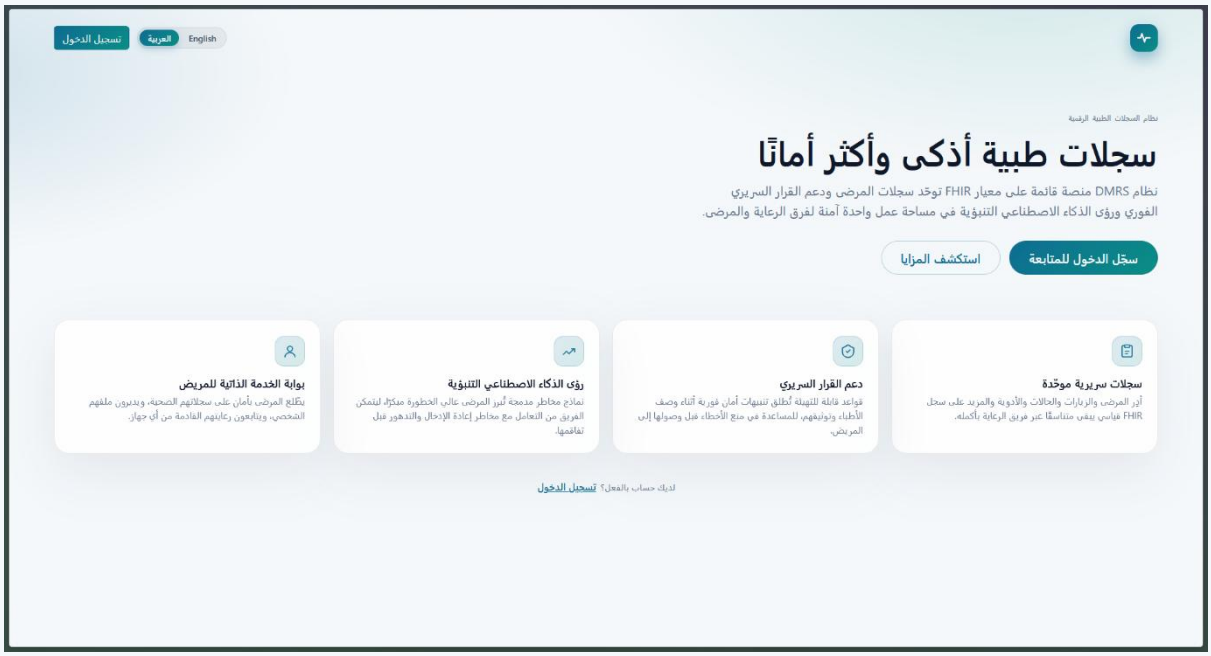
https://github.com/Khair69/DMRS/blob/master/docs/api-reference.md

# الفصل الرابع – التطبيق العملي

نعرض في هذا الفصل أبرز واجهات النظام مرتّبةً بحسب دور المستخدم، مع وصفٍ تفصيلي لكلّ شاشة (لقطة شاشة) يبيّن مكوّناتها ووظيفتها وتدفّق العمل عليها. ولأنّ هذا الكتاب يوثّق النظام، خُصّص لكلّ شاشةٍ موضعٌ مؤطَّر تُدرَج فيه لقطة الشاشة الفعلية، يعلوه تعليقٌ (caption) ويصاحبه وصفٌ يقوم مقام اللقطة ويتيح للقارئ تصوّر الواجهة وفهم دورها ضمن المنظومة. ونشير إلى أنّ جميع الواجهات متجاوبة وثنائية اللغة (عربي/إنجليزي) مع تبديلٍ فوري لاتجاه الكتابة (RTL/LTR)، وأنّ عناصر التنقّل تظهر لكلّ مستخدمٍ بحسب دوره وصلاحياته.

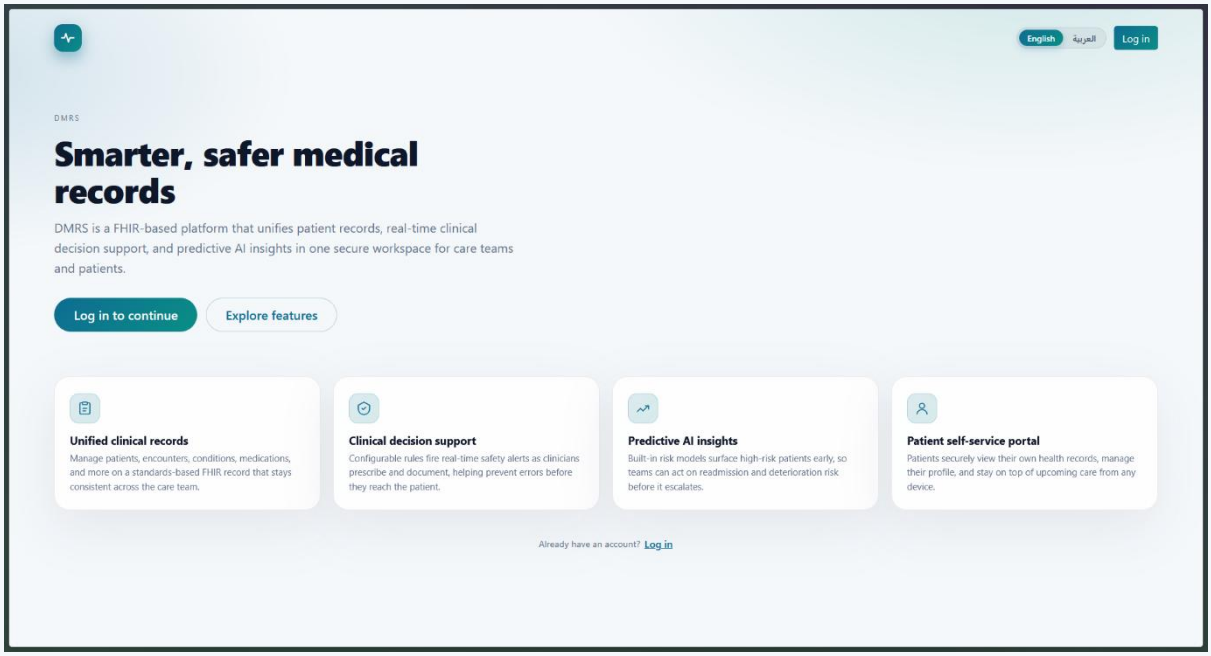
## ٤-١ واجهات الزائر والمصادقة

هي الواجهات التي يراها المستخدم قبل الدخول إلى مساحة العمل، وتقتصر على التعريف بالنظام وبدء تسجيل الدخول، من دون أيّ وصولٍ إلى البيانات السريرية.



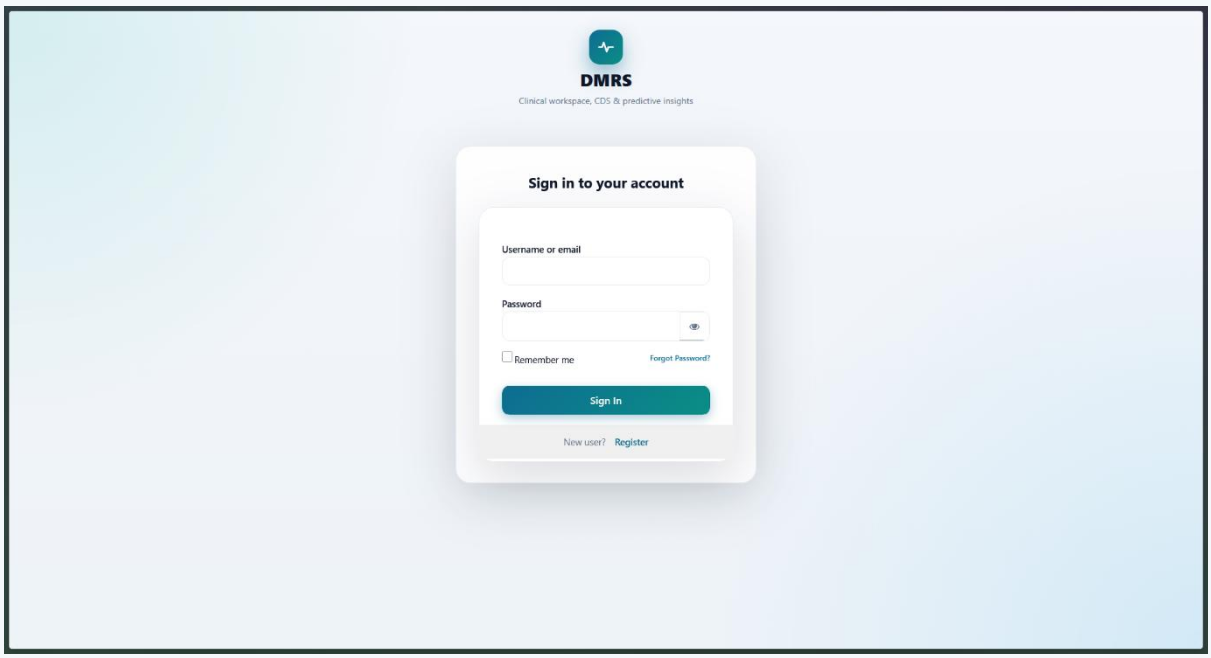

**الوصف:** صفحةٌ تعريفية بتصميمٍ عصري تظهر للزائر غير المسجّل ضمن تخطيطٍ خاص (LandingLayout). تتضمّن شريطاً علوياً يحمل اسم المنصّة ومكوّن تبديل اللغة وزرّ «تسجيل الدخول»، ثمّ قسماً رئيسياً (Hero) يعرض العنوان «سجلات طبية أذكى وأكثر أمانًا» مع وصفٍ موجز وزرَّي «تسجيل الدخول للمتابعة» و«استكشاف الميزات». يليه قسم بطاقات الميزات الأربع: السجلات السريرية الموحّدة، ودعم القرار السريري، ورؤى الذكاء الاصطناعي التنبؤية، وبوابة الخدمة الذاتية للمريض، لكلٍّ منها أيقونةٌ وعنوانٌ ووصف.

*الشكل ٤-١: صفحة الترحيب التعريفية بالنظام وبطاقات الميزات الرئيسية.*



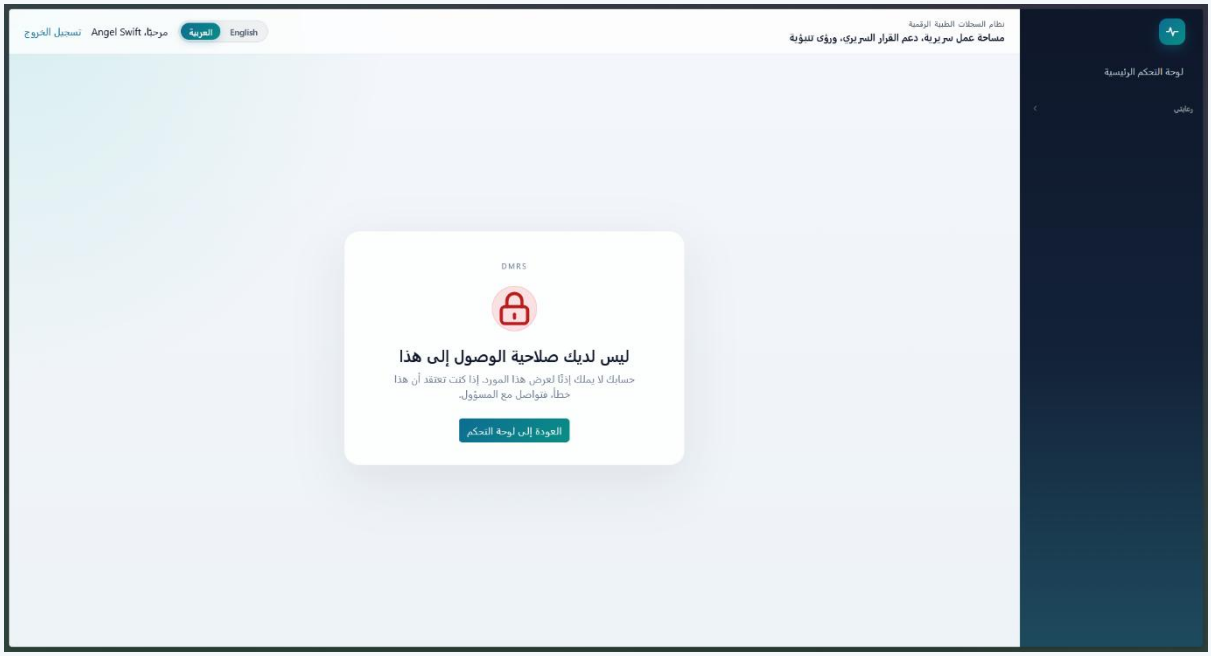

**الوصف:** زرٌّ في الشريط العلوي يتيح التبديل الفوري بين العربية والإنجليزية؛ وعند الاختيار تتغيّر جميع نصوص الواجهة ويُقلَب اتجاه الصفحة بين RTL وLTR من دون إعادة تحميل، مع حفظ تفضيل اللغة لاستخدامه في الزيارات اللاحقة. تظهر اللقطة الواجهة نفسها بالعربية ثمّ بالإنجليزية للمقارنة.

*الشكل ٤-٢: مكوّن تبديل اللغة وقلب اتجاه الواجهة بين العربية والإنجليزية.*



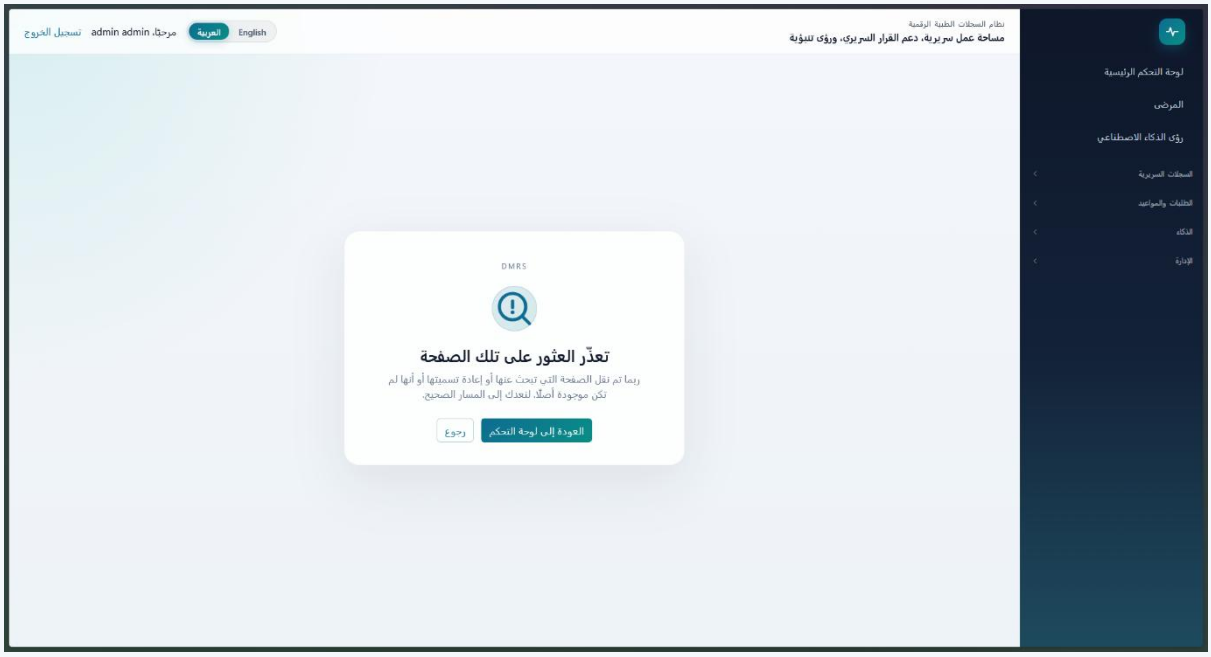

**الوصف:** عند النقر على «تسجيل الدخول» يُحوَّل المستخدم إلى صفحة Keycloak الآمنة لإدخال بريده وكلمة سرّه وفق تدفّق OAuth2/OIDC Authorization Code) مع PKCE). وبعد التحقّق يعود إلى مساحة العمل حاملاً رمز JWT الذي تُقرَأ منه الأدوار لتحديد ما يظهر له من أقسام. [١٨]

*الشكل ٤-٣: شاشة تسجيل الدخول الآمنة على خادم Keycloak.*




**الوصف:** صفحةٌ مصمّمة بعنايةٍ تظهر عند محاولة الوصول إلى موردٍ غير مصرّحٍ به لدور المستخدم، تعرض رسالةً واضحة وأيقونةً وزرّ عودةٍ إلى الصفحة الرئيسية، بما يحافظ على اتّساق التجربة وأمانها.

*الشكل ٤-٤: صفحة «رفض الوصول» عند انعدام الصلاحية.*

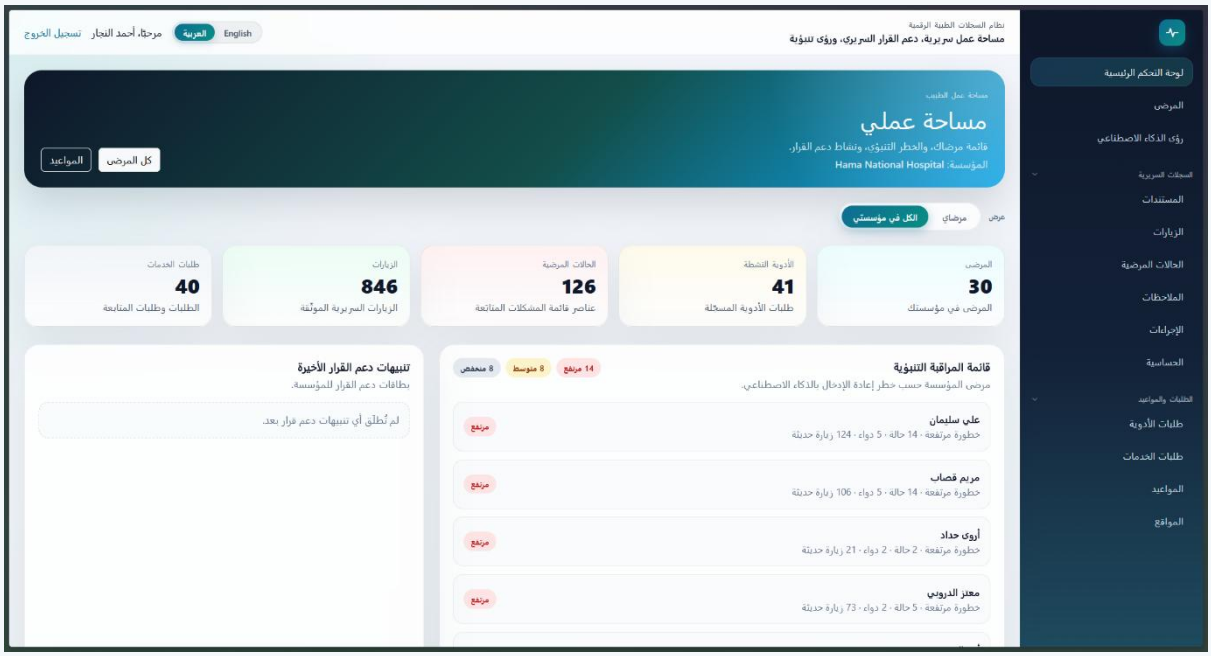



**الوصف:** صفحةٌ متّسقة مع هوية النظام تظهر عند طلب مسارٍ غير موجود، مع رسالةٍ ودودة ورابط عودة، تجنّباً لظهور أخطاء فجّة للمستخدم.

*الشكل ٤-٥: صفحة «الصفحة غير موجودة» (٤٠٤).*

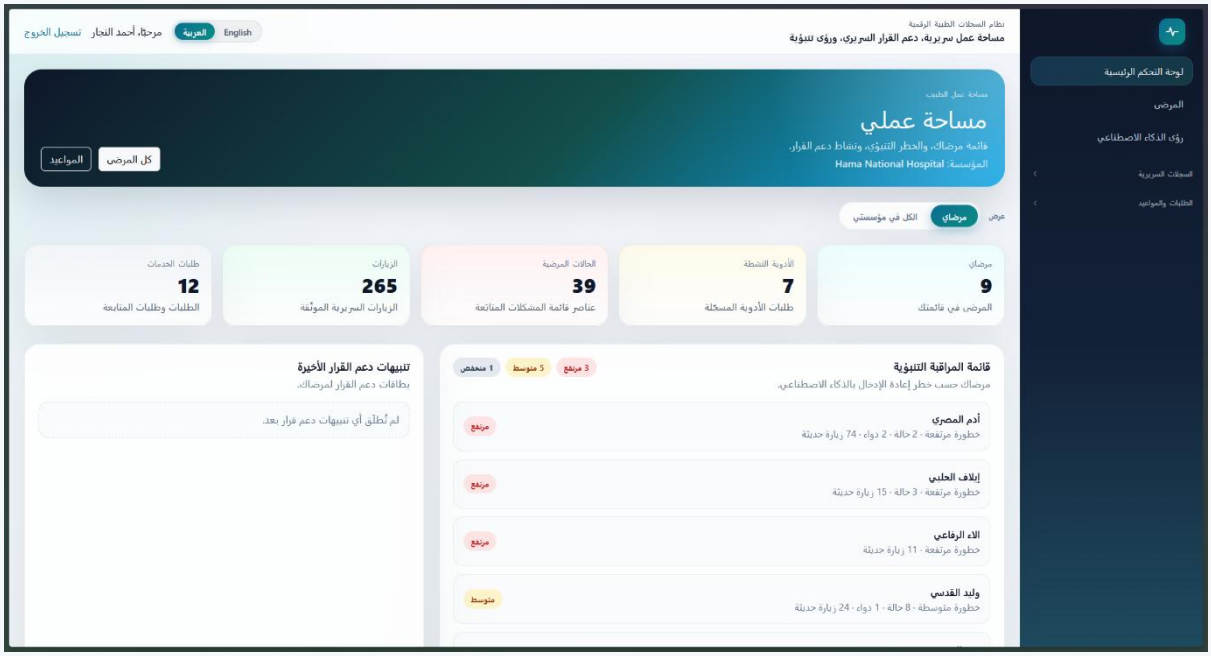
## ٤-٢ واجهات الطبيب/الممارس

تشكّل واجهات الطبيب جوهر مساحة العمل السريرية، وتغطّي اللوحة الرئيسية وإدارة المرضى ومخطّط المريض وتسجيل البيانات السريرية والطلبات والمواعيد واستعراض التاريخ والوثائق.



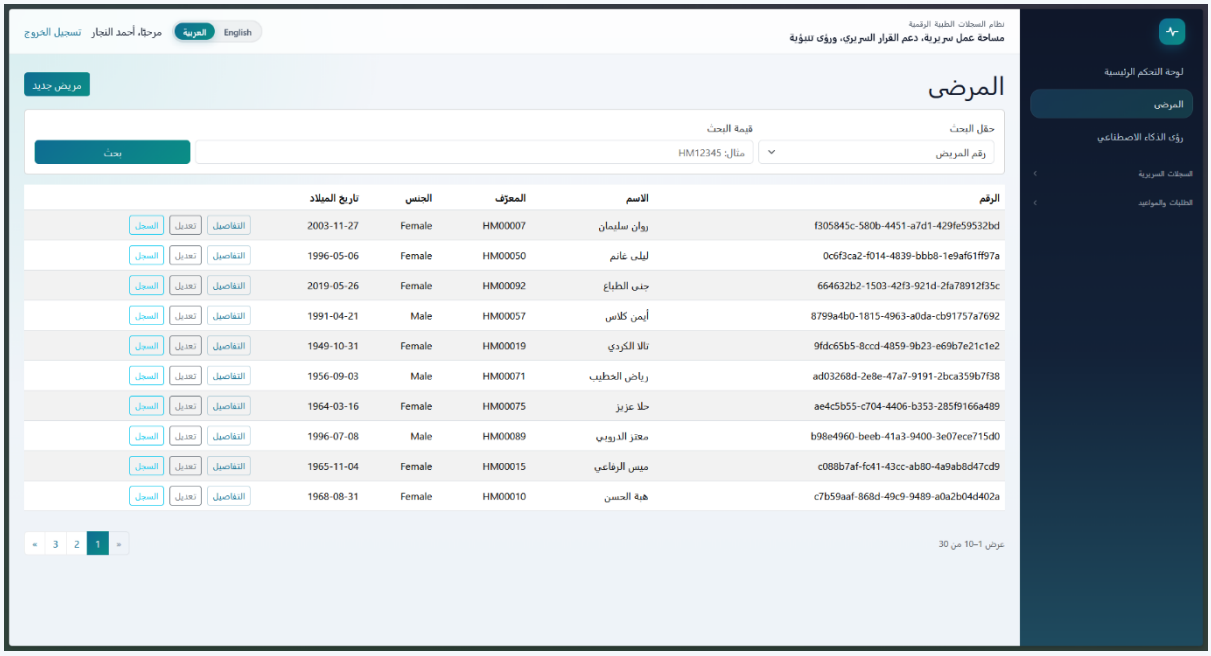

**الوصف:** شريطٌ جانبي قابل للطيّ ينظّم الوصول إلى أقسام النظام ضمن مجموعاتٍ منطقية: سريري (الزيارات والحالات والملاحظات والإجراءات والحساسيات والوثائق)، وطلبات (الأدوية وطلبات الخدمة والمواعيد والمواقع)، وذكاء (الرؤى الذكية والنماذج الخارجية)، ورعايتي (صحّتي وملفّي)، ومؤسستي (المؤسسة والكادر)، وإدارة (المؤسسات وورشة دعم القرار). ولا تظهر لكلّ مستخدمٍ إلا المجموعات التي يسمح بها دوره.

*الشكل ٤-٦: شريط التنقّل الجانبي مع المجموعات القابلة للطيّ.*



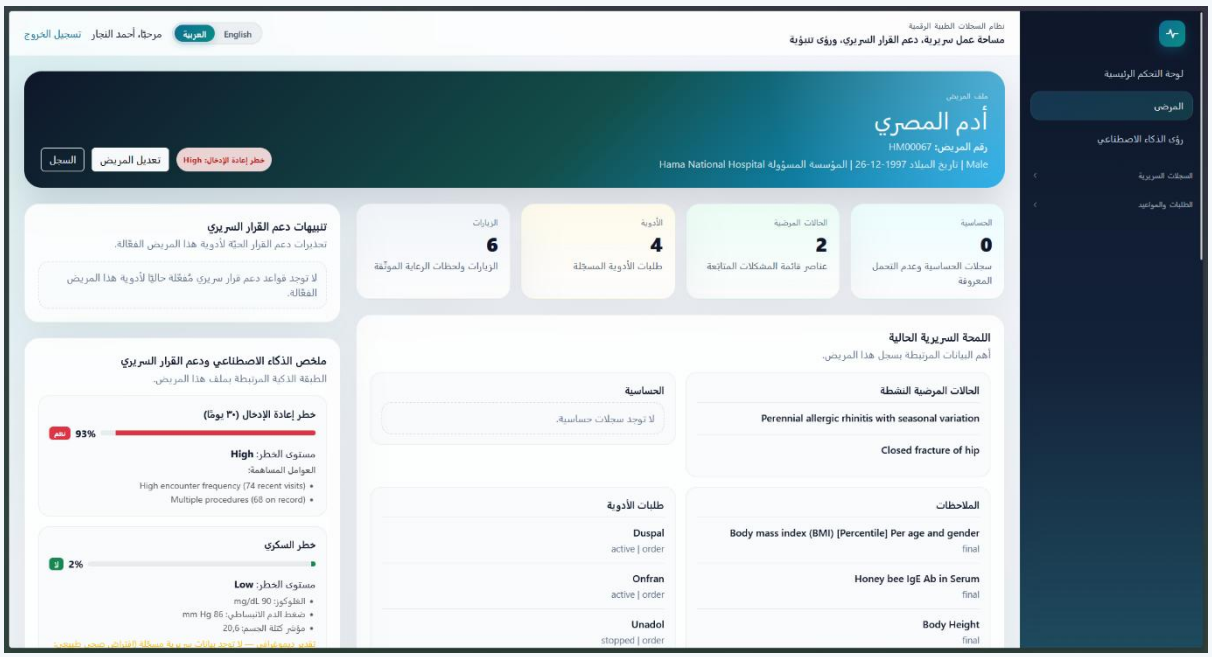

**الوصف:** تظهر بعد الدخول وتعرض بطاقات مؤشّراتٍ رئيسية (أعداد المرضى والمواعيد والأدوية...)، و«قائمة مراقبةٍ تنبّؤية» (Predictive Watchlist) ترتّب المرضى بحسب خطر إعادة الدخول، إضافةً إلى المواعيد القادمة وأحدث تنبيهات دعم القرار. ويتيح شريط التنقّل الجانبي مجموعاتٍ قابلة للطيّ (سريري، طلبات، ذكاء، رعايتي، مؤسستي، إدارة) تظهر بحسب دور المستخدم.

*الشكل ٤-٧: لوحة عمل الطبيب مع المؤشّرات وقائمة المراقبة التنبّؤية.*



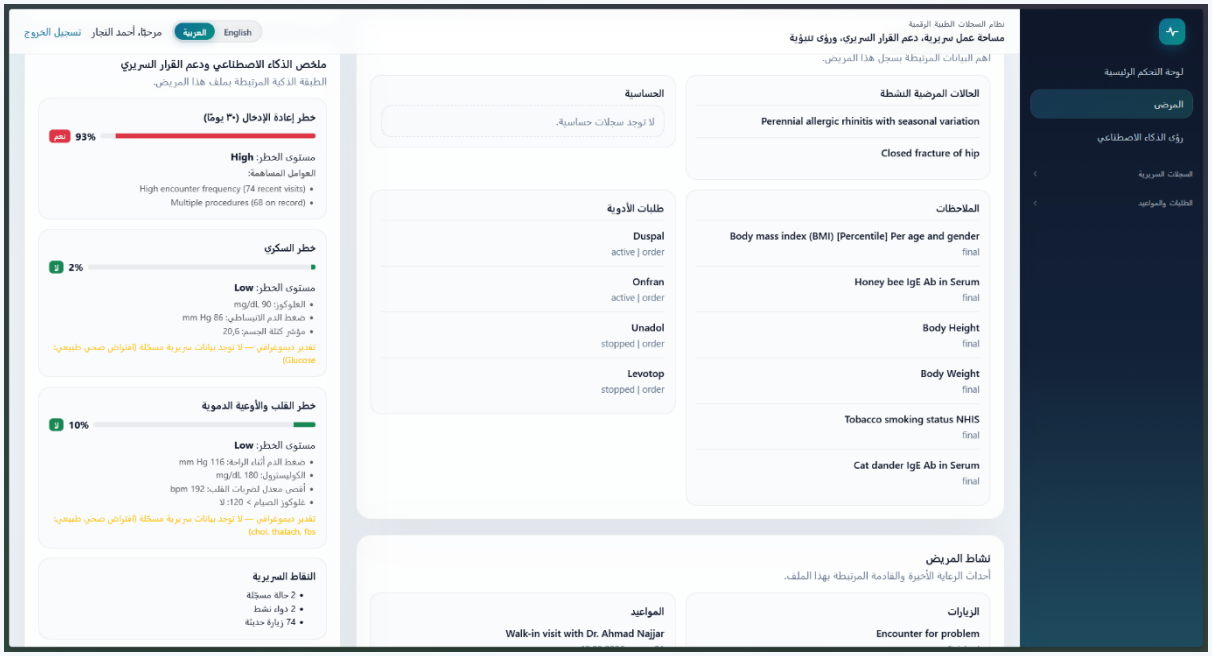

**الوصف:** تعرض قائمةً بالمرضى مع إمكانية البحث والترقيم الصفحي (Paginator) على جانب العميل، وأزرارٍ للإنشاء والتعديل والتفاصيل والتاريخ. يفتح النقر على مريضٍ مخطّطه الكامل. ويظهر لكلّ مريضٍ اسمه ومعرّفه المقروء وأبرز بياناته.

*الشكل ٤-٨: صفحة قائمة المرضى مع البحث والترقيم الصفحي.*



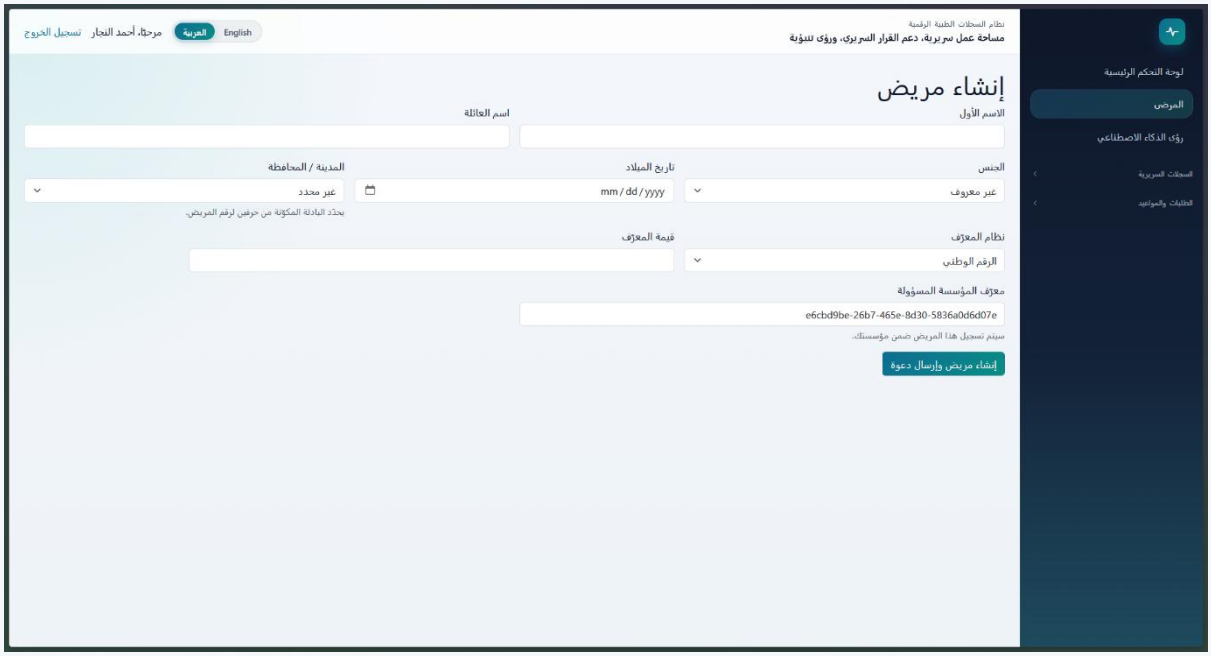

**الوصف:** هي أهمّ شاشةٍ سريرية؛ تلخّص مريضاً واحداً عبر ترويسةٍ ديموغرافية ومعرّفٍ مقروء، وبطاقات مؤشّرات (الحساسيات، الحالات، الأدوية)، وقوائم بأحدث العناصر السريرية، إضافةً إلى أزرار روابطٍ سريعة مثل «طلب دواء جديد» و«إضافة حالة» و«الرؤى الذكية».

*الشكل ٤-٩: مخطّط المريض: الترويسة والمؤشّرات والعناصر السريرية الحديثة.*




**الوصف:** ضمن مخطّط المريض تظهر لوحة «موجز الذكاء الاصطناعي ودعم القرار» وفيها بطاقات المخاطر الثلاث (إعادة الدخول، السكري، القلب)؛ تعرض كلّ بطاقةٍ نسبةً مئوية ومستوىً (منخفض/متوسط/مرتفع) والسمات التي استخدمها النموذج (مثل الغلوكوز وكتلة الجسم وعدد الأدوية)، وملاحظة «مُقدَّر/معوَّض» عندما تكون بعض المدخلات ناقصة للإشارة إلى يقينٍ أقل.

*الشكل ٤-١٠: بطاقات المخاطر الثلاث بالذكاء الاصطناعي على مخطّط المريض.*

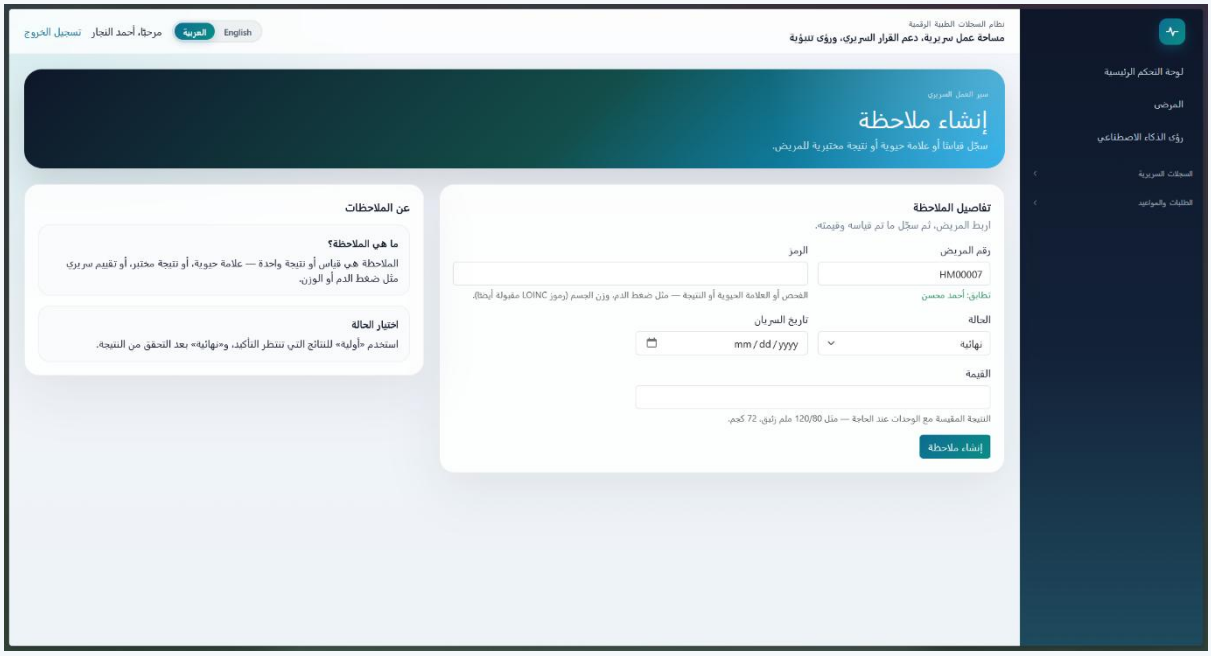



**الوصف:** نموذجٌ يجمع البيانات الديموغرافية للمريض (الاسم، الجنس، تاريخ الميلاد، معلومات التواصل) مع تحقّقٍ فوري من صحّتها وفق معيار FHIR قبل الحفظ. وعند الحفظ يُولَّد للمريض معرّفٌ مقروء على شكل هويةٍ (Identifier) إضافةً إلى معرّفه الداخلي.

*الشكل ٤-١١: نموذج إدخال/تعديل بيانات المريض مع التحقّق.*

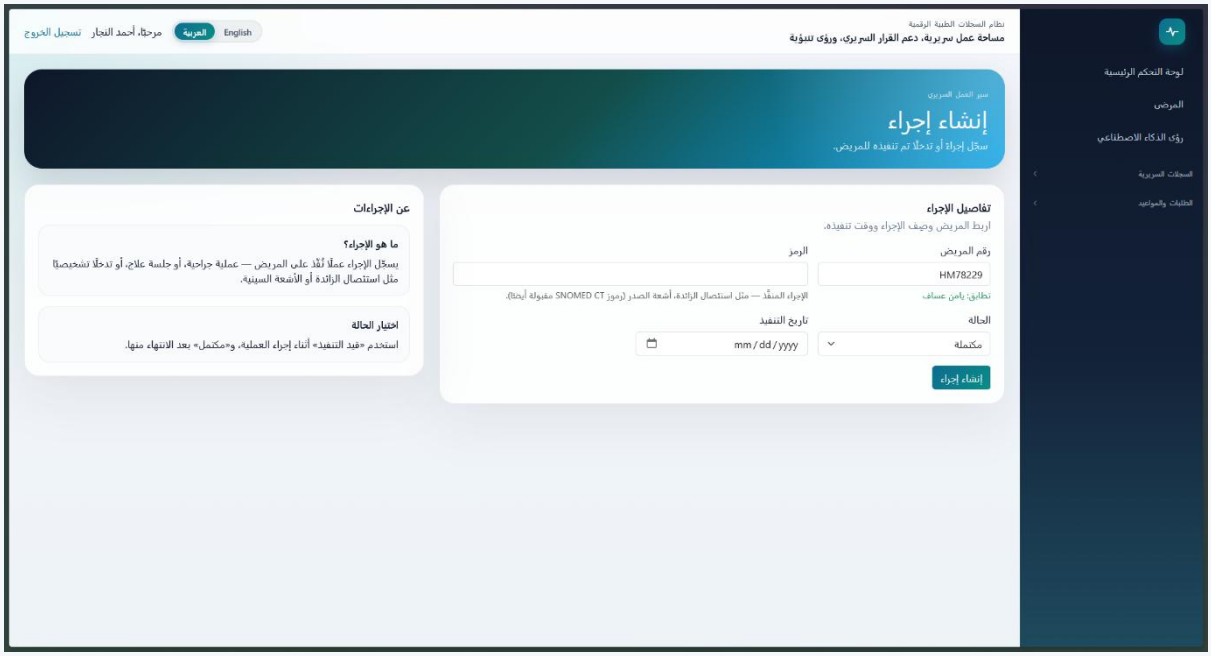
**الوصف:** نموذجٌ لتسجيل تشخيصٍ/حالة مرضية يتضمّن رمز الحالة وحالتها السريرية وتاريخها وربطها بالمريض، إلى جانب صفحة قائمةٍ تعرض حالات المريض المسجّلة. وتنعكس الحالة مباشرةً في المخطّط وتغذّي اللوحات ونماذج المخاطر.

*الشكل ٤-١٢: نموذج تسجيل حالة مرضية وقائمة الحالات.*



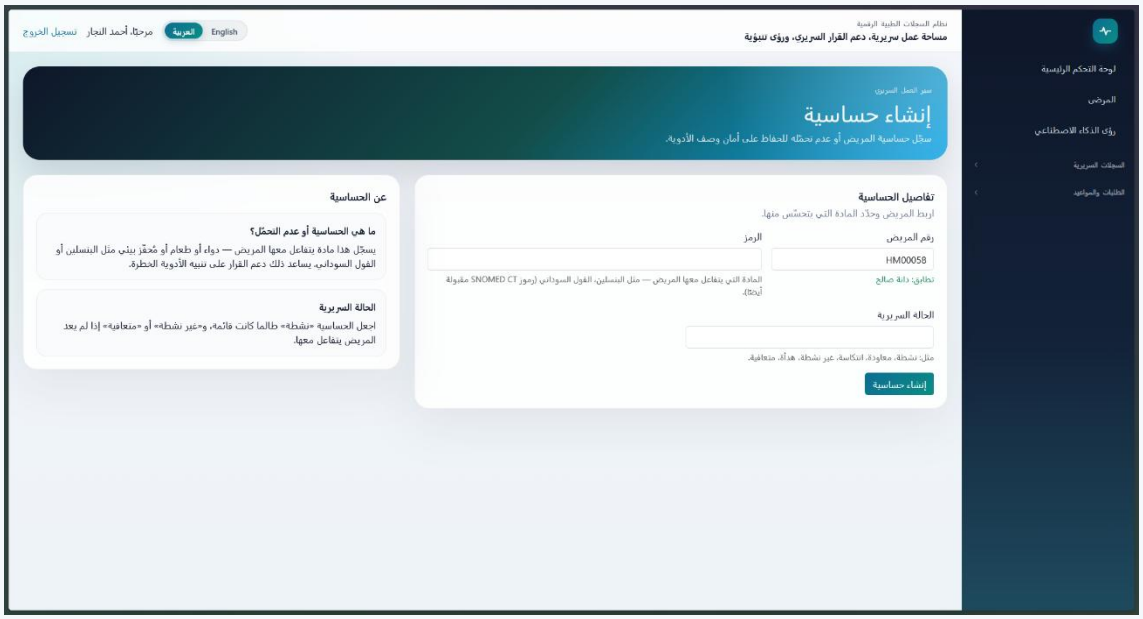

**الوصف:** نموذجٌ لإدخال قياسٍ أو نتيجة فحص (سكّر الدم، الضغط، كتلة الجسم...) باختيار نوع الملاحظة عبر رمز LOINC [٢٠] وإدخال قيمتها ووحدتها وتاريخها؛ وتُستخدَم هذه القيم لاحقاً في استخراج سمات نماذج المخاطر.

*الشكل ٤-١٣: نموذج تسجيل ملاحظة/قياس سريري.*




**الوصف:** نموذجٌ لتسجيل إجراءٍ نُفِّذ للمريض كمورد Procedure (نوع الإجراء وحالته وتاريخه والمريض المعني)، مع صفحات قائمةٍ وتفاصيلٍ وتاريخ، ويغذّي عددُ الإجراءات إحدى سمات نموذج إعادة الدخول.

*الشكل ٤-١٤: نموذج تسجيل إجراء طبي.*

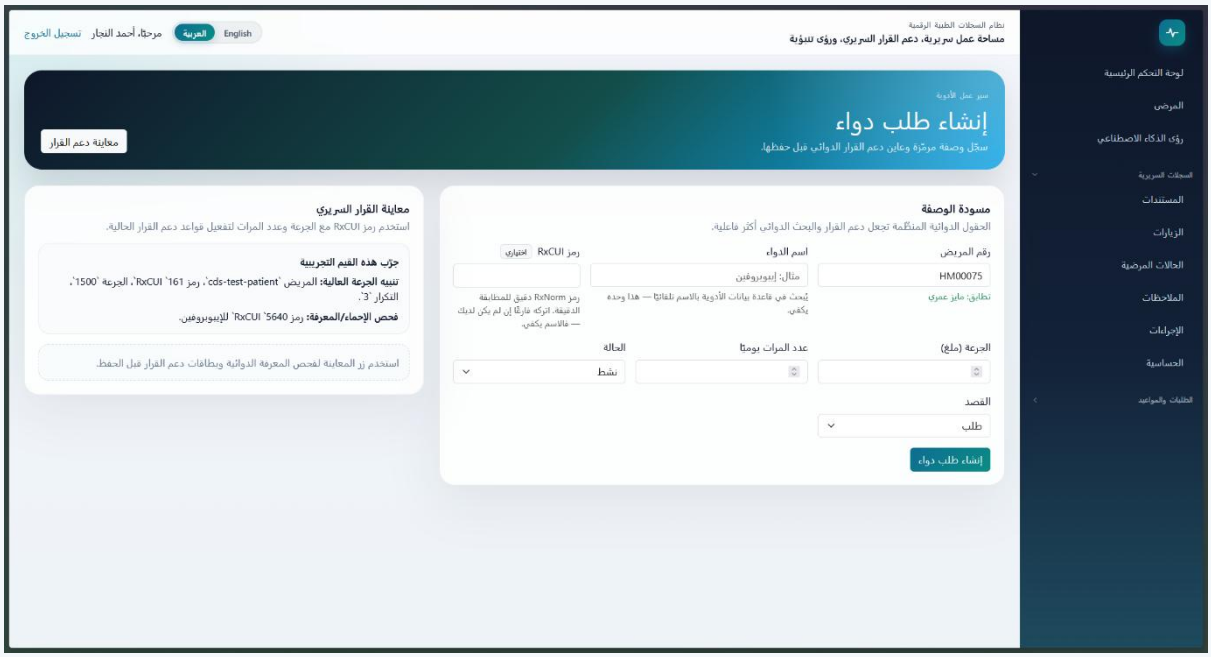



**الوصف:** نموذجٌ لتسجيل فرط حساسيةٍ تجاه مادةٍ أو دواء كمورد AllergyIntolerance، يظهر ضمن مؤشّرات مخطّط المريض ويُسهم في فحوص الأمان عند وصف الأدوية.

*الشكل ٤-١٥: نموذج تسجيل فرط الحساسية.*

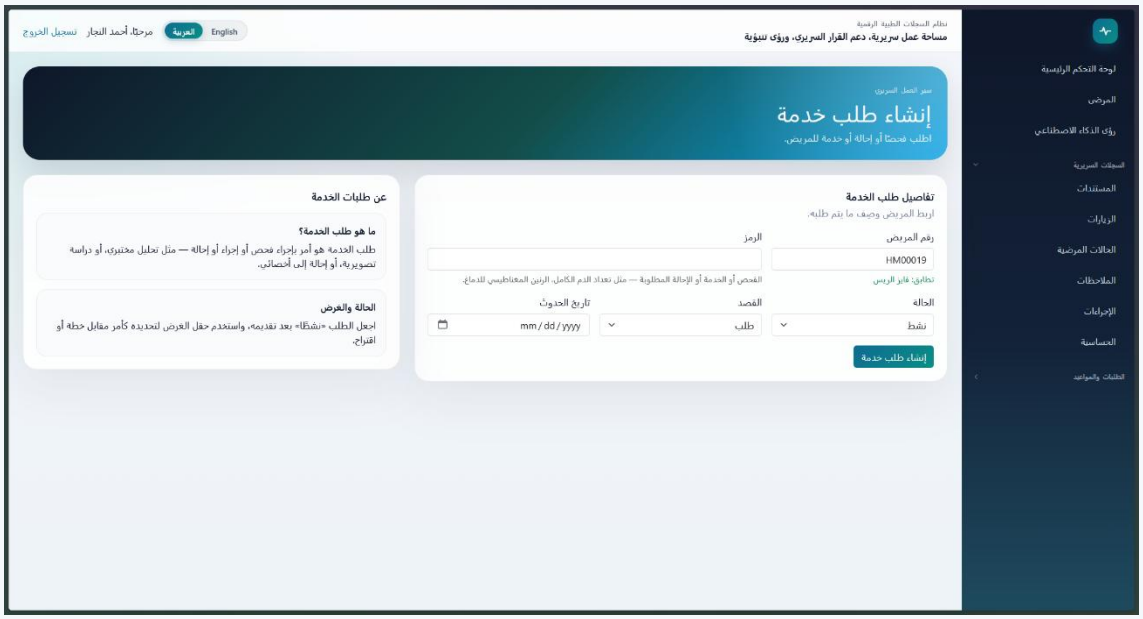
**الوصف:** نموذجُ لتوثيق زيارةٍ سريرية كمورد Encounter (نوعها وحالتها وفترتها والمشاركين والموقع)، ويُسهم عددُ الزيارات الحديثة في تقدير خطر إعادة الدخول.

*الشكل ٤-١٦: نموذج توثيق زيارة سريرية.*




**الوصف:** نموذجٌ لوصف دواءٍ يحدّد الدواء والجرعة والتعليمات. عند الحفظ يستعلم النظام عن معرفة الدواء ويقيّم قواعد دعم القرار، وقد تظهر بطاقة تنبيهٍ (مثل تحذير جرعةٍ زائدة أو تكرار مكوّنٍ دوائي أو خطر إعادة دخولٍ مرتفع) يقرّر الطبيب على ضوئها الإبقاء على الطلب أو تعديله. كما تُسجَّل البطاقة ضمن تنبيهات اللوحة الأخيرة.

*الشكل ٤-١٧: وصف الدواء مع ظهور بطاقة تنبيه دعم القرار.*

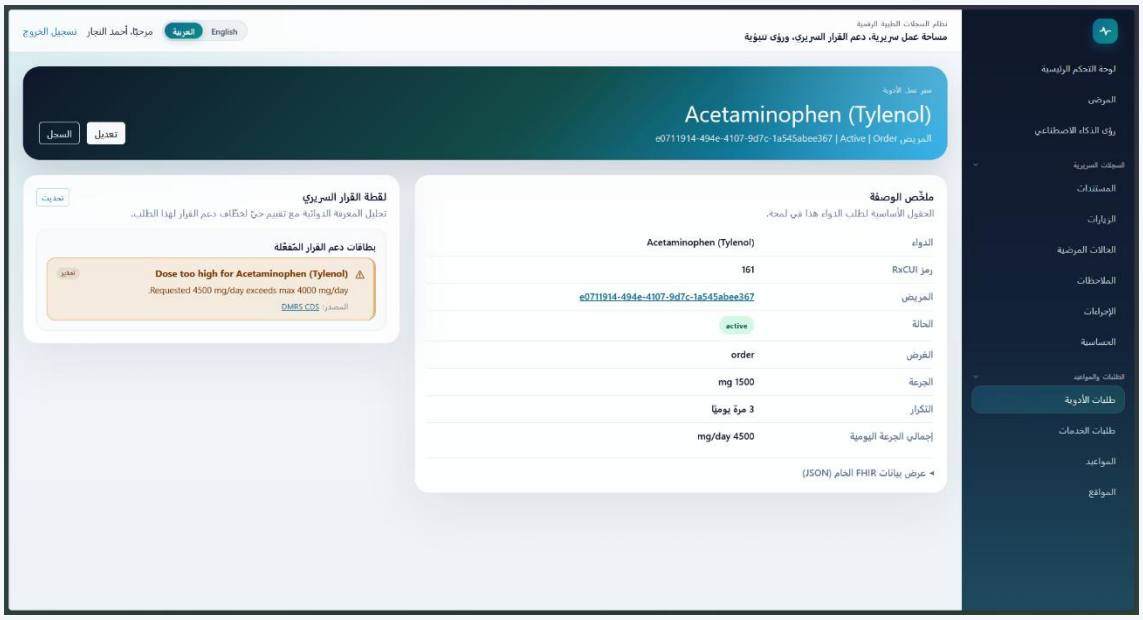



**الوصف:** شاشاتٌ لإنشاء طلبات الخدمة والإحالات (ServiceRequest) وإدارتها (قائمة، إنشاء، تعديل، تفاصيل، تاريخ)، وتربط الطلب بالمريض والمؤسسة المعنية.

*الشكل ٤-١٨: شاشة طلبات الخدمة والإحالات.*

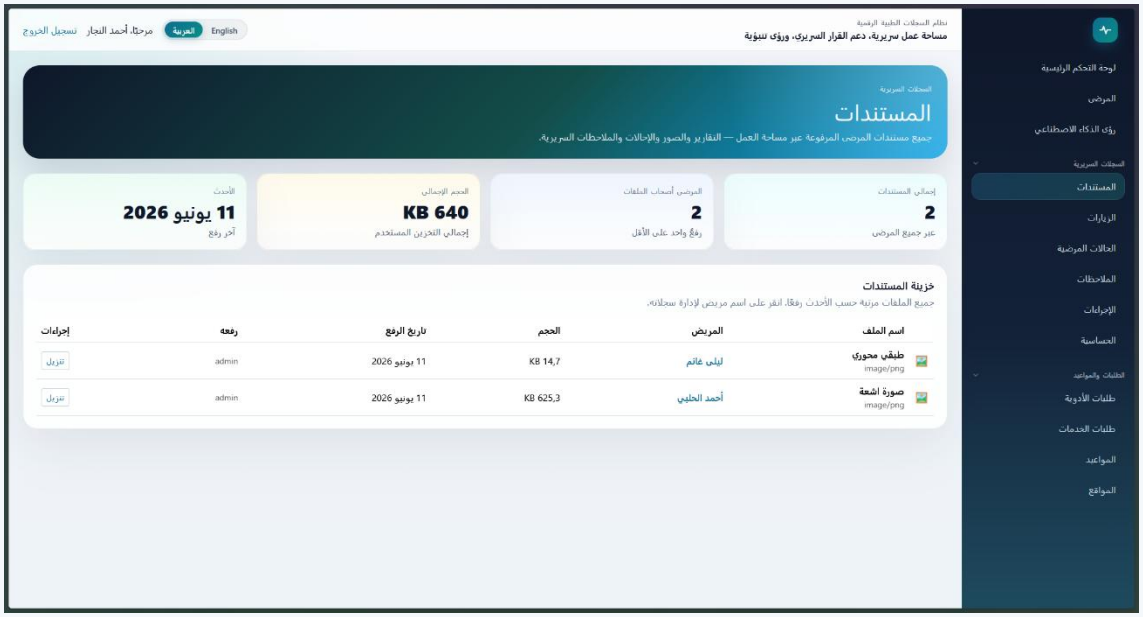
**الوصف:** شاشةٌ لجدولة المواعيد (Appointment) تحدّد الوقت والحالة، وتظهر المواعيد القادمة في لوحة العمل وفي بوابة المريض المعني.

*الشكل ٤-١٩: شاشة جدولة المواعيد وإدارتها.*




**الوصف:** صفحةٌ تعرض تفاصيل طلب دواءٍ للمريض (الدواء والجرعة والتكرار وإجمالي الجرعة اليومية)، وتُظهر بطاقة تنبيه دعم القرار السريري عند تجاوز الجرعة القصوى المسموحة.

*الشكل ٤-٢٠: تفاصيل طلب دواء مع ظهور بطاقة تنبيه دعم القرار.*

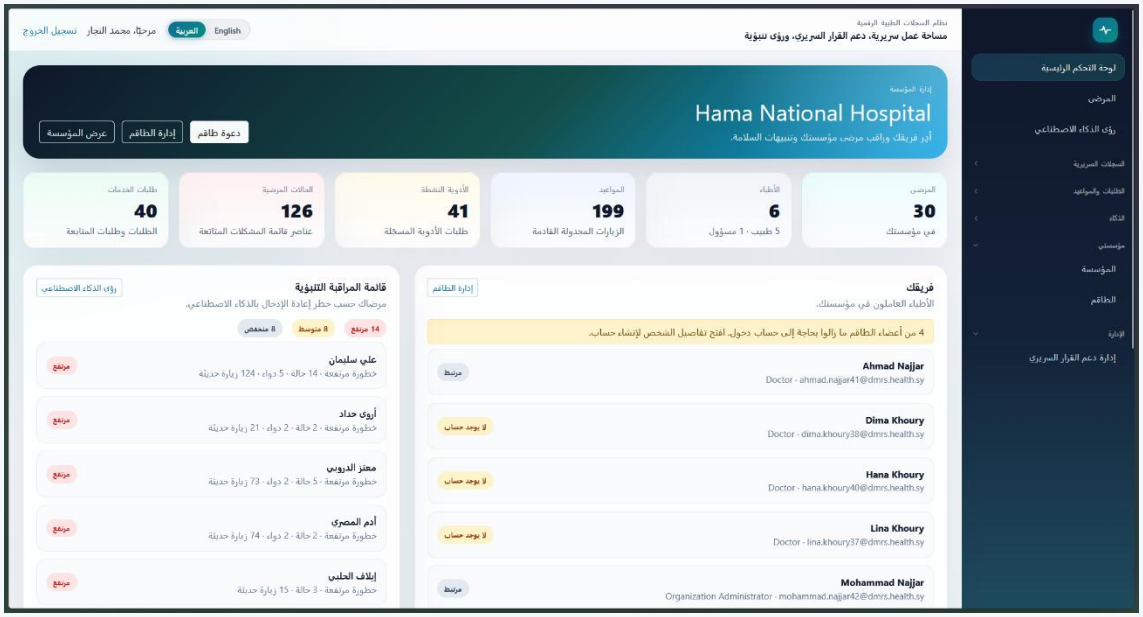


**الوصف:** لوحةٌ ضمن مخطّط المريض ترفع الملفّات السريرية (PDF/صور حتى ٢٠ ميغابايت) وتسردها مع إمكانية التنزيل أو الحذف، ما يجمع الوثائق المرتبطة بالمريض في مكانٍ واحد آمن.

*الشكل ٤-٢١: لوحة رفع وإدارة وثائق المريض.*

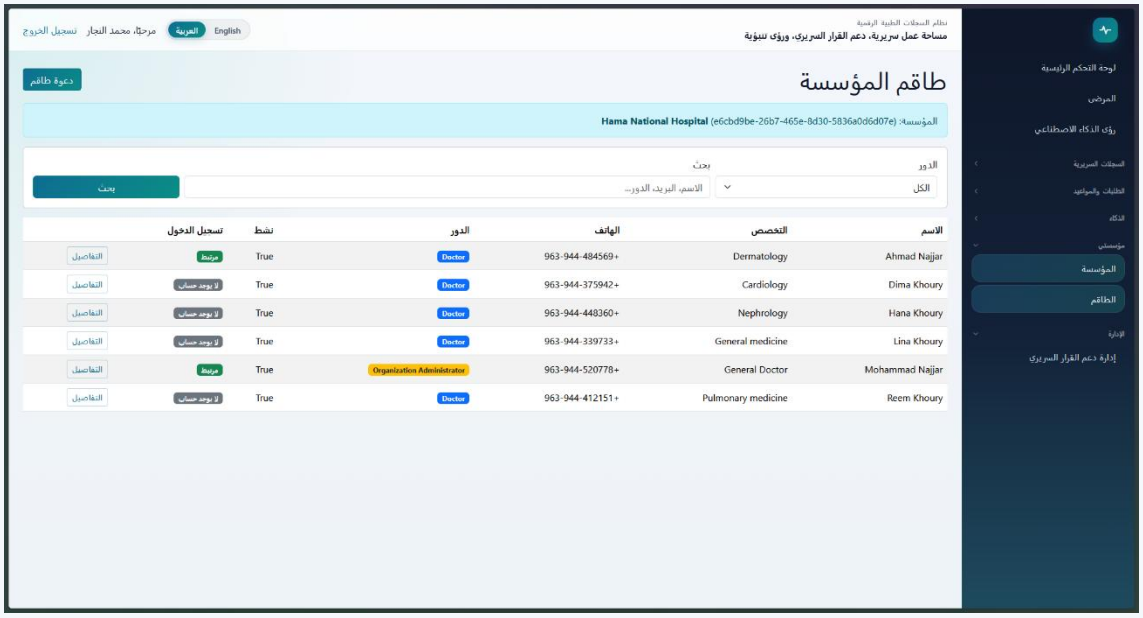
## ٤-٣ واجهات مدير المؤسسة

تضيف واجهات مدير المؤسسة قدراتٍ إدارية فوق القدرات السريرية، تشمل لوحته الخاصة وإدارة المؤسسة والكادر وآلية الدعوة والربط وورشة دعم القرار.




**الوصف:** لوحةٌ موجزة لمدير المؤسسة تعرض مؤشّرات مؤسسته (المرضى، الكادر، المواعيد، التنبيهات) ومداخل سريعة لإدارة الكادر والدعوات وورشة دعم القرار، بما يجمع المهامّ الإدارية في مكانٍ واحد.

*الشكل ٤-٢٢: لوحة عمل مدير المؤسسة.*

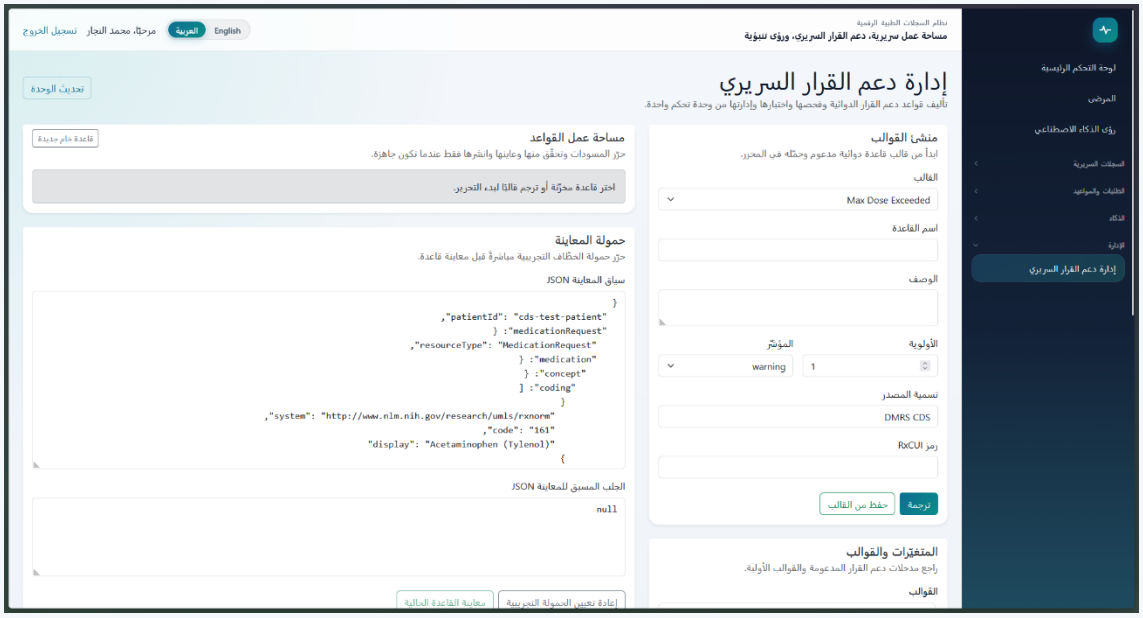



**الوصف:** جدولٌ يعرض كادر المؤسسة مع بحثٍ وتصفية، وأزرارٍ لعرض التفاصيل وإدارة الحسابات، ويمثّل لوحة المدير لمتابعة فريق العمل ضمن مؤسسته.

*الشكل ٤-٢٣: جدول إدارة كادر المؤسسة مع البحث.*

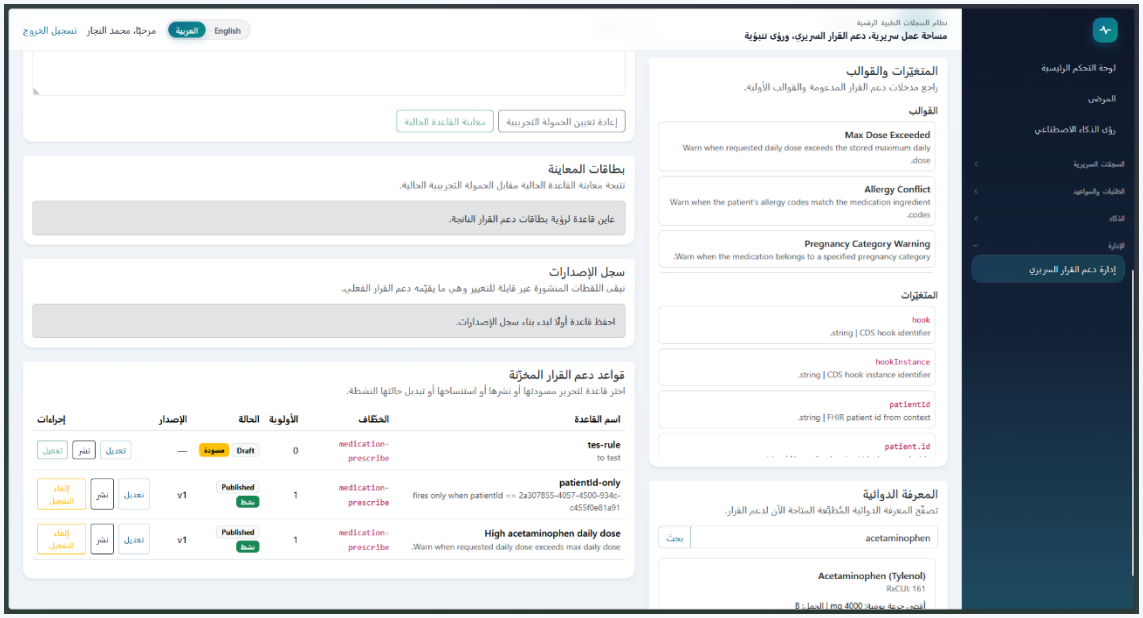
**الوصف:** نموذج دعوةٍ يملأ فيه المدير بيانات الكادر ويختار الدور، فيُنشَأ سجلّ FHIR للممارس ويُولّد رمز/رابط دعوة يُسلّم للشخص المعني لربط حسابه لاحقاً.

*الشكل ٤-٢٤: نموذج دعوة عضو كادر جديد.*




**الوصف:** ورشةٌ متكاملة لتأليف قواعد دعم القرار: تبدأ من قالبٍ جاهز (مثل «خطر إعادة الدخول» أو «تعدّد الأدوية» أو «الجرعات») أو من الصفر، وتحرّر تعبير الإطلاق (JSON-Logic [٢١]) على متغيّرات دعم القرار) ونصّ البطاقة، مع عرضٍ لقائمة المتغيّرات المتاحة.

*الشكل ٤-٢٥: ورشة تأليف قواعد دعم القرار – التحرير.*

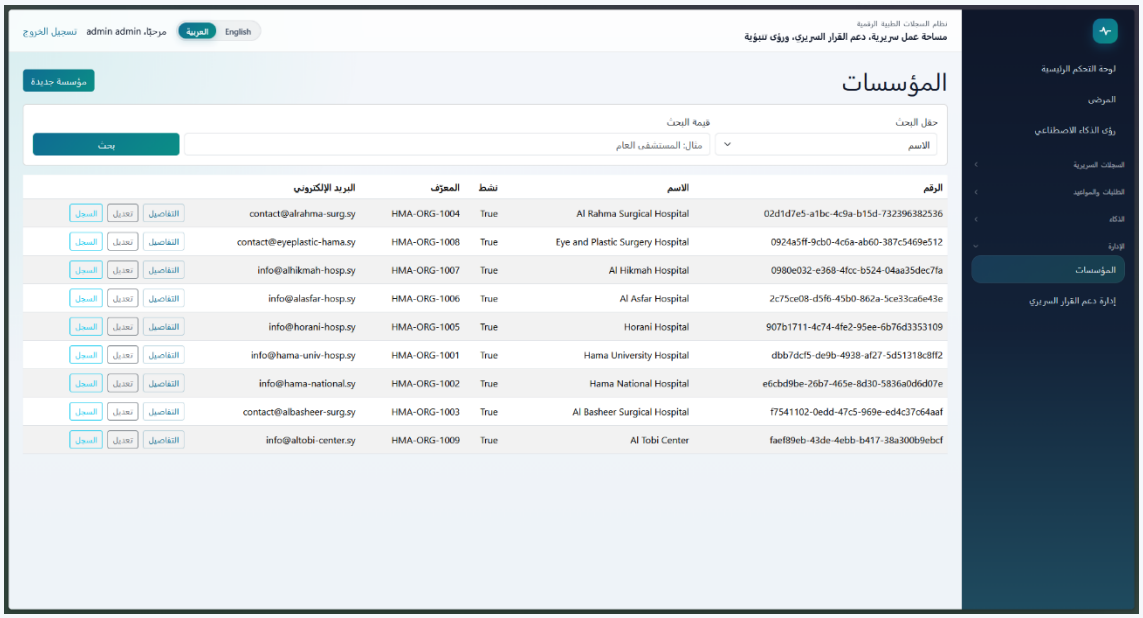



**الوصف:** بعد التحرير يتحقّق المحرّر من القاعدة ويعاينها على سياقٍ تجريبي ليبيّن ما إذا كانت ستُطلِق بطاقة، ثمّ ينشرها (نسخةً غير قابلة للتعديل) ويفعّلها لتعمل حيّاً، مع إمكانية إلغاء التفعيل أو الأرشفة، وكلّ نشرٍ مُصدَّر النسخ وقابل للاستعراض في التاريخ.

*الشكل ٤-٢٦: معاينة قاعدة دعم القرار ونشرها وتفعيلها.*

## ٤-٤ واجهات مدير النظام

يملك مدير النظام أعلى صلاحية، وتضاف إلى واجهاته إدارة المؤسسات على مستوى المنصّة وإدارة نماذج الذكاء الاصطناعي الخارجية.




**الوصف:** صفحةٌ تسرد جميع المؤسسات في المنصّة مع إمكانية البحث والترقيم، وأزرارٍ للإنشاء والتعديل والتفاصيل والتاريخ، وتمثّل المدخل الإداري لتنظيم المنصّة في مؤسساتٍ مستقلّة.

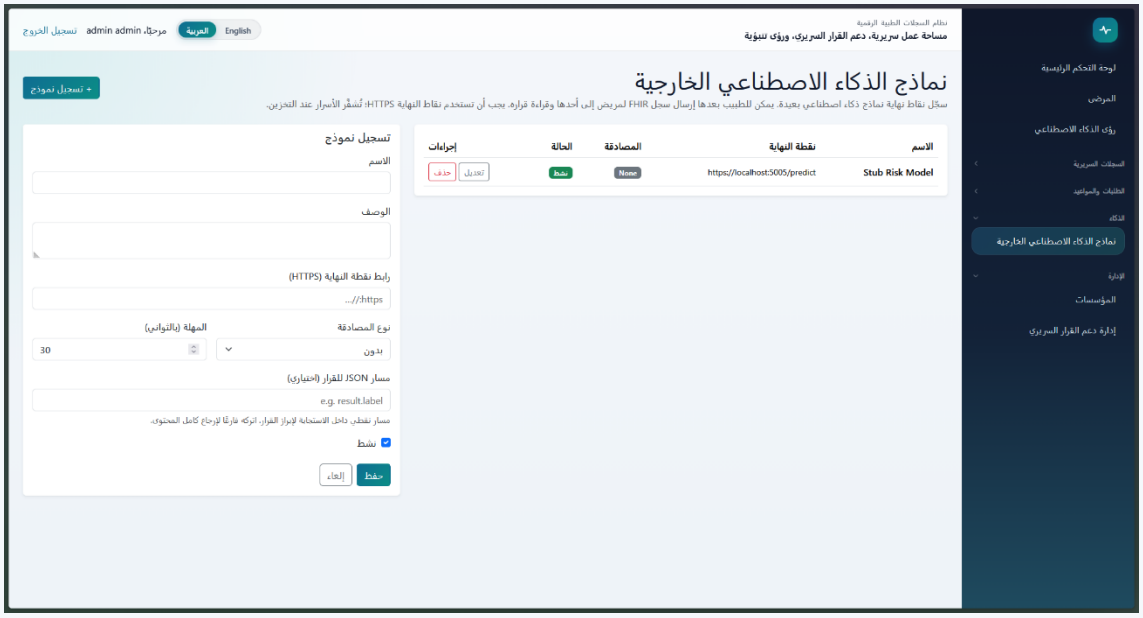
*الشكل ٤-٢٧: صفحة إدارة المؤسسات على مستوى المنصّة.*

**الوصف:** نموذجٌ لإنشاء مؤسّسةٍ جديدة على مستوى المنصّة وإدخال بياناتها الأساسية: الاسم والاسم المستعار والمعرّف ومعلومات التواصل والعنوان، ثمّ حفظها.

*الشكل ٤-٢٨: نموذج إنشاء مؤسّسة جديدة.*

**الوصف:** صفحةٌ تعرض تفاصيل المؤسسة (الاسم والمعرّف والحالة ومعلومات التواصل والعنوان)، مع مداخل لتعديلها وإدارة كادرها وإضافة مسؤولٍ لها واستعراض سجلّ تعديلاتها.

*الشكل ٤-٢٩: صفحة تفاصيل المؤسسة.*

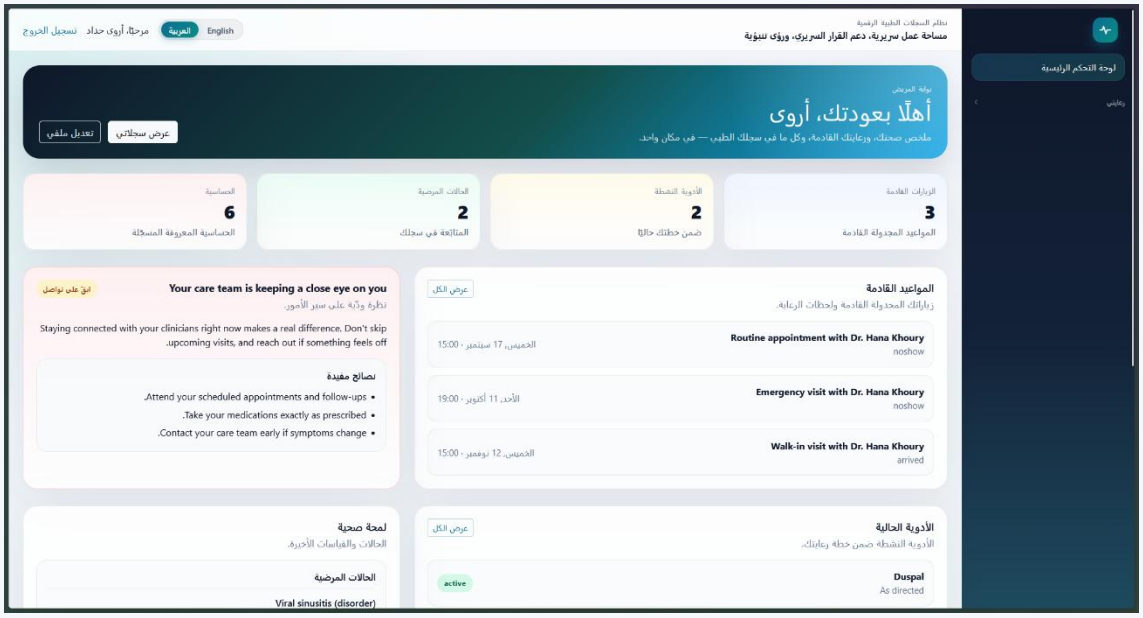



**الوصف:** لوحةٌ يسجّل فيها المدير نماذج ذكاء اصطناعي مستضافةً خارجياً: الاسم والوصف والرابط الآمن (HTTPS) ونوع المصادقة (لا شيء/مفتاح API/Bearer) واسم الترويسة والسرّ (يُخزَّن مشفّراً عبر ASP.NET Data Protection [٢٥] ولا يُعاد للعملاء) ومهلة الاتصال ومسار القرار في استجابة JSON وحالة التفعيل. وتعرض اللوحة النماذج المسجّلة مع إمكانية التعديل والتفعيل/التعطيل.

*الشكل ٤-٣٠: لوحة تسجيل وإدارة نماذج الذكاء الاصطناعي الخارجية.*

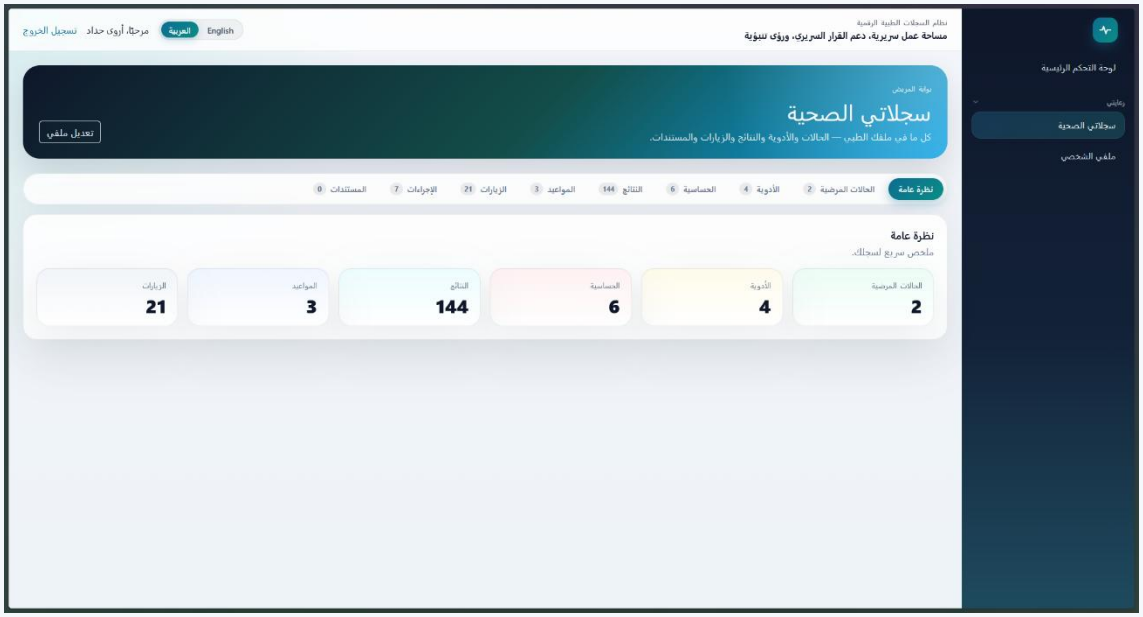
## ٤-٥ بوابة المريض

توفّر بوابة المريض تجربةً مبسّطة وآمنة يطّلع من خلالها المريض على سجلّه ضمن نطاق صلاحيته (Patient scope) فقط، فلا يرى سوى بياناته هو.



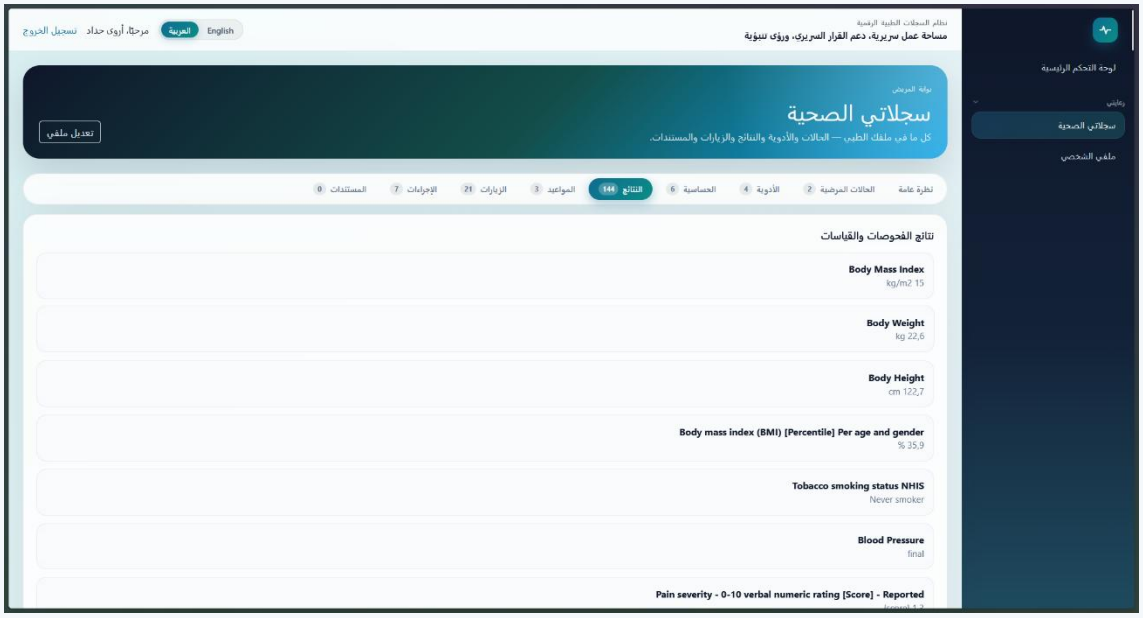

**الوصف:** لوحةٌ ترحيبية للمريض تعرض ملخّصاً سريعاً لحالته ومواعيده القادمة ومداخل إلى «صحّتي» و«ملفّي»، بأسلوبٍ مبسّط يناسب غير المختصّين.

*الشكل ٤-٣١: لوحة المريض الترحيبية.*




**الوصف:** تعرض ملخّص السجل الصحّي للمريض: حالاته المرضية وأدويته وملاحظاته ومواعيده القادمة بطريقةٍ مبسّطة. وتُقيَّد البيانات ضمن نطاق المريض فلا يرى سوى سجلّه هو.

*الشكل ٤-٣٢: بوابة «صحّتي» تعرض ملخّص سجل المريض.*

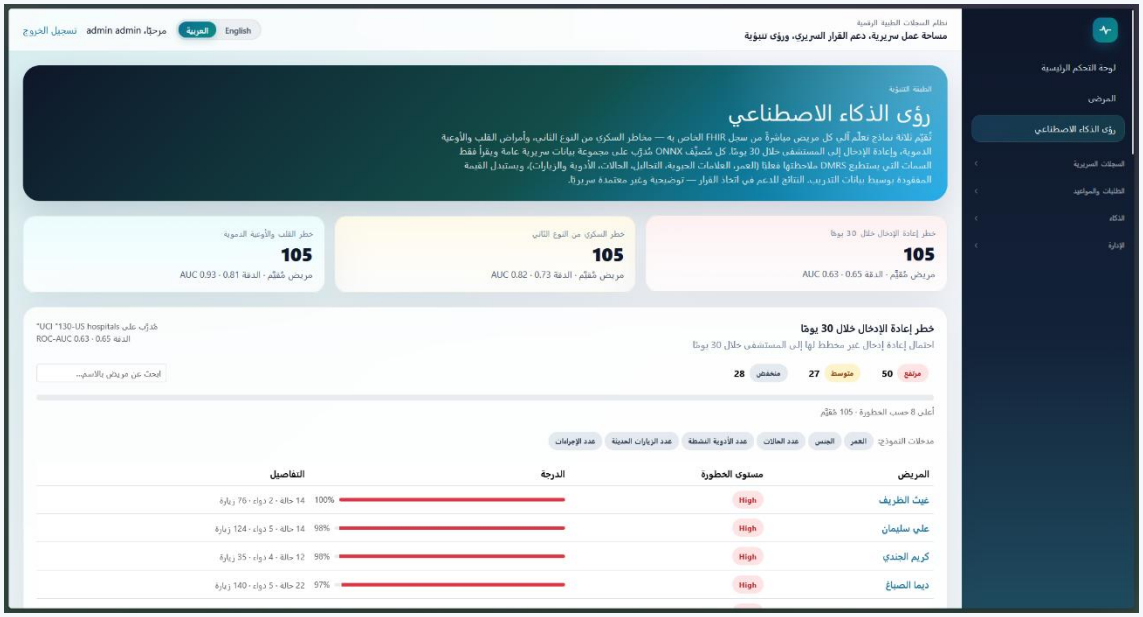



**الوصف:** تعرض ضمن بوابة «صحّتي» تبويب النتائج قائمةً بنتائج الفحوصات والقياسات السريرية للمريض مع قيمها ووحداتها، مثل مؤشّر كتلة الجسم والوزن والطول وضغط الدم وحالة التدخين.

*الشكل ٤-٣٣: تبويب نتائج الفحوصات والقياسات في بوابة «صحّتي».*

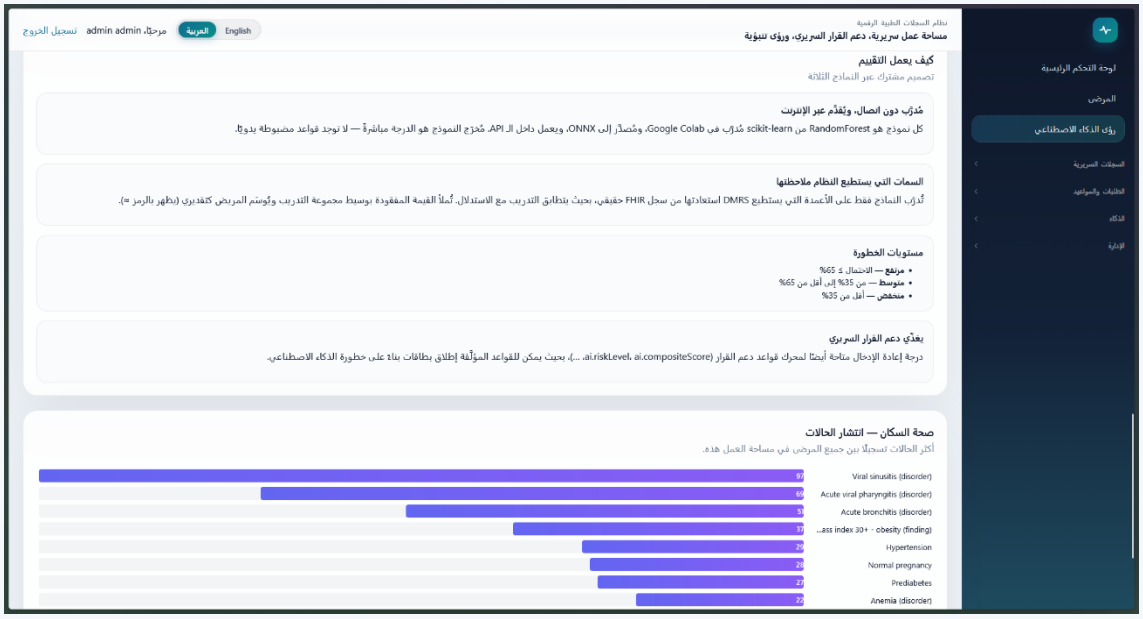
## ٤-٦ لوحة الرؤى الذكية AI Insights

تقدّم لوحة الرؤى الذكية نظرةً على مستوى المرضى تتجاوز المريض الواحد، وتجمع بين قائمة المراقبة والشرح والتحليل السكّاني.




**الوصف:** تعرض لكلٍّ من النماذج الثلاثة توزيع المخاطر عبر مجموعة المرضى وقائمة مراقبةٍ بالأعلى خطورة (محسوبة عبر مسارات الدفعات batch بطلبٍ واحد لتقليل عدد الطلبات)، ما يساعد الفِرق على ترتيب أولوياتها والتدخّل المبكر.

*الشكل ٤-٣٤: لوحة الرؤى الذكية: توزيع المخاطر وقوائم المراقبة.*




**الوصف:** يعرض القسم السفلي شرحاً مشتركاً «كيف يُحسَب الخطر» يوضّح السمات والتعويض والمستويات، إضافةً إلى تحليلٍ لانتشار الحالات المرضية في المنصّة (Population Health) يبيّن أكثر الحالات شيوعاً.

*الشكل ٤-٣٥: شرح آلية حساب الخطر وتحليل انتشار الحالات.*

بهذا تكون واجهات النظام قد غطّت كامل دورة العمل السريري والإداري، بدءاً من تعريف الزائر بالنظام، مروراً بعمل الطبيب اليومي وإدارة المؤسسة والمنصّة، وصولاً إلى تمكين المريض والرؤى الذكية على مستوى المرضى، ضمن تجربةٍ موحّدة ومتجاوبة وثنائية اللغة.

# الفصل الخامس – الخاتمة والآفاق المستقبلية

قدّم هذا المشروع نظاماً متكاملاً لإدارة السجلات الطبية الرقمية يجمع بين سجلٍّ قائم على معيار FHIR HL7 R5، ومحرّك دعم قرارٍ سريري قابل للتهيئة، وثلاثة نماذج ذكاءٍ اصطناعي تعمل على بيانات المريض الحقيقية، إضافةً إلى سجلٍّ لنماذج الذكاء الاصطناعي الخارجية، وبوابة خدمةٍ ذاتية للمريض، وواجهةٍ ثنائية اللغة. وبذلك يعالج المشروع مشكلتَي تشتّت البيانات وغياب الدعم الاستباقي للقرار، ويرفع كفاءة العمل الطبي وجودته، ويضع أساساً تقنياً موثوقاً وقابلاً للتوسّع نحو تطبيقاتٍ صحية أوسع.

وقد أثبت المشروع عملياً جدوى البناء فوق معيارٍ عالمي مثل FHIR في توحيد البيانات وضمان قابلية تبادلها، وجدوى دمج طبقة ذكاءٍ اصطناعي تعمل على سماتٍ يمكن للنظام استخراجها فعلاً من سجلّ المريض، بما يجعل التقديرات واقعيةً وقابلة للعرض في موضع الرعاية. كما أبرزت ورشة تأليف قواعد دعم القرار مرونة المنظومة في تكييف التنبيهات وفق احتياجات كلّ مؤسسة.

ومع اكتمال الوظائف الأساسية، تبقى أمام المشروع آفاقٌ واسعة للتطوير نوجز أبرزها فيما يأتي:

## الآفاق المستقبلية

### إضافة منظومة اختبارات آلية

بناء تغطيةٍ اختبارية للواجهات البرمجية ومحرّك دعم القرار وخدمات المخاطر، لرفع موثوقية النظام وضمان عدم تراجع الجودة مع تطوّره.

### النشر السحابي والحاويات

تهيئة النظام للنشر عبر الحاويات (Docker) [٢٣] مع إدارةٍ آمنة للأسرار وشهادات TLS وخطوط تكاملٍ ونشرٍ مستمر (CI/CD)، بما يسهّل التشغيل في بيئات الإنتاج.

### ربط مصدر بيانات الأدوية الحيّ

استبدال مزوّد معرفة الأدوية التجريبي بمصدرٍ حيّ مثل RxNorm [٢٢] لإثراء فحوص الجرعات والتداخلات الدوائية وتغطية أوسع للأدوية.

### توسيع نماذج الذكاء الاصطناعي

إضافة نماذج تنبّؤية جديدة وإعادة تدريب النماذج الحالية على بياناتٍ محلية حقيقية لرفع دقّتها، والاستفادة من سجلّ النماذج الخارجية لدمج خدمات ذكاءٍ متخصّصة.

## تطوير تطبيق موبايل

إتاحة تطبيقٍ محمول للمرضى والكادر الطبي يوسّع نطاق الوصول إلى النظام ويعزّز انخراط المريض في رعايته.

## تعزيز الأمان وحماية الخصوصية

إضافة سجلّ تدقيقٍ (Audit Log) للوصول إلى البيانات الصحية، وتشفير البيانات أثناء التخزين، بما يتوافق مع متطلّبات الخصوصية الصحية (مثل HIPAA وGDPR) قبل أيّ نشرٍ واقعي.

## توسيع التشغيل البيني

دعم استيراد/تصدير حِزَم (Bundles) FHIR [١]والتكامل مع أنظمةٍ صحية أخرى عبر مواصفات SMART on FHIR [١٩] الكاملة، بما يحوّل النظام إلى عقدةٍ ضمن منظومةٍ صحية أوسع.

# الفصل السادس – الملحق

## ٦-١ روابط المشروع

الشيفرة المصدرية الكاملة للنظام متاحةٌ على مستودع GitHub الآتي:

مستودع المشروع على GitHub: https://github.com/Khair69/DMRS

## ٦-٢ ملخّص حزمة التقنيات

يلخّص الجدول الآتي حزمة التقنيات المستخدمة في النظام بحسب الطبقة:

| الطبقة | التقنيات |
|---|---|
| الخادم (Backend) | ASP.NET Core ١٠, Entity Framework Core ١٠, #C |
| قاعدة البيانات | PostgreSQL (مع تخزين JSONB لموارد FHIR) |
| معيار البيانات الصحية | FHIR R5 HL7 عبر مكتبة .NET SDK Firely |
| الواجهة الأمامية | Blazor WebAssembly (.NET ١٠), Bootstrap, طبقة ترجمة عربي/إنجليزي |
| المصادقة والتفويض | Keycloak (OAuth٢ / OpenID Connect), رموز JWT, نطاقات SMART |
| الذكاء الاصطناعي | Python + scikit-learn (تدريب), ONNX Runtime, sklٍ٢onnx + ONNX (استدلال) |
| معرفة الأدوية | خدمة DMRS.MedicineInfo.Api (مع إمكانية RxNorm الحيّ) |
| أدوات مساندة | Serilog (التسجيل), ASP.NET Data Protection (حماية الأسرار), Swagger/OpenAPI |

## ٦-٣ موارد FHIR المدعومة

تغطّي الجوانب السريرية والإدارية، يلخّصها FHIR R5 يدير النظام مجموعةً من موارد [١] الجدول الآتي:

| المورد (Resource) | الدور في النظام |
|---|---|
| Patient | بيانات المريض الديموغرافية ومعرّفاته. |
| Practitioner / PractitionerRole | الكادر الطبي وأدوارهم ضمن المؤسسات. |
| Organization | المؤسسات الصحية المسجّلة في المنصّة. |
| Location | المواقع والمرافق التي تُربَط بها المواعيد والزيارات. |
| Encounter | الزيارات السريرية وفتراتها ومشاركوها. |
| Appointment | المواعيد المجدولة بين المرضى والكادر. |

| المورد (Resource) | الدور في النظام |
|---|---|
| Condition | التشخيصات والحالات المرضية للمريض. |
| Observation | القياسات ونتائج الفحوص (سكّر، ضغط، كتلة جسم...). |
| Procedure | الإجراءات الطبية التي نُفِّذت للمريض. |
| MedicationRequest | طلبات/وصفات الأدوية. |
| AllergyIntolerance | حالات فرط الحساسية تجاه المواد والأدوية. |
| ServiceRequest | طلبات الخدمة والإحالات. |

*جدول ٦-١: موارد FHIR R5 التي يديرها النظام.*

## ٦-٤ مسرد المصطلحات

نوضّح فيما يأتي أبرز المصطلحات التقنية الواردة في هذا الكتاب:

| المصطلح | التعريف |
|---|---|
| FHIR | معيار عالمي لتبادل البيانات الصحية يمثّل المفاهيم السريرية على شكل موارد قياسية. |
| CDS | دعم القرار السريري: تنبيهاتٌ وتوصيات تُعرَض للطبيب في موضع الرعاية. |
| CDS Hooks | مواصفة لربط أنظمة دعم القرار بأنظمة السجلات عند أحداثٍ معيّنة. |
| JSON-Logic | صياغةٌ منطقية بصيغة JSON تُستخدَم لتعريف شروط إطلاق قواعد دعم القرار. |
| ONNX | صيغةٌ مفتوحة لتمثيل نماذج تعلّم الآلة تتيح تشغيلها عبر بيئاتٍ مختلفة. |
| LOINC | نظام ترميزٍ عالمي للفحوص والقياسات المخبرية والسريرية. |
| OAuth٢ / OIDC | بروتوكولا تفويضٍ ومصادقة يعتمدهما خادم Keycloak. |
| JWT | رمز دخولٍ موقّع يحمل هويّة المستخدم وأدواره ويُتحقَّق منه في كلّ طلب. |
| SMART on FHIR | إطارٌ لتطبيق نطاقات صلاحياتٍ (System/User/Patient) فوق معيار FHIR. |
| RBAC | تحكّمٌ بالوصول قائم على الأدوار. |
| Soft Delete | حذفٌ منطقي (ناعم) يحتفظ بالسجل مع وسمه محذوفاً بدل إزالته فعلياً. |
| Synthea | أداةٌ لتوليد بيانات مرضى تركيبية لأغراض الاختبار والعرض. |

*جدول ٦-٢: مسرد بأبرز المصطلحات التقنية.*

## ٦-٥ تشغيل النظام (مختصر)

يتألّف تشغيل النظام من إعداد قاعدتَي بيانات PostgreSQL وتطبيق هجرات EF Core للقاعدة الرئيسية، وتشغيل خادم Keycloak مع مجالٍ باسم DMRS وعميلٍ للواجهة الأمامية، ثمّ تشغيل

الخدمات الثلاث بالترتيب: خدمة معلومات الأدوية، فالواجهة البرمجية الرئيسية (مع توثيق Swagger)، فالواجهة الأمامية. وتُحمَّل بيانات تجريبية (حِزَم Synthea) عبر أدوات التهيئة في وضع التطوير لتعبئة اللوحات ودعم القرار وبطاقات المخاطر.

# المراجع

[1] HL7 International. "HL7 FHIR Release 5 (R5) Specification." https://hl7.org/fhir/

[2] Firely. "Firely .NET SDK (HL7.Fhir) Documentation." https://docs.fire.ly/

[3] Microsoft. "ASP.NET Core Documentation." https://learn.microsoft.com/aspnet/core/

[4] Microsoft. "C# Language Documentation." https://learn.microsoft.com/dotnet/csharp/

[5] Microsoft. "Entity Framework Core Documentation." https://learn.microsoft.com/ef/core/

[6] The PostgreSQL Global Development Group. "PostgreSQL Documentation." https://www.postgresql.org/docs/

[7] Microsoft. "ASP.NET Core Blazor Documentation." https://learn.microsoft.com/aspnet/core/blazor/

[8] Keycloak. "Keycloak Server Documentation." https://www.keycloak.org/documentation

[9] Microsoft. "ONNX Runtime Documentation." https://onnxruntime.ai/

[10] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python." JMLR 12 (2011): 2825-2830. https://scikit-learn.org/

[11] ONNX. "sklearn-onnx (skl2onnx) Documentation." https://onnx.ai/sklearn-onnx/

[12] Bootstrap Team. "Bootstrap Documentation." https://getbootstrap.com/

[13] CDS Hooks. "CDS Hooks Specification." https://cds-hooks.org/

[14] UCI / Kaggle. "Pima Indians Diabetes Database." https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

[15] UCI Machine Learning Repository. "Heart Disease Data Set." https://archive.ics.uci.edu/dataset/45/heart+disease

[16] B. Strack et al. "Diabetes 130-US hospitals for years 1999-2008 Data Set." UCI ML Repository, 2014. https://archive.ics.uci.edu/dataset/296/

[17] MITRE. "Synthea: Synthetic Patient Population Simulator." https://syntheticheath.github.io/synthea/

[18] OpenID Foundation. "OpenID Connect Core 1.0 / OAuth 2.0." https://openid.net/connect/

[19] HL7 International. "SMART App Launch Framework." https://hl7.org/fhir/smart-app-launch/

[20] Regenstrief Institute. "LOINC — Logical Observation Identifiers Names and Codes." https://loinc.org/

[21] JsonLogic. "JsonLogic — Build complex rules, serialize them as JSON." https://jsonlogic.com/

[22] U.S. National Library of Medicine. "RxNorm." https://www.nlm.nih.gov/research/umls/rxnorm/

[23] Docker Inc. "Docker Documentation." https://docs.docker.com/

[24] Serilog. "Serilog — Structured logging for .NET." https://serilog.net/

[25] Microsoft. "ASP.NET Core Data Protection." https://learn.microsoft.com/aspnet/core/security/data-protection/

[26] SmartBear / OpenAPI Initiative. "OpenAPI Specification (Swagger)." https://swagger.io/specification/

[27] Microsoft. ".NET Platform Documentation." https://learn.microsoft.com/dotnet/

[28] W3C / WebAssembly Community Group. "WebAssembly Specification." https://webassembly.org/

[29] SNOMED International. "SNOMED CT." https://www.snomed.org/

[30] OpenMRS. "OpenMRS Documentation and FHIR Module (openmrs-module-fhir2)." https://openmrs.org/ , https://github.com/openmrs/openmrs-module-fhir2

[31] OpenEMR. "OpenEMR Features and FHIR API Documentation." https://www.open-emr.org/

[32] HAPI FHIR. "HAPI FHIR — The Open Source FHIR API for Java." https://hapifhir.io/

[33] Medplum. "Medplum Documentation — FHIR-native Developer Platform." https://www.medplum.com/

[34] Epic. "Epic on FHIR — Specifications." https://fhir.epic.com/

[35] Oracle Health. "Oracle Health (Cerner) Ignite APIs for FHIR." https://docs.oracle.com/health/