الجمهورية اليمنية
وزارة التعليم العالي والبحث العلمي
جامعة العلوم الحديثة
كلية الهندسة وتكنولوجيا المعلومات
قسم تقنية المعلومات

Republic of Yemen
Ministry of higher education
And scientific research
University of Modern Sciences
Faculty of Computer and
Information Technology
Department of information
technology

مشروع الملف الطبي الموحد في المستشفيات والمراكز الصحية في اليمن

A Unified Medical File in Hospitals and Health Centers in Yemen

مشروع تخرج مقدم إلى جامعة العلوم الحديثة – كلية الهندسة وتكنولوجيا المعلومات

كجزء من متطلبات نيل درجة البكالوريوس في كلية الهندسة وتكنولوجيا المعلومات – قسم تقنية المعلومات

للعام 2018–2019م

أسماء أعضاء الفريق:

1- أسامة عبده الهمداني
2- أنس احمد الحردي
3- ربيع عبدالله الدحومة
4- محمد عبدالله مظفر
5- يوسف حمود فرحان

إشراف:

د/ مجيب الأشول
أ/ نجيب البازلي

الجمهورية اليمنية
وزارة التعليم العالي والبحث العلمي
جامعة العلوم الحديثة
كلية الهندسة وتكنولوجيا المعلومات
قسم تقنية المعلومات

Republic of Yemen
Ministry of higher education
And scientific research
University of Modern Sciences
Faculty of Computer and
Information Technology
Department of information
technology

مشروع الملف الطبي الموحد في المستشفيات والمراكز الصحية في اليمن

A Unified Medical File in Hospitals and Health Centers in Yemen

مشروع تخرج مقدم إلى جامعة العلوم الحديثة – كلية الهندسة وتكنولوجيا المعلومات

كجزء من متطلبات نيل درجة البكالوريوس في كلية الهندسة وتكنولوجيا المعلومات – قسم تقنية المعلومات

للعام 2018–2019م

أسماء أعضاء الفريق:

1- أسامة عبده الهمداني
2- أنس احمد الحردي
3- ربيع عبدالله الدحومة
4- محمد عبدالله مظفر
6- يوسف حمود فرحان

إشراف:

د/ مجيب الأشول
أ/ نجيب البازلي

قال تعالى:

وَقُل رَّبِّ زِدْنِي عِلْمًا

صَدَقَ اللَّهُ العَظِيمُ

الإهداء

إلى وطننا الحبيب، إلى النور الذي ينير لنا درب النجاح إلى والدينا الغاليين الذين كانوا يضيئون لنا الطريق ويساندوننا أنتم وهبتمونا الحياة والأمل والنشأة على شغف الاطلاع والمعرفة، إلى كل من أضاء بعلمه عقل غيره إلى الشموع التي تحترق لتضيء للآخرين إلى دكاترتنا الفاضلين، إلى زملائنا ثم إلى كل من علمني حرفاً أصبح سنا برقه يضيء الطريق أمامي، إلى كل من وقف معنا.

الشكر والتقدير

نحمدك ربنا ونشكرك على أن يسرت لنا إتمام هذا المشروع على الوجه الذي نرجو أن ترضى به عننا، نتوجه بالشكر الجزيل الى عمادة كلية الهندسة وتقنية المعلومات ممثلة بالدكتور الفاضل نشوان الذبحاني و إلى د/ مجيب الاشول على الجهود الكبيرة التي بذلها بتوجيهاته لنا في تنفيذ المشروع، ونتوجه بالشكر الجزيل إلى أ/ نجيب مهدي البازلي أستاذنا الفاضل لك منا كل الثناء والتقدير لما قدمته لنا من نصائح ومعلومات ومساندة في تنفيذ المشروع فأنت أهل للشكر والتقدير فوجب علينا تقديرك فلك منا كل الثناء والتقدير، والشكر والتقدير ايضاً لكل معلمينا الذين رافقونا في سنواتنا الاربع فمنكم تعلمنا أن المستحيل يتحقق ، كما ونتقدم بالشكر ايضاً لكل الاهل والأصدقاء والزملاء والذين دعموا مسيرتنا في وقت الصعاب ووقفوا الى جوارنا بكل ما استطاعوا به ، كما لن ننسى ان نشكر كل جندي مجهول قدم لنا نصيحة او استشارة او مساعدة في وقت حاجة كان لها الأثر فيما وصلنا اليه اليوم وساعدنا في انجاز هذا المشروع.

من لا يشكر الناس لا يشكر الله ،،

والشكر اولاً واخيراً لمولانا عز وجل ولي نعمتنا ،،،،،،

والله الموفق ...

التعهد

يتعهد كلاً منا:

1- أسامة عبده الهمداني
2- أنس احمد الحردي
3- ربيع عبد الله الدحومة
4- محمد عبد الله مظفر
5- يوسف حمود فرحان

طلبة جامعة العلوم الحديثة بكلية الحاسوب وتقنية المعلومات قسم تقنية المعلومات نتعهد باننا نحن من بذل كل الجهد لإنجاز مشروع الملف الطبي الموحد في المستشفيات والمراكز الصحية في اليمن هذا بما فيه من مميزات وانه لم يتم النسخ من اي مصادر اخرى وسوف يتم التسليم في الوقت المحدد وأننا سوف نتحمل كافة المسؤوليات المترتبة للتنفيذ وتطبيق المشروع ولكم فايق الاحترام والتقدير.

الشهادة:

المشرف:

الأستاذ: نجيب مهدي البازلي.

ملخص المشروع:

مشروع الملف الطبي الموحد في المستشفيات والمراكز الصحية في اليمن

(A Unified Medical File in Hospitals and Health Centers in Yemen)

هو نظام يربط السجل الطبي في جميع المستشفيات الخاصة والعامة والعيادات في موقع ويب عبر الانترنت، حيث ان السجل الطبي الإلكتروني هو سجل إلكتروني لبيانات المريض الصحية ويتولد نتيجة الزيارات المتعددة لمركز الرعاية الصحية والمستشفيات والتاريخ الطبي للعلاج والبيانات المخبرية وتقارير الأشعة، ، كما يفيد الطبيب على الاطلاع على السجل الطبي للمريض واتخاذ التشخيص او القرار المناسب والتحديث المبني على المعرفة المسبقة للمضاعفات لسجل المريض وكذلك يسهل النظام الحصول على رقم سجل طبي موحد يسمح للمريض والطبيب الاطلاع على الملف الطبي من أي مكان عن طريق شبكة الإنترنت.

حيث سيكون الموقع سهل الاستخدام بحيث يمكن التعامل معه بسهولة ولا يتطلب مهارات متقدمة في الحاسوب عند استخدامه.

حيث قمنا ب استخدام ASP.net ، C#.net ،Android وهي من اللغات والتقنيات البرمجية الحديثة والتي تستخدم في بناء وتطوير المواقع والتي تتميز بالحماية العالية والمرونة وقاعدة بيانات (SQLServer) لإنشاء قاعدة البيانات حيث قامت بتطويرها شركة مايكروسوفت والتي تعد من أبرز الشركات العالمية في مجال البرمجيات، حيث تتميز بمميزات كثيرة ومن أهمها تخزين بينات كبيرة.

المحتوى

المحتويات

الإهداء ..... **III**

الشكر والتقدير ..... **IV**

التعهد ..... **V**

الشهادة: ..... **V**

ملخص المشروع: ..... **VI**

المحتوى ..... **VII**

فهرس الرسومات ..... **XI**

فهرس الجداول ..... **XIII**

الفصل الأول: المقدمة (Introduction) **1**

1.1 المقدمة: 2

1.2 وصف آلية سير النظام القائم: 3

1.3 مشكلة النظام القائم: 4

1.4 أهداف النظام 4

1.5 أهمية المشروع 5

1.6 تقنيات تجميع البيانات: 5

1.6.1 المقابلات الشخصية: 6

1.6.2 الملاحظات أو المشاهدات: 7

1.6.3 المستندات والوثائق للنظام الحالي: 8

1.7 منهجية البحث: 11

1.8 خطة المشروع: 12

الفصل الثاني مرحلة التعريف بالنظام (System Definition Phase) 13

2.1 المقدمة: 14

2.2 التقنيات المستخدمة في جمع البيانات 15

2.2.1 المقابلة الشخصية: 15

2.3 وصف النظام المقترح 20

2.3.1 تعريف النظام 20

2.3.2 مميزات النظام 20

2.3.3 وظائف النظام 21

2.4 وثيقة متطلبات النظام 22

2.4.1 أنواع متطلبات النظام: 22

2.5 دراسة الجدوى 24

2.5.1 دراسة الجدوى الاقتصادية (Economic Feasibility) 25

2.5.3 دراسة الجدوى التشغيلية (Operational Feasibility Study): 27

2.5.4 المخطط الزمني: 28

الفصل الثالث مرحلة توصيف العمليات (Process Modeling) 29

4.1 نمذجة العمليات (Data Flow Diagram) 30

4.1.1 مخطط السياق (Context Diagram) 35

4.1.2 المخطط الصفري (Level 0) 36

4.1.3 المخططات التفصيلية (Data Diagram) 37

4.2 وصف البيانات (Data Description Data Dictionary) 46

4.2.1 قاموس البيانات (Data Dictionary):.....46

4.2.2 ترميز البيانات (Data Encoding):.....51

4.3 وصف العمليات (الخوارزميات) (Process Description).....52

الفصل الرابع مرحلة توصيف البيانات (Data Modeling) 61

3.1 نمذجة البيانات (Data Modeling): - 62

الفصل الخامس مرحلة تصميم النظام (System Design Phase)..... 63

5.1 المقدمة: - 64

5.2 تصميم قاعدة البيانات (Designing Database):.....64

5.2.1 الجداول والحقول (Tables and Attributes): 64

5.2.2 تصميم القاعدة(Data Constrains) : 68

5.2.3 جمل الاستعلامات(Retrieve Statements): 71

5.3 تصميم واجهات المستخدم (Designing User Interface):.....74

5.3.1 هيكل الواجهات(Interface Structure):.....74

5.3.2 الواجهات الرئيسة(Main Screen): 75

5.3.3 واجهات الإضافة الى النظام(Add information Screens):.....78

5.3.4 استعلامات النظام(System Queries):.....85

5.3.5 تقارير النظام(System Reports): 91

الفصل السادس تصميم مخططات انسيابية للنظام والخوارزميات..... 93

6.1 خوارزمية الدخول للملف الطبي (عبر الويب او الإصدار المكتبي او الاندرويد): 94

6.2 خوارزمية إضافة بيانات للملف الطبي (عبر الإصدار المكتبي):.....95

6.3 خوارزمية تعديل البيانات للملف الطبي (عبر الإصدار المكتبي): 96

6.4 خوارزمية البحث عن الرقم للملف الطبي واستعراض البيانات: 97

الفصل السابع مرحلة تنفيذ النظام 98

7.1 تنفيذ النظام: – 99

7.2 اختبار النظام: – 99

الفصل الـثـامـــن الخاتمة.....104

8.1 الخلاصة:..... 105

8.1 الاستنتاجات:..... 106

8.1.1 الصعوبات: 106

8.1.2 الخبرات المكتسبة:..... 107

8.1 المقترحات والتوصيات: 108

8.1.1 المقترحات: 108

8.1.1 التوصيات: 108

8.1 الخاتمة: 109

المراجع (References).....110

الملاحق: 111

– كود النظام: 111

فهرس الرسومات

رسم توضيحي 1-1 دورة تطوير منهجية Agile.....11
رسم توضيحي 1-2 الخطة الزمنية للمشروع.....12
رسم توضيحي 3-2 انواع دراسة الجدوى 25
رسم توضيحي 3-4 مخطط Context Diagram 35
رسم توضيحي 3-5 مخطط Context 0.....36
رسم توضيحي 3-6 المخطط التفصيلي لتسجيل مستشفى Level 1 37
رسم توضيحي 3-7 المخطط التفصيلي لتسجيل مختبر Level 1 38
رسم توضيحي 3-8 المخطط التفصيلي لتسجيل طبيب Level 1.....39
رسم توضيحي 3-9 المخطط التفصيلي لدخول مستشفى Level 1 40
رسم توضيحي 10 -3 المخطط التفصيلي لدخول مختبر Level 1.....41
رسم توضيحي 3-11 المخطط التفصيلي لدخول طبيب Level 1.....42
رسم توضيحي 3-12 المخطط التفصيلي لدخول مريض Level 1 43
رسم توضيحي 3-13 المخطط التفصيلي لإدارة الموقع Level 1.....44
رسم توضيحي 3-14 المخطط التفصيلي للاشراف والرقابة Level 1.....45
رسم توضيحي 3-15 خوارزمية التسجيل (مستشفى،مختبر،طبيب).....53
رسم توضيحي 3-16 خوارزمية انشاء سجل طبي (مريض) 54
رسم توضيحي 3-17 خوارزمية الدخول (مستشفى، مختبر، طبيب).....55
رسم توضيحي 3-18 خوارزمية الدخول (مريض).....56
رسم توضيحي 3-19 خوارزمية إضافة للملف الطبي (طبيب) 57
رسم توضيحي 3-20 خوارزمية إضافة للملف الطبي (مختبر).....58
رسم توضيحي 3-21 خوارزمية التعديل على الملف الطبي (طبيب) 59
رسم توضيحي 3-22 خوارزمية التعديل على الملف الطبي (مختبر).....60
رسم توضيحي 4-23 المخطط العلائقي لقاعدة البيانات ERD.....62
رسم توضيحي 5-24 هيكل الواجهات 74

رسم توضيحي 25-5 الشاشة الرئيسية للإصدار الكمبيوتر (C#.net) 75
رسم توضيحي 26-5 الشاشة الرئيسية للإصدار الويب (Asp.net) 76
رسم توضيحي 27-5 شاشة الرئيسية للإصدار الهاتف (Android) 77
رسم توضيحي 28-5 شاشة إضافة سجل طبي 82
رسم توضيحي 29-5 شاشة استعلام عن بيانات المريض من خلال الرقم الطبي عبر نسخة الكمبيوتر 85
رسم توضيحي 30 -5 شاشة استعلام عن بيانات المريض من خلال الرقم الطبي عبر نسخة الويب 86
رسم توضيحي 31-5 شاشة استعلام عن بيانات المريض من خلال الرقم الطبي عبر نسخة (Android) 87
رسم توضيحي 32-5 تقرير عن ملف المريض..... 91
رسم توضيحي 33-5 تقرير عن التاريخ الطبي للمريض 91
رسم توضيحي 34 -5 تقرير لكل الزيارات المرضية..... 92
رسم توضيحي 35 -5 تقرير لكل الادوية الخاصة بالجلسة 92
رسم توضيحي 36 -6 خوارزمية الدخول للملف الطبي (عبر الويب او الإصدار المكتبي او الاندرويد)..... 94
رسم توضيحي 37-6 خوارزمية إضافة بيانات للملف الطبي (عبر الإصدار المكتبي) 95
رسم توضيحي 38-6 خوارزمية تعديل البيانات للملف الطبي (عبر الإصدار المكتبي)..... 96
رسم توضيحي 39-6 خوارزمية البحث عن الرقم للملف الطبي واستعراض البيانات..... 97
رسم توضيحي 40-7 شاشة اختبار عملية الإضافة والحفظ للمستشفى 99
رسم توضيحي 41-7 شاشة اختبار عملية توليد الحساب بعد عملية الحفظ للمختبرات 100
رسم توضيحي 42-7 شاشة اختبار عملية الإضافة والحفظ للأخبار على الموقع 100
رسم توضيحي 43-7 شاشة اختبار عملية التعديل والحفظ للإعلانات 101
رسم توضيحي 44-7 شاشة اختبار عملية التعديل والحفظ للتاريخ المرضي 101
رسم توضيحي 45-7 شاشة اختبار عملية التعديل والحفظ للزيارة المرضية 102
رسم توضيحي 46-7 شاشة اختبار عملية الإضافة والحفظ للمستخدمين 102
رسم توضيحي 47-7 شاشة اختبار عملية التعديل والتغير لكلمة المرور للمستخدمين 103

فهرس الجداول

جدول 1-2 متطلبات تصميم النظام المادية 23
جدول 2-2 متطلبات تصميم النظام (برمجيات) 23
جدول 3-2 التكلفة الاولية للنظام 26
جدول 4-2 الخطة الزمنية 28
جدول 5-3 يوضح الرموز المستخدمة في برنامج Power designer 30
جدول 6-3 جدول الاطباء (Doctors) 46
جدول 7-3 جدول المرضى (Patients) 47
جدول 8-3 جدول المستشفيات (Hospitals) 47
جدول 9-3 جدول المختبرات (Labs) 48
جدول 10-3 جدول الادوية (Medicines) 48
جدول 11-3 جدول المدراء (Managers) 48
جدول 12-3 جدول الصلاحيات (Powers) 49
جدول 13-3 جدول الأقسام (Sections) 49
جدول 14-3 جدول المرضى_الادوية (Medicines_Patients) 49
جدول 15-3 جدول المرضى_المختبرات (Patients_ Labs) 50
جدول 16-3 جدول المرضى _الاطباء (Patients_ Doctors) 50
جدول 17-3 جدول ترميز المرضى 51
جدول 18-3 يوضح الرموز المستخدمة في الخوارزميات 52
جدول 19-5 تصميم جدول المستشفيات في قاعدة البيانات 64
جدول 20-5 تصميم جدول للأطباء في قاعدة البيانات 65
جدول 21-5 تصميم جدول للمختبرات في قاعدة البيانات 65
جدول 22-5 تصميم جدول المرضى في قاعدة البيانات 66
جدول 23-5 تصميم جدول المستخدمين في قاعدة البيانات 66
جدول 24 -5 تصميم جدول الأدوية في قاعدة البيانات 67
جدول 25-5 تصميم جدول التحاليل في قاعدة البيانات 67

الفصل الأول:

المقدمـــة

(Introduction)

1.1 المقدمة:

بما اننا نعيش في القرن العشرين بما يسمى عصر المعلومات، ويرجع الفضل الاول والاخير لانتشار المعلومات بمنتهى السهولة الى الحاسوب، حيث اصبح من أهمّ الاختراعات المعاصرة، والتي يحتاجها الإنسان حتّى يتمكن من توفير الكثير من الجهد، والطاقة التي يبذلها في القيام ببعض المهام، ويعد عنصراً اساسياً في كل مؤسسة وجامعة ومنزل، مما جعل هناك ضرورة مُلحة للاعتماد علية لتسهيل كافة العمليات والنشاطات، وقد سهلت التقنية العديد من المهام مثل دخولها في مجال الاتصالات، والعديد من الصناعات، وكانت تكنولوجيا المعلومات سبب في تحسين القدرة على الابتكار والتطوير على كافة الأصعدة، تتفاوت بين مجموعة واسعة من المجالات ، منها مجال الطب فمنذ اختراع الحاسب الآلي وهو يلعب دورا فعالا في المستشفيات فالصحة والطب لا حياة الا بهما حيث توظف أنظمة الكمبيوتر للحفاظ على سجلات المرضى و غالبا ما يكون ضروريا للحفاظ على سجلات مفصلة من التاريخ الطبي للمريض فالأطباء كثيرا ما تتطلب من المعلومات حول تاريخ عائلة المريض والامراض الجسدية والادوية الموصوفة من قبل ويمكن تخزين هذه المعلومات على نحو فعال للكميات الضخمة من البيانات الطبية في قاعدة بيانات الكمبيوتر فالعديد من المعدات الطبية الحديثة والصغيرة مبنية علي الحواسيب المبرمجة في ادائها بناء على توجيهات مبرمجة مسبقا كما ساهم في جعل أجهزة الأشعة والمختبرات والعناية والفحص وغيرها من الأمور المألوفة أكثر كفاءة وقدرة أكبر على قراءة البيانات وتحليلها في تلك الأجهزة الأمر الذي يعني وصولًا أسرع إلى التشخيص ومن ثم العلاج، وانطلاقاً من زمن رقمي تلعب فيه الآلة دورا قويا في تخفيف العبء المُلقى على البشر، ليكون شريكا للطبيب في تحديد العلاج و تشخيص المرض وفي إجراء العمليات الجراحية أيضا والان باتت الحواسيب تجري بعض العمليات الطبية المعقدة التي تتطلب أدوات دقيقة، بالإضافة إلى كل ما سبق الدقة الفريدة التي قد تضاهي براعة الجراحين أنفسهم مما يزيد من احتمالية نجاة المريض أو نجاح العملية، ومن المرجح بأن التطور في هذا المجال قد يجعل هذه التقنية تحل مكان الجراحين بشكل كامل في المستقبل ليكون الكمبيوتر ذو قيمة عظيمة في جميع مجالات الحياة.

1.2 وصف آلية سير النظام القائم:

من خلال النزول الميداني وجدنا ان هناك نوعين من العمل في المستشفيات:

أولا عمل شبة آلي:

وهي ان هناك بعض المستشفيات والمراكز الصحية تعمل بطريقة يدوية عدا العمليات الحسابية فان هناك نظام حسابي بسيط يقوم بتنفيذ هذه العمليات الحسابية فقط دون حفظ أي بيانات او ربط بيانات المريض ويكون عمل مستقل خاص بالحسابات فقط.

ثانيا وجود نظام المستشفيات:

وجدنا ان هناك العديد من المستشفيات تمتلك نظام مستشفى متكامل ومن خلال النزول وجدنا ان معظم هذه المستشفيات تعطي الرقم الطبي للمرضى الذين يكتب لهم عمليات جراحية في المستشفى وذلك لمتابعة حالتهم المرضية ومتابعة حسابات المريض وتوليد تقارير شاملة عن المريض، ماعدا المرضى الذين لا يجرون عمليات فإن هناك تسجيل بالنظام ولكن بدون إعطاء المريض رقم صحي وانما تسجل بياناتهم بالنظام بشكل مؤقت (خاص بسندات الفحوصات والمختبرات فقط).

وكل هذا المستشفيات والمراكز الصحية تعمل بشكل مستقل ولا تشارك أي بيانات عن أي مريض الى أي جهات او أنظمة أخرى مما يكلف المريض إعادة كل الخطوات في كل جهة صحية.

1.3 مشكلة النظام القائم:

- ان لكل مستشفى او مركز صحي قاعدة بيانات خاصة لا تشارك بياناتها مع أنظمة أخرى وقد لا تعتمد على الوثائق الخارجية مما يكلف المريض إعادة ما تم فعلة مسبقاً.
- ضياع وتلف الملفات الطبية الورقية في أكثر من مستشفى ومركز صحي.
- غموض لدى الطبيب في التشخيص ومعرفة الادوية المستخدمة مسبقا وصعوبة اتخاذ القرار المناسب.
- صعوبة حصول المريض على بياناته السابقة المتفرقة (التقارير) في أكثر من مستشفى او مركز صحي.
- عدم وجود (قاعدة بيانات مركزية) يجمع السجلات الطبية الخاصة بالمريض المتواجدة في أكثر من مستشفى.

1.4 أهداف النظام

- توفير (قاعدة بيانات مركزية) لتوحيد الملفات الطبية للمريض والمتواجدة في أكثر من مستشفى ومركز صحي.
- إعطاء المريض رقم طبي موحد وموثوق في جميع المستشفيات والمراكز الصحية.
- يسهل للأطباء والمختبر إضافة التقارير لملف المريض بسهولة.
- توفير المعلومات عن حالة المريض والأدوية المستخدمة ونتائج الفحوصات السابقة وذلك لتحسين اتخاذ القرار المناسب من قبل الطبيب.
- الحصول على الملف الطبي عن طريق الرقم الطبي الموحد بشكل سريع في أي مكان عبر شبكة الانترنت.

1.5 أهمية المشروع

تكمن أهمية المشروع في تطوير الملف الطبي الإلكتروني الموحد لجميع المرضى بحيث يسهل للمريض والطبيب الوصول للمعلومات الخاصة بالمريض لتسهيل تشخيص حالة المريض بسهولة ومعرفة أسباب المرض والادوية المستخدمة من قَبل ونتائج الفحوصات السابقة للمريض وذلك عبر توفير منصة تُسهل عمل المراكز الطبية وتحسين الحالة الصحية لدى المجتمع وتسهيل عمل الأطباء وضمان لعدم أتلاف المعلومات الصحية للمريض تعتبر النظم الصحية القوية أساسا لتحقيق نتائج صحية أفضل كما ان النظم الصحية هي العمود الفقري للمجتمع

ويساعد نظام المعلومات للأطباء في معرفة التاريخ الصحي للمريض والأدوية ونتائج الفحوصات اللازمة وذلك لاتخاذ القرار المناسب، كما تدعم المزيد من الشفافية والموثوقية من خلال زيادة الوصول الى المعلومات.

يبني المشروع القدرة على توليد وإدارة واستخدام المعلومات الصحية بهدف التوصل الى تعزيز المزيد من التعاون والاستفادة من نظام المعلومات الصحية ويشجع على استخدام آليات اتخاذ القرارات المناسبة ومواكبة التكنولوجيا والاستفادة منها مما تقدمه من سهولة وسرعة وتواصل دائم ويقلل التكاليف والاعباء على المرضى.

1.6 تقنيات تجميع البيانات:

تعتبر هذه المرحلة من أهم المراحل حيث تعتمد دقة وجودة النظام على دقة جودة البيانات والتي تم جمعها وتحليلها لتحديد الاحتياجات والمتطلبات الخاصة بالمستخدم، حيث يقوم المحلل في هذه المرحلة بجمع كل البيانات والمعلومات عن النظام القائم التي تساعده على فهم طبيعة النظام وآلية عمله ومشاكلة القائمة، وفيما يلي بعض التقنيات لجمع البيانات:

1.6.1 المقابلات الشخصية:

هي طريقة للحصول على البيانات والمعلومات من خلال الحوار المباشر بين المحلل والمستخدم تعتبر المقابلة الشخصية أكثر الطرق استخداماً لجمع المعلومات.

ولمقابلة ناجحة يجب أن نعتمد التخطيط السليم والإعداد المسبق لها لذلك على محلل النظم الاستعداد للمقابلة وان يأخذ في اعتباره النقاط الهامة أثناء المقابلة وبعد الانتهاء منها.

مميزات المقابلة الشخصية:

تعتبر المقابلة من أهم وسائل جمع البيانات لما تتمتع به من مميزات أهمها:

- الحصول على المعلومات من مصدرها المباشر وهم المستخدمون وبشكل مباشر.
- خلق علاقة طيبة مع المستخدمين.
- التعرف على الحقائق من وجهات نظر مختلفة.
- التعرف على المصطلحات المستخدمة في الشركة.

قمنا بزيارة الى مستشفى السلام الدولي والتقينا الدكتورة نسيم محمد والتي تعمل في الشؤون الإدارية في مستشفى السلام الدولي.

وبعد التعارف وتبادل التحية، طرحنا عليها عدة أسئلة وتبادلنا الحديث حول العديد من النقاط بشأن السجل الطبي للمريض.

تم سرد المقابلة بالتفصيل في الفصل الثاني.

كما قمنا بزيارة لمستشفى عبد القادر المتوكل والتقينا بالدكتور عادل السراجي المدير الإداري في المستشفى.

وطرحنا عليه عدة أسئلة حول السجل الطبي وتم سرد المقابلة في الفصل الثاني.

1.6.2 الملاحظات أو المشاهدات:

هي المراقبة المباشرة لأجل التعرف على طبيعة العمل بشكل ميداني، وفي بعض الأوقات نستخدم في عملية التحقق من صحة البيانات التي تم جمعها بطرق أخرى، ويجب أن يكون محلل النظم ذو مقدرة على المراقبة الدائمة أثناء العمل.

النقاط التي يجب التركيز عليها من قبل المحلل حتى تكون المراقبة مفيدة وفعالة: –

- التفاعل بين المستخدمين والنظام. ماهية البيانات التي يقوم المستخدم بإدخالها والعمليات التي تتم على البيانات ومخرجات هذه العمليات.
- تعامل المستخدمين مع بعضهم البعض.
- طريقة أداء الاعمال.
- الإجراءات المتبعة في الظروف الحرجة.
- كيف تتم مواجهة المشكلة.
- من يتخذ القرار.
- كيف يتم استغلال الوقت.

حيث ومن خلال المراقبة والملاحظة وجدنا ان المريض يتم إعطائه رقم طبي في المستشفيات الكبيرة والتي تمتك أنظمة خاصة مثل مستشفى جامعة العلوم والتكنولوجيا ومستشفى السعودي الألماني و مستشفى 48 الطبي وغيرها من المستشفيات تعطي رقم طبي (خاص بالمستشفى) لمرضى معينين الذين يجرون عمليات

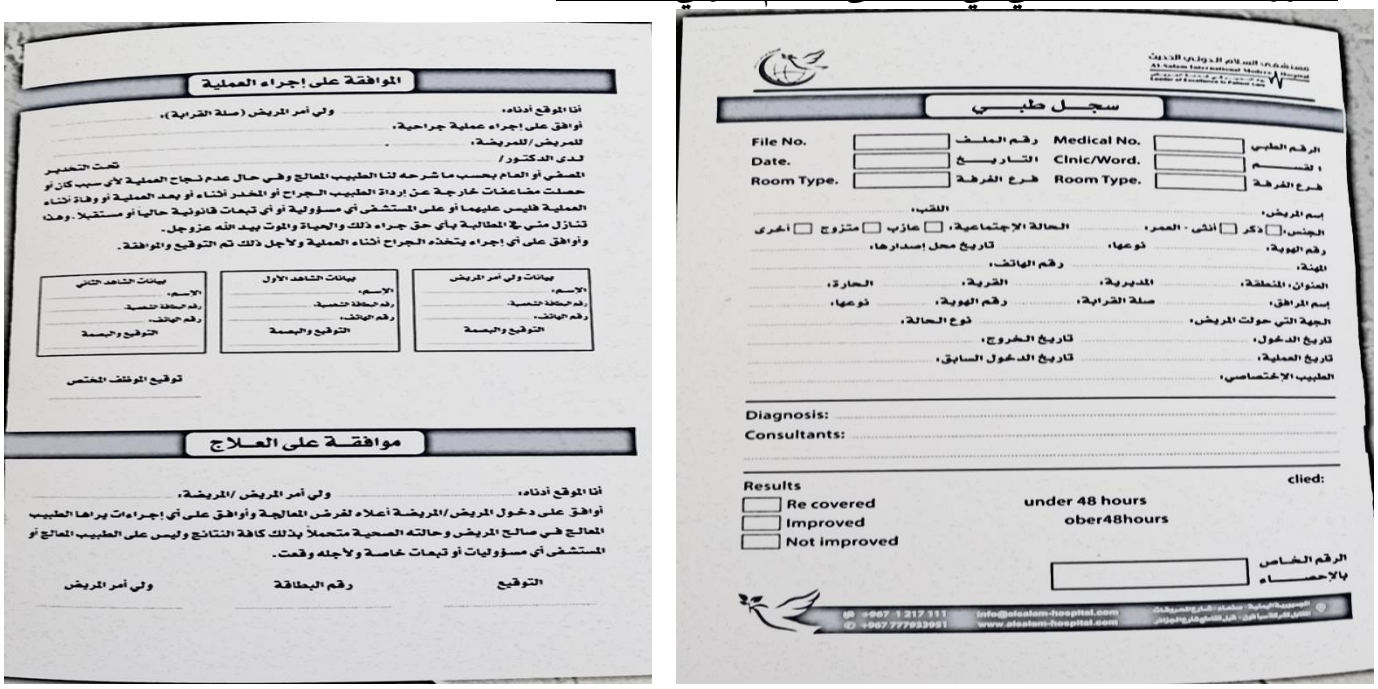
جراحية او في حالة رقود او في عدة حالات يجمع هذا السجل الطبي جميع البيانات عن المريض من تقارير عن حالته الصحية ونتائج الفحوصات والأدوية المستخدمة وبينما هناك مرضى يتم التعامل معهم بطريقة حسابية وعن طريق سندات خاصة بالفحوصات وغيرها وبعض المستشفيات والمراكز تعتمد على العمل اليدوي مع أنظمة محاسبية فقط من اجل تنظيم الأمور المالية فقط.

1.6.3 المستندات والوثائق للنظام الحالي:

هي عملية دراسة وتحليل كافة وثائق المؤسسة مثل تقارير الميزانية وتقارير انشاء الخطط السابقة ومحاضر اجتماعات المدراء وجدولة الاعمال فواتير المبيعات والمشتريات والملفات والاستمارات وغيرها.

نستطيع من خلال هذه الوثائق معرفة الهيكل التنظيمي للمؤسسة والأهداف والخطط وكذلك معرفة دراسة النظام التي تمت سابقاً (في حالة وجود نظام آلي سابق) وتعتبر المستندات والنماذج التي تتعامل معها المؤسسة من الوثائق الهامة التي يحرص محلل النظم على دراستها وتحليلها.

صورة لمستند الملف الطبي في مستشفى السلام الدولي الحديث:



سجل طبي

File No. رقم الملف Medical No. الرقم الطبي
Date. التاريخ Clnic/Word. القسم
Room Type. فرع الغرفة Room Type. فرع الغرفة

إسم المريض: اللقب:
الجنس: ذكر أنثى العمر: الحالة الاجتماعية: عازب متزوج أخرى
رقم الهوية: نوعها: تاريخ محل إصدارها:
المهنة: رقم الهاتف:
العنوان: المنطقة: المديرية: القرية: الحارة:
إسم المرافق: صلة القرابة: رقم الهوية: نوعها:
الجهة التي حولت المريض: نوع الحالة:
تاريخ الدخول: تاريخ الخروج:
تاريخ العملية: تاريخ الدخول السابق:
الطبيب الإختصاصي:

Diagnosis:
Consultants:

Results
Re covered
Improved
Not improved
under 48 hours
ober48hours
clied:

الرقم الخاص بالإحصاء

الموافقة على إجراء العملية

أنا الموقع أدناه، ولي أمر المريض (صلة القرابة)،
أوافق على إجراء عملية جراحية،
للمريض/للمريضة،
لدى الدكتور/
تحت التخدير المحلي أو العام بحسب ما شرحه لنا الطبيب المعالج وفي حال عدم نجاح العملية لأي سبب كان أو حصلت مضاعفات خارجة عن إرادة الطبيب الجراح أو المخدر أثناء أو بعد العملية أو وفاة أثناء العملية فليس عليهما أو على المستشفى أي مسؤولية أو أي تبعات قانونية حاليا أو مستقبلا. وهذا تنازل مني في المطالبة بأي حق جراء ذلك والحياة والموت بيد الله عزوجل.
وأوافق على أي إجراء يتخذه الجراح أثناء العملية ولأجل ذلك تم التوقيع والموافقة.

بيانات ولي أمر المريض: الإسم، رقم البطاقة الشخصية، رقم الهاتف، التوقيع والبصمة
بيانات الشاهد الأول: الإسم، رقم البطاقة الشخصية، رقم الهاتف، التوقيع والبصمة
بيانات الشاهد الثاني: الإسم، رقم البطاقة الشخصية، رقم الهاتف، التوقيع والبصمة

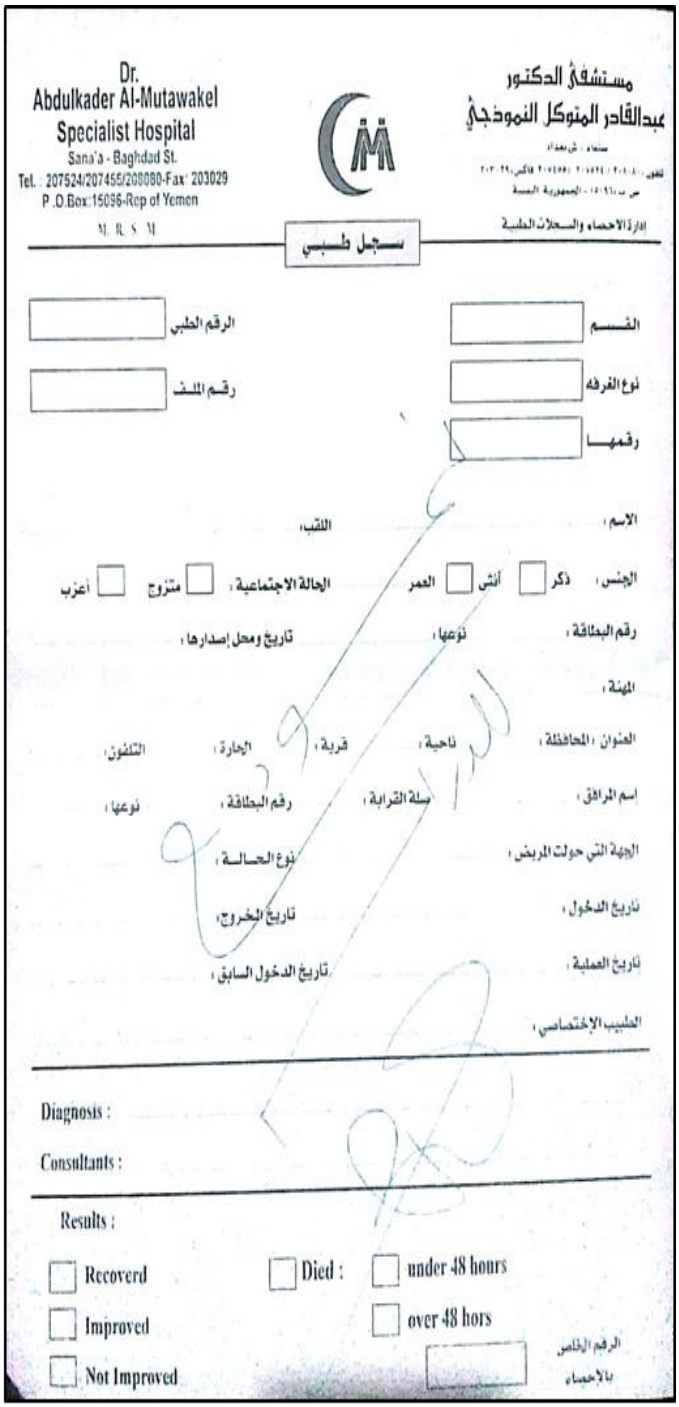
توقيع الموظف المختص

موافقة على العلاج

أنا الموقع أدناه، ولي أمر المريض /المريضة،
أوافق على دخول المريض/المريضة أعلاه لغرض المعالجة وأوافق على أي إجراءات يراها الطبيب المعالج في صالح المريض وحالته الصحية متحملاً بذلك كافة النتائج وليس على الطبيب المعالج أو المستشفى أي مسؤوليات أو تبعات خاصة ولأجله وقعت.

التوقيع رقم البطاقة ولي أمر المريض

صورة لمستند الملف الطبي في مستشفى عبد القادر المتوكل:



مستشفى الدكتور
عبدالقادر المتوكل النموذجي
صنعاء - شارع بغداد
تلفون : ٢٠٧٥٢٤ / ٢٠٧٤٥٥ / ٢٠٨٠٨٠ فاكس : ٢٠٣٠٢٩
ص.ب : ١٥٠٩٦ - الجمهورية اليمنية
إدارة الاحصاء والسجلات الطبية

Dr.
Abdulkader Al-Mutawakel
Specialist Hospital
Sana'a - Baghdad St.
Tel. : 207524/207455/208080-Fax: 203029
P .O.Box:15096-Rep of Yemen
M. R. S. M

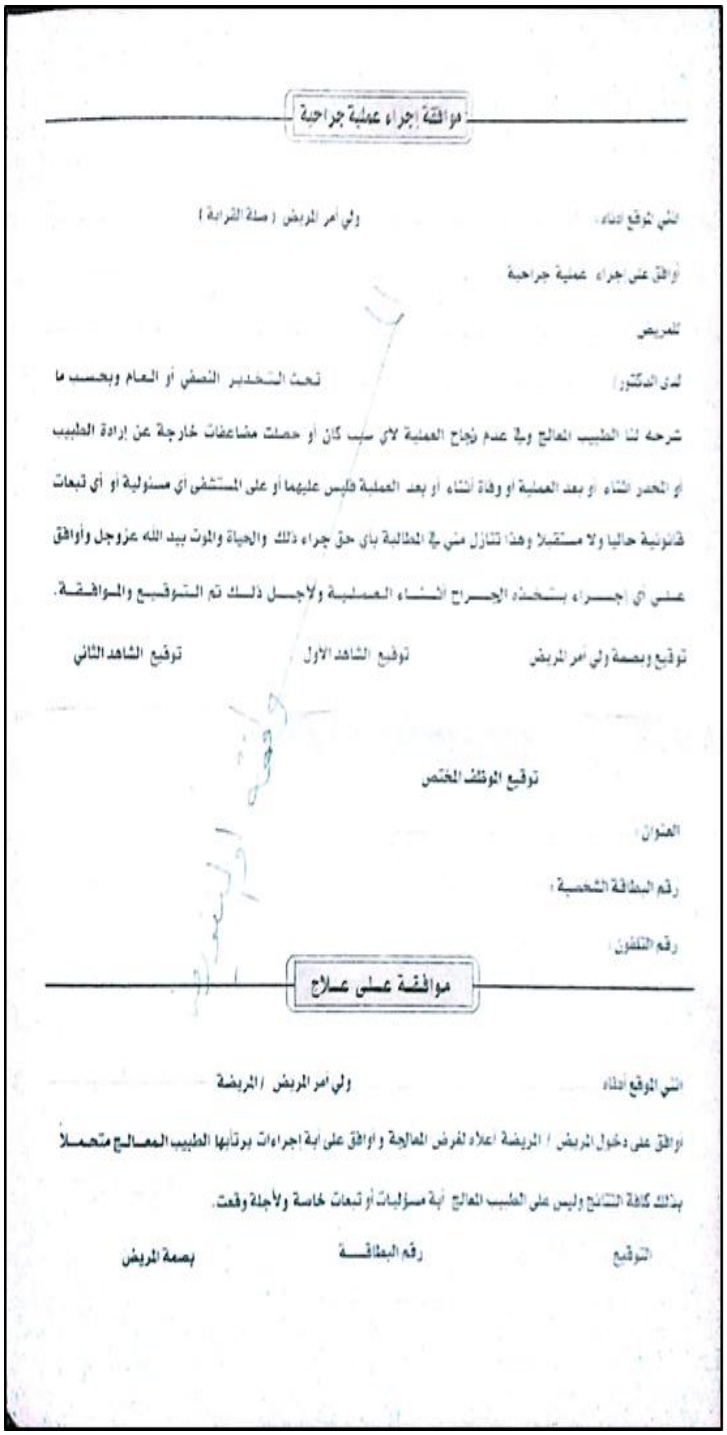
سجل طبي

القسم ☐ الرقم الطبي ☐
نوع الغرفه ☐ رقم الملف ☐
رقمها ☐

الاسم : اللقب :
الجنس : ذكر ☐ أنثى ☐ العمر الحالة الاجتماعية : ☐ متزوج ☐ أعزب
رقم البطاقة : نوعها : تاريخ ومحل إصدارها :
المهنة :
العنوان : المحافظة : ناحية : قرية : الحارة : التلفون :
اسم المرافق : صلة القرابة : رقم البطاقة : نوعها :
الجهة التي حولت المريض : نوع الحالة :
تاريخ الدخول : تاريخ الخروج :
تاريخ العملية : تاريخ الدخول السابق :
الطبيب الإختصاصي :

Diagnosis :
Consultants :

Results :
☐ Recoverd ☐ Died : ☐ under 48 hours
☐ Improved ☐ over 48 hors
☐ Not Improved
الرقم الخاص بالإحصاء ☐



موافقة إجراء عملية جراحية

انني الموقع أدناه : ولي أمر المريض (صلة القرابة)

أوافق على إجراء عملية جراحية

للمريض

لدى الدكتور : تحت التخدير النصفي أو العام وبحسب ما

شرحه لنا الطبيب المعالج وفي عدم نجاح العملية لأي سبب كان أو حصلت مضاعفات خارجة عن إرادة الطبيب

أو المحدر أثناء أو بعد العملية أو وفاة أثناء أو بعد العملية فليس عليهما أو على المستشفى اي مسئولية أو أي تبعات

قانونية حاليا ولا مستقبلا وهذا تنازل مني في المطالبة بأي حق جراء ذلك والحياة والموت بيد الله عزوجل وأوافق

على أي إجراء يتخذه الجراح أثناء العملية ولأجل ذلك تم التوقيع والموافقة.

توقيع وبصمة ولي أمر المريض توقيع الشاهد الأول توقيع الشاهد الثاني

توقيع الموظف المختص

العنوان :

رقم البطاقة الشخصية :

رقم التلفون :

موافقة على علاج

انني الموقع أدناه ولي أمر المريض / المريضة

أوافق على دخول المريض / المريضة أعلاه لغرض المعالجة وأوافق على أية اجراءات يرتأيها الطبيب المعالج متحملاً

بذلك كافة النتائج وليس على الطبيب المعالج أية مسؤليات أو تبعات خاصة ولأجلة وقعت.

التوقيع رقم البطاقة بصمة المريض

صورة لبعض بيانات السجل الطبي في مستشفى عبد القادر المتوكل:

Abdulkader Almutawakel Specialist Hospital
No. 13 Bldg. No. 1 Sana'a - Baghdad St
Tel.: 208080 /207524
Fax: 409718
P.O. Box: 15096

مستشفى الدكتور عبدالقادر المتوكل النموذجي
صنعاء ش بغداد رقم ١٣ عمارة رقم ١
هاتف: ٢٠٧٥٢٤/٢٠٨٠٨٠
فاكس: ٤٠٩٧١٨
ص.ب: ١٥٠٩٦

الأدوية المأخوذة

رقم الطلب: ٦٩٦٥٤ — تاريخ الحركة ٠٧/٠٢/٢٠١٧

اسم المريض: ابراهيم محمد صالح المحفدي

رقم العملية: ١٢٨٩ — اسم العملية: كود عام - عمليات تجميل — الطبيب: ١٠٤٧ د / بشير عثمان

من وقت: ٠١:٠٠ ص — اجمالي اجراءات مستلزمات العمليات ٣٠٠٠

الى وقت: ٠١:٢٠ ص — اجمالي مبلغ اجراءات التخدير: ٦٠٠٠

ملاحظات:

العملة: ريال يمني

نوع الدواء	رقم الدواء / وصف الدواء	من وقت / الى وقت	القوة / النوع	الكمية	الوحدة	السعر	الاجمالي	ملاحظات
	٠١-٠٠٣٣ / برولين ٣/٠	٠١:٠٠ ص / ٠١:٢٠ ص		٠,١٧	باكت	٣,٦١١,٧٣	٦١٣,٩٩	
	٠١-٠٠٦٤ / مخدر موضعي	٠١:٠٠ ص / ٠١:٢٠ ص		٠,٠٤	فيالة	٦٤٢,٤٧	٢٥,٧٠	
	٠١-٠٠٦٦ / بربوفول امبول ٢٠مل	٠١:٠٠ ص / ٠١:٢٠ ص		١,٠٠	امبول	٨٥٩,٣٤	٨٥٩,٣٤	
	٠١-٠٢١١ / محلول رنجر	٠١:٠٠ ص / ٠١:٢٠ ص		١,٠٠	قربة	٣٦٠,٠٠	٣٦٠,٠٠	٠١
	٠١-٠١٨١ / فنتالين	٠١:٠٠ ص / ٠١:٢٠ ص		٠,١٠	امبول	٢,٦٠٠,٠٠	٢٦٠,٠٠	
	٠١-٠١٩٦ / كابيولات١٨	٠١:٠٠ ص / ٠١:٢٠ ص		١,٠٠	حبة	٥٩,٠٠	٥٩,٠٠	
	٠١-٠١٩٧ / كابيولات٢٠	٠١:٠٠ ص / ٠١:٢٠ ص		١,٠٠	حبة	٥٧,٧٧	٥٧,٧٧	
	٠١-٠٠٩٢ / لصقه ميكردون١٥*١٠	٠١:٠٠ ص / ٠١:٢٠ ص		٢,٠٠	حبة	٧٤,٠٠	١٤٨,٠٠	
							٢,٣٨٣,٨٠	
						الاجمالي العام:	٢,٣٨٣,٨٠	

Designed by: Technology Stars +967-1-212614
info@tstars.com

صفحة: ١ / ١

بكوثر
شباط ٧، ٢٠١٧ ١١:٤٩ ق.ظ

1.7 منهجية البحث:

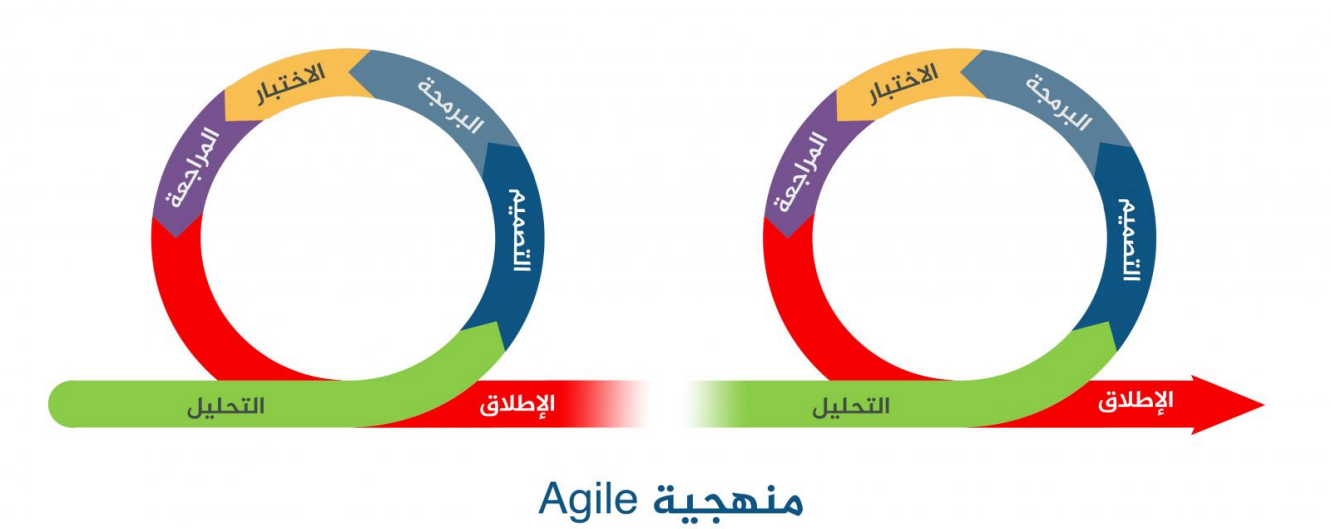
المنهجية المتبعة في هذا النظام هي منهجية (agile processes) أو بما يسمى (بالتطور التزايدي) حيث ان التخطيط يكون فيه تزايدي لذلك التغيير يكون أسهل أي انه يمكن التعديل أو تغيير متطلبات الزبون بسهولة.

- **منهجيه ال Agile:**

هي واحدة من أحدث وأفضل الطرق الناجحة لتطوير البرمجيات وإدارتها خلاقا للطرق التقليدية والكلاسيكية في إدارة المشاريع أو ما يعرف بـ Waterfall ، وتركز منهجيه Agile على خلق وتكوين فريق عمل يتسم بالعملية والانجاز الذكي للمهام، والابتعاد عن الآليات والخطوات البيروقراطية التي تضيع من وقت الفريق دون الوصول الى إنجاز ملموس كالانشغال بتوثيق شفرة المصدر الخاصة بالمنتج، مما يشتت التركيز عن المنتج نفسه، وتولي Agile دائما تركيز أكبر على المنتج والخصائص التي تصل للمستخدم النهائي حيث تعتبر أن المستخدم النهائي هو الحكم والمقياس لنجاح أو فشل منتجها ونلاحظ هذا في نظامنا من خلال:

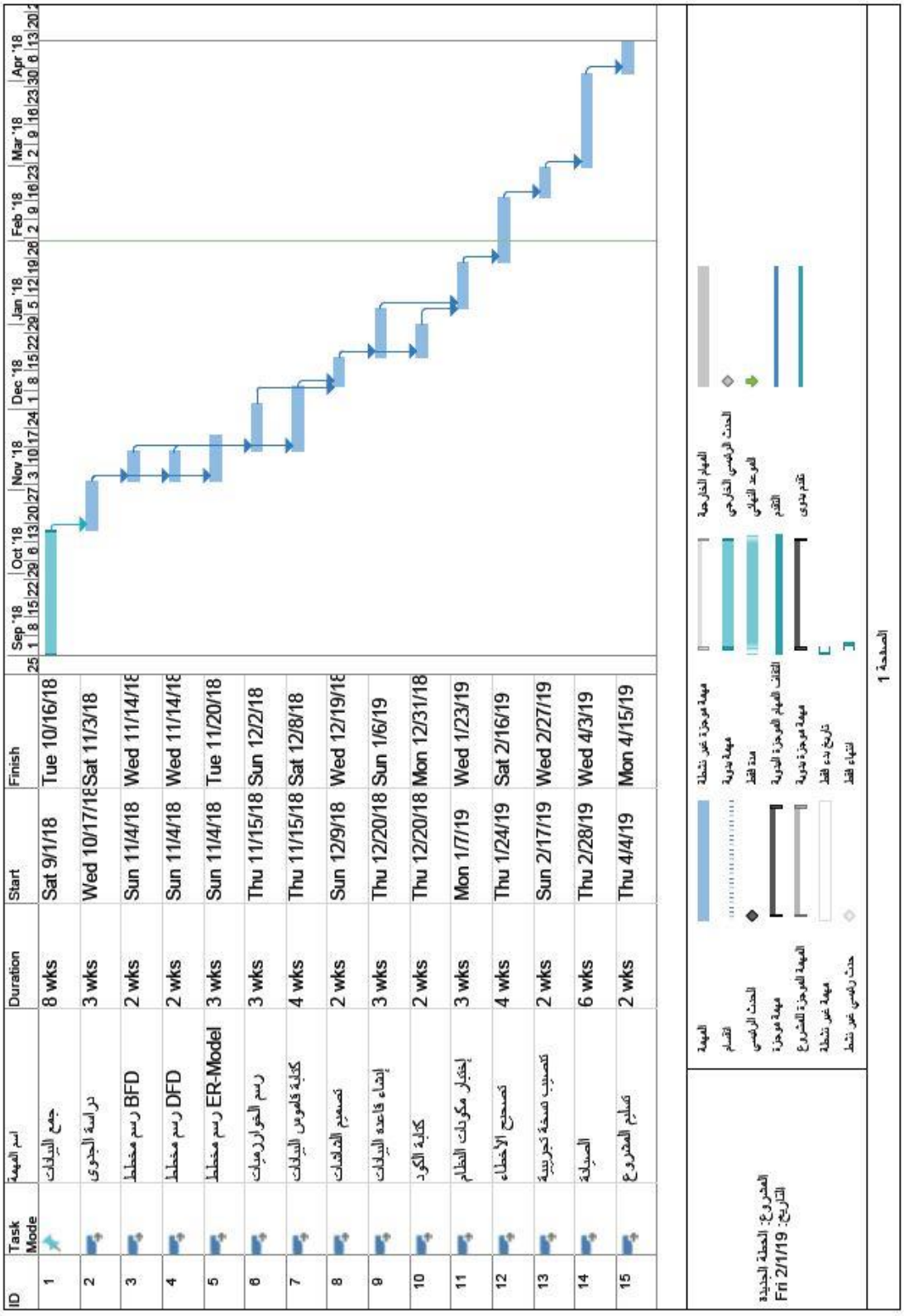
1- القدرة على إضافة خدمات جديدة لم تكن محدده من قبل.

2- سهولة العودة الى مرحلة سابقة.



رسم توضيحي 1-1 دورة تطوير منهجية Agile

1.8 خطة المشروع:



ID	Task Mode	اسم المهمة	Duration	Start	Finish
1		جمع البيانات	8 wks	Sat 9/1/18	Tue 10/16/18
2		دراسة الجدوى	3 wks	Wed 10/17/18	Sat 11/3/18
3		رسم مخطط BFD	2 wks	Sun 11/4/18	Wed 11/14/18
4		رسم مخطط DFD	2 wks	Sun 11/4/18	Wed 11/14/18
5		رسم مخطط ER-Model	3 wks	Sun 11/4/18	Tue 11/20/18
6		رسم الخوارزميات	3 wks	Thu 11/15/18	Sun 12/2/18
7		كتابة قاموس البيانات	4 wks	Thu 11/15/18	Sat 12/8/18
8		تصميم الشاشات	2 wks	Sun 12/9/18	Wed 12/19/18
9		إنشاء قاعدة البيانات	3 wks	Thu 12/20/18	Sun 1/6/19
10		كتابة الكود	2 wks	Thu 12/20/18	Mon 12/31/18
11		إختبار مكونات النظام	3 wks	Mon 1/7/19	Wed 1/23/19
12		تصحيح الأخطاء	4 wks	Thu 1/24/19	Sat 2/16/19
13		تصميم صفحة تجريبية	2 wks	Sun 2/17/19	Wed 2/27/19
14		الصيانة	6 wks	Thu 2/28/19	Wed 4/3/19
15		تسليم المشروع	2 wks	Thu 4/4/19	Mon 4/15/19

رسم توضيحي 2-1 الخطة الزمنية للمشروع

الفصل الثاني

مرحلة التعريف بالنظام

(System Definition Phase)

2.1 المقدمة:

في خضم التطورات التكنولوجيا التي يشهدها العالم في القرن الجديد كان لابد للمنظمات أن تعمل على رسم الخطط والسياسات الهادفة إلى تطوير الأنظمة المعتمدة بكافة أشكالها لتأمين خدمات سريعة ومتطورة تربطها بالمنظمات العالمية عموماً وفي منظمات القطاع المعني بشكل خاص ، ويعد نظام السجل الطبي الالكتروني إحدى المتطلبات المعاصرة التي حسنت من عمل المنظمات الصحية باتجاه انجاز أهدافها وربطها بالعالم الخارجي، ونظراً لما تعانيه مستشفيات الجمهورية اليمنية من مشاكل عدة بسبب محدودية تعاملاتها وتبنيها أسلوب العمل الورقي التقليدي وأنظمة مستقلة ، فقد سعينا إلى بناء نموذج السجل الطبي الالكتروني الموحد يظم جميع المستشفيات والمرافق الصحية يقوم على إدارة معلومات المريض داخل المستشفى وخارجها بشكل الكتروني دون استخدام الأوراق، وتعد الصحة الالكترونية احد أنماط الإدارة الالكترونية المعاصرة ، إذ تقوم الصحة الالكترونية بتوفير الاستشارات والخدمات والمعلومات الطبية إلى المريض عبر وسائل الكترونية وتعد السجلات الطبية الالكترونية واحدة من أفضل الأشكال المحورية التي تعتمد عليها عملية تقديم الرعاية الصحية داخل المستشفيات وبين مختلف أنواع المنظمات الطبية العالمية.

ولا تختلف السجلات الطبية الالكترونية عن السجلات الورقية التقليدية في وظيفتها والهدف منها، لكنها تختلف كليا في طبيعتها وخواصها وامكانيات استخدامها وفوائدها، فهي تمثل نقطة مركزية تصب فيها وتشتق عنها قنوات عديدة من المعلومات المرتبطة بتقديم الرعاية الصحية للمريض من قبل جميع المؤسسات الصحية، كما أنها تمتاز بدقة محتواها وسهولة الوصول إليها من خلال تكاملها مع مصادر المعلومات المختلفة من خلال نظم شبكات المعلومات.

وقد اعتمدنا على بناء هذا النظام وفق أحدث اللغات والتقنيات البرمجية.

إذ سيكون للنظام المقترح بعد تطبيقه مجموعة من المزايا:

1. أتاح النظام المقترح قاعدة بيانات كبيرة تحتوي على المعلومات الضرورية لكل مريض مما سهل العودة إلى تاريخ المريض الطبي من قبل الأطباء في جميع المستشفيات.
2. وفر النظام استخدام أكثر من مستفيد في وقت واحد.
3. خفض النظام المقترح من الكلف الورقية والجهد والوقت.

حافظ النظام المقترح على معلومات المريض الطبية من خلال وجود نظام حماية وامان مع تحديد صلاحيات الدخول.

2.2 التقنيات المستخدمة في جمع البيانات

2.2.1 المقابلة الشخصية:

- التقينا بالدكتورة نسيم والتي تعمل في الشؤون الإدارية في مستشفى السلام الدولي ووجهنا الأسئلة التالية:

س / لمن يعطى السجل الطبي في النظام.

ج/ يعطى للمرضى المقرر لهم عملية جراحية او رقود في المستشفى فقط.

س/ والمرضى الذي ليس لديهم عملية جراحية هل يتم إعطائهم رقم طبي.

ج / لا انما تسجل بياناتهم بشكل مؤقت في النظام من اجل سندات الحسابات فقط.

س/ هل يتم حفظ بيانات المريض بالرقم الطبي في النظام.

ج/ نعم يتم حفظ بيانات كل مريض عن طريق إعطاء المريض رقم طبي يخزن جميع بياناته الصحية والمالية في هذا الحساب.

س/ كيف يتم حفظ بيانات المريض.

ج / يتم حفظها بطريقتين أولا طريقة الكترونية تحفظ البيانات بالنظام والطريقة الثانية يدوية أي أرشفة ملف المريض في المستشفى

س / هل يمكن للمريض استرجاع معلوماته الطبية بعد مرور فترة (3 الى 4) سنوات من العملية الجراحية.

ج / نعم يمكنه الرجوع اليها واسترجاع البيانات اما من النظام او عبر سحب ملفه من الأرشيف ومتابعة حالته.

س/ إذا خرج المريض خارج البلاد وضاعت عنه ملفاته الطبية كيف يستطيع التواصل معكم او هل يستطيع الحصول عليها.

ج/ لا يوجد آلية لهذا ولا نستطيع الا إذا تواصل مع الإدارة بشكل شخصي.

س/ هل يتم رفع بيانات المريض عبر الانترنت.

ج/ لا وانما يتم حفظها بالنظام فقط ولا يتم مشاركتها بشبكة الانترنت.

- كما قمنا بالنزول الميداني لمستشفى الدكتور عبدالقادر المتوكل والتقينا بالدكتور عادل السراجي المدير الإداري للمستشفى وقمنا بطرح عدة أسئلة حول السجل الطبي كما يلي:

س/ ماهي المعلومات والوثائق التي تطلب من المريض عند دخوله المستشفى؟

جـ / يتم استقبال المريض عند دخوله للمستشفى في قسم الاستقبال والحجز والمواعيد إذا كانت معاينة لا يتم منحه رقم طبي او يحول الى الصندوق او إلى العيادات او يتم تحويله قسم الطوارئ ان استدعت الحالة، و إذا كانت الحالة

تستدعي الرقود في المستشفى او عملية جراحية او إذا كان يملك تقارير اوليه او افادة من جهة موثوقة مثل المستشارين او كبار الأطباء يمنح رقم طبي.

س/ ماهي المشاكل والصعوبات التي تواجه المستشفى في الملف الطبي الاولي؟

- الادخال اليدوي.
- عدم اكمال البيانات وعد اهتمام من قبل مدخل البيانات لجهلة بالفائدة او ما يجنيه من المخرجات فيما بعد.
- القصور في البيانات من قبل المريض.
- عدم قدرة الموظفين على استخدام الحاسوب.

س/ هل يتم البحث عن رقم الملف الطبي للمريض قبل تسجيله؟

جـ / نعم عند تسجيل الاسم يظهر هل الاسم موجود من قبل او الرقم الطبي ان وجد.

س/ هل يتم حفظ الملفات الطبي والإضافات الطبية؟ ومتي يتم؟

جـ / نعم يتم حفظها من حين تولد الملف الطبي ويتم التوسع في قاعدة البيانات.

س/ كل كم يتم اجراء نسخ احتياطي للبيانات BACKUP

جـ / يومياً.

س/ هل الرقم الطبي مهم؟ ولماذا؟

جـ / نعم لأنه يساعد بشكل كبير في التشخيص ولان اتخاذ القرار الصائب مهم واي خطاء قد يكلف ثمن باهظاً ان لم تكن روحاً!

س/ كيف يتم التنسيق بين الطبيب واقسام المستشفى مثل المختبرات والاشعة؟

جـ / الطبيب قد يطلب فحوصات معينة للمريض ثم يقوم المريض بسداد التكاليف ويتبع الإجراءات.

س/ هل يتم التواصل بين الطبيب واقسام المستشفى مثل المختبرات او الصيدلة؟

جـ / لا يتم التواصل بشكل مباشرةَ وانما عن طريق بعض الإجراءات في النظام.

س/ هل يطلع المريض على الملف الطبي؟ وكيف يتم تصدير الملف الطبي للمريض للخارج إذا احتاج الملف؟

جـ / نعم بصلاحيات محددة أي لا يقدر التعديل او التحديث على ملفه ويتم تصدير ملفة بشرط انه يأتي للمستشفى ويتم إعطاءه تقارير طبية للحالة ان استدعى ذلك ولا يقدر المريض الوصول عن بعد او استلام تقاريره ويقصد هنا مثل مختبرات العولقي التي اتاحت للمرضى الوصول لنتائجهم عن بعد عبر شبكة الانترنت.

س/ هل يتم تحديث الملف الطبي للمريض وكيف يقوم الطبيب بمتابعة حالة المريض؟

جـ / نعم يتم تحديث البيانات التي اخذت في المستشفى ويتم متابعة حالة المريض طول مكوثه في المستشفى.

س/ ماهي محتويات الملف الطبي؟

1- نموذج الدخول والخروج (يقوم المريض بإدخالها):

- بيانات تعريفية عن المريض مثل – الاسم، العمر، الجنس.
- بيانات إدارية مثل – تاريخ الدخول، الطبيب المعالج، تاريخ العملية.
- بيانات طبية مثل – التشخيص، الاشعة.

2- الشكوى المرضية (يقوم الطبيب بإدخالها):

- فحص سريري.
- فحوصات مخبرية او كشافات.
- تقارير العملية او إحالة للرقود.

س/ الى أي مدى يجدر الاهتمام بالملف الطبي؟

جـ / الي حد كبير لأنه يتم التعامل مع المريض بالرقم وسجله الطبي اما لحفظ سرية المعلومات الخاصة او بياناته الشخصية والإدارية والحسابات الخاصة به.

س/ ما مدى صعوبة ربط أنظمة المستشفيات في ملف طبي موحد كقاعدة بيانات مشتركة؟

جـ / هناك عدة صعوبات مثل:

- خصوصية المستشفى للبيانات الإدارية.
- اختلاف الأنظمة القائمة في المستشفيات النظام الأردني والنظام الهندي والنظام اليمني وغيرها.

2.3 وصف النظام المقترح

2.3.1 تعريف النظام

هو نظام يربط السجل الطبي في جميع المستشفيات الخاصة والعامة والعيادات في موقع ويب عبر الانترنت، حيث يعطى لكل مريض كود او رقم طبي خاص به ويتبع ذلك اسم مستخدم وكلمة مرور وذلك لحماية المعلومات الخاصة بالمريض يستطيع تغيير اسم المستخدم وكلمة المرور عبر الموقع.

وكذلك يعطى رقم خاص لكلاً من المستشفيات او العيادات او المختبرات او المركز الطبي يتبع هذا المركز عدة مستخدمين وهم الأطباء ويكون لهم واجهات وصلاحيات خاصة بتخصصهم الطبي تمكنهم من إضافة المعلومات اللازمة للملف الطبي الإلكتروني للمريض.

وتمكن المريض من الدخول عبر الموقع بالرقم الطبي واسم المستخدم وكلمة المرور الى ملفة الطبي وعرضه لأي طبيب في حالة زيارة أي مستشفى في أي مكان وذلك لاتخاذ القرارات الصحيحة للمريض ومعرفة التاريخ الطبي والأدوية المستخدمة من قبل المريض.

2.3.2 مميزات النظام

1- تم تصميم النظام وفق أحدث اللغات البرمجية.

2- استخدام قاعدة بيانات قوية وآمنة وتدعم كمية بيانات هائلة.

3- جعل النظام قابل للتعامل من قبل الأشخاص العاديين والذين لا يملكون أي خبرات تقنية سوى كيفية التعامل مع الكمبيوتر او تصفح الانترنت.

4- توحيد السجل الطبي في جميع المستشفيات والمراكز الصحية.

5- إمكانية الوصول الية من أي مكان عبر أي جهاز يدعم تصفح الانترنت.

6- إضافة عدد لا محدود من المستشفيات والمراكز الصحية والأطباء والمرضى.

7- تسهيل للمريض والأطباء الكثير من الوقت والجهد.

8- توفير احصائيات ومعلومات.

2.3.3 وظائف النظام

1. إعطاء رقم طبي إلكتروني موحد في جميع المستشفيات والمرافق الصحية.
2. إمكانية إضافة مستشفيات ومختبرات وعيادات ومراكز صحية وتوفير معلومات عنها.
3. إضافة أطباء عبر رقم خاص بالطبيب واسم مستخدم وكلمة مرور يسمح له بالتعديل على الملف الطبي للمريض.
4. يتم إعطاء أسم مستخدم وكلمة مرور للملفات المرضى من أجل الحماية والخصوصية.
5. يسمح للمريض بمتابعة ملفه الطبي من أي مستشفى في اليمن او خارج اليمن عن طريق رقمه الطبي بتوفر الانترنت.
6. تسمح للطبيب بالدخول الى ملف المريض من أجل اتخاذ القرارات المناسبة لصحة المريض.
7. يكون هذا الرقم الطبي كوثيقة للتاريخ الطبي للمريض.
8. منح صلاحيات للمستشفيات والمرافق الصحية والأطباء من اجل الوصول الى الملف الطبي.

2.4 وثيقة متطلبات النظام

نُعرف في هذه المرحلة الوظائف التي يجب ان يقوم بها النظام وخصائصه الأساسية ويتطلب تحديد متطلبات النظام العديد من المناقشات والمشاورات مع الزبائن والمستخدمين وهي عملية تشبه عملية تحليل متطلبات البرمجية.

2.4.1 أنواع متطلبات النظام:

2.4.1.1 المتطلبات الوظيفية (المجردة):

هنا نقوم بتحديد الوظائف الأساسية التي يجب ان يحققها هذا النظام على مستوى عال من التجريد ومنها:

1- يوجد لدية قاعدة بيانات كبيرة التخزين وارشفة ملفات المرضى والمرافق الصحية وبيانات الأطباء.

2- القدرة على التكامل مع المرضى وبسلاسة وانتظام.

3- تفادي الأخطاء التي تحصل في المراكز الطبية.

4- توحيد السجل الطبي في جميع المرافق الصحية.

5- تخزين نتائج الفحوصات والتقارير الطبية في ملفات المرضى وضمان العثور عليها بسهولة.

6- تسهيل وصول المريض والطبيب للسجل الطبي عبر رقم طبي للمساعدة في التشخيص واتخاذ القرار المناسب من قبل الطبيب المعالج.

7- إمكانية الوصول الى الملف الطبي من داخل اليمن او خارجة عبر شبكة الانترنت.

8- تنظيم بيانات المرضى الموزعة في أكثر من مستشفى في ملف طبي موحد (قاعدة بيانات مركزية).

2.4.1.2 متطلبات تصميم النظام:

- مكونات مادية:

الأجهزة (Hardware)	التوضيح
أجهزة الكمبيوتر	جهاز يستخدم للدخول الى الموقع
الأجهزة الذكية	موبايل، تأبلت، ايباد
جهاز مودم	جهاز لربط الحاسوب بالأنترنت

جدول 1-2 متطلبات تصميم النظام المادية

- مكونات برمجية:

البرمجيات (Software)	التوضيح
Windows 7/8/10 operating system	نظام التشغيل الذي سوف يبنى عليه النظام
Microsoft Office word 2016	محرر نصوص
Microsoft Office project 2016	برنامج لإدارة المشاريع
Power designer & Smart Draw	برامج لرسم مخططات النظام

جدول 2-2 متطلبات تصميم النظام (برمجيات)

2.4.1.3 المتطلبات غير الوظيفية:

- الحجم: قاعدة بيانات كبيرة لأرشفة الملفات.
- الموثوقية: القدرة على ضمان العمل دون وجود أي اخفاق.
- الأمن: قدرة النظام على التصدي للاختراقات.
- سهولة الصيانة: سهولة اكتشاف الأخطاء وإصلاحها لمصمم النظام فقط.
- سهولة الاستخدام: القدرة على استخدام النظام بدون أي صعوبة.
- الجاهزية: جاهزية الوصول للنظام في أي وقت.

2.5 دراسة الجدوى

تعتبر دارسة الجدوى مهمة محدودة تؤدى بواسطة مجموعة من الافراد الذين تم اختيارهم بعناية من داخل المنشأة أو بواسطة مجموعة من الخبرات من خارجها لفحص وتقييم النظام الحالي وتقديم الاستشارات والتوضيحات إلى الساعيين في تنفيذ المشروع ورصد ما سيحتاجه النظام الجديد ليكون بديل للنظام القائم وإدخال نظم مرتبطة بالحاسب الآلي بالمنشأة من خلال مدى الفعالية والكفاءة في ذلك.

تهتم دراسة الجدوى بالفوائد التي تجني من النظام وبتكاليف إنشاؤه وتطويره وتشغيله، إذا يتم فيها تقليل التكلفة المتوقعة من هذا النظام، او العائدات الإضافية، أو الأرباح، إضافة الى أي فوائد أخرى قد تنجم عن استخدام النظام المقترح.

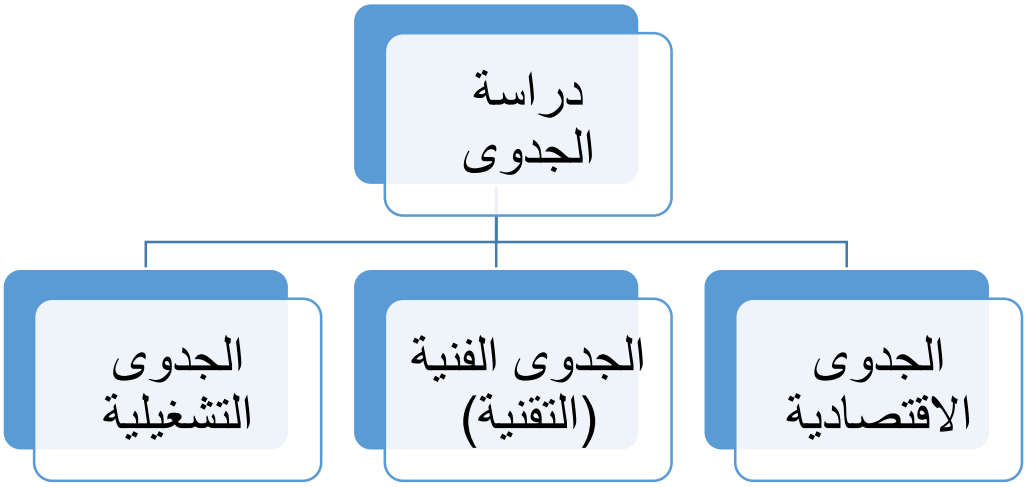
ولتقييم الفرص الاستثمارية من المشروع، فإن المؤسسة تلجا الى استخدام الأساليب التي اعتادت استخدامها، والتي تعتمد على فكرة تحليل العلاقة بين المنافع والتكاليف، علماً بانة من الممكن تحديد التكاليف والمنافع من النوع، بينما لا يمكن تحديد قيمة بعضها الاخر، فإذا كانت التكاليف والمنافع من النوع التي يمكن تحديد قيمتها قبل بانها تكاليف ومنافع ملموسة وإذا كانت من الصعب تحديد قيمتها حالياً الا انها على المدى البعيد تعود الى قيم محددة فإنها تسمى عند إذا تكاليف ومنافع غير ملموسة.

- تعريف دراسة الجدوى:

تعتبر دراسة الجدوى القلب النابض في مرحلة الدراسة الأولية، ودراسة الجدوى لأي مشروع تكون عادة تقرير ما إذا كان النظام القائم صالحاً لإجراء عليه عمليات البرمجية بالحاسب الآلي أم لا، ولقد تمت دراسة الجدوى النظام القائم وقمنا بعملية تقييم النظام وذلك من خلال دراسة الأنواع الأساسية لهذه المرحلة.

❖ **الهدف من دراسة الجدوى:**

1. نقطة أساسية في اتخاذ القرار بصورة عملية.
2. تحديد النظرة العامة للموقع والتكلفة والاهداف.
3. تحديد نجاح او فشل المشروع.



رسم توضيحي 3-2 انواع دراسة الجدوى

2.5.1 دراسة الجدوى الاقتصادية (Economic Feasibility)

خلاصة هذه الدراسة ان تكون الفائدة من النظام الجديد أكثر من تكلفته والهدف من ذلك هو اتخاذ القرار بشأن هذا النظام او اختيار نظام يكون اقل تكلفة وأكثر منفعة أي انها تهتم بعملية مقارنة تكاليف النظام السابق نسبة الى العائد من هذا النظام وتهدف دراسة الجدوى الاقتصادية إلى تحديد الفوائد العامة للمشروع وتقدير التكاليف الأولية والمستمرة للمشروع والمقارنة فيها بينهما كالتالي: –

الفوائد العامة للمشروع:

- أتاح النظام المقترح قاعدة بيانات كبيرة تحتوي على المعلومات الضرورية لكل مريض مما سهل العودة إلى تاريخ المريض الطبي من قبل الأطباء في جميع المستشفيات.
- وفر النظام استخدام أكثر من مستفيد في وقت واحد.
- خفض النظام المقترح من الكلف الورقية والجهد والوقت.
- تسهيل الحصول على المعلومات الطبية شاملة وموحدة وبطريقة سهلة.

التكاليف الأولية للمشروع:

الوصف	التكلفة
نظام السجل الطبي الإلكتروني الموحد	$1000

جدول 3-2 التكلفة الاولية للنظام

2.5.2 دراسة الجدوى الفنية (التقنية)Technical Feasibility Study :

وهي المتعلقة بالتكنولوجيا المستخدمة لإنشاء النظام وتتحقق الجدوى الفنية للنظام من خلال دراسة قابلية النظام للتطبيق والتي تتم من خلال:

- دراسة توفر المكونات المادية اللازمة لعمل النظام.
- دراسة توفر المكونات البرمجية اللازمة لعمل النظام.

دراسة المكونات المادية:

- جهاز حاسب آلي.
- أجهزة ذكية (هاتف ذكي–تأبلت–أي باد).
- جهاز مودم.
- خط انترنت.
- مكتب.

دراسة المكونات البرمجية:

- Windows 7/8/10 operating system – نظام التشغيل الذي سوف يبنى عليه النظام.
- Microsoft Office word 2016 – محرر نصوص.
- Power designer & Smart Draw – برامج لرسم مخططات النظام.
- Microsoft Office project 2016 – برنامج لإدارة المشاريع.
- Oracle – قواعد بيانات.

2.5.3 دراسة الجدوى التشغيلية (Operational Feasibility Study):

يتم فيها دراسة المشروع من حيث قابلية تطبيقه (تشغيله) حيث سيتم استخدام النظام بسهوله لاحتوائه على واجهات جذابة ومفهومة وتدعم التعامل السريع وتسهل على المرضى والأطباء الحصول على المعلومات الكافية والكاملة عن النظام وخدماته وكذلك الحصول على الإرشادات والتعليمات المطلوبة وتسهيل عملية الوصول الى البيانات بكل سهوله وسرعة ومن خلال دراسة قابلية تشغيل النظام نستطيع القول بأن المشروع مجدي تشغيلياً.

2.5.4 المخطط الزمني:

الخطة الزمنية لتحليل وتصميم وتنفيذ النظام بشكل كلي: (2018/10/28م) – (2019/4/04م)

المرحلة	الوقت
توصيف البيانات	2018/10/28م – 2018/11/01م
رسم مخططات العمليات	2018/10/4م – 2018/11/15م
رسم مخططات البيانات	2018/11/18م– 2018/11/29م
رسم الخوارزميات	2018/12/02م – 2018/12/06م
انشاء قاموس البيانات	2018/12/09م – 2018/12/13م
التصميم	2018/12/16م – 2018/12/20م
كتابة الكود	2018/12/23م – 2019/01/10م
أنشاء قاعدة البيانات	2019/01/13م – 2019/01/24م
اختبار مكونات النظام	2019/01/25م – 2019/02/02م
تصحيح الاخطاء	2019/2/03م – 2019/02/16م
تجربة الموقع ضمن الشبكة	2019/2/17م – 2019/02/21م
تدريب	2019/2/24م – 2019/02/28م
رفع الموقع على الانترنت	2019/3/03م– 2019/03/21م
الصيانة	2019/3/24م – 2019/04/04م

جدول 4-2 الخطة الزمنية

الفصل الثالث

مرحلة توصيف العمليات

(Process Modeling)

4.1 نمذجة العمليات (Data Flow Diagram)

مخطط تدفق البيانات: -

مخطط تدفق البيانات هو رسم بياني لتدفق البيانات بين مختلف العمليات في الأعمال التجارية، وهو أيضاً تقنية رسم بياني تصوّر تدفق المعلومات التي تطبق في نقل البيانات من مدخلات إلى المخرجات، وتوفر بطريقة بسيطة وبديهية وصف العمليات التجارية دون تركيز على التفاصيل من نظم الحاسوبية، ويمكن وصف مخطط تدفق البيانات بأنه الرسم الذي يوضح حركة البيانات الكينونات الخارجية والعمليات ومخازن البيانات داخل النظام.

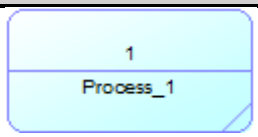
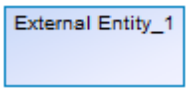
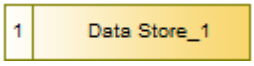
يمكن تمثيل نظم المعلومات المادية والمنطقية بأربعة رموز فقط، وهي:

1- العملية Process

2- تدفق البيانات Data Flow

3- مخزن البيانات Data Store

4- كينونة Entity

شرح الرمز	الرمز
العملية: (Process) هو النشاط الذي يحول المدخلات الى مخرجات (معالجة).	
كينونة: (Entity) أي عنصر خارج النظام، وله دور رئيسي في تزويد النظام بالبيانات او استلام المعلومات منه.	
مخزن البيانات: (Data Store) هي مكان تخزين البيانات لحين الحاجة إليها (ملف، قاعدة بيانات).	
تدفق البيانات: (Data Flow) هي حركة البيانات في النظام.	

جدول 5-3 يوضح الرموز المستخدمة في برنامج Power designer

جمع البيانات

من خلال النزول الميداني والمقابلات الشخصية والحصول على الوثائق ومستندات في عدة مستشفيات في امانة العاصمة ومن اجل عمل النظام وخلصنا الى تقسيم الموقع الى ثلاثة اقسام وهي كالتالي:

1. **قسم التسجيل.**

- **تسجيل مستشفى: -**
  - يقوم ممثل المستشفى بتقديم طلب التسجيل كمستشفى.
  - يقوم الموقع بعرض شاشة تسجيل المستشفى.
  - يقوم ممثل المستشفى بإدخال بيانات المستشفى.
  - يقوم الموقع بحفظ بيانات المستشفى في ملف اشعارات المستشفى.
  - يقوم الموقع بأرسال اشعار للممثل المستشفى بالرد عبر البريد الإلكتروني او الهاتف.
  - يقوم الموقع بالرسال اشعار الى وحدة الإدارة بطلب التسجيل.
  - تقوم وحدة الإدارة بالتأكد من صحة بيانات المستشفى.
  - تقوم وحدة الإدارة بالتواصل مع ممثل المستشفى عبر البريد الإلكتروني او الهاتف ومنحة الموافقة، وإعطاءه حساب في الموقع للدخول الى المواقع وممارسة أنشطة المستشفى على الموقع.
- **تسجيل مختبر: -**
  - يقوم ممثل المختبر بتقديم طلب التسجيل مختبر.
  - يقوم الموقع بعرض شاشة تسجيل المختبر.
  - يقوم ممثل المختبر بإدخال بيانات المختبر.
  - يقوم الموقع بحفظ بيانات المختبر في ملف اشعارات المختبرات.
  - يقوم الموقع بأرسال اشعار للممثل المختبر بالرد عبر البريد الإلكتروني او الهاتف.
  - يقوم الموقع بالرسال اشعار الى وحدة الإدارة بطلب التسجيل.
  - تقوم وحدة الإدارة بالتأكد من صحة بيانات المختبر.

- تقوم وحدة الإدارة بالتواصل مع ممثل المختبر عبر البريد الإلكتروني او الهاتف ومنحة الموافقة، وإعطاءه حساب في الموقع للدخول الى المواقع وممارسة أنشطة المختبر على الموقع.

- **تسجيل طبيب: -**
  - يقوم الطبيب بتقديم طلب التسجيل كطبيب.
  - يقوم الموقع بعرض شاشة تسجيل الاطباء.
  - يقوم الطبيب بإدخال بياناته.
  - يقوم الموقع بحفظ بيانات الطبيب في ملف اشعارات الاطباء.
  - يقوم الموقع بأرسال اشعار للطبيب بالرد عبر البريد الإلكتروني او الهاتف.
  - يقوم الموقع بالرسال اشعار الى وحدة الإدارة بطلب التسجيل.
  - تقوم وحدة الإدارة بالتأكد من صحة بيانات الطبيب.
  - تقوم وحدة الإدارة بالتواصل مع الطبيب عبر البريد الإلكتروني او الهاتف ومنحة الموافقة، وإعطاءه حساب في الموقع للدخول الى المواقع وممارسة أنشطته على الموقع.

2. قسم الدخول.

- **دخول مستشفى: -**
  - يقوم المستشفى بتقديم طلب الدخول للموقع.
  - يقوم الموقع بعرض شاشة الدخول.
  - يقوم ممثل المستشفى بإدخال بيانات الحساب اسم المستخدم وكلمة المرور.
  - يقوم الموقع بالسماح للمستشفى بالدخول للموقع بعد التأكد من صحة البيانات.
  - يقوم موظف المستشفى بطلب تسجيل مريض.
  - يقوم الموقع بعرض شاشة إضافة سجل طبي للمريض.
  - يقوم الموظف بإدخال بيانات المريض.
  - يقوم الموقع بحفظ بيانات المريض ومنحة رقم طبي واشعار الموظف بالإضافة وإعطاء المريض رقم طبي وكلمة مرور.

- **دخول مختبر.**
  - يقوم المختبر بتقديم طلب تسجيل الدخول للموقع.
  - يقوم الموقع بعرض شاشة تسجيل الدخول.
  - يقوم ممثل المختبر بإدخال بيانات الحساب اسم المستخدم وكلمة المرور.
  - يقوم الموقع بالسماح للمختبر بالدخول للموقع بعد التأكد من صحة البيانات.
  - يقوم المختبر بطلب الدخول الى سجل الفحوصات الخاصة بالمريض.
  - يقوم الموقع بطلب رقم السجل الطبي للمريض.
  - يقوم موظف المختبر بإدخال رقم السجل الطبي للمريض.
  - يقوم الموقع بعرض طلبات الفحوصات الخاصة بالمريض.
  - يقوم الموظف بإضافة نتائج الفحوصات في ملف المريض.
- **تسجيل دخول طبيب.**
  - يقوم الطبيب بتقديم طلب الدخول للموقع.
  - يقوم الموقع بعرض شاشة الدخول.
  - يقوم الطبيب بإدخال بيانات الحساب اسم المستخدم وكلمة المرور.
  - يقوم الموقع بالسماح للطبيب بالدخول للموقع بعد التأكد من صحة البيانات.
  - يقوم الطبيب بطلب الدخول الى سجل الطبي الخاصة بالمريض.
  - يقوم الموقع بطلب رقم السجل الطبي للمريض.
  - يقوم الطبيب بإدخال رقم السجل الطبي للمريض.
  - يقوم الموقع بعرض البيانات الطبية الخاصة بالمريض.
  - يقوم الطبيب بإضافة التقارير الطبية والفحوصات المطلوبة والادوية المطلوبة في ملف المريض.

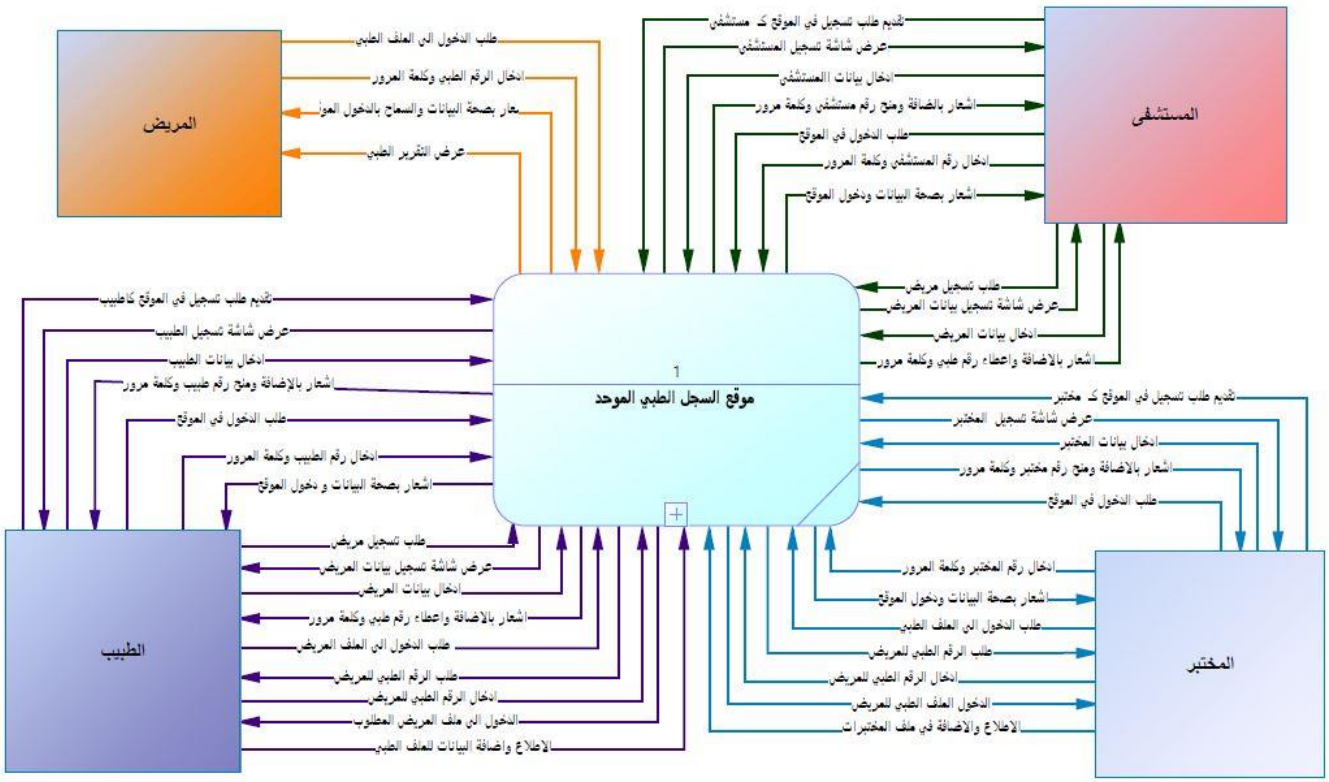
- **دخول مريض.**
  - يقوم المريض بتقديم طلب الدخول للموقع.
  - يقوم الموقع بعرض شاشة تسجيل الدخول.
  - يقوم المريض بإدخال بيانات الحساب رقم السجل الطبي وكلمة المرور.
  - يقوم الموقع بالسماح للمختبر بالدخول للموقع بعد التأكد من صحة البيانات.
  - يقوم المريض بالطلاع على معلومات الملف الطبي الخاص به.

3. قسم الإدارة.

- **إدارة الموقع: –**
  - يقوم المشرف على الموقع بإضافة الاخبار للموقع.
  - يقوم المشرف بالتعديل في محتوى الموقع وإصدار التحديثات الدورية للموقع.
- **الاشراف والرقابة: –**
  - تقوم وحدة التسجيل بإرسال طلبات التسجيل للإدارة.
  - يقوم المشرف بالاطلاع على ملف الاشعارات الخاصة بالمستشفى والمختبر والطبيب.
  - يقوم المشرف بالتأكد من صحة البيانات ومعرفة موقع المستشفى او المختبر ومعرفة المؤهل العلمي للطبيب.
  - يقوم المشرف باعتماد البيانات في حال كانت صحيحة.
  - يقوم المشرف بمنح الصلاحيات اللازمة للحساب المطلوب.
  - يقوم المشرف بإرسال اشعار للمقدم الطلب عبر البريد الإلكتروني او الهاتف.

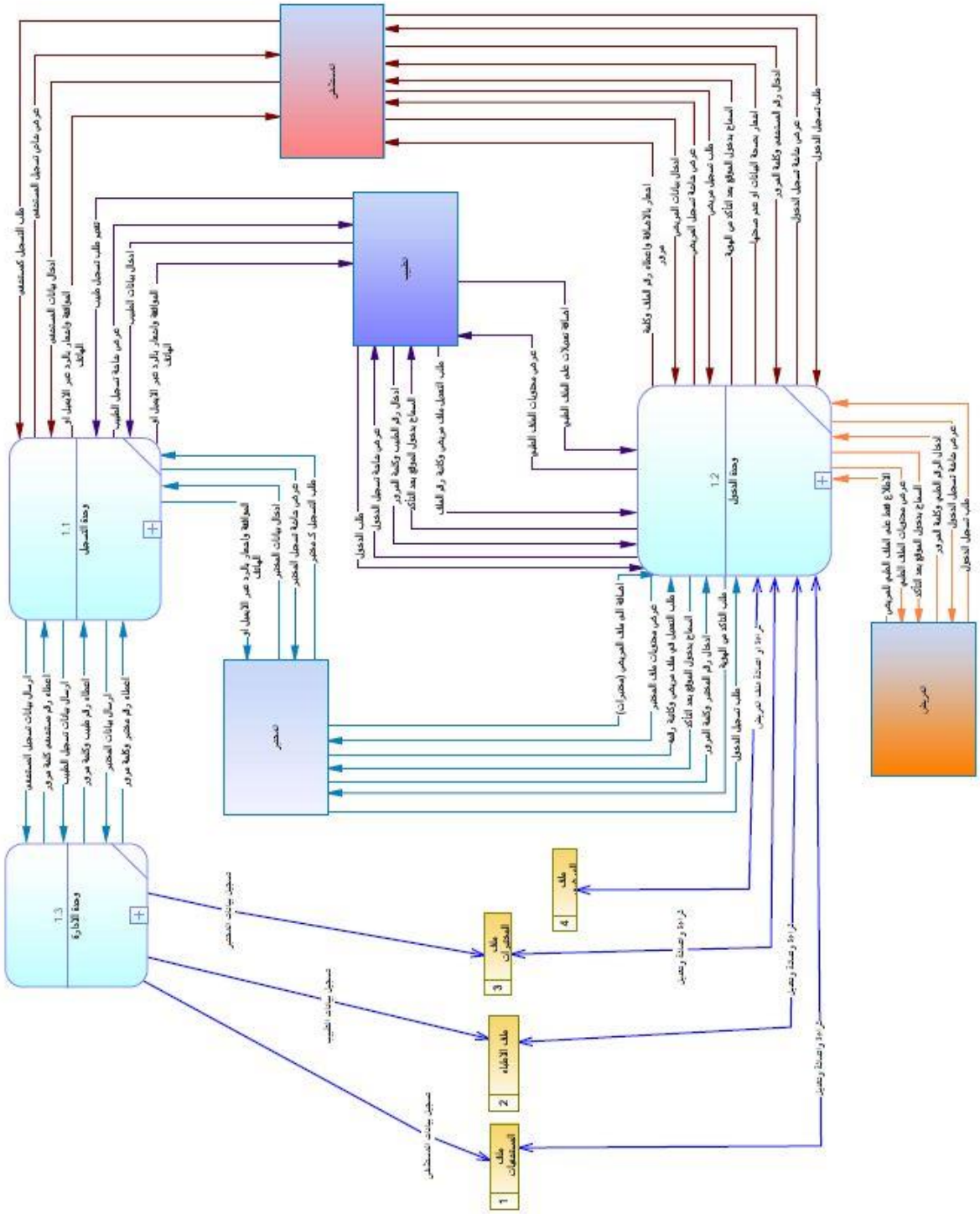
حيث قمنا بدمج التشعبات الكثيرة الخاصة بالملف الطبي للمريض الى هذه الأقسام وتم تبسيط النظام حيث يصبح النواة الأولى للعمل ومن خلال هذه المخططات سيتم توضيح آلية تدفق البيانات الخاصة بالمواقع حيث سيتم ذكرها في المخططات التفصيلية بشكل اكثر وضوح في مخططات المستوى الأول Level 1 .

4.1.1 مخطط السياق (Context Diagram)



رسم توضيحي 4-3 مخطط Context Diagram

4.1.2 المخطط الصفري (Level 0)

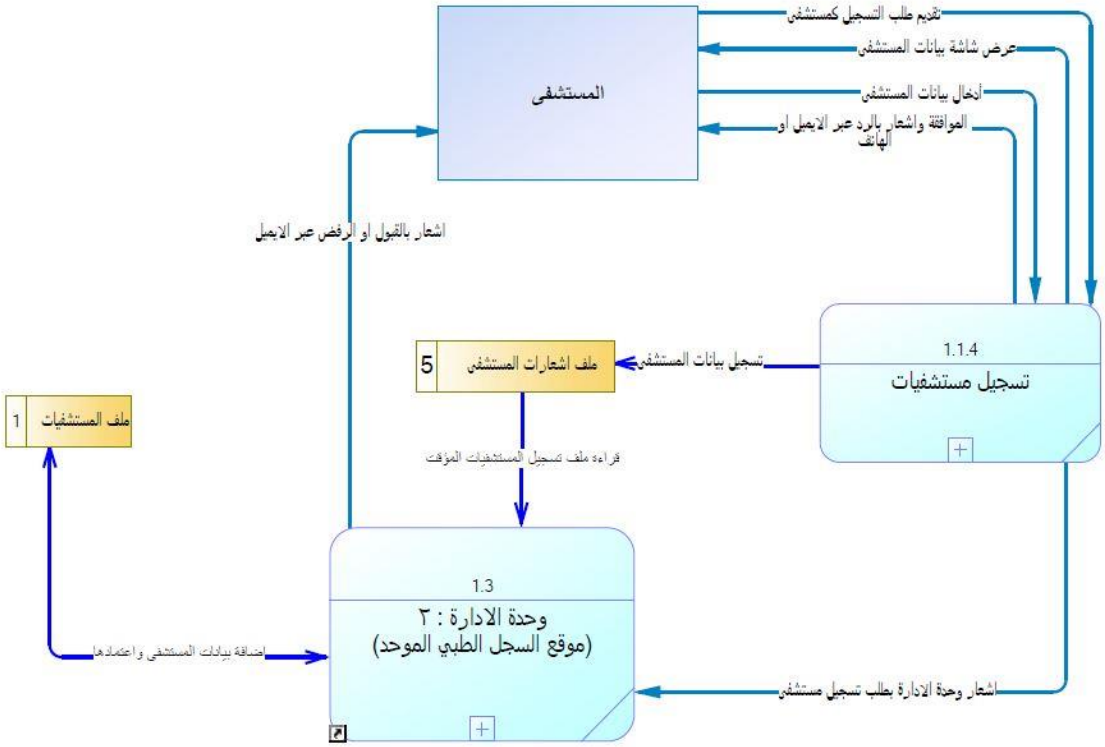


رسم توضيحي 5-3 مخطط Context 0

4.1.3 المخططات التفصيلية (Data Diagram)

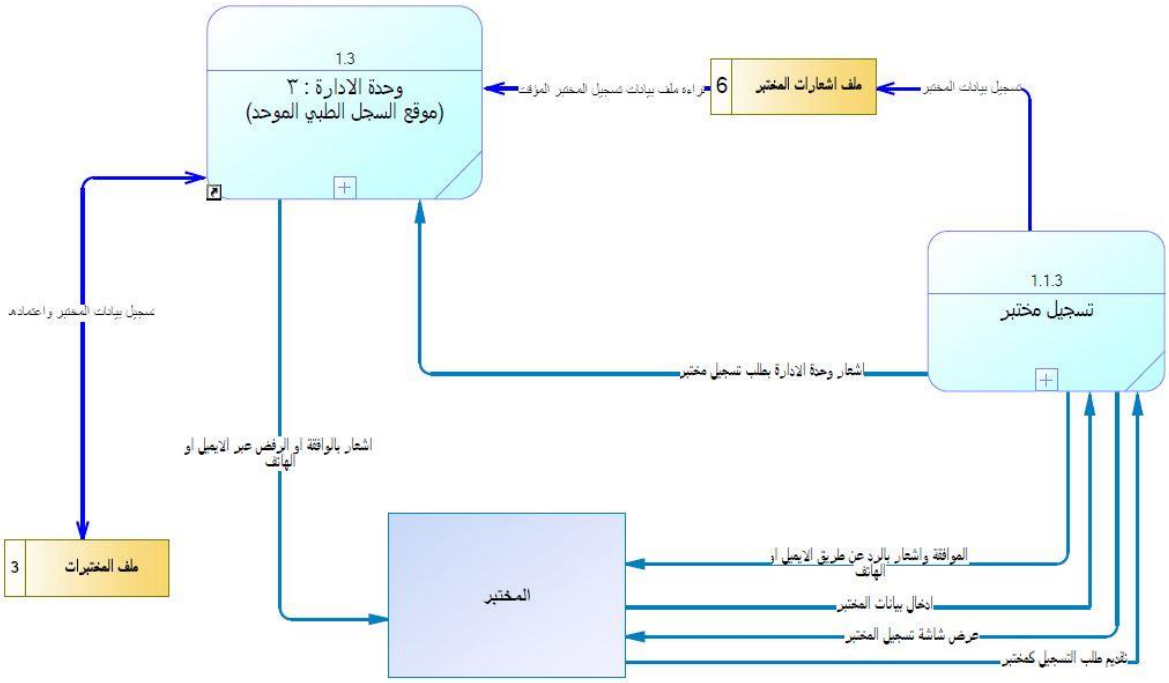
4.1.3.1 المخططات التفصيلية لوحدة التسجيل (Level 1)

- **المخطط التفصيلي لتسجيل مستشفى (Level 1):** –
  - يقوم ممثل المستشفى بتقديم طلب التسجيل كمستشفى.
  - يقوم الموقع بعرض شاشة تسجيل المستشفى.
  - يقوم ممثل المستشفى بإدخال بيانات المستشفى.
  - يقوم الموقع بحفظ بيانات المستشفى في ملف اشعارات المستشفى.
  - يقوم الموقع بأرسال اشعار للممثل المستشفى بالرد عبر البريد الإلكتروني او الهاتف.
  - يقوم الموقع بالرسال اشعار الى وحدة الإدارة بطلب التسجيل.
  - تقوم وحدة الإدارة بالتأكد من صحة بيانات المستشفى.
  - تقوم وحدة الإدارة بالتواصل مع ممثل المستشفى عبر البريد الإلكتروني او الهاتف ومنحة الموافقة، وإعطاءه حساب في الموقع للدخول الى المواقع وممارسة أنشطة المستشفى على الموقع.



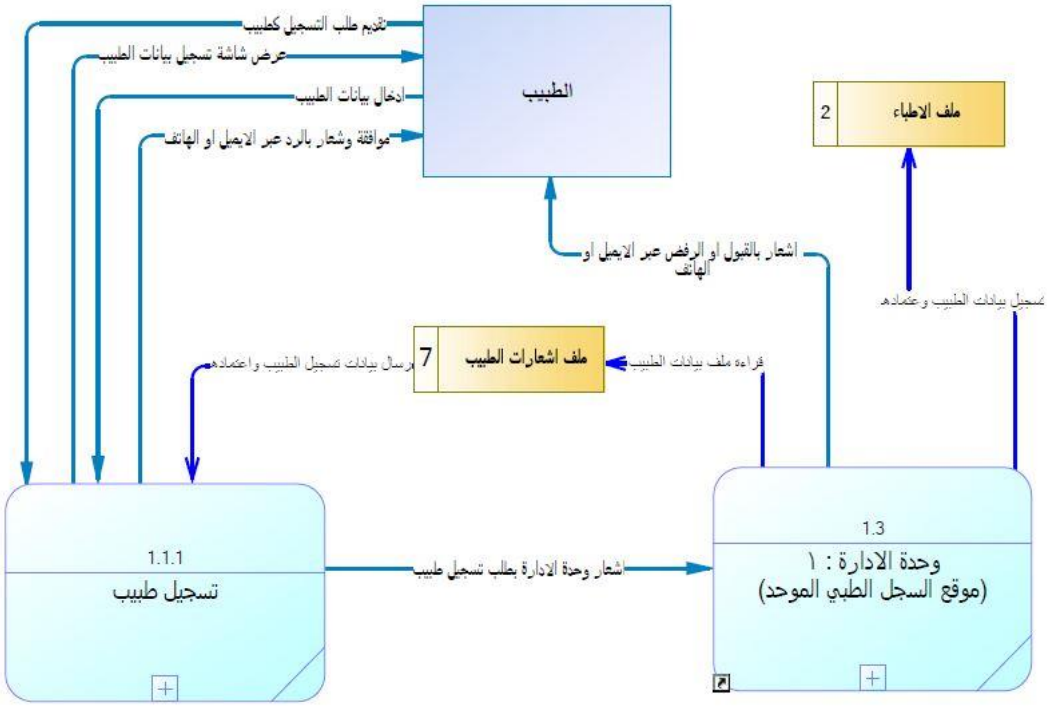
رسم توضيحي 6-3 المخطط التفصيلي لتسجيل مستشفى Level 1

- **المخطط التفصيلي لتسجيل مختبر (Level 1): -**
  - يقوم ممثل المختبر بتقديم طلب التسجيل مختبر.
  - يقوم الموقع بعرض شاشة تسجيل المختبر.
  - يقوم ممثل المختبر بإدخال بيانات المختبر.
  - يقوم الموقع بحفظ بيانات المختبر في ملف اشعارات المختبرات.
  - يقوم الموقع بأرسال اشعار للممثل المختبر بالرد عبر البريد الإلكتروني او الهاتف.
  - يقوم الموقع بالرسال اشعار الى وحدة الإدارة بطلب التسجيل.
  - تقوم وحدة الإدارة بالتأكد من صحة بيانات المختبر.
  - تقوم وحدة الإدارة بالتواصل مع ممثل المختبر عبر البريد الإلكتروني او الهاتف ومنحة الموافقة، وإعطاءه حساب في الموقع للدخول الى المواقع وممارسة أنشطة المختبر على الموقع.



رسم توضيحي 7-3 المخطط التفصيلي لتسجيل مختبر Level 1

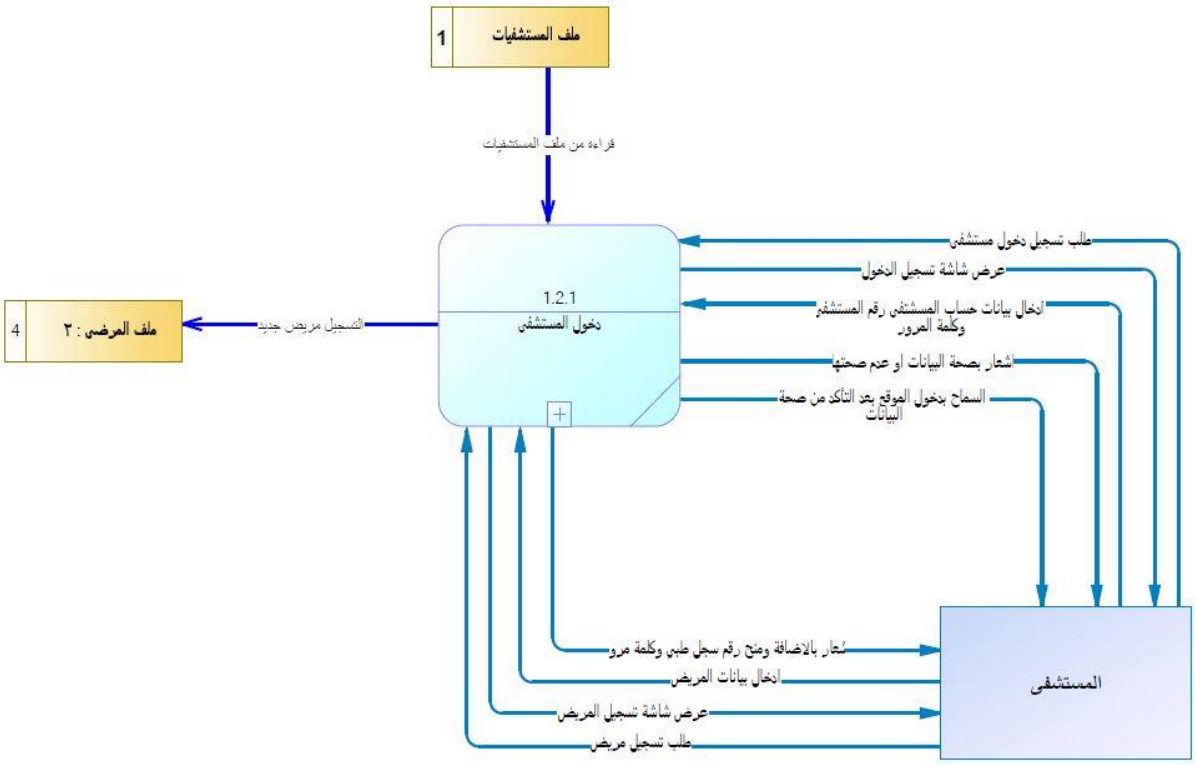
- **المخطط التفصيلي لتسجيل طبيب (Level 1): -**
  - يقوم الطبيب بتقديم طلب التسجيل كطبيب.
  - يقوم الموقع بعرض شاشة تسجيل الاطباء.
  - يقوم الطبيب بإدخال بياناته.
  - يقوم الموقع بحفظ بيانات الطبيب في ملف اشعارات الاطباء.
  - يقوم الموقع بأرسال اشعار للطبيب بالرد عبر البريد الإلكتروني او الهاتف.
  - يقوم الموقع بالرسال اشعار الى وحدة الإدارة بطلب التسجيل.
  - تقوم وحدة الإدارة بالتأكد من صحة بيانات الطبيب.
  - تقوم وحدة الإدارة بالتواصل مع الطبيب عبر البريد الإلكتروني او الهاتف ومنحة الموافقة، وإعطاءه حساب في الموقع للدخول الى المواقع وممارسة أنشطته على الموقع.



رسم توضيحي 3-8 المخطط التفصيلي لتسجيل طبيب Level 1

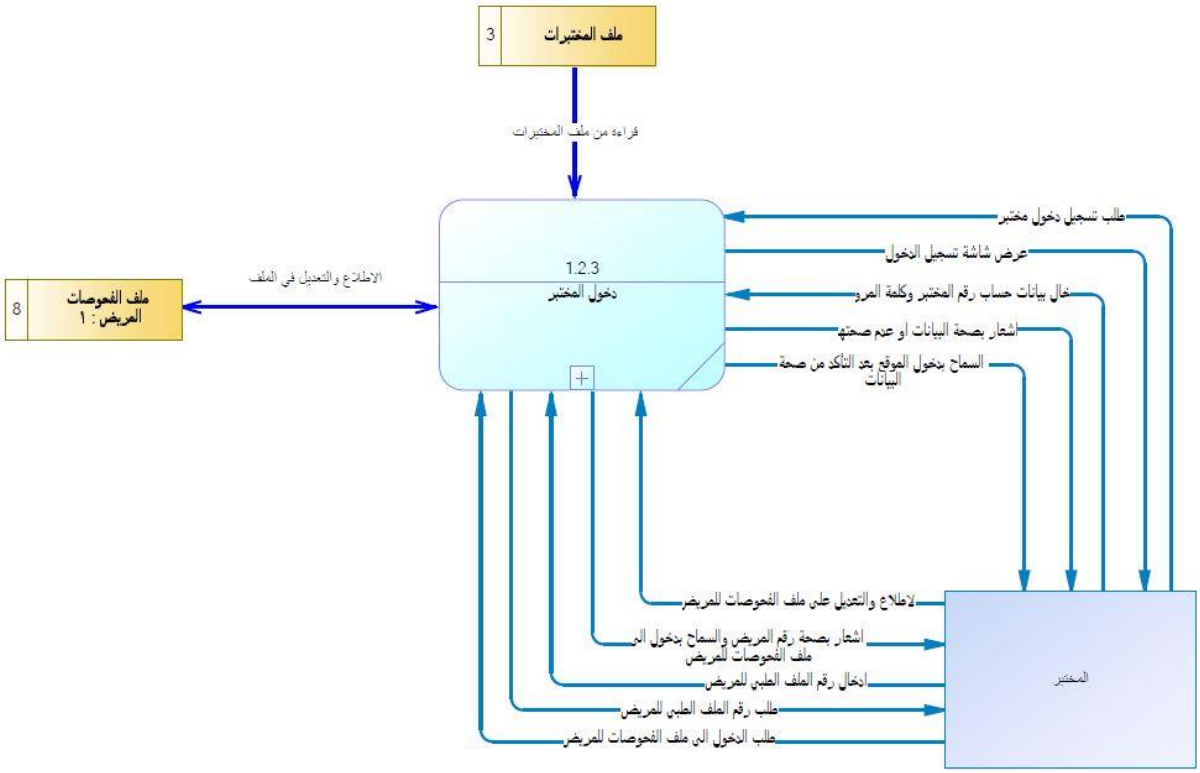
4.1.3.2 المخططات التفصيلية لوحدة الدخول (Level 1)

- **المخطط التفصيلي لدخول مستشفى (Level 1):** –
  - يقوم المستشفى بتقديم طلب الدخول للموقع.
  - يقوم الموقع بعرض شاشة الدخول.
  - يقوم ممثل المستشفى بإدخال بيانات الحساب اسم المستخدم وكلمة المرور.
  - يقوم الموقع بالسماح للمستشفى بالدخول للموقع بعد التأكد من صحة البيانات.
  - يقوم موظف المستشفى بطلب تسجيل مريض.
  - يقوم الموقع بعرض شاشة إضافة سجل طبي للمريض.
  - يقوم الموظف بإدخال بيانات المريض.
  - يقوم الموقع بحفظ بيانات المريض ومنحة رقم طبي واشعار الموظف بالإضافة وإعطاء المريض رقم طبي وكلمة مرور.



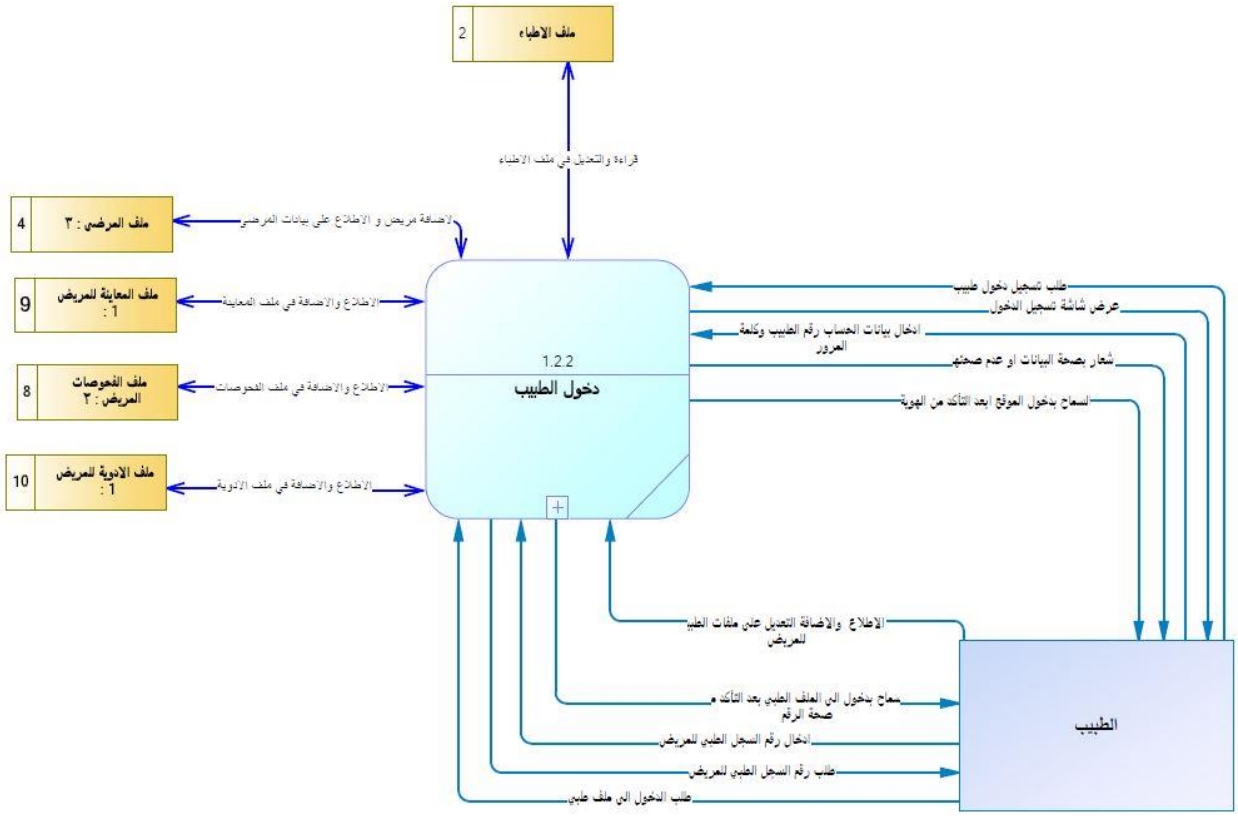
رسم توضيحي 9-3 المخطط التفصيلي لدخول مستشفى Level 1

- **المخطط التفصيلي لدخول مختبر (Level 1):** –
  - يقوم المختبر بتقديم طلب الدخول للموقع.
  - يقوم الموقع بعرض شاشة الدخول.
  - يقوم ممثل المختبر بإدخال بيانات الحساب اسم المستخدم وكلمة المرور.
  - يقوم الموقع بالسماح للمختبر بالدخول للموقع بعد التأكد من صحة البيانات.
  - يقوم المختبر بطلب الدخول الى سجل الفحوصات الخاصة بالمريض.
  - يقوم الموقع بطلب رقم السجل الطبي للمريض.
  - يقوم موظف المختبر بإدخال رقم السجل الطبي للمريض.
  - يقوم الموقع بعرض طلبات الفحوصات الخاصة بالمريض.
  - يقوم الموظف بإضافة نتائج الفحوصات في ملف المريض.



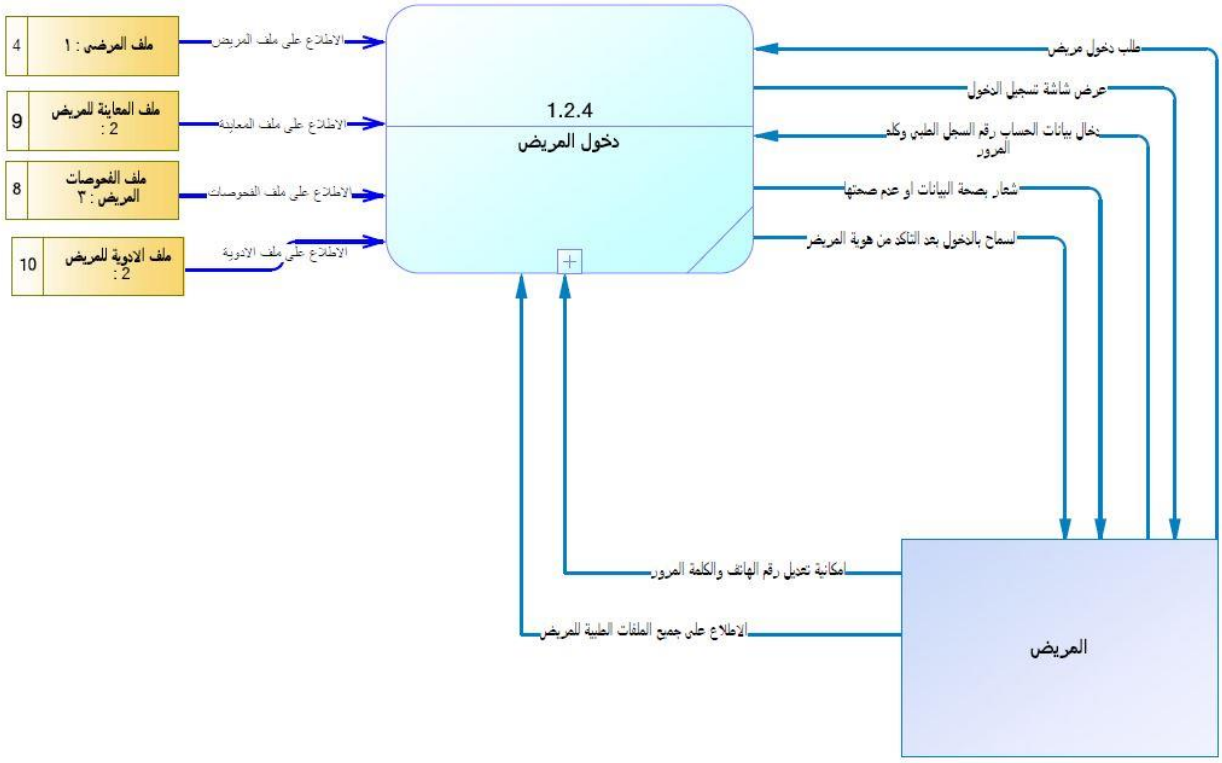
رسم توضيحي 10 -3 المخطط التفصيلي لدخول مختبر Level 1

- **المخطط التفصيلي لدخول طبيب (Level 1):** -
  - يقوم الطبيب بتقديم طلب الدخول للموقع.
  - يقوم الموقع بعرض شاشة الدخول.
  - يقوم الطبيب بإدخال بيانات الحساب اسم المستخدم وكلمة المرور.
  - يقوم الموقع بالسماح للطبيب بالدخول للموقع بعد التأكد من صحة البيانات.
  - يقوم الطبيب بطلب الدخول الى سجل الطبي الخاصة بالمريض.
  - يقوم الموقع بطلب رقم السجل الطبي للمريض.
  - يقوم الطبيب بإدخال رقم السجل الطبي للمريض.
  - يقوم الموقع بعرض البيانات الطبية الخاصة بالمريض.
  - يقوم الطبيب بإضافة التقارير الطبية والفحوصات المطلوبة والادوية المطلوبة في ملف المريض.



رسم توضيحي 11-3 المخطط التفصيلي لدخول طبيب Level 1

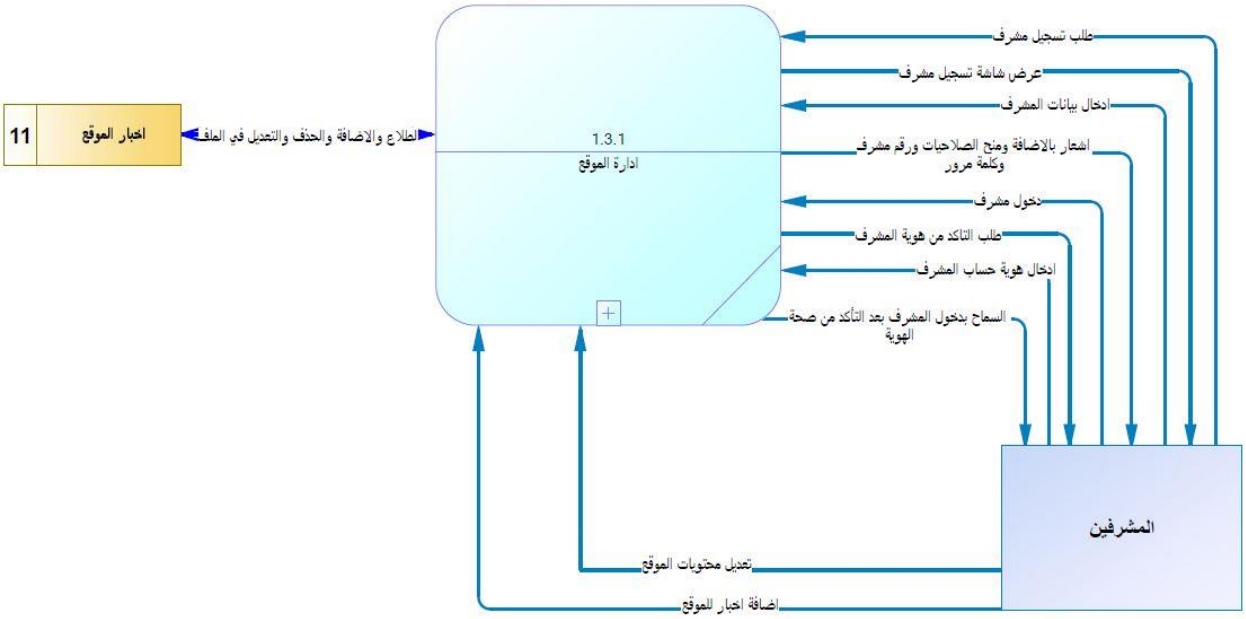
- **المخطط التفصيلي لدخول مريض (Level 1):** -
  - يقوم المريض بتقديم طلب الدخول للموقع.
  - يقوم الموقع بعرض شاشة تسجيل الدخول.
  - يقوم المريض بإدخال بيانات الحساب رقم السجل الطبي وكلمة المرور.
  - يقوم الموقع بالسماح للمختبر بالدخول للموقع بعد التأكد من صحة البيانات.
  - يقوم المريض بالطلاع على معلومات الملف الطبي الخاص به.



رسم توضيحي 12-3 المخطط التفصيلي لدخول مريض Level 1

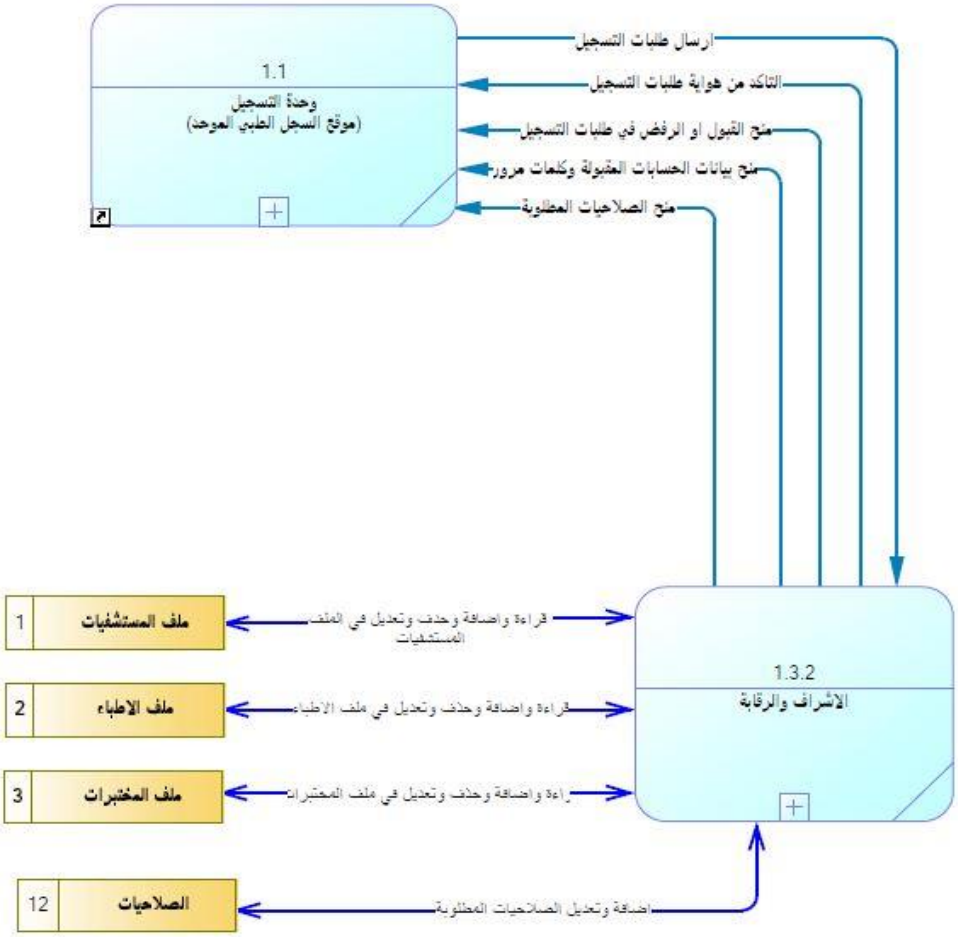
4.1.3.3 المخططات التفصيلية لوحدة الادارة (Level 1)

- **المخطط التفصيلي لإدارة الموقع (Level 1):** –
  - يقوم المشرف على الموقع بإضافة الاخبار للموقع.
  - يقوم المشرف بالتعديل في محتوى الموقع وإصدار التحديثات الدورية للموقع.



رسم توضيحي 13-3 المخطط التفصيلي لإدارة الموقع Level 1

- **المخطط التفصيلي للإشراف والرقابة (Level 1):** –
  - تقوم وحدة التسجيل بإرسال طلبات التسجيل للإدارة.
  - يقوم المشرف بالاطلاع على ملف الاشعارات الخاصة بالمستشفى والمختبر والطبيب.
  - يقوم المشرف بالتأكد من صحة البيانات ومعرفة موقع المستشفى او المختبر ومعرفة المؤهل العلمي للطبيب.
  - يقوم المشرف باعتماد البيانات في حال كانت صحيحة.
  - يقوم المشرف بمنح الصلاحيات اللازمة للحساب المطلوب.
  - يقوم المشرف بإرسال اشعار للمقدم الطلب عبر البريد الإلكتروني او الهاتف.



رسم توضيحي 14-3 المخطط التفصيلي للاشراف والرقابة Level 1

4.2 وصف البيانات (Data Description Data Dictionary)

4.2.1 قاموس البيانات (Data Dictionary):

- **جدول الاطباء (Doctors):**

الوصف	نوع البيانات	القيد	اسم الحقل
رقم الطبيب	Number	P.K	D_NO
اسم الطبيب	Text	–	D_NAME
العنوان	Text	–	D_ADDRESS
البريد الالكتروني	Text	–	D_MAIL
الهاتف	Number	–	D_PHONE
الوظيفة	Text	–	D_JOP
الجنس	Byte	–	D_SEX
الجنسية	Text	–	D_NATIONALITY
الحالة	Byte	–	D_STATE
تاريخ الميلاد	Date	–	D_DATE_OF_BIRTH
تاريخ التسجيل	Date	–	D_DATE_OF_REGISTER

جدول 6-3 جدول الاطباء (Doctors)

- **جدول المرضى (Patients):**

اسم الحقل	القيد	نوع البيانات	الوصف
P_NO	P.K	Number	رقم المريض
P_NAME	–	Text	اسم المريض
P_ADDRESS	–	Text	العنوان
P_DATE_OF_REGISTER	–	Date	تاريخ التسجيل
P_PHONE	–	Number	الهاتف
P_AGE	–	Byte	العمر
P_SEX	–	Byte	الجنس
P_NATIONALITY	–	Text	الجنسية
P_DATE_OF_BIRTH	–	Date	تاريخ الميلاد
P_STATE	–	Byte	الحالة

جدول 7-3 جدول المرضى (Patients)

- **جدول المستشفيات** (Hospitals):

اسم الحقل	القيد	نوع البيانات	الوصف
H_NO	P.K	Number	رقم المستشفى
H_NAME	–	Text	اسم المستشفى
H_ADDRESS	–	Text	العنوان
H_MAIL	–	Varchar	البريد

جدول 8-3 جدول المستشفيات (Hospitals)

- **جدول المختبرات (Labs):**

اسم الحقل	القيد	نوع البيانات	الوصف
L_ON	P.K	Number	رقم المختبر
L_NAME	–	Text	اسم المختبر
L_ADDRESS	–	Text	عنوان المختبر
L_MAIL	–	Text	بريد المختبر

جدول 9-3 جدول المختبرات (Labs)

- **جدول الادوية (Medicines):**

اسم الحقل	القيد	نوع البيانات	الوصف
MED_NO	P.K	Number	رقم الدواء
MED_NAME	–	Text	اسم الدواء

جدول 10-3 جدول الادوية (Medicines)

- **جدول المدراء (Managers):**

اسم الحقل	القيد	نوع البيانات	الوصف
M_NO	P.K	Number	رقم المدير
M_NAME	–	Text	اسم المدير

جدول 11-3 جدول المدراء (Managers)

- **جدول الصلاحيات (Powers):**

اسم الحقل	القيد	نوع البيانات	الوصف
POW_NO	P.K	Number	رقم الصلاحية
M_NO	F.K	Number	رقم المدير
POW_TAIP	-	Text	نوع الصلاحية

جدول 12-3 جدول الصلاحيات (Powers)

- **جدول الأقسام (Sections):**

اسم الحقل	القيد	نوع البيانات	الوصف
H_NO	F.K	Number	رقم المستشفى
D_NO	F.K	Number	رقم الطبيب
S_NO	P.K	Number	رقم القسم
S_NAME	-	Text	اسم القسم

جدول 13-3 جدول الأقسام (Sections)

- **جدول المرضى_الادوية (Patients_Medicines):**

اسم الحقل	القيد	نوع البيانات	الوصف
MED_NO	F.K	Number	رقم الدواء
P_NO	F.K	Number	رقم المريض
POTION	-	Number	الجرعة
TERM	-	Text	فترة الاستخدام
MED_NOT	-	Text	ملاحظات الدواء

جدول 14-3 جدول المرضى_الادوية (Medicines_Patients)

- **جدول المرضى_المختبرات (Patients_ Labs):**

اسم الحقل	القيد	نوع البيانات	الوصف
L_NO	F.K	Number	رقم المختبر
P_NO	F.K	Number	رقم المريض
C_NO	P.K	Number	رقم الفحص
C_NAME	–	Text	اسم الفحص
C_Rustle	–	Text	نتيجة الفحص

جدول 15-3 جدول المرضى_المختبرات (Patients_ Labs)

- **جدول المرضى _الاطباء (Patients_ Doctors):**

اسم الحقل	القيد	نوع البيانات	الوصف
D_NO	F.K	Number	رقم الطبيب
P_NO	F.K	Number	رقم المريض
VIST_DATE	–	Date	تاريخ الزيارة
DIAGNOSIS	–	Text	التشخيص

جدول 16-3 جدول المرضى _الاطباء (Patients_ Doctors)

4.2.2 ترميز البيانات (Data Encoding):

ترميز البيانات وتشفيرها:

هو التعبير عن مفرداتها برمز أو بشفرة مختصرة تدل عليها وتميز بعضها عن بعض من أفضل ما يمكن استخدامه الترميز العددي وذلك للعديد من الميزات التي يتميز بها.

ترميز المرضى:

يحتاج كل مريض الى رقم خاص يميزه عن غيره ويبقى هذا الرقم ثابتاً لا يتغير، وتوجد الكثير من البيانات المريض لكن بعض منها قد يتغير فمثلا قد يذهب المريض الى أكثر من مستشفى فما هو الحل عندما يذهب المريض الى مستشفى؟

هنا سوف تبرز الحاجة الى تغيير الرقم وهذه المشكلة، لذلك اعتمدنا على ترقيم المريض على حسب:

- سنة تسجيل المريض في موقع السجل الطبي الموحد وتمثل بخانتين.
- رقم المستشفى الذي تم التسجيل فيه وتمثل بخانتين.
- رقم تسلسلي للمريض ويمثل 6 خانات.
- وسيصبح الرقم مكوناً من تسع خانات كالتالي:

1	**9**	**02**	**000001**
سنة التسجيل		**رقم المركز الطبي**	**000002 – 999999 رقم متسلسل**

جدول 17-3 جدول ترميز المرضى

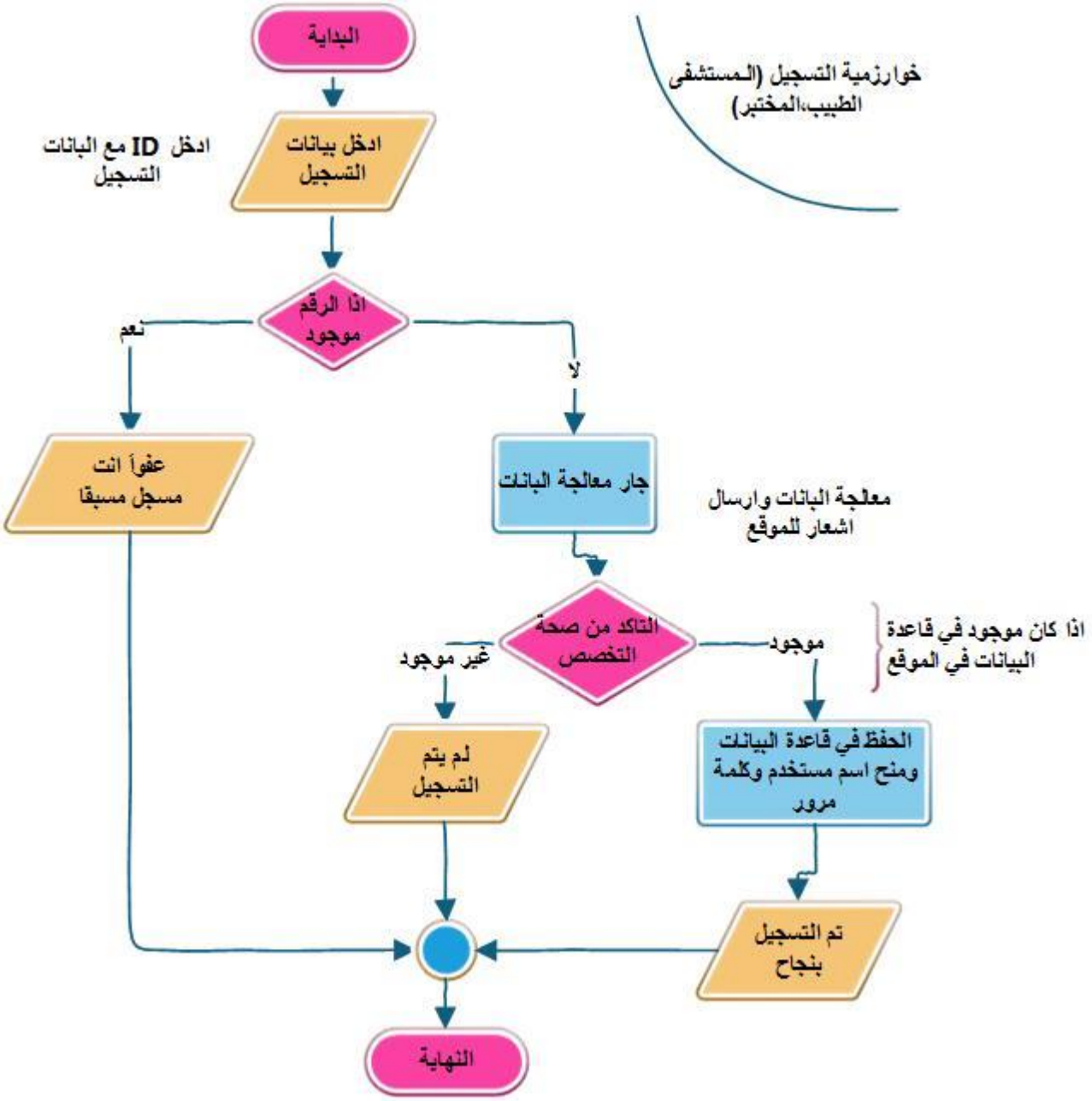
- حيث ان هذا المريض رقم 000001 وتم تسجيله من مستشفى ..... في عام 2019م.

4.3 وصف العمليات (الخوارزميات) (Process Description)

شرح الرمز	الرمز
بداية او نهاية	
قراءة – كتابة (Read – Write)	
معالجة (Processing)	
اتجاه سير العمليات	

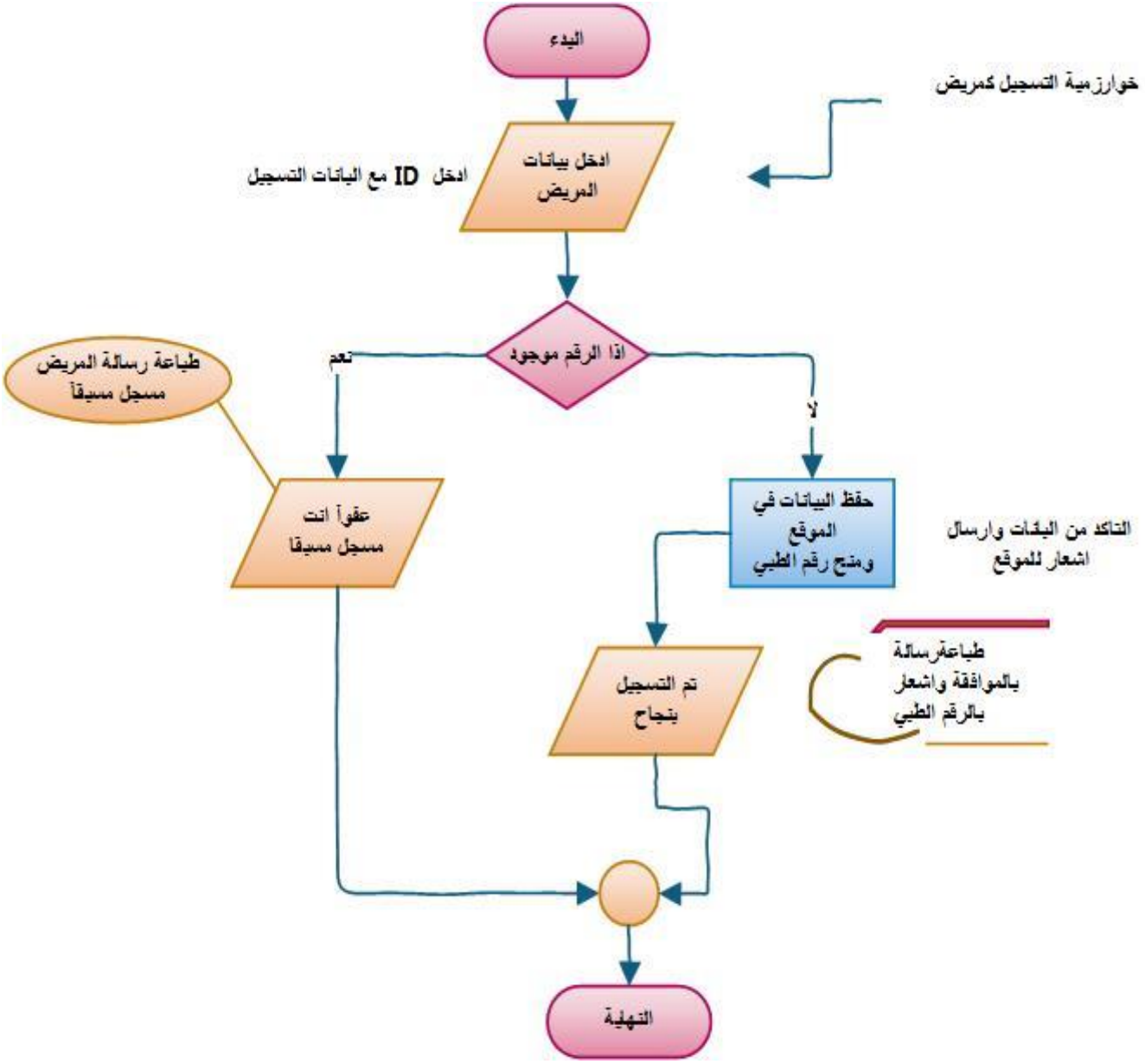
جدول 18-3 يوضح الرموز المستخدمة في الخوارزميات

1- خوارزمية انشاء حساب (مستشفى، مختبر، طبيب):



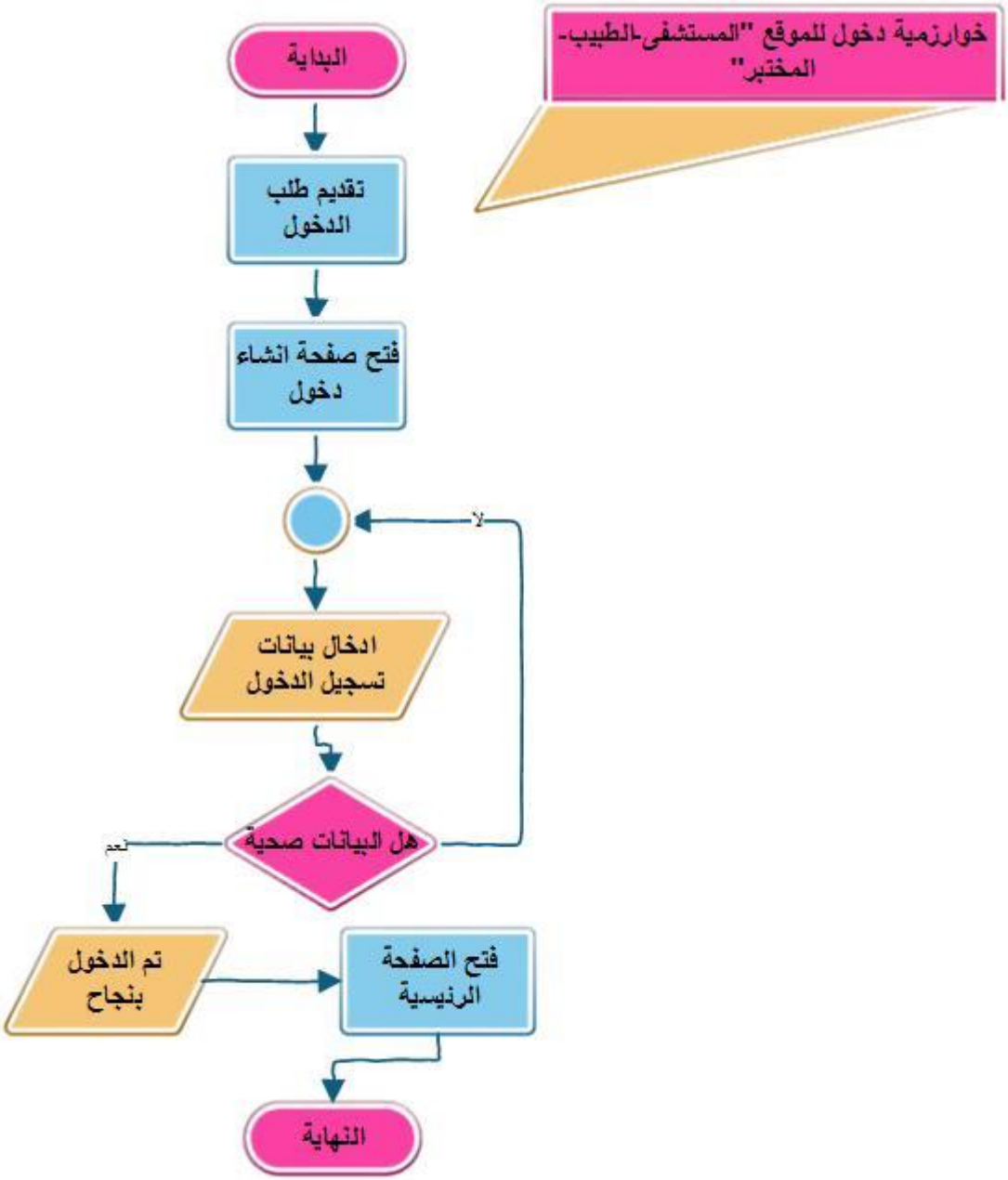
رسم توضيحي 15-3 خوارزمية التسجيل (مستشفى،مختبر،طبيب)

2- خوارزمية انشاء سجل طبي (مريض):



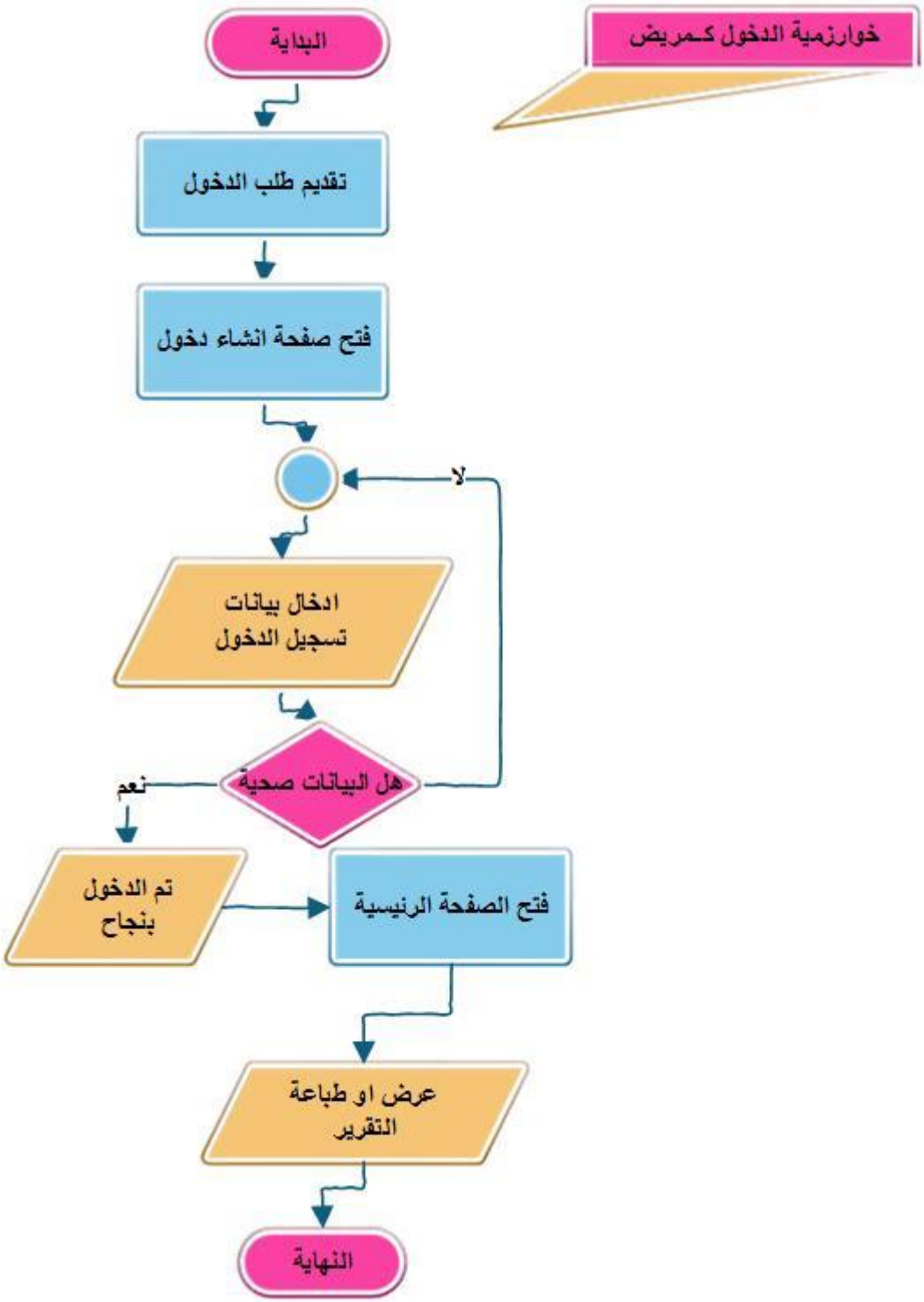
رسم توضيحي 16-3 خوارزمية انشاء سجل طبي (مريض)

3- خوارزمية الدخول (مستشفى، مختبر، طبيب):



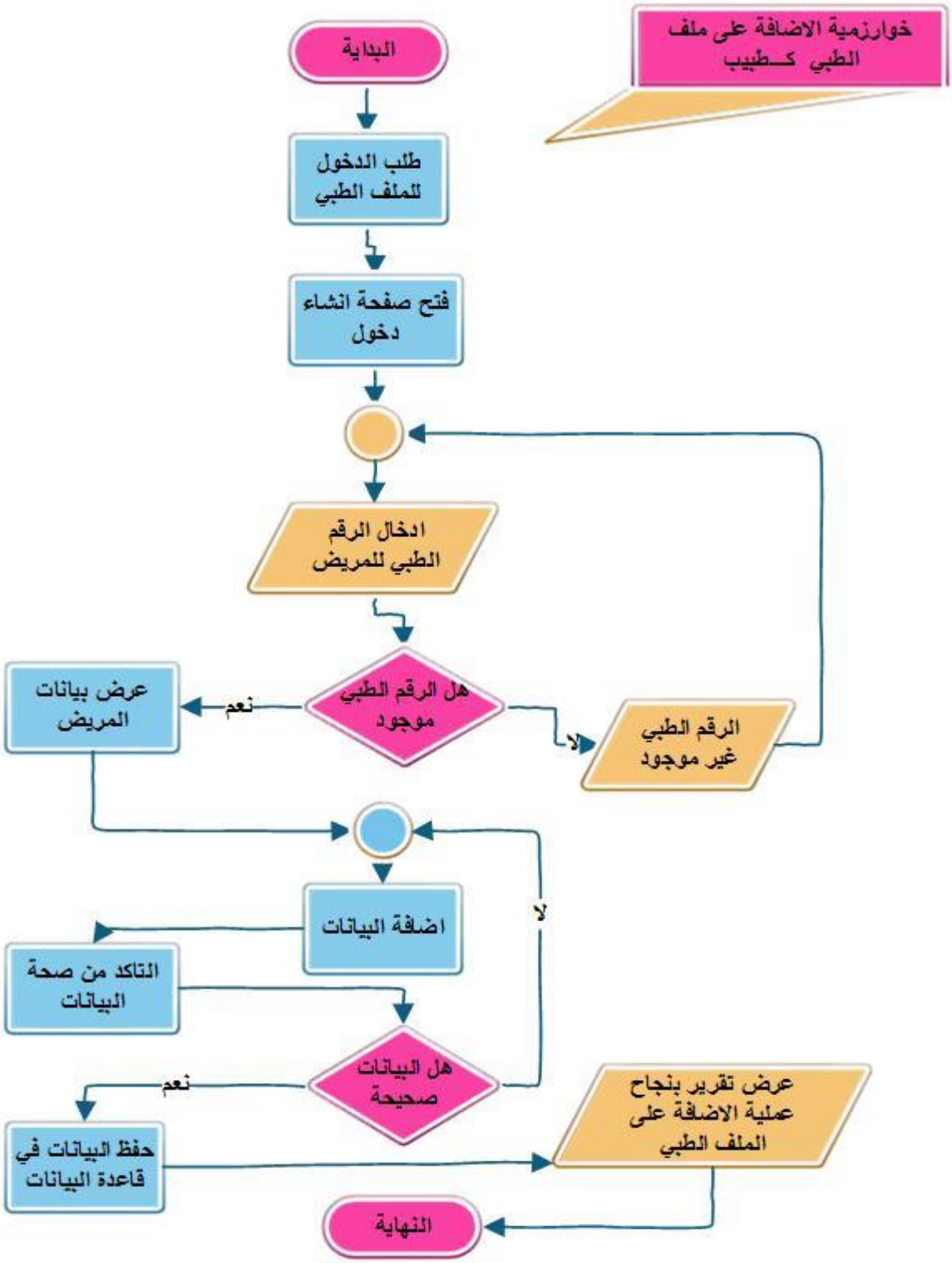
رسم توضيحي 17-3 خوارزمية الدخول (مستشفى، مختبر، طبيب)

4- خوارزمية الدخول (مريض):



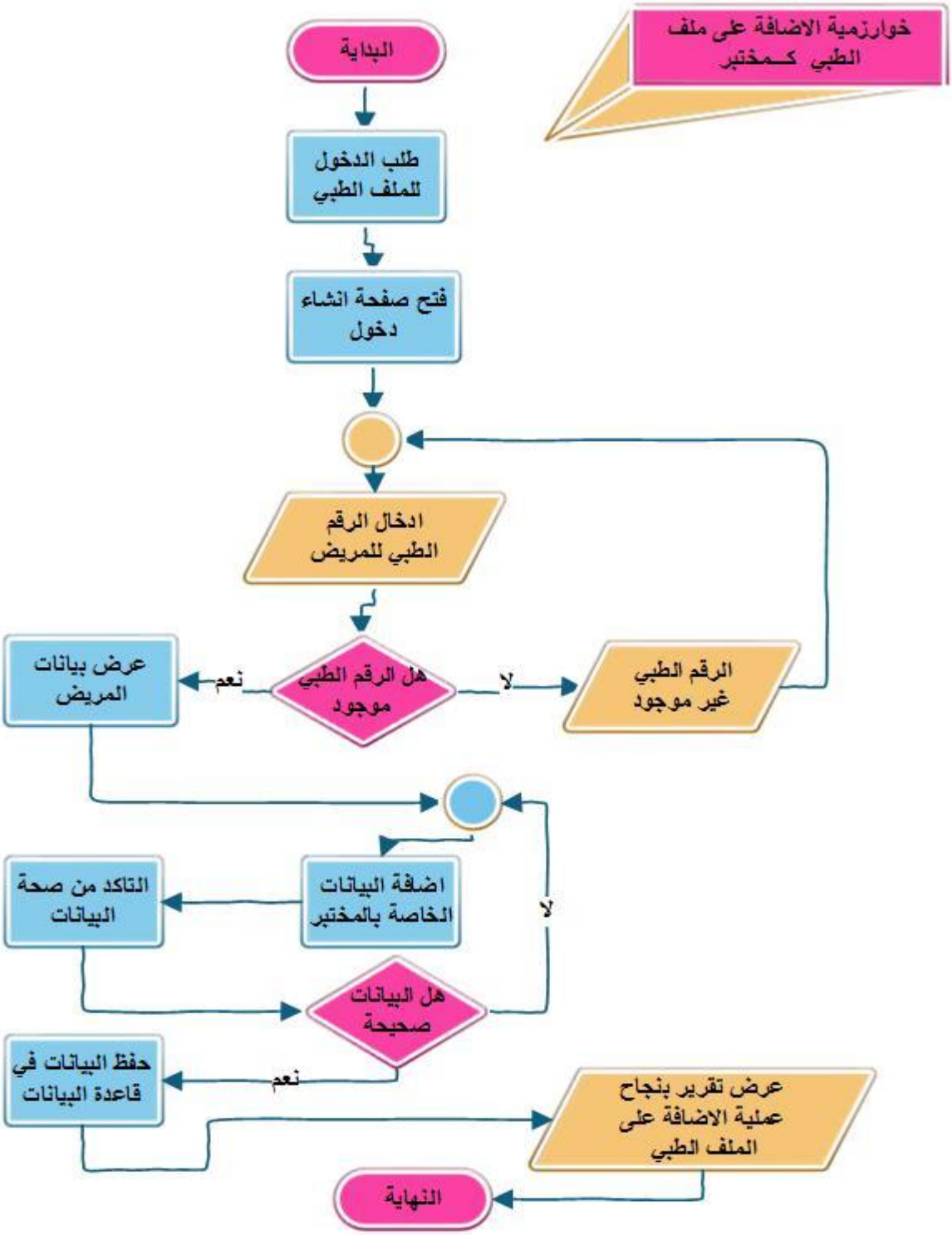
رسم توضيحي 18-3 خوارزمية الدخول (مريض)

5- خوارزمية إضافة للملف الطبي (طبيب):



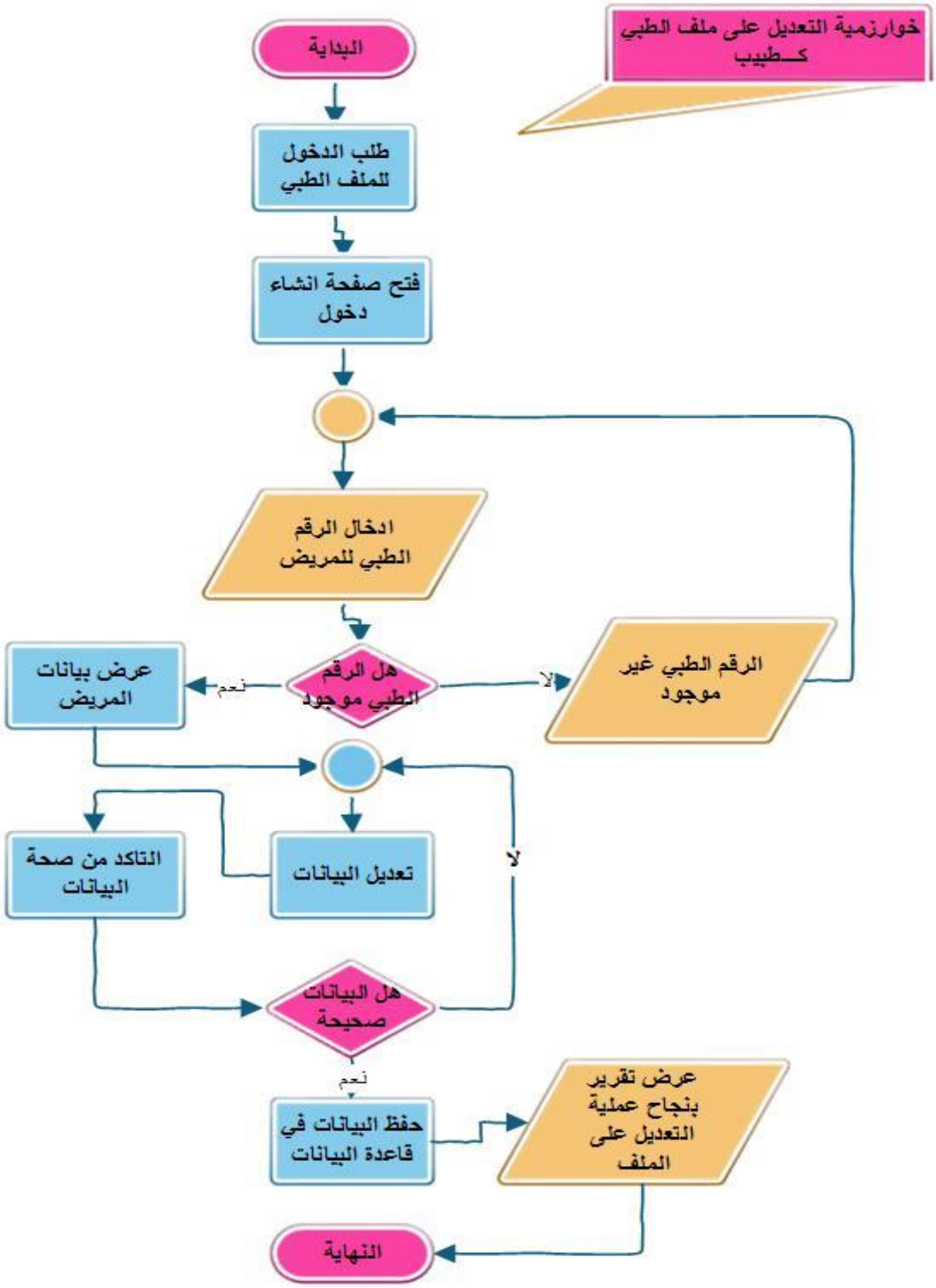
رسم توضيحي 19-3 خوارزمية إضافة للملف الطبي (طبيب)

6- خوارزمية إضافة للملف الطبي (مختبر):



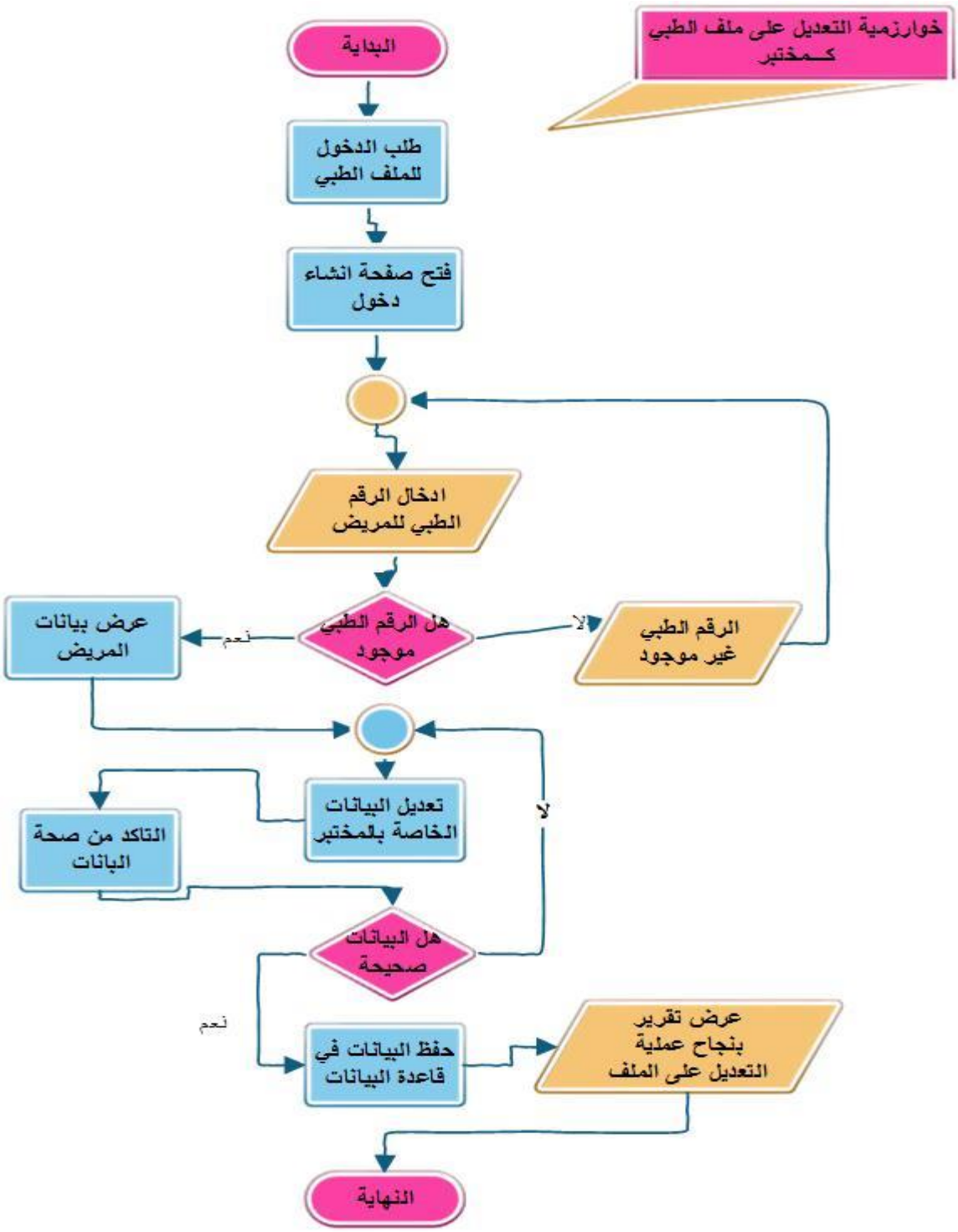
رسم توضيحي 20-3 خوارزمية إضافة للملف الطبي (مختبر)

7- خوارزمية التعديل على الملف الطبي (طبيب):



رسم توضيحي 21-3 خوارزمية التعديل على الملف الطبي (طبيب)

8- خوارزمية التعديل على الملف الطبي (مختبر):



رسم توضيحي 22-3 خوارزمية التعديل على الملف الطبي (مختبر)

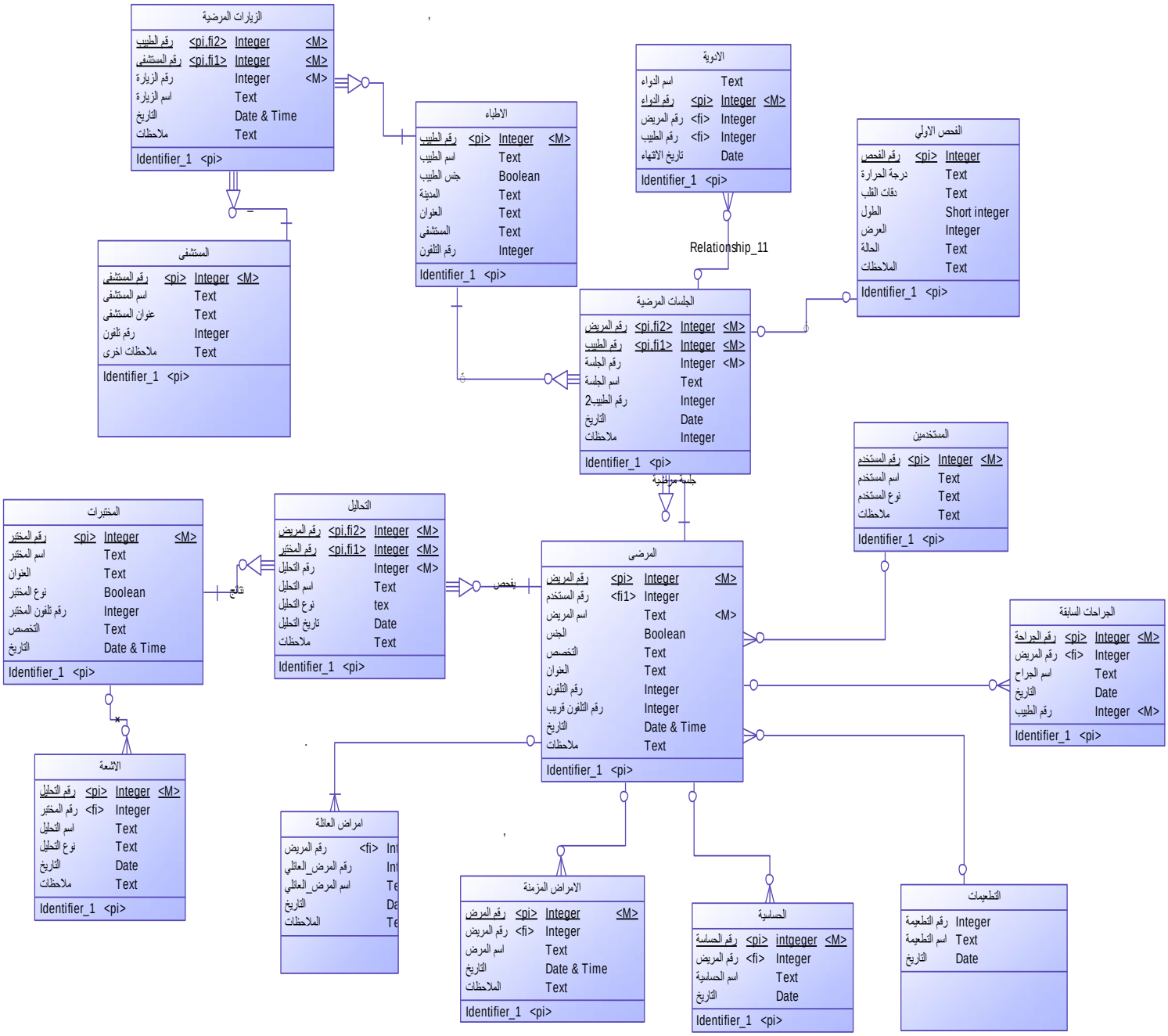
الفصل الرابع

مرحلة توصيف البيانات

(Data Modeling)

3.1 نمذجة البيانات (Data Modeling): –

3.1.1 المخطط العلائقي ERD (Entity relationship diagram)



رسم توضيحي 23-4 المخطط العلائقي لقاعدة البيانات ERD

الفصل الخامس

مرحلة تصميم النظام

(System Design Phase)

5.1 المقدمة: -

تعتمد هذا المرحلة على مرحلة التحليل رغم أهميتها الا انها تعمد على نجاح المحللين في مرحلة التحليل واقترابه من احتياجات المشروع حيث ان مدخلات مرحلة التصميم هي مخرجات عملية التحليل ومخرجاتها هي:

5.2 تصميم قاعدة البيانات (Designing Database):

5.2.1 الجداول والحقول (Tables and Attributes):

1- تصميم جدول للمستشفيات في قاعدة البيانات الاسم البرمجي (Hospital):

المفتاح الرئيسي (Primary key)	الفراغ (Null)	الاطوال (Length)	نوع البيانات (Type)	الاسم البرمجي (Name)	الرقم (Number)
yes	No	4	int	Host_id	1
No	Yes	30	nvachar	Host_name	2
No	Yes	4	int	Host_city_id	3
No	Yes	25	nvachar	Host_address	4
No	Yes	20	nvachar	Host_type	5
No	Yes	15	nvachar	Host_phone	6
No	Yes	15	nvachar	Host_phone_eng	7
No	Getdate()	4	date	Host_date_reg	8
No	Yes	32	int	Host_user_id	9
No	Yes	32	int	Host_num_log	10

جدول 19-5 تصميم جدول المستشفيات في قاعدة البيانات

2- تصميم جدول للأطباء في قاعدة البيانات الاسم البرمجي (Doctor):

المفتاح الرئيسي (Primary key)	الفراغ (Null)	الاطوال (Length)	نوع البيانات (Type)	الاسم البرمجي (Name)	الرقم (Number)
yes	No	4	int	doct_id	1
No	Yes	30	nvachar	doct _name	2
No	Yes	1	bit	doct _gender	3
No	Yes	20	nvachar	doct _address	4
No	Yes	15	nvachar	doct _elig	5
No	Yes	4	int	Doct_Sections	6
No	Getdate()	4	date	doct _brith	7
No	Getdate()	4	date	doct _date_reg	8
No	Yes	4	int	doct _user_id	9
No	Yes	4	int	doct _num_log	10
No	Yes	4	Int	Doct_city	11
No	Yes	15	nvachar	Doct_phone	12

جدول 5-20 تصميم جدول للأطباء في قاعدة البيانات

3- تصميم جدول للمختبرات في قاعدة البيانات الاسم البرمجي (laboratory):

المفتاح الرئيسي (Primary key)	الفراغ (Null)	الاطوال (Length)	نوع البيانات (Type)	الاسم البرمجي (Name)	الرقم (Number)
yes	No	4	int	Lab_id	1
No	Yes	20	nvachar	Lab_name	2
No	Yes	4	int	Lab _city_id	3
No	Yes	20	nvachar	Lab _address	4
No	Yes	15	nvachar	Lab _type	5
No	Yes	15	nvachar	Lab _phone	6
No	Yes	15	nvachar	Lab _phone_eng	7
No	Getdate()	//	date	Lab _date_reg	8
No	Yes	4	int	Lab _user_id	9
No	Yes	4	int	Lab _num_log	10
No	Yes	30	nvachar	Lab _num_eng	11

جدول 5-21 تصميم جدول للمختبرات في قاعدة البيانات

4- تصميم جدول المرضى في قاعدة البيانات الاسم البرمجي (Patients):

المفتاح الرئيسي (Primary key)	الفراغ (Null)	الاطوال (Length)	نوع البيانات (Type)	الاسم البرمجي (Name)	الرقم (Number)
yes	No	4	int	P_id	1
No	Yes	30	nvachar	P _name	2
No	Yes	1	bit	P _gender	3
No	Getdate()	4	date	P _brith	4
No	Yes	20	nvachar	P _work	5
No	Yes	15	nvachar	P _type_identity	6
No	Yes	15	nvachar	P _num_identity	7
No	Yes	4	int	P_city	8
No	Yes	30	nvachar	P_address	9
No	Yes	15	nvachar	P_phone	10
No	Getdate()	4	date	P_date_reg	11
No	Yes	4	int	P_user_id	12
No	Yes	4	int	P_blood_type	13
No	Yes	2	image	P_image	14

جدول 22-5 تصميم جدول المرضى في قاعدة البيانات

5- تصميم جدول المستخدمين في قاعدة البيانات الاسم البرمجي (Users):

المفتاح الرئيسي (Primary key)	الفراغ (Null)	الاطوال (Length)	نوع البيانات (Type)	الاسم البرمجي (Name)	الرقم (Number)
yes	No	4	int	Users_id	1
No	Yes	30	nvachar	User _name	2
No	Yes	30	nvachar	User_name_logo	3
No	Yes	max	nvachar	Users _pass	4
No	Yes	1	bit	Users _active	5
No	Yes	4	int	Users _type	6
No	() Getdate	4	date	Users _date_eng	7

جدول 23-5 تصميم جدول المستخدمين في قاعدة البيانات

6- تصميم جدول الأدوية في قاعدة البيانات الاسم البرمجي (Medicine):

المفتاح الرئيسي (Primary key)	الفراغ (Null)	الاطوال (Length)	نوع البيانات (Type)	الاسم البرمجي (Name)	الرقم (Number)
yes	No	4	int	Medic_id	1
No	Yes	40	nvachar	Medic _name	2
No	Yes	40	nvachar	Medic _dose	3
No	Yes	30	nvachar	Medic _rng_use	4
No	Getdate()	4	date	Medic _date_reg	6
No	Yes	4	int	Medic _sess_id	7

جدول 24 -5 تصميم جدول الأدوية في قاعدة البيانات

7- تصميم جدول التحاليل في قاعدة البيانات الاسم البرمجي (Analyzes):

المفتاح الرئيسي (Primary key)	الفراغ (Null)	الاطوال (Length)	نوع البيانات (Type)	الاسم البرمجي (Name)	الرقم (Number)
yes	No	4	int	Analyz_id	1
No	Yes	30	nvachar	Analyz _name	2
No	Yes	10	nvachar	Analyz _type	3
No	Yes	15	nvachar	Analyz _matter	4
No	Yes	max	nvachar	Analyz _resu	5
No	Yes	max	nvachar	Analyz_diagnosis	6
No	() Getdate	4	date	Analyz_date_reg	7
No	Yes	4	int	Analyz_sess_id	8
No	Yes	4	int	Analyz _ Lab _id	9
No	Yes	30	nvachar	Analyz_name_level	10

جدول 25-5 تصميم جدول التحاليل في قاعدة البيانات

5.2.2 تصميم القاعدة (Data Constrains) :

تم تصـــــميم قاعدة البيانات بلغة SQL SERVER وقاعدة البيانات مكونة من 30 جدول في هذا المرحلة سوف نتطرق الى الجداول الرئيسة تم تصميم القاعدة عن طريق كتابة الأكواد الاتية:

1- جدول المستشفيات:

```
CREATE TABLE [dbo].[TB_hospital]
    [host_id] [int] IDENTITY(1,1) NOT NULL,
    [host_name] [nvarchar](30) NULL,
    [host_city_id] [int] NULL,
    [host_address] [nvarchar](25) NULL,
    [host_type] [nvarchar](20) NULL,
    [host_phone] [nvarchar](15) NULL,
    [host_phone_eng] [nvarchar](15) NULL,
    [host_date_reg] [date] NULL,
    [host_user_id] [int] NULL,
    [host_num_log] [int] NULL,
    [host_name_eng] [nvarchar](30) NULL,
 CONSTRAINT [PK_TB_hospital] PRIMARY KEY CLUSTERED
```

2- جدول الأطباء:

```
CREATE TABLE [dbo].[TB_doctors](
    [doct_id] [int] IDENTITY(1,1) NOT NULL,
    [doct_name] [nvarchar](30) NULL,
    [doct_gender] [bit] NULL,
    [doct_city] [int] NULL,
    [doct_address] [nvarchar](20) NULL,
    [doct_elig] [nvarchar](15) NULL,
    [doct_sections] [int] NULL,
    [doct_brith] [date] NULL,
    [doct_date_reg] [date] NULL,
    [doct_host_id] [int] NULL,
    [doct_user_id] [int] NULL,
    [doct_num_log] [int] NULL,
    [doct_phone] [nvarchar](15) NULL,
    [doct_nat] [int] NULL,
 CONSTRAINT [PK_TB_doctors] PRIMARY KEY CLUSTERED
```

3- جدول المختبرات:

```
CREATE TABLE [dbo].[TB_laboratory](
    [lab_id] [int] IDENTITY(1,1) NOT NULL,
    [lab_name] [nvarchar](20) NULL,
    [lab_city_id] [int] NULL,
    [lab_address] [nvarchar](20) NULL,
    [lab_tybe] [nvarchar](15) NULL,
    [lab_phone] [nvarchar](15) NULL,
    [lab_phone_eng] [nvarchar](15) NULL,
    [lab_date_reg] [date] NULL,
    [lab_user_id] [int] NULL,
    [lab_num_log] [int] NULL,
    [lab_name_eng] [nvarchar](30) NULL,
 CONSTRAINT [PK_TB_laboratory] PRIMARY KEY CLUSTERED
```

4- جدول المرضى:

```
CREATE TABLE [dbo].[TB_patients](
    [p_id] [int] IDENTITY(1,1) NOT NULL,
    [p_name] [nvarchar](30) NULL,
    [p_gender] [bit] NULL,
    [p_birth] [date] NULL,
    [p_work] [nvarchar](20) NULL,
    [p_type_Identity] [nvarchar](15) NULL,
    [p_num_Identity] [nvarchar](15) NULL,
    [p_city] [int] NULL,
    [p_address] [nvarchar](30) NULL,
    [p_phone] [nvarchar](15) NULL,

 CONSTRAINT [PK_TB_patients] PRIMARY KEY CLUSTERED
```

5- جدول المستخدمين:

```
CREATE TABLE [dbo].[TB_users](
    [users_Id] [int] IDENTITY(1,1) NOT NULL,
    [user_name] [nvarchar](30) NULL,
    [user_name_login] [nvarchar](30) NULL,
    [user_pass] [nvarchar](max) NULL,
    [user_type] [int] NULL,
    [user_active] [bit] NULL,
    [user_date_eng] [date] NULL,
 CONSTRAINT [PK_TB_users] PRIMARY KEY CLUSTERED
```

6- جدول الأدوية:

```
CREATE TABLE [dbo].[TB_Medicine](
    [medic_id] [int] IDENTITY(1,1) NOT NULL,
    [medic_name] [nvarchar](40) NULL,
    [med_dose] [nvarchar](40) NULL,
    [med_rng_use] [nvarchar](30) NULL,
    [med_date_reg] [date] NULL,
    [med_sess_id] [int] NULL,
 CONSTRAINT [PK_TB_Medicine] PRIMARY KEY CLUSTERED
(
```

7- جدول التحاليل:

```
CREATE TABLE [dbo].[TB_analyzes](
    [analyz_id] [int] IDENTITY(1,1) NOT NULL,
    [analyz_name] [nvarchar](30) NULL,
    [analyz_tybe] [nvarchar](10) NULL,
    [analyz_matter] [nvarchar](15) NULL,
    [analyz_resu] [nvarchar](max) NULL,
    [analyz_diagnosis] [nvarchar](max) NULL,
    [analyz_date_reg] [date] NULL,
    [analyz_sess_id] [int] NULL,
    [analyz_lab_id] [int] NULL,
    [analyz_name_level] [nvarchar](30) NULL,
 CONSTRAINT [PK_TB_analyzes] PRIMARY KEY CLUSTERED
```

5.2.3 جمل الاستعلامات (Retrieve Statements):

هناك أكثر من 70 جملة للاستعلام في قاعدة البيانات تخص جلب البيانات وفلترة المعلومات حسب طلب المستخدم منها ما يجلب تقارير خاصة ومنها ما يجلب معلومات على الشاشة وعلى موقع الويب الخاص وتتوزع على معلومات كلا من (المستشفيات – المختبرات – الأطباء – المرضى (الملف الطبي) – المستخدمين – الإعلانات – الأخبار) من أجل عرض سهولة الوصل للمعلومات بسرعة وبدقة حماية تم أنشاء جمل الاستعلام في الإجراءات المخزنة (Stored Procedures) في قاعدة البيانات سيتم عرض بعض هذا الاستعلامات:

1- استعلام عن معلومات المستشفيات محدد:

```
Create proc [dbo].[GET_HOSP_single]
@ID nvarchar(10)
as
SELECT [host_id]
      ,[host_name]
      ,[city_name]
      ,[host_address]
      ,[host_type]
      ,[host_phone]
      ,[host_phone_eng]
      ,[host_date_reg]
      ,[user_name_login]
      ,[user_pass]
      ,[host_num_log]
      ,[host_name_eng]
  FROM [dbo].[TB_hospital]

  INNER join TB_citys
ON TB_citys.city_id=TB_hospital.host_city_id
 INNER join TB_users
 on TB_users .users_Id = TB_hospital .host_user_id
where host_id=@ID
```

2- استعلام عن كل المستخدمين:

```
create proc [dbo].[GET_ALL_USER]
as
SELECT *FROM [dbo].[TB_users]

INNER join TB_type_user
ON TB_type_user.user_id= TB_users.user_type
```

3- استعلام عن معلومات الشخصية للمريض محدد:

```
create proc [dbo].[GET_FILE_PATIENTS]
@ID nvarchar(15)
as
SELECT [p_id]
      ,[p_name]
      ,[gender_name]
      ,[p_birth]
      ,[p_work]
      ,[p_type_ldentity]
      ,[p_num_ldentity]
      ,[city_name]
      ,[p_address]
      ,[p_phone]
      ,[p_phone_neer]
      ,[p_date_reg]
      ,[p_user_id]
      ,[nat_name]
      ,[p_insurance]
      ,[blood_name]
      ,[p_image]
      ,[p_code]
  FROM [dbo].[TB_patients]

  INNER join TB_gender
ON TB_gender.gender_id=TB_patients.p_gender
INNER join TB_citys
ON TB_citys.city_id=TB_patients.p_city
INNER join TB_nationality
ON TB_nationality.nat_id=TB_patients.p_nat
INNER join TB_blood_type
ON TB_blood_type.blood_id=TB_patients.p_blood_type

where p_code=@ID
```

4- استعلام عن جميع التحاليل:

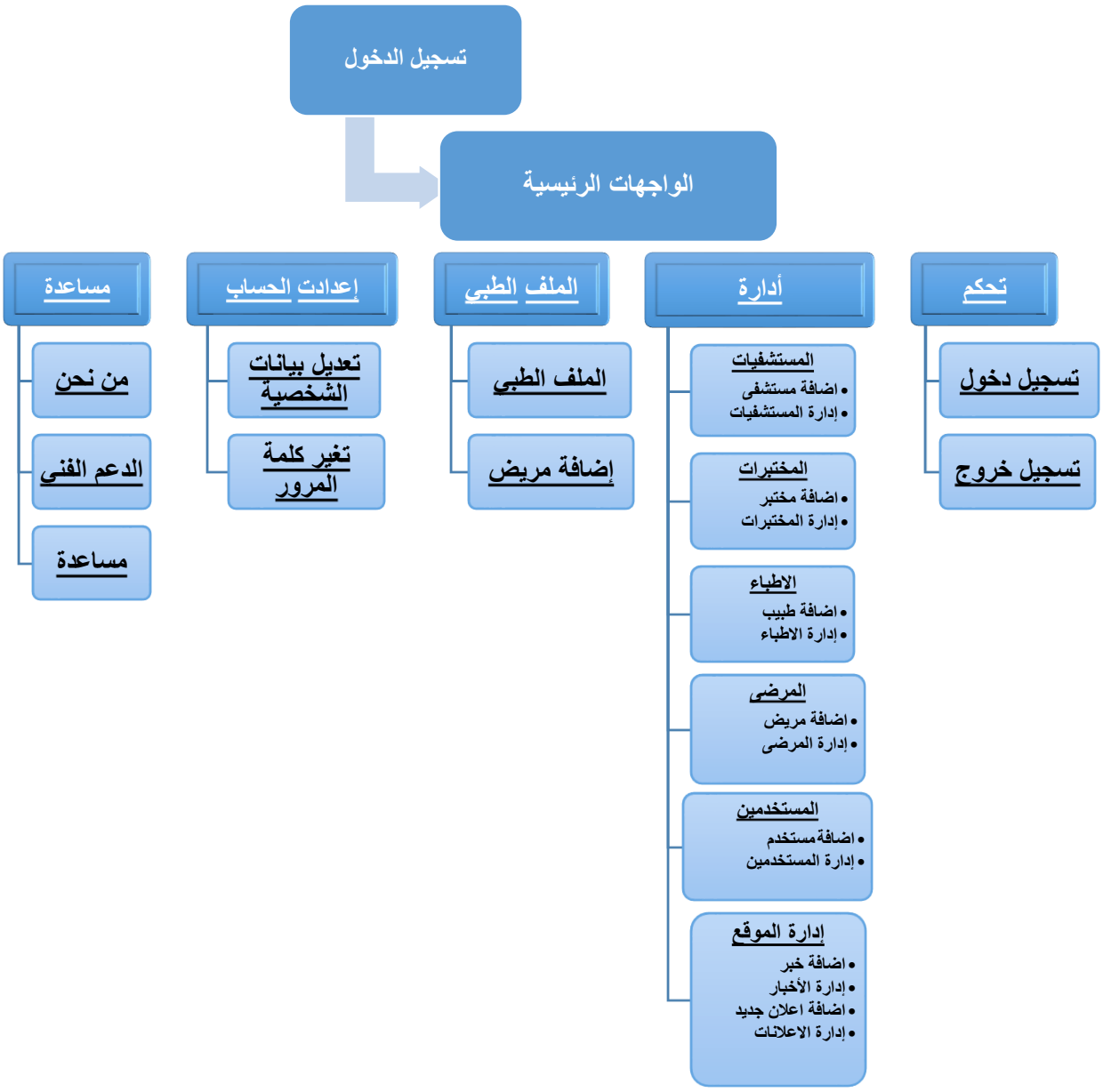
```
create proc [dbo].[GET_ALL_analyzes_PAT]
@ID nvarchar(10)
as

SELECT *
  FROM [dbo].[TB_analyzes]

  where analyz_sess_id=@ID
```

5- استعلام البحث عن مريض بكل البيانات الموجودة:

```
ALTER proc [dbo].[SEARCH_PATIENT]
@ID nvarchar(30)
as
SELECT [p_id]
      ,[p_name]
      ,[gender_name]
      ,[p_birth]
      ,[p_work]
      ,[nat_name]
      ,[p_type_Identity]
      ,[p_num_Identity]
      ,[city_name]
      ,[p_address]
      ,[p_phone]
      ,[p_phone_neer]
      ,[blood_name]
      ,[p_insurance]
      ,[p_date_reg]
      ,[p_user_id]
      ,[p_code]
  FROM TB_patients

INNER join TB_gender
ON TB_gender.gender_id= TB_patients.p_gender

INNER join TB_citys
ON TB_citys.city_id=TB_patients.p_city

INNER join TB_nationality
ON TB_nationality.nat_id=TB_patients.p_nat

INNER join TB_blood_type
ON TB_blood_type.blood_id=TB_patients.p_blood_type

where
CONVERT(nvarchar,[p_id])+
[p_name]+
[gender_name]+
CONVERT(nvarchar,[p_birth])+
[p_work]+
[nat_name]+
[p_type_Identity]+
[p_num_Identity]+
[city_name]+
[p_address]+
[p_phone]+
[p_phone_neer]+
[blood_name]+
[p_insurance]+
CONVERT(nvarchar,[p_date_reg])+
CONVERT(nvarchar,[p_user_id])+
CONVERT(nvarchar,[p_code])
like '%' + @ID + '%'
```

5.3 تصميم واجهات المستخدم (Designing User Interface):

5.3.1 هيكل الواجهات (Interface Structure):

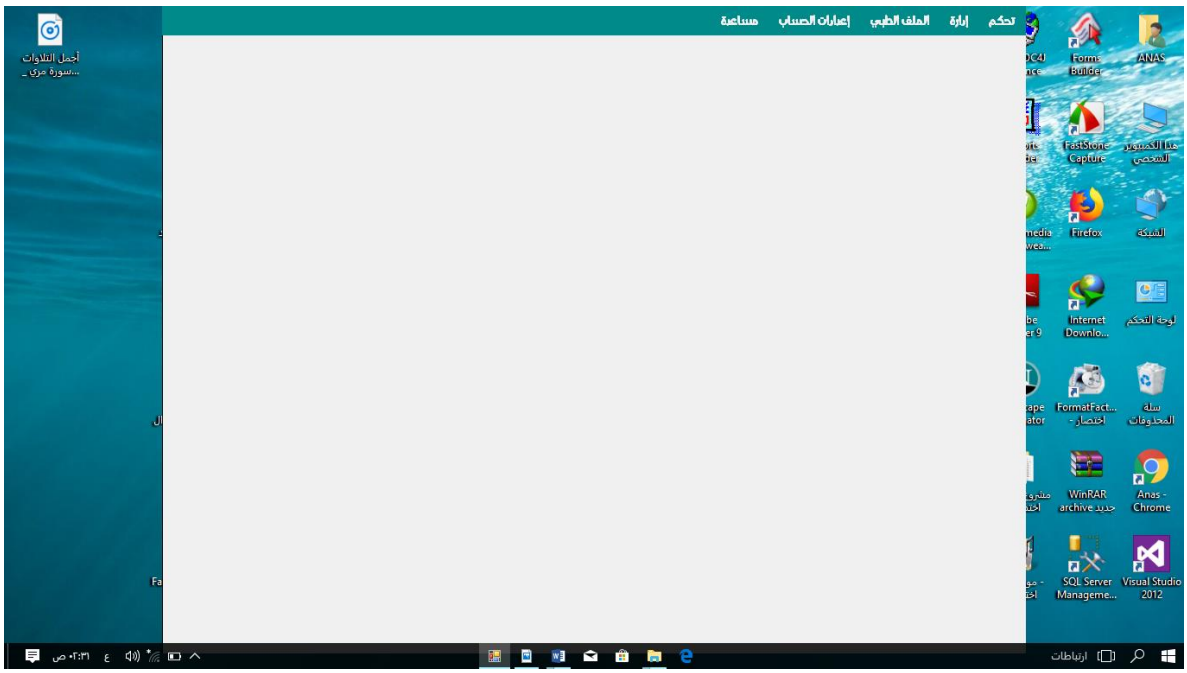


رسم توضيحي 5-24 هيكل الواجهات

5.3.2 الواجهات الرئيسة (Main Screen):

1- شاشة الدخول للإصدار الكمبيوتر (C#.net):

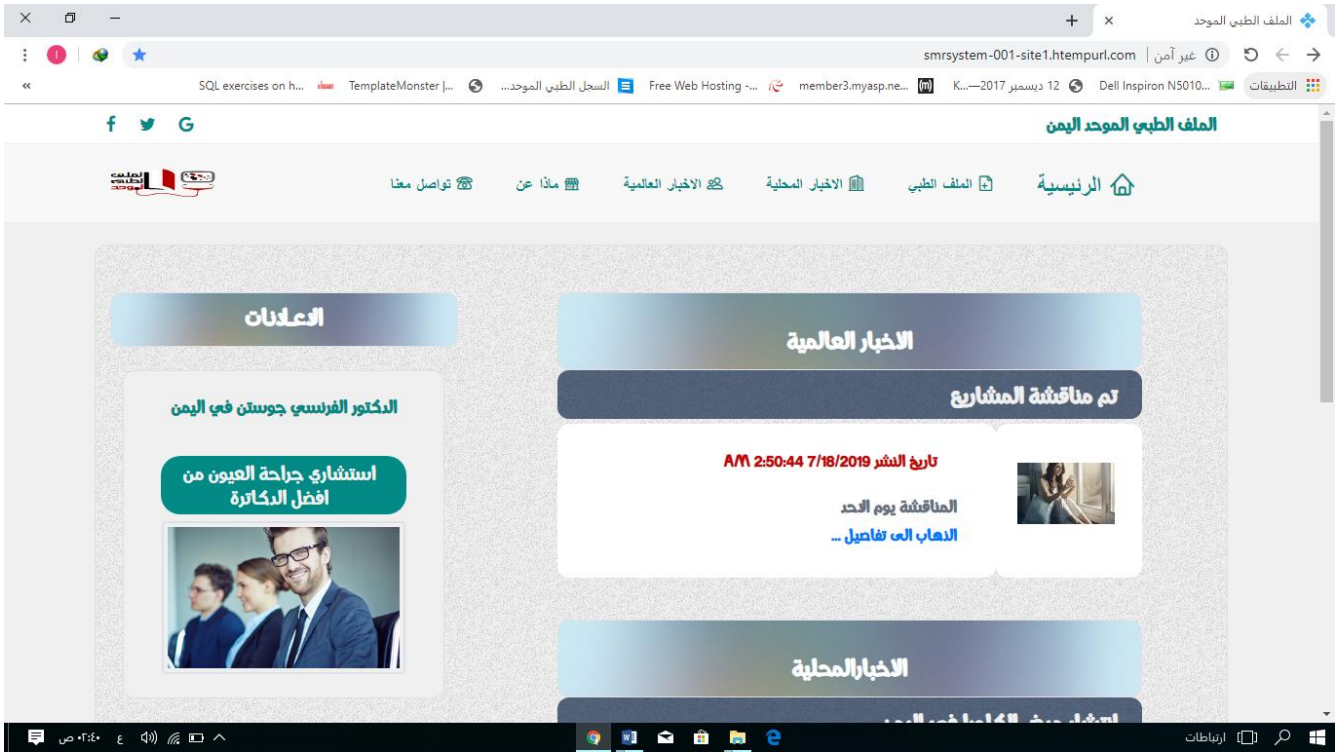
- الشاشة الرئيسية للإصدار الكمبيوتر (C#.net):



رسم توضيحي 25-5 الشاشة الرئيسية للإصدار الكمبيوتر (C#.net)

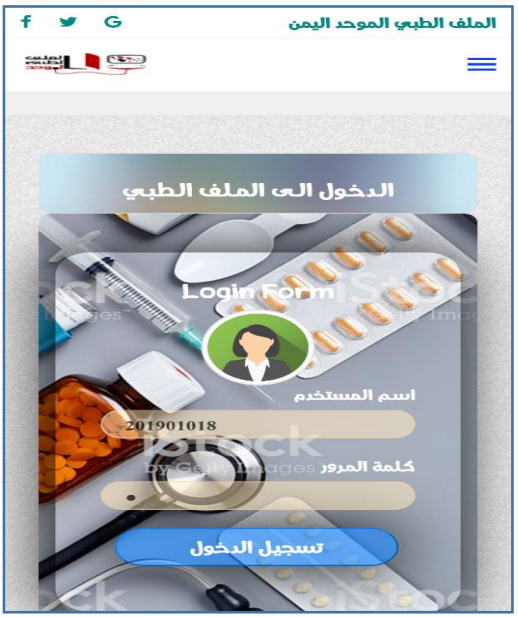
2- شاشة الدخول للملف الطبي للإصدار الويب (Asp.net):

- الشاشة الرئيسية للإصدار الويب (Asp.net):



رسم توضيحي 26-5 الشاشة الرئيسية للإصدار الويب (Asp.net)

3- شاشة الدخول للملف الطبي للإصدار الهاتف (Android):



- شاشة الرئيسية للإصدار الهاتف (Android):

رسم توضيحي 27-5 شاشة الرئيسية للإصدار الهاتف (Android)

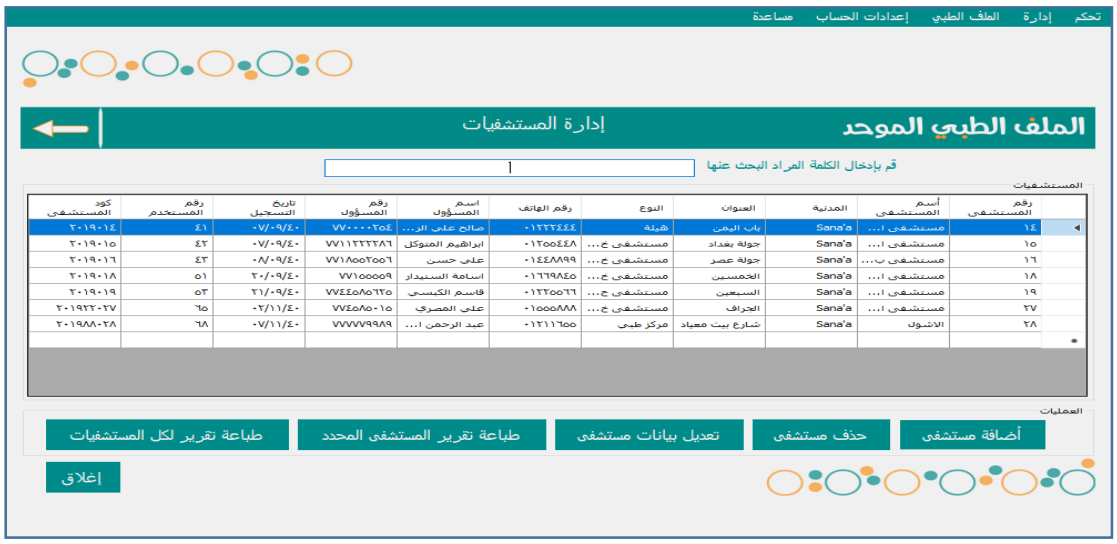
5.3.3 واجهات الإضافة الى النظام (Add information Screens):

1- شاشة إدخال بيانات مستشفى:

عبر هذي الواجهة يمكننا إضافة بيانات مستشفى من خلال مربع الاسم والنوع والمدينة ورقم الهاتف وأيضا اسم مدير المستشفى ويتم حفظ البيانات عبر زر الحفظ الى قاعدة البيانات، وتتم العملية من خلال (Desktop):

2- شاشة إدارة المستشفيات

عبر هذي الواجهة يمكننا عرض بيانات المستشفيات من خلال مربع البحث بالاسم او الرقم او المدينة او رقم الهاتف او اسم مدير المستشفى من خلال أي نص يوجد في معرف المستشفى مما يسرع ويسهل عملية البحث كما يوجد في هذا الشاشة ايضاً العمليات الخاصة بالمستشفى (إضافة او حذف او تعديل مستشفى من قاعدة البيانات):

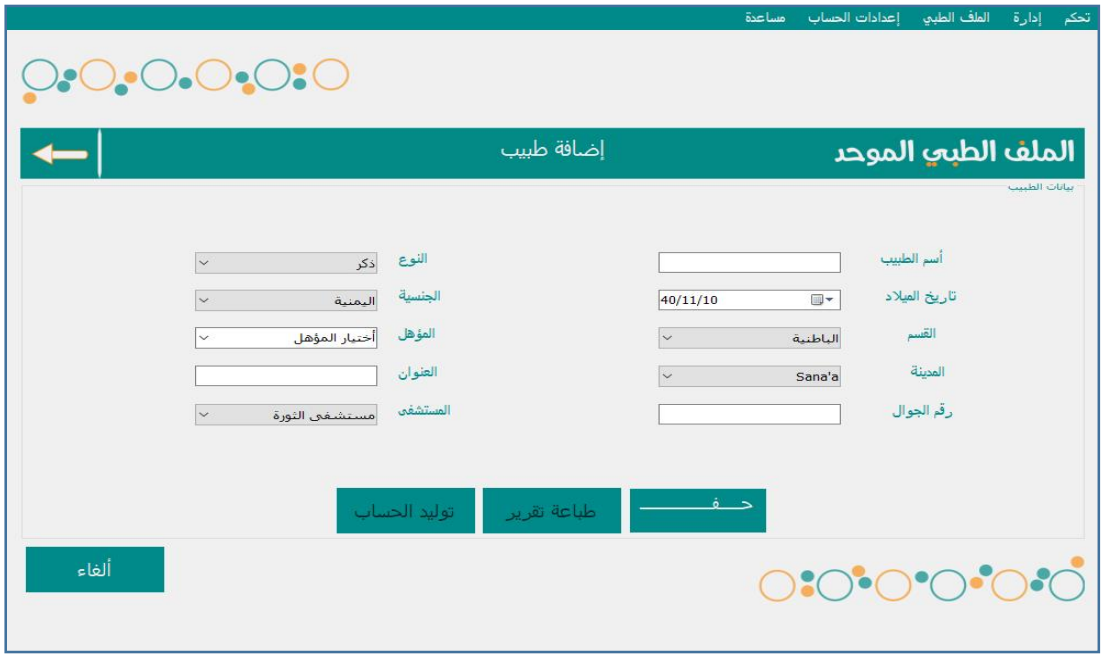


3- شاشة إدخال بيانات مختبر:

عبر هذي الواجهة يمكننا إضافة بيانات مختبر من خلال مربع الاسم والنوع والمدينة والعنوان ورقم الهاتف وأيضا اسم مسؤول المختبر ورقمة ويتم حفظ البيانات عبر زر الحفظ الى قاعدة البيانات.

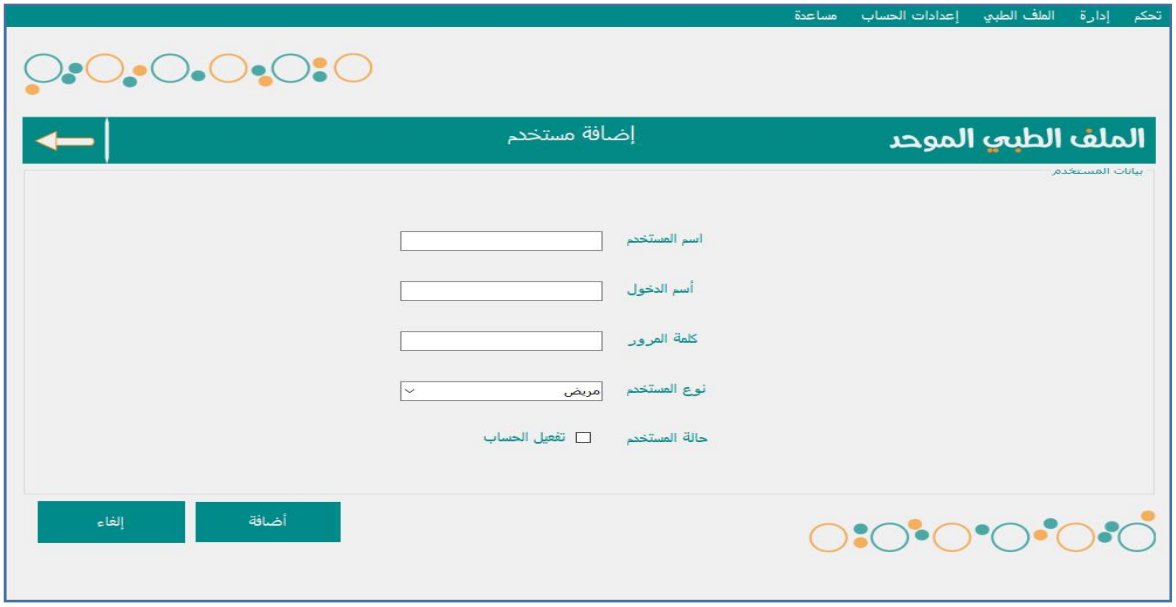
4- شاشة إدخال بيانات طبيب:

عبر هذي الواجهة يمكننا إضافة بيانات الطبيب من خلال مربع الاسم والنوع والمدينة ورقم الهاتف وأيضا المؤهل والقسم والجنسية ويتم حفظ البيانات عبر زر حفظ في قاعدة البيانات.



5- شاشة إدخال بيانات مستخدم:

عبر هذي الواجهة يمكننا إضافة بيانات المستخدم من خلال مربع الاسم كلمة المرور اسم الحركي للدخول وأيضا نوع المستخدم هل هو طبيب او مستشفى او طبيب... وحالة حسابة ويتم حفظ البيانات عبر زر حفظ في قاعدة البيانات.



6- شاشة إدارة المستخدمين:

عبر هذي الواجهة يمكننا عرض بيانات المستخدمين من خلال مربع البحث بالاسم او الرقم او نوع او تاريخ التسجيل من خلال أي نص يوجد في معرف المستخدمين مما يسرع ويسهل عملية البحث كما يوجد في هذا الشاشة ايضاَ العمليات الخاصة بالمستخدمين (إضافة او حذف او تعديل مستخدم من قاعدة البيانات):

7- شاشة إضافة الاخبار للموقع:

إضافة خبر للموقع من خلال عنوان الخبر ونبذه عن الخبر ثم التفاصيل وحالة الخبر مما يسهل التعميم وافادة الجهات المختصة والمستفيدة في تمرير ووصول المعلومة للمناطق المستهدفة:

8- شاشة إدارة الإعلانات الخاصة بالموقع:

عبر هذي الواجهة يمكننا عرض بيانات الاعلانات من خلال مربع البحث بالرقم او عنوان او تفاصيل الاعلان او من تاريخ الإعلان من خلال أي نص يوجد في معرف الإعلانات مما يسرع ويسهل عملية البحث كما يوجد في هذا الشاشة ايضاَ العمليات الخاصة بالإعلانات (إضافة او حذف او تعديل مستخدم من قاعدة البيانات):

9- شاشة إضافة سجل طبي:

عن طريق ادخال البيانات الأساسية للمريض التي تشمل معلومات الهوية الشخصية إضافة للمهنة والعنوان ورقم الجوال للمريض واحد أقاربه والجنسية وفصيلة الدم وصورة المريض الشخصية:

رسم توضيحي 28-5 شاشة إضافة سجل طبي

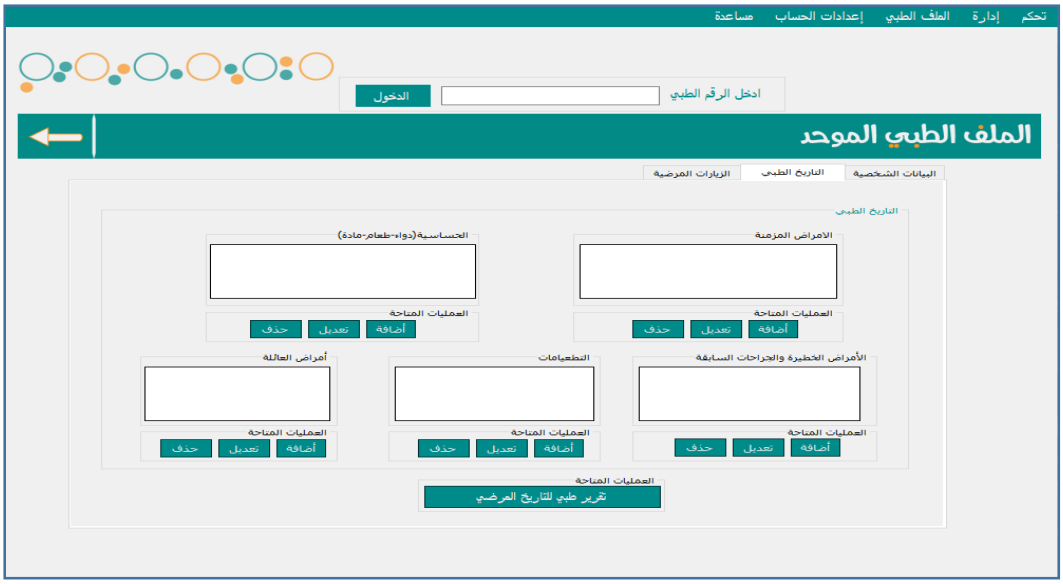
10- شاشة الملف الطبي:

هي عبارة عن شاشة باعلاها يوجد زر بحث بالرقم الطبي للمريض والدخول اليه وكما يتم عرض 3 نوافذ عن معلومات التفصيلية للمريض التي تتمثل بالبيانات الشخصية والتاريخ الطبي والزيارات المرضية:

أ_ البيانات الشخصية للمريض من خلال بيانات الهوية الشخصية ورقم الجوال المريض واحد أقاربه وفصيلة الدم والجنسية وتاريخ التسجيل:

ب_ التاريخ الطبي في النافذة المنبثقة الوسطى:

يحتوي على بيانات التاريخ الطبي من خلال عرض الامراض المزمنة والحساسية من الدواء او الطعام او أي مواد ويحتوي على الامراض الخطيرة والجراحات السابقة وإمكانية إضافة او تعديل او حذف في البرنامج (Desktop):



ج_ الزيارات المرضية من خلال البرنامج (exe):

في هذي النافذة يتم عرض الزيارات المرضية من خلال القسم المستشفى وتاريخ تسجيل الحالة والشكوى المرضية والدخول لجلسات المعينة واضافة التعديل او حذف زيارة في السجل الطبي في ال(Desktop) :

11- إعدادات الحساب:

تغير كلمة مرور للمستخدم من خلال عرض اسم المستخدم وتعديل كلمة المرور القديمة واضافة كلمة المرور الجديدة مع التأكد وزر حفظ البيانات:

وهناك العديد من الواجهات لم يتسنى لنا عرضها بسبب كثرة الواجهات وحاولنا عرض معظم الواجهات فقط.

5.3.4 استعلامات النظام (System Queries):

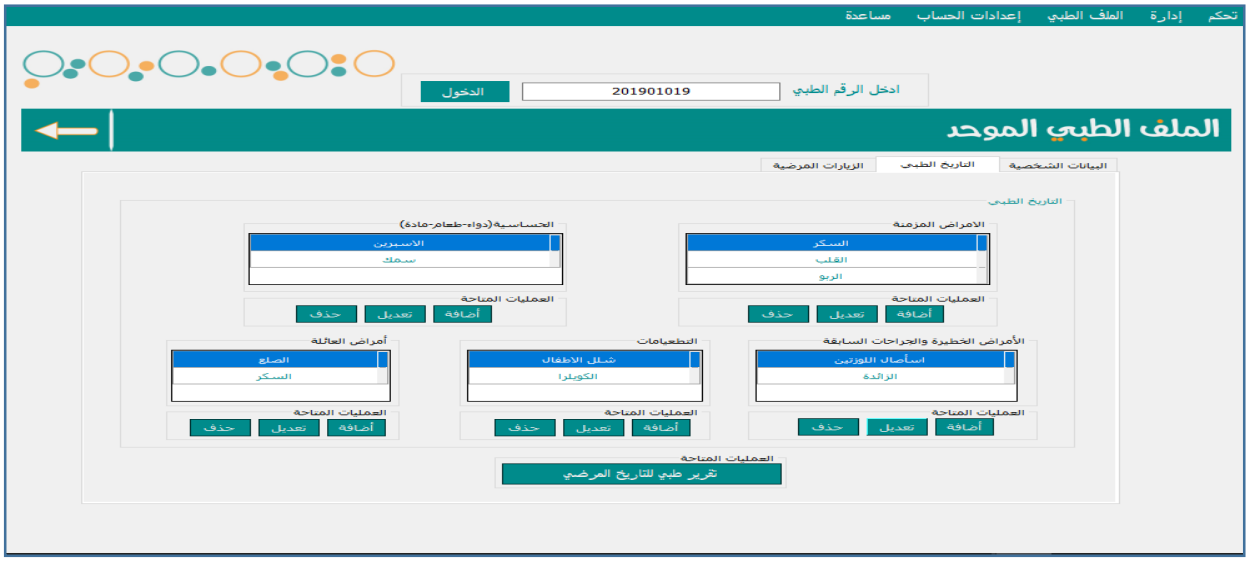
1- شاشة استعلام عن بيانات المريض من خلال الرقم الطبي عبر نسخة الكمبيوتر:

في هذي النافذة يتم عرض استعلامات النظام من خلال الرقم الطبي وإظهار جميع البيانات الطبية للمريض:

رسم توضيحي 29-5 شاشة استعلام عن بيانات المريض من خلال الرقم الطبي عبر نسخة الكمبيوتر

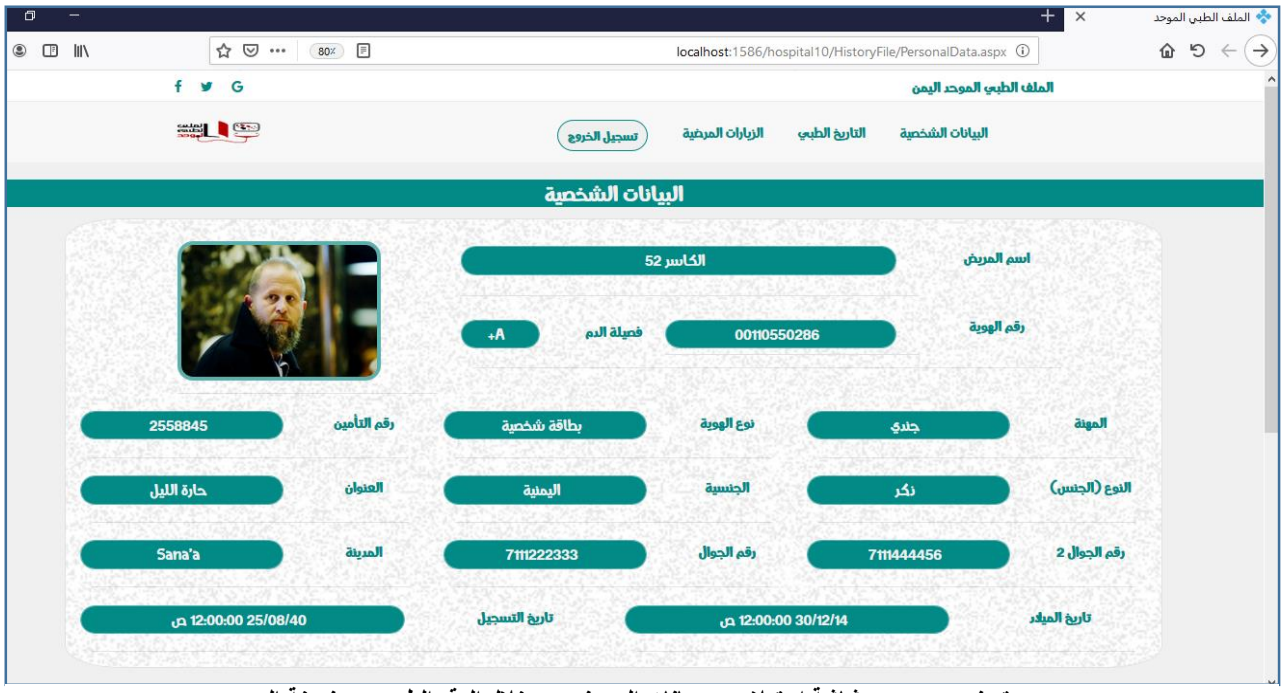
2- شاشة استعلام عن التاريخ الطبي عبر نسخة الكمبيوتر:

في هذي النافذة يتم عرض استعلامات النظام من خلال الرقم الطبي وإظهار جميع البيانات الطبية للمريض:



3- شاشة استعلام عن بيانات المريض من خلال الرقم الطبي عبر نسخة الويب:

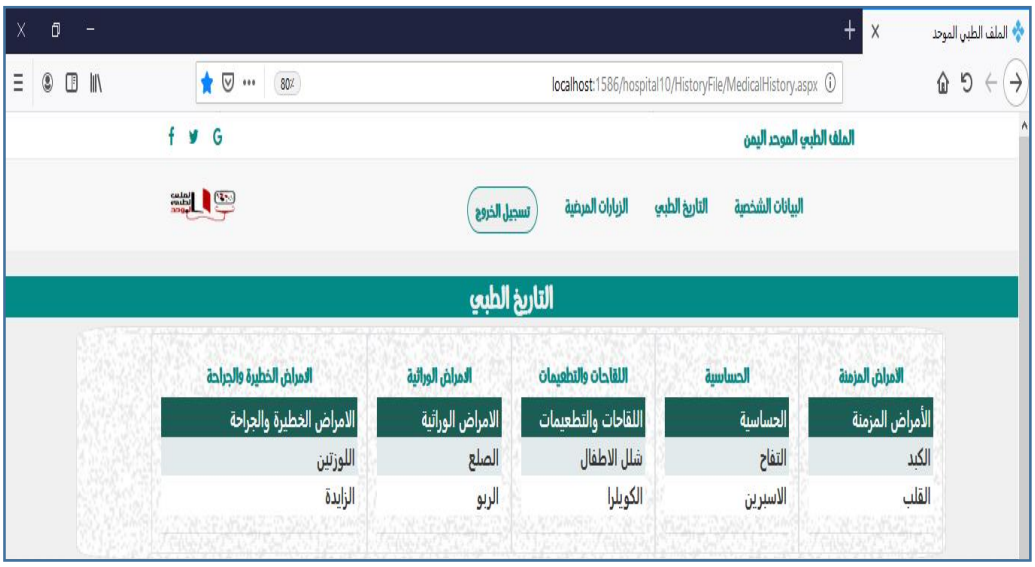
عرض معلومات السجل الطبي من خلال الويب وهي عبارة عن صفحة ويب بعد الدخول بالرقم الطبي للمريض يتم عرض 3 نوافذ عن معلومات التفصيلية للمريض التي تتمثل بالبيانات الشخصية والتاريخ الطبي والزيارات المرضية:



رسم توضيحي 30 -5 شاشة استعلام عن بيانات المريض من خلال الرقم الطبي عبر نسخة الويب

4- شاشة استعلام عن التاريخ الطبي عبر نسخة الويب:

يحتوي على بيانات التاريخ الطبي من خلال عرض الامراض المزمنة والحساسية من الدواء او الطعام او أي مواد ويحتوي على الامراض الخطيرة والجراحات السابقة وإمكانية العرض والطباعة:



5- شاشة استعلام عن بيانات المريض من خلال الرقم الطبي عبر نسخة (Android):

عرض معلومات السجل الطبي من خلال الاندرويد بعد الدخول الى التطبيق والدخول بالرقم الطبي للمريض يتم عرض 3 نوافذ عن معلومات التفصيلية للمريض التي تتمثل بالبيانات الشخصية والتاريخ الطبي والزيارات المرضية.

رسم توضيحي 31-5 شاشة استعلام عن بيانات المريض من خلال الرقم الطبي عبر نسخة (Android)

6- شاشة استعلام عن التاريخ الطبي عبر نسخة (Android):

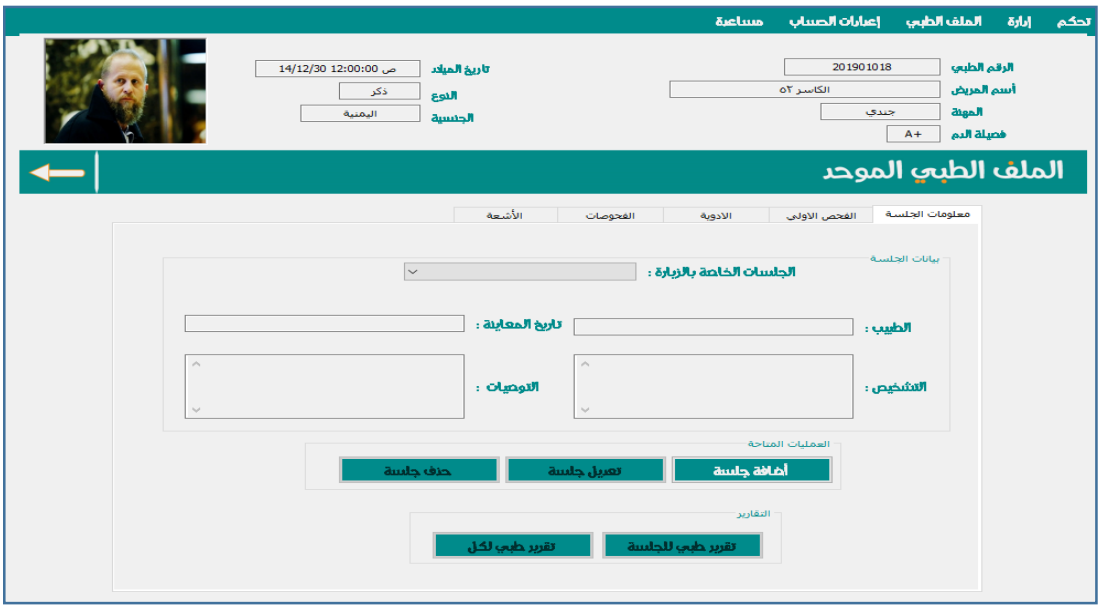
يحتوي على بيانات التاريخ الطبي من خلال عرض الامراض المزمنة والحساسية من الدواء او الطعام او أي مواد ويحتوي على الامراض الخطيرة والجراحات السابقة وإمكانية العرض والطباعة.

7- شاشة استعلام عن الزيارات المرضية عبر نسخة الكمبيوتر:

في هذي النافذة يتم عرض الزيارات المرضية من خلال القسم المستشفى وتاريخ تسجيل الحالة والشكوى المرضية والدخول لجلسات المعينة واضافة او تعديل او حذف زيارة في السجل الطبي في ال(Desktop):

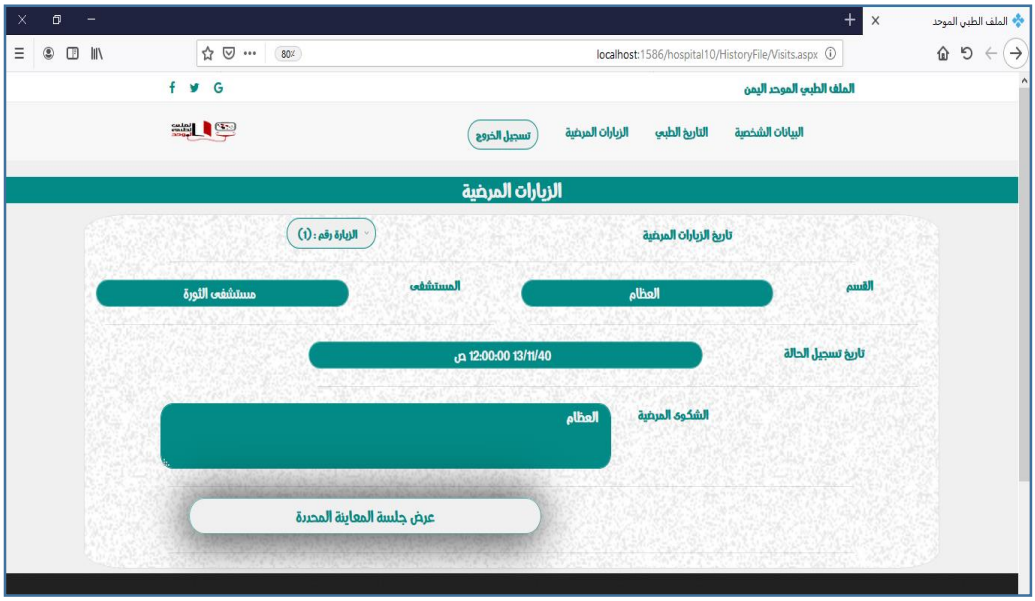
8- شاشة استعلام عن الجلسة المرضية عبر نسخة الكمبيوتر:

في هذي النافذة يتم عرض الجلسات المرضية من خلال معلومات الجلسة والفحص الاولي والأدوية والفحوصات والاشعة في السجل الطبي واضافة المعلومات والتعديل عليها وطباعتها في ال(Desktop):



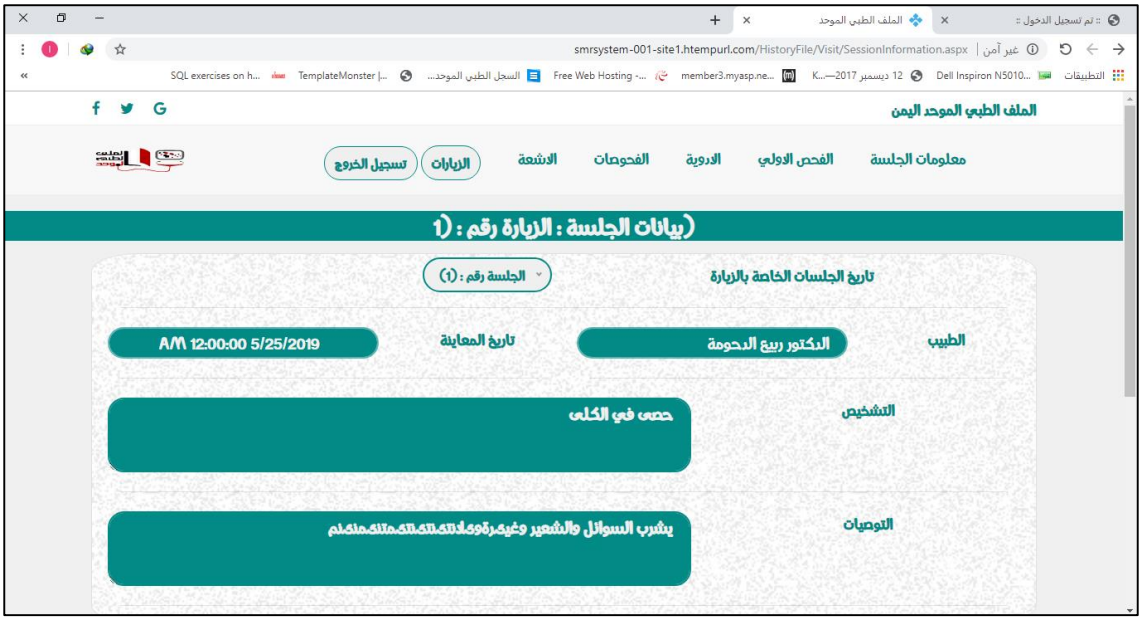
9- شاشة استعلام عن الزيارات المرضية عبر نسخة الويب:

في هذي النافذة يتم عرض الزيارات المرضية من خلال القسم المستشفى وتاريخ تسجيل الحالة والشكوى المرضية والدخول لجلسات المعينة وطباعة التقارير من السجل الطبي في ال (Web):



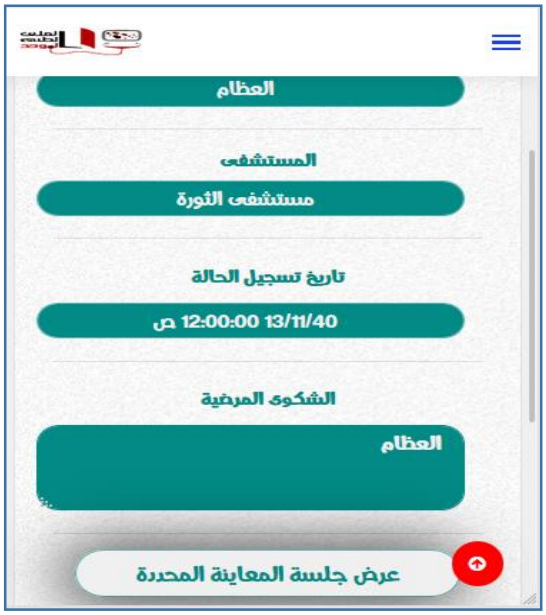
10- شاشة استعلام عن الجلسة المرضية عبر نسخة الويب:

في هذي النافذة يتم عرض الجلسات المرضية من خلال معلومات الجلسة والفحص الاولي والأدوية والفحوصات والاشعة في السجل الطبي وعرض البيانات وطباعتها من السجل الطبي في ال (Web):



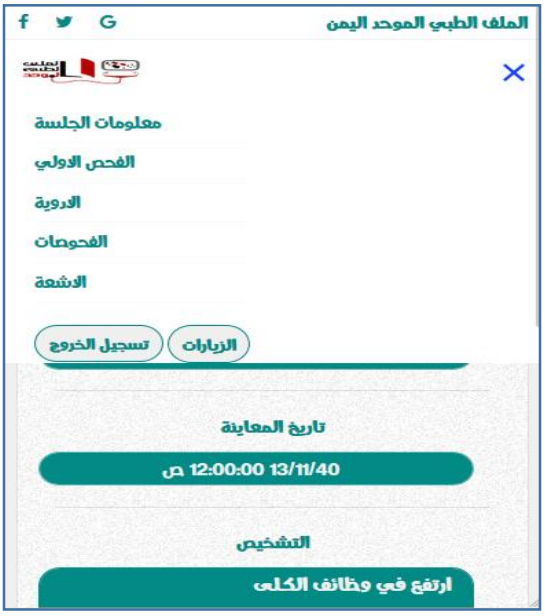
11- شاشة استعلام عن الزيارات المرضية عبر نسخة (Android):

في هذي النافذة يتم عرض الزيارات المرضية من خلال القسم المستشفى وتاريخ تسجيل الحالة والشكوى المرضية والدخول لجلسات المعينة وطباعة التقارير من السجل الطبي من الجوال من خلال تطبيق الاندرويد.



12- شاشة استعلام عن الجلسة المرضية عبر نسخة الويب:

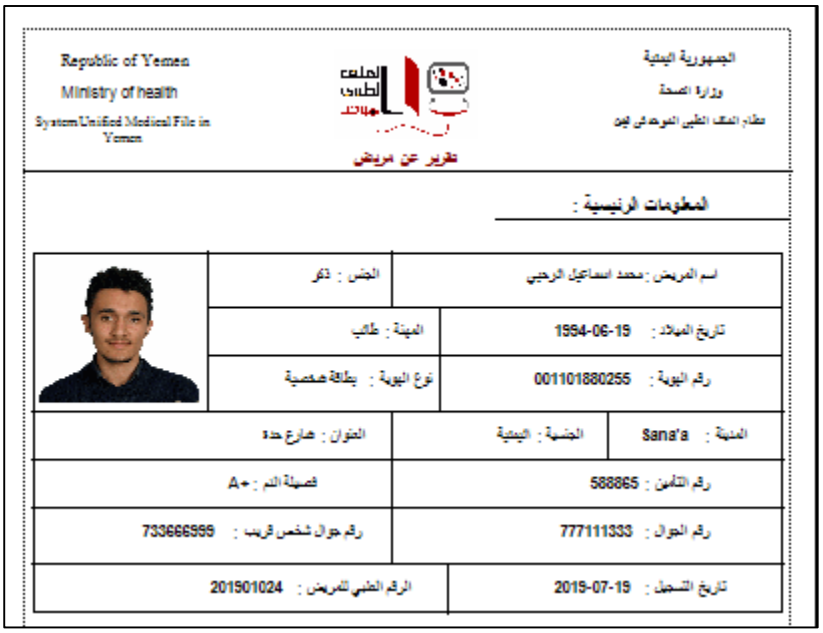
في هذي النافذة يتم عرض الجلسات المرضية من خلال معلومات الجلسة والفحص الاولي والأدوية والفحوصات والاشعة في السجل الطبي وعرض المعلومات وطباعتها في والدخول لجلسات المعينة وطباعة التقارير من السجل الطبي من الجوال من خلال تطبيق الاندرويد.



5.3.5 تقارير النظام (System Reports):

التقارير تبين كيف اعداد العمل وإنجازه وطباعة التقارير تتم مفردة او مجموعة تقارير كلي التقارير خاص:

1- تقرير عن ملف المريض:



Republic of Yemen
Ministry of health
System Unified Medical File in Yemen

الجمهورية اليمنية
وزارة الصحة
نظام الملف الطبي الموحد في اليمن

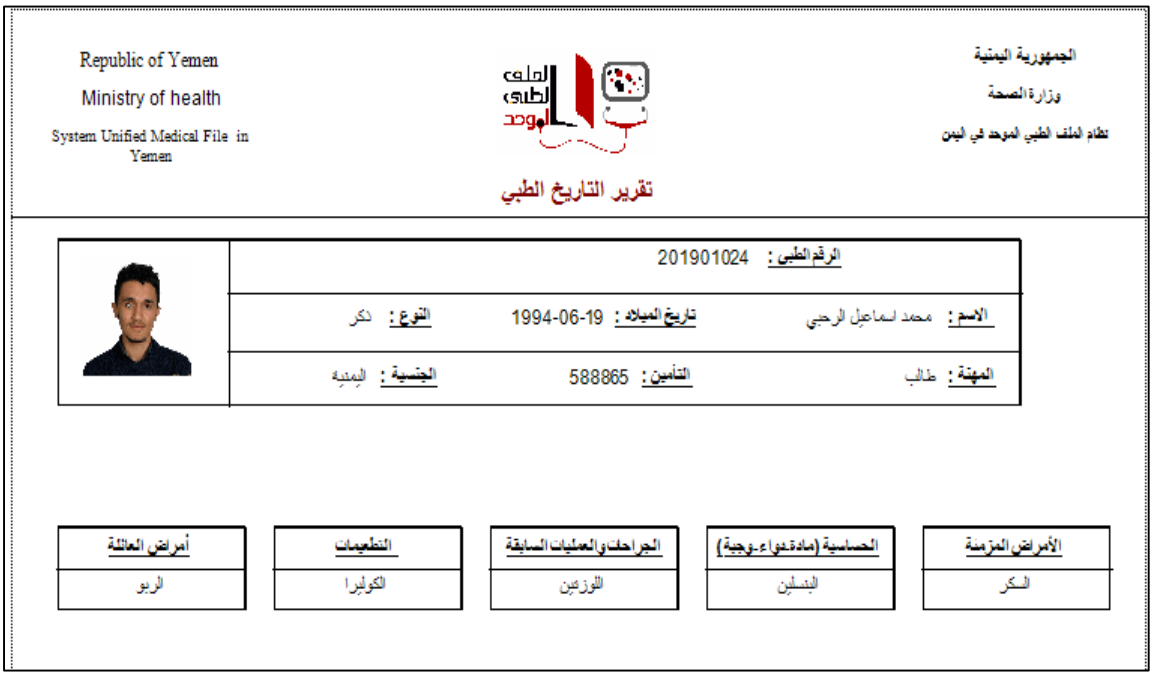
تقرير عن مريض

المعلومات الرئيسية :

اسم المريض : محمد اسماعيل الرجبي	الجنس : ذكر	
تاريخ الميلاد : 1994-06-19	المهنة : طالب	
رقم الهوية : 001101880255	نوع الهوية : بطاقة شخصية	
المدينة : Sana'a	الجنسية : اليمنية	العنوان : شارع حده
رقم التأمين : 588865	فصيلة الدم : A+	
رقم الجوال : 777111333	رقم جوال شخص قريب : 733666999	
تاريخ التسجيل : 2015-07-19	الرقم الطبي للمريض : 201901024	

رسم توضيحي 5-32 تقرير عن ملف المريض

2- تقرير عن التاريخ الطبي للمريض:



Republic of Yemen
Ministry of health
System Unified Medical File in Yemen

الجمهورية اليمنية
وزارة الصحة
نظام الملف الطبي الموحد في اليمن

تقرير التاريخ الطبي

الرقم الطبي : 201901024		
الاسم : محمد اسماعيل الرجبي	تاريخ الميلاد : 1994-06-19	النوع : ذكر
المهنة : طالب	التأمين : 588865	الجنسية : اليمنية

الأمراض المزمنة	الحساسية (مادة-دواء-وجبة)	الجراحات والعمليات السابقة	التطعيمات	أمراض العائلة
السكر	البنسلين	اللوزتين	الكوليرا	الربو

رسم توضيحي 5-33 تقرير عن التاريخ الطبي للمريض

3- تقرير لكل الزيارات المرضية:

الجمهورية اليمنية
وزارة الصحة
نظام الملف الطبي الموحد في اليمن

Republic of Yemen
Ministry of health
System Unified Medical File in Yemen

تقرير لكل الزيارات المرضية

البيانات الشخصية للمريض :

الرقم الطبي : 201901022

الاسم : عبدالله النور | تاريخ الميلاد :16-06-1998 | النوع : ذكر

المهنة : طالب | التأمين : 25552 | الجنسية : اليمنية

معلومات عن الزيارات المرضية :

الزيارة	المستشفى	القسم	تاريخ الزيارة	الشكوى المرضية
(الزيارة رقم : (1	مستشفى الخمسين	مسالك بولية	2019-05-25	الألم في الكلى
(الزيارة رقم : (2	مستشفى بن سينا	العظام	2019-07-19	كسر في الذراع الايمن

رسم توضيحي 34 -5 تقرير لكل الزيارات المرضية

3- تقرير لكل الادوية الخاصة بالجلسة:

الجمهورية اليمنية
وزارة الصحة
نظام الملف الطبي الموحد في اليمن

Republic of Yemen
Ministry of health
System Unified Medical File in Yemen

تقرير الأدوية

البيانات الشخصية للمريض :

الرقم الطبي : 201901024

الاسم : محمد اسماعيل الرجبي | تاريخ الميلاد : 19-06-1994 | النوع : ذكر

فصيلة الدم : A+ | الجنسية : اليمنية

معلومات عن الادوية :

الجلسة المرضية : (الجلسة رقم : (1 | تاريخ المعاينة : 2019-07-19

المستشفى : مستشفى المتوكل | القسم : مسالك بولية | الطبيب : الدكتور ربيع الدحومه

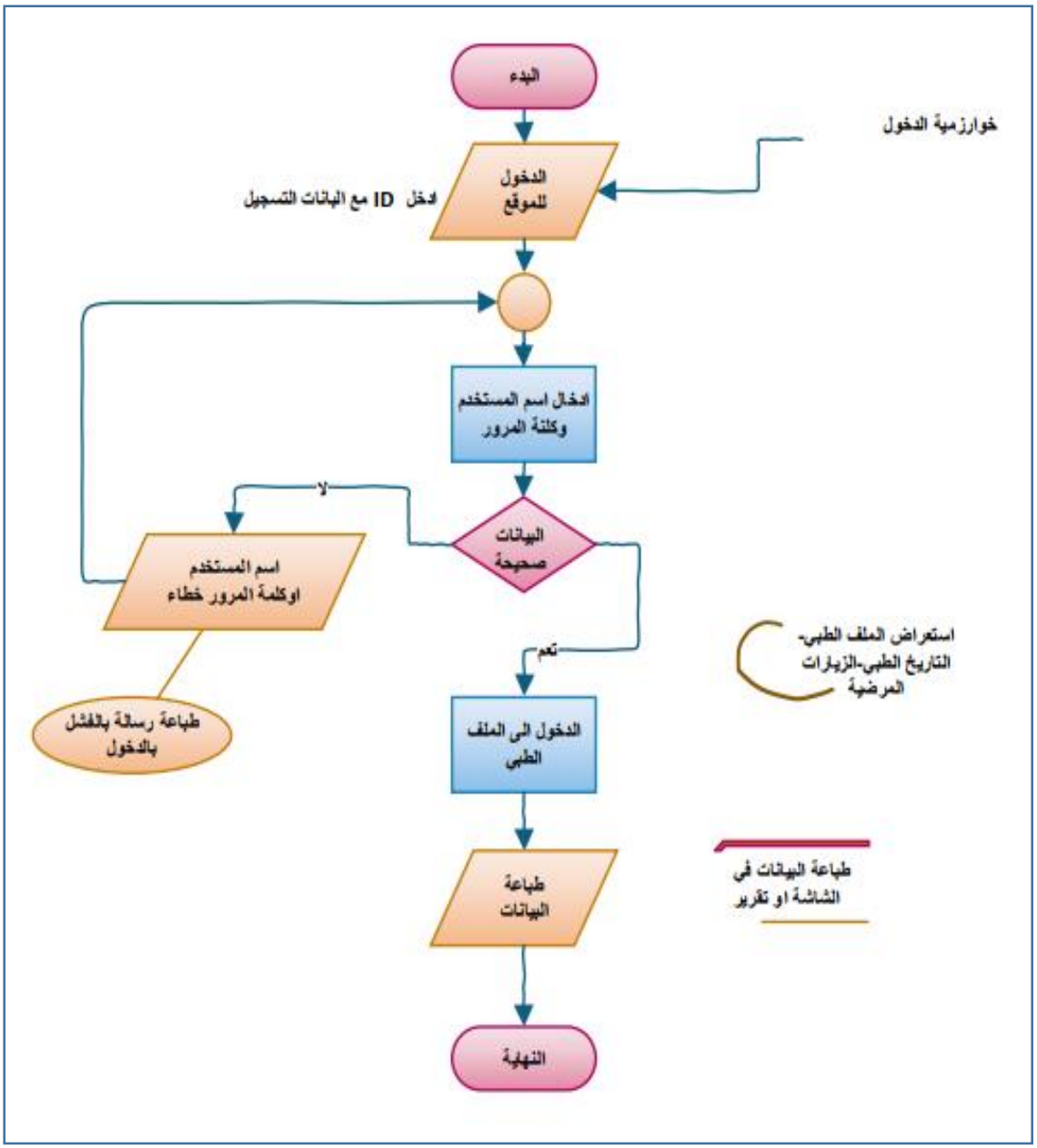
أسم الدواء	أستخدام الدواء	فترة أستخدام الدواء	تاريخ تسجيل الدواء
اسبرين	بعد الاستيقاظ من النوم	يوم 30	2019-07-19
sksnpo	عند الألم	شهرين	2019-07-19

رسم توضيحي 35 -5 تقرير لكل الادوية الخاصة بالجلسة

الفصل السادس

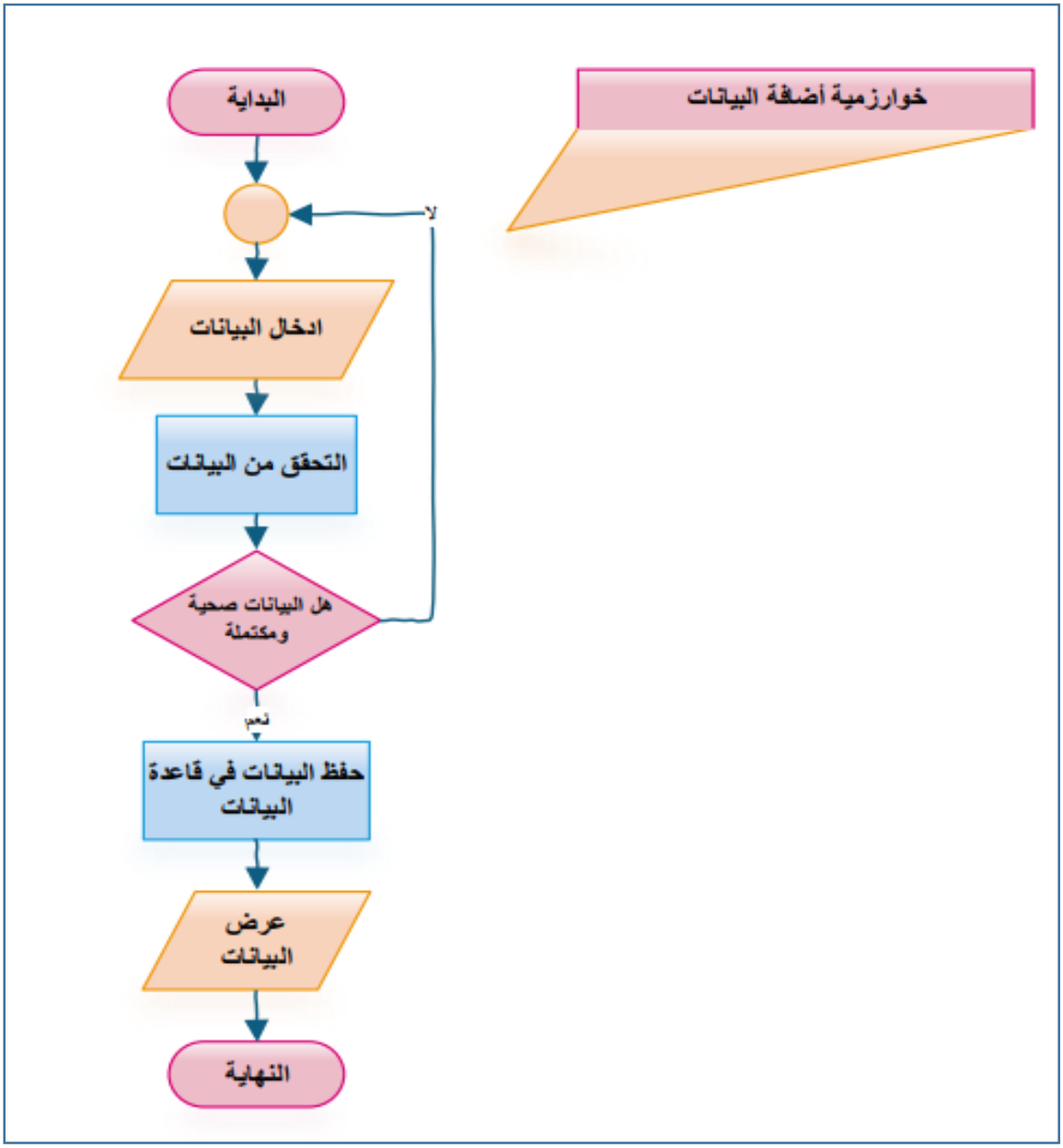
تصميم مخططات انسيابية للنظام والخوارزميات

6.1 خوارزمية الدخول للملف الطبي (عبر الويب او الإصدار المكتبي او الاندرويد):



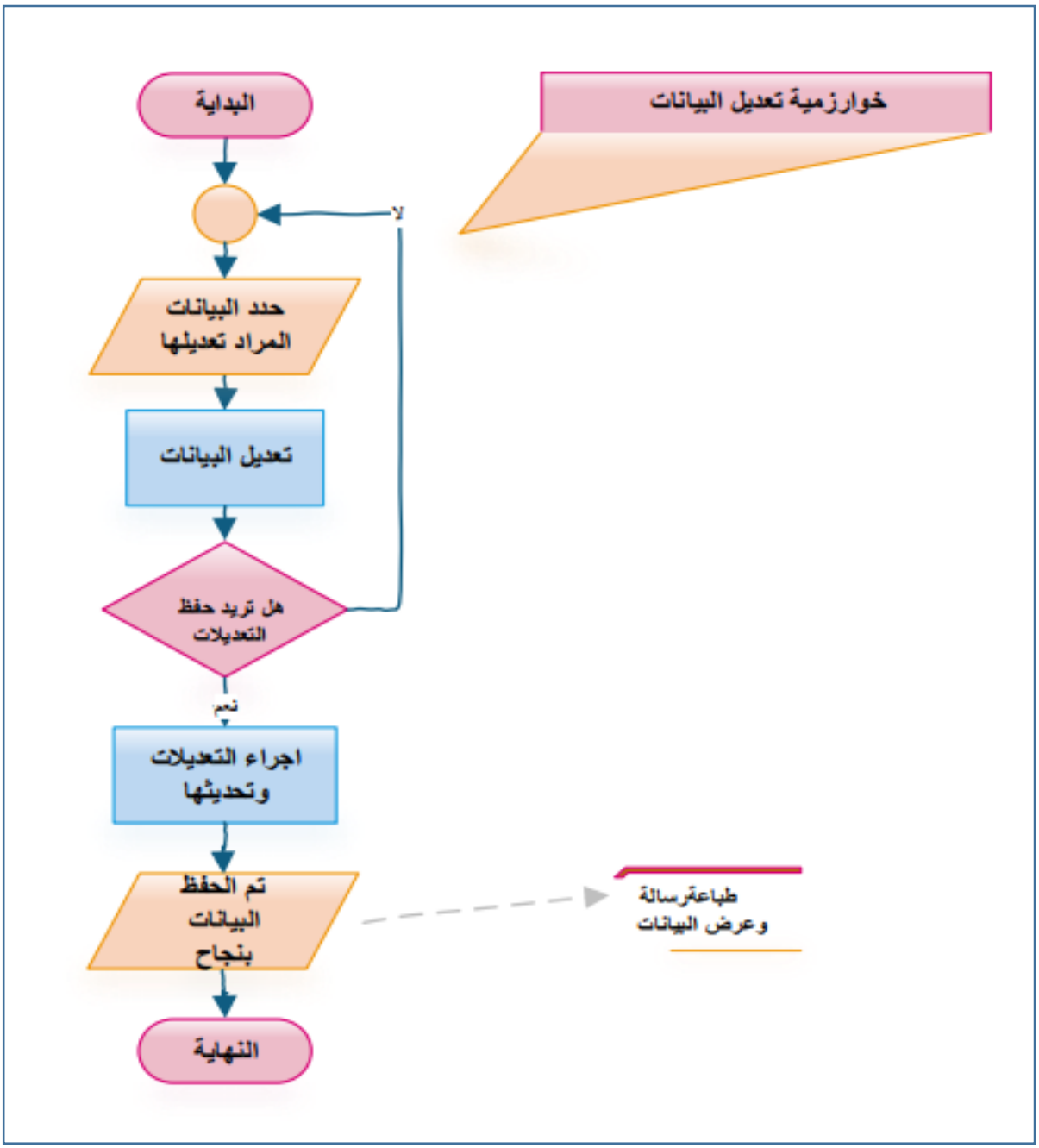
رسم توضيحي 36 -6 خوارزمية الدخول للملف الطبي (عبر الويب او الإصدار المكتبي او الاندرويد)

6.2 خوارزمية إضافة بيانات للملف الطبي (عبر الإصدار المكتبي):



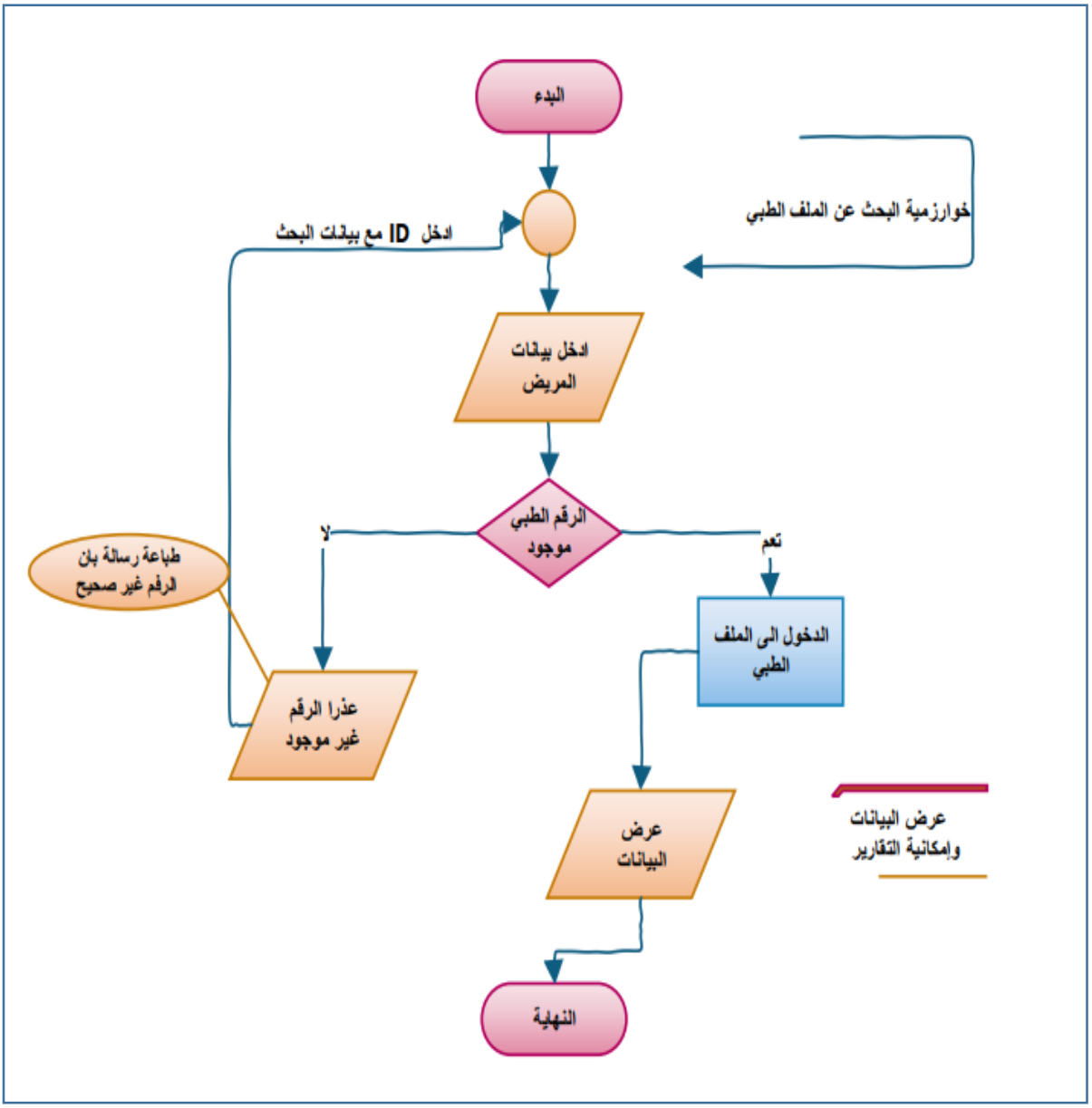
رسم توضيحي 37-6 خوارزمية إضافة بيانات للملف الطبي (عبر الإصدار المكتبي)

6.3 خوارزمية تعديل البيانات للملف الطبي (عبر الإصدار المكتبي):



رسم توضيحي 38-6 خوارزمية تعديل البيانات للملف الطبي (عبر الإصدار المكتبي)

6.4 خوارزمية البحث عن الرقم للملف الطبي واستعراض البيانات:



رسم توضيحي 39-6 خوارزمية البحث عن الرقم للملف الطبي واستعراض البيانات

الفصل السابع

مرحلة تنفيذ النظام

7.1 تنفيذ النظام: -

وفي هذه المرحلة بعد أن يتم بناء قواعد البيانات وكتابة البرنامج وتطبيق النظام فعليا واختباره يتم توضيح هذه النقاط في هذا الفصل بالإضافة لوضع النقاط الأساسية لتشغيل النظام وكيفية التعامل معه.

من خلال تثبيت النظام وفتح النظام يظهر لنا واجهة تسجيل الدخول ومن خلال نوعية اسم المستخدم وكلمة المرور يتم تحديد المستخدم صلاحيات المستخدم إذا كان طبيب او مختبر او طبيب او مدير وكل ونوع يكون له صلاحيات محددة حسب تخصصه فمثلا المريض صلاحيته طبع بيانات الملف الطبي الخاص به ولا يسمح له بالتعديل او إضافة أي بيانات أخرى.

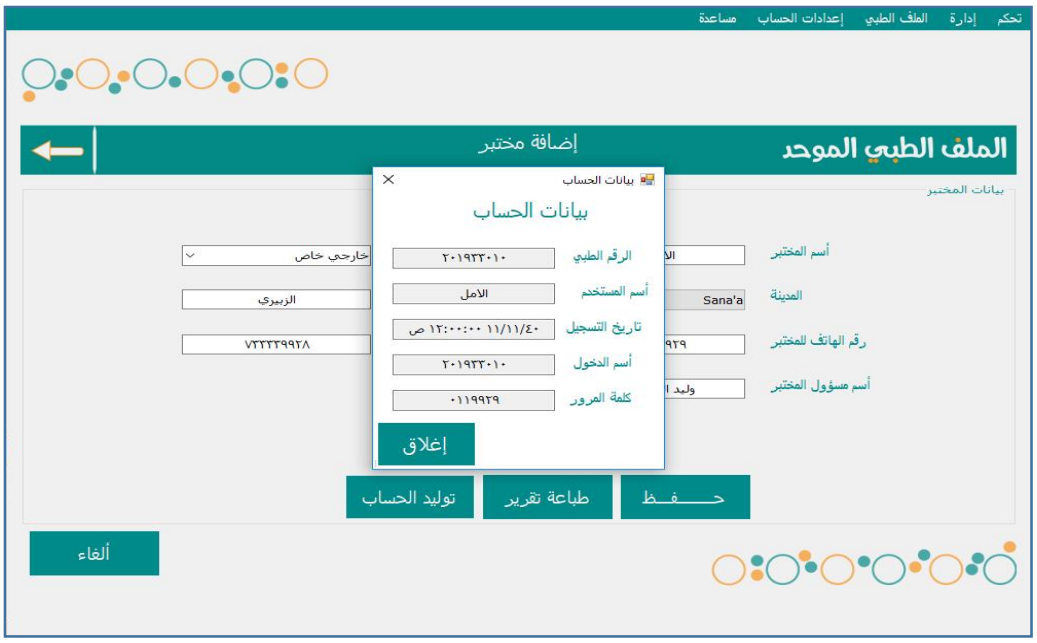
7.2 اختبار النظام: -

نذكر في هذا القسم عملية الاختبار للنظام سواء كان اختبار الوحدة او اختبار التكامل او اختبار القبول:

1- اختبار عملية الإضافة والحفظ للمستشفى:

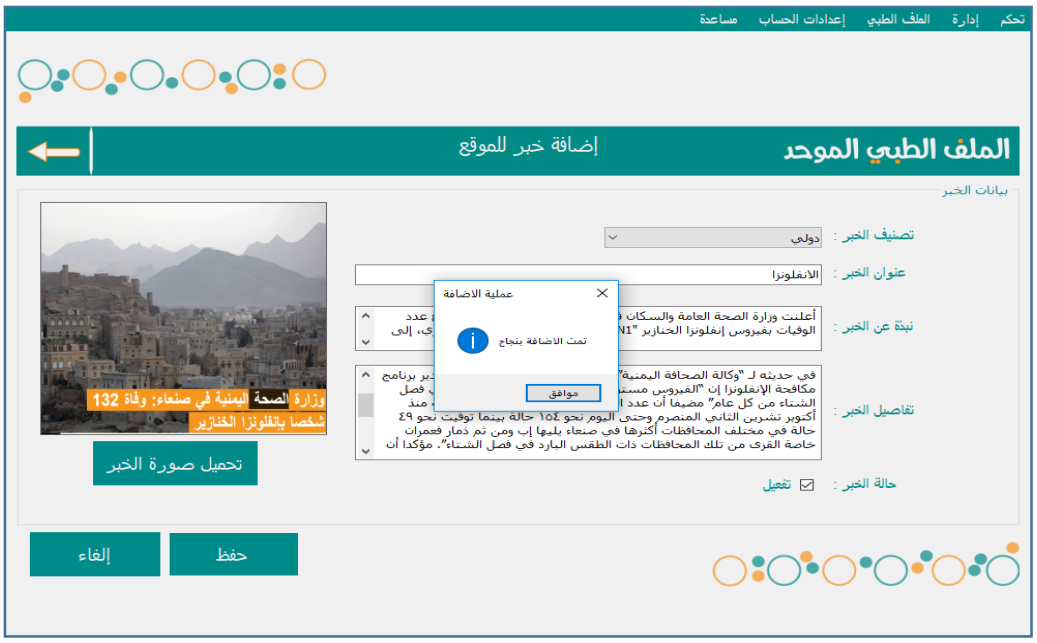
رسم توضيحي 40-7 شاشة اختبار عملية الإضافة والحفظ للمستشفى

2- اختبار عملية توليد الحساب بعد عملية الحفظ للمختبرات:



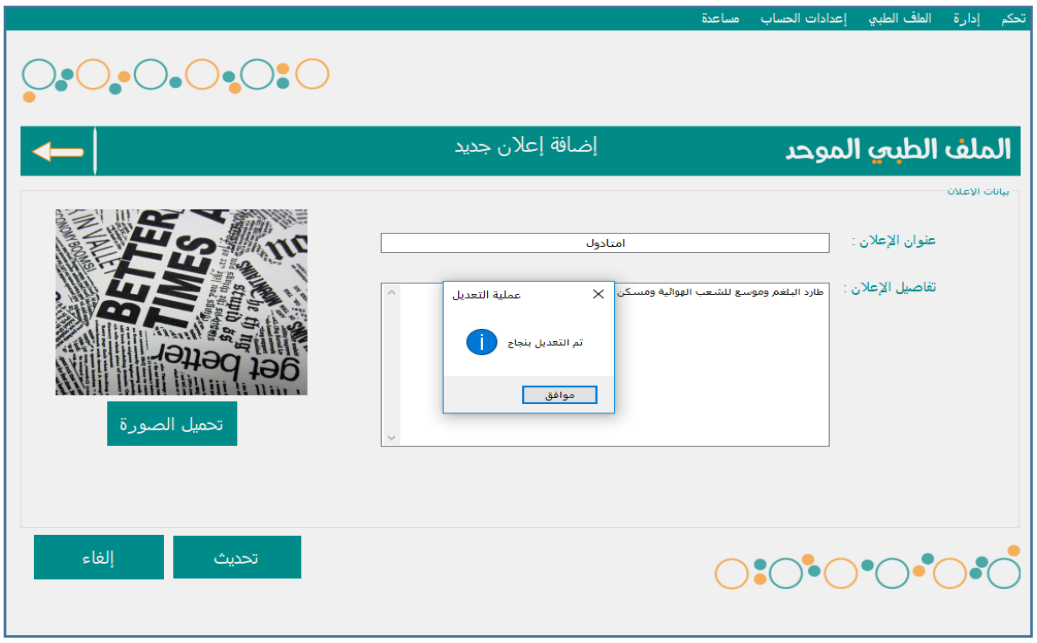
رسم توضيحي 41-7 شاشة اختبار عملية توليد الحساب بعد عملية الحفظ للمختبرات

3- اختبار عملية الإضافة والحفظ للأخبار على الموقع:



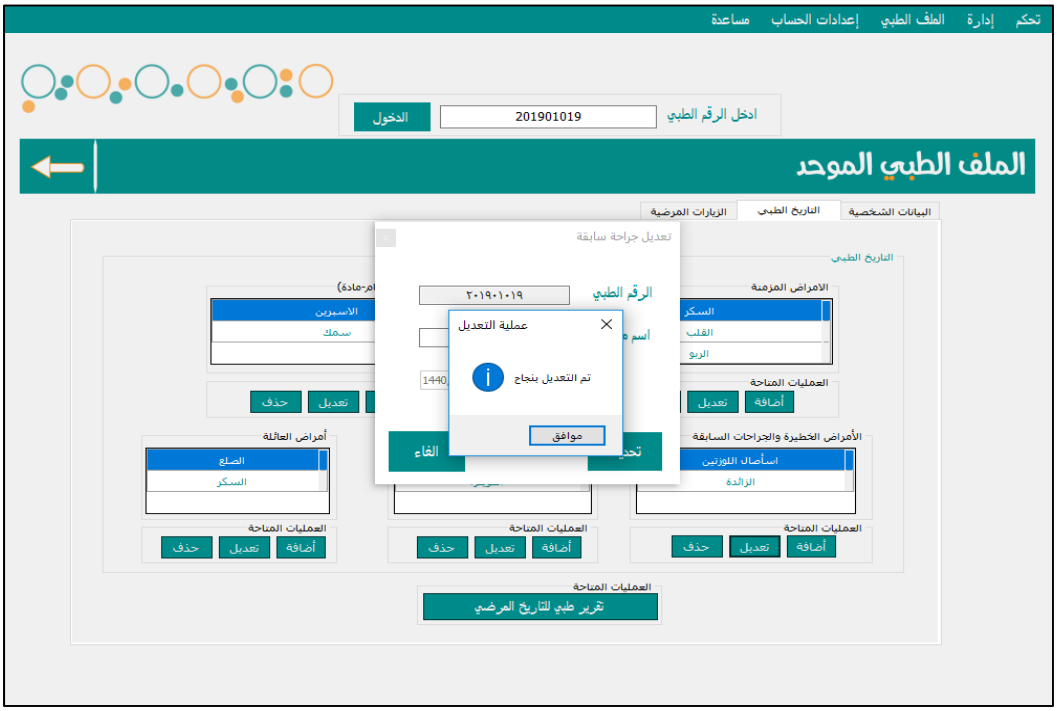
رسم توضيحي 42-7 شاشة اختبار عملية الإضافة والحفظ للأخبار على الموقع

4- اختبار عملية التعديل والحفظ للإعلانات:



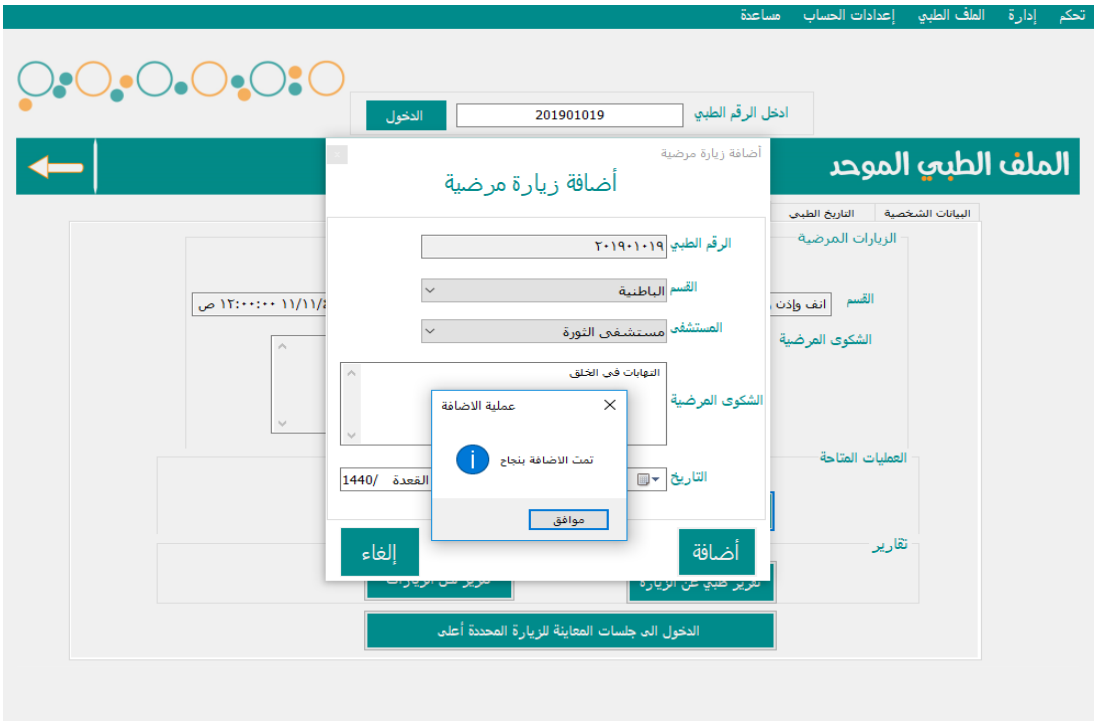
رسم توضيحي 43-7 شاشة اختبار عملية التعديل والحفظ للإعلانات

5- اختبار عملية التعديل والحفظ للتاريخ المرضي:



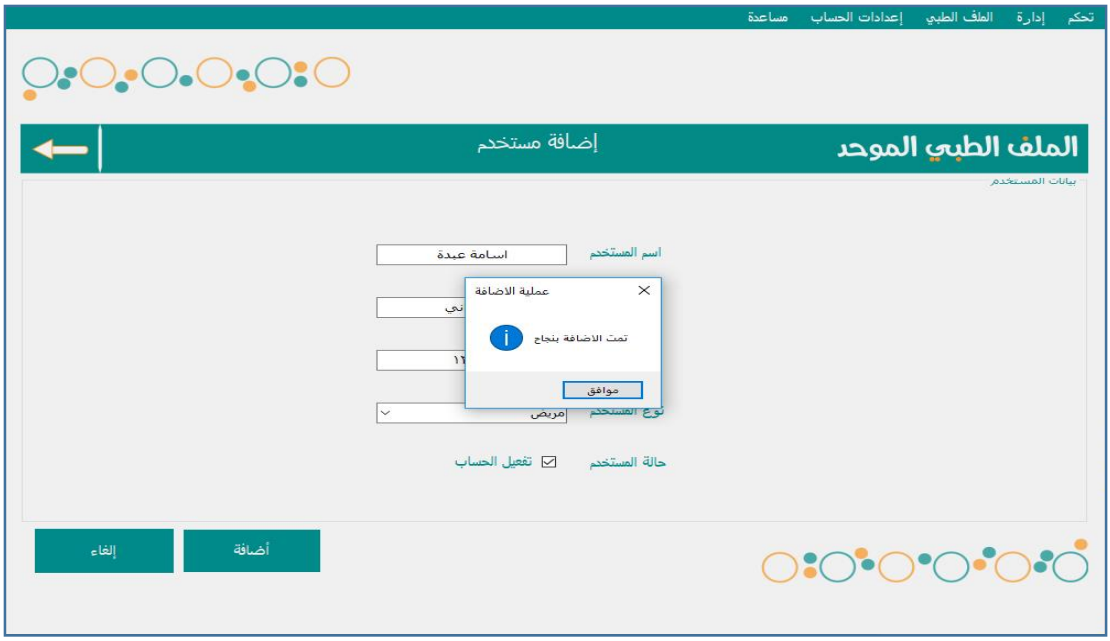
رسم توضيحي 44-7 شاشة اختبار عملية التعديل والحفظ للتاريخ المرضي

6- اختبار عملية التعديل والحفظ للزيارة المرضية:



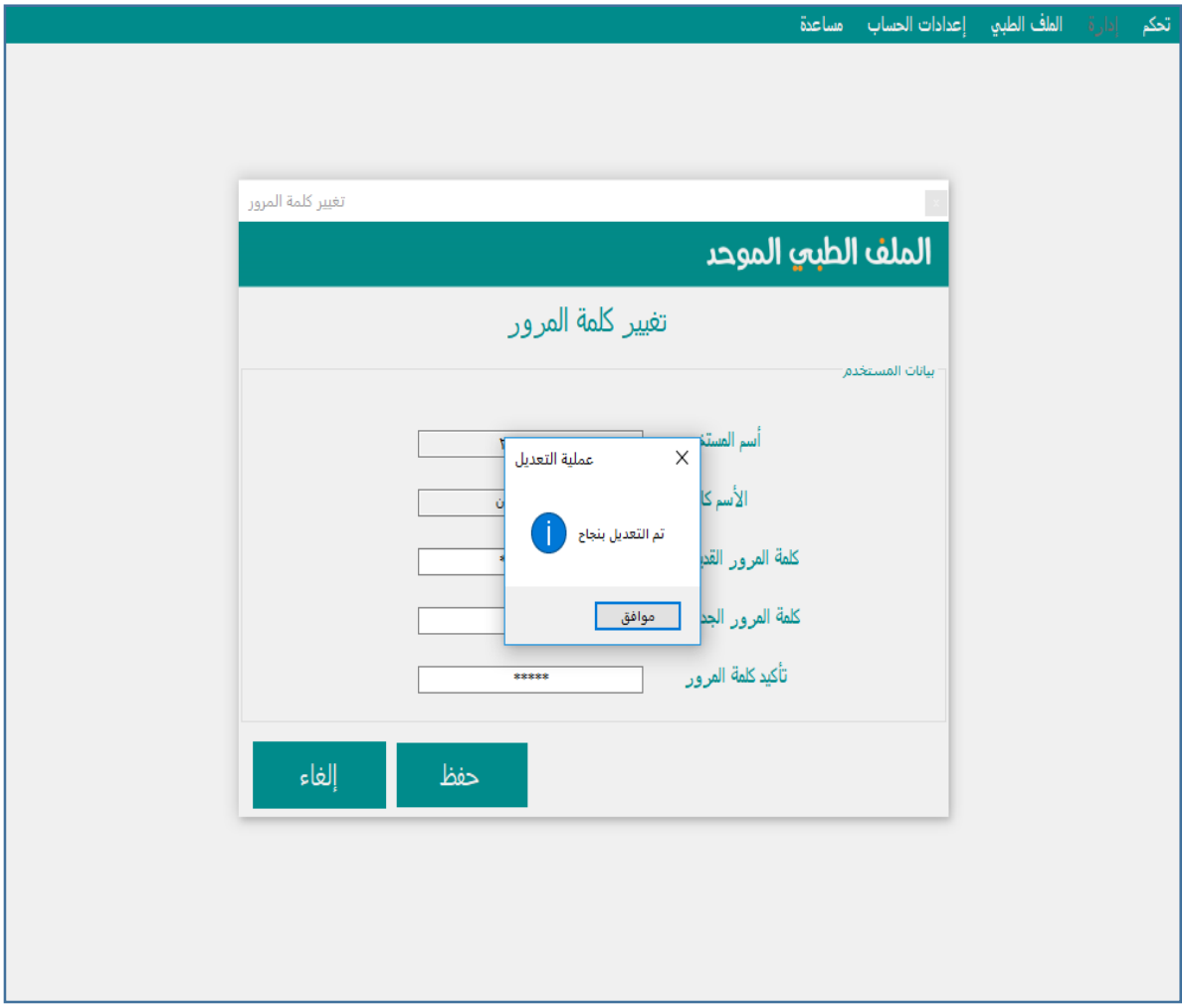
رسم توضيحي 45-7 شاشة اختبار عملية التعديل والحفظ للزيارة المرضية

7- اختبار عملية الإضافة والحفظ للمستخدمين:



رسم توضيحي 46-7 شاشة اختبار عملية الإضافة والحفظ للمستخدمين

8- اختبار عملية التعديل والتغير لكلمة المرور للمستخدمين:



رسم توضيحي 47-7 شاشة اختبار عملية التعديل والتغير لكلمة المرور للمستخدمين

الفصل الـثـامـــن

الخاتمة

8.1 الخلاصة:

توحيد السجل الطبي وتوفير قاعدة بيانات مركزية في أكثر من مستشفى ومركز صح توفير رقم طبي موحد وموثوق للمريض في جميع المستشفيات والمراكز الصحية سهولة الوصول للأطباء والمختبرات وإضافة التعديلات لملف المريض بسهولة توفير المعلومات عن حالة المريض والأدوية المستخدمة ونتائج الفحوصات السابقة وذلك لتحسين اتخاذ القرار المناسب من قبل الطبيب.

القضاء على العمل اليدوي الذي كان سابقا عبر الارشفة اليدوية اصدار التقارير المطلوبة بحيث يمكننا بواسطة النظام الاطلاع على أي بيانات تخص بيانات الامراض والدكاترة والمستشفيات والمختبرات والمستخدمين أيضا عند طلب استفسار عن أي بيانات من السابقين ذكرهم يمكننا الاطلاع على معلوماته في وقت قياسي وبأقل جهد مقارنه بالنظام المستقل او اليدوي والبحث في السجلات.

تم تنظيم الملف الطبي بالبيانات الكاملة للمريض والاحتفاظ بها في قاعدة البيانات بحيث يمكننا الوصول للبيانات من أي مكان وسلامة البيانات حيث كانت البيانات معرضة للفقدان والضياع في النظام المستقل او اليدوي.

8.1 الاستنتاجات:

بعد الانتهاء من توثيق المشروع، نختم في هذا الفصل الصعوبات التي واجهها فريق العمل لهذا النظام والخبرات المكتسبة اثناء العمل علية، وأيضا تذكر الخطط والعمل المستقبلي لتطوير النظام.

8.1.1 الصعوبات:

من الصعوبات التي واجهتنا:

1- صعوبة التعامل مع التوقيت (Date time) وذلك بسبب اختلاف مواقيت النظام مع المواقيت الزمنية العالمية إيجاد صعوبة في المراجع كون النظام لم يطبق من قبل وانما أفكار أي لم يكن هناك نظام لأخذ المدخلات او المشكال واصلاحها وتم استخلاص المشاكل من خلال المحاكاة وإدخال البيانات المحاكية للواقع.

2- صعوبة جعل النظام يتأقلم مع جميع المستشفيات بسبب استقلال البيانات في كل إدارة على حدة.

3- انانية بعض المدراء في تطبيق النظام لما يتيح ويوضح معلومات المريض مما يسمح للجميع بمشاركة المعلومات وتقليل تقيد المعلومة او احتكارها.

8.1.2 الخبرات المكتسبة:

- تطوير فريق العمل بالتعامل مع قواعد البيانات ال (SQL Server) للاستعلام عن كل البيانات واستخراج البيانات المطلوبة.
- تطوير فريق المشروع بالتعامل مع اللغات المختلفة من (ASP .Net & Android & C#) والابداع فيها.
- تحليل النظام بشكل سلس يسهل على الفريق استخراج وإنتاج نظام يعمل بالشكل المطلوب.
- كيف مواجهة المشاكل والعمل بروح الجماعية الواحدة وإدارة الوقت بالشكل الصحيح وتقسيم العمل بين افراد الفريق.
- التخطيط للعمل المستقبلي.

نستطيع نقول ان نظامنا الحالي انجز اغلب الاعمال الإدارية وانشاء الملف الطبي والتنظيم في التاريخ الطبي وكيفية الحفاظ على البيانات ومساعدة الاطباء ذوي الاختصاص في القرارات المبنية على معرفة مسبقة وبإذن الله عز وجل في المستقبل القريب سيتم تطوير النظام ليشمل الاتي:

- اخراج احصائيات لوزارة الصحة بعدد الامراض المنتشرة ومكافحة الاوبة والحد من انتشارها.
- عمل توعية شاملة بأهمية الملف الطبي وتدريب الكوادر لاستخدام النظام.
- يكون هذا العمل أساس لتطوير نظام الصحة في اليمن مما يحد من المعانة والحد من الاوبة وتوحيد العمل في المؤسسات الحكومية والخاص مما يساهم في التعاون ومشاركة المعلومة والتنافس النزيه في ضوابط واطارا الدولة الحديثة.

8.1 المقترحات والتوصيات:

8.1.1 المقترحات:

على ضوء النتائج المحصلة من النظام الالي وبعد مناقشة النتائج تبين ان هذا التحليل قد حقق معرفة بعض الحلول استنادا الى دقة النتائج حيث تم التواصل الى الاستنتاجات الاتية:

- للأنترنت في الملف الطبي أربع خدمات أساسية وهي:

1. الوصول الدائم والفعال مع معلومات المريض بسهولة وباي مكان عن بعد ومشاركة المعلومة من خلال التعاون بإدخال البيانات من أكثر مصدر.
2. تكامل وسلامة البيانات من خلال دقة المعلومة وصحتها في قاعدة البيانات.
3. الحفاظ على البيانات من التلف من خلال حفظها في سيرفر قاعدة البيانات.
4. تنبه الجهات المختصة بالإحصائيات الأوبئة مما يسمح بوضع منهجية للحفاظ على صحة المجتمع.

8.1.1 التوصيات:

1. نشر ثقافة تقنية المعلومة داخل البيئة الصحية
2. تحويل الأوراق الملموسة لمعلومات المريض الى بيانات رقمية
3. توفير البيئة التقنية التحتية التي تساعد في بناء الصحة الالكترونية
4. تصميم وبناء برامج التي تساعد طلاب الطب في محاكة النظام وأهميته للمريض والطبيب على حد سواء من يريد تطوير النظام وعمل بيانات شاملة للمريض وربط المعلومات مع وزارة الصحة للاتخاذ خطط التنمية الشاملة يكون هذا النظام حجر أساس وبداية لتكملة مشوار التطوير والرقي بالمجتمع اليمني.

8.1 الخاتمة:

لابد لنا ونحن نخطو خطواتنا الأخيرة في الحياة الجامعية (مرحلة البكالوريوس) من وقفة نعود إلى أعوام قضيناها في رحاب الجامعة مع أساتذتنا الكرام الذين قدموا لنا الكثير باذلين بذلك جهودا كبيرة في بناء جيل الغد لتبعث الأمة من جديد من قبل أن نمضي نقدم أسمى آيات الشكر والامتنان والتقدير والمحبة إلى الذين حملوا أقدس رسالة في الحياة إلى الذين مهدوا لنا طريق العلم والمعرفة إلى جميع أساتذتنا الأفاضل.

المراجع (References)

وزارة الصحة – اليمن – صنعاء

مستشفى عبدالقادر المتوكل – صنعاء – شارع بغداد

مستشفى السلام الدولي – صنعاء – شارع الجزائر

مستشفى السعودي الالماني – صنعاء – الجراف

مستشفى السبعين للأمومة والطفولة – صنعاء – السبعين

محمد القصيمي (2012)،" نظام السجل الطبي الالكتروني"، جامعة الجنان، طرابلس، لبنان.

محمد قندوز (2013)، " إدارة المستشفيات – دراسة ماجستير "، جامعة الجزائر، الجزائر.

منظمة الصحة العالمية (2005)، " دليل العاملين في السجلات الطبية في البلدان النامية "، المكتب الإقليمي لشرق المتوسط.

مستشفى الملك فيصل التخصصي (2016)، " بوابة الخدمات الإلكترونية للمرضى (صحتي) "، المملكة العربية السعودية.

موقع: eservices.kfshrc.edu.sa "بوابة الخدمة الإلكترونية للمرضى (صحتي)"

موقع: https://ar.wikipedia.org/w/index.php?title= إلكتروني_طبي_سجل &oldid=27748233

موقع: www.moh.gov.om " البوابة الصحية الإلكترونية – وزارة الصحة – عُمان".

موقع: www.doctory-eg.com "إدارة العيادات الطبية"،

موقع: www.al-eman.com .

موقع www.myasp.net .

قناة المبدع والدكتور خالد السعداني – Khalid ESSAADANI على اليوتيوب على

هذا الرابط: https://www.youtube.com/channel/UCjKpb_hadZ3BQkHcTy81MRA

الملاحق:

- **كود النظام:**

  - كلاس الاتصال بقاعدة البيانات:

```
namespace SMR_system_of_yemen.DAL
{
    class DataAccessLayar
    {

        SqlConnection sqlconnection;

        //الاتصال الاوبجيكت باستنساخ يقوم
        public DataAccessLayar()
        {
            sqlconnection = new SqlConnection(@"Data Source=.;Initial
Catalog=DB_Medical_File;Integrated Security=True");
        }

        //البيانات بقاعدة الاتصال فتح أجراء

        public void Open()
        {
            if (sqlconnection.State != ConnectionState.Open)
            {
                sqlconnection.Open();
            }
        }

        //البيانات بقاعدة الاتصال بإغلاق الخاص الاجراء
        public void Close()
        {
            if (sqlconnection.State == ConnectionState.Open)
                sqlconnection.Close();
        }

        //البيانات قاعدة من البيانات قراءة أجراء

        public DataTable SelectData(string Stored_procedure,
SqlParameter[] param)
        {
            SqlCommand sqlcmd = new SqlCommand();
            sqlcmd.CommandType = CommandType.StoredProcedure;
            sqlcmd.CommandText = Stored_procedure;
            sqlcmd.Connection = sqlconnection;

            if (param != null)
            {
```

```
                for (int i = 0; i < param.Length; i++)
                {
                    sqlcmd.Parameters.Add(param[i]);
                }
            }
            SqlDataAdapter da = new SqlDataAdapter(sqlcmd);
            DataTable dt = new DataTable();
            da.Fill(dt);
            return dt;
        }

        //البيانات قاعدة الى والحذف والتعديل الاضافة أجراء
        public int ExecuteCommand(string Stored_procedure, SqlParameter[]
    param)
        {
            int r = 0;
            SqlCommand sqlcmd = new SqlCommand();
            sqlcmd.CommandType = CommandType.StoredProcedure;
            sqlcmd.CommandText = Stored_procedure;
            sqlcmd.Connection = sqlconnection;

            if (param != null)
            {
                sqlcmd.Parameters.AddRange(param);
            }

            r = sqlcmd.ExecuteNonQuery();

            return r;
        }




    }
}
```

- **كلاس تنفيذ أوامر الإضافة والحذف والتعديل:**

```
namespace SMR_system_of_yemen.BL
{
    class CLS_PATIENTS
    {
        public int ADD_PATIENT(string name,bool gender,DateTime birth,string
work,string type_identity, string num_identity
            ,int city,string address,string phone,string phone_neer,int nat,string
insurance,int blood,byte[] img)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
            SqlParameter[] param = new SqlParameter[14];

            param[0] = new SqlParameter("@p_name", SqlDbType.NVarChar, 30);
            param[0].Value = name;

            param[1] = new SqlParameter("@p_gender", SqlDbType.Bit);
            param[1].Value = gender;

            param[2] = new SqlParameter("@p_birth", SqlDbType.Date);
            param[2].Value = birth;

            param[3] = new SqlParameter("@p_work", SqlDbType.NVarChar, 20);
            param[3].Value = work;

            param[4] = new SqlParameter("@p_type_Identity", SqlDbType.NVarChar, 15);
            param[4].Value = type_identity;

            param[5] = new SqlParameter("@p_num_Identity", SqlDbType.NVarChar, 15);
            param[5].Value = num_identity;

            param[6] = new SqlParameter("@p_city", SqlDbType.Int);
            param[6].Value = city;

            param[7] = new SqlParameter("@p_address", SqlDbType.NVarChar, 30);
            param[7].Value = address;

            param[8] = new SqlParameter("@p_phone", SqlDbType.NVarChar, 15);
            param[8].Value = phone;

            param[9] = new SqlParameter("@p_phone_neer", SqlDbType.NVarChar, 15);
            param[9].Value = phone_neer;

            param[10] = new SqlParameter("@p_nat", SqlDbType.Int);
            param[10].Value = nat;

            param[11] = new SqlParameter("@p_insurance", SqlDbType.NVarChar, 30);
            param[11].Value = insurance;

            param[12] = new SqlParameter("@p_blood_type", SqlDbType.Int);
            param[12].Value = blood;

            param[13] = new SqlParameter("@p_image", SqlDbType.Image);
            param[13].Value = img;
```

```
            r = DAL.ExecuteCommand("ADD_PATIENT", param);
            DAL.Close();
            return r;

        }

        public int UPDATE_PATIENT(string name, bool gender, DateTime birth, string
work, string type_identity, string num_identity
            , int city, string address, string phone, string phone_neer, int nat,
string insurance, int blood, byte[] img,string ID)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
            SqlParameter[] param = new SqlParameter[15];

            param[0] = new SqlParameter("@p_name", SqlDbType.NVarChar, 30);
            param[0].Value = name;

            param[1] = new SqlParameter("@p_gender", SqlDbType.Bit);
            param[1].Value = gender;

            param[2] = new SqlParameter("@p_birth", SqlDbType.Date);
            param[2].Value = birth;

            param[3] = new SqlParameter("@p_work", SqlDbType.NVarChar, 20);
            param[3].Value = work;

            param[4] = new SqlParameter("@p_type_Identity", SqlDbType.NVarChar, 15);
            param[4].Value = type_identity;

            param[5] = new SqlParameter("@p_num_Identity", SqlDbType.NVarChar, 15);
            param[5].Value = num_identity;

            param[6] = new SqlParameter("@p_city", SqlDbType.Int);
            param[6].Value = city;

            param[7] = new SqlParameter("@p_address", SqlDbType.NVarChar, 30);
            param[7].Value = address;

            param[8] = new SqlParameter("@p_phone", SqlDbType.NVarChar, 15);
            param[8].Value = phone;

            param[9] = new SqlParameter("@p_phone_neer", SqlDbType.NVarChar, 15);
            param[9].Value = phone_neer;

            param[10] = new SqlParameter("@p_nat", SqlDbType.Int);
            param[10].Value = nat;

            param[11] = new SqlParameter("@p_insurance", SqlDbType.NVarChar, 30);
            param[11].Value = insurance;

            param[12] = new SqlParameter("@p_blood_type", SqlDbType.Int);
            param[12].Value = blood;

            param[13] = new SqlParameter("@p_image", SqlDbType.Image);
            param[13].Value = img;
```

```
    param[14] = new SqlParameter("@ID", SqlDbType.NVarChar,30);
    param[14].Value = ID;

    r = DAL.ExecuteCommand("UPDATE_PATIENT", param);
    DAL.Close();
    return r;

}

public DataTable GET_ALL_PATIENTS()
{
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();

    DataTable Dt = new DataTable();
    Dt = DAL.SelectData("GET_ALL_PATIENTS", null);
    DAL.Close();
    return Dt;

}

public DataTable SEARCH_PATIENT(string ID)
{
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DataTable Dt = new DataTable();
    SqlParameter[] param = new SqlParameter[1];
    param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 30);
    param[0].Value = ID;
    Dt = DAL.SelectData("SEARCH_PATIENT", param);
    DAL.Close();
    return Dt;
}

public int DELETE_PATIENT(string ID, string ID2)
{
    int r = 0;
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DAL.Open();
    SqlParameter[] param = new SqlParameter[2];
    param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 30);
    param[0].Value = ID;
    param[1] = new SqlParameter("@ID2", SqlDbType.NVarChar, 30);
    param[1].Value = ID2;
    r = DAL.ExecuteCommand("DELETE_PATIENT", param);
    DAL.Close();
    return r;

}



public DataTable GET_CODE_P(string brith,string phone)
{
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();

    DataTable Dt = new DataTable();
    SqlParameter[] param = new SqlParameter[2];
```

```
        param[0] = new SqlParameter("@p_brith", SqlDbType.Date);
        param[0].Value = brith;

        param[1] = new SqlParameter("@p_Phone", SqlDbType.NVarChar,15);
        param[1].Value = phone;

        Dt = DAL.SelectData("GET_CODE_P", param);
        DAL.Close();
        return Dt;

    }

    public DataTable GET_FILE_PATIENTS(string ID)
    {
        DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();

        DataTable Dt = new DataTable();
        SqlParameter[] param = new SqlParameter[1];

        param[0] = new SqlParameter("@ID", SqlDbType.NVarChar,15);
        param[0].Value = ID;

        Dt = DAL.SelectData("GET_FILE_PATIENTS", param);
        DAL.Close();
        return Dt;

    }

    public DataTable GET_ALL_diseases_chronic(string ID)
    {
        DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();

        DataTable Dt = new DataTable();
        SqlParameter[] param = new SqlParameter[1];

        param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 15);
        param[0].Value = ID;

        Dt = DAL.SelectData("GET_ALL_diseases_chronic", param);
        DAL.Close();
        return Dt;
    }

    public int ADD_diseases_chronic(string name,DateTime date,string ID)
    {
        int r = 0;
        DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
        DAL.Open();
        SqlParameter[] param = new SqlParameter[3];

        param[0] = new SqlParameter("@name", SqlDbType.NVarChar, 30);
        param[0].Value = name;

        param[1] = new SqlParameter("@date", SqlDbType.Date);
        param[1].Value = date;

        param[2] = new SqlParameter("@ID", SqlDbType.NVarChar, 30);
```

```
    param[2].Value = ID;

    r = DAL.ExecuteCommand("ADD_diseases_chronic", param);
    DAL.Close();
    return r;
}

public int UPDATE_diseases_chronic(string name,string dis_id)
{
    int r = 0;
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DAL.Open();
    SqlParameter[] param = new SqlParameter[2];

    param[0] = new SqlParameter("@name", SqlDbType.NVarChar, 30);
    param[0].Value = name;

    param[1] = new SqlParameter("@dis_id", SqlDbType.NVarChar, 10);
    param[1].Value = dis_id;

    r = DAL.ExecuteCommand("UPDATE_diseases_chronic", param);
    DAL.Close();
    return r;
}

public int DELETE_diseases_chronic(string dis_id)
{
    int r = 0;
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DAL.Open();
    SqlParameter[] param = new SqlParameter[1];

    param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 10);
    param[0].Value = dis_id;

    r = DAL.ExecuteCommand("DELETE_diseases_chronic", param);
    DAL.Close();
    return r;
}

//التطعيم

public DataTable GET_ALL_vaccine(string ID)
{
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();

    DataTable Dt = new DataTable();
    SqlParameter[] param = new SqlParameter[1];

    param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 15);
    param[0].Value = ID;

    Dt = DAL.SelectData("GET_ALL_vaccine", param);
    DAL.Close();
    return Dt;

}
```

```
public int ADD_vaccine(string name, DateTime date, string ID)
{
    int r = 0;
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DAL.Open();
    SqlParameter[] param = new SqlParameter[3];

    param[0] = new SqlParameter("@name", SqlDbType.NVarChar, 30);
    param[0].Value = name;

    param[1] = new SqlParameter("@date", SqlDbType.Date);
    param[1].Value = date;

    param[2] = new SqlParameter("@ID", SqlDbType.NVarChar, 30);
    param[2].Value = ID;

    r = DAL.ExecuteCommand("ADD_vaccine", param);
    DAL.Close();
    return r;
}

public int UPDATE_vaccine(string name, string vac_id)
{
    int r = 0;
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DAL.Open();
    SqlParameter[] param = new SqlParameter[2];

    param[0] = new SqlParameter("@name", SqlDbType.NVarChar, 30);
    param[0].Value = name;

    param[1] = new SqlParameter("@vac_id", SqlDbType.NVarChar, 10);
    param[1].Value = vac_id;

    r = DAL.ExecuteCommand("UPDATE_vaccine", param);
    DAL.Close();
    return r;
}

public int DELETE_vaccine(string vac_id)
{
    int r = 0;
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DAL.Open();
    SqlParameter[] param = new SqlParameter[1];

    param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 10);
    param[0].Value = vac_id;

    r = DAL.ExecuteCommand("DELETE_vaccine", param);
    DAL.Close();
    return r;
}

// السابقة الجراحات
public DataTable GET_ALL_surgery_per(string ID)
{
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
```

```
    DataTable Dt = new DataTable();
    SqlParameter[] param = new SqlParameter[1];

    param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 15);
    param[0].Value = ID;

    Dt = DAL.SelectData("GET_ALL_surgery_per", param);
    DAL.Close();
    return Dt;

}

public int ADD_surgery_per(string name, DateTime date, string ID)
{
    int r = 0;
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DAL.Open();
    SqlParameter[] param = new SqlParameter[3];

    param[0] = new SqlParameter("@name", SqlDbType.NVarChar, 30);
    param[0].Value = name;

    param[1] = new SqlParameter("@date", SqlDbType.Date);
    param[1].Value = date;

    param[2] = new SqlParameter("@ID", SqlDbType.NVarChar, 30);
    param[2].Value = ID;

    r = DAL.ExecuteCommand("ADD_surgery_per", param);
    DAL.Close();
    return r;
}

public int UPDATE_surgery_per(string name, string ssr_id)
{
    int r = 0;
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DAL.Open();
    SqlParameter[] param = new SqlParameter[2];

    param[0] = new SqlParameter("@name", SqlDbType.NVarChar, 30);
    param[0].Value = name;

    param[1] = new SqlParameter("@ssr_id", SqlDbType.NVarChar, 10);
    param[1].Value = ssr_id;

    r = DAL.ExecuteCommand("UPDATE_surgery_per", param);
    DAL.Close();
    return r;
}


public int DELETE_surgery_per(string ssr_id)
{
    int r = 0;
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DAL.Open();
```

```
    SqlParameter[] param = new SqlParameter[1];

    param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 10);
    param[0].Value = ssr_id;

    r = DAL.ExecuteCommand("DELETE_surgery_per", param);
    DAL.Close();
    return r;
}

// الحساسية
public DataTable GET_ALL_sensitivity(string ID)
{
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();

    DataTable Dt = new DataTable();
    SqlParameter[] param = new SqlParameter[1];

    param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 15);
    param[0].Value = ID;

    Dt = DAL.SelectData("GET_ALL_sensitivity", param);
    DAL.Close();
    return Dt;

}

public int ADD_sensitivity(string name, DateTime date, string ID)
{
    int r = 0;
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DAL.Open();
    SqlParameter[] param = new SqlParameter[3];

    param[0] = new SqlParameter("@name", SqlDbType.NVarChar, 30);
    param[0].Value = name;

    param[1] = new SqlParameter("@date", SqlDbType.Date);
    param[1].Value = date;

    param[2] = new SqlParameter("@ID", SqlDbType.NVarChar, 30);
    param[2].Value = ID;

    r = DAL.ExecuteCommand("ADD_sensitivity", param);
    DAL.Close();
    return r;
}

public int UPDATE_sensitivity(string name, string sen_id)
{
    int r = 0;
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DAL.Open();
    SqlParameter[] param = new SqlParameter[2];

    param[0] = new SqlParameter("@name", SqlDbType.NVarChar, 30);
    param[0].Value = name;
```

```
    param[1] = new SqlParameter("@sen_id", SqlDbType.NVarChar, 10);
    param[1].Value = sen_id;

    r = DAL.ExecuteCommand("UPDATE_sensitivity", param);
    DAL.Close();
    return r;
}

public int DELETE_sensitivity(string sen_id)
{
    int r = 0;
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DAL.Open();
    SqlParameter[] param = new SqlParameter[1];

    param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 10);
    param[0].Value = sen_id;

    r = DAL.ExecuteCommand("DELETE_sensitivity", param);
    DAL.Close();
    return r;
}


//العائلة امراض
public DataTable GET_ALL_diseases_family(string ID)
{
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();

    DataTable Dt = new DataTable();
    SqlParameter[] param = new SqlParameter[1];

    param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 15);
    param[0].Value = ID;

    Dt = DAL.SelectData("GET_ALL_diseases_family", param);
    DAL.Close();
    return Dt;
}

public int ADD_diseases_family(string name, DateTime date, string ID)
{
    int r = 0;
    DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
    DAL.Open();
    SqlParameter[] param = new SqlParameter[3];

    param[0] = new SqlParameter("@name", SqlDbType.NVarChar, 30);
    param[0].Value = name;

    param[1] = new SqlParameter("@date", SqlDbType.Date);
    param[1].Value = date;

    param[2] = new SqlParameter("@ID", SqlDbType.NVarChar, 30);
    param[2].Value = ID;

    r = DAL.ExecuteCommand("ADD_diseases_family", param);
    DAL.Close();
```

```
            return r;
        }

        public int UPDATE_diseases_family(string name, string dfam_id)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
            SqlParameter[] param = new SqlParameter[2];

            param[0] = new SqlParameter("@name", SqlDbType.NVarChar, 30);
            param[0].Value = name;

            param[1] = new SqlParameter("@dfam_id", SqlDbType.NVarChar, 10);
            param[1].Value = dfam_id;

            r = DAL.ExecuteCommand("UPDATE_diseases_family", param);
            DAL.Close();
            return r;
        }

        public int DELETE_diseases_family(string dfam_id)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
            SqlParameter[] param = new SqlParameter[1];

            param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 10);
            param[0].Value = dfam_id;

            r = DAL.ExecuteCommand("DELETE_diseases_family", param);
            DAL.Close();
            return r;
        }
        // ================================== المرضية الزيارات
============================


        //الطبي الرقم باسخدام المرضية الزيارات كل استدعاء

        public DataTable GET_ALL_STATUS_PAT(string ID)
        {
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();

            DataTable Dt = new DataTable();
            SqlParameter[] param = new SqlParameter[1];

            param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 15);
            param[0].Value = ID;

            Dt = DAL.SelectData("GET_ALL_STATUS_PAT", param);
            DAL.Close();
            return Dt;
        }

        //الزيارة رقم باستخدام المرضي الزيارات كل استدعاء
```

```
        public DataTable GET_ALL_STATUS_PAT_veiw(int ID)
        {
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();

            DataTable Dt = new DataTable();
            SqlParameter[] param = new SqlParameter[1];

            param[0] = new SqlParameter("@ID", SqlDbType.Int);
            param[0].Value = ID;

            Dt = DAL.SelectData("GET_ALL_STATUS_PAT_veiw", param);
            DAL.Close();
            return Dt;
        }


        //طبية زيارة اضافة
        public int ADD_STATUS_PAT(string id_pat, string id_sec, string compl,
DateTime date_reg, string id_host, string stat_level)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
            SqlParameter[] param = new SqlParameter[6];

            param[0] = new SqlParameter("@id_pat", SqlDbType.NVarChar, 15);
            param[0].Value = id_pat;

            param[1] = new SqlParameter("@sec", SqlDbType.NVarChar,10);
            param[1].Value = id_sec;

            param[2] = new SqlParameter("@compl", SqlDbType.NVarChar,2000);
            param[2].Value = compl;

            param[3] = new SqlParameter("@date_reg", SqlDbType.Date);
            param[3].Value = date_reg;

            param[4] = new SqlParameter("@host_st", SqlDbType.NVarChar, 10);
            param[4].Value = id_host;

            param[5] = new SqlParameter("@stat_name_level", SqlDbType.NVarChar, 20);
            param[5].Value = stat_level;

            r = DAL.ExecuteCommand("ADD_STATUS_PAT", param);
            DAL.Close();
            return r;
        }

        //طبية زيارة تعديل
        public int UPDATE_STATUS_PAT(string id_pat, string id_sec, string compl,
DateTime date_reg, string id_host,string ID)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
            SqlParameter[] param = new SqlParameter[6];

            param[0] = new SqlParameter("@id_pat", SqlDbType.NVarChar, 15);
```

```
            param[0].Value = id_pat;

            param[1] = new SqlParameter("@sec", SqlDbType.NVarChar, 10);
            param[1].Value = id_sec;

            param[2] = new SqlParameter("@compl", SqlDbType.NVarChar, 2000);
            param[2].Value = compl;

            param[3] = new SqlParameter("@date_reg", SqlDbType.Date);
            param[3].Value = date_reg;

            param[4] = new SqlParameter("@host_st", SqlDbType.NVarChar, 10);
            param[4].Value = id_host;

            param[5] = new SqlParameter("@ID", SqlDbType.NVarChar, 10);
            param[5].Value = ID;

            r = DAL.ExecuteCommand("UPDATE_STATUS_PAT", param);
            DAL.Close();
            return r;
        }

        // طبية زيارة حذف
        public int DELETE_STATUS_PAT(string ID)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
            SqlParameter[] param = new SqlParameter[1];

            param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 10);
            param[0].Value = ID;

            r = DAL.ExecuteCommand("DELETE_STATUS_PAT", param);
            DAL.Close();
            return r;
        }

        //==================================الجلسات
المرضية==================================

        //  الزيارة رقم باستخدام المرضية الجلسات كل اسدعاء
        public DataTable GET_ALL_SESSION_PAT(string ID)
        {
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();

            DataTable Dt = new DataTable();
            SqlParameter[] param = new SqlParameter[1];

            param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 10);
            param[0].Value = ID;

            Dt = DAL.SelectData("GET_ALL_SESSION_PAT", param);
            DAL.Close();
            return Dt;
        }

        //  الجلسة رقم باستخدام المرضية الجلسات كل اسدعاء
```

```
        public DataTable GET_ALL_SESSION_PAT_view(int ID)
        {
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();

            DataTable Dt = new DataTable();
            SqlParameter[] param = new SqlParameter[1];

            param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 10);
            param[0].Value = ID;
            Dt = DAL.SelectData("GET_ALL_SESSION_PAT_view", param);
            DAL.Close();
            return Dt;
        }

        //علاجية جلسة اضافة
        public int ADD_SESSION_PAT(int id_stat, string doct, string diagn, string
recomm, DateTime date_reg, string sess_level)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
            SqlParameter[] param = new SqlParameter[6];

            param[0] = new SqlParameter("@id_stat", SqlDbType.Int);
            param[0].Value = id_stat;

            param[1] = new SqlParameter("@doct", SqlDbType.NVarChar, 10);
            param[1].Value = doct;

            param[2] = new SqlParameter("@diagn", SqlDbType.NVarChar, 2000);
            param[2].Value = diagn;

            param[3] = new SqlParameter("@recomm", SqlDbType.NVarChar ,2000);
            param[3].Value = recomm;

            param[4] = new SqlParameter("@date_reg", SqlDbType.Date );
            param[4].Value = date_reg;

            param[5] = new SqlParameter("@sess_level", SqlDbType.NVarChar,20);
            param[5].Value = sess_level;

            r = DAL.ExecuteCommand("ADD_SESSION_PAT", param);
            DAL.Close();
            return r;
        }

        //علاجية الجلسة تعديل
        public int UPDATE_SESSION_PAT(string doct, string diagn, string recomm,
string ID)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
            SqlParameter[] param = new SqlParameter[4];


            param[0] = new SqlParameter("@doct", SqlDbType.NVarChar, 10);
            param[0].Value = doct;
```

```
            param[1] = new SqlParameter("@diagn", SqlDbType.NVarChar, 2000);
            param[1].Value = diagn;

            param[2] = new SqlParameter("@recomm", SqlDbType.NVarChar, 2000);
            param[2].Value = recomm;

            param[3] = new SqlParameter("@ID", SqlDbType.NVarChar,10);
            param[3].Value = ID;

            r = DAL.ExecuteCommand("UPDATE_SESSION_PAT", param);
            DAL.Close();
            return r;
        }

        //علاجية الجلسة حذف
        public int DELETE_SESSION_PAT(string ID)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
            SqlParameter[] param = new SqlParameter[1];

            param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 10);
            param[0].Value = ID;

            r = DAL.ExecuteCommand("DELETE_SESSION_PAT", param);
            DAL.Close();
            return r;
        }

        //================================ الاولي الفحص ========================

        //  الجلسة رقم باستخدام  الاولي الفحص كل اسدعاء
        public DataTable GET_ALL_CHECKUP1_PAT(int ID)
        {
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();

            DataTable Dt = new DataTable();
            SqlParameter[] param = new SqlParameter[1];

            param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 10);
            param[0].Value = ID;
            Dt = DAL.SelectData("GET_ALL_CHECKUP1_PAT", param);
            DAL.Close();
            return Dt;
        }

        //اولي فحص اضافة
        public int ADD_CHECKUP1_PAT(string sess_id, string temp, string press, string
heart, string weigh, string lengt)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
            SqlParameter[] param = new SqlParameter[6];

            param[0] = new SqlParameter("@sess_id", SqlDbType.NVarChar,10);
```

```
            param[0].Value = sess_id;

            param[1] = new SqlParameter("@temp", SqlDbType.NVarChar, 10);
            param[1].Value = temp;

            param[2] = new SqlParameter("@press", SqlDbType.NVarChar, 10);
            param[2].Value = press;

            param[3] = new SqlParameter("@heart", SqlDbType.NVarChar, 10);
            param[3].Value = heart;

            param[4] = new SqlParameter("@weigh", SqlDbType.NVarChar, 10);
            param[4].Value = weigh;

            param[5] = new SqlParameter("@lengt", SqlDbType.NVarChar, 10);
            param[5].Value = lengt;

            r = DAL.ExecuteCommand("ADD_CHECKUP1_PAT", param);
            DAL.Close();
            return r;
        }
        //اولي فحص تعديل
        public int UPDATE_CHECKUP1_PAT(string sess_id, string temp, string press,
string heart, string weigh, string lengt)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
            SqlParameter[] param = new SqlParameter[6];

            param[0] = new SqlParameter("@sess_id", SqlDbType.NVarChar, 10);
            param[0].Value = sess_id;

            param[1] = new SqlParameter("@temp", SqlDbType.NVarChar, 10);
            param[1].Value = temp;

            param[2] = new SqlParameter("@press", SqlDbType.NVarChar, 10);
            param[2].Value = press;

            param[3] = new SqlParameter("@heart", SqlDbType.NVarChar, 10);
            param[3].Value = heart;

            param[4] = new SqlParameter("@weigh", SqlDbType.NVarChar, 10);
            param[4].Value = weigh;

            param[5] = new SqlParameter("@lengt", SqlDbType.NVarChar, 10);
            param[5].Value = lengt;

            r = DAL.ExecuteCommand("UPDATE_CHECKUP1_PAT", param);
            DAL.Close();
            return r;
        }
        //اولي فحص حذف
        public int DELETE_CHECKUP1_PAT(string ID)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
```

```
            SqlParameter[] param = new SqlParameter[1];

            param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 10);
            param[0].Value = ID;

            r = DAL.ExecuteCommand("DELETE_CHECKUP1_PAT", param);
            DAL.Close();
            return r;
        }

        //============================الادوية====================
        //  الجلسة رقم باستخدام  الادوية كل اسدعاء
        public DataTable GET_ALL_Medicine_PAT(int ID)
        {
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();

            DataTable Dt = new DataTable();
            SqlParameter[] param = new SqlParameter[1];

            param[0] = new SqlParameter("@ID", SqlDbType.NVarChar, 10);
            param[0].Value = ID;
            Dt = DAL.SelectData("GET_ALL_Medicine_PAT", param);
            DAL.Close();
            return Dt;
        }

        //دواء اضافة
        public int ADD_Medicine_PAT(string med_name, string med_dos, string med_rng,
    string med_id_sess)
        {
            int r = 0;
            DAL.DataAccessLayar DAL = new DAL.DataAccessLayar();
            DAL.Open();
            SqlParameter[] param = new SqlParameter[4];

            param[0] = new SqlParameter("@med_name", SqlDbType.NVarChar, 40);
            param[0].Value = med_name;

            param[1] = new SqlParameter("@med_dos", SqlDbType.NVarChar, 40);
            param[1].Value = med_dos;

            param[2] = new SqlParameter("@med_rng", SqlDbType.NVarChar, 30);
            param[2].Value = med_rng;

            param[3] = new SqlParameter("@med_id_sess", SqlDbType.NVarChar, 10);
            param[3].Value = med_id_sess;

            r = DAL.ExecuteCommand("ADD_Medicine_PAT", param);
            DAL.Close();
            return r;
        }
}
```

- كود form الإضافة والتعديل للمريض:

```
nmespace SMR_system_of_yemen.PL
{
    public partial class FRM_ADD_PATIENTS : Form
    {
        public string state = "add";
        BL.CLS_GET_COMBOX prd=new BL.CLS_GET_COMBOX();
        BL.CLS_PATIENTS patt = new BL.CLS_PATIENTS();

        public FRM_ADD_PATIENTS()
        {
            InitializeComponent();
            btn_cre_cod.Enabled = false;
            btn_pat_rpt.Enabled = false;


            cmbCity.DataSource = prd.GET_ALL_CITY();
            cmbCity.DisplayMember = "city_name";
            cmbCity.ValueMember = "city_id";

            cmbBlood.DataSource = prd.GET_ALL_BLOOD();
            cmbBlood.DisplayMember = "blood_name";
            cmbBlood.ValueMember = "blood_id";

            cmbGender.DataSource = prd.GET_ALL_GENDER();
            cmbGender.DisplayMember = "gender_name";
            cmbGender.ValueMember = "gender_id";

            cmbNation.DataSource = prd.GET_ALL_NATIONALITY();
            cmbNation.DisplayMember = "nat_name";
            cmbNation.ValueMember = "nat_id";

        }

        private void FRM_ADD_PATIENTS_Load(object sender, EventArgs e)
        {

        }

        private void comboBox2_SelectedIndexChanged(object sender, EventArgs e)
        {

        }

        private void button1_Click(object sender, EventArgs e)
        {
            OpenFileDialog ofd = new OpenFileDialog();
            ofd.Filter = "الصور ملفات | *.JPG; *.PNG; *.GIF; .BMP";
            if(ofd.ShowDialog()== DialogResult.OK)
            {
                picImageP.Image = Image.FromFile(ofd.FileName);
            }
        }

        private void btn_Paddseve_Click(object sender, EventArgs e)
        {
```

```
            try
            {
                if (txt_Paddress.Text != string.Empty || txt_Pinsurance.Text !=
string.Empty || txt_Pname.Text != string.Empty || txt_PnumLdaintity.Text !=
string.Empty || txt_Pphone.Text != string.Empty || txt_Pphoneneer.Text !=
string.Empty || txt_Pwork.Text != string.Empty)
                {
                    MemoryStream ms = new MemoryStream();
                    picImageP.Image.Save(ms, picImageP.Image.RawFormat);
                    byte[] byteImage = ms.ToArray();

                    if (state == "add")
                    {

                        int i = 0;
                        i = patt.ADD_PATIENT(txt_Pname.Text.Trim(),
Convert.ToBoolean(cmbGender.SelectedValue), Convert.ToDateTime(date_Pbirth.Text)
                            , txt_Pwork.Text.Trim(),
cmbLdentity.SelectedItem.ToString(), txt_PnumLdaintity.Text.Trim(),
Convert.ToInt32(cmbCity.SelectedValue), txt_Paddress.Text.Trim(),
txt_Pphone.Text.Trim(),
                            txt_Pphoneneer.Text.Trim(),
Convert.ToInt32(cmbNation.SelectedValue), txt_Pinsurance.Text.Trim(),
Convert.ToInt32(cmbBlood.SelectedValue), byteImage);

                        if (i > 0)
                        {
                            MessageBox.Show("بنجاح الاضافة تمت", "الاضافة عملية",
MessageBoxButtons.OK, MessageBoxIcon.Information);
                            btn_cre_cod.Enabled = true;
                            btn_pat_rpt.Enabled = true;

                        }
                        else
                            MessageBox.Show("لم تمت الاضافة هناك خطاء ...!!", "الاضافة عملية",
MessageBoxButtons.OK, MessageBoxIcon.Information);

                    }
                    else
                    {
                        int i = 0;
                        i = patt.UPDATE_PATIENT(txt_Pname.Text.Trim(),
Convert.ToBoolean(cmbGender.SelectedValue), Convert.ToDateTime(date_Pbirth.Text)
                            , txt_Pwork.Text.Trim(),
cmbLdentity.SelectedItem.ToString(), txt_PnumLdaintity.Text.Trim(),
Convert.ToInt32(cmbCity.SelectedValue), txt_Paddress.Text.Trim(),
txt_Pphone.Text.Trim(),
                            txt_Pphoneneer.Text.Trim(),
Convert.ToInt32(cmbNation.SelectedValue), txt_Pinsurance.Text.Trim(),
Convert.ToInt32(cmbBlood.SelectedValue), byteImage, L_IDCOD.Text);

                        if (i > 0)
                        {
                            MessageBox.Show("بنجاح التعديل تم", "التعديل عملية",
MessageBoxButtons.OK, MessageBoxIcon.Information);
                            btn_cre_cod.Enabled = true;
                            btn_pat_rpt.Enabled = true;
                        }
```

```
                        else
                            MessageBox.Show("التعديل عملية", "خطاء هناك التعيل يتم لم ...!!", 
MessageBoxButtons.OK, MessageBoxIcon.Information);


                    }

                    FRM_PATIENTS.getpatForm.dataGrid_patient.DataSource = 
patt.GET_ALL_PATIENTS();
                    FRM_Patint_information.getinformationForm.p_tb = 
patt.GET_FILE_PATIENTS(lab_p_code.Text);
                    FRM_Patint_information.getinformationForm.Refresh();
                }
                else
                {
                    MessageBox.Show("كامل البيانات ادخال يجب", "حطاء", MessageBoxButtons.OK, 
MessageBoxIcon.Error);
                }
            }

            catch (Exception )
            {
                //MessageBox.Show("بالانترنت الاتصال من تحقق ... معذرة", "الاتصال", 
MessageBoxButtons.OK, MessageBoxIcon.Error);

            }
        }

        private void btn_cre_cod_Click(object sender, EventArgs e)
        {
            try
            {
                DataTable x = patt.GET_CODE_P(date_Pbirth.Text, 
txt_Pphone.Text.Trim());



                FRM_GET_CODE frm = new FRM_GET_CODE();

                //frm.txtCodeP.Text = x.Rows[0][3].ToString();

                frm.txtCodeP.Text = x.Rows[0]["p_code"].ToString();



                frm.txtName.Text = x.Rows[0]["p_name"].ToString();


                frm.txtDataReg.Text = x.Rows[0]["p_date_reg"].ToString();


                frm.txtNameLogin.Text = x.Rows[0]["p_code"].ToString();

                frm.txtPassLogin.Text = x.Rows[0]["p_phone"].ToString();

                frm.ShowDialog();
            }
```

```
            catch
            {
                MessageBox.Show("معذرة ... تحقق من الاتصال بالانترنت", "الاتصال",
MessageBoxButtons.OK, MessageBoxIcon.Error);
            }
        }

        private void picImageP_Click(object sender, EventArgs e)
        {

        }

        private void btn_Pclose_Click(object sender, EventArgs e)
        {
            Close();
        }

        private void picImageP_DoubleClick(object sender, EventArgs e)
        {
            OpenFileDialog ofd = new OpenFileDialog();
            ofd.Filter = "ملفات الصور | *.JPG; *.PNG; *.GIF; .BMP";
            if (ofd.ShowDialog() == DialogResult.OK)
            {
                picImageP.Image = Image.FromFile(ofd.FileName);
            }
        }

        private void btn_pat_rpt_Click(object sender, EventArgs e)
        {
            try
            {
                RPT.rpt_pat_single_code myReport = new RPT.rpt_pat_single_code();

                myReport.SetParameterValue("@p_brith", date_Pbirth.Text);
                myReport.SetParameterValue("@p_Phone", txt_Pphone.Text.Trim());
                RPT.FRM_RPT_HOST myform = new RPT.FRM_RPT_HOST();
                myform.crystalReportViewer1.ReportSource = myReport;
                myform.ShowDialog();
            }
            catch
            {
                MessageBox.Show("معذرة ... تحقق من الاتصال بالانترنت", "الاتصال",
MessageBoxButtons.OK, MessageBoxIcon.Error);
            }
        }

        private void label14_Click(object sender, EventArgs e)
        {
            this.Close();
        }
    }
}
```