



جامعة طرابلس كلية تقنية المعلومات

تقرير Clinic Management System(CMS)
في مادة Advanced Programming with java

اعداد:

2231807856 مطاع صبري الطربان
2231806247 منيب محمد العجيلي
2231807532 أيوب ناجي حمزة

تحت اشراف:

د. مي مفتاح البعباع

خريف – 2025

1. نظرة عامة عن النظام:

نظام إدارة العيادة هو تطبيق مكتبي مكتوب بلغة Java باستخدام مكتبة Swing

وقاعدة بيانات MySQL.

يهدف النظام إلى إدارة بيانات المرضى داخل العيادة بطريقة سهلة ومنظمة، بحيث

يسمح للمستخدم بإضافة مرضى جدد، تعديل معلوماتهم، حذفهم، عرضهم في جدول،

بالإضافة إلى تصدير البيانات إلى ملف نصي.

2. التقنيات والأدوات المستخدمة :

- لغة البرمجة : Java
- واجهات : Swing GUI
- بيئة التطوير : Net beans
- قاعدة البيانات : MySQL
- الاتصال بقاعدة البيانات : JDBC
- استخدام Collections (ArrayList)
- برمجة كينونية : OOP
- Multithreading
- ملفات الإدخال والإخراج : I/O Streams

3. أهم الخصائص التي يوفرها النظام :

- إضافة مريض جديد
- تعديل بيانات مريض
- حذف مريض
- تحديث وعرض جميع المرضى في جدول
- عرض الوقت بشكل حي (ساعة تتحدّث كل ثانية)
- تعبئة الحقول تلقائيًا عند اختيار صف من الجدو
- إظهار عدد المرضى بشكل تلقائي
- اختيار التشخيص من خلال ComboBox
- تصدير بيانات المرضى إلى ملف نصي (Patient.txt)
- تصميم واجهة رسومية بسيطة وسهلة الاستخدام

4. طريقة تنفيذ كل مطلوب في المشروع :

1. استخدام (Exception Handling (Try & catch :

تم استخدام آلية معالجة الأخطاء try & catch في جميع العمليات الحساسة داخل النظام،
مثل:

- عمليات الإضافة
- التعديل
- الحذف
- الاتصال بقاعدة البيانات
- تصدير البيانات إلى الملفات

وذلك لضمان استمرار عمل البرنامج حتى في حالة حدوث خطأ غير متوقع، ومنع توقف
الواجهة أو فقدان البيانات.
هذا يعكس تطبيقاً فعلياً لأفضل ممارسات البرمجة وزيادة موثوقية البرنامج.

2. Collections Framework:

تم استخدام ArrayList لتخزين وتجميع بيانات المرضى المسترجعة من قاعدة البيانات.
كما تم الاعتماد عليها في:

- عرض البيانات داخل جدول الواجهة (JTable)
- تنفيذ عمليات المعالجة المختلفة على السجلات

يتيح هذا استخدام بنية بيانات ديناميكية ومرنة، مما يحسن الأداء ويوفر إدارة أسهل
لعدد كبير من السجلات.

3. JDBC وعمليات CRUD:

تم ربط النظام بقاعدة بيانات MySQL باستخدام JDBC من خلال كلاس مخصص لإدارة البيانات: PatientDAO

ويشمل هذا الربط تنفيذ عمليات CRUD كاملة:

- Create: إضافة مريض جديد إلى قاعدة البيانات
- Read: جلب جميع المرضى وعرضهم في جدول الواجهة
- Update: تعديل بيانات مريض محدد
- Delete: حذف سجل من قاعدة البيانات

كما تم استخدام PreparedStatement لضمان الحماية من هجمات SQL Injection وتحسين الأداء.

4. Gui & event Handling باستخدام مكتبة swing:

تم تصميم واجهة رسومية متكاملة باستخدام مكونات Swing، وتشمل:

JFrame – JButton – JTable – JTextField – JComboBox – JLabel – JScrollPane

وتم تطبيق Event Handling من خلال:

- ActionListener: لربط الأزرار بالأوامر المنفذة
- ListSelectionListener: لملء الحقول تلقائيًا عند اختيار صف من الجدول

مما يوفر تجربة استخدام سلسة وواضحة.

5. I/O Steams :

يتضمن النظام إمكانية تصدير بيانات المرضى إلى ملف نصي باستخدام الصنف FileWriter حيث يتم إنشاء ملف خارجي يحتوي على تقرير منسق يشمل جميع بيانات المرضى، وذلك بالاعتماد على مفهوم تدفقات الإدخال والإخراج (I/O Streams) لتنظيم عملية التعامل مع الملفات.

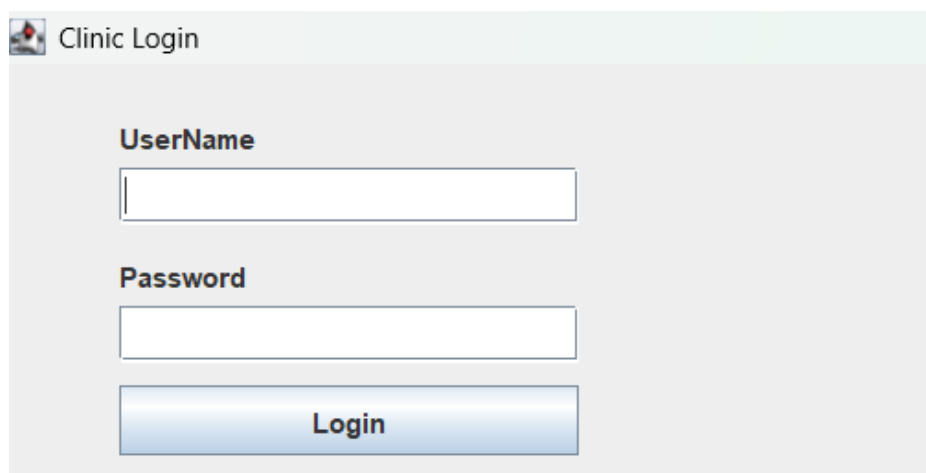
6. Multithreading :

يحتوي النظام على ميزة الساعة الحية التي تعمل في الخلفية باستخدام Thread منفصل. يقوم هذا thread بتحديث الوقت في الواجهة كل ثانية دون التأثير على أداء البرنامج أو إيقاف الواجهة. وبذلك يتم تحقيق أداء سلس في الواجهة الرسومية.

5. ScreenShots :

(1) واجهة تسجيل الدخول :

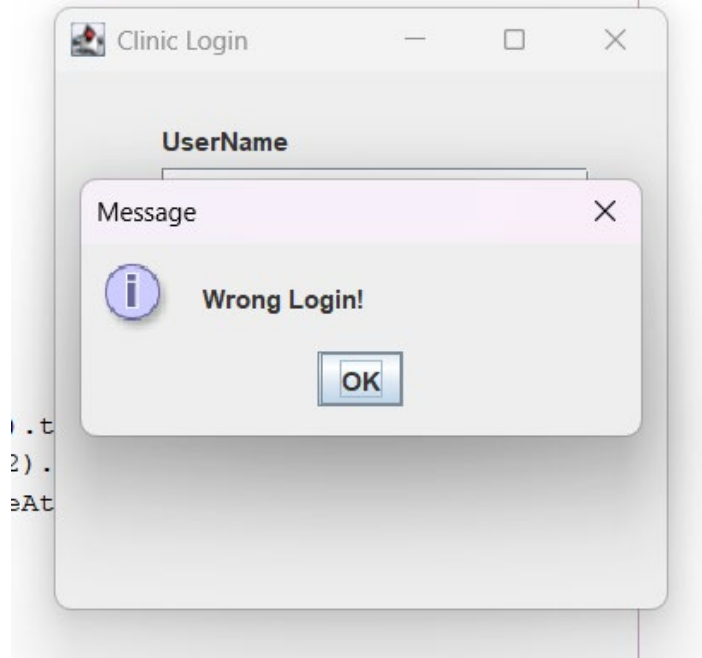
حيث يقوم المستخدم بإدخال اسم المستخدم وكلمة المرور للتمكن من الوصول إلى نظام إدارة العيادة. تهدف هذه الواجهة إلى حماية النظام ومنع الوصول غير المذوّل.

A screenshot of a web application titled "Clinic Login". The form has a light gray background. It contains two input fields: "UserName" and "Password", both with white text boxes and blue borders. Below the "Password" field is a blue "Login" button with white text. The title "Clinic Login" is displayed in a small font at the top left of the form area.

الشكل (1): واجهة تسجيل الدخول للنظام.

(2) التحقق من صحة بيانات تسجيل الدخول :

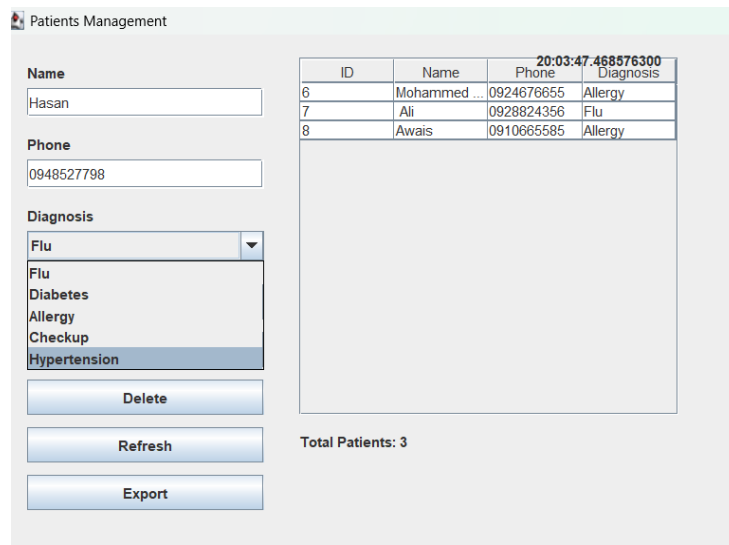
رسالة تنبيه تظهر عند إدخال بيانات غير صحيحة حيث يتم استخدام Exception Handling للتحقق من صحة الإدخال.



الشكل (2) رسالة خطأ تسجيل دخول.

(3) الواجهة الرئيسية بعد تسجيل الدخول (Main Dashboard) :

شاشة عرض جميع المرضى المسجلين في النظام، ويتم جلب هذه البيانات من قاعدة البيانات باستخدام JDBC وعرضها داخل جدول (JTable).



الشكل (3) رسالة خطأ تسجيل دخول.

4) إضافة مريض جديد (Add Patient Form) :

حيث يقوم المستخدم بإدخال اسم المريض، رقم الهاتف، واختيار التشخيص من قائمة جاهزة، ثم الضغط على زر Add لتخزين البيانات في قاعدة البيانات.

The screenshot shows the 'Patients Management' application window. On the left, there are input fields for 'Name' (containing 'Hasan'), 'Phone' (containing '0948527798'), and a 'Diagnosis' dropdown menu (set to 'Hypertension'). Below these are buttons for 'Add', 'Edit', 'Delete', 'Refresh', and 'Export'. On the right, there is a table with columns 'ID', 'Name', 'Phone', and 'Diagnosis'. The table contains three rows of data. Below the table, it says 'Total Patients: 3'. A 'Message' dialog box is open in the foreground, displaying an information icon, the text 'Saved!', and an 'OK' button.

ID	Name	Phone	Diagnosis
6	Mohammed ...	0924676655	Allergy
7	Ali	0928824356	Flu
8	Awais	0910665585	Allergy

الشكل (4) اضافة مريض

بعد الاضافة اصبح الجدول هكذا :

تم إضافة Hasan ب ID رقم 10 وتم زيادة عداد المرضى ليصبح 4

The screenshot shows the 'Patients Management' application window after adding a new patient. The 'Name' field now contains 'Ali' and the 'Phone' field contains '0928824356'. The 'Diagnosis' dropdown menu is set to 'Flu'. The 'Add' button is highlighted. The table on the right now contains four rows of data. Below the table, it says 'Total Patients: 4'.

ID	Name	Phone	Diagnosis
6	Mohammed ...	0924676655	Allergy
7	Ali	0928824356	Flu
8	Awais	0910665585	Allergy
10	Hasan	0948527798	Hypertension

الشكل (5) الجدول بعد الاضافة

(5) حذف مريض :

حيث يتم عرض رسالة تأكيد قبل تنفيذ عملية الحذف لضمان عدم إزالة السجلات بالخطأ.

Patients Management

Name:

Phone:

Diagnosis:

ID	Name	Phone	Diagnosis
6	Mohammed ...	0924676655	Allergy
8	Ali	0910665585	Allergy
10	Hasan	0948527798	Hypertension

Total Patients: 3

Confirm Delete

Are you sure to delete this patient?

الشكل (6) الجدول بعد الاضافة-

Patients Management

Name:

Phone:

Diagnosis:

ID	Name	Phone	Diagnosis
6	Mohammed ...	0924676655	Allergy
10	Hasan	0910665585	Hypertension

Total Patients: 2

الشكل (7) تم حذف Ali.

6. توزيع المهام :

♦ مطاع صبري الطربان :

- تصميم وتنفيذ واجهة المستخدم (GUI) باستخدام مكتبة Swing.
- ربط الأزرار بالأحداث (Event Handling).
- تنسيق الواجهات وتحسين سهولة الاستخدام.
- المساهمة في اختبار الواجهة والتأكد من خلوها من الأخطاء.

♦ منيب محمد العجيلي :

- تصميم وربط قاعدة البيانات MySQL.
- تنفيذ عمليات CRUD باستخدام JDBC.
- إنشاء كلاس PatientDAO.
- التعامل مع Exception Handling لضمان استقرار النظام.
- اختبار الاتصال بقاعدة البيانات.

♦ أيوب ناجي حمزة :

- تنفيذ ميزة تصدير البيانات إلى ملف نصي باستخدام File I/O (FileWriter).
- تطبيق Multithreading لعرض الساعة الحية.
- استخدام Collections (ArrayList) لتخزين البيانات.
- المساهمة في توثيق المشروع وكتابة التقرير النهائي.
- اختبار النظام واكتشاف الأخطاء .

7. التحديات والدروس المستفادة :

واجهت بعض الصعوبات في التعامل مع قاعدة البيانات وربطها بالواجهة، وفي تحديث الجدول تلقائياً بعد العمليات. كما تعلمت كيفية استخدام Threads دون التأثير على الواجهة، وكيفية كتابة ومعالجة الملفات باستخدام FileWriter.