



وزارة التعليم العالي والبحث العلمي

جامعة طرابلس

كلية تقنية المعلومات/ قسم هندسة البرمجيات

مشروع في مقرر جافا متقدمة ITSE322 بعنوان

حجز تذاكر الأفلام

أعد كجزء لهذا المقرر

اعداد

2211845034 سلمى حمدي الصغير

2211845182 صفاء محمد القضاظ

2211846481 ندى يوسف خليفة

2211846597 هداية عبدالناصر بنّيس

تحت إشراف الأستاذة

مي البعباع

1. نظرة عامة على النظام :

نظام حجز تذاكر الأفلام عبر الإنترنت هو تطبيق يهدف إلى توفير وسيلة بديلة ومريحة للعملاء لشراء تذاكر السينما .يتيح النظام للمستخدمين تسجيل الدخول (أو التسجيل كجديد) ، ثم اختيار الفيلم ، والمسرح الذي يعرض الفيلم فيه ، وتوقيت العرض ، وعدد التذاكر المطلوبة . بعد تأكيد التفاصيل، يمكن للمستخدمين المتابعة لحجز التذاكر ، مع إمكانية الدفع عبر خيارات مختلفة مثل بطاقات الائتمان أو الخصم أو الخدمات المصرفية عبر الإنترنت أو المحافظ الإلكترونية .يعرض النظام تفاصيل الحجز ويقوم بتوليد إيصال يمكن طباعته لتقديمه في شبك التذاكر للحصول على التذاكر الفعلية .بالإضافة إلى ذلك، يوفر النظام وظيفة لإلغاء التذاكر المحجوزة ولأن النظام مصمم ليكون آلياً، لا يحتاج الموظفون إلى التدخل يدوياً في الطلبات بمجرد تلقيها عبر النظام بعد تغذية البيانات في قاعدة البيانات.

2. التقنيات والأدوات المستخدمة :

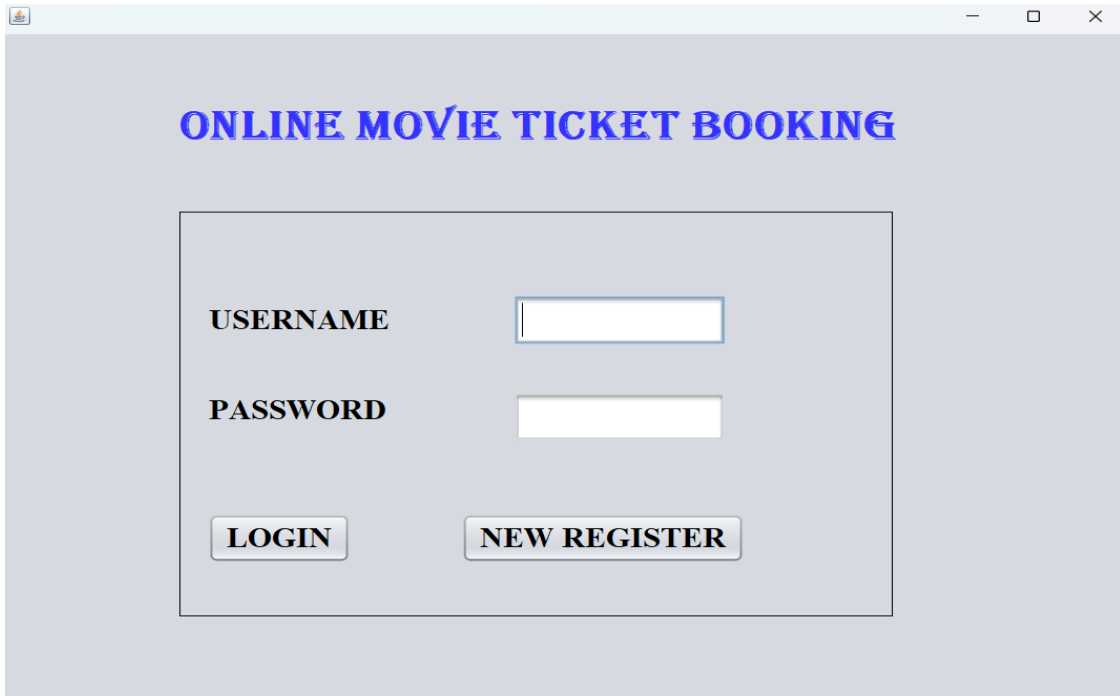
- لغة البرمجة : java
- بيئة التطوير المتكاملة (IDE) : NetBeans
- بناء واجهة المستخدم الرسومية (GUI Framework) : Java Swing
- قاعدة البيانات : MySQL
- اتصال قاعدة البيانات : JDBC (Java Database Connectivity)
- نظام التحكم بالإصدار (Version Control System) : GitHub لاستضافة مستودع المشروع

3. الميزات الرئيسية المنفذة :

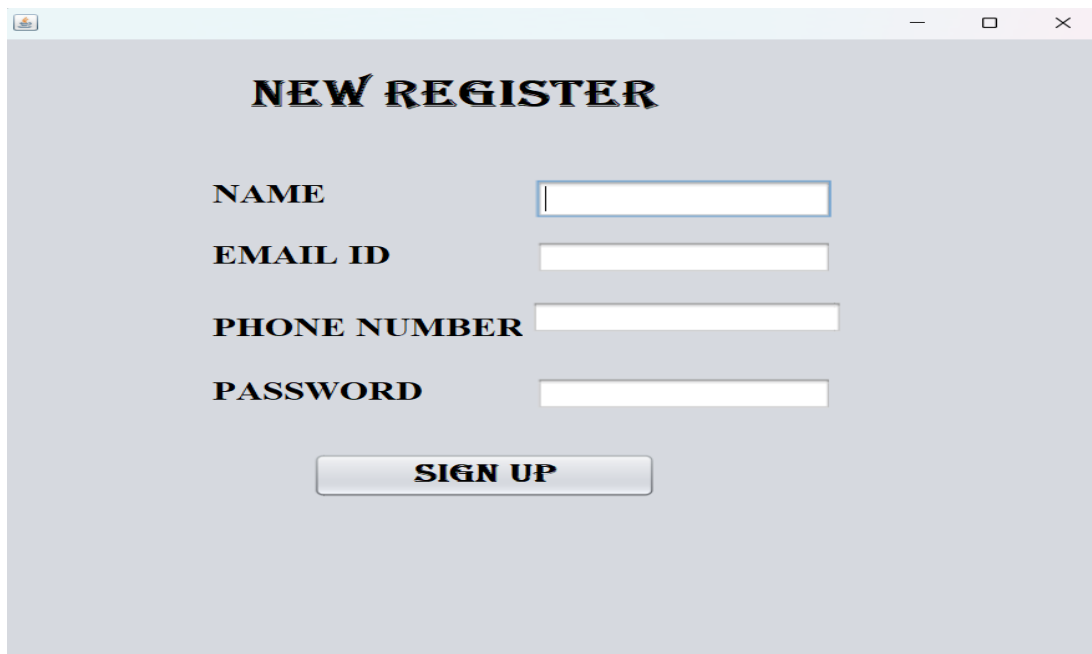
- a. وحدة تسجيل دخول المستخدمين الحاليين.
- b. وحدة تسجيل مستخدمين جدد.
- c. اختيار الفيلم.

- d. اختيار المسرح.
- e. اختيار التاريخ.
- f. اختيار وقت العرض.
- g. تحديد عدد التذاكر المطلوب حجزها.
- h. عرض تفاصيل الحجز (عدد التذاكر، المسرح، الفيلم، التاريخ، وقت العرض، السعر).
- i. إمكانية حجز التذكرة بعد التأكيد.
- j. إمكانية إلغاء التذكرة.
- k. اختيار نوع الدفع (بطاقة ائتمان، بطاقة خصم، خدمات مصرفية عبر الإنترنت، محفظة).
- l. إدخال تفاصيل البطاقة (رقم البطاقة، اسم حامل البطاقة، CVV، تاريخ انتهاء الصلاحية).
- m. توليد إيصال تفصيلي يحتوي على معلومات الحجز والسعر.
- n. رسالة شكر بعد إتمام الحجز.

2. لقطات شاشة (screenshots) من التطبيق :



الشكل 1: واجهة تسجيل الدخول (User Login Interface)



A screenshot of a web browser window displaying a registration form titled "NEW REGISTER". The form is set against a light gray background. It contains four input fields, each preceded by a label: "NAME", "EMAIL ID", "PHONE NUMBER", and "PASSWORD". Below these fields is a "SIGN UP" button. The browser window has standard OS controls (minimize, maximize, close) in the top right corner.

NEW REGISTER

NAME

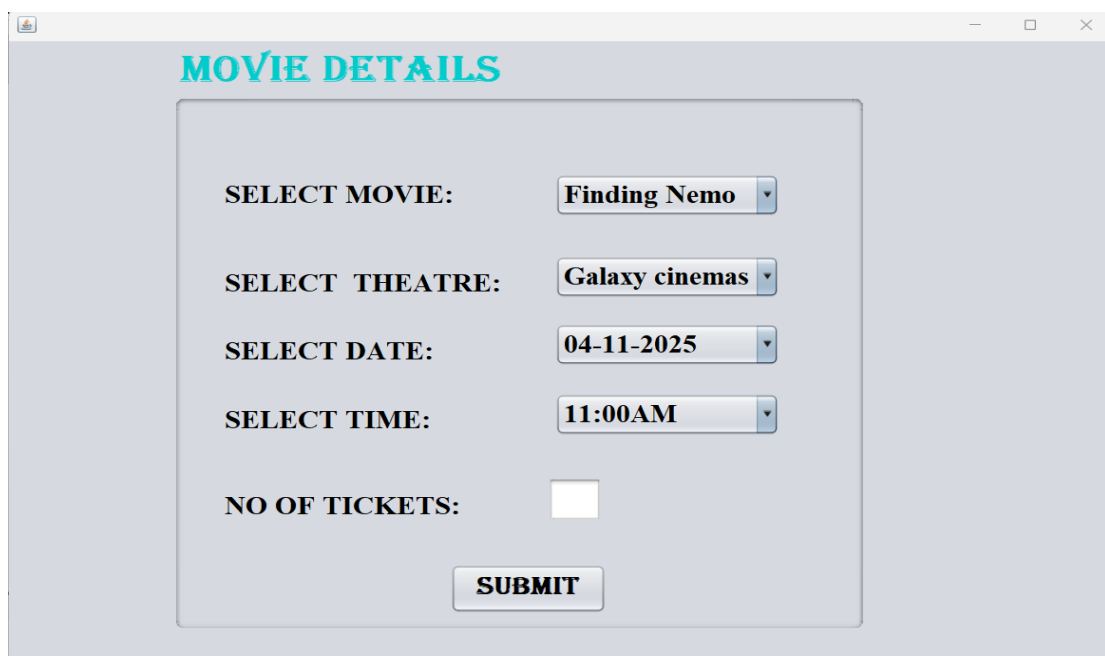
EMAIL ID

PHONE NUMBER

PASSWORD

SIGN UP

الشكل 2: واجهة تسجيل مستخدم جديد (New User Registration Screen)



A screenshot of a web browser window displaying a movie selection form titled "MOVIE DETAILS". The form is enclosed in a rounded rectangle on a light gray background. It features five dropdown menus for "SELECT MOVIE:", "SELECT THEATRE:", "SELECT DATE:", and "SELECT TIME:", with values "Finding Nemo", "Galaxy cinemas", "04-11-2025", and "11:00AM" respectively. Below these is a text input field for "NO OF TICKETS:". A "SUBMIT" button is positioned at the bottom of the form. The browser window includes standard OS controls in the top right corner.

MOVIE DETAILS

SELECT MOVIE: Finding Nemo

SELECT THEATRE: Galaxy cinemas

SELECT DATE: 04-11-2025

SELECT TIME: 11:00AM

NO OF TICKETS:

SUBMIT

الشكل 3: واجهة اختيار الفيلم وتفاصيل العرض (Movie and Show Details Selection Interface)

BOOKING DETAILS

NO OF TICKETS:	3
THEATRE:	Galaxy cinemas
MOVIE:	Finding Nemo
DATE:	04-11-2025
SHOW:	11:00AM
FARE:	300.0

BOOK **CANCEL**

الشكل 4: واجهة تفاصيل الحجز وتأكيد (Booking Details and Confirmation Interface)

CONFORM BOOKING

NO OF TICKETS:	3	THEATER:	Galaxy cinemas
MOVIE:	Finding Nemo	DATE:	04-11-2025
FARE:	300.0	TIME:	11:00AM

CARD DETAILS

Card number

Card holder name

CVV EXP 01 2018

PROCEED

الشكل 5: واجهة إدخال تفاصيل الدفع (Payment Details Entry Interface)

The screenshot shows a web application window titled "CONFIRM DETAILS" in red. Below the title, there are two columns of information: "NO OF TICKETS: 3", "THEATER: Galaxy cinemas", "MOVIE: Finding Nemo", "DATE: 04-11-2025", "FARE: 300.0", and "TIME: 11:00AM". Below this, there is a section titled "PAYMENT TYPE" in blue. It contains four radio button options: "Credit card", "Debit card", "NET BANKING", and "PAYTM WALLET". At the bottom, there are two buttons: "Make Payment" and "Go Back".

NO OF TICKETS: 3	THEATER: Galaxy cinemas
MOVIE: Finding Nemo	DATE: 04-11-2025
FARE: 300.0	TIME: 11:00AM

PAYMENT TYPE

☐ Credit card
☐ Debit card
☐ NET BANKING
☐ PAYTM WALLET

Make Payment **Go Back**

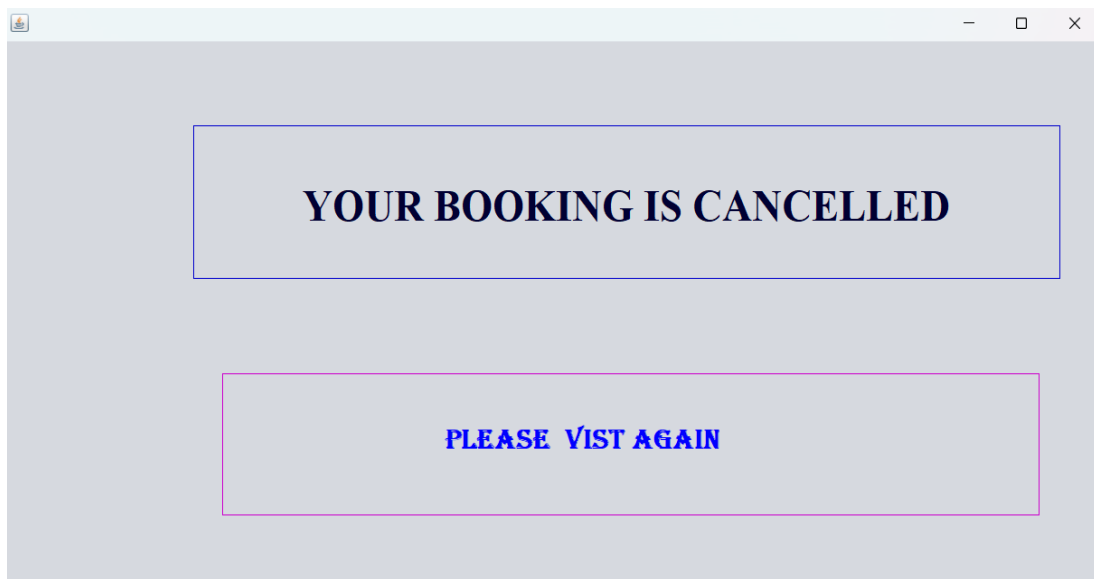
الشكل 6: واجهة اختيار نوع الدفع (Payment Type Selection Interface)

The screenshot shows a web application window titled "RECEIPT" in blue. It contains a table with the following information: "NO OF TICKETS: 3", "THEATRE: Galaxy cinemas", "MOVIE: 04-11-2025", "DATE: 11:00AM", "SHOW: Finding Nemo", and "FARE: 300.0". Below the table, there is a green text message: "THANKS FOR BOOKING".

NO OF TICKETS:	3
THEATRE:	Galaxy cinemas
MOVIE:	04-11-2025
DATE:	11:00AM
SHOW:	Finding Nemo
FARE:	300.0

THANKS FOR BOOKING

الشكل 7: واجهة الإيصال النهائي (Final Receipt Interface)



الشكل 8: واجهة تأكيد الإلغاء (Booking Cancellation Confirmation Interface)

3. شرح لكيفية تنفيذ كل موضوع مطلوب :

1) معالجة الاستثناءات (Exception Handling) :

استخدم المشروع تقنية معالجة الاستثناءات بفعالية لضمان عدم توقف التطبيق عند حدوث أي أخطاء. في كلاس movie، تم احتواء العمليات الحساسة داخل كتل try-catch-finally، وخصوصًا أثناء تنفيذ الحجز والتعامل مع قاعدة البيانات. في receipt، ظهرت هذه التقنية بشكل أكثر تقدمًا داخل دالة done() التابعة لـ SwingWorker، حيث يتم التعامل مع النتائج النهائية للعمليات الخلفية وعرض رسائل واضحة للمستخدم عند نجاح الحجز أو فشله، مع إعادة تمكين الأزرار بغض النظر عن النتيجة. كما استخدمت في دالة Button2ActionPerformed() أثناء تنفيذ إلغاء الحجز، حيث يتم تنفيذ استعلامات حذف وتحديث تتطلب حماية كاملة عبر try-catch، ويتم تطبيق rollback() لتفادي تلف البيانات إذا حدث خطأ و في register، نجدها أثناء تنفيذ عملية إدخال المستخدم الجديد في قاعدة البيانات، لحماية العملية من الفشل غير المتوقع. كذلك، في كل من register و payment، تم استخدام catch داخل دالة main() للتحكم بالأخطاء الناتجة من تحميل واجهة المستخدم الرسومية بنظام Nimbus .

(2) الاتصال بقاعدة البيانات (JDBC) :

يعتمد المشروع بشكل أساسي على JDBC للاتصال بقاعدة بيانات MySQL ، وقد تم تصميم كلاس خاص لذلك باسم javaApplication4 ليحتوي على دالة (ConnecrDb) التي تنشئ الاتصال وتستخدم من جميع الكلاسات مثل movie, recepit, و register. في كلاس movie، يُستخدم JDBC لاستعلام جداول movies_inventory و bookings من أجل تعبئة عناصر الواجهة، وكذلك لتنفيذ عمليات الحجز. أما في كلاس recepit، فيتم تنفيذ سلسلة من العمليات المتقدمة مثل جلب بيانات الحجز، حذف السجل من جدول bookings وإرجاع التذاكر إلى جدول movies_inventory وكل ذلك داخل معاملة واحدة يتم التحكم بها من خلال setAutoCommit(false) و commit() لضمان الاتساق وأما register، فيقوم بتنفيذ استعلام INSERT لتسجيل بيانات المستخدم الجديد باستخدام Statement .

(3) واجهة المستخدم الرسومية ومعالجة الأحداث (GUI + Event Handling) :

تم تصميم واجهة المشروع بالكامل باستخدام مكتبة Swing في Java ، مع تخصيص كل نافذة لتقوم بوظيفة معينة بطريقة منظمة وسلسة. كلاس login يحتوي على نافذة بسيطة لإدخال بيانات المستخدم، تشمل حقول إدخال وكلمة مرور وزر تسجيل الدخول. كلاس movie يُعد الأكثر تفاعلية، حيث يعرض قائمة بالأفلام والمواعيد باستخدام ComboBox، ويتعامل مع اختيار المستخدم للتوقيت والفيلم لعرض التفاصيل المناسبة. في payment، يتم إدخال بيانات البطاقة البنكية من خلال عناصر مثل JTextField, PasswordField، و ComboBox، ويتم تنفيذ الحدث عند الضغط على زر "proceed" بنقل المستخدم إلى واجهة إكمال الحجز recepit. يعرض تأكيد الحجز ويحتوي على أزرار للتنقل أو الإلغاء، ويستخدم myStatusLabel لعرض حالة تنفيذ العملية ديناميكياً. أما register، فيوفر واجهة تسجيل متكاملة تشمل الاسم، الإيميل، رقم الهاتف، وكلمة المرور، مع زر تسجيل لتنفيذ العملية.

(4) إطار عمل المجموعات (Collections Framework) :

في كلاس movie، تم استخدام بنية Vector<String> لتخزين البيانات المستخرجة من قاعدة البيانات مثل أسماء الأفلام، التوقيتات، والمسارح، ثم تعبئتها داخل عناصر JComboBox بشكل ديناميكي. هذا الاستخدام يُظهر كيف تم الاعتماد على إطار المجموعات لجعل الواجهة تفاعلية وقابلة للتحديث المباشر بناءً على بيانات قاعدة البيانات.

(5) تيارات الإدخال والإخراج (I/O Streams) :

ضمن كلاس register، تم استخدام تقنية Logger لتوثيق الأحداث والتفاعلات داخل البرنامج

بشكل احترافي. حيث تم تهيئة ConsoleHandler و FileHandler لتسجيل الرسائل البرمجية من نوع INFO، وحفظها في ملف مخصص داخل المسار C:/temp/application_test.log. هذه العملية تتيح تتبع الأداء العام للتطبيق، والكشف المبكر عن أي أخطاء أو استثناءات قد تحدث أثناء تشغيل الواجهة أو تنفيذ الاستعلامات، مما يساهم في تحسين صيانة البرنامج وتطويره مستقبلاً.

(6) تعدد الخيوط (Multithreading)

تم تطبيق تعدد الخيوط باحترافية داخل كلاس receipt باستخدام SwingWorker عندما يضغط المستخدم زر التأكيد للحجز، يتم تنفيذ العملية في الخلفية دون تجميد واجهة المستخدم، بينما يتم عرض رسالة "جاري المعالجة..." داخل myStatusLabel بعد اكتمال المعالجة، يتم التفاعل مع المستخدم من خلال دالة done() التي تُظهر رسالة نجاح أو فشل، وتعيد عرض واجهة الإيصال أو تنقله إلى النافذة التالية و هذا يضمن تجربة مستخدم سلسة دون انتظار أو توقف واجهة البرنامج.

8. توزيع المهام بين أعضاء الفريق:

تم إنجاز هذا المشروع من قبل فريق عمل متكامل، حيث تم تقسيم الأدوار الفنية والبرمجية، على النحو التالي:

- صفاء محمد القضااض
قامت بتطوير كامل عملية تسجيل الدخول، من تصميم واجهة المستخدم إلى تنفيذ التحقق من صحة البيانات وربطها بقواعد البيانات، مع الاهتمام بإظهار رسائل الخطأ والتنبيهات .
- هداية عبد الناصر بنيس
تولت مهمة تصميم وتنفيذ واجهة تسجيل مستخدم جديد، بما في ذلك إعداد حقول الإدخال ، بالإضافة إلى تنفيذ استعلامات الإدخال الخاصة بقاعدة البيانات وتوثيق العملية ضمن الكود البرمجي.

▪ سلمى حمدي الصغير

تولّت مسؤولية تطوير وظيفة حجز التذاكر، حيث قامت ببناء الواجهة الخاصة بعرض العروض والتوقيات، وإعداد الاستعلامات الخاصة بالتحديث والإدراج في قاعدة البيانات بما يضمن دقة عملية الحجز.

▪ ندى يوسف خليفة

نفّذت ميزة إلغاء الحجز من خلال تطوير واجهة مخصصة لذلك، بالإضافة إلى إعداد العمليات الخلفية التي تتضمن حذف البيانات المستهدفة وتحديث حالة المخزون، مع ضمان أمان المعاملة عبر التعامل الصحيح مع قاعدة البيانات.

وبالإضافة إلى المهام المذكورة أعلاه، تم تنفيذ بقية الواجهات والوظائف البرمجية الأخرى كعمل جماعي مشترك بين أعضاء الفريق، بما في ذلك التنسيق بين النوافذ، إعداد التصميم العام، وتكامل البيانات بين الأجزاء المختلفة لضمان تقديم مشروع متكامل .

9. التحديات والدروس المستفادة:

• من التحديات استخدام SwingWorker لأول مرة:

تم تطبيق SwingWorker لتجنب تجميد الواجهة أثناء تنفيذ الحجز ، لكن كانت هناك صعوبة في استيعاب تدفق doInBackground ثم done()، خاصة في إعادة تمكين الأزرار وتحديث حالة الحجز.

• ومن التحديات ضبط المعاملة (Transactions) في الإلغاء:

تنفيذ عملية commit() و rollback() داخل إلغاء الحجز تطلب دقة كبيرة في ترتيب الاستعلامات والتأكد من أن كل خطوة تنفذ بشكل صحيح، وإلا قد تُلغى التذاكر دون أن تُعاد فعليًا إلى المخزون.

• ومن التحديات نقل البيانات بين الكلاسات:

تم تمرير البيانات بين النوافذ باستخدام Constructors، ولكن التعامل مع أنواع مختلفة (مثل String و double و int) بشكل منظم كان تحديًا في أول الأمر، خاصة في التنقل بين payment و receipt.

- ومن الدروس المستفادة هو ان التوثيق في وقت التنفيذ لا يقل أهمية عن الكود نفسه: حيث ان اعتماد Logger ساعد لاحقاً في معرفة نقاط الفشل بسرعة دون الحاجة لطباعة في الكونسول فقط، وأعطى المشروع شكلاً احترافياً أكثر
- ابرز تحدي : تنفيذ المشروع كان تحت ضغط كبير بسبب ضيق الوقت، وتزامنه مع امتحانات ومشاريع أخرى في مواد مختلفة، مما جعل تنظيم الوقت والتفرغ الجزئي للعمل البرمجي تحدياً حقيقياً.