



الجمهورية العربية السورية
جامعة دمشق - كلية الهندسة المعلوماتية
قسم هندسة البرمجيات ونظم المعلومات

تقرير مشروع مادة البرمجة 3 Programming 3 project report

إعداد الطلاب:

ملهم محمد لؤي زكريا

محمد نور علي دياب

ملهم أحمد حمدان

مشرف العملي: م. ميساء العبودي

الأسستاذ المشرف: م. ليال جديد		رقم الفئة: 12	
نظامي أو إعادة عملي أو إعادة عملي ملتزم بالحضور	السنة	اسم الأب	الاسم والكنية
نظامي	الثانية	علي	محمد نور دياب
نظامي	الثانية	محمد لؤي	ملهم زكريا
نظامي	الثانية	أحمد	ملهم حمدان



Scan the QR code to view
the project on GitHub

User journey

يبدأ التطبيق بواجهة لتسجيل الدخول أو لإنشاء حساب جديد للزبون، مع مراعاة صلاحية المعلومات المقدمة لإنشاء الحساب وعدم وجود حساب مسبق بنفس الـ email. ويحوي التطبيق على 3 واجهات أساسية مختلفة كل منها تخدم الشخص الذي يسجل دخوله عن طريق واجهة الـ log in أو توماتيكياً حسب نوع الحساب وصلاحياته، وهي:

1. واجهة الزبون:

تظهر لديه كافة الوجبات الموجودة في المطعم مع مكوناتها وأسعارها، ويختار العدد الذي يريده من كل وجبة وإضافته للسلة ثم يختار نوع الطلب (Dine-in, Delivery, Special) ثم تظهر له واجهة بالفاتورة يضع فيها قيمة الـ tip الذي يود تركه، ثم بإمكانه تثبيت الطلب أو العودة وتعديله. وبإمكان الزبون العودة للطلبات التي طلبها مسبقاً عن طريق زر يظهر له جميع فواتير الطلبات السابقة مع حالاتها (Pending, Preparing, Ready, Done, Deleted).

2. واجهة الموظف:

ويكون فيها عرض جميع الطلبات غير الجاهزة ذات الحالة (Pending, Preparing, Ready) وبإمكانه تغيير حالة كل طلب بناءً على مستجدات الطلب إلى (Preparing, Ready, Done) أو حذف الطلب إذا طلب الزبون ذلك (Deleted).

3. واجهة المدير:

يستطيع فيها إدارة الوجبات من إضافة وحذف وتعديل الوجبات الموجودة في المطعم، وبإمكانه عرض الإحصائيات التي يحتاجها من مثل:

- اسم الزبون الدائم على المطعم.
- الوجبة الأكثر طلباً في المطعم.
- العائدات اليومية: كل يوم مع عائداته في ذلك اليوم.
- عدد الطلبات اليومية: كل يوم مع عدد الطلبات في ذلك اليوم.

والمدير هو الوحيد الذي بإمكانه إنشاء حسابات للموظفين، طبعاً مع التأكد من صلاحية المعلومات وعدم التكرار.

Project implementation

في تنفيذ المشروع تم الاعتماد على تخزين المعلومات بشكل كامل في ملفات نصية وقراءتها وكتابتها وتعديلها عن طريق الـ BufferedWriter والـ BufferedReader مع مراعاة كافة الاستثناءات التي يمكن أن يتم مواجهتها (IOException).

ويتم معالجة المعلومات عن طريق بني معطيات هي المجموعات (Collections) مثل الـ ArrayList, HashMap واستخدام بعض الطرائق الموجودة فيها لحالات مثل الإضافة والحذف والوصول والترتيب وإيجاد أكبر وأصغر قيمة وغيرها.

ولبلورة فكرة الـ multithreading تم استخدامها في إرسال الإشعارات المناسبة للتأكيد نجاح بعض العمليات للزبون (حين الدخول وإنشاء حساب جديد وتسجيل طلب جديد) والموظف (في حال تحديث حالة الطلب) والمدير (في حال إضافة أو تعديل أو حذف وجبة وفي حال إنشاء حساب موظف)، و تم استخدامها في الكتابة على الملفات النصية، بحيث حين ورود أي معلومة يتم معالجتها ثم إنشاء وظيفة لمعالجتها على الملفات النصية (إضافة أو تعديل أو حذف) وتحديثها في الواجهات وإرسال الإشعارات المناسبة لكل مستخدم على حدة، مع العلم أنه تم مراعاة الـ Thread Safety لضمان عدم حصول مشاكل حين الكتابة على ذات الملفات بذات الوقت وتم التأكد من أن كل الداتا المستخدمة ضمن البرنامج لضمان استقرار البرنامج وعدم إدخال داتا خاطئة ومعالجة كل الاستثناءات الممكنة عن طريق الاستثناءات المعرفة مسبقاً في اللغة (IOException, NumberFormatException) وعن طريق بعض الاستثناءات يلي تم تعريفها واستخدامها في قسم الـ backend logic.

ولإنشاء الواجهات تم استخدام swing والاستفادة من كل المكونات المعرفة في مكتبة swing وإنشاء method خاصة للربط المنطقي مع ال backend logic وأيضاً استخدمنا بعض المكونات الموجودة في ال awt مثل كلاسات Font , File chooser color , وتم إنشاء method مساعدة في كل كلاس في ال GUIs التي تساعد على إنشاء custom button and custom panelMeal وأيضاً تم إنشاء frame أساسي يحوي logo المطعم ويحوي على setPanel وهي تمثل الواجهات المختلفة عبر التطبيق وكل panel تم استدعاؤها على SwingUtilities.invokeLater() لضمان عرض المكونات بشكل متزامن وتحميلهم بشكل سليم على كل واجهة وتم استخدام الكثير من المكونات مثل التنوع في استخدام الخطوط، التنوع في استخدام الالوان وتم استخدام الثوابت في المكتبة swing مثل center وغيرها.

Project Structure

المشروع مكوّن من شقين:

1. BackendLogic

ويحوي كل الطرائق التي تعالج البيانات وتتعامل مع الملفات النصية، وتم تقسيمها بالpackages:

a. Users:

وتحوي كل ما يتعلق بحسابات المستخدمين من زبائن وموظفين بالإضافة إلى جميع العمليات الممكنة على معلوماتهم من إضافة وحذف وعن طريقها يمكن معرفة الزبون الدائم على المطعم من خلال مقارنة عدد الطلبات السابقة لكل زبون.

b. Meals:

حيث تعمل على إضافة وتعديل وحذف الوجبات لمدير المطعم وفيها طرائق ترجع الوجبة الأكثر طلباً في المطعم بناءً على عداد الطلبات الموجود لكل وجبة.

c. Orders:

تحوي كلاسين مهمين وهم ال OrderItem وال Order حيث يحوي كل order عدداً من ال order items المكونة من الوجبات مع كمياتها وثمانها وعن طريقها يمكن إضافة الطلبات (بالنسبة للزبون) وقراءتها وتعديل حالتها (بالنسبة للموظف) ويمكن منها معرفة العائدات اليومية وعدد الطلبات في كل يوم عن طريق الرجوع للملفات وتصنيف كل طلب حسب تاريخه.

d. Notifications:

وفيها ال Notification thread والتي يتم إرسالها للزبون أو الموظف أو المدير كما تم الحديث عنها مسبقاً.

e. Files:

وفيها ال addToFile thread والتي يتم فيها الإضافة والكتابة على الملفات، حيث بناءً على الكائن الممرر له يتم الكتابة على الملف المناسب المتعلق بهذا الكائن.

2. GUIs

ويحوي كل الواجهات المستخدمة في التطبيق والمبنية عن طريق ال swing، وتم تقسيمها بالpackages:

a. mainframe:

فيه ال frame الأساسي الذي يتم من خلاله يتم التنقل في واجهات التطبيق كله.

b. loginInterface:

الواجهات التي يسجل فيها المستخدم دخوله أو ينشئ فيها الزبون حساباً جديداً له.

c. employeeInterface:

وفيها يتم إظهار جميع الطلبات للموظف مع إمكانية تعديل حالة كل طلب أو حذفه.

d. managerInterface:

وتحوي الواجهات الخاصة بالمدير وهي واجهة إنشاء حساب لموظف وواجهة إدارة الوجبات وواجهة لإظهار الإحصائيات حسب المطلوب.

e. orderInterface:

وفيها الواجهات الخاصة بالزبون لإضافة الطلبات ومراجعة ومعرفة حالة الطلبات السابقة.