



دولة ليبيا
جامعة طرابلس
كلية تقنية المعلومات



أنماط تصميم البرمجيات [ITSE424]

نظام إدارة المخازن

عمل الطلبة :-

216185053

فرج نوري شعيب

217185215

معتز صلاح خلف الله

216181126

معاذ عيسي عبد الرحمن

216180534

محمد إبراهيم الصول

القسم :-

هندسة البرمجيات

تحت إشراف الدكتورة :-

مي مفتاح البعباع

كما هو المعتاد لدي الجميع فان إدارة المخازن ما هي الا عملية تتم بها تنظيم الأدوار و البضائع من حيث الكميات و المواد المخزنة و إدارة مكان التخزين و تشمل مراقبة البضائع و التنظيم و التخطيط الفعال لإدارتها بكفاءة و فعالية كما تتضمن الاشراف عن عمليات التخزين و فحصها و تسجيلها علي مستوى عالي من الأداء و اجراء عمليات الصيانة لمكونات النظام المادية و البرمجية باقل تكلفة ممكنة كما تهدف لكافة عمليات التخطيط لتحديد الأهداف المرجوة إنجازها و الطريقة التي يتم بها ذلك ها هنا ، هي عن طريق نظام إدارة المخازن .

يساهم نظام إدارة المخازن في نجاح أو فشل العديد من الشركات ، حيث يعد المخزن وسيطاً في سلسلة التوريد و الذي يؤثر علي كل من تكاليف الخدمة و سلسلة المنتجات ، كما تعتبر عملية إدارة المخازن بفاعلية عملية ليست بسيطة ، فالتخطيط و اتخاذ القرار و تنظيم المخزن و المراقبة لكل المنتجات و التحكم في التخزين و التوزيع كلها عوائق و تحديات تواجه النظام و ترتبط بإدارة الشركة و سوق العمل يهدف نظام إدارة المخازن بكل بساطة إلي تحديد و تقسيم الأدوار لكل من الموظفين و المدراء علي حد سواء كل حسب توصيفه الخاص لأداء وظائف معينة بحيث يكون بطبيعة الحال للمدراء بعض المزايا و الخصائص التي تميزهم عن الموظفين منها تعيين موظفين جدد و إضافة فئات جديدة كما يتيح له إضافة منتجات جديدة و أيضا عرض تاريخ ووقت ونوع العملية و ماهية العملية التي تمت علي المنتجات.

• Java FX

تم استخدامها لبناء الواجهات التفاعلية القابلة للاستخدام من الواجهات النهائية للمستخدم ليستطيع التفاعل مع النظام حيث لا يعلم ما الذي يحدث من عمليات في الخلفية .

• IntelliJ IDE

وهي بيئة متكاملة لتطوير اللغة جافا معترف به كواحد من أفضل أدوات تطوير جافا في الصناعة

<https://www.jetbrains.com/idea/>

• MySQL database

نظراً لما تملكه من مزايا السرعة و الثبات ، مما نتج عنه نظام يبرز بدون التضحية بالاعتمادية و التكاملية .

<https://www.mysql.com/>

· CSS

نظراً لأنها توفر التخطيط والألوان والخطوط؛ وهذا يمكن أن يساعد في تحسين قابلية الوصول ويوفر مرونةً في التحكم بخصائص العرض وطريقة تنسيقها اعتماداً على المستخدم.

تساعد CSS أيضاً في تسهيل التعديلات، إذ يمكن تعديل ملف CSS واحد للتأثير على العديد من الواجهات في النظام.

· FXML

FXML هي لغة تمييز لواجهة مستخدم تعتمد على XML تم إنشاؤها بواسطة Oracle Corporation لتعريف واجهة المستخدم الخاصة بتطبيق JavaFX. تقدم شركة FXML بديلاً لتصميم واجهات المستخدم باستخدام التعليمات البرمجية الإجرائية ، وتسمح باستخلاص تصميم البرنامج من منطق البرنامج.

• Github

هو خدمة استضافة "مستودع" مفتوحة المصدر، تستخدم في مجموعة متنوعة من لغات البرمجة المختلفة، وتتابع التغييرات المختلفة التي تم إجراؤها على كل تكرار. الخدمة قادرة على القيام بذلك عن طريق استخدام، git: وهو نظام التحكم في المراجعة الذي يتم تشغيله في واجهة الأوامر.

يؤدي استخدام GitHub إلى تسهيل التعاون مع الزملاء والأقران والنظر إلى الإصدارات السابقة من عملك، إذا لم تكن تستخدمه لمشاريع التشفير الخاصة بك من قبل.

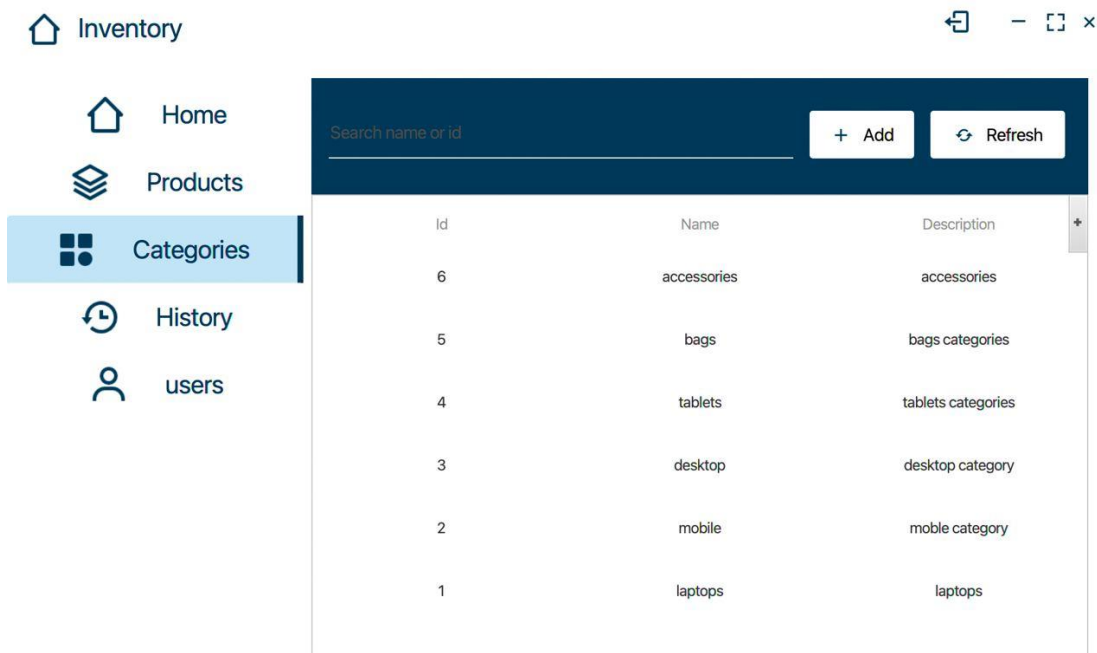
Results

4- النتائج

بالنسبة لمدير النظام فان المزايا التي التي يملكها موضحة بالصور كما تم ذكرها سابقا و توضيح تفصيلها :-



شكل " 1.4 "



شكل " 2.4 "

Inventory

Home

Products

Categories

History

users

Search name or id

Refresh

Id	Name	Price (N)	description	Category	Event Type	Date
1	test	999.0	test	laptops	added	13 July 2022 1...
2	test	2345.0	dfbnod	laptops	added	16 July 2022 1...
3	test	999.0	test	laptops	purchased	16 July 2022 1...
4	test	2345.0	dfbnod	laptops	deleted	16 July 2022 1...
5	M LÔÃ	3343.0	لٲٲ	laptops	added	16 July 2022 1...
6	M LÔÃ	3343.0	لٲٲ	laptops	deleted	17 July 2022 1...
7	test	13.0	blaaaa	mobile	added	18 July 2022 ...

شكل " 3.4 "

Inventory

Home

Products

Categories

History

users

Search name or id

+ Add

Refresh

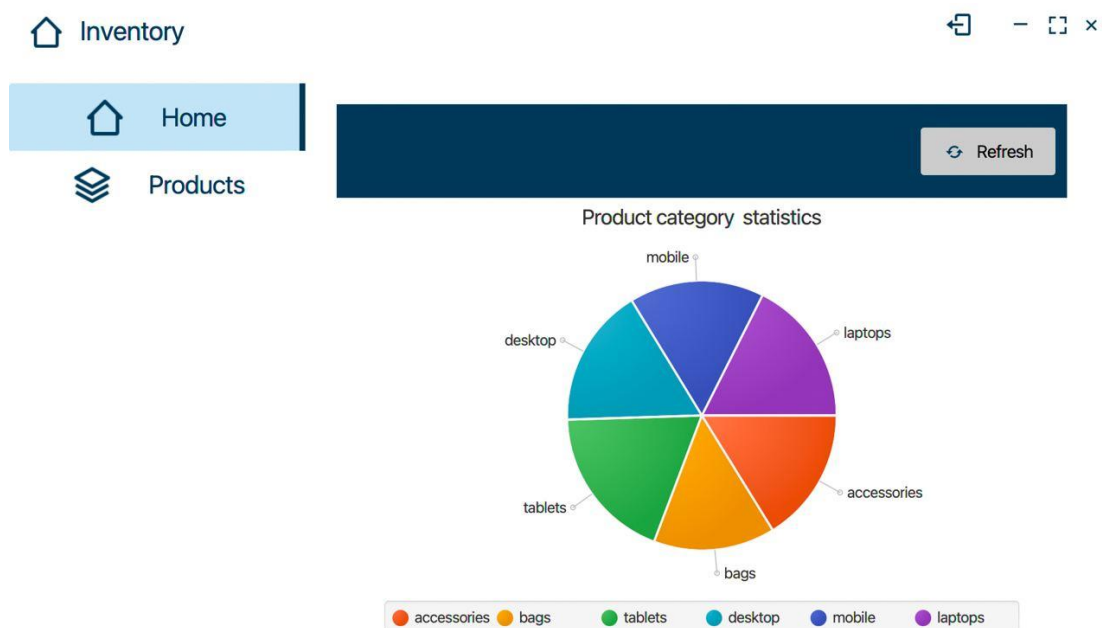
Id	First Name	Last Name	Email	Phone number	Gender	role
6	moad	N	moad@gmail.c...	Male	0	employee
5	testbla	testbla	testbla@testbl...	Male	0	admin
4	test2345	test2345	faraj@bla.com	0	Male	
3	test	test	test@gmail.com	Male	0	employee
2	faraj2	shuaib	farajshuaib@g...	Male	0	employee
1	faraj	shuaib	faraj.shuaip97...	Male	0	admin

شكل " 4.4 "

اما التالي فهي من الواجهات العرض للمستخدم بعد تسجيل الدخول:-



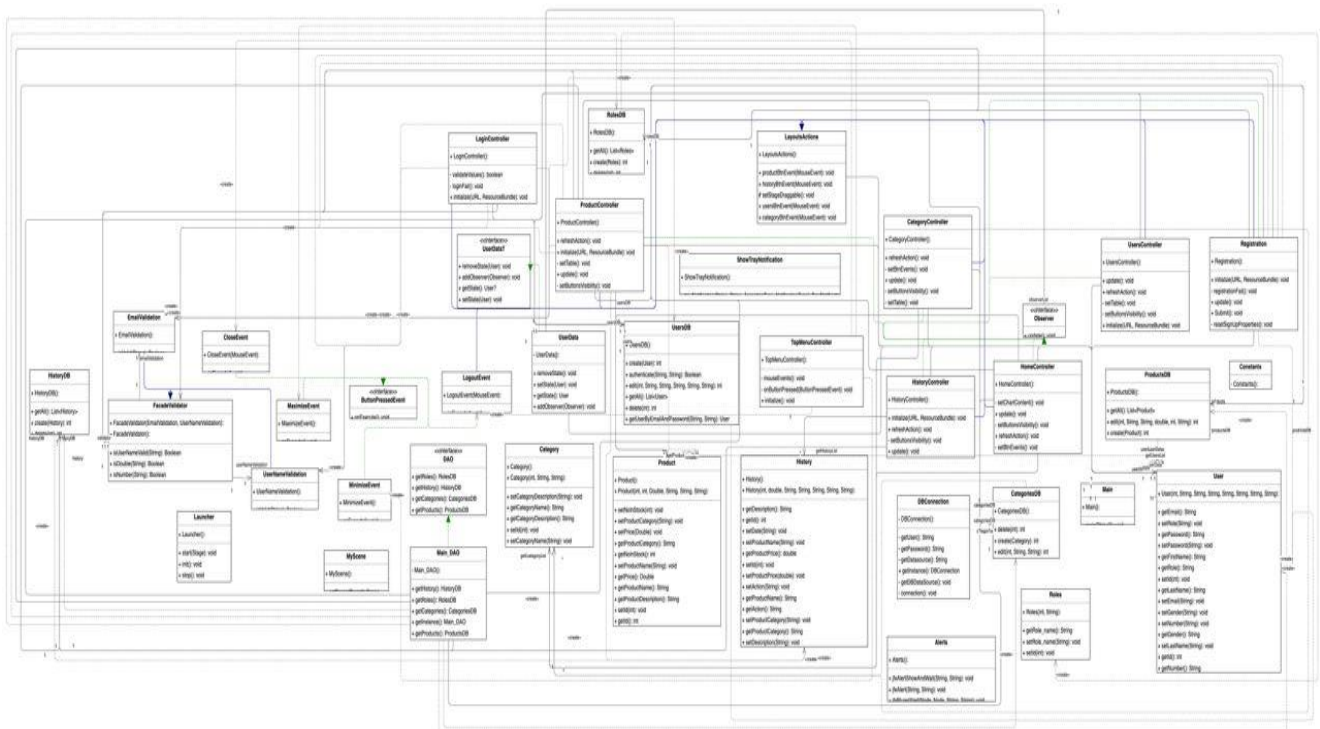
شكل " 5.4 "



شكل " 6.4 "

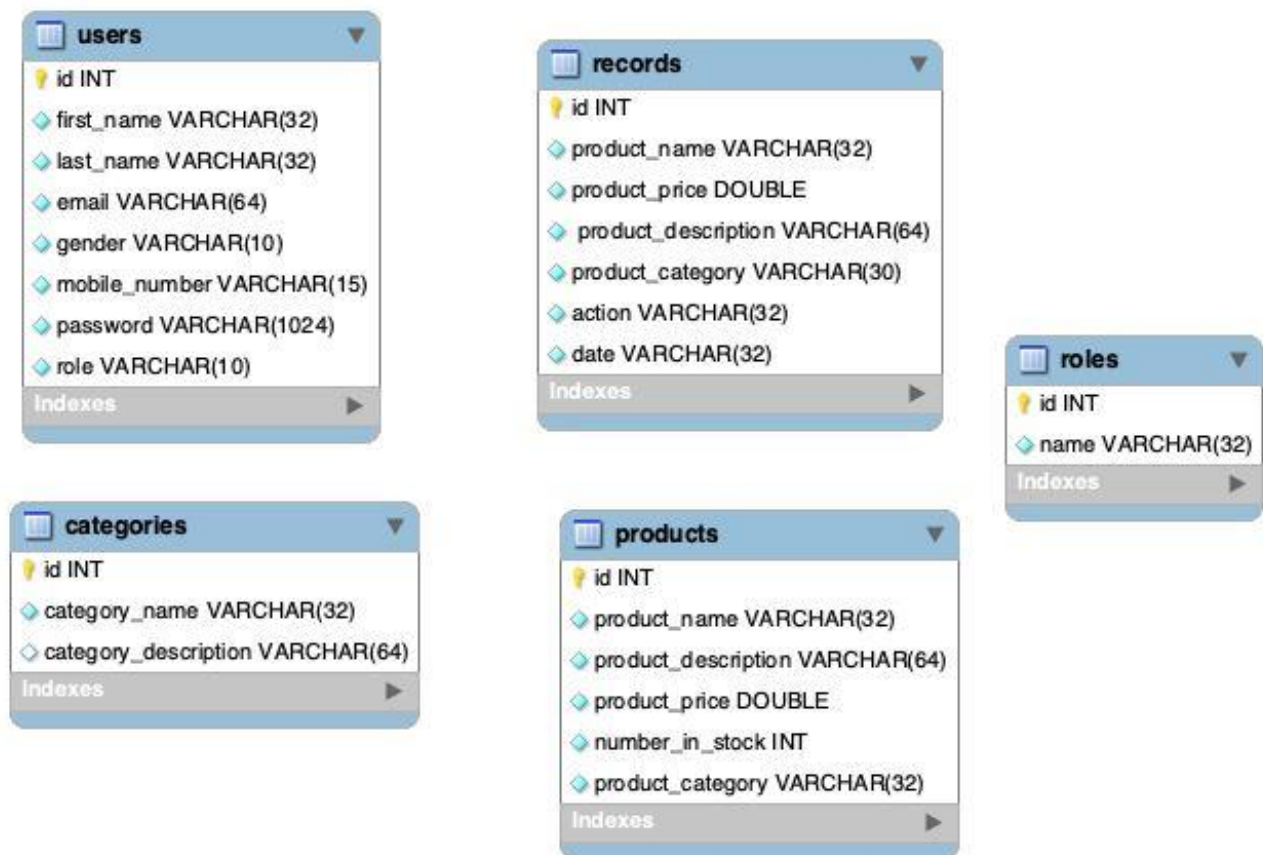
شکل " 7.4 "

كما يوضح مخطط الفئة التالي كل العلاقات بين الفئات :-



شکل " 8.4 "

يستخدم النظام قاعدة بيانات من نظام MySQL كما تم توضيحه مسبقاً، تحتوي على بيانات المستخدم ، التصنيفات ، البضائع وغيرها ، جميعها موضحة في نموذج ERD التالي :-



شكل " 9.4 "

أما بالنسبة للأنماط المستخدمة فإننا استخدمنا الأنماط التالية :

1.Singleton:

النمط (Singleton) يمكننا من تكوين اتصال مع قاعدة البيانات مرة واحدة دون الحاجة لإنشاء أكثر من كائن من (class) لإنشاء قاعدة البيانات .

2. Decorator

توجد عدة فئات (classes) تتطلب التعامل مع قاعدة البيانات, ولهذا استخدمنا النمط (Decorator) لتنسيق كل تعاملات مع أي controller مع Database Access Object عبر فئة (class) واحد.

3. Facade

في عمليات الإدخال تحدث عدة أخطاء في عملية إدخال المستخدم للبيانات، مما يسبب خطأ في تنسيق البيانات في قواعد البيانات، لهذا يجب التحقق من صحة البيانات أثناء إدخال المستخدم للبيانات. يوجد عدة نماذج إدخال لإدخال البيانات وكل نموذج منها يحتاج للتحقق (validation) خاص به ولهذا صممنا فئة (class) يقوم بهذه العملية، وبالتالي ينتج عن هذه العملية (class) ذو حجم كبير يصعب التعديل

عليه (class explosion)، لذلك قد تم استخدام النمط (facade) لتقسيم الفئات والوصول اليهم عبر فئة واحدة من قبل الفئة الرئيسية (main class)، والنتيجة هي الحصول على فئة واحدة قابلة للتطوير والتعديل عليها بسلاسة.

4. Observable

تم إنشاء فئة (class) تختص بالبيانات التي يقوم بإدخالها المستخدم إلى النظام ويقوم بمتابعة كل فئة (class) تعمل هذه الفئة على تطبيق فئة (observer interface) و تم بناءها أو استدعائها بحيث تكون مبنية على بيانات المستخدم و كل فئة تتضمن (observer interface) بحيث تعمل علي تضمين دالة تقوم بإعادة تعيين خاصية userData وعن طريق استدعائها من قبل (UserData class) لذلك فإنه عبر القيام بتنفيذ حلقة تقوم بتنفيذ دالة (update) فإن كل فئة تقوم بتضمين (observer interface) لأنه و بطريقة ما تعتبر مبنية عليها .

5. Command

يمكنك هذا النمط من التقليل من إنشاء الدوال، حيث يمكنك إنشاء دالة واحدة تقوم بتنفيذ أحداث مختلفة حسب الكائن الممر للدالة في وقت التنفيذ. الدالة (`onButtonPressed`) تستقبل كائنات مختلفة، يتم تنفيذ إجراءات مختلفة حسب كل كائن عن طريق استدعاء الدالة (`onEvent`) التي تطبقها كل فئة الكائنات من (`ButtonPressedEvent`).

تتضمن حدود النظام في مدى تطبيق الأنماط المقترحة بكفاءة أعلى و علي مستوي أفضل من المهارة و الدقة و لأنه قد تم الاعتماد على لغة البرمجة (java FX) فإنه وبالتالي يوجد بعض القصور من حيث إمكانيات اللغة من تطبيق بعض الخواص أو الدوال و يرجع ذلك لأنها لا زالت تحتاج الكثير من التطوير أو كما يصفها خبراء اللغة (إنها لغة غير ناضجة) و ان هناك لغات برمجة أخرى أكثر كفاءة لأنها تركز على جزئية الواجهات اكثر من الهيكلية بحد ذاتها و يعتبر هذا هو الأهم و لأن العديد من اللغات الأخرى تركز علي جزئية الغاء القصور و سد الثغرات ، ثم ان تطبيق نظام ادارة المخازن علي مستوي اعلي من الاعتمادية و المرونة و تحقيق كافة خصائص النظام فانه يحتاج الكثير من الوقت للتحليل الجيد و التطوير المستمر بالإضافة الي الاختبار الجيد المصاحب للتوثيق لكل الجزئيات التي تم تطويرها او تعرضت لأي تعديل و نظرا للوقت المعطي و المهارات التي سبق لنا و ان درسناها فإنه كان من الصعب تطوير نظام كامل يكون فيه فعال علي مستوي المؤسسات بالمواد و المصادر المتوفرة لدينا في الوقت الحالي .

- Dr. Hassan Ibrahim (visual programming)
- GitHub
- Java FX 9 by example
- Head First Design Patterns: A Brain-Friendly Guide 1st Edition
-
- Mastering JavaFX 10: Build advanced and visually stunning Java applications
- Formal IT students

- Eric freeman : Head First Design Patterns: A Brain-Friendly Guide 1st Edition .
- Sergey Grinev : Mastering JavaFX 10: Build advanced and visually stunning Java applications.
- Dr.Hassan ibrahim
- Formal IT students Gudins
- ITSE423
- For UI Gudins : Dribbble
- Geeks for Geeks

Table of content

جدول المحتويات

رقم الصفحة	العنوان	ر.م
2	Abstract	1
3	Introduction	2
4	Materials and Methods	3
7	Results	4
15	Limitations	5
16	Acknowledgments	6
17	References	7

Table of Figures

جدول الاشكال

رقم الصفحة	العنوان	ر.م
7	شكل " 1.4 "	1
7	شكل " 2.4 "	2
8	شكل " 3.4 "	3
8	شكل " 4.4 "	4
9	شكل " 5.4 "	5
9	شكل " 6.4 "	6
10	شكل " 7.4 "	7
10	شكل " 8.4 "	8
11	شكل " 9.4 "	9