**جامعة تشرين**

**كلية الهندسة المعلوماتية**

**السنة الرابعة**

**قسم البرمجيات ونظم المعلومات**

**تصميم موقع ويب لبيع وشراء الأغراض المستعملة**

**مشروع فصلي**

**إعداد الطلاب**

مايا أحمد ناصر احمد غسان تركمان

**إشراف**

د. باسل حسن

العام الدراسي 2021-2022

المحتويات

[فهرس الأشكال: 4](#_Toc111579222)

[فهرس الجداول: 5](#_Toc111579223)

[مقدمة: 6](#_Toc111579224)

[1. توصيف النظام: 7](#_Toc111579225)

[1.1 متطلبات النظام: 8](#_Toc111579226)

[1.1.1 المتطلبات الوظيفية: 8](#_Toc111579227)

[2.1.1 المتطلبات غير الوظيفية: 9](#_Toc111579228)

[2.1 مخططات UML: 10](#_Toc111579229)

[1.2.1 مخطط حالة الاستخدام Use Case Diagram: 10](#_Toc111579230)

[2.2.1 مخططات الحالة state machine diagrams: 14](#_Toc111579231)

[3.2.1 مخطط الصفوف class diagram: 15](#_Toc111579232)

[2. التقنيات والبرامج المستخدمة: 16](#_Toc111579233)

[1.2 تقنيات الـ Frontend: 16](#_Toc111579234)

[1.1.2 لغة HTML: 16](#_Toc111579235)

[2.1.2 CSS 16](#_Toc111579236)

[3.1.2 JavaScript: 16](#_Toc111579237)

[4.1.2 React JS: 17](#_Toc111579238)

[5.1.2 DevExtreme React Grid: 18](#_Toc111579239)

[6.1.2 Visual studio code: 18](#_Toc111579240)

[2.2 تقنيات الـ Backend: 18](#_Toc111579241)

[1.2.2 قاعدة بيانات MySQL: 18](#_Toc111579242)

[2.2.2 Java: 19](#_Toc111579243)

[3.2.2 Spring: 19](#_Toc111579244)

[4.2.2 Spring Boot: 20](#_Toc111579245)

[5.2.2 MySQL Workbench: 20](#_Toc111579246)

[6.2.2 Spring Tool Suit (STS) 20](#_Toc111579247)

[3. تصميم قاعدة البيانات Database Design: 21](#_Toc111579248)

[4. صفحات الموقع الأساسية: 23](#_Toc111579249)

[1.4 الصفحة الرئيسية للموقع Home Page: 23](#_Toc111579250)

[2.4 صفحة جميع المنتجات: 25](#_Toc111579251)

[3.4 صفحة تسجيل حساب أو تسجيل دخول: 26](#_Toc111579252)

[4.4 صفحة إضافة منتج: 28](#_Toc111579253)

[5.4 الصفحة الشخصية للمستخدم profile: 30](#_Toc111579254)

[6.4 صفحة المسؤول: 30](#_Toc111579255)

[7.4 واجهة تتبع طلب الشراء: 31](#_Toc111579256)

[5. الخاتمة: 32](#_Toc111579257)

[6. مقترحات لتطوير المشروع: 33](#_Toc111579258)

[المراجع: 34](#_Toc111579259)

# فهرس الأشكال:

|  |  |
| --- | --- |
| الشكل 1. **مراحل BCrypt** | 9 |
| الشكل 2. **مخطط حالة الاستخدام** | 10 |
| الشكل 3. **مخطط الحالة للطلب** | 14 |
| الشكل 4. **مخطط الحالة للمنتج** | 14 |
| الشكل 5. **مخطط الصفوف** | 15 |
| الشكل 6. **مخطط ERD** | 22 |
| الشكل 7. **الصفحة الرئيسية** | 23 |
| الشكل 8. **فئات المنتجات** | 23 |
| الشكل 9. **المنتج البارز** | 24 |
| الشكل 10. **المنتجات المضافة حديثاً** | 24 |
| الشكل 11. **تفاصيل المنتج** | 25 |
| الشكل 12. **جميع المنتجات** | 25 |
| الشكل 13. **نافذة تسجيل حساب Signup** | 26 |
| الشكل 14. **رسالة الخطأ عند إدخال اسم مستخدم موجود مسبقاً** | 27 |
| الشكل 15. **نافذة تسجيل دخول Login** | 27 |
| الشكل 16. **رسالة الخطأ عند إدخال اسم مستخدم غير موجود مسبقاً** | 28 |
| الشكل 17. **رسالة الخطأ عند إدخال كلمة مرور خاطئة** | 28 |
| الشكل 18. **شريط التنقل للمستخدم المسجل** | 28 |
| الشكل 19. **إضافة منتج** | 29 |
| الشكل 20. **طلب منتج** | 29 |
| الشكل 21. **الصفحة الشخصية** | 30 |
| الشكل 22. **واجهة منتجات المسؤول** | 30 |
| الشكل 23. **واجهة تتبع الطلبات** | 31 |
| الشكل 24. **واجهة تتبع طلب الشراء** | 31 |

# فهرس الجداول:

|  |  |
| --- | --- |
| جدول 1. **توصيف حالة الاستخدام login** | 11 |
| جدول 2. **توصيف حالة الاستخدام add product** | 12 |
| جدول 3. **توصيف حالة الاستخدام purchase** | 13 |

# مقدمة:

مع تقدم الزمن، يتم تكديس المزيد والمزيد من الأغراض المستعملة في المنازل أو المكاتب ولا يمكن إعادة استخدامها بشكل أفضل. لذلك سنقوم بتصميم نظام يمكّن المستخدمين من الاستفادة من هذه الأغراض وإدارتها عن طريق نشر معلومات عنها وبيعها. كذلك يمكن لكل مستخدم التصفح والبحث لشراء ما يحتاج إليه.

ظهرت الحاجة لهذه الفكرة في الآونة الأخيرة بشكل كبير ولا سيما مع الظروف الاقتصادية الصعبة التي يعيشها الناس، حيث لا تعود الفائدة على الشخص الذي يبيع فقط، وإنما على المشتري أيضاً عن طريق شرائه لأغراض بأسعار أقل مما لو اشتراها جديدة.

تم تصميم وبناء المشروع كموقع إلكتروني على الإنترنت، حيث يمكن للمستخدمين عرض السلع المستعملة، وتصنيفها حسب الفئات، كما يمكنهم إضافة السلع الخاصة بهم لبيعها.

الشرط الرئيسي للنظام هو أن يكون قادراً على السماح للمستخدمين بإجراء عمليات البيع والشراء للسلع المستعملة عبر الإنترنت بسهولة وأمان.

# توصيف النظام:

يهدف نظامنا بشكل أساسي إلى إدارة العلاقة مع الزبون وفهم احتياجاته وتحسينها مع الوقت، حيث يستهدف المؤسسات والشركات.

يجب على المستخدمين أن يسجلوا دخولهم إلى النظام من أجل أن يقوموا بوظائف النظام. بعد تسجيل الدخول، يمكن لجميع المستخدمين أن يقوموا بإدارة الزبائن customers. حيث يمكن إضافة زبون جديد أو تعديل زبون أو حذف زبون. يتم تعريف الزبون باسمه وعنوانه المفصّل والرمز البريدي والمدينة والدولة ورقم الهاتف الثابت ورقم الجوال والإيميل. وفي حال كان الزبون شركة، اسم المدير ورقم أعمال الشركة والقوة العاملة (عدد الموظفين) وتاريخ الإنشاء ورقم السجل التجاري بالإضافة إلى الموقع الإلكتروني.

كما تتم إدارة الزبائن المحتملين Leads من إضافة وحذف وتعديل ومراقبة، حيث لديهم نفس معلومات الزبون.

يمكن أن يضيف المستخدم المنتجات الخاصة به إلى النظام. ويقوم كل مستخدم بتحديد خطوات البيع sales steps الخاصة بخطة مبيعاته.

يتيح النظام للمستخدمين إنشاء opportunities وإدارتها، حيث ترتبط كل opportunity بمنتج واحد وزبون واحد. حيث عند إنشاء الـ opportunity تصبح في حالة open، ثم يتم المرور بخطوات البيع إلى حين انتهائها فتصبح في حالة closed، حيث إما أن تكون won أو lost.

كما يمكن أن ينشئ المستخدم مهام ضمن النظام وإدارتها ويقوم بعرض المهام الخاصة به، حيث يقوم بتحديد موعد المهمة وعنوانها. يمكن أن ترتبط المهمة بزبون أو موظف. ويوفر النظام تقويم لإدارة مهام المستخدم.

يمكن أن يقوم المستخدم المسؤول admin بإضافة موظفيه إلى النظام وإنشاء حسابات كمستخدمين عاديين لهم.

يمكن أن يعرض المسؤول إحصائيات وتقارير لعمله، حيث يوفر النظام عرض مخطط القمع (Funnel chart) للشركة. بالإضافة إلى إدارة خطط عمله التجارية.

حيث يمكن أن يُنشئ خطط عمل تجارية تتضمن دراسة للخطة وإحصائيات لفهم مدى جدواها.

يتم أولاً تحديد الهدف من الخطة أو الرؤية الخاصة بالخطة، ثم تطبيق تحليل SWOT عليها لتحديد نقاط القوة والضعف وفهم السوق ولتحديد الإجراءات اللازم اتباعها لتحقيق أفضل نتيجة.

ثم يتم إجراء عدة دراسات وإحصائيات لتحديد أفضل الزبائن الذين لديهم أكبر احتمالية للشراء.

يتم تطبيق تحليل pareto لتحديد الزبائن الذين حققوا أكبر ربح للشركة. ويتم عرض مخطط ABC لهم. كما يسمح النظام بوضع أرباح متوقعة من الزبائن في المستقبل وتصنيف المستخدمين وفقاً للأرباح السابقة والأرباح المتوقعة.

كما يتم تطبيق تحليل RFM، الذي يصنف الزبائن وفقاً للمدة الزمنية منذ آخر شراء، وتكرارات الشراء، والربح التراكمي منه خلال فترة زمنية محددة.

وأخيراً يتم تصنيف الزبائن وفقاً لعاملين، العامل الأول هو نسبة الأرباح المتوقعة والعامل الثاني هو مدى إمكانية الوصول إلى الزبون.

يمكن أن يقوم المستخدمون بتسجيل الخروج logout.

## 1.1 متطلبات النظام:

### 1.1.1 المتطلبات الوظيفية:

1. تسجيل دخول وتسجيل خروج login and logout.
2. إدارة الزبائن من إضافة وتعديل وحذف.
3. إدارة المستخدمين المحتملين Leads.
4. إدارة المنتجات.
5. تحديد خطوات البيع sales steps وتعديلها وحذفها.
6. إنشاء opportunity وإدارته وتتبع حالته.
7. إدارة المهام وعرضها.
8. عرض التقويم.
9. عرض مخطط القمع (Funnel chart).
10. إضافة الموظفين وإنشاء حسابات لهم.
11. إدارة خطط العمل التجارية، حيث في كل خطة يتم تحديد الهدف منها وتطبيق تحليل SOWT، وتحليلات pareto و RFM و IPO/IPA.

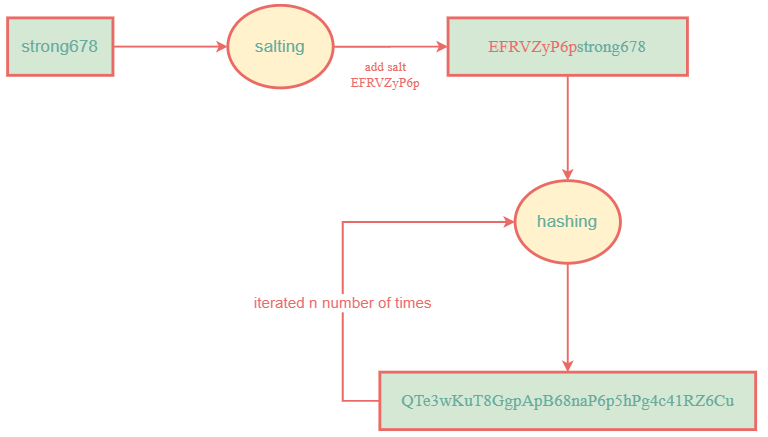
### 2.1.1 المتطلبات غير الوظيفية:

* **الأمان:**

**يتم تشفير كلمات المرور والبيانات الحساسة باستخدام خوارزمية BCrypt، حيث تمر عملية التشفير بالمراحل التالية:**

* **يتم أولاً إضافة تسلسل عشوائي من الأحرف إلى كلمة المرور.**
* **ثم يتم تجزئة كلمة المرور.**
* **تحافظ عملية التجزئة على تكرار نفسها لعدد محدد من الجولات ، يسمى عامل التكلفة.**

**يبين الشكل التالي مراحل الخوارزمية:**

****

الشكل 1. **مراحل BCrypt**

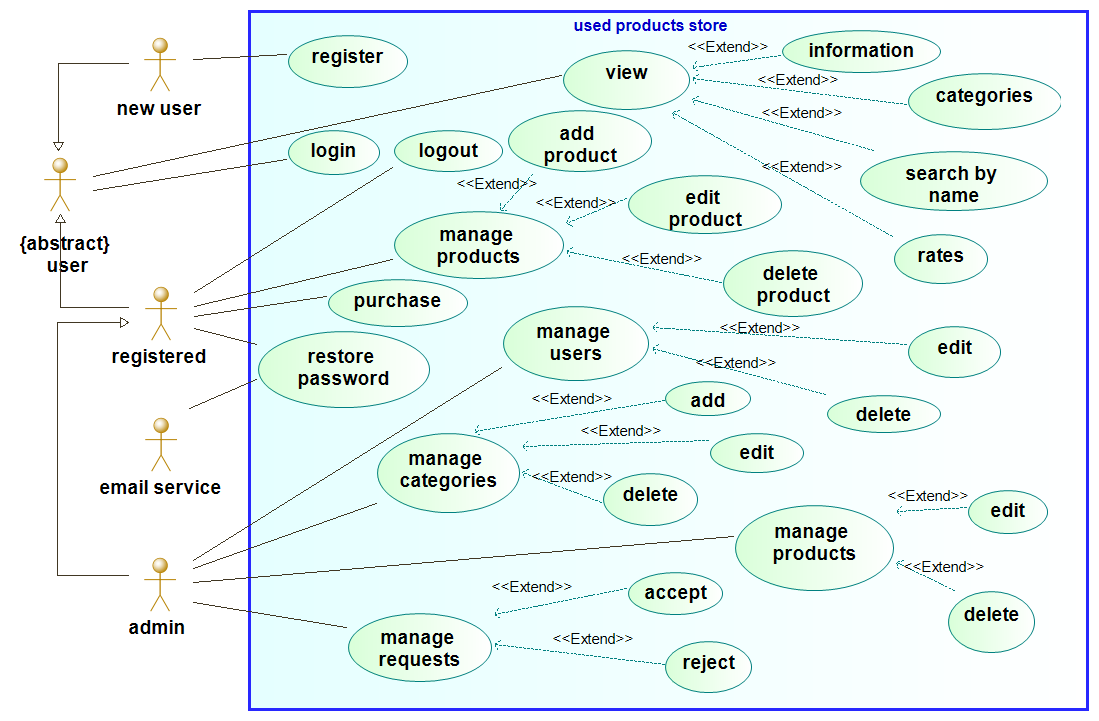
* **قابلية التوسيع والصيانة:**

**حيث النظام مبني على معمارية الـ MVC. إن المفهوم الأشهر لنمط MVC هو أنه يعتمد على مبدأ فصل الاهتمامات SoC (Separation of concerns)، حيث تعد تطبيقات الويب الحديثة معقدة للغاية، وقد يكون إجراء تغيير في بعض الأحيان مشكلة كبيرة.**

**الاعتماد على هذا المبدأ يتيح للتطبيق أن يكون قابلاً للصيانة والتوسيع.**

## 2.1 مخططات UML:

### 1.2.1 مخطط حالة الاستخدام Use Case Diagram:

****

الشكل 2. **مخطط حالة الاستخدام**

**يوضح مخطط حالة الاستخدام تفاعل المستخدم مع النظام، حيث يكون المستخدم إما مسجل أو مسؤول، كما يمكن أن يكون غير مسجل. يمكن لجميع المستخدمين عرض السلع وفئاتها والبحث عن سلعة وتسجيل حساب جديد. بمجرد تسجيل الدخول يكون المستخدم قادراً على إضافة سلعة لبيعها، كما يمكنه تعديل السلع الخاصة به أو حذفها. ويمكنه أيضاً اختيار سلعة لشرائها. في حال نسيان المستخدم لكلمة المرور يمكنه استعادتها عن طريق البريد الإلكتروني.**

**تتلخص صلاحيات المستخدم المسؤول في إدارة المستخدمين والسلع والفئات والطلبات من إضافة وتعديل وحذف.**

* **جداول توصيف حالات الاستخدام:**

**يبين الجدول 1 توصيف حالة الاستخدام login:**

|  |  |
| --- | --- |
| Name: | Login |
| Short description: | The user logs into the system using his user name and password. |
| Precondition: | The user has registered. |
| Postcondition: | The user can add and purchase products, he can also logout.  The admin can manage users, products, categories and requests. |
| Error situations: | 1. Invalid email. 2. Wrong password. |
| System state in the event of an error: | The system show an error message and the user wouldn’t access to his account. |
| Actors: | user / admin |
| Trigger: | The user clicks Login button. |
| Standard process: | 1. User enters his user name and password and hits login button. 2. System confirms that user name and password are valid. 3. User successfully logs in. |
| Alternative processes: | (2`) user name or password is not valid.  (3`) System shows an error message. |

جدول 1. **توصيف حالة الاستخدام login**

**يبين الجدول 2 توصيف حالة الاستخدام add product:**

|  |  |
| --- | --- |
| Name: | add product |
| Short description: | User adds a used product to be shown in the website. |
| Precondition: | The user is logged in. |
| Postcondition: | The product is sent to the admin to be revised and he either accepts it or rejects it. |
| Error situations: | 1. User lost internet connection. 2. Admin rejects the product. |
| System state in the event of an error: | The product is not shown in the website. |
| Actors: | registered / admin |
| Trigger: | The user clicks add product button. |
| Standard process: | 1. User enters product information and hits add product button. 2. A request is sent to the admin. 3. Admin accepts the request. |
| Alternative processes: | (3`) admin does not accept the request.  (4`) The product is not shown in the website. |

جدول 2. **توصيف حالة الاستخدام add product**

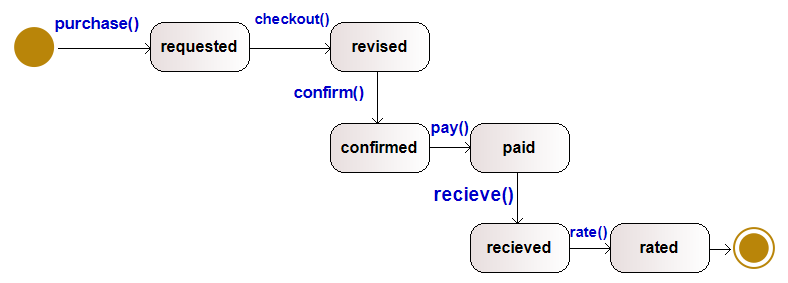
**يبين الجدول 3 توصيف حالة الاستخدام purchase:**

|  |  |
| --- | --- |
| Name: | purchase |
| Short description: | User can purchase a product shown in the website. |
| Precondition: | The user is logged in. |
| Postcondition: | Product is purchased and removed from the website. |
| Error situations: | 1. User does not confirm the order. |
| System state in the event of an error: | Product is not purchased. |
| Actors: | registered / admin |
| Trigger: | The user clicks purchase button. |
| Standard process: | 1. User hits purchase button. 2. Purchase order is created. 3. User revises the order. 4. User confirms the order. 5. User pays. 6. User receives the product. |
| Alternative processes: | (4`) User does not confirm the order.  (5`) Product is not purchased. |

جدول 3. **توصيف حالة الاستخدام purchase**

### 2.2.1 مخططات الحالة state machine diagrams:

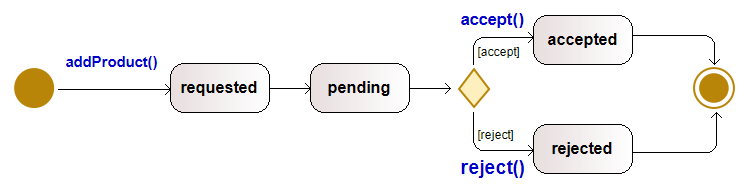
* **مخطط الطلب order:**

****

الشكل 3. **مخطط الحالة للطلب**

**تبدأ عملية الشراء بتقديم المستخدم لطلب شراء فتصبح حالة الطلب requested، يقوم بعدها بالتحقق من الطلب فينتقل إلى الحالة revised أي تمت مراجعته، بعدها يتم التأكيد فيصبح الطلب مؤكداً، وبعد الدفع يصبح في حالة paid ثم يتم استلام السلعة فيصبح في حالة مُستَلَم، في النهاية يقوم الشاري بتقييم البائع فينتهي الطلب.**

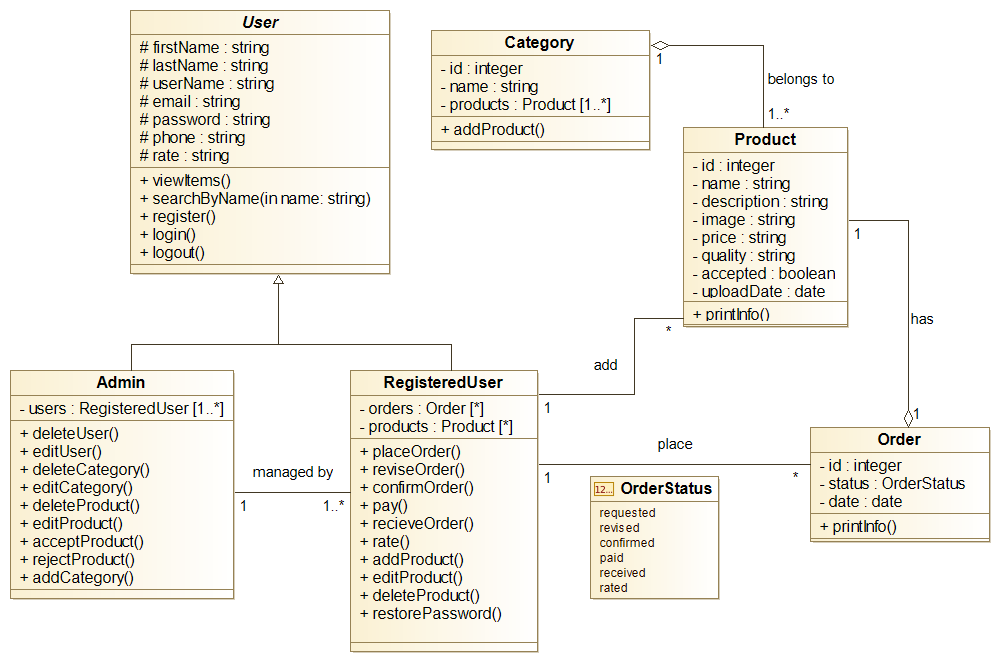
* **مخطط المنتج product:**

****

الشكل 4. **مخطط الحالة للمنتج**

**يرسل المستخدم (البائع) طلب لإضافة منتج، فيصبح المنتج في حالة انتظار حتى تتم معالجة الطلب من قبل المسؤول، فيمكن قبول الطلب -بالتالي يصبح المنتج متوفراً- أو رفضه.**

### 3.2.1 مخطط الصفوف class diagram:

****

الشكل 5. **مخطط الصفوف**

# التقنيات والبرامج المستخدمة:

## 1.2 تقنيات الـ Frontend:

### 1.1.2 لغة HTML:

**HTML هي اختصار لـ Hyper Text Markup Language (لغة ترميز النص التشعبي) هي اللغة المستخدمة لإنشاء صفحات الويب. HTML ليست لغة برمجة. إنها لغة ترميز، مما يعني أنها نظام لتحديد ووصف المكونات المختلفة لصفحة الويب مثل العناوين والفقرات والقوائم. يشير الترميز إلى الهيكل الأساسي للصفحة (يمكنك التفكير فيه على أنه مخطط تفصيلي مفصل يمكن قراءته آلياً). لا تحتاج إلى مهارات البرمجة -فقط الصبر والحس السليم- لكتابة HTML. [1]**

### 2.1.2 CSS

**بينما يتم استخدام HTML لوصف المحتوى في صفحة الويب، تصف CSS (cascading style sheets) كيف يجب أن يبدو المحتوى. يتم التحكم في جميع الخطوط والألوان وصور الخلفية وتباعد الأسطر وتخطيط الصفحة وما إلى ذلك باستخدام CSS. يمكنك أيضاً إضافة مؤثرات خاصة ورسوم متحركة بسيطة إلى صفحتك. [1]**

### 3.1.2 JavaScript:

**JavaScript هي لغة برمجة تضيف التفاعل والسلوكيات إلى صفحات الويب، بما في ذلك (على سبيل المثال لا الحصر):**

* **التحقق من إدخالات النموذج لإدخالات صالحة.**
* **تبديل تصميم عنصر أو موقع بأكمله.**
* **تحميل يغذي التمرير بالمزيد من المحتوى تلقائياً.**
* **جعل المتصفح يتذكر المعلومات المتعلقة بالمستخدمين.**
* **إنشاء عناصر واجهة مستخدم، مثل مشغلات الفيديو المضمنة أو مدخلات النماذج الخاصة.**

**قد تسمع أيضاً مصطلح DOM المستخدمة فيما يتعلق بجافا سكريبت. يرمز DOM إلى Document Object Model )نموذج كائن المستند(، ويشير إلى القائمة الموحدة لعناصر صفحة الويب التي يمكن الوصول إليها ومعالجتها باستخدام JavaScript (أو لغة برمجة نصية أخرى). [1]**

### 4.1.2 React JS:

* **ما هيReact ؟ [2]**

هي مكتبة JavaScript تعريفية وفعالة ومرنة لبناء واجهات المستخدم. تتيح إنشاء واجهات مستخدم معقدة انطلاقاً من أجزاء صغيرة ومعزولة من التعليمات البرمجية تسمى المكونات "components". نستخدم المكونات لإخبار React بما نريد عرضه على الشاشة. عندما تتغير البيانات، ستقوم React بتحديث المكونات وإعادة عرضها بكفاءة.

يأخذ المكوّن وسطاء تسمى props (اختصاراً لـ "Properties")، ويعيد تسلسلاً هرمياً من المكونات الجزئية لعرضها باستخدام طريقة تسمى render(). تعرض الطريقة render() وصفاً لما سيُعرَض على الشاشة. تأخذ React الوصف وتعرض النتيجة. على وجه الخصوص، تُعيد عنصر React، وهو وصف بسيط لما سيتم عرضه.

يستخدم معظم مطوري React صيغة خاصة تسمى "JSX" مما يجعل كتابة هذه البنى أسهل. تدعم JSX القوة الكاملة لجافا سكريبت. يمكن وضع أي تعبيرات JavaScript ضمن أقواس داخل JSX. كل عنصر من عناصر React هو كائن JavaScript يمكن تخزينه في متغير أو تمريره في البرنامج.

* **ميزات React:**

تعتمد على البرمجة التصريحية Declarative Programming، حيث تسهّل إنشاء واجهات مستخدم تفاعلية. تسمح أيضاً بتصميم طرق عرض بسيطة لكل حالة في التطبيق، وتقوم React بتحديث المكونات الصحيحة فقط وعرضها عندما تتغير البيانات. تجعل المكونات التصريحية الكود أكثر قابلية للتنبؤ وأسهل في التصحيح. [3]

كما تسمح بإنشاء تطبيقات من صفحة واحدة single-page websites. التطبيق أحادي الصفحة هو تطبيق يقوم بتحميل صفحة HTML واحدة وجميع الملفات الضرورية (مثل JavaScript و CSS) المطلوبة لتشغيل التطبيق. لا تتطلب أي تفاعلات مع الصفحة أو الصفحات اللاحقة ذهاباً وإياباً إلى الخادم، مما يعني عدم إعادة تحميل الصفحة. [4]

* **إنشاء مشروع React جديد وتشغيله: [5]**

تعد تقنية Create React App بيئة مريحة لتعلم React، وهي أفضل طريقة لبدء إنشاء تطبيق جديد من صفحة واحدة في React.

تقوم بإعداد بيئة التطوير الخاصة بك بحيث يمكنك استخدام أحدث ميزات JavaScript، وتوفر تجربة مطور رائعة، وتحسن تطبيقك للإنتاج. يجب أن يكون لديك Node> = 14.0.0 و npm> = 5.6 على جهازك. لإنشاء مشروع، قم بتنفيذ الأمر التالي على موجه الأوامر Terminal:

npx create-react-app my\_app

الآن لتشغيل المشروع، ننتقل أولاَ إلى مجلد المشروع الذي تم إنشاءه:

cd my\_app

ثم ننفذ الأمر التالي:

npm start

### 5.1.2 DevExtreme React Grid:

DevExtreme React Grid هو مكون يعرض بيانات الجدول من مصدر محلي أو بعيد. وهو يدعم الفرز والتصفية والتجميع وخيارات تشكيل البيانات الأخرى وتحديد الأسطر وتحرير البيانات. [6]

### 6.1.2 Visual studio code:

Visual Studio Code هو محرر أكواد خفيف ولكنه قوي يعمل على الحاسب ومتاح لأنظمة Windows وmacOS وLinux. يدعم JavaScript و TypeScript و Node.js بشكل افتراضي ولديه نظام غني من الإضافات للغات الأخرى (مثل C++ وC# وJava وPython وPHP وGo و.NET). [7]

## 2.2 تقنيات الـ Backend:

### 1.2.2 قاعدة بيانات MySQL:

**يستخدم نظام قاعدة بيانات** MySQL **بنية خادم-عميل. الخادم،** mysqld**، هو البرنامج الذي يتعامل بالفعل مع قواعد البيانات. لإخبار الخادم بما يجب القيام به، يتم استخدام برنامج عميل ينقل المطلوب عن طريق عبارات مكتوبة بلغة الاستعلام الهيكلية (**SQL**). تتم كتابة برامج العميل لأغراض متنوعة، ولكن كل منها يتفاعل مع الخادم من خلال الاتصال به، وإرسال عبارات** SQL **لإجراء عمليات قاعدة البيانات، وتلقي النتائج. [8]**

### 2.2.2 Java:

**إن** Java **ليست مجرد لغة برمجة. هناك الكثير من لغات البرمجة المتاحة، لكن القليل منها له تأثير كبير.** Java **عبارة عن منصة كاملة، بها مكتبة ضخمة، تحتوي على الكثير من التعليمات البرمجية القابلة لإعادة الاستخدام، وبيئة تنفيذ توفر خدمات مثل الأمان، وقابلية النقل عبر أنظمة التشغيل، وجمع البيانات المهملة تلقائياً. كمبرمج، ستحتاج إلى لغة ذات قواعد بسيطة ودلالات مفهومة (أي ليس** C++). **تمنحك بعض اللغات إمكانية النقل وجمع البيانات المهملة وما شابه، ولكنها لا تحتوي على الكثير من المكتبات، مما يفرض عليك إنشاء مكتبة خاصة بك إذا كنت تريد رسومات رائعة أو الوصول إلى الشبكات أو قاعدة البيانات. حسناً، تحتوي** Java **على كل شيء — لغة جيدة، وبيئة تنفيذ عالية الجودة، ومكتبة ضخمة. هذا المزيج هو ما يجعل** Java **اقتراحاً لا يقاوم للعديد من المبرمجين. [9]**

### 3.2.2 Spring:

* **ما هو Spring؟ [10]**

Spring **يجعل برمجة** Java **أسرع وأسهل وأكثر أماناً للجميع. جعلَ تركيز** Spring **على السرعة والبساطة والإنتاجية إطار عمل** Java **الأكثر شهرة في العالم. يثق المطورون في جميع أنحاء العالم في مكتبات** Spring **المرنة. يقدم** Spring **تجارب مبهجة لملايين المستخدمين النهائيين كل يوم - سواء كان ذلك عبر البث التلفزيوني أو التسوق عبر الإنترنت أو عدداً لا يحصى من الحلول المبتكرة الأخرى. يحتوي** Spring **أيضاً على مساهمات من جميع الأسماء الكبيرة في مجال التكنولوجيا، بما في ذلك** AlibabaوAmazonوGoogleوMicrosoft **والمزيد.**

* **ماذا تفعل Spring؟ [10]**
* Microservices
* Reactive
* Cloud
* Web apps
* Serverless
* Event Driven
* Batch

### 4.2.2 Spring Boot:

**يعد** Spring Boot **أحد أكثر أطر العمل شيوعاً المستخدمة لتطوير البرامج لأي تطبيق مؤسسي. من المفيد تطوير نظام موزع حديث قائم على السحابة مثل تطبيق قائم على الخدمات المصغرة. كان** Spring Boot **أحد المشاريع الرئيسية لفريق** Pivotal **وتم إطلاقه في عام 2013. يُسهل** Spring Boot **إنشاء تطبيقات** Spring **قائمة بذاتها يمكنك تشغيلها. ميزتها هي إمكانية البدء بأقل قدر من الإعدادات. تحتاج معظم تطبيقات** SpringBoot **إلى القليل جداً من إعدادات** Spring**. [11]**

### 5.2.2 MySQL Workbench:

**يعد** MySQL Workbench **أداة مرئية موحدة لمهندسي قواعد البيانات والمطورين وموظفي قواعد البيانات. يوفر** MySQL Workbench **نمذجة البيانات وتطوير** SQL **وأدوات الإدارة الشاملة لتكوين الخادم وإدارة المستخدم والنسخ الاحتياطي وغير ذلك الكثير. يتوفر** MySQL Workbench **على أنظمة التشغيل** Windows **و** Linux **و** Mac OS X**. [12]**

**يمكن تحمليه من الرابط التالي:** [**https://dev.mysql.com/downloads/workbench/**](https://dev.mysql.com/downloads/workbench/) **.**

### 6.2.2 Spring Tool Suit (STS)

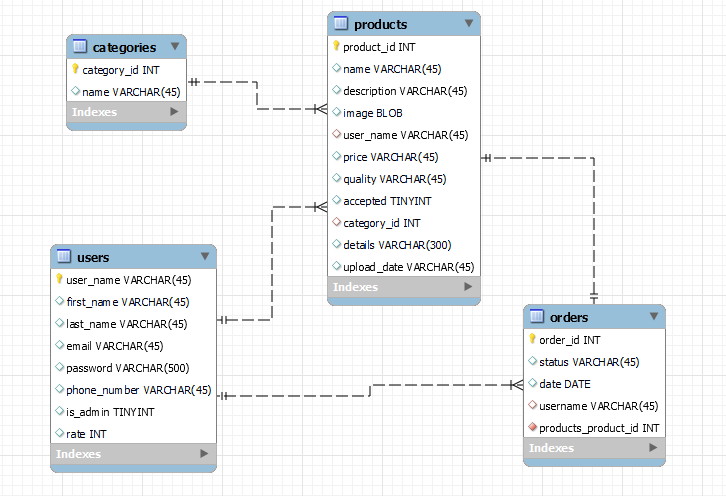
**يجعل** Spring Tool Suite 4 **من السهل البدء. يتيح لك التكامل المباشر والسهل الاستخدام للانتقال من لا شيء إلى تطبيق** Spring Boot **قيد التشغيل في ثوانٍ. [13]**

**يمكن تحمليه من الرابط التالي:** [**https://spring.io/tools**](https://spring.io/tools) **.**

# تصميم قاعدة البيانات Database Design:

**تم إنشاء قاعدة البيانات باستخدام نظام إدارة قاعدة البيانات MySQL. هناك أربعة جداول في قاعدة البيانات، هي جدول المستخدمين users، وجدول المنتجات products، وجدول الفئات categories، وجدول الطلبات orders، يبين الشكل 6 مخطط كيان-علاقة لقاعدة البيانات (ERD).**

* **جدول المستخدمين users : يتم ضمنه تسجيل معلومات عن المستخدمين، حيث يحوي اسم المستخدم user\_name وهو المفتاح الرئيسي، والاسم والكنية first\_name و last\_name والبريد الإلكتروني email وكلمة المرور password، بالإضافة إلى رقم الموبايل phone\_number وتقييم المستخدم rate، وعمود is\_admin يعبر فيما إذا كان المستخدم مسؤول أم لا.**
* **جدول المنتجات products: يتم ضمنه تسجيل معلومات عن المنتجات الموجودة في النظام، ويحوي رقم المنتج product\_id وهو رقم فريد يمثل المفتاح الرئيسي للجدول، واسم المنتج name ووصف قصير له description والجودة quality والسعر price والتفاصيل details وتاريخ الإضافة upload\_date وصورة المنتج image، بالإضافة إلى اسم المستخدم user\_name وهو مفتاح أجنبي من الجدول users، ورقم الفئة التي يتبع لها category\_id وهو مفتاح أجنبي من الجدول categories والعمود accepted لتحديد فيما إذا كان المنتج في حالة انتظار القبول أو تم قبوله.**
* **جدول الفئات categories: يحوي على رقم الفئة category\_id وهو مفتاح رئيسي واسم الفئة name.**
* **جدول الطلبات orders: يتم ضمنه تسجيل معلومات عن الطلبات ويحوي على رقم الطلب order\_id وهو مفتاح رئيسي، وتاريخ تقديم الطلب date واسم مقدم الطلب user\_name وهو مفتاح أجنبي من الجدول users، ورقم المنتج product\_id وهو مفتاح أجنبي من الجدول products وحالة الطلب status.**

****

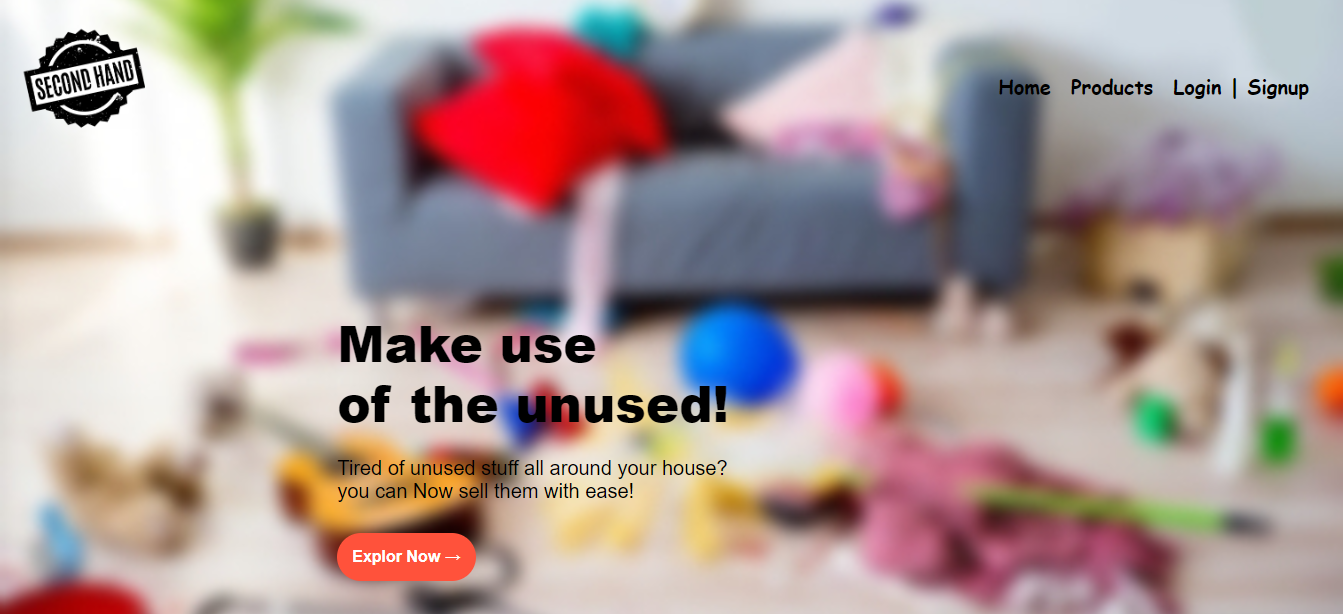
الشكل 6. **مخطط ERD**

# صفحات الموقع الأساسية:

## 1.4 الصفحة الرئيسية للموقع Home Page:

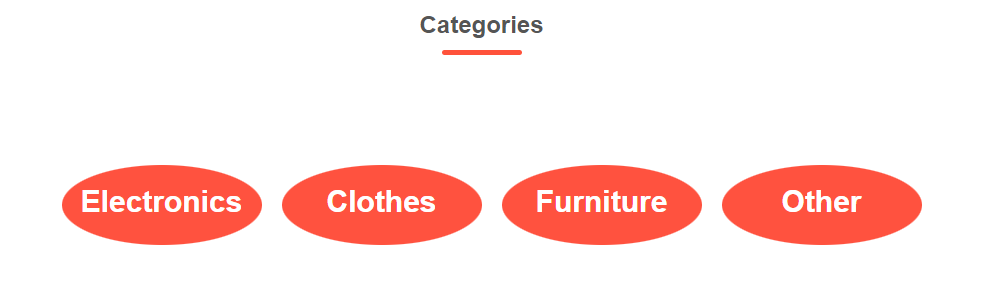
**تتضمن الصفحة الرئيسية للموقع شريط التنقل Navigation bar، الذي يحوي روابط الموقع، حيث يظهر للمستخدم الجديد رابط عرض جميع المنتجات ورابط تسجيل حساب أو تسجيل الدخول، كما يتم عرض الفئات والمنتج البارز والمنتجات المميزة بالإضافة إلى المنتجات المضافة حديثاً.**

**يبين الشكل 7 واجهة الموقع الرئيسية وشريط التنقل بالنسبة لمستخدم جديد:**

****

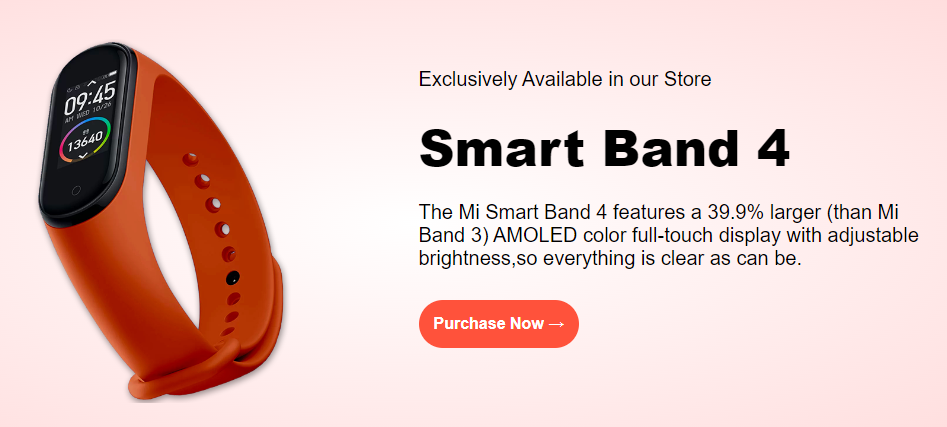
الشكل 7. **الصفحة الرئيسية**

**يبين الشكل 8 عرض فئات المنتجات:**

****

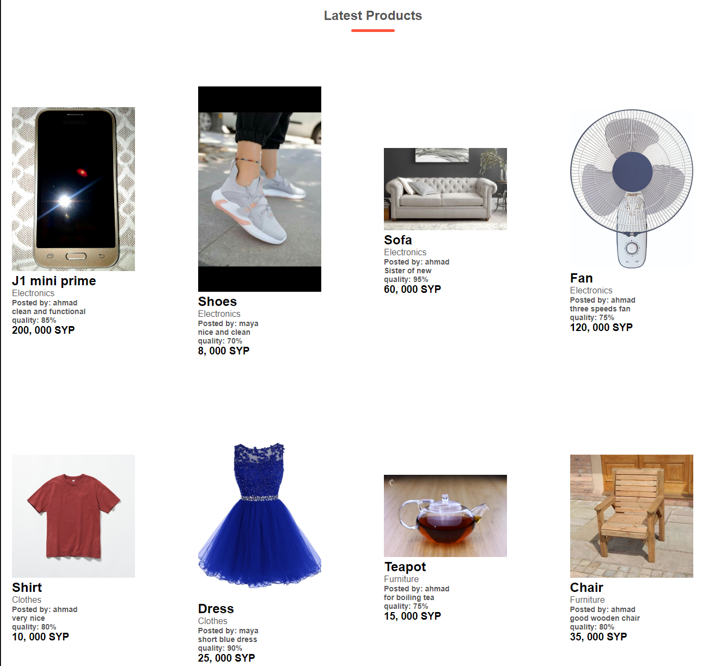
الشكل 8. **فئات المنتجات**

**يبين الشكل 9 المنتج البارز:**

****

الشكل 9. **المنتج البارز**

**يبين الشكل 10 عرض المنتجات المضافة حديثاً:**

****

الشكل 10. **المنتجات المضافة حديثاً**

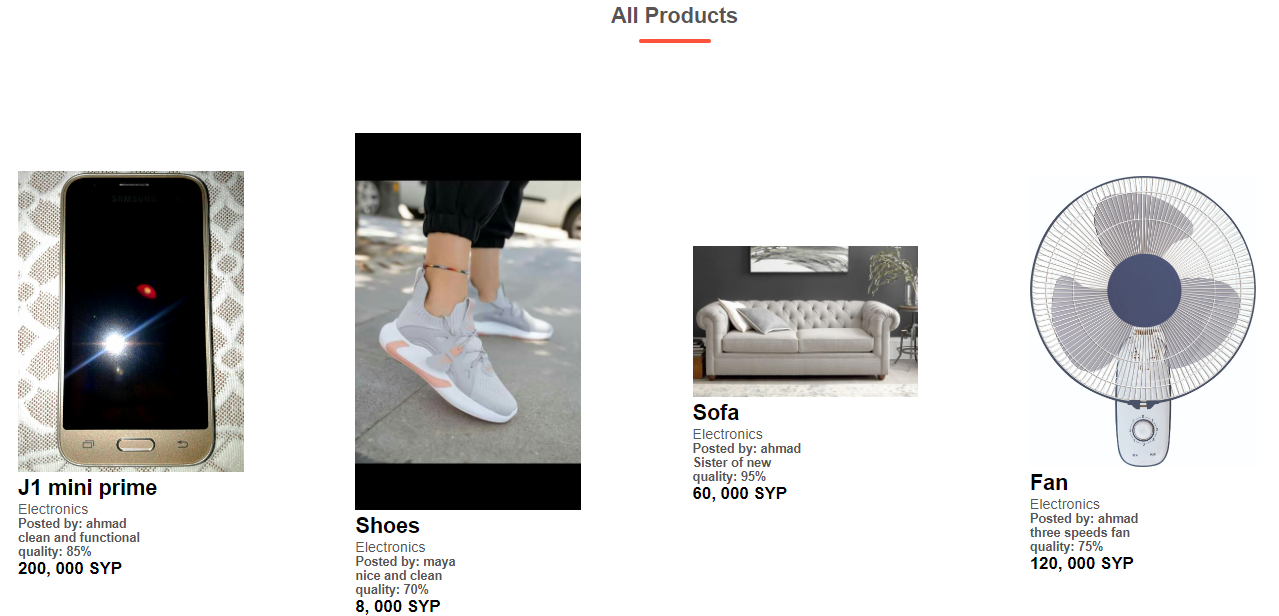
**عند الضغط على منتج يتم عرض تفاصيله، وهي صورة المنتج واسمه والفئة التي يتبع لها ووصف قصير عنه، بالإضافة إلى صاحب المنتج وجودة المنتج وسعره وتفاصيله وتاريخ الإضافة. كما هو مبين في الشكل 11:**

****

الشكل 11. **تفاصيل المنتج**

## 2.4 صفحة جميع المنتجات:

**عند الضغط على الرابط products الموجود في شريط التنقل، ننتقل إلى صفحة جميع المنتجات، وهي على الشكل التالي:**

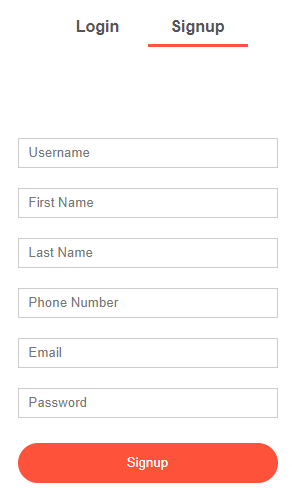
****

الشكل 12. **جميع المنتجات**

## 3.4 صفحة تسجيل حساب أو تسجيل دخول:

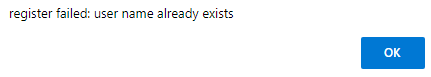
**عند الضغط على الرابط Login | Signup في شريط التنقل، ننتقل إلى صفحة تسجيل حساب أو تسجيل دخول.**

**في حال كان المستخدم غير مسجل (أي لا يملك حساب في الموقع) يختار نافذة تسجيل حساب Signup، حيث يقوم بإدخال المعلومات المطلوبة (اسم المستخدم، الاسم والكنية، رقم الموبايل، الإيميل، كلمة المرور)، ثم يضغط على الزر Signup وفي حال تم تسجيل الحساب بشكل صحيح، يتم نقل المستخدم إلى الصفحة الرئيسية، أما في حال عدم نجاح عملية التسجيل يتم عرض رسالة الخطأ الحاصلة للمستخدم.**

****

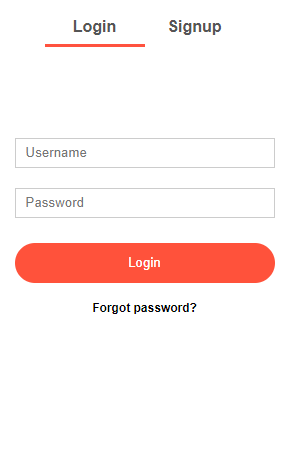
الشكل 13. **نافذة تسجيل حساب Signup**

**إذا قام المستخدم بإدخال اسم مستخدم موجود مسبقاً في النظام يتم عرض رسالة خطأ توضح هذا الأمر، وهي على الشكل التالي:**

****

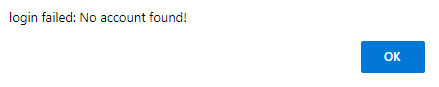
الشكل 14. **رسالة الخطأ عند إدخال اسم مستخدم موجود مسبقاً**

**في حال كان المستخدم مسجل (أي يملك حساب في الموقع) يختار نافذة تسجيل دخول Login، حيث يقوم بإدخال المعلومات المطلوبة (اسم المستخدم، كلمة المرور)، ثم يضغط على الزر Login وفي حال تم تسجيل الدخول بشكل صحيح، يتم نقل المستخدم إلى الصفحة الرئيسية، أما في حال عدم نجاح عملية تسجيل الدخول يتم عرض رسالة الخطأ الحاصلة للمستخدم.**

****

الشكل 15. **نافذة تسجيل دخول Login**

**إذا قام المستخدم بإدخال اسم مستخدم غير موجود مسبقاً في النظام يتم عرض رسالة خطأ توضح هذا الأمر، وهي على الشكل التالي:**

****

الشكل 16. **رسالة الخطأ عند إدخال اسم مستخدم غير موجود مسبقاً**

**إذا قام المستخدم بإدخال اسم مستخدم صحيح ولكن كلمة المرور خاطئة يتم عرض رسالة خطأ توضح هذا الأمر، وهي على الشكل التالي:**

****

الشكل 17. **رسالة الخطأ عند إدخال كلمة مرور خاطئة**

**عند الدخول بشكل صحيح ونقل المستخدم إلى الصفحة الرئيسية، يصبح المستخدم قادراً على إضافة منتج جديد، بالإضافة إلى استعراض جميع المنتجات الخاصة به وتعديلها أو حذفها. ويمكنه أيضاً تسجيل الخروج logout.**

**يوضح الشكل التالي الروابط الإضافية التي يمكن للمستخدم الوصول إليها:**

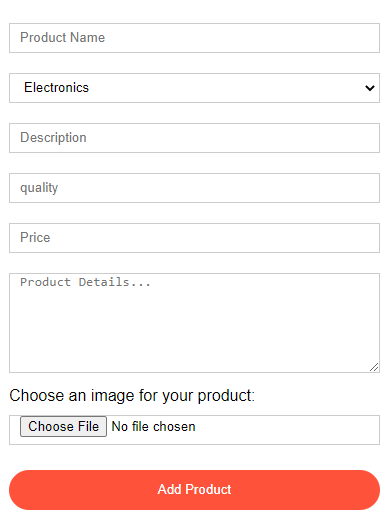
****

الشكل 18. **شريط التنقل للمستخدم المسجل**

## 4.4 صفحة إضافة منتج:

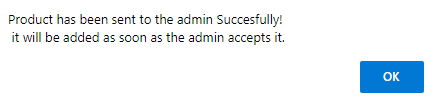
**تظهر للمستخدم المسجل فقط، عند الضغط على الرابط Add New Product + في شريط التنقل، ننتقل إلى صفحة إضافة منتج.**

**لإضافة منتج جديد يجب على المستخدم أن يقو بإدخال اسم المنتج ويختار إحدى الفئات الموجودة في النظام، ثم يكتب وصف قصير للمنتج، ثم يُدخِل الجودة؛ وهي نسبة مئوية تتراوح بين 0 و100، والسعر، ويكتب تفاصيل أكثر عن المنتج في حال أراد ذلك، وأخيراً يختار صورة للمنتج.**

****

الشكل 19. **إضافة منتج**

**عند نجاح عملية الإضافة، يصبح المنتج في حالة انتظار حتى يتم قبوله من قبل المسؤول، ويتم إعلام المستخدم بهذا الأمر عن طريق عرض رسالة توضح الأمر. كما هو مبين في الشكل التالي:**

****

الشكل 20. **طلب منتج**

## 5.4 الصفحة الشخصية للمستخدم profile:

**عند تسجيل الدخول، يتم استبدال رابط Login | Signup باسم وكنية المستخدم المسجل حالياً، عند الضغط على هذا الاسم يتم الانتقال إلى صفحة المستخدم الشخصية، التي تعرض المنتجات التي قام بإضافتها والمعروضة في الموقع (أي تم قبولها من قبل المسؤول)، بحيث يمكنه تعديلها أو حذفها.**

****

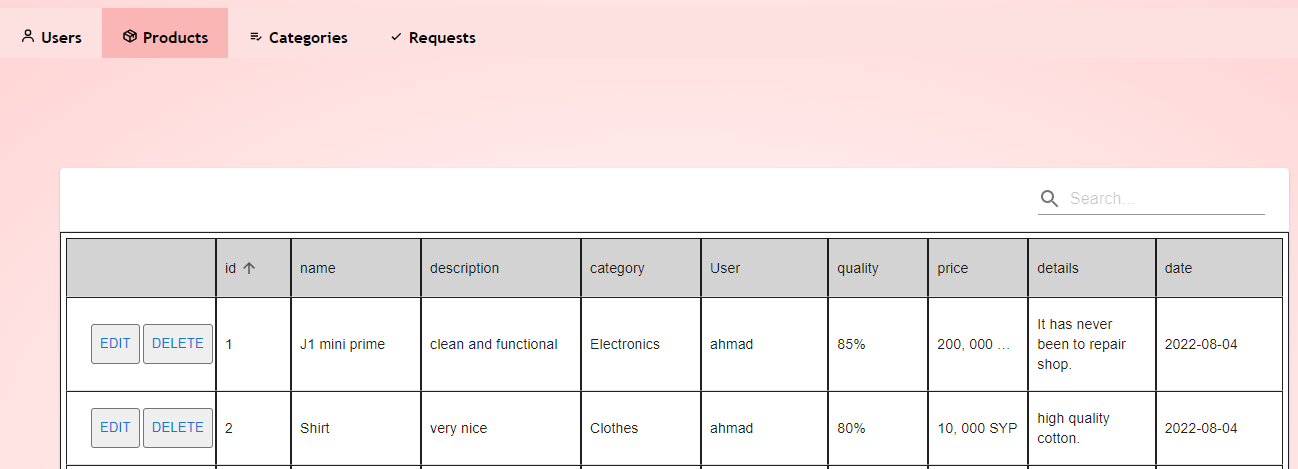
الشكل 21. **الصفحة الشخصية**

## 6.4 صفحة المسؤول:

**يمكن الوصول إليها من قبل المستخدم المسؤول فقط. حيث يمكن عرض جميع المستخدمين والمنتجات والفئات والطلبات.**

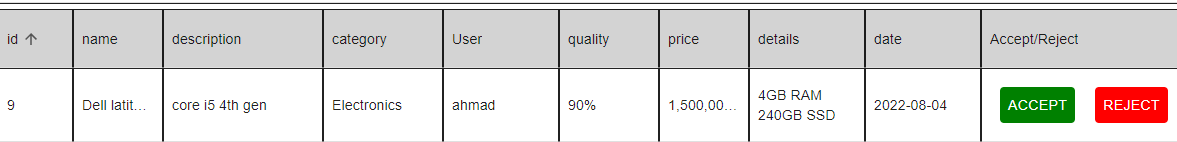
**يملك المستخدم المسؤول صلاحية تعديل أو حذف أي مستخدم أو منتج أو فئة، بالإضافة إلى إضافة فئة جديدة، وهو مسؤول أيضاً عن تتبع طلبات المنتجات من قبول أو رفض.**

**يبين الشكل التالي المنتجات في واجهة المسؤول:**

****

الشكل 22. **واجهة منتجات المسؤول**

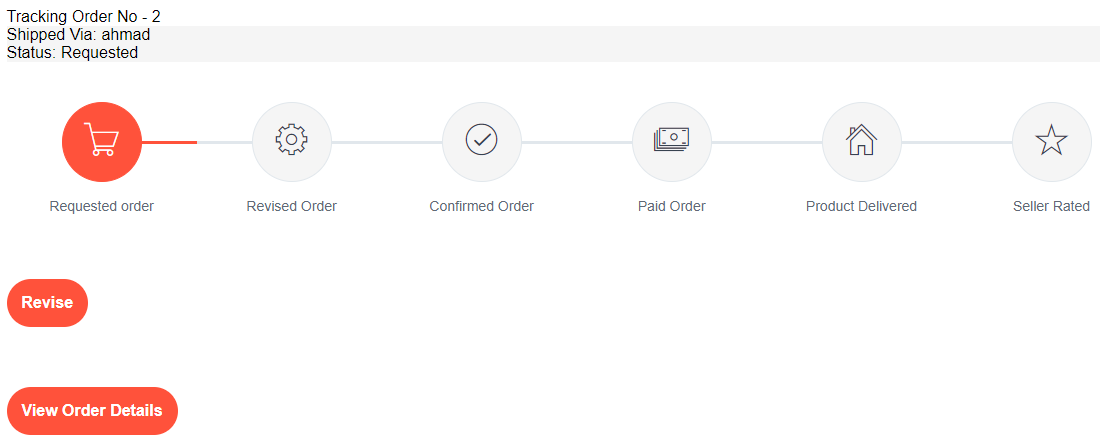
**ويبين الشكل التالي الطلبات في واجهة المسؤول:**

****

الشكل 23. **واجهة تتبع الطلبات**

## 7.4 واجهة تتبع طلب الشراء:

**عند الضغط على زر الشراء** Purchase**، يتم إنشاء طلب شراء جديد ووضعه في الحالة** requested**، بعدها يمكن للمستخدم الانتقال إلى الحالة التالية بالضغط على الزر واتباع التعليمات، كما يمكنه عرض تفاصيل الطلب.**

****

الشكل 24. **واجهة تتبع طلب الشراء**

# الخاتمة:

كان هذا المشروع يهدف بشكل أساسي إلى تطوير موقع ويب لتزويد المستخدمين بمنصة لبيع السلع المستعملة عبر الإنترنت. تشمل الوظائف الرئيسية تسجيل المستخدم وتسجيل الدخول، وعرض جميع السلع المستعملة وفئات السلع، وشراء السلعة المفضلة للمستخدم، وتحميل العناصر المستعملة التي يرغب المستخدم في بيعها، وعرض حالة الطلبات الخاصة به. ومن جانب آخر، المستخدم المسؤول يقوم بإدارة المستخدمين والفئات والسلع وتتبع الطلبات.

بعد اكتمال تطوير المشروع، تم تحقيق الوظائف المتضمنة في المتطلبات الأساسية للنظام. تم اختبار التطبيق على متصفحات مختلفة. أثناء إعداد المشروع، تم تلقي فهم عميق لتعلم Java و JavaScript. في الوقت نفسه، علمنا أننا بحاجة إلى مزيد من الدراسات والتحسينات المتقدمة. نظراً لقيود الوقت والتعلم، لا تزال العديد من جوانب التطبيق ووظائفه بحاجة إلى التحسين. في الدراسة المستقبلية، سيتم تحسين هذه الجوانب والوظائف وتحسينها لجعل التطبيق أكثر احترافاً واكتمالاً.

# مقترحات لتطوير المشروع:

على الرغم من أنه تمت تلبية المتطلبات الأساسية للنظام، إلا أنه يمكن تحسين العديد من جوانبه. يمكن إضافة الوظائف التالية لتحسين النظام:

* تزويد النظام بآلية تقييم المستخدمين باستخدام النجوم عند استلام السلعة من قبل المشتري.
* إضافة نظام دردشة يسمح للبائع والمشتري بالتواصل ضمن النظام.
* إضافة اللغة العربية وإمكانية التنقل بين اللغتين.
* إضافة خدمة دفع إلكتروني عن طريق سيرياتيل كاش و MTN كاش.

# المراجع:

[1]. Robbins, J. N. (2018). *Learning Web Design: A Beginner’s Guide To Html, CSS, JavaScript, And Web Graphics* (Fifth Edition). ISBN: 978-1-491-96020-2. O’Reilly Media.

[2] What is React? Accessed 3.8.2022

<https://reactjs.org/tutorial/tutorial.html#what-is-react>

[3] React: A JavaScript library for building user interfaces. Accessed 3.8.2022

<https://reactjs.org/>

[4] Single Page Application. Accessed 3.8.2022

<https://reactjs.org/docs/glossary.html#single-page-application>

[5] Create React App. Accessed 3.8.2022

<https://reactjs.org/docs/create-a-new-react-app.html#create-react-app>

[6] React Grid – Getting Started. Accessed 3.8.2022

<https://devexpress.github.io/devextreme-reactive/react/grid/docs/guides/getting-started/#react-grid---getting-started>

[7] Visual Studio Code. Accessed 3.8.2022

<https://code.visualstudio.com/docs>

[8] DuBois, P. (2014). *MySQL Cookbook* (Third Edition). ISBN: 978-1-449-37402-0. O’Reilly Media.

[9] Horstmann, C. S. (2019). *Core Java: Volume I-Fundamentals* (Eleventh Edition). ISBN: 978-0-13-516630-7. Pearson Education.

[10] Why Spring? Accessed 3.8.2022

<https://spring.io/why-spring>

[11] Rajput, D. (2019). *Designing Applications with Spring Boot 2.2 and React JS: Step-by-step guide to design and develop intuitive full stack web applications* (First Edition). ISBN: 978-93-88511-643. BPB Publications.

[12] MySQL Workbench. Accessed 3.8.2022

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

[13] Spring Tools 4. Accessed 3.8.2022

<https://spring.io/tools>