Taiz University

AL Saeed Faculty For Engineering & IT

SE Department

Level Three

Pharmacy Management System

By

Ismail Saeed Al-Sharehi (Team Leader)

Mubarak Ashraf Sharaf

Mohammad Ali Mahyoub

Mohammad Radwan Mahyoub Hassan

Mohammad Nabil Al-Shoukani

Supervisor:

Eng Abdulsallam

Project Type in Visual Studio: Windows Forms Application

Database Type: SQL Server

Technology Way : Entity Framework

Libraries and Frameworks Used:

- **Entity Framework** (Framework for database handling)
- **System.Windows.Forms**
- **System.Drawing**
- **System.IO**
- **System.Configuration**
- **System.ComponentModel**

The tasks assigned to each member in the department are as follows:

- **Mohammed Ali Mahyoub**: Responsible for implementing and developing the main interface and the login system.
- **Mubarak Ashraf**: Responsible for implementing and developing the sales and purchases interfaces.
- **Ismail Saeed Al-Sharehi**: Responsible for implementing and developing the profile interface.
- **Mohammed Nabil**: Responsible for implementing the user management, medicine management, company management, categories, and types interfaces.
- **Mohammed Radwan**: Responsible for implementing and developing the backend functionalities for the interfaces created by Mohammed Nabil.

Project Description (Abstract) :

The Pharmacy Management System is an electronic system designed to manage all the essential operations within a pharmacy, including purchasing, selling, and drug management, as well as user permission management.

The system is built using **Windows Forms technology** and relies on SQL Server as the database, with **Entity Framework** being used to handle database operations.

Main Modules of the System:

- Drug Management
- Purchase Management
- Sales Management
- User Management
- Employee Management

System Usage and Interface:

Upon launching the system, the login screen is displayed, prompting the user to enter their email and password. Based on their role (Manager, Supervisor, Employee), the relevant sections of the system are shown.

After logging in, the main interface appears, containing a sidebar for navigation between the following modules:

- **Drug Management Interface:** Adding, modifying, and deleting drugs with the ability to display images, prices, and dates.
- **Purchasing Interface:** Entering supplier information, purchase invoices, and details, as well as generating PDF reports.
- **Sales Interface:** Registering daily sales transactions with customer details and generating sales invoices.
- **User Management Interface:** Managing system accounts and defining user permissions.
- **Employee Management Interface:** Storing employee details linked to the system.

Each screen includes a dynamic search interface for easy data retrieval, and confirmation messages are shown for critical operations such as deletion or modification.

1. *Overview*

In pharmacies or health centers, we often notice that many tasks are repeated daily—like tracking medications, registering user information, issuing invoices, and other basic functions. These tasks are often handled manually or spread across different systems, which leads to issues like time waste, data duplication, and reduced efficiency.

Through this project, we aimed to design a unified and integrated system that helps manage all these tasks from one place. The system will cover treatment management, user data, invoices, and suppliers, and will be accessible via the internet to make employees' work easier and save significant time and effort.

2. *Project Objective*

The goal of this project is to apply concepts of **System Analysis and Design** to build a comprehensive system for managing treatments and invoices. During the project, we will use modeling tools and diagrams to define system requirements, represent its behavior, and design the overall system architecture.

3. *Stakeholders / Beneficiaries*

Below are the main stakeholders who will interact with the system and their roles:

Stakeholder	Role in the System
System Administrator	Manage accounts, permissions, and monitor system performance
Employee & Manager	Issue, modify, and print invoices
Employee Manager	Track treatments, enter quantities, types, and categories
Customers / Visitors	Purchase treatments and receive invoices
System Admins	Manage user and supplier permissions, oversee system activity and access

4. *System Description*

The proposed system is a semi-integrated system that includes the following features:

- Full user management with permission control.
- Management of treatments (adding, editing, selling, and purchasing).
- A comprehensive invoicing system to create, edit, and print invoices.
- Management of manufacturers, types, and categories associated with treatments.
- Secure login based on user roles.

5. *Functional Requirements*

First: Login System
Secure login for all users.

Second: User Management

- **Add User:** Register a new user.
- **Delete User:** Remove inactive users.
- **Edit User:** Update user information.
- **Change User Permissions:** Define or update access levels.

Third: Treatment Management

- **Add Treatment:** Enter new treatment data.
- **Delete Treatment:** Remove treatment from the system.
- **Edit Treatment:** Modify existing treatment information.
- **Browse Treatments:** Display list of available treatments.
- **Sell Treatment:** Record treatment sales.
- **Purchase Treatment:** Log new treatment purchases and add to inventory.

Fourth: Invoice Management

- **Create Invoice:** Issue a new invoice with required details.
- **Edit Invoice:** Modify existing invoice data.
- **Print Invoice:** Print a copy of the invoice.
- **Browse Invoices:** View past invoice records.

Fifth: Management of Manufacturers, Types, and Categories

- **Add Information:** Register new data for companies, types, or categories.
- **Edit Information:** Update existing records.
- **Delete Information:** Remove unnecessary data.

Non-Functional Requirements for the Pharmacy Management System

This section addresses non-functional requirements—how the system should work overall. These aspects are crucial for the system's effectiveness and operation, even if they don't relate directly to core functions.

1. Usability

The system should be easy to use without requiring technical expertise. The interface must be clean and clear, support Arabic, and include simple instructions for users.

2. Performance

The system must be fast and responsive, even when used by multiple staff members at once. It should respond within seconds and handle moderate user load without performance issues.

3. Security

Since the system handles sensitive data (treatments and user info), it must have strong security. Each user logs in with a username and password, and permissions are defined by role. Critical operations like delete or edit should be logged for auditing.

4. Reliability

The system must be dependable and operate continuously, even if minor errors occur. It should avoid frequent crashes and maintain high uptime (availability).

5. Maintainability

Over time, system updates or modifications may be required. Thus, the system should be built in a maintainable way—with organized code and clearly separated components for future developers.

6. Scalability
If there's a need to add features later (like reports or appointment management), this should be possible without rebuilding the system. The design must allow easy expansion.

7. Compatibility
The system should work on different devices—PCs, smartphones, and tablets—and support various browsers like Google Chrome and Firefox.

8. Backup & Recovery
To prevent data loss, there must be daily or weekly backup procedures, and a clear, fast way to recover data if an issue arises.

9. Real-time Updates
Any changes or actions—like sales or adding treatments—must reflect instantly without requiring page reloads.

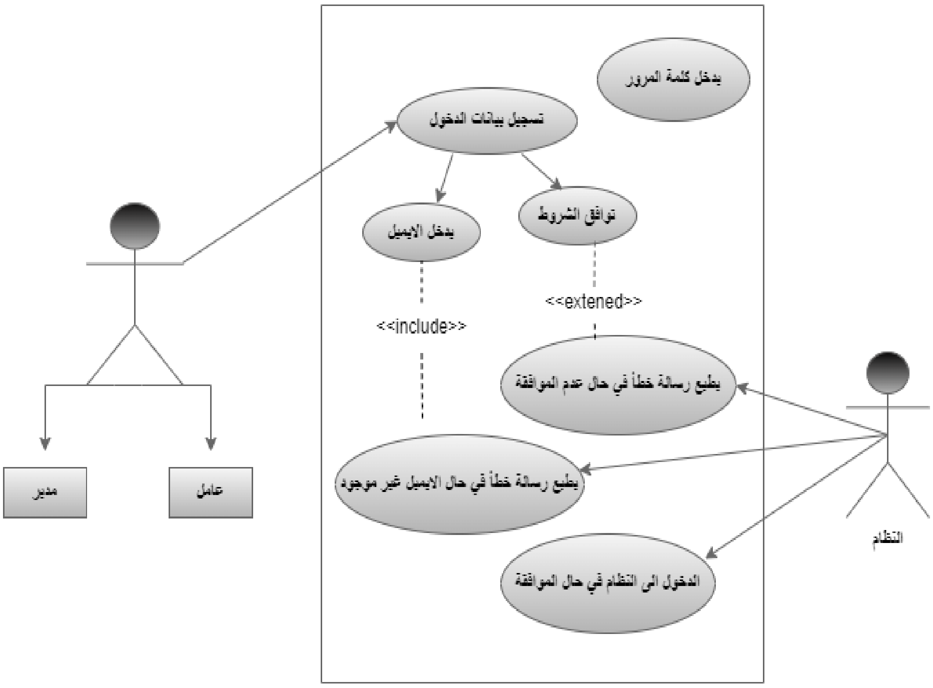
10. Compliance
The system must comply with any local data privacy and protection laws, especially concerning customer and user information.

LoginUse Case

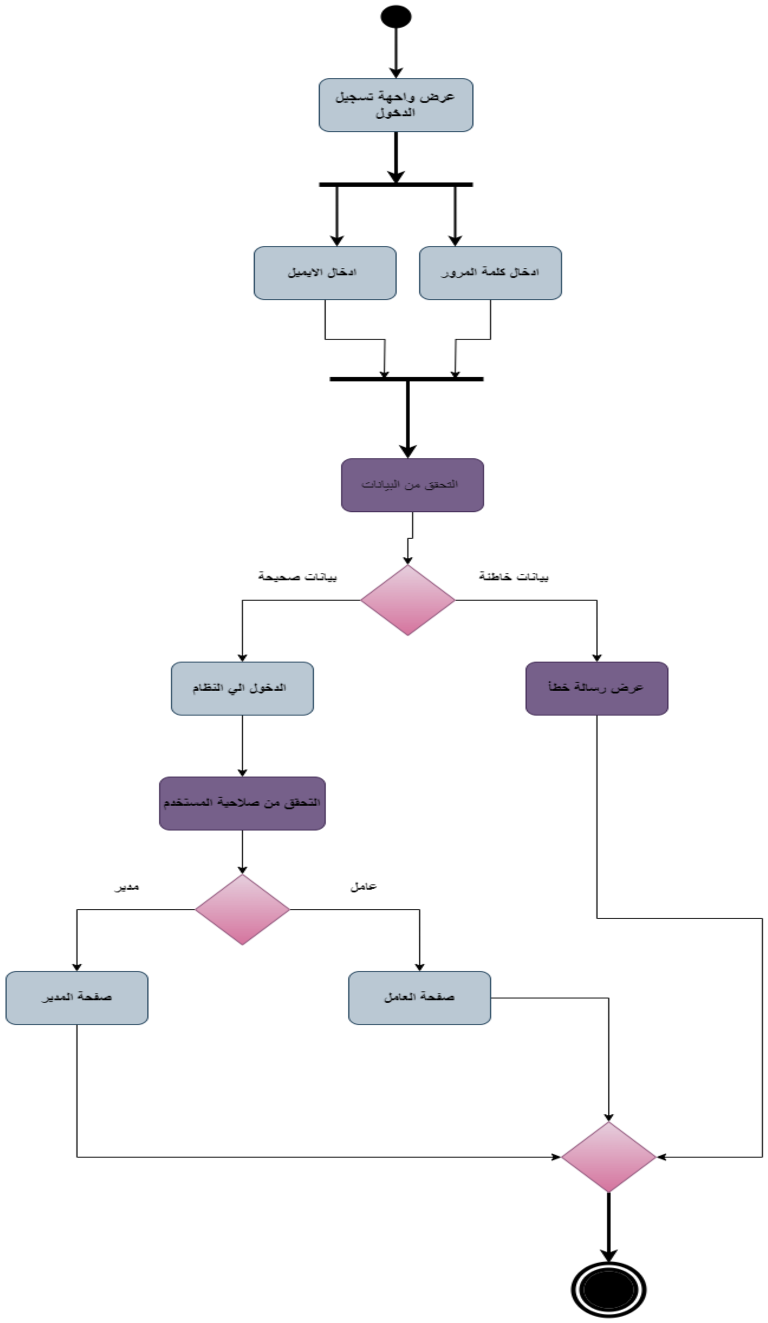
Operation No.	Use Case Name	Actors	Inputs	Processes	Outputs	Preconditions	Postconditions
1	Login	User (Employee and Admin), System	Employee credentials (email and	System verifies the input data	If the data is correct → user logs in. Otherw	The employee must be registered in the database	Employee is granted access to the system

			password)		ise → an error message is displayed asking to verify inputs.		

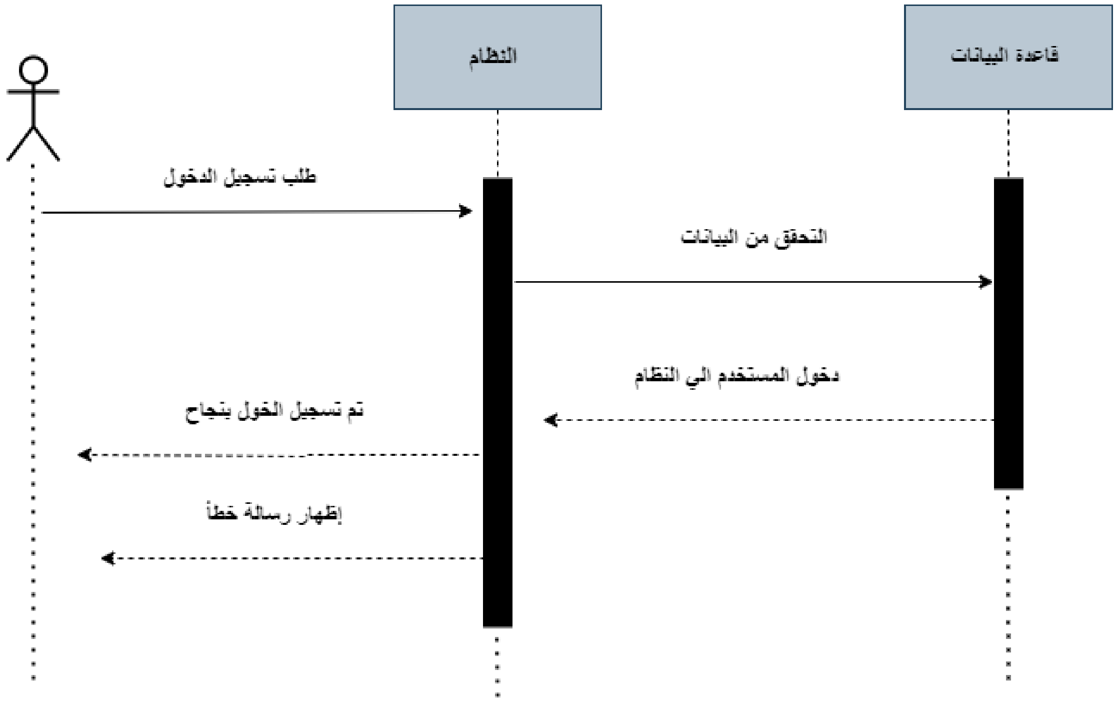
Use Case Diagram for Login


يدخل كلمة المرور
تسجيل بيانات الدخول
يدخل الايميل
توافق الشروط
<<extened>>
<<include>>
يطبع رسالة خطأ في حال عدم الموافقة
يطبع رسالة خطأ في حال الايميل غير موجود
الدخول الى النظام في حال الموافقة
مدير
عامل
النظام

Activity Diagram



Sequence Diagram



Object: Medicine

Attributes:

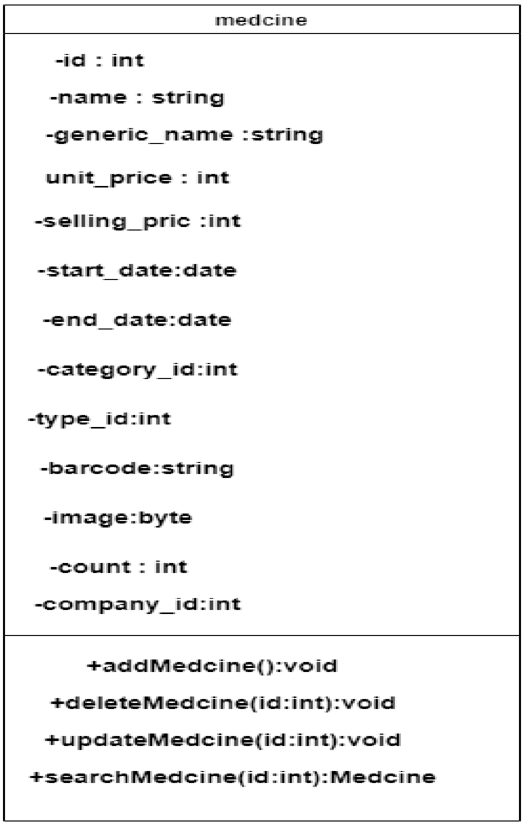
- **id**: A unique identifier automatically assigned to each medicine in the database, used to distinguish between different medicines.
- **name**: The trade name of the medicine as it appears in pharmacies or the market; this is the name commonly known to consumers.
- **generic_name**: The active scientific name of the medicine, used medically regardless of the manufacturer.
- **unit_price**: The purchase price per unit of the medicine paid by the institution (e.g., from a supplier or manufacturer).
- **selling_price**: The price at which the medicine is sold to the final customer, which may include a profit margin.
- **start_date**: The date when the medicine became available in stock or in the medical system.
- **end_date**: The expiration date of the medicine, used to determine when the medicine should be discarded or prohibited from sale.
- **category_id**: An identifier representing the therapeutic category to which the medicine belongs (e.g., antibiotic, analgesic, vitamin...).
- **type_id**: An identifier for the pharmaceutical form of the medicine (e.g., tablet, syrup, injection...).
- **barcode**: The barcode printed on the medicine's packaging, used for quick identification during sales or inspection.

- **image**: An image of the medicine stored as binary data (byte), used for graphical interface display.
- **count**: The number of units currently available in stock for this medicine, used for inventory management.
- **company_id**: An identifier indicating the company that manufactured or supplied the medicine.

Methods:

- **addMedicine()**: A function responsible for adding a new medicine to the database. It validates the entered data and stores it.
- **deleteMedicine(id: int)**: A function that deletes a specific medicine from the system using its unique identifier. It is used to remove unavailable or expired medicines.
- **updateMedicine(id: int)**: A function that allows editing existing medicine data, such as changing the price, quantity, or updating based on new changes.
- **searchMedicine(id: int): Medicine**: A function used to search for a specific medicine in the system by its ID and returns all its data if found.

Medicine Class Diagram



medcine

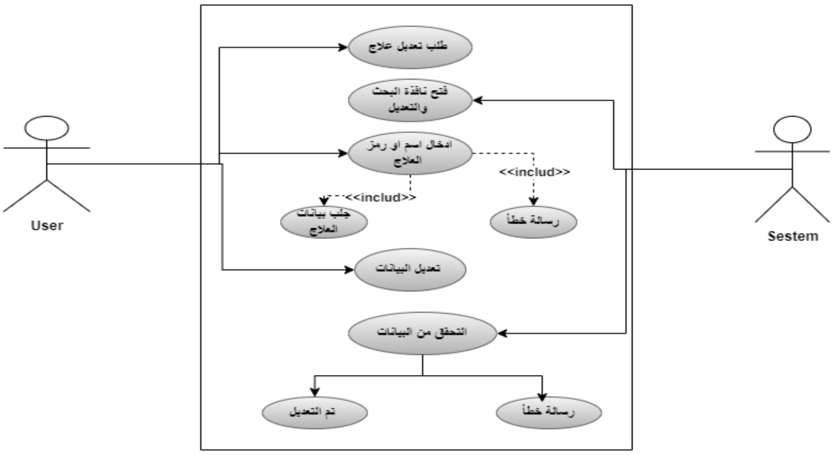
-id : int
-name : string
-generic_name :string
unit_price : int
-selling_pric :int
-start_date:date
-end_date:date
-category_id:int
-type_id:int
-barcode:string
-image:byte
-count : int
-company_id:int

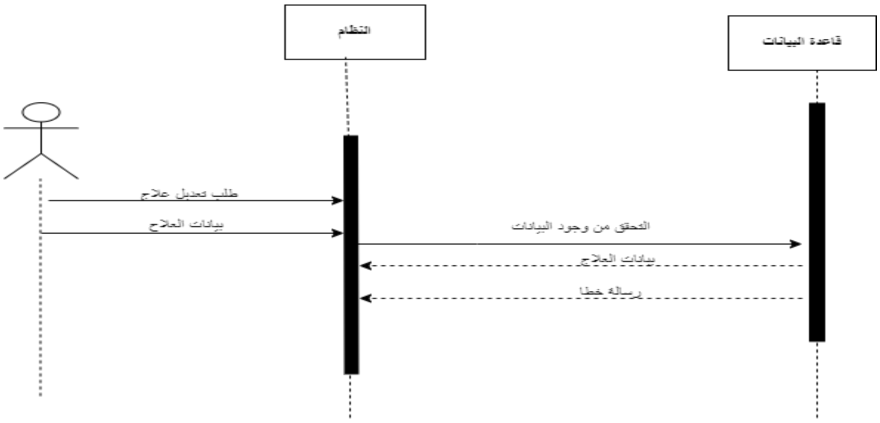
+addMedcine():void
+deleteMedcine(id:int):void
+updateMedcine(id:int):void
+searchMedcine(id:int):Medcine

Edit Medicine – Use Case Description

No.	Use Case Name	Actors	Inputs	Processes	Outputs	Pre-conditions	Post-conditions
	Edit Existing Medicine	System and Users (Staff/Admin)	Medicine data to be edited	1. Request to edit medicine 2. Open search/edit window 2.1 Enter name or barcode — Retrieve medicine data—	— Medicine successfully updated — Or error message shown	Medicine must already exist in the database	Data is updated and saved in the database Or, an error message is displayed

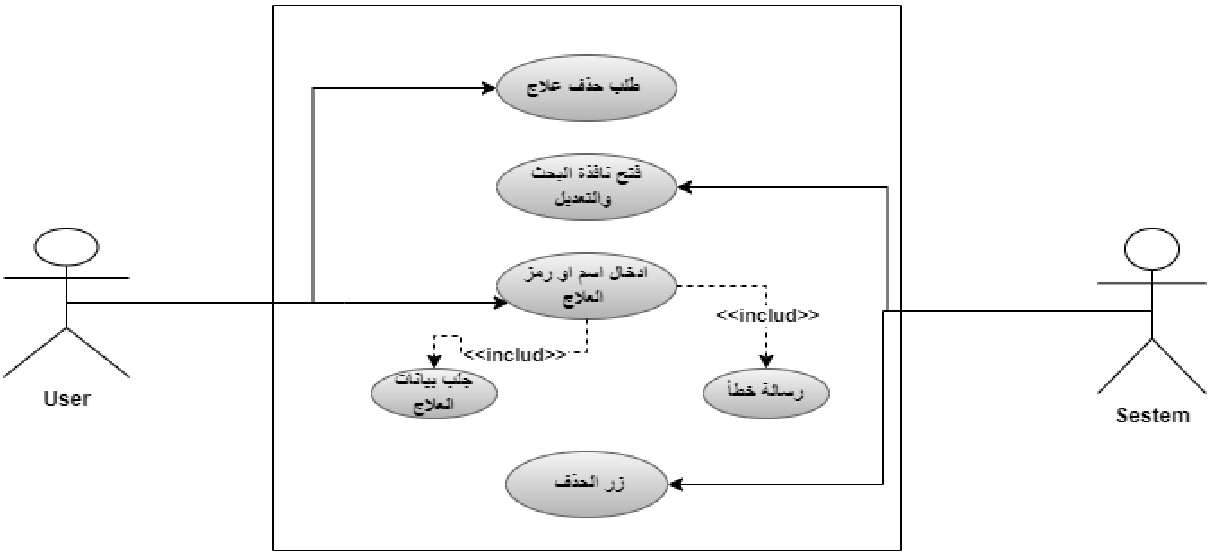
				Show error message if not found3. Edit data4. Submit updated data			

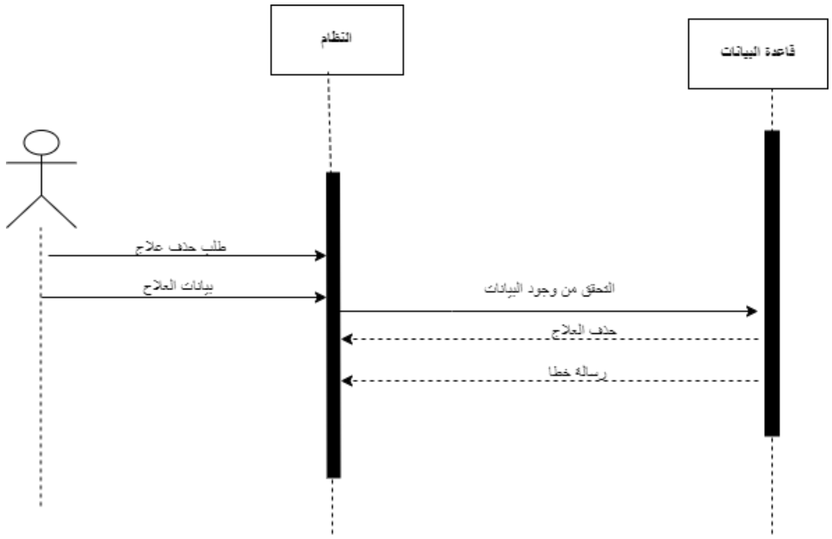




Delete Medicine – Use Case Description

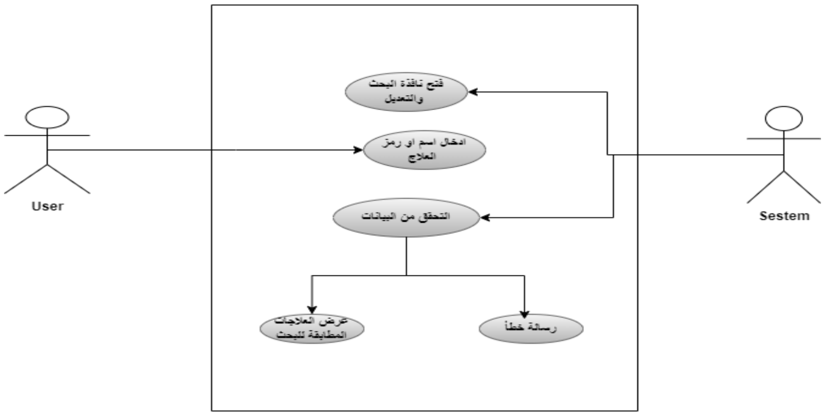
No.	Use Case Name	Actors	Inputs	Processes	Outputs	Pre-conditions	Post-conditions
	Delete Existing Medicine	System and Users (Staff/Admin)	Data of the medicine to be deleted	1. Request to delete medicine2. Open search/edit window 3. Enter medicine name or barcode — Retrieve medicine data— Show error message if not found4. Delete data from system	— Medicine deleted successfully— Or error message shown	Medicine must already exist in the database	Data is deleted from the database Or, an error message is displayed

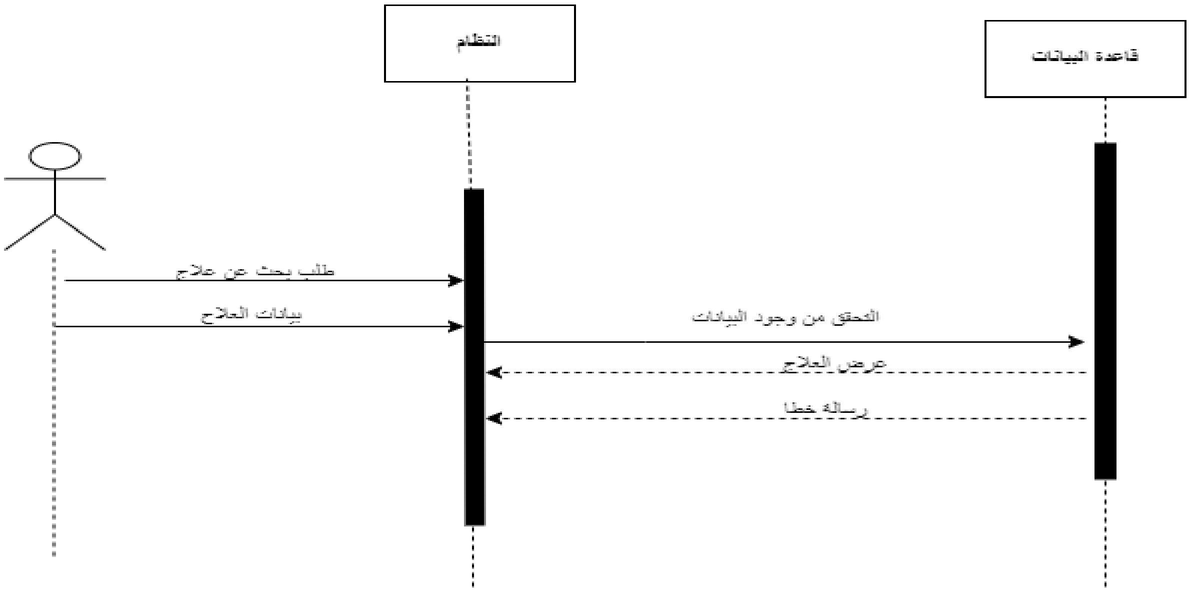




Search for Medicine – Use Case Description

No.	Use Case Name	Actors	Inputs	Processes	Outputs	Pre-conditions	Post-conditions
	Search for Existing Medicine	System and Users (Staff/Admin)	Information about the medicine to be searched for	1. Open the delete/edit window 2. Enter medicine name or barcode3. Validate data– Display matching medicines– Or show an error message	– Medicine data displayed– Or an error message	Medicine must already exist in the database	Matching medicine data is displayed Or, an error message is shown



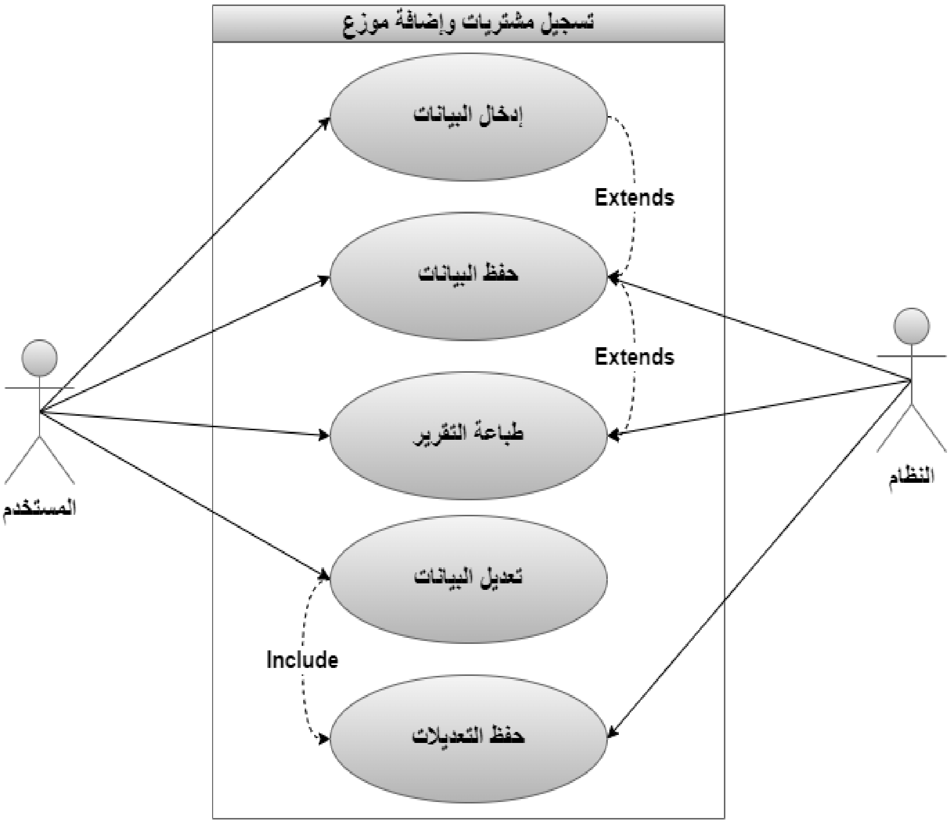


Use Case: Register Purchases and Add Supplier

No.	Use Case Name	Inputs	Processes	Outputs	Pre-conditions	Actors	Post-conditions
0	Register Purchases and Add Supplier	- Supplier Name- Phone- Email- Fax- Date- Status- Logo- Purchases Table: • Trade	- Enter data- Save data into the database - Print report	- Sales report- Error message	- Purchase data must be prepared (medicines to be added)- User must be logged	User, Supplier	- Data saved in the database- Report printed

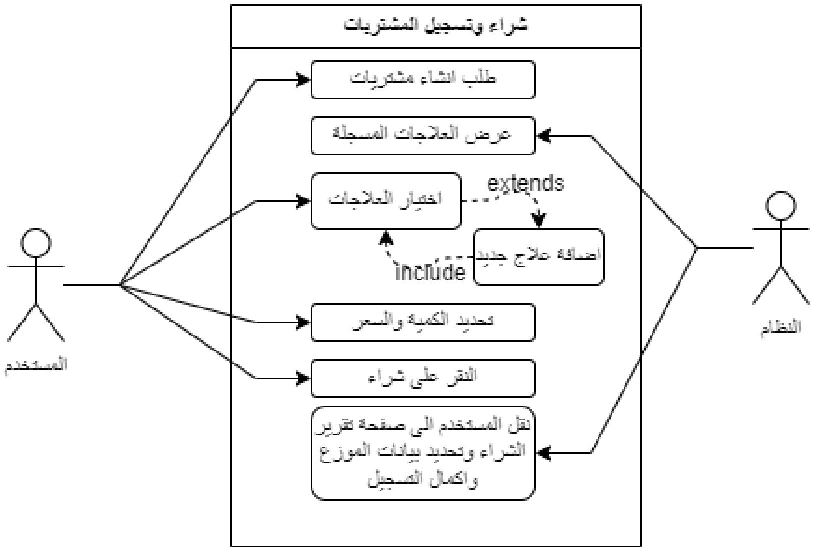
		Name • Generic Name • Price • Quantity • Total			into the system		

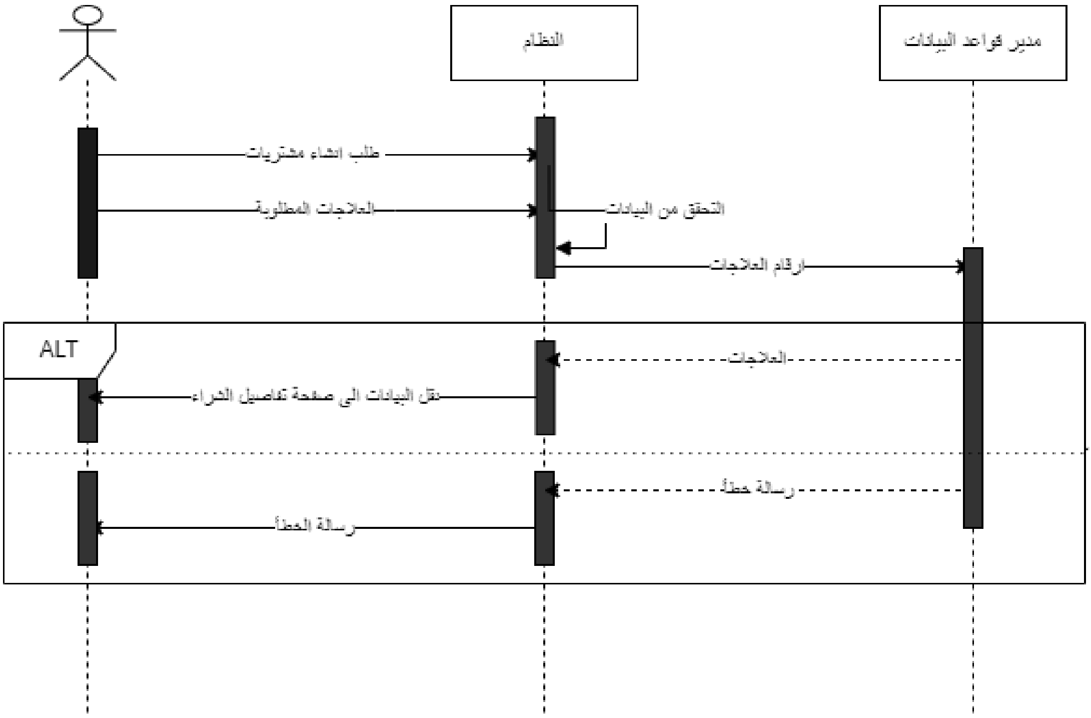
Add Supplier



Use Case: Sell and Register Purchases

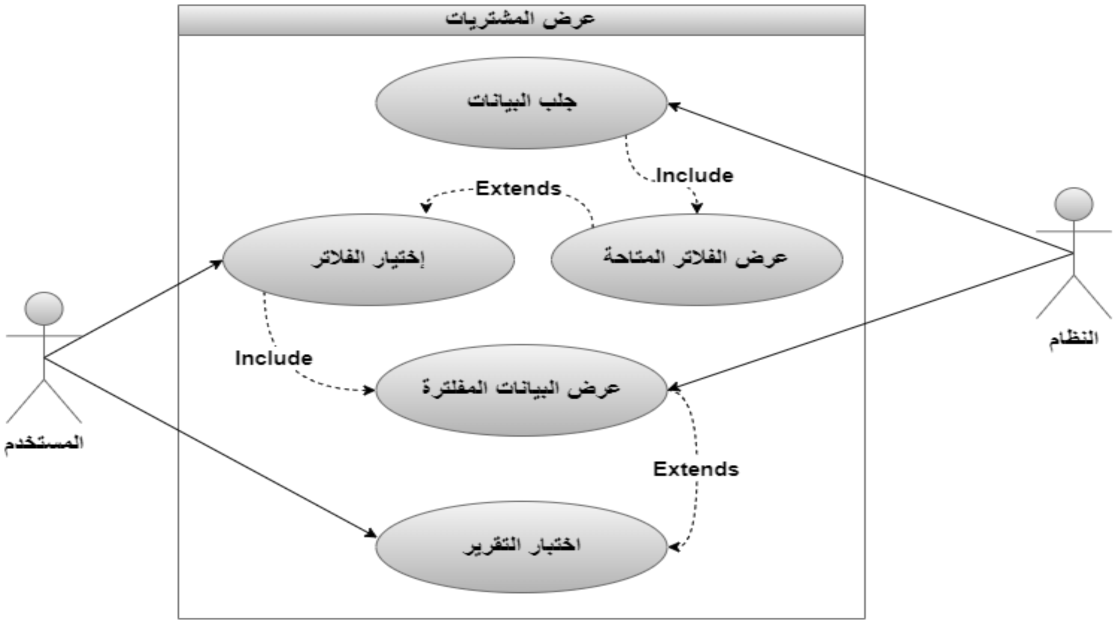
رقم العملية	3
اسم الحالة	Sell and Register Purchases
المدخلات	- Medicine Name (اسم العلاج) - Quantity (العدد) - Price (السعر)
العمليات	- Display available medicines (عرض العلاجات المتاحة) - Add medicine to the sales list (إضافة علاج إلى القائمة) - Set price and quantity (تحديد السعر والكمية) - Click "Sell" to register and print sales report (النقر على بيع لتسجيل وطباعة تقرير العملية)
المخرجات	- List of sales with prices and quantities (قائمة بالمبيعات والسعر والكمية)
الشروط المسبقة	- User must be logged into the system (مستخدم مُسجل دُخول في النظام)
الممثلين	System User (Worker, Manager) ((مستخدم النظام (عامل، مدير)
الشروط التابعة	- Sales list data transferred to sales confirmation interface and report printing (نقل البيانات إلى واجهة تأكيد البيع وطباعة التقرير)


شراء وتسجيل المشتريات
طلب انشاء مشتريات
عرض العلاجات المسجلة
اختيار العلاجات
extends
اضافة علاج جديد
include
تحديد الكمية والسعر
النقر على شراء
نقل المستخدم الى صفحة تقرير الشراء وتحديد بيانات الموزع واكمال التسجيل
المستخدم
النظام


النظام
مدير قواعد البيانات
طلب انشاء مشتريات
العلاجات المطلوبة
التحقق من البيانات
ارقام العلاجات
ALT
العلاجات
نقل البيانات الى صفحة تفاصيل الشراء
رسالة خطأ
رسالة الخطأ

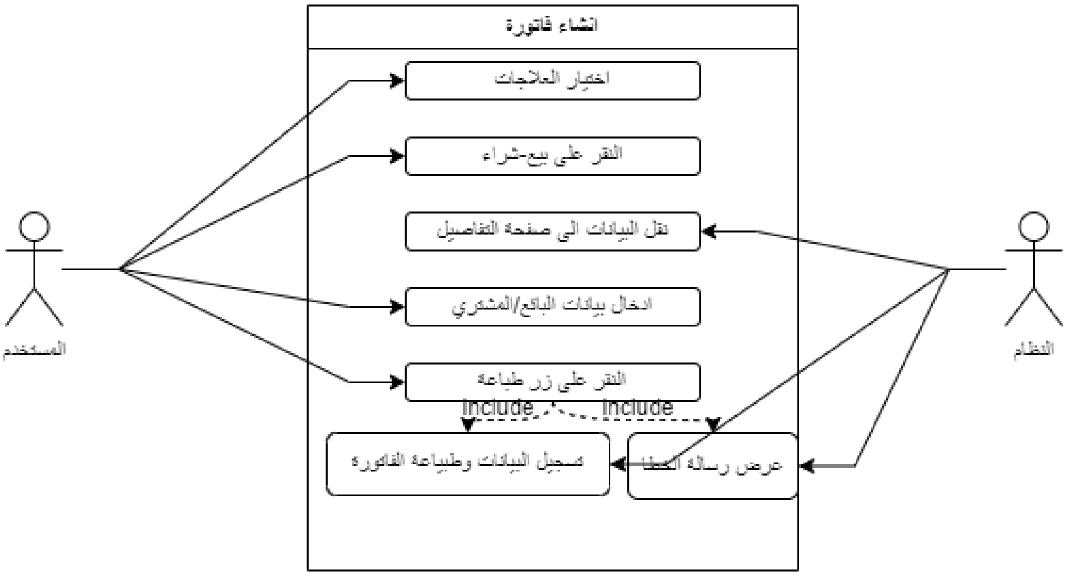
Use Case: Display Purchases

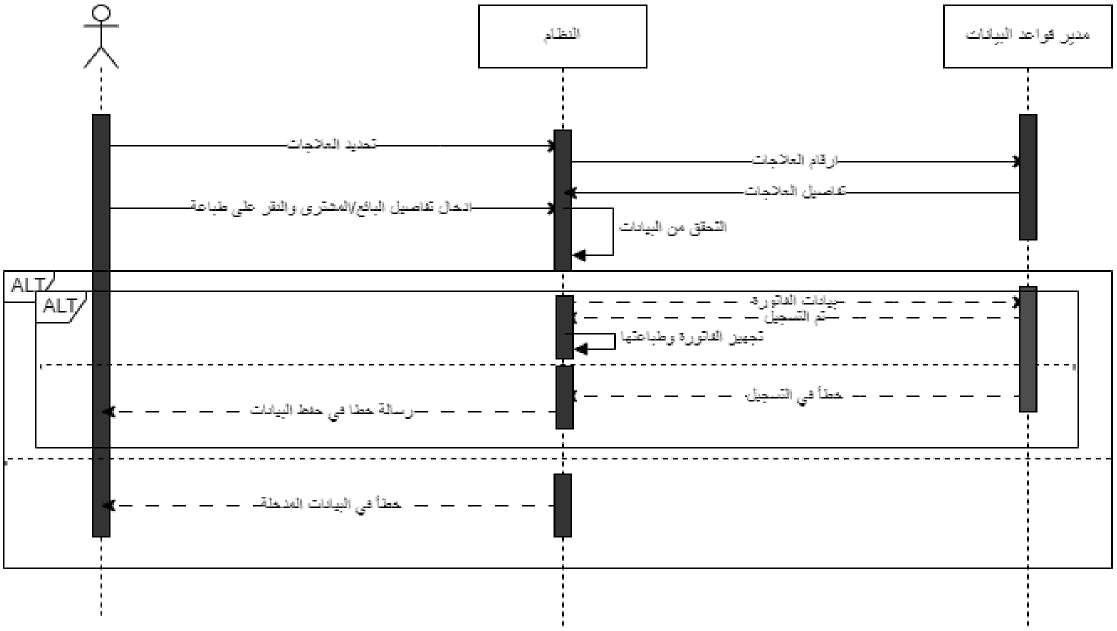
رقم العملية	1
اسم الحالة	Display Purchases (عرض المشتريات)
المدخلات	Filter data (بيانات الفلاتر)
العمليات	- Retrieve purchase data from the database (جلب بيانات الشراء من قاعدة البيانات) - Display filter options based on purchase data (عرض بيانات الفلاتر حسب بيانات الشراء) - Update displayed data based on selected filters (تحديث عرض البيانات حسب الفلاتر) - Double-click to edit or print a report (النقر المزدوج لتعديل أو طباعة تقرير)
المخرجات	- Purchase data (بيانات الشراء) - Purchase reports (تقارير الشراء) - Error messages (رسائل الخطأ)
الشروط المسبقة	Availability of purchase data in the database (توفر بيانات شراء في قاعدة البيانات)
الممثلين	User (Worker, Manager) ((المستخدم (عامل، مدير)
الشروط التابعة	None (----)



Use Case: Create Invoice

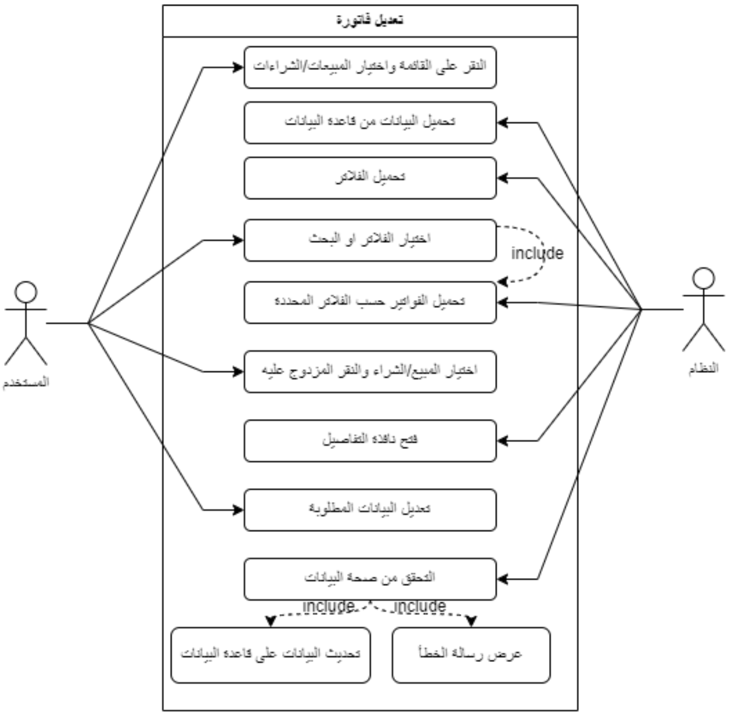
رقم العملية	10
اسم الحالة	Create Invoice (إنشاء فاتورة)
المدخلات	Medicines, Buyer/Seller data (العلاجات، بيانات المشتري/البائع)
الممثلين	System User (Worker, Manager) ((مستخدم النظام (عامل، مدير)
العمليات	- User selects the medicines (يقوم المستخدم باختيار العلاجات) - User clicks on Sell/Buy (يقوم بالنقر على بيع/شراء) - System transfers data to the details page (يقوم النظام بنقل البيانات إلى صفحة التفاصيل) - User fills out the form (يقوم المستخدم بتعبئة النموذج) - User clicks on Print (يقوم المستخدم بالنقر على طباعة) - System prints the invoice (يقوم النظام بطباعة الفاتورة)
المخرجات	Paper invoice (فاتورة ورقية)
الشروط المسبقة	Medicines data exists in the database, Buyer/Seller data exists (بيانات العلاجات في قاعدة البيانات، بيانات مشتري/بائع)
الشروط التابعة	Paper invoice printed, Invoice data saved in the database (فاتورة ورقية، بيانات الفاتورة مخزنة في قاعدة البيانات)

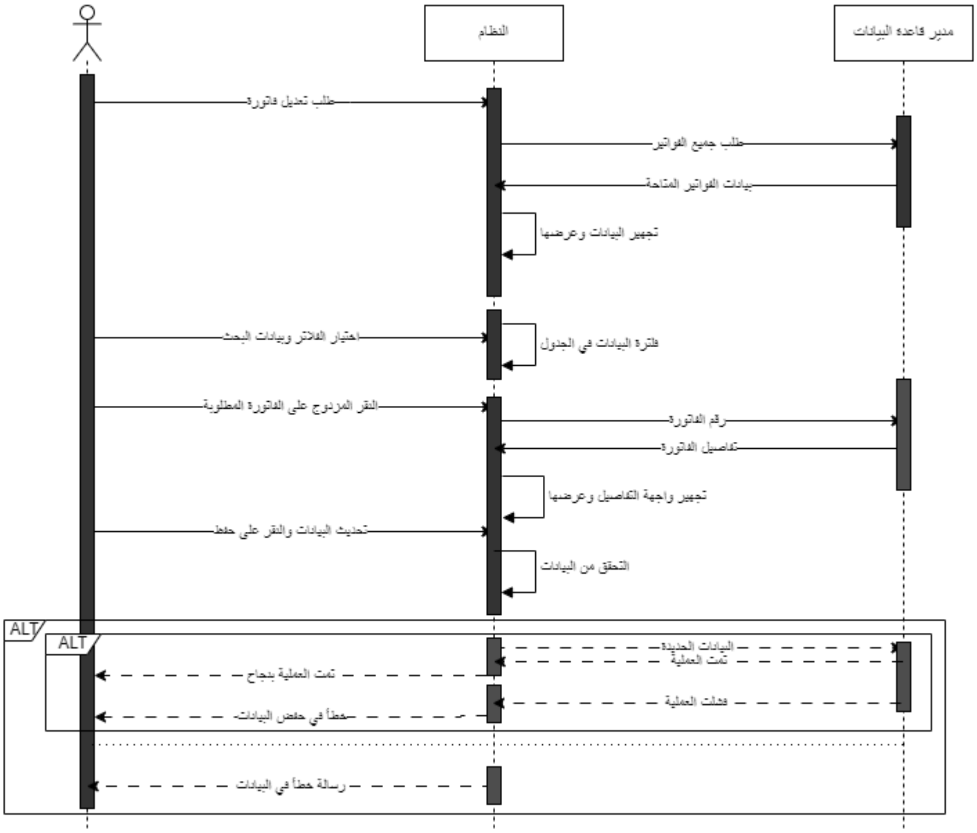




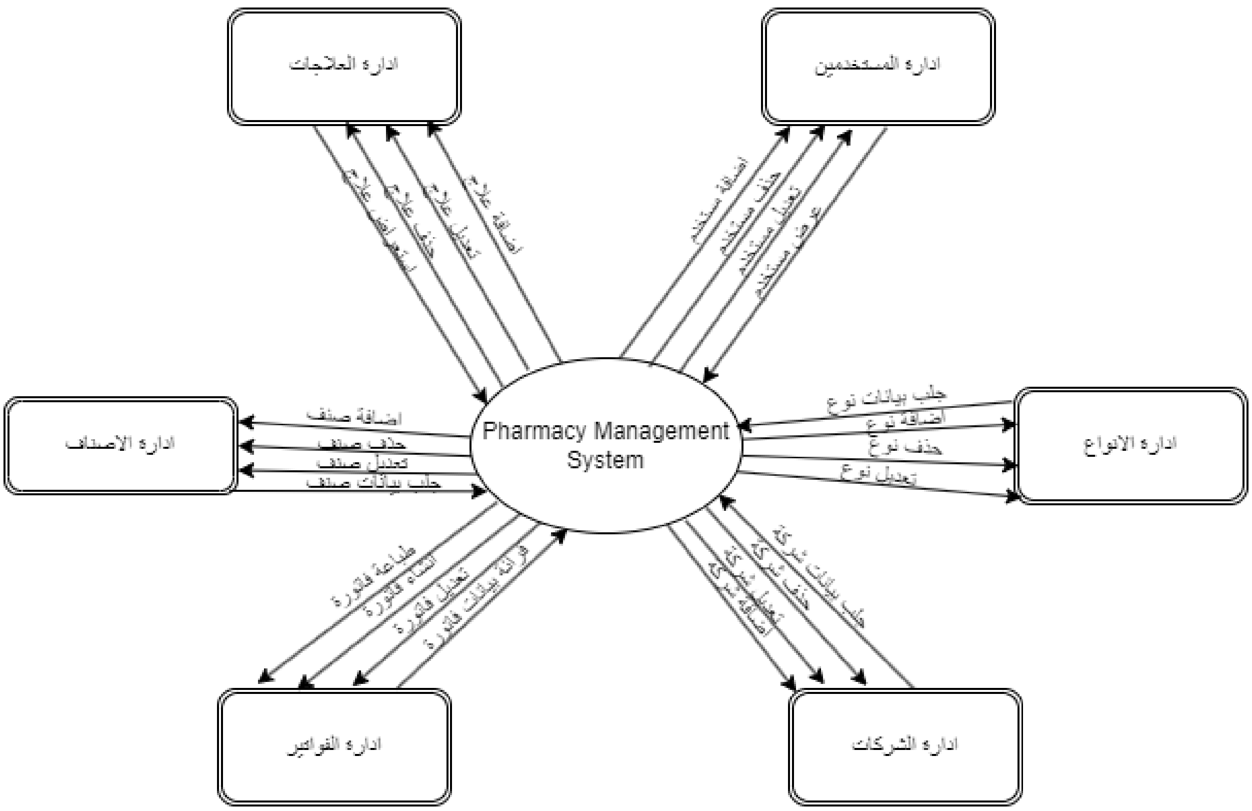
Edit Invoice

Process Number	10
Use Case Name	Edit Invoice
Inputs	Invoice Number
Actors	System User (Worker, Manager)
Operations	1. User selects Sales/Purchases.2. System retrieves data with appropriate filters.3. User selects filters to find the required invoice.4. System updates invoice display based on filters.5. User double-clicks the selected invoice.6. System opens details window.7. User edits the invoice.8. User clicks Save to save changes.9. System validates new inputs.10. System updates data or shows error message if any issue occurs.
Outputs	Paper invoice
Preconditions	Medicine data exists in the database.Buyer/Seller data exists.
Postconditions	Paper invoice generated.Invoice data saved in database.

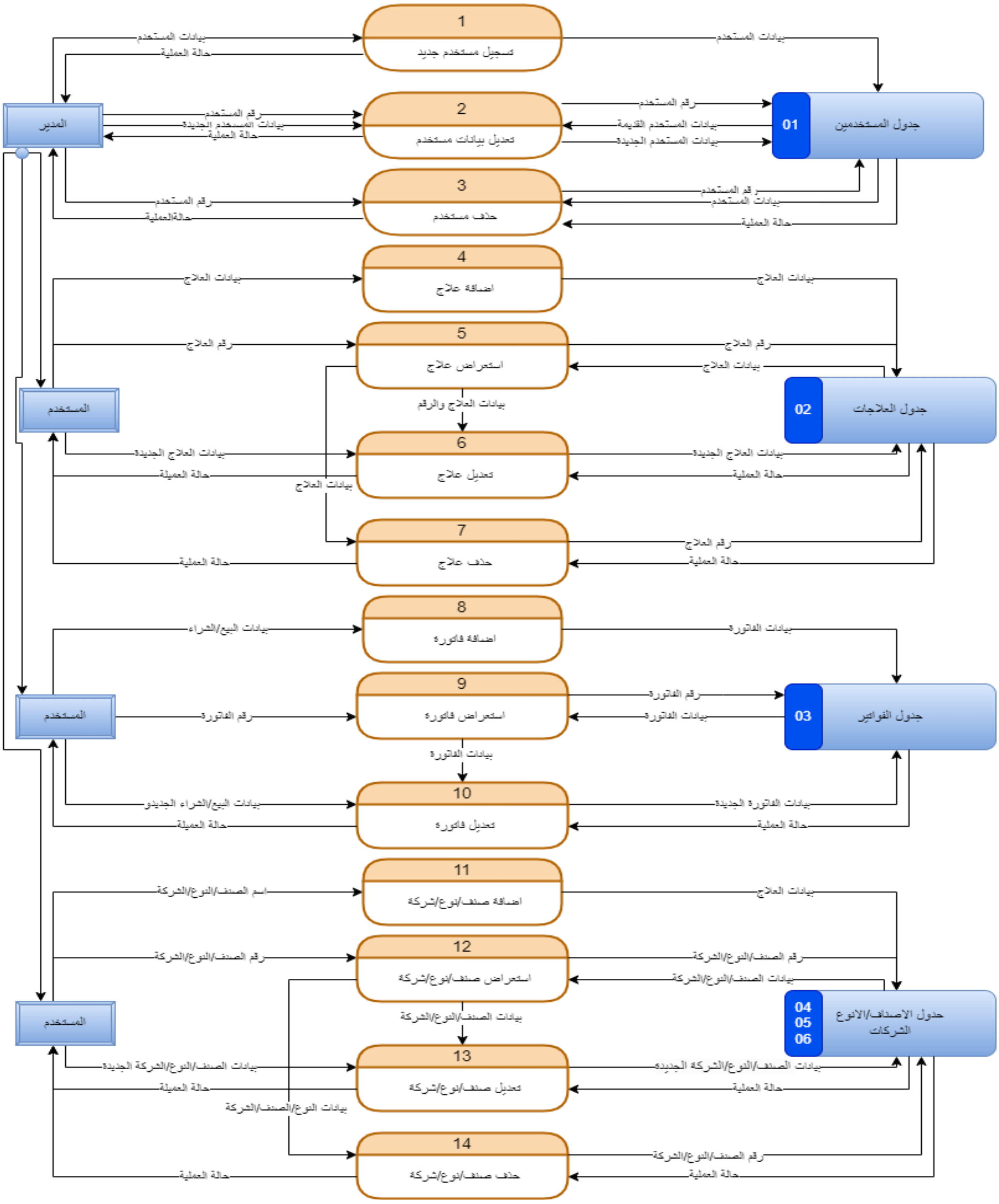

تعديل فاتورة
النقر على القائمة واختيار المبيعات/الشراءات
تحميل البيانات من قاعده البيانات
تحميل الفلاتر
اختيار الفلاتر او البحث
include
تحميل الفواتير حسب الفلاتر المحدده
اختيار المبيع/الشراء والنقر المزدوج عليه
فتح نافذه التفاصيل
تعديل البيانات المطلوبة
التحقق من صحة البيانات
include
include
تحديث البيانات على قاعده البيانات
عرض رسالة الخطأ
المستخدم
النظام


النظام
مدير قاعده البيانات
طلب تعديل فاتورة
طلب جميع الفواتير
بيانات الفواتير المتاحة
تجهيز البيانات وعرضها
اختيار الفلاتر وبيانات البحث
فلترة البيانات في الجدول
النقر المزدوج على الفاتورة المطلوبة
رقم الفاتورة
تفاصيل الفاتورة
تجهيز واجهة التفاصيل وعرضها
تحديث البيانات والنقر على حفظ
التحقق من البيانات
ALT
ALT
البيانات الجديدة
تمت العملية
تمت العملية بنجاح
فشلت العملية
خطأ في حفظ البيانات
رسالة خطأ في البيانات

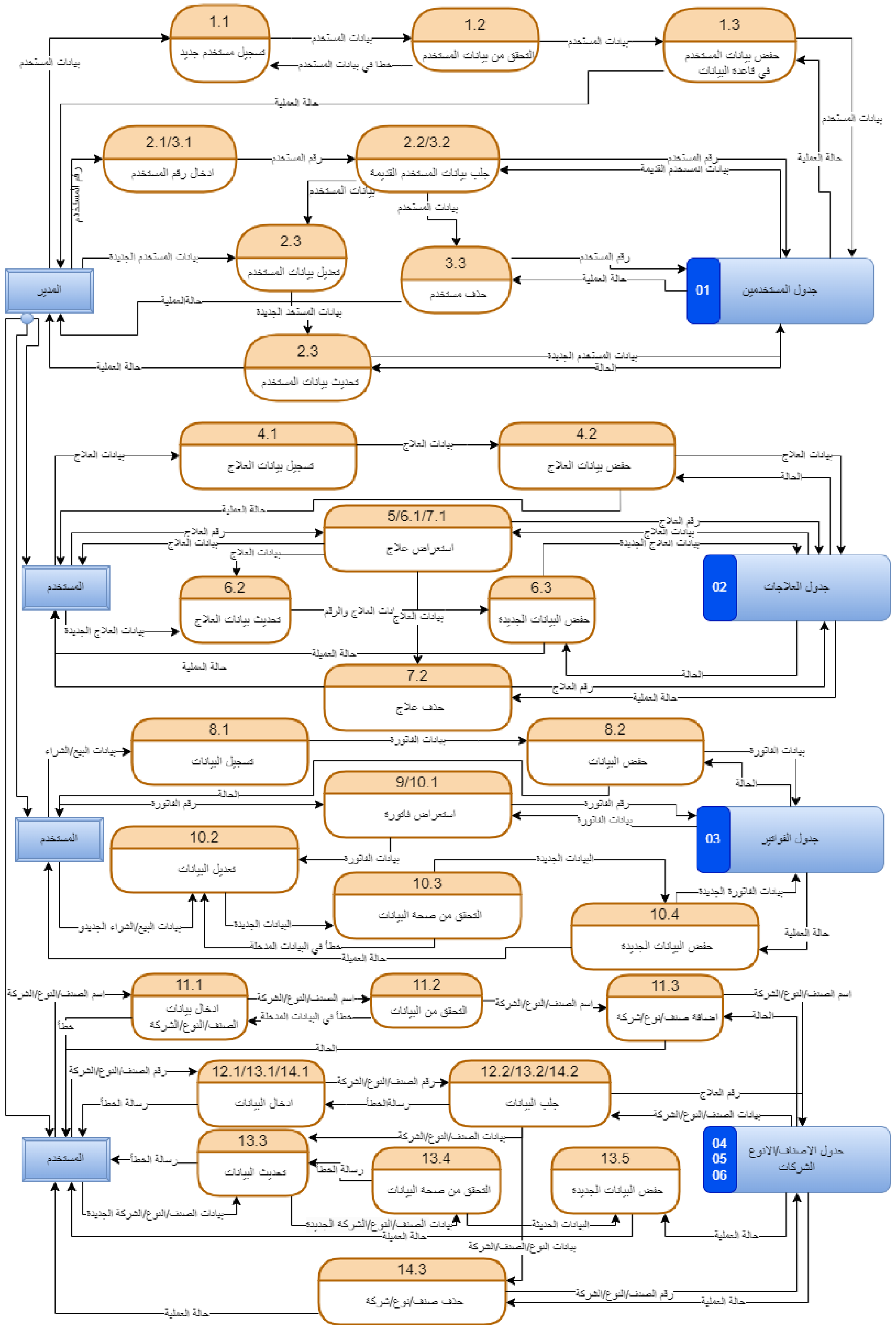
Context


Pharmacy Management System
اداره العلاجات
اضافة علاج
تعديل علاج
حذف علاج
استعراض علاج
اداره المستخدمين
اضافة مستخدم
حذف مستخدم
تعديل مستخدم
عرض مستخدم
اداره الاصناف
اضافة صنف
حذف صنف
تعديل صنف
جلب بيانات صنف
اداره الانواع
جلب بيانات نوع
اضافة نوع
حذف نوع
تعديل نوع
اداره الفواتير
طباعة فاتورة
انشاء فاتورة
تعديل فاتورة
قراءة بيانات فاتورة
اداره الشركات
جلب بيانات شركة
حذف شركة
تعديل شركة
اضافة شركة

مخطط تدفق البيانات المستوى1



مخطط تدفق البيانات المستوى2


1.1
تسجيل مستخدم جديد
1.2
التحقق من بيانات المستخدم
1.3
حفض بيانات المستخدم في قاعدة البيانات
بيانات المستخدم
خطا في بيانات المستخدم
حالة العملية
2.1/3.1
ادخال رقم المستخدم
2.2/3.2
جلب بيانات المستخدم القديمة
رقم المستخدم
بيانات المستخدم القديمة
2.3
تعديل بيانات المستخدم
3.3
حذف مستخدم
بيانات المستخدم الجديدة
حالةالعملية
بيانات المستخد الجديدة
2.3
تحديث بيانات المستخدم
الحالة
المدير
01
جدول المستخدمين
4.1
تسجيل بيانات العلاج
4.2
حفض بيانات العلاج
بيانات العلاج
الحالة
5/6.1/7.1
استعراض علاج
رقم العلاج
بيانات العلاج الجديدة
المستخدم
02
جدول العلاجات
6.2
تحديث بيانات العلاج
6.3
حفض البيانات الجديدة
بيانات العلاج والرقم
حالة العميلة
7.2
حذف علاج
8.1
تسجيل البيانات
8.2
حفض البيانات
بيانات البيع/الشراء
بيانات الفاتورة
9/10.1
استعراض فاتوره
رقم الفاتورة
03
جدول الفواتير
10.2
تعديل البيانات
10.3
التحقق من صحة البيانات
10.4
حفض البيانات الجديدة
البيانات الجديدة
بيانات الفاتورة الجديدة
بيانات البيع/الشراء الجديدو
خطأ في البيانات المدخلة
11.1
ادخال بيانات الصنف/النوع/الشركة
11.2
التحقق من البيانات
11.3
اضافة صنف/نوع/شركة
اسم الصنف/النوع/الشركة
خطأ
12.1/13.1/14.1
ادخال البيانات
12.2/13.2/14.2
جلب البيانات
رقم الصنف/النوع/الشركة
رسالة الخطأ
رسالةالخطأ
رقم العلاج
بيانات الصنف/النوع/الشركة
04
05
06
جدول الاصناف/الانوع الشركات
13.3
تحديث البيانات
13.4
التحقق من صحة البيانات
13.5
حفض البيانات الجديدة
بيانات الصنف/النوع/الشركة الجديدة
البيانات الحديثة
بيانات النوع/الصنف/الشركة
14.3
حذف صنف/نوع/شركة