جامعة بوليتيكنيك فلسطين

الخليل – فلسطين

كلية الهندسة والتكنولوجيا

دائرة الهندسة الكهربائية والحاسوب

اسم المشروع

المدير المالي الشخصي

أسم الطالب

عمر محمد عباس القاضي

بناء على نظام كلية الهندسة والتكنولوجيا وإشراف ومتابعة المشرف المباشر على المشروع وموافقة أعضاء اللجنة الممتحنة , تم تقديم هذا المشروع إلى دائرة الهندسة الكهربائية والحاسوب ذلك للوفاء بمتطلبات درجة البكالوريوس في الهندسة تخصص هندسة كهربائية فرع هندسة أنظمة الحاسوب .

توقيع مشرف المشروع

.....

توقيع اللجنة الممتحنة

.....

.....

توقيع رئيس الدائرة

جامعة بوليتكنك فلسطين
Palestine Polytechnic University
(PPU)
The Library — المكتبة
Acc. ١٢٦٢٢ الرقم المتسلسل
Class. ..... رقم التصنيف

الملـــخص

نظام المدير الشخصي هو نظام يقوم على تحقيق مجموعة من الخدمات , التي تهدف في النهاية إلى إعانة المدير حصوله على القدر الكافي من المعلومات التي يمكن أن يحتاج إليها لمعرفة مختلف الأمور التي لها علاقة بالتعاملات التجارية , و غيرها من الأمور و ذلك من خلال شبكة الانترنت , و هذا من جهة , و أما من جهة أخرى فهو نظام يساعد المتعامل على تسهيل عملياته التجارية مع الشركة , و كذلك يساعد الموظف داخل الشركة في التعاملات التجارية ومعرفة وضعه داخل الشركة.

و حتى يتم إنشاء و تحقيق هذا النظام , كان لا بد من التعامل مع مجموعة من الأمور و منها جمع المعلومات التي لها علاقة بالخدمات التي سيتم بناؤها .

Abstract

This personal manager system aim to realization a set of service , that aim in the finish to support the manager to get on complete information about trade transaction , other important information that can the manager get it from internet , in other side this system helps customers to trade communication with company and helps employees in the company to make relationship with other countries and the information of them

And to attainment this system we must deal with a set of objects that have a relationship with system as collect information and other to build it .

جدول المـــــــــحتويات
(Table of contents)

صفحة عنوان المشروع و تواقيع المشرفين و رئيس القسم I
صفحة العنوان الرئيسية II
صفحة الإهداء III
صفحة الشكر و التقدير IV
صفحة التلخيص V
Abstract VI
جدول المـــــــــحتويات(Table of contents) VII
فصول المشروع ..... (1-102)
المراجع ..... (103)
الملحقات ..... (105)

الفـصل الأول مقـــــــدمة(Introduction)

1.1 لفكرة العامة وأهمية المشروع (eneral idea about project and its importance) 1
1.1.1 الفكرة العامة للمشروع 1
1.1.2 أهمية المشروع(project and its importance) 2
1.2 الدراسات السابقة(Literature review) 3
1.3 تكاليف النظام المقدرة (Estimated Cost) 3
1.3.1 تكلفة الأجهزة(Hardware) 4
1.3.2.تكلفة البرمجة(Software Cost) 5
1.3.3 تكلفة المصادر البشرية(Human Resources cost) 6
1.4 المخطط الزمني (Time plan / project schedule) 6
1.4.1 جدولة المشروع (Project Schedule) 6
1.4.1.1 الفترات الزمنية لكل مهمة (Task duration and schedule) 7
1.4.1.2 تحديد المهمة لكل شخص ((Allocation of activity to Engineer(s)) 8
1.5 إدارة المخاطر(Risk management) 9
1.6محتويات التقرير (Report contents) 10

الفصل الثـاني تحديد متطلبات النظام (System requirement specification)

2.1المواضيع النظرية للمشروع (Theoretical subjects related to the main ideas of the project) 21
2.1.1 قواعد البيانات 12

2.1.2 UML 12
2.1.3 المجموعات الموجودة في الحسابات المالية 13
2.2 Information about special components 13
2.2.1 Microsoft Office 2003 13
2.2.2 Microsoft Visual Studio .NET 2003 14
2.2.3 Microsoft SQL Server 2000 15
2.3 متطلبات النظام (System requirements) 15
2.3.1 تعريف النظام (System Definition) 15
2.3.2 متطلبات النظام الوظيفية (Functional Requirements) 15
2.3.3 متطلبات النظام غير الوظيفية (Non-Functional Requirements) 16
2.3.3.1 متطلبات المشروع(Product Requirements) 17
2.3.3.2 الافتراضات و الاعتماديات (Assumption and Dependencies) 19

الفصل الثالث التصميم البنـــــــــائي للنظام (Architectural Design). 20
3.1 أهداف النظام (System Objectives) 20
3.2 الشكل العام للنظام (General Block Diagram) 21
3.3 كيفية عمل النظام (How system Works) 23
3.4 مخططات نماذج النظام (System Modeling) 25
3.4.1 مخططات العمليات النشطة (Activity Diagrams) 25
3.4.2 مخطط حالات الاستخدام (Use Cases). 35
3.4.3 مخططات تتابع عمليات النظام (Sequence Diagrams) 37
3.4.4 مخطط حالات النظام (State Diagram) 46

الفصل الرابع تصميم النظام(System Design) 47
4.1 خيارات تصميم النظام 47
4.1.1 نظام إدارة قواعد البيانات Microsoft SQL Server 2000 47
4.1.2 بيئة البرمجة ASP.NET 48
4.1.3 لغة النظام (اللغة العربية) 48
4.1.4 اختيار بناء للمشروع وليس تطوير على مشاريع سابقة 49
4.1.5 لماذا تم اختيار المشروع 49
4.2 معلومات تفصيلية عن مكونات النظام 50

4.2.1 مكونات النظام..... 51
4.2.1.1 التطبيع..... 51
4.2.1.2 تصميم الكائنات المكونة للنظام..... 54
4.2.1.2.1 الكائنات المجردة بدون وجود علاقات بينها أو عناصر بداخلها..... 55
4.2.1.2.2 الارتباطات بين الكائنات..... 56
4.2.1.2.3 الارتباطات بين الكائنات بوجود العناصر المكونة لهذه الكائنات..... 57
4.2.1.2.4 الارتباطات بين الكائنات بوجود العناصر المكونة لهذه الكائنات والعمليات 58
4.2.1.3.1 جداول قواعد البيانات (Database_Tables)..... 59
4.3 واجهات التطبيق للمستخدم (GUI) 63
4.4 المخططات الانسيابية و الخوارزميات 82
4.4.1 المخططات الانسيابية على تشفير كلمة المرور المدخلة..... 82

الفصل الخامس: البناء التطبيقي للنظام و فحص النظام(Implementation & Testing)

5.1مقدمة (Introduction)..... 84
5.2 متطلبات النظام من أنظمة التشغيل و الاجهزة (Hardware & O.S)..... 85
5.3أمثلة على بناء اجزاء النظام برمجياً (Implementation Examples)..... 85
5.4اختبار آلية عمل النظام (Testing)..... 95

الفصل السادس الاستنتاجات و الأعمال المستقبلية (Conclusions & Future Works)

6.1 الإستنتاجات(Conclusions)..... 100
6.2 الاعمال التطويرية والمستقبلية (Future Work)..... 101

قائمة الجداول
(List of Tables)

رقم الجدول	اسم الجدول:	صفحة
1.1	تكلفة الأجهزة	4
1.2	تكاليف البرمجيات	5
1.3	تكلفة المصادر البشرية	6

1.4	جدولة المشروع	6
1.5	الفترات الزمنية لكل مهمة	7
1.6	أنواع المخاطر (Types of Risks)	9

قائمة الأشكال والرسومات
(List Of Figures)

رقم الشكل	اسم الشكل :	صفحة
1.1	المخطط الزمني للمشروع	8
3.1	الشكل العام للنظام (General Block Diagram)	22
3.2	(Activity Diagrams) المستخدم مع الصفحة الرئيسية للنظام	26
3.3	(Activity Diagrams) المدير مع الصفحة التابعة لبيانات السلع	27
3.4	(Activity Diagrams) المستخدم مع الصفحة التابعة للعمليات المالية	28
3.5	(Activity Diagrams) للمستخدم مع الصفحة التابعة لبيانات المتعاملين	29
3.6	Activity Diagrams) المدير مع الصفحة التابعة لبيانات المتعاملين	30
3.7	(Activity Diagrams) للمستخدم المتعامل مع الصفحة التابعة للبضائع	31
3.8	(Activity Diagrams) للمستخدم المتعامل مع الصفحة التابعة لتحديد طلبية وارسالها	32
3.9	(Activity Diagrams) للمستخدم العامل مع الصفحة التابعة لفاتورة بيع	33
3.10	(Activity Diagrams) للمستخدم العامل مع الصفحة التابعة لتحديد فاتورة شراء	34
3.11	المخطط الاستخدامي (Use Case) للمدير والعامل والمتعامل	35
3.12	المخطط الاستخدامي (Use Case) لزبون الشركة وصفحة المشتريات وصفحة المبيعات	36
3.13	مخطط تتابع العمليات (Sequence diagram) التي يقوم بها المستخدم مع الصفحة الرئيسية	37
3.14	مخطط تتابع عمليات المستخدم مع صفحة بيانات السلع	38
3.15	مخطط تتابع عمليات المستخدم مع صفحة بيانات العاملين	39
3.16	مخطط تتابع العمليات المستخدم مع صفحة لبيانات المتعاملين	40
3.17	مخطط تتابع العمليات المستخدم مع صفحة العمليات المالية	41
3.18	مخطط تتابع العمليات للزبون مع صفحة قسم المشتريات	42
3.19	مخطط تتابع العمليات للزبون مع صفحة عرض البضائع	43
3.20	مخطط تتابع العمليات للعامل مع صفحة فاتورة بيع	44
3.21	مخطط تتابع العمليات للعامل مع صفحة فاتورة شراء	45
3.22	(State Diagram) المستخدم مع الدخول الى النظام	46

4.1	الكائنات المجردة دون وجود علاقات بينها أو عناصر داخلها	55
4.2	الارتباطات بين الكائنات	56
4.3	الارتباطات بين الكائنات بوجود عناصرها.	57
4.4	الارتباطات والعناصر والعمليات على الكائنات	58
4.5	واجهة صفحة الدخول الى النظام	64
4.6	واجهة صفحة المدير	65
4.7	واجهة اضافة منتج	66
4.8	واجهة ازالة منتج	66
4.9	واجهة البحث عن منتج	67
4.10	واجهة تعديل محتويات السلعة	68
4.11	واجهة اضافة الحسابات البنكية	69
4.12	إدراج حركة بنكية إلى حساب موجود مسبقا	69
4.13	واجهة إضافة بيانات المتعاملين	70
4.14	واجهة بحث عن بيانات المتعاملين	71
4.15	واجهة تعديل بيانات المتعاملين	72
4.16	واجهة حذف بيانات المتعاملين	73
4.17	واجهة اضافة بيانات الموظفين	74
4.18	واجهة بحث في بيانات الموظفين	75
4.19	واجهة تعديل بيانات الموظفين	76
4.20	واجهة حذف بيانات الموظفين	77
4.21	واجهة تعديل كلمة مرور	78
4.22	واجهة صفحة الموظف	78
4.23	واجهة فاتورة تصدير	79
4.24	واجهة فاتورة شراء	79
4.25	واجهة صفحة المتعاملين	80
4.26	واجهة تحديد طلبية	80
4.27	واجهة عرض البضاعة	81
4.28	المخططات الانسيابية على تشفير كلمة المرور المدخلة	82
5.1	الرسالة الناتجة من عدم إدخال كلمة المرور أو اسم المستخدم على صفحة الدخول الى النظام	95
5.2	الرسالة الناتجة توضح عدم إدخال التاريخ بشكل صحيح	96
5.3	الرسالة الناتجة توضح اختيار الحالة الاجتماعية للموظف أعزب وله اولاد	97
5.4	الرسالة الناتجة عن ادخال تاريخ الانتهاء قبل تاريخ الانتاج	98
5.5	الرسالة الناتجة عن إدخال البريد الالكتروني بشكل غير صحيح	99

الفصل الأول

المقدمة

1.1الفكرة العامة وأهمية المشروع (General idea about project and its importance)

1.1.1 الفكرة العامة للمشروع:

هذا المشروع عبارة عن تطبيق إداري وحسابي حيث انه سيتكون من ثلاثة أجزاء رئيسية وهي:

-قسم المدير:حيث يكون من صلاحيته إدخال البيانات للمتعاملين والموظفين وبيانات السلع وإدخال حسابات مصرفية.

-قسم المتعاملين:حيث يسمح له بتحدي طلبيه يريدها وعرض للمنتجات الشركة وما لديها من سلع والتعرف على تعاملاته مع الشركة.

_قسم الموظف: حيث يكون من صلاحيته إدخال فواتير للشراء والبيع وعرض للطلبيات ومعرفة وضع الموظف داخل الشركة.

1.1.2 أهمية المشروع(project and its importance)

وتكمن أهمية هذا المشروع في إمكانية تعامل شركة معينة مع أي زبون مستورد أو مورد عن طريق الانترنت وكذلك تعامل الشركة مع الموظفين .

حيث أن أهميته تظهر في

- السهولة في التطبيق : وذلك بتسهيل العمليات الإدارية والحسابية البسيطة.
- أن يحقق الدقة التامة في تنفيذ العمليات الإدارية والحسابية البسيطة.
- أن يحقق السرعة في انجاز العمليات الإدارية والحسابية البسيطة.
- أن يكون مرنا قابلا للتعديل.
- أن يسهل حركة التعامل بين الشركة والزبائن.

كما سيقوم المشروع بتوفير العديد من الخدمات الموجودة على شبكة أل Webنذكر منها لا الحصر:
إنشاء خدمة Web خاصة بقواعد البيانات عامة الغرض.

1.2الدراسات السابقة(Literature review)

يوجد دراسات قليلة عن فكرة هذا المشروع حيث أنه يوجد هناك دراسات متعلقة بالتطبيقات المالية كعمل نموذج server/clientحيث تتشارك جميع الوحدات بقاعدة بيانات مركزية وتتصل فيما بينها بنوع معين من بروتوكولات الاتصال الشبكية ولكن هذا النظام غير قادر على التعامل مع شبكة الانترنت.

حيث أننا لن نحتاج إلى مختلف الوحدات المكونة للتطبيق المالي لأنها تتصل فيما بينها بشكل فيزيائي لذلك سوف نستخدم بيئة ASP.NET (Active Server Pages) و سوف يتم بناء قواعد البيانات باستخدام SQL وخدمات web لتصميم تطبيق يمكنه إرسال واستقبال البيانات باستخدام الانترنت وبروتوكول HTTP.

1.3 التكاليف المقدرة للنظام(Estimated Cost)

لحساب التكلفة الإجمالية للنظام لا بد من الأخذ بعين الاعتبار مصادر النظام بأنواعها البشرية والبرمجة والأجهزة.وبناءً على ذلك تقسم التكلفة إلى أقسام رئيسية هي تكلفة الأجهزة وتكلفة البرمجة وتكلفة المصادر البشرية و تكاليف أخرى.

1.3.1 تكلفة الأجهزة(Hardware):

جميع ما سيذكر في الجداول سيتم استئجاره.

جدول (1.1): تكلفة الأجهزة

الجهاز	العدد	تكلفة كل قطعة
جهاز حاسوب Pentium4,2400MHz 40GB HDD,128 MHz RAM ,Monitor 17",Keyboard and mouse	1	200دولار
طابعه	1	100 دولار
المجموع		300دولار

1.3.2.تكلفة البرمجة(Software Cost):

يمكن تقدير تكلفة البرمجيات اللازمة كما هو موضح في الجدول

جدول (1.2): تكلفة البرمجيات

البرنامج	تكلفته
ويندوز XP Professional	50 دولار
مايكروسوفت أوفيس XP	50 دولار
مجموعة أسطوانات Microsoft Visual Studio.Net	10 دولار
برنامج Microsoft Visio 2003	100 دولار
DBMS (MS SQL Server 2000)	400 دولار
المجموع	610 دولار

1.3.3 تكلفة المصادر البشرية(Human Resources cost):

جدول(1.3): تكلفة المصادر البشرية

عدد المطورين	ساعات العمل(شهريا)	التكلفة لكل ساعة	التكلفة الكلية(شهريا)
1	90	$10	$900

1.4 المخطط الزمني (Time plan / project schedule)

1.4.1 جدولة المشروع (Project Schedule)

جدول (1.4):جدولة المشروع

Task ID	Description
1	جمع معلومات عامة عن المشروع هذا يتضمن البحث في جميع المصادر الموجودة في المكتبة أو بالانترنت عن كل ما يتعلق بفكرة المشروع
2	تحديد الخدمات الأساسية التي سيقوم النظام بتقديمها
3	تحديد ما هي متطلبات النظام من مكونات مادية أو من مكونات برمجية
4	تحليل وتحديد أهم المخاطر المتوقعة في حالة تطبيق المشروع
5	وضع جدولة كاملة للطريقة التي سيتم فيها تنفيذ هذا المشروع
6	تحديد system requirements & user requirements &functional requirements
7	تحديد كيفية عمل هذا النظام في حالة تطبيقه

8	القيام بعمل modeling كامل لل system
9	تحديد الخيارات المقترحة لعملية التصميم الكامل للنظام
10	بداية التصميم الفعلي للنظام من عمل تصميم داخلي للنظام وتصميم واجهات(GUI)
11	كتابة وعمل الخوارزميات اللازمة لعمل هذا النظام
12	بداية عمل تطبيق كامل للمشروع
13	إجراء عمليات ال (testing) اللازمة لان يعمل هذا النظام بشكل كامل
14	Documentation

1.4.1.1 الفترات الزمنية لكل مهمة (Task duration and schedule)

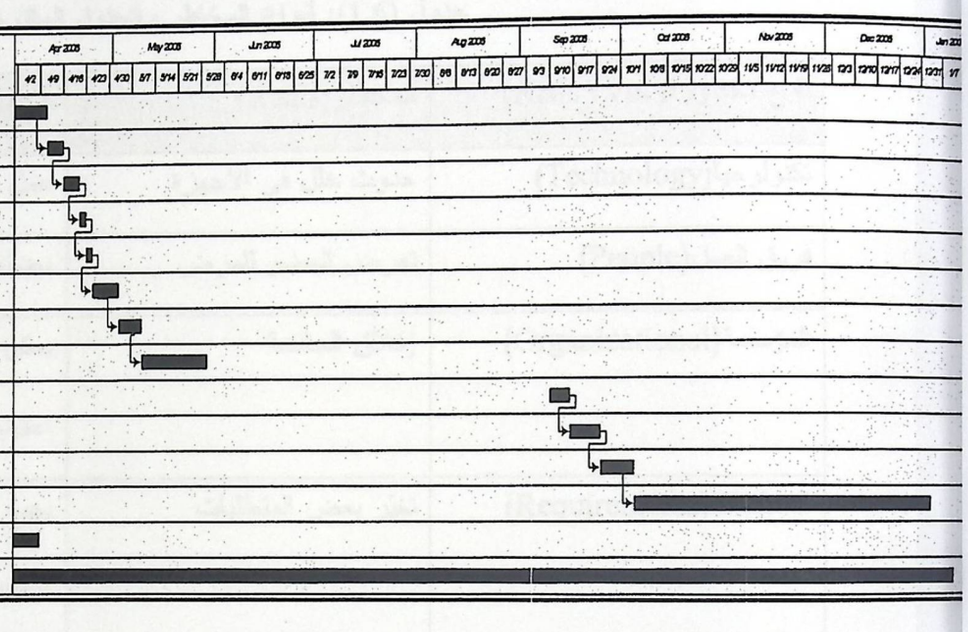
جدول (1.5) :الفترات الزمنية لكل مهمة

Task	Duration(days)
T1	10
T2	5
T3	5
T4	2
T5	2
T6	8
T7	7
T8	20
T9	6
T10	9
T11	10
T12	90
T13	8
T14	175

1.4.1.2 تحديد المهمة لكل شخص (Allocation of activity to Engineer(s))

جميع المهمات سيقوم بها عمر لوجوده وحيدا في المشروع.

ID	Task Name	Start	Finish	Duration
1	Task 1	4/1/2008	4/10/2008	10d
2	Task 2	4/11/2008	4/15/2008	5d
3	Task 3	4/16/2008	4/20/2008	5d
4	Task 4	4/21/2008	4/22/2008	2d
5	Task 5	4/23/2008	4/24/2008	2d
6	Task 6	4/25/2008	5/2/2008	8d
7	Task 7	5/3/2008	5/9/2008	7d
8	Task 8	5/10/2008	5/29/2008	20d
9	Task 9	9/10/2008	9/15/2008	6d
10	Task 10	9/16/2008	9/24/2008	9d
11	Task 11	9/25/2008	10/4/2008	10d
12	Task 12	10/5/2008	1/2/2007	90d
13	Task 13	4/1/2008	4/8/2008	8d
14	Task 14	4/1/2008	1/10/2007	285d



شكل (1.1) : المخطط الزمني للمشروع

1.5 إدارة المخاطر(Risk management)

هذه بعض المخاطر التي من المحتمل أن يواجهها النظام مع بعض الحلول المقترحة لتقليل تـأثير هذه المخاطر كما هو موضح في الجدول التالي:

جدول (1.6): أنواع المخاطر والحلول المقترحة

نوع الخطورة(Risk Type)	المخاطر(Risks)	الحلول المقترحة(Strategies)
تكنولوجيا(Technology)	حدوث خلل في الأجهزة	عمل ما يسمى ب Backup Plan
فريق العمل(People)	تعرض العضو للمرض	مضاعفة عدد ساعات العمل يوميا
المؤسسة(Organizational)	إغلاق الجامعة	يمكن يكتمل العمل في أي مكان غير الجامعة مع كل احتياجاته لإنجاز العمل على أكمل وجه.
المتطلبات(Requirements)	تغيّر بعض المتطلبات	يجب أن نكون على اتصال دائم مع المؤسسة لمعرفة التغيير في وقت مبكر.
أدوات(Tools)	قد نحتاج إلى استخدام تقنيات ليس لديهم الخبرة الكافية فيها	الأخذ بعين الاعتبار هذه النقطة وجعل وقت فائض بعد كل عملية للتدريب على أي تقنية جديدة
تقدير(Estimation)	قد تكون بعد التقديرات غير صحيح	يجب عمل فحص للنظام بعد كل مرحلة من مراحل البناء والتطوير.

1.6محتويات التقرير (Report contents)

يتكون هذا التقرير من ست وحدات ،حيث سيتم العمل على انجاز القسم الأول من المشروع من الوحدة الأولى وجزء من الوحدة الرابعة ضمن الجدول الزمني الأول للمشروع ،وإتمام المشروع الوحدة الخامسة والوحدة السادسة في الجدول الزمني الثاني المقدر للمشروع.

في الوحدات الأربعة الأولى سيتم توضيح الأمور التالي:

-في الوحدة الأولى :سيتم شرح أهمية المشروع ومدى الحاجة إليه في السوق ، كما سيتم التطرق في هذه الوحدة إلى الدراسات السابقة ،إضافة إلى توضيح التكاليف المادية التي يتطلبها هذا المشروع منذ لحظة البداية وحتى الانتهاء منه، ومن محتويات الوحدة الأولى الخطة الزمنية والجدول الزمني الذي سوف يتم العمل وفقه لانجاز هذا المشروع ، وتنتهي هذه الوحدة بشرح المخاطر التي يمكن ان يتعرض لها المشروع أو القائمين عليه.

في الوحدة الثانية : تبدأ هذه الوحدة بالجانب النظري الذي يعتمد عليه المشروع مع بيان كيفية العمليات التي سوف يتم استخدامها لانجاز هذا المشروع ، بعد ذلك سيتم شرح متطلبات النظام بنوعيها الوظيفي وغير الوظيفي ووضع العلاقة بين المتطلبات التي يحتاجها النظام والنتائج التي يخرجها المشروع.

الوحدة الثالثة: تتطرق هذه الوحدة إلى الأهداف المتوقع انجازها بعد إتمام هذا المشروع والتي سوف يتم العمل من اجل تلبيتها، وبعد ذلك سيتم وضع رسمه توضح الشكل العام للنظام
وبعد ذلك توضيح آلية عمل النظام ، وأما الجزء الأخير في هذه الوحدة سوف يتطرق إلى توضيح عمل الأقسام التي سيتكون منها النظام الكلي باستخدام مجموعة من النماذج والمخططات التوضيحية.

الوحدة الرابعة: هذه الوحدة هي الوحدة التي يتم فيها عملية بناء النظام لذلك فان في هذه الوحدة سيتم شرح ومناقشة خيارات التصميم مع وضع الأسباب لاختيار هذه الخيارات وعلاقتها مع النظام ، وسوف يتم

وضع الجداول المقترحة لهذا النظام ,كذلك سيتم عرض واجهات التطبيق المفترضة لكل من الزبون ومسئول الشركة والموظفين التي سوف يتم تطبيقها على هذا النظام.

.

و كذلك سيتم شرح الخوارزميات الأساسية والتي من المتوقع العمل وفقها ومن ثم شرح جميع المخططات النهائية (Flow Charts) والتي بناء عليها يتم برمجة الأنظمة والالتزام بها إضافة الى إيضاح الخطوات التفصيلية كعملية تصميم قواعد البيانات .

- الوحدة الخامسة : في هذه الوحدة سيتم العمل على بناء أجزاء المشروع و اختبارها للوصول الى المنتج النهائي .

- الوحدة السادسة : في هذه الوحدة سوف يتم وضع خلاصة العمل و النتائج و الاستنتاجات بالإضافة إلى مجموعة من التوصيات من اجل الاستفادة من هذا المشروع مستقبلاً او التعديل عليه .

الفصل الثاني

2.1المواضيع النظرية للمشروع (Theoretical subjects related to the main ideas of the project)

2.1.1 قواعد البيانات:

قاعدة البيانات هي الموقع الذي يتم فيه التخزين المنظم للمعلومات ، والمقصود بالتخزين المنظم هو تفريغ المعلومات بناء على ترتيب معين يعتمده القائم على قاعدة البيانات ، أما النظام أو الطريقة المتبعة للتسجيل والصيانة والتحديث لهذه المعلومات وتوفيرها كلما ظهرت الحاجة لها فهو ما يعرف بنظام القواعد البيانية .

2.1.2 UML:

لغة النمذجة الموحّدة Unified Modeling Language ، أو UML ، هي لغة نمذجة رسومية تقدم لنا صيغة لوصف العناصر الرئيسية للنظم البرمجية. (هذه العناصر تسمّى artifacts مشغولات في UML). بطبيعتها تتّجه UML نحو بناء البرمجيات كائنية المنحى object oriented.

2.1.3 المجموعات الموجودة في الحسابات المالية:

1- حسابات البيع: حيث يتم جمع العمليات التي تتم على أي عملية للبيع للزبون.
2- حسابات الشراء: حيث يتم جمع العمليات التي تتم على أي عملية للشراء للزبون .

3 -البنك :هي عبارة عن الأموال المملوكة للمنشاة وتحتفظ بها في شكل حساب جاري لدى البنك.

2.2 Information about special components:

2.2.1 Microsoft Office 2003:

يعتبر برنامج Office 2003 ، وهو الإصدار التالي لـ Microsoft Office XP حيث يجعل Office 2003 من السهولة بمكان أن تتخذ الأشخاص قرارات أكثر فعالية والحصول على أفضل النتائج. وتشمل التحسينات الأخرى ما يلي :

- التكامل مع منتجات وتقنيات Microsoft Share Point™ Products and Technologies "v2.0" مما يساعد فرق العمل على التعاون معاً بسرعة وببراعة، ويحسن قدراتك على المساهمة في بيئات العمل الملحة والهامة .

- توفير اعتماد لغة XML في Microsoft Office Word 2003 ، و Microsoft Office Excel 2003، و Microsoft Office Access 2003 لأعمال المجموعات البنائية التي يمكن

استخدامها في إنشاء حلول من شأنها تحسين قدرة الفرد على التوقع والإدارة والاستجابة لتغيرات السوق والتغيرات داخل المؤسسات .

- تطبيق مفهوم اعتماد "إدارة حقوق المعلومات" عن طريق تمكين الأعمال التجارية من حماية المعلومات الخاصة بها ضد سوء الاستعمال واستغلال الملكية بشكل غير مشروع .

2.2.2 Microsoft Visual Studio .NET 2003 :

تعتبر حزمة التطوير Microsoft Visual Studio .NET 2003 هى أداة التطوير الشاملة لبناء ونشر التطبيقات بسرعة للعمل تحت كلاً من نظام Windows، والويب، والأجهزة النقالة.
مع Microsoft Visual Studio .NET 2003، فقد تم إضفاء مزيد من السهولة على مهام البرمجة التى كانت قبلاً معقدة، وتستهلك وقتًا، ومحبطة.

تتيح Visual Studio .NET 2003 للمطورين على نصب تحدياتهم البرمجية الضاغطة باستخدام المهارات وأصول الكود التى قاموا بالفعل بتطويرها. وعند استخدامها مع نظام ملقم Windows Server 2003، تقدم Visual Studio .NET 2003 حلول برمجية طرفية من أجل تطوير، ونشر، تطبيقات متصلة، ويُعتمد عليها.

2.2.3 Microsoft SQL Server 2000:

يمكننا الخادم SQL Server من تغيير بعض إعدادات بدء التشغيل عن طريق وسيطات بدء تشغيل الخادم Server Startup Parameters. تفيدنا كثيرا هذه الوسيطات عند ظهور المشاكل او أوقات إصلاح وصيانة قواعد البيانات في جهاز الخادم، كما تمكنك من التحكم أكثر في طريقة وأسلوب نقل البيانات بين العملاء Clients والخادم نفسه Server، والاهم من ذلك تحديد مواقع ملفات قواعد البيانات كقاعدة البيانات الرئيسية **master** وملفات log.

2.3 متطلبات النظام (System requirements)

2.3.1تعريف النظام (System Definition)

يقوم هذا النظام بتنفيذ تطبيق وإداري وحسابي بسيط من خلال مستعرض لشبكة web اذ يمكننا من الوصول إلى المعلومات الإدارية المالية للشركة من إي مكان حيث سيقوم النظام بتسجيل جميع الحركات التي تحدث داخل الشركة من حيث تسجيل جميع فواتير الشراء او البيع وتسجيل طلبيات الزبائن.

إضافة إلى ذلك فإن النظام سيعمل على تزويد الزبائن بخدمات تعامل مع قسمي المبيعات والمشتريات من خلال شبكة الانترنت.

2.3.2 متطلبات النظام الوظيفية (Functional Requirements)

المهمة الأساسية للمشروع هي:

-إدارة المبيعات والمشتريات وذلك عن طريق التعامل مع المتعاملين والمستوردين والموردين حيث انه سيكون هناك طلبيات صادرة وأخرى واردة.

-سيكون هناك موظفين للقيام بعملية تسجيل الطلبيات الصادرة والواردة وذلك بعمل فاتورة خاصة بهم.

_سيكون هناك جزء للإدارة المالية من حسابات الشراء وحسابات البيع والحسابات البنكية.

2.3.3 متطلبات النظام غير الوظيفية (Non-Functional Requirements)

المتطلبات غير الوظيفية يمكن التعبير عنها بسهولة و ذلك لأنها ترتبط بأشياء قياسية ومن هذه المتطلبات التي سوف نراعيها في نظامنا مايلي:

- الاستخدامية(Usability).
- الاعتمادية(Reliability).
- السرية(Security).
- الأداء(Performance).
- بيئة النظام(System environment).

كما ويوجد فيه متطلبات أخرى مثل:

- السرعة(Speed).
- التكلفة(Cost).
- وقت التسليم(Time to Market):هذا المشروع سيسلم قبل 2006/6/6.

2.3.3.1متطلبات المشروع(Product Requirements).

- الاعتمادية: يتوقع من إعطاء الشيء المتوقع منه و عدم حدوث تغيير أو أخطاء في أداء النظام.

- الاستخدامية: أن الاستخدامية في هذا النظام تظهر من خلال:

- إن بيئة العمل للمستخدم هي windows واستخدام Internet Explorer حيث يمكن استخدام لوحة المفاتيح Keyboard أو الفأرة mouse لأغراض تشغيل و استخدام النظام.

- يتم تشغيل النظام من خلال Internet Explorer يمكن الانتقال بين مكوناتها باستخدام الفأرة أو لوحة المفاتيح, حيث أن استخدام النظام لا يحتاج إلى خبرة كبيرة في مجال الحاسوب.

- كما أن نماذج الإدخال سهلة و واضحة يسهل فهمها و التعامل معها بقدر قليل من التدريب للمستخدم حيث لا يوجد أي غموض في شاشات الإدخال.

- السرعة: إن عملية تخزين البيانات و استرجاع وإظهار النتائج المطلوبة من العمليات التي يتم إجراءها على هذا النظام هي سريعة جداً مقارنة مع الطريقة التقليدية و لكن قد يحتاج لبعض الوقت في عملية إدخال البيانات اللازمة لبعض المعاملات .

- السرية: وتتحقق السرية باستخدام عملية تشفير كلمة المرور لكل من المستخدمين أو القائمين على هذا النظام بحيث لايمكن استنتاجها هذه الكلمة من قاعدة البيانات كما أن السرية تتحقق من خلال الفصل الكامل بين الصلاحيات الممنوحة للزبائن عنها تلك الممنوحة لمسئول الشركة.

وتكمن السرية في :

1- النظام لا يقبل كلمة مرور اقل من 7 حروف.

2- وجد فصل كامل للصلاحيات الممنوحة لكل موظف من موظفي الأقسام كما يوجد فصل كامل بين واجهات المدير وواجهات العامل وواجهات المتعامل.

3- استخدام أسماء أخرى للأعمدة الموجودة في قواعد البيانات مع تعمد الخطأ بشكل متعمد من الناحية الإملائية لبعض العناصر الداخلية وتسمية تعكس المعنى الحقيقي في مواقع أخرى مما يجعل المحاول الاستفادة من هذه الجداول لديه الكثير من الاحتمالات .

4- استخدام خاصية الجلسات (Session) المدعومة وبقوة في لغة (ASP.NET) في جميع الملفات بحيث انه لا يمكن الوصول إلى أي ملف إلا من خلال صفحة الدخول وليس من خلال كتابة الارتباط في متصفح الانترنت.

5- وجود تسجيل الخروج في كل صفحة بحيث يتجه الى صفحة الدخول وتنتهي كل الإعدادات الخاصة به وإذا أراد الدخول مرة أخرى فيجب عليه وضع اسم المستخدم وكلمة المرور مرة اخرى.

6- استخدام اقتران لعمل تشفير لكلمة المرور قبل عملية التخزين داخل قاعدة البيانات .

7- استخدام مزايا الأمان القوية التي يوفرها SQL Server 2000 في جميع خطوات التعامل مع قواعد البيانات.

8- عمل تسجيل خروج تلقائي عند ترك النظام لمدة 05 دقيقة.

- الأداء: يعتبر الأداء من أهم القضايا التي يجب أخذها بعين الاعتبار في الأنظمة الإدارية, و نتوقع أن يكون النظام ذو أداء جيد و فاعلية بحيث يقوم بتحقيق الهدف المرجو منه بسرعة و كفاية عالية و بدقة حيث سوف يعمل النظام بشكل متكامل يؤثر كل منه في الأجزاء الأخرى.

- بيئة النظام.

- سيطبق النظام لإدارة نظام إداري ومحاسبي شخصي مع تقديم خدمات على شبكة الانترنت .

- مستخدمو البرنامج هم أصحاب الشركات والزبائن .

- لن يحتاج مستخدمو البرنامج إلى خبرة عالية في مجال الحاسوب و لكن يجب أن يكون عنده معلومات أساسية للتعامل مع نظام التشغيل ويندوز واستخدام الانترنت و قليل من التدريب على كيفية استخدام هذا النظام .

2.3.3.2 الافتراضات و الاعتماديات (Assumption and Dependencies)

هناك العديد من الافتراضات للمشروع نذكر منها لا للحـــصر:

- عندما اي مستخدم لا يقوم بادخال كلمة المرور بشكل صحيح فان النظام سيمنعه من الدخول الى مكونات النظام.

- عندما يكون النظام غير متصلا بالانترنت فان هذا النظام لايعمل.

الفصل الثالث

التصميم البنائي للنظام

(Architectural Design)

3.1 أهداف النظام (System Objectives)

ان الهدف من النظام هو خدمة اصحاب الشركات حيث يسهل هذا النظام من اجراء العمليات الادارية والحسابية من خلال إدخال البيانات اللازمة ومن ثم اجراء التعديلات اللازمه عليها لكون ان هذه العمليات تخضع الى كثير من الوقت حيث يسهل هذا النظام اجراء هذه العمليات بشكل اسهل واسرع حيث يكون النظام قادر على:

1-تلخيص البيانات المالية وعرضها في صورة تقارير

2-عرض لحركة النقود من خلال البيانات المالية الموجودة.

3-تخزين معلومات حول المعاملات المالية وهي بنود تحويلات الدفع والاستلام من اجل كل ايداع او سحب.

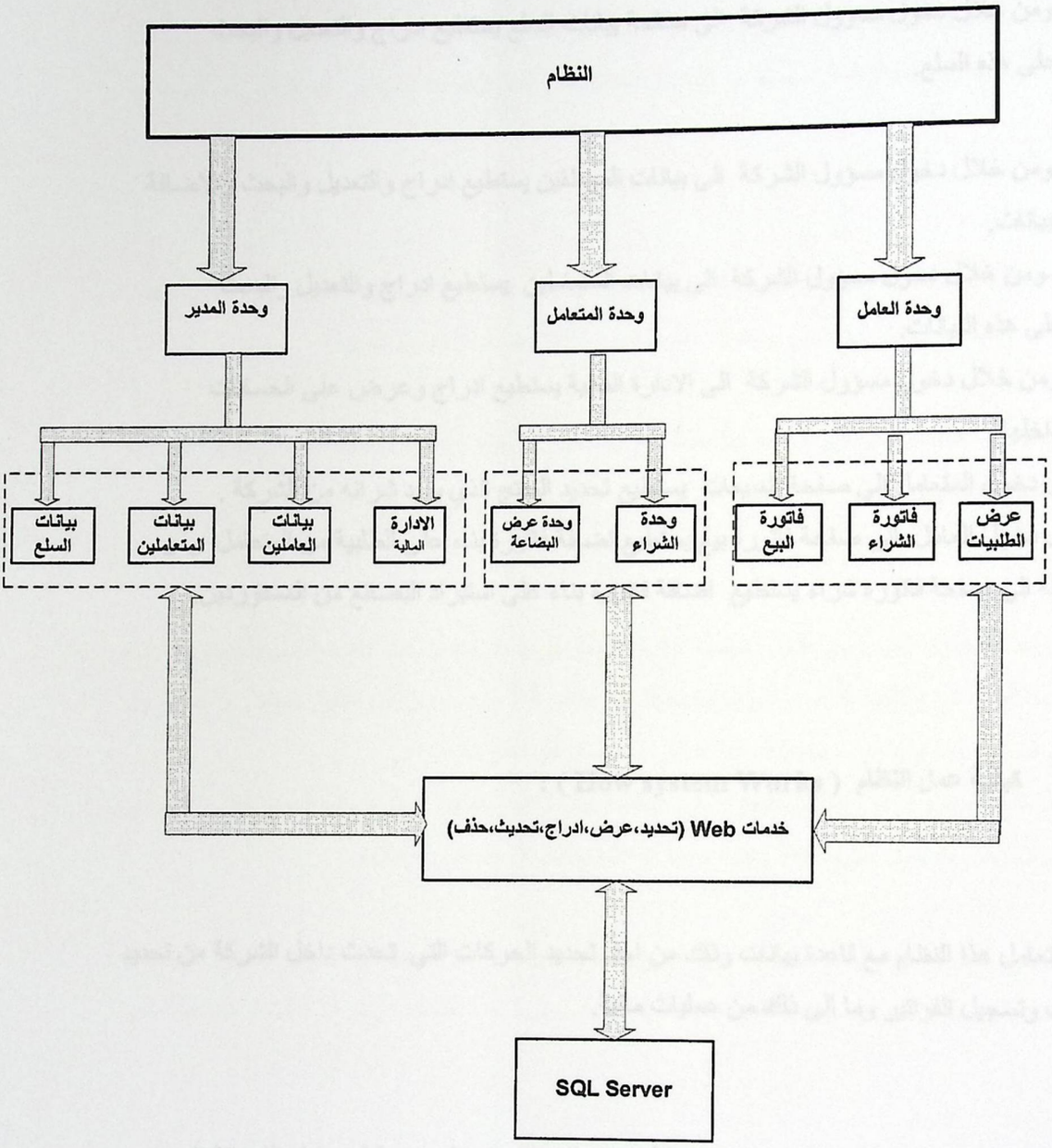
4-العمل على إيجاد الوضع المالي والوقوف على حقيقة ذلك الوضع.

5- عرض تقرير المبيعات والمشتريات لزبون معين.

6-عرض قائمة بأسماء منتجات الشركة مع المواصفات التي يحتاج اليها المشتري.

3.2 الشكل العام للنظام (General Block Diagram) :

يتكون هذا النظام من ثلاثة وحدات اساسية أولا هما وحدة المدير الشخصي والوحدة الثانية وحدة الزبائن والثالثة وحدة العامل وكل وحدة من هذه الوحدات تتكون من الاقسام الموضحة في الشكل:



الشكل 3.1 الشكل العام للنظام (General Block Diagram)

-من خلال هذا النظام تكون جميع المعاملات الادارية و المالية مخزنة على web في قاعدة بيانات ويستطيع مسؤول الشركة الذي يملك الدخول الى هذا الموقع من التعديل على هذه المعلومات اي يقوم بعملية الحذف والاضافة والتعديل على هذه المعلومات.

-ومن خلال دخول مسؤول الشركة الى صفحة بيانات السلع يستطيع ادراج والتعديل والبحث والاضافة على هذه السلع.

-ومن خلال دخول مسؤول الشركة الى بيانات الموظفين يستطيع ادراج والتعديل والبحث والاضافة على هذه البيانات.

_-ومن خلال دخول مسؤول الشركة الى بيانات المتعاملين يستطيع ادراج والتعديل والبحث والاضافةعلى هذه البيانات.

-ومن خلال دخول مسؤول الشركة الى الادارة المالية يستطيع ادراج وعرض على الحسابات الموجودة داخلها .

-ومن خلال دخول المتعامل الى صفحة المبيعات يستطيع تحديد المنتج الذي يريد شرائه من الشركة .

-ومن خلال دخول العامل الى صفحة فاتورة بيع يستطيع اضافة فاتورة بناء على الطلبية من المتعامل ومن خلال دخوله الى صفحة فاتورة شراء يستطيع اضافة فاتورة بناء على استيراد البضائع من المستوردين.

3.3 كيفية عمل النظام (How system Works) :

يتعامل هذا النظام مع قاعدة بيانات وذلك من اجل تحديد الحركات التي تحدث داخل الشركة من تحديد للطلبيات وتسجيل الفواتير وما الى ذلك من عمليات مالية.

-عند دخول المدير الى هذا النظام يجب التأكد من الصلاحيات الممنوحة لمسؤول الشركة المستخدم (الاسم وكلمة المرور) ونوعية المستخدم واذا تم التحقق من الصلاحيات فسيعتبر المسؤول مسموحا له بالعمل .

-اما بالنسبة لقسم الزبائن فإنه مزود بكلمة مرور واسم مستخدم مفصوله عن قسم مسؤول الشركة فإذا تم التحقق من اسم المستخدم وكلمة المرور ونو ع المستخدم يسمح لهذا الزبون الوصول الى الخدمات التي تقدمها الشركة.

-وعند دخول العامل الى هذا النظام يجب التأكد من الصلاحيات الممنوحة لهذا الموظف المستخدم (الاسم وكلمة المرور) ونوعية المستخدم واذا تم التحقق من الصلاحيات فسيعتبر الموظف مسموحا له بالعمل .

- يتم ادخال البيانات اللازمة عن طريق المدير ويقوم بادخال مجموعة من البيانات المطلوبة التي تتعلق ببيانات السلع وبيانات الموظفين وبيانات المتعاملين والقيام بعمليات الادارة المالية

- فالمسؤول عندما يدخل الى صفحة بيانات السلع يستطيع ان يقوم بادراج او حذف سجل معين كإضافة كإضافة منتج جديد او اجراء تعديلات على السجلات الموجودة .

- اما عند دخول المسؤول الى صفحة بيانات الموظفين حيث سيحدد طبيعة الموظف واعطائه الصلاحيات في امكانية الدخول الى النظام حيث ان هناك عدد محدود من الموظفين المسموح لهم بدخول الى صفحة النظام واجراء جميع العمليات التي تحدث داخل الشركة.
- ويستطيع المسؤول من خلال الدخول الى صفحة بيانات المتعاملين من اضافة المتعاملين واعطائهم الصلاحيات بالدخول الى صفحة النظام واجراء التعديلات الازمه على بياناتهم.

- يستطيع الزبون الدخول الى صفحة النظام من للقيام بمجموعة من العمليات ضمن قسمي المبيعات والمشتريات مثل حجز قائمة باسماء بعض المنتجات او عرض قائمة لما لديه من منتجات لإطلاعها على الشركة.

3.4 مخططات نماذج النظام (System Modeling) :

في هذا القسم سيتم استخدام (Unified Modeling Language) UML لاستعراض المخططات الأولية اللازمة لعملية بناء النظام وهي عبارة عن مجموعة من الأشكال و المخططات التوضيحية والتي تستخدم لوصف تصميم النظام البرمجي (software system) .

و في هذه الوحدة سيتم استخدام مجموعة من النماذج (Models) التوضيحية حسب نظام UML لوصف هذا النظام وهي كالتالي :

1. مخططات العمليات النشطة (Activity Diagrams) .
2. مخطط حالات الاستخدام (Use Cases) .
3. مخططات تتابع عمليات النظام (Sequence Diagrams) .
4. مخططات التنقل بين حالات النظام (State Diagrams) .

3.4.1 مخططات العمليات النشطة (Activity Diagrams):

المخطط البياني النشط (Active Diagram) يقوم على توضيح مراحل العمل و الوظائف أو الحركات (Actions) المختلفة الصادرة عن المستخدم أو عن عملية خارجية .

و هذا النظام يوضح كيف يتفاعل المستخدم مع النظام على الانترنت.

و لتوضيح هذا المخطط بالنسبة للمستخدم مع النظام يتبين كما يلي:

3.4 مخططات نماذج النظام (System Modeling) :

في هذا القسم سيتم استخدام (Unified Modeling Language) UML لاستعراض المخططات الأولية اللازمة لعملية بناء النظام وهي عبارة عن مجموعة من الأشكال و المخططات التوضيحية والتي تستخدم لوصف تصميم النظام البرمجي (software system) .

و في هذه الوحدة سيتم استخدام مجموعة من النماذج (Models) التوضيحية حسب نظام UML لوصف هذا النظام وهي كالتالي :

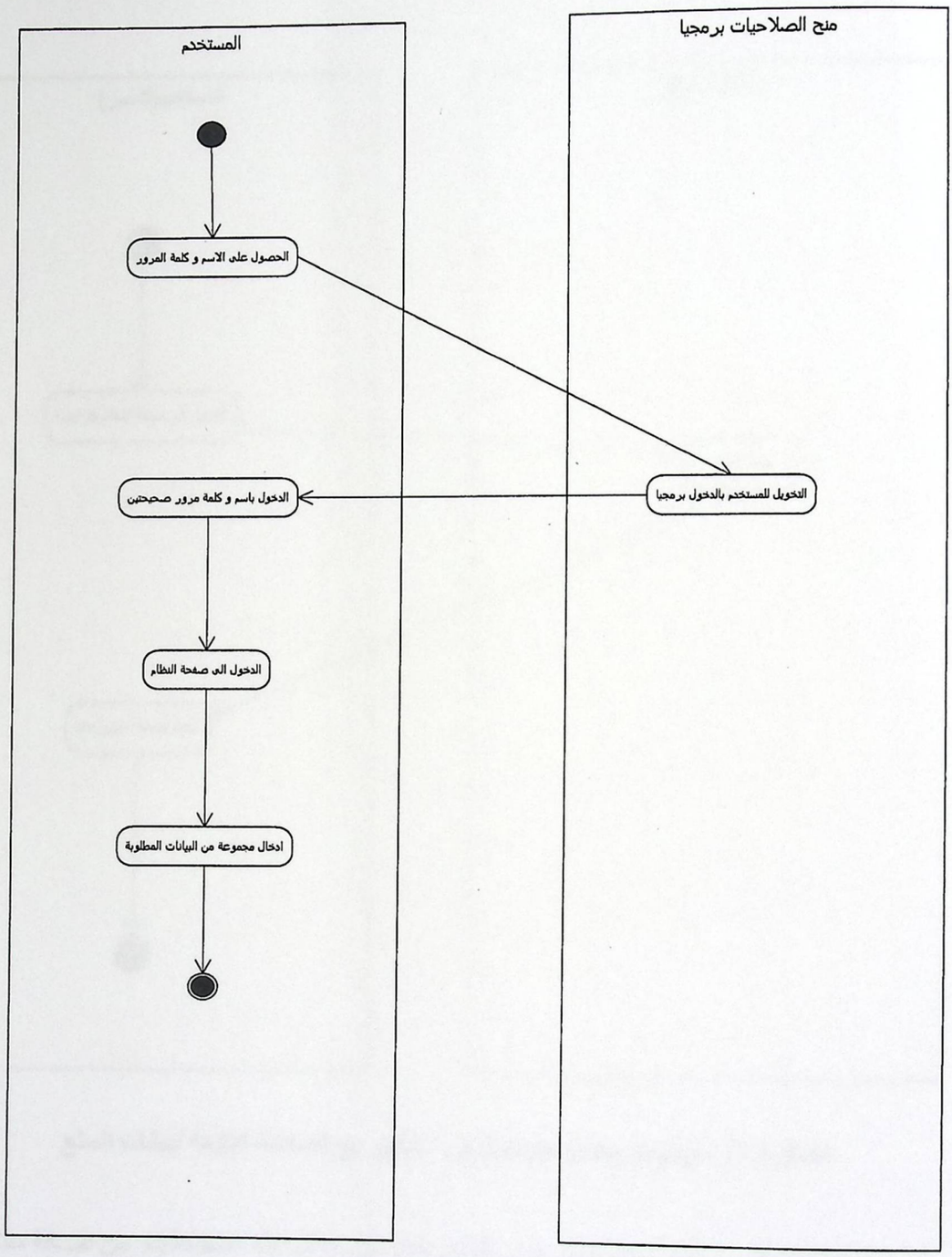
1. مخططات العمليات النشطة (Activity Diagrams) .
2. مخطط حالات الاستخدام (Use Cases) .
3. مخططات تتابع عمليات النظام (Sequence Diagrams) .
4. مخططات التنقل بين حالات النظام (State Diagrams) .

3.4.1 مخططات العمليات النشطة (Activity Diagrams):

المخطط البياني النشط (Active Diagram) يقوم على توضيح مراحل العمل و الوظائف أو الحركات (Actions) المختلفة الصادرة عن المستخدم أو عن عملية خارجية .

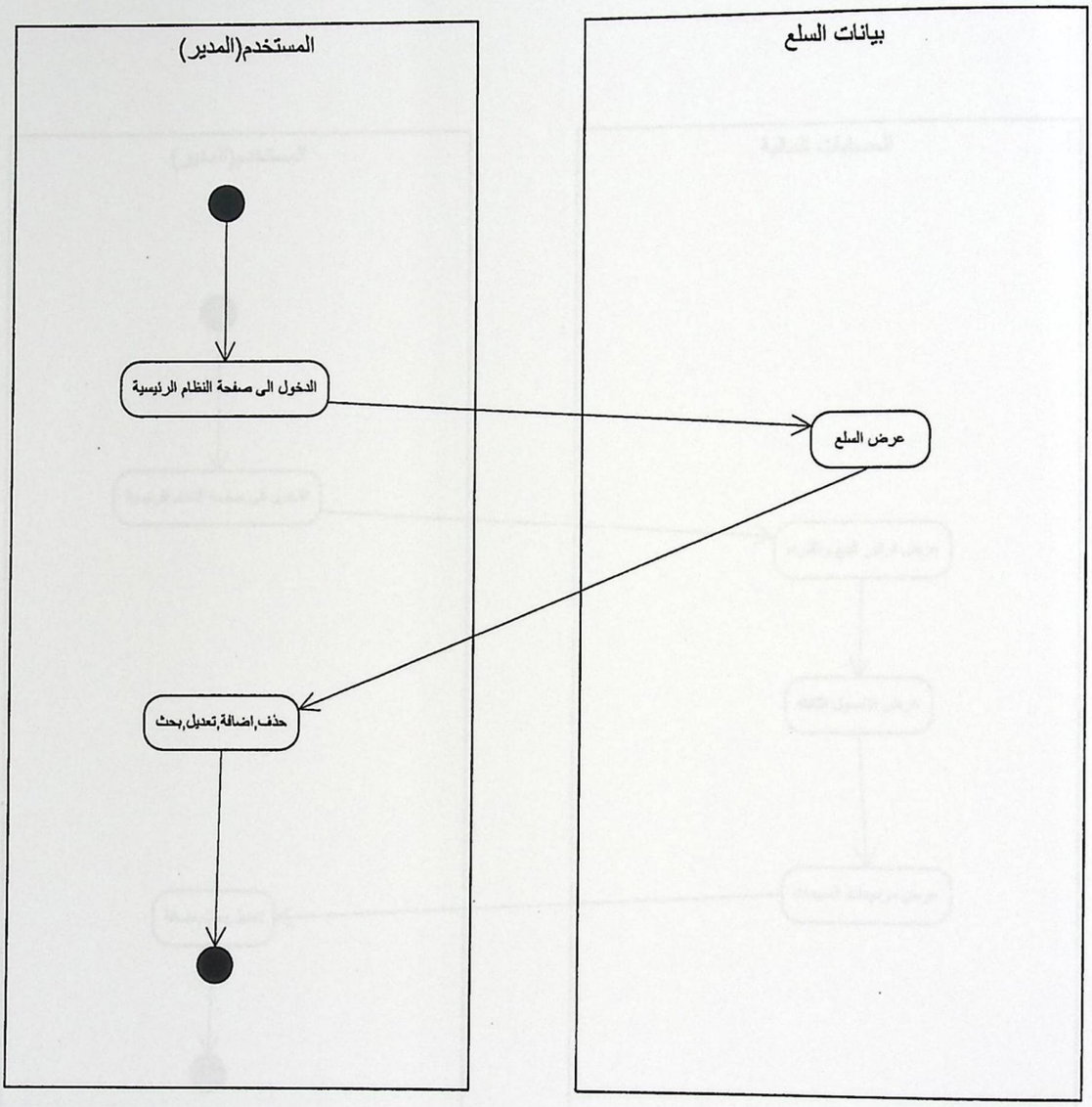
و هذا النظام يوضح كيف يتفاعل المستخدم مع النظام على الانترنت.

و لتوضيح هذا المخطط بالنسبة للمستخدم مع النظام يتبين كما يلي:



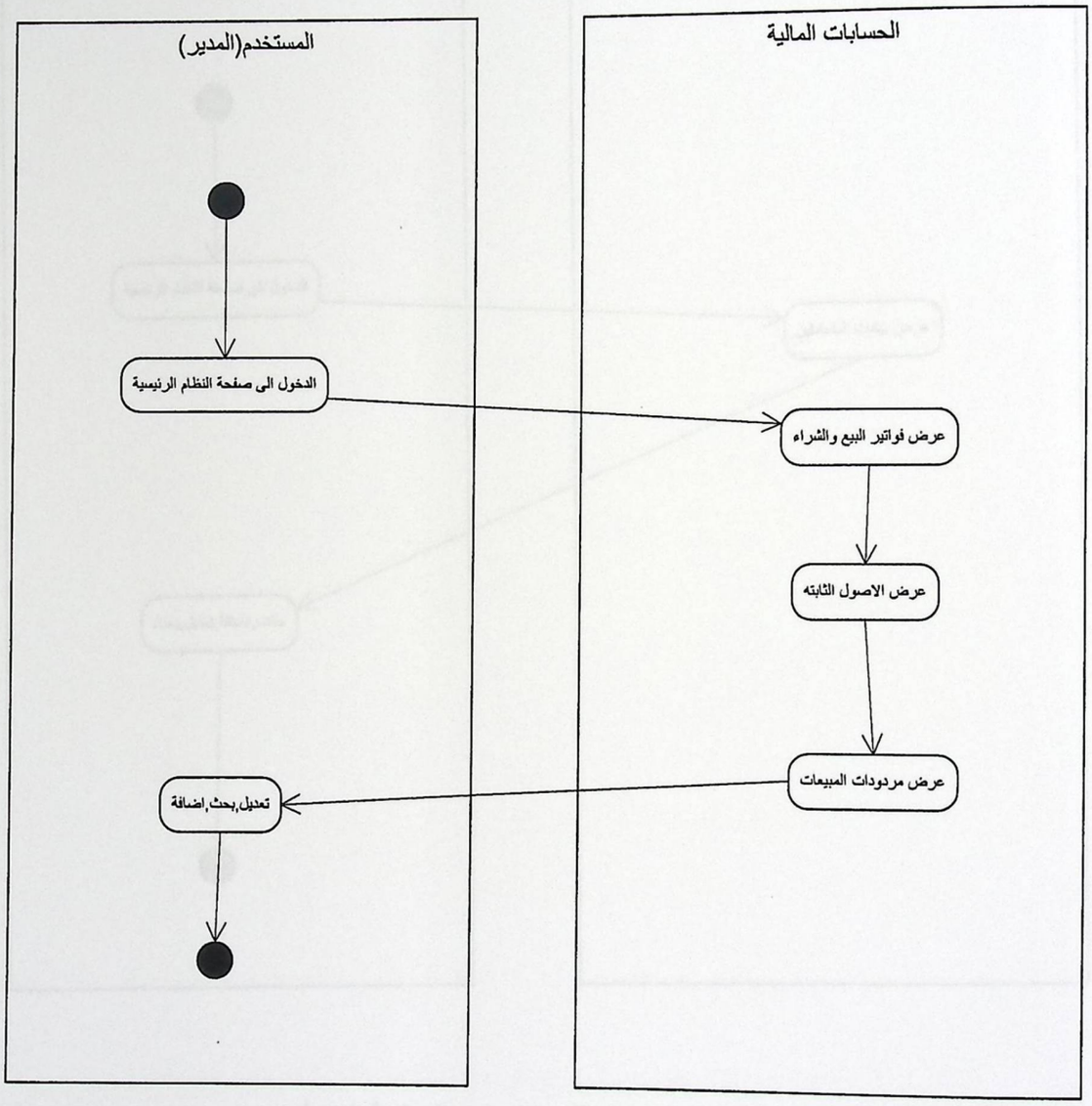
الشكل 3.2: (Activity Diagrams)المستخدم مع الصفحة الرئيسية للنظام

يبين هذا الشكل كيف يتم تسجيل الدخول الى النظام من خلال منح المستخدم الصلاحية في استخدام النظام..



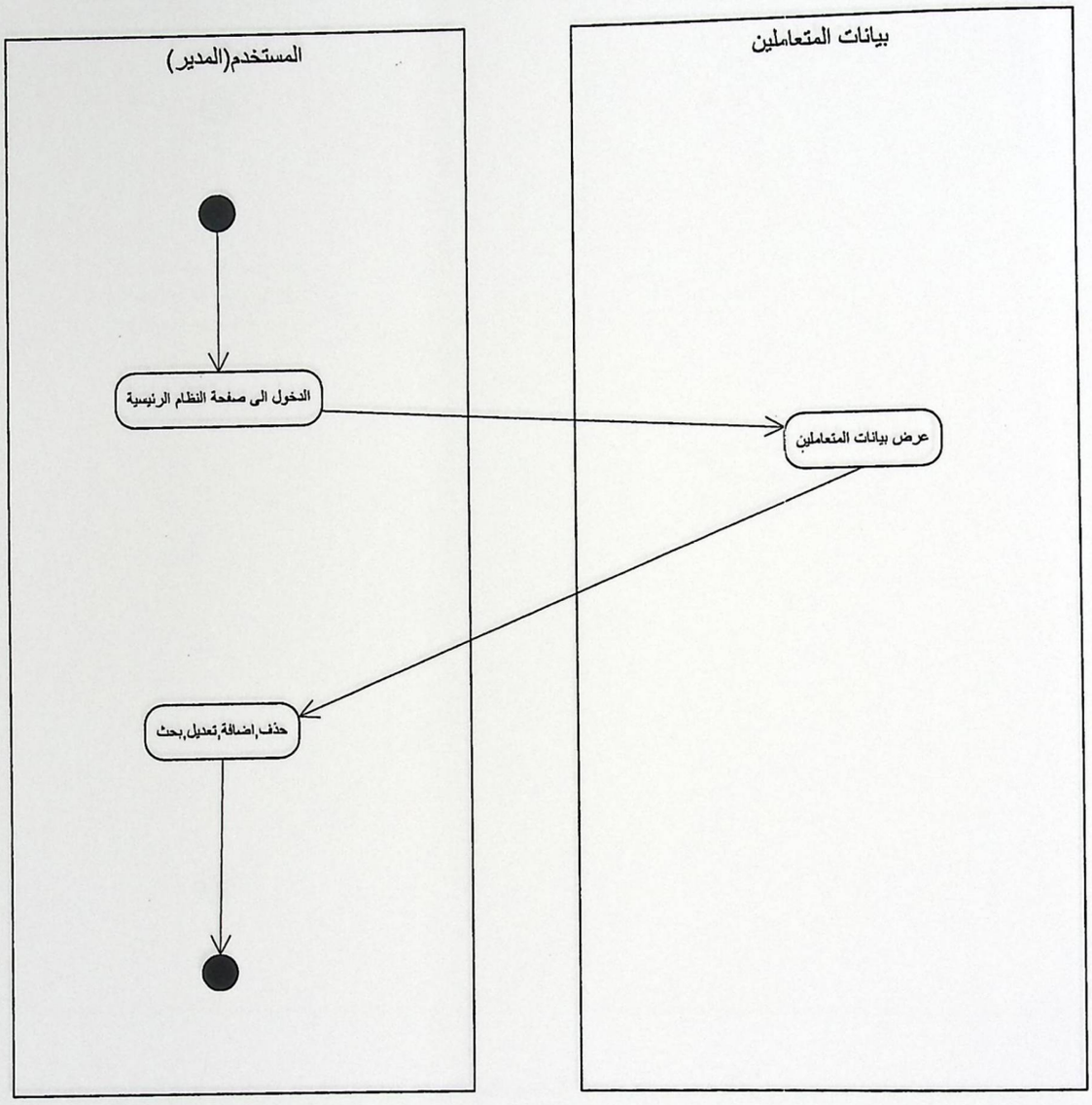
الشكل 3.3 : (Activity Diagrams) المدير مع الصفحة التابعة لبيانات السلع

يبن هذا الشكل دخول المدير الى بيانات السلع حيث من خلال هذه الصفحة يتم من اضافة منتجات جديدة د او التعديل عليها او حذف منتج.



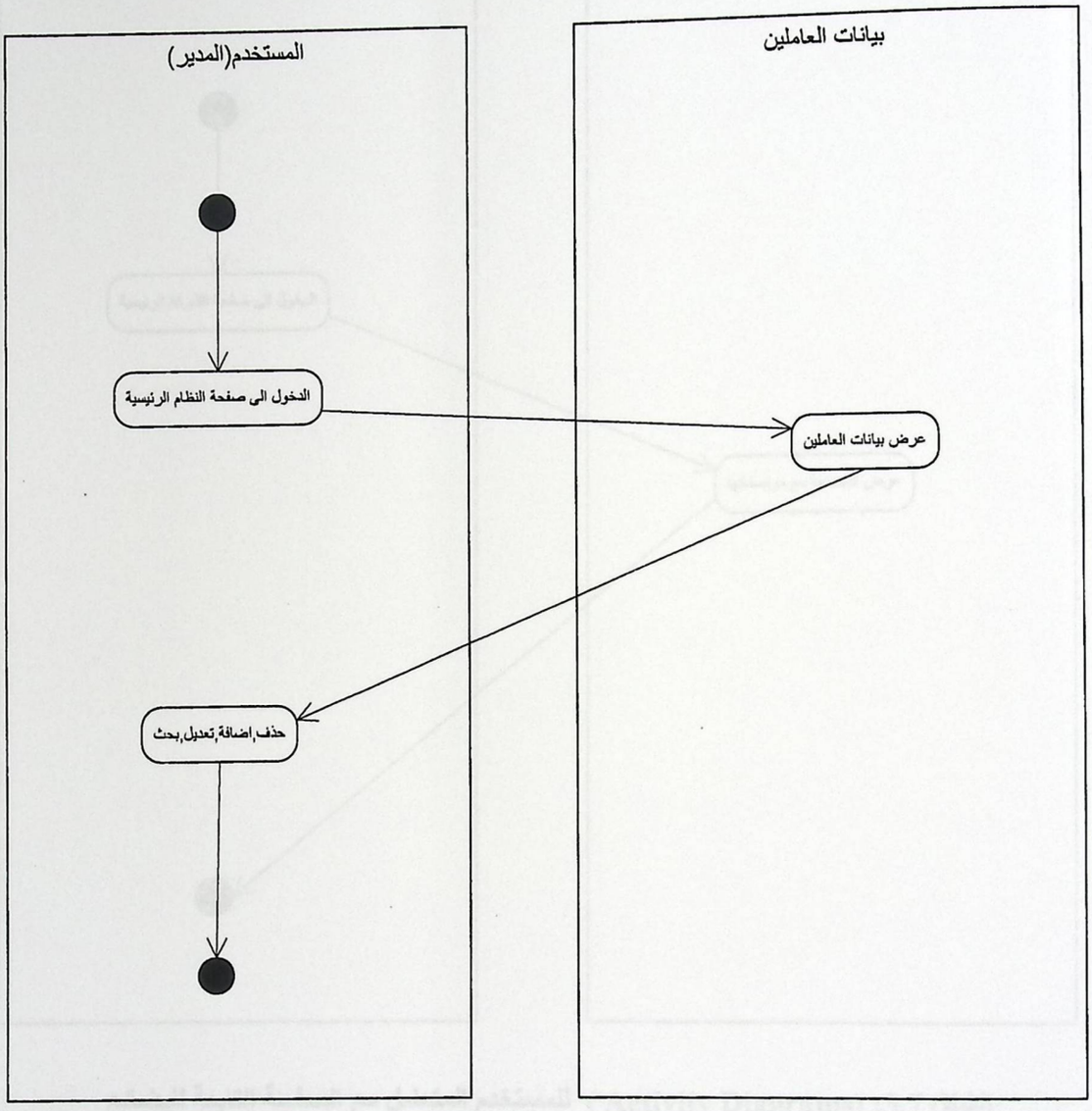
الشكل 3.4 : (Activity Diagrams) المستخدم مع الصفحة التابعة للعمليات المالية

يبن هذا الشكل دخول المستخدم الى صفحة العمليات المالية حيث من خلال هذه الصفحة يتم عرض فواتير البيع والشراء ومقدرة مسؤول الشركة على عرض الاصول الثابته وعرض لمردودات الشركة وعلى ادراج احذف او تعديل على اي من هذه العمليات .



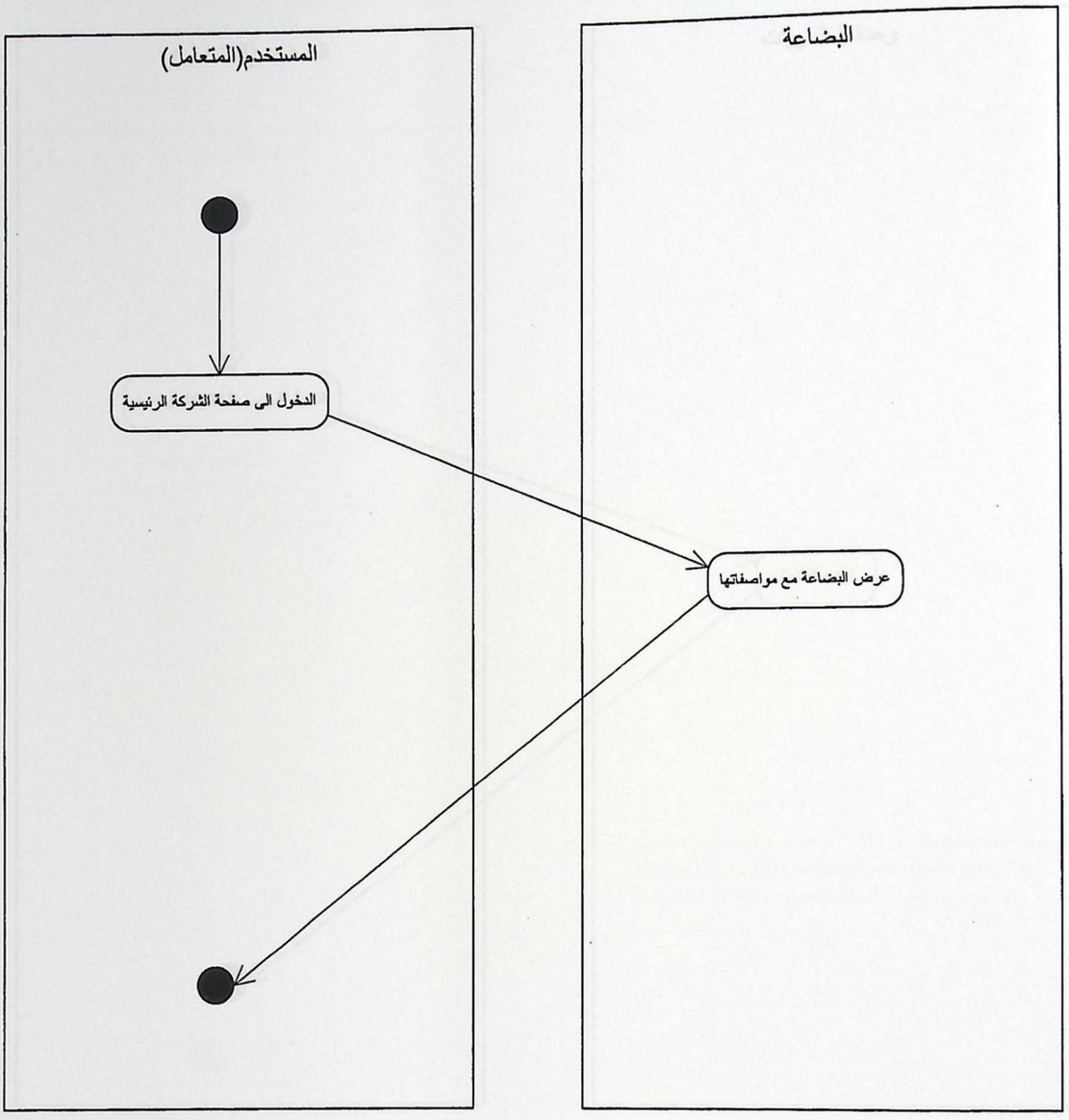
الشكل 3.5 :(Activity Diagrams) للمستخدم مع الصفحة التابعة لبيانات المتعاملين

يبن هذا الشكل دخول المستخدم الى بيانات المتعاملين حيث من خلال هذه الصفحة يتم حذف او اضافة او تعديل او بحث للمتعاملين واعطائهم الصلاحيات بالدخول الى صفحة النظام.



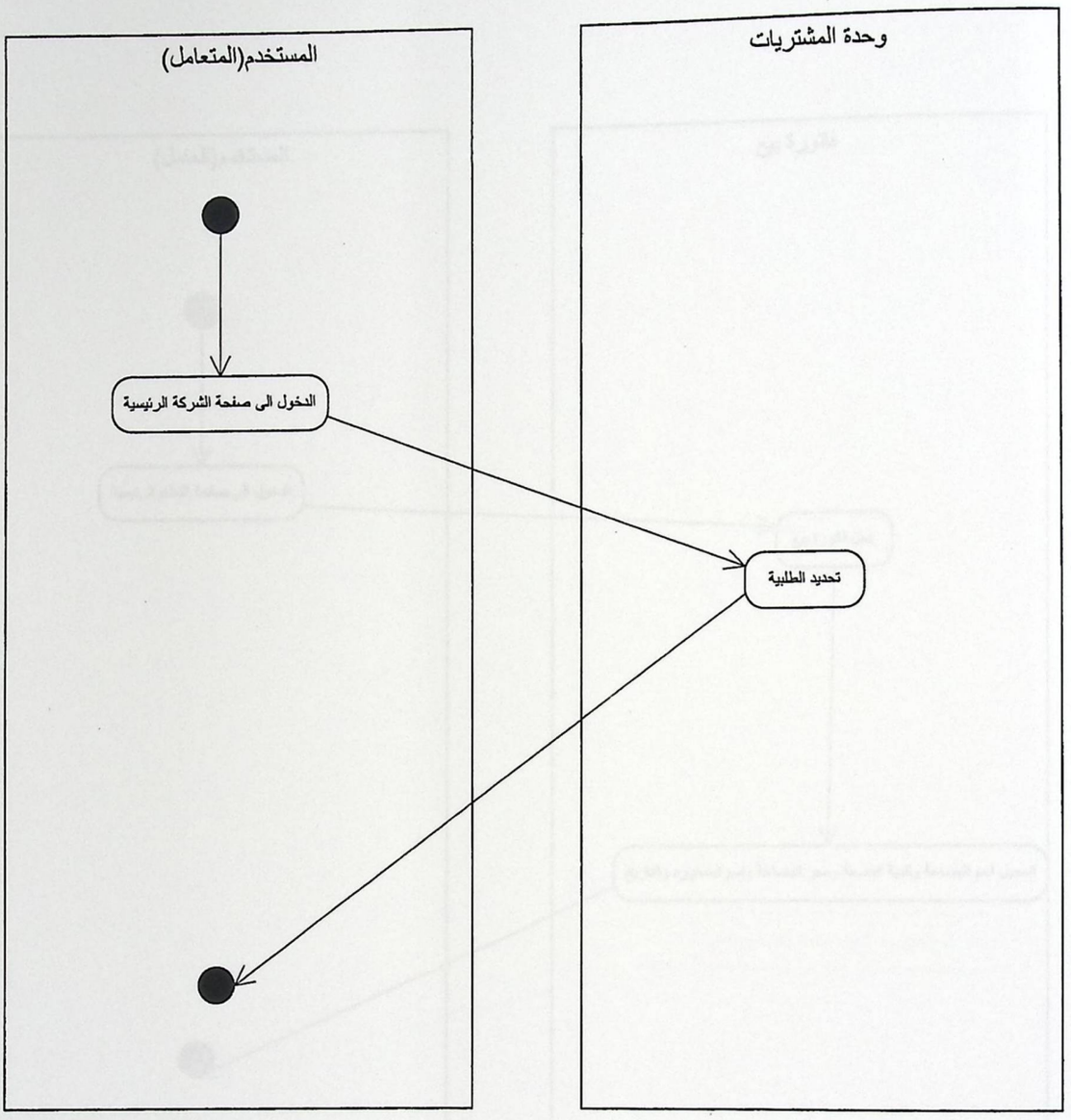
الشكل 3.6 :(Activity Diagrams) المدير مع الصفحة التابعة لبيانات المتعاملين

يبن هذا الشكل دخول المستخدم الى صفحة العاملين حيث من خلال هذه الصفحة يتم حذف او اضافة او تعديل او بحث للعاملين واعطائهم الصلاحيات بالدخول الى صفحة النظام.



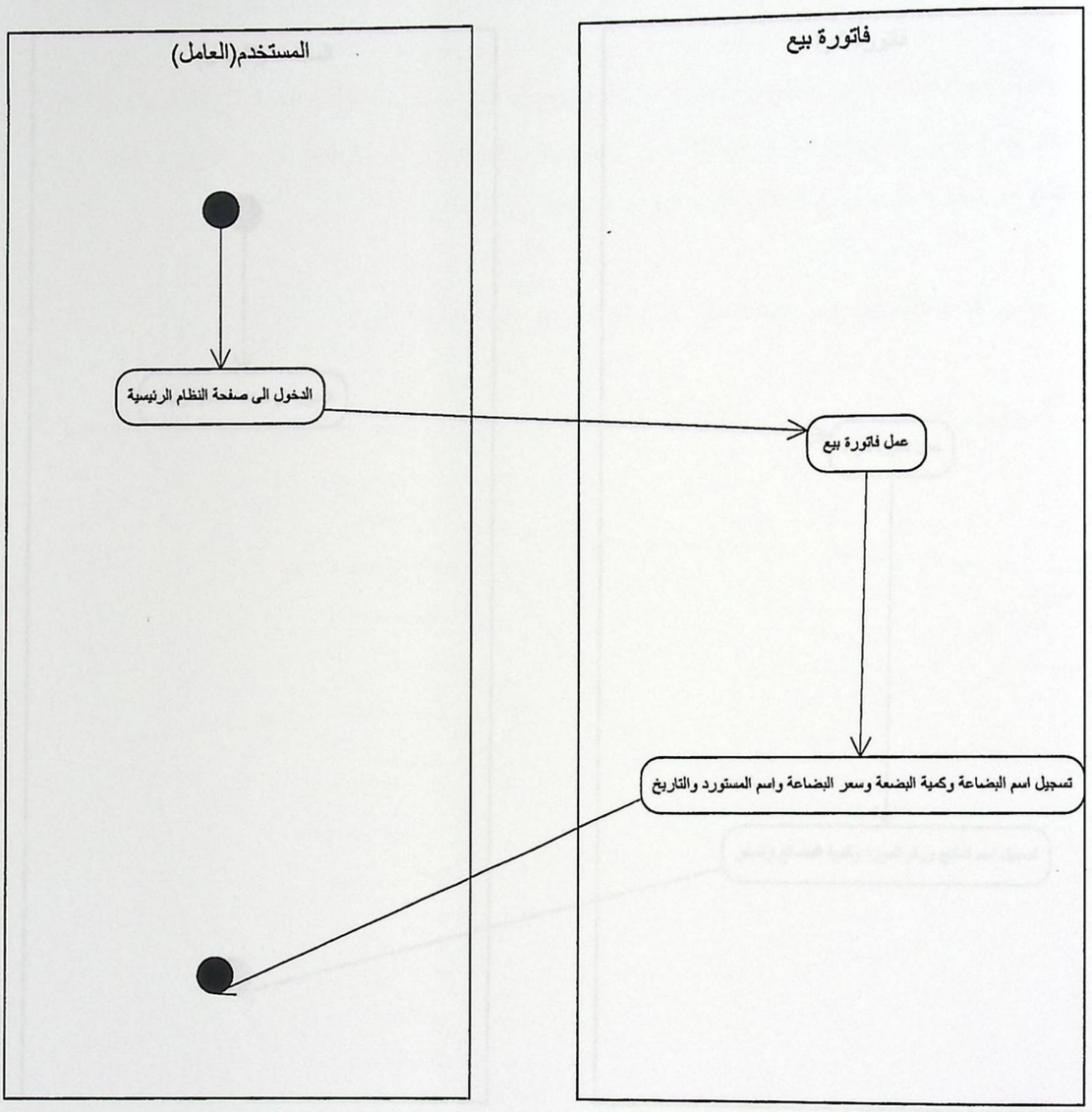
الشكل 3.7: (Activity Diagrams) للمستخدم المتعامل مع الصفحة التابعة للبضائع

يبن هذا الشكل دخول المستخدم الى صفحة المتعاملين حيث من خلال هذه الصفحة يتم عرض جميع السلع لدى الشركة مع الموصفات والاسعار.



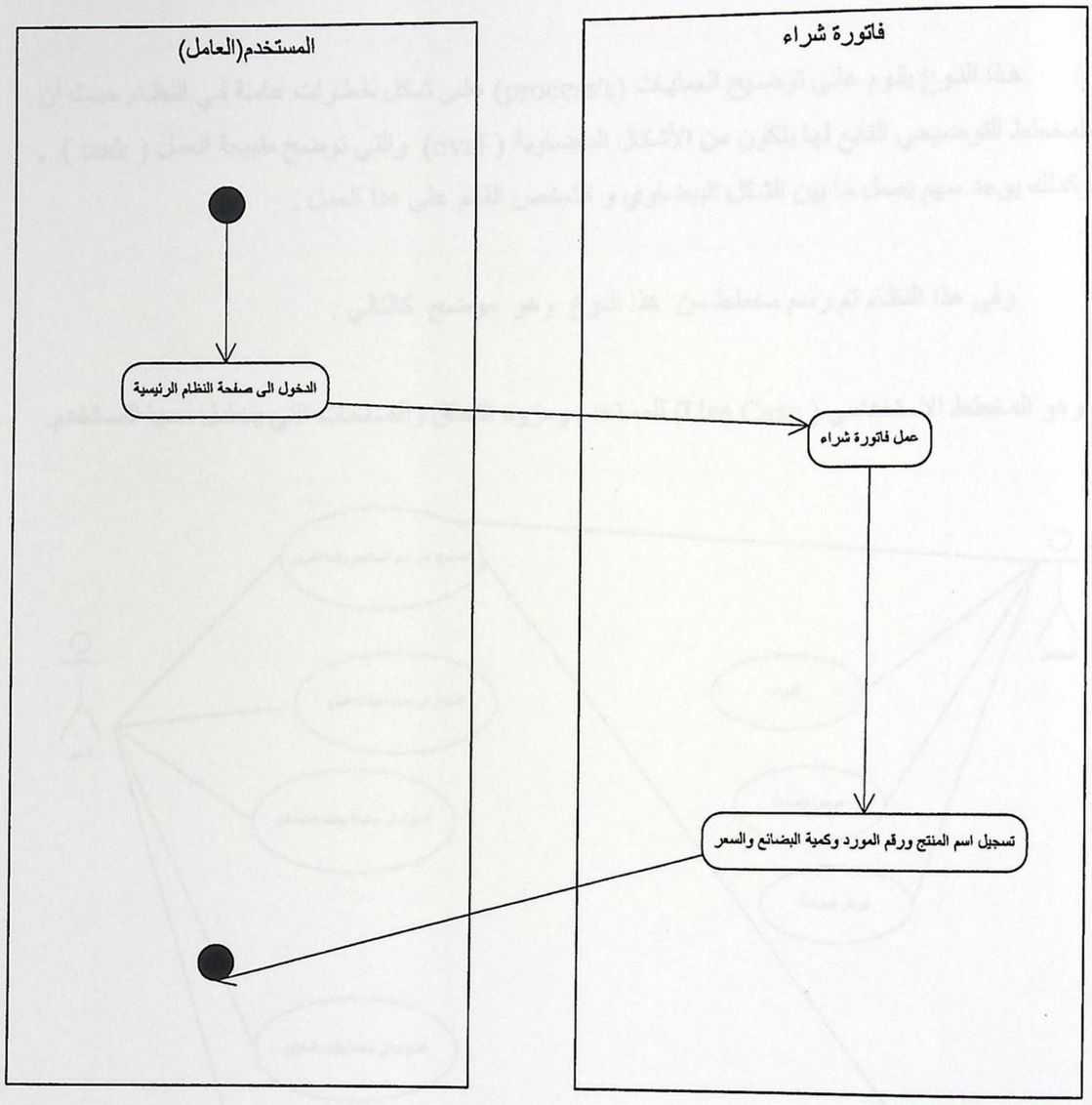
الشكل. 3.8 :(Activity Diagrams) للمستخدم المتعامل مع الصفحة التابعة لتحديد طلبية وارسالها

يبن هذا الشكل دخول المتعامل الى صفحة المشتريات حيث من خلال هذه الصفحة يقوم المتعامل بتحديد الطلبية التي يريدها.



الشكل 3.9 (Activity Diagrams) للمستخدم العامل مع الصفحة التابعة لفاتورة بيع

يبن هذا الشكل دخول العامل الى صفحة فاتورة بيع حيث من خلال هذه الصفحة يقوم العامل بتحديد الفاتورة التي يريدها.



الشكل 3.10 (Activity Diagrams) للمستخدم العامل مع الصفحة التابعة لتحديد فاتورة شراء

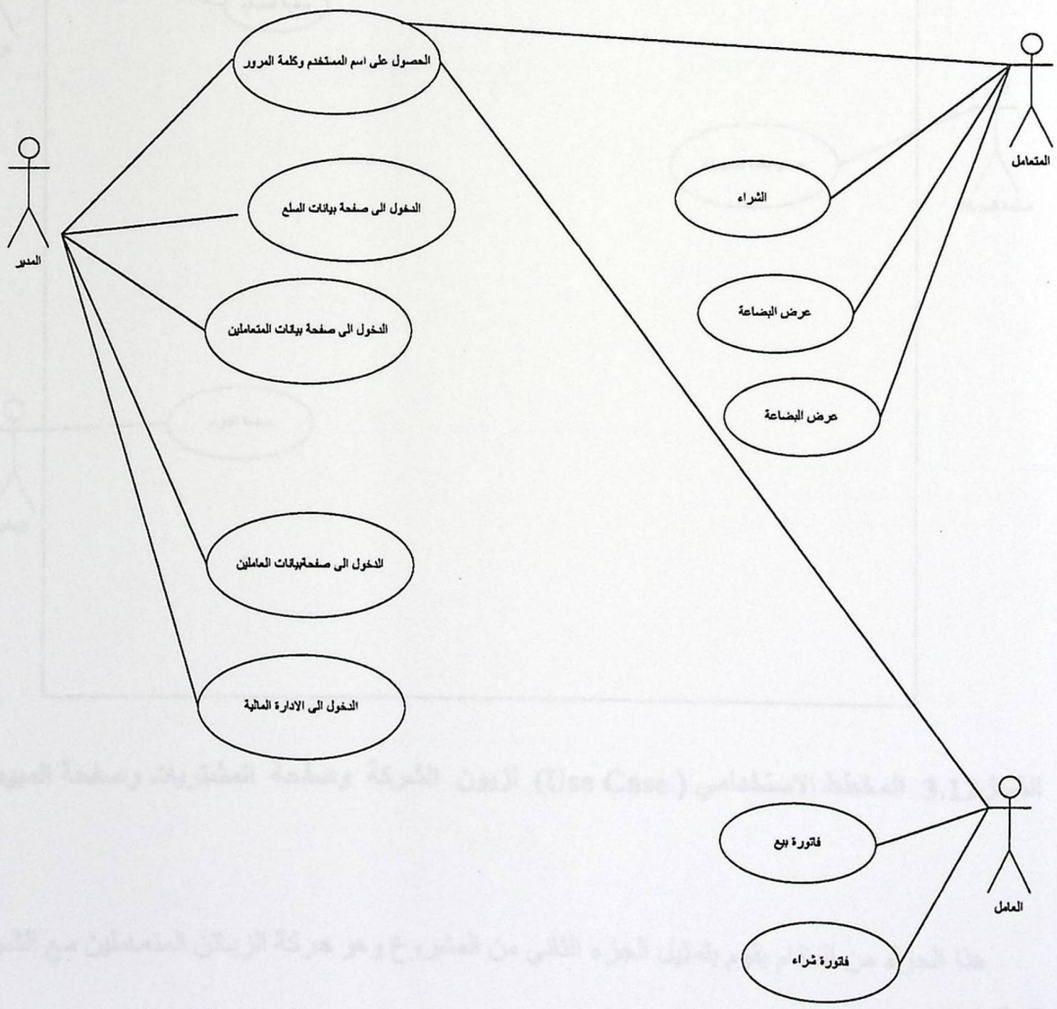
يبن هذا الشكل دخول العامل الى فاتورة شراء حيث من خلال هذه الصفحة يقوم العامل بتحديد الفاتورة التي يريدها.

3.4.2 مخطط حالات الاستخدام (Use Cases) .

هذا النوع يقوم على توضيح العمليات (process's) على شكل خطوات عامة في النظام حيث أن المخطط التوضيحي التابع لها يتكون من الأشكال البيضاوية (oval) والتي توضح طبيعة العمل (task) , وكذلك يوجد سهم يصل ما بين الشكل البيضاوي و الشخص القائم على هذا العمل .

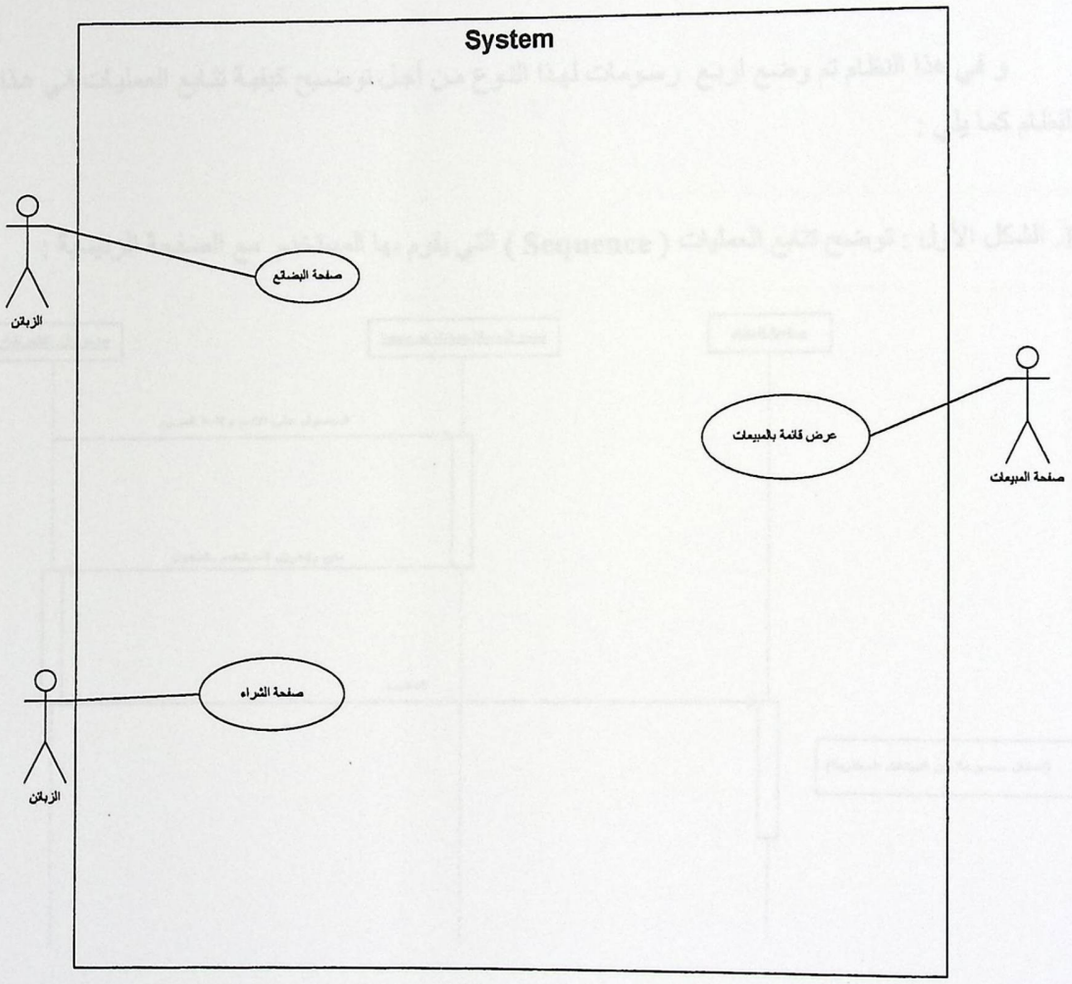
وفي هذا النظام تم رسم مخطط من هذا النوع وهو موضح كالتالي :

وهو المخطط الاستخدامي (Use Case) للمستخدم ومزود التحقق والصفحات التي يتعامل معها المستخدم.



الشكل3.11 المخطط الاستخدامي (Use Case) للمدير والعامل والمتعامل

نلاحظ من الشكل انه يمكن لهذا النظام تقديم الكثير من الخدمات وفي مقدمتها مقدرة النظام على معالجة وعرض جميع انواع العمليات التي يتم التعامل بها داخل الشركة حيث تتضمن هذه العمليات ما تم دخوله الى الشركة من منتجات وخروجها اضافة الى عرض التقارير المالية حول الوضع المالي في الوقت والمكان المناسبين عبر شبكة الانترنت



الشكل3.12 المخطط الاستخدامي (Use Case) لزبون الشركة وصفحة المشتريات وصفحة المبيعات

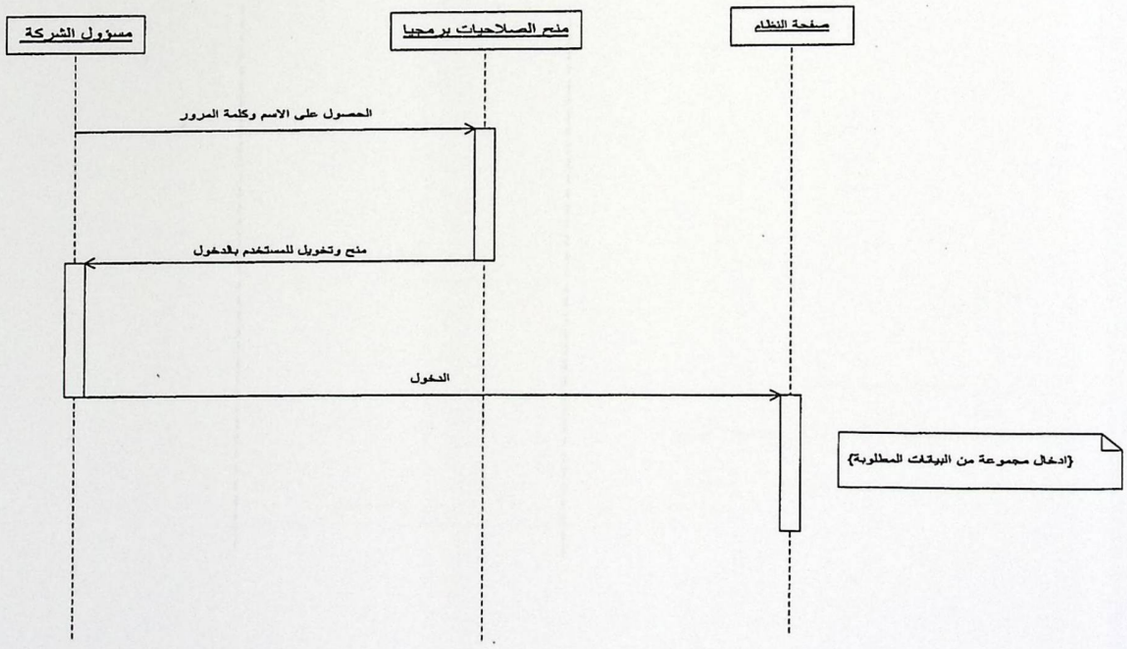
هذا الجزء من النظام يقوم بتمثيل الجزء الثاني من المشروع وهو حركة الزبائن المتعاملين مع الشركة ومنتجاتها.

3.4.3 مخططات تتابع عمليات النظام (Sequence Diagrams) .

هذا النوع من المخططات البيانية يعتبر الأفضل في توضيح العلاقة ما بين أجزاء النظام (objects) حيث يقوم على ربط كائنات النظام المختلفة (Classes) ببعضها عن طريق خطوط أفقية , وهذه الخطوط تمثل انتقال الرسائل (messages) ما بين العناصر (objects) المختلفة.

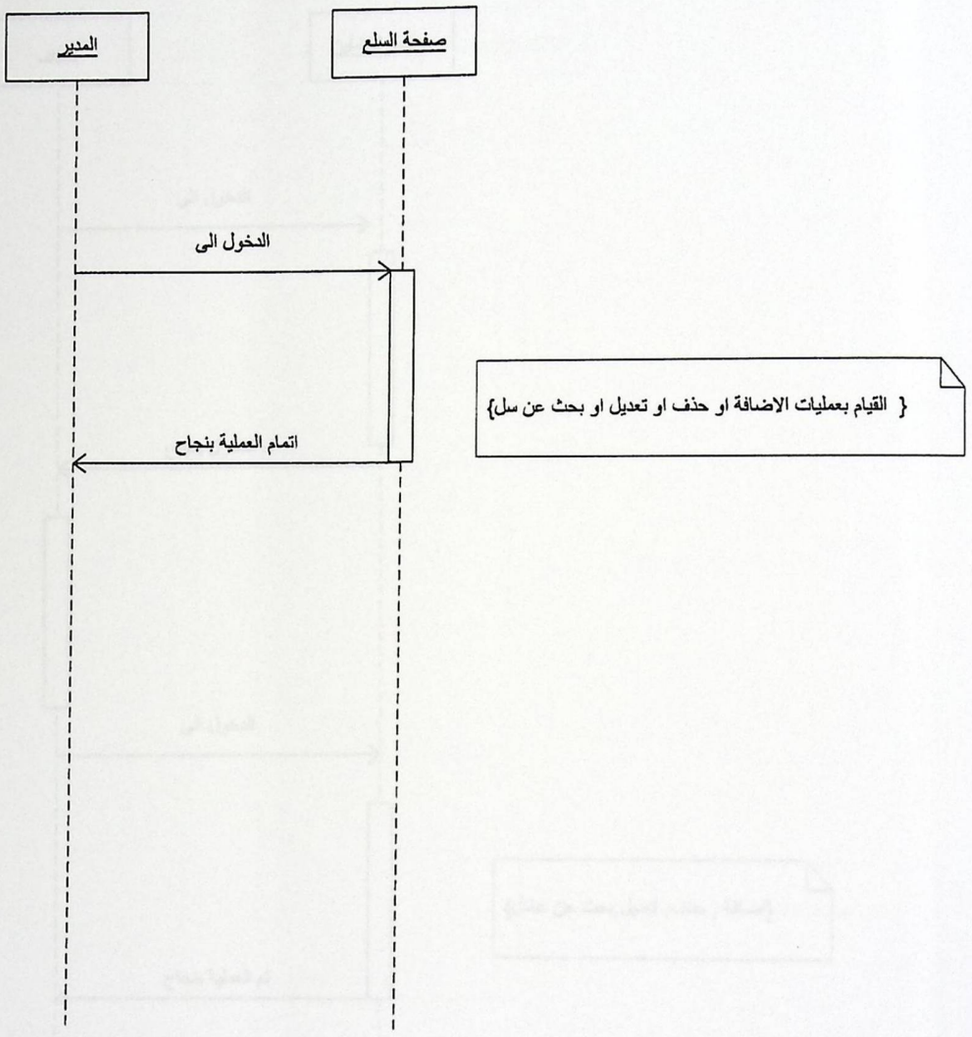
و في هذا النظام تم وضع اربع رسومات لهذا النوع من أجل توضيح كيفية تتابع العمليات في هذا النظام كما يلي :

1. الشكل الأول : توضح تتابع العمليات (Sequence) التي يقوم بها المستخدم مع الصفحة الرئيسية :



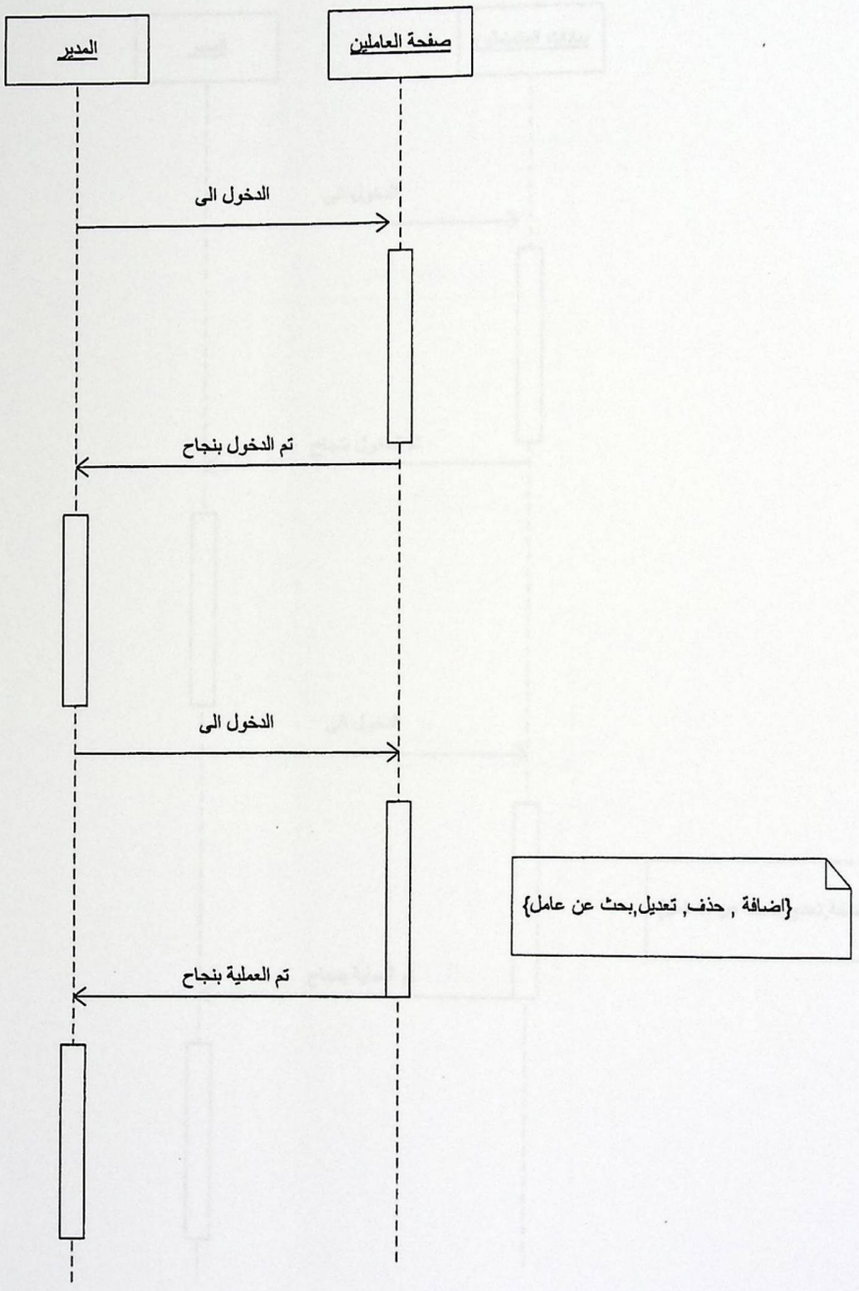
الشكل 3.13 مخطط تتابع العمليات (Sequence diagram) التي يقوم بها المستخدم مع الصفحة الرئيسية

2. الشكل الثاني : توضح تتابع العمليات التي يقوم بها المستخدم مع الصفحة التابعة لبيانات السلع :



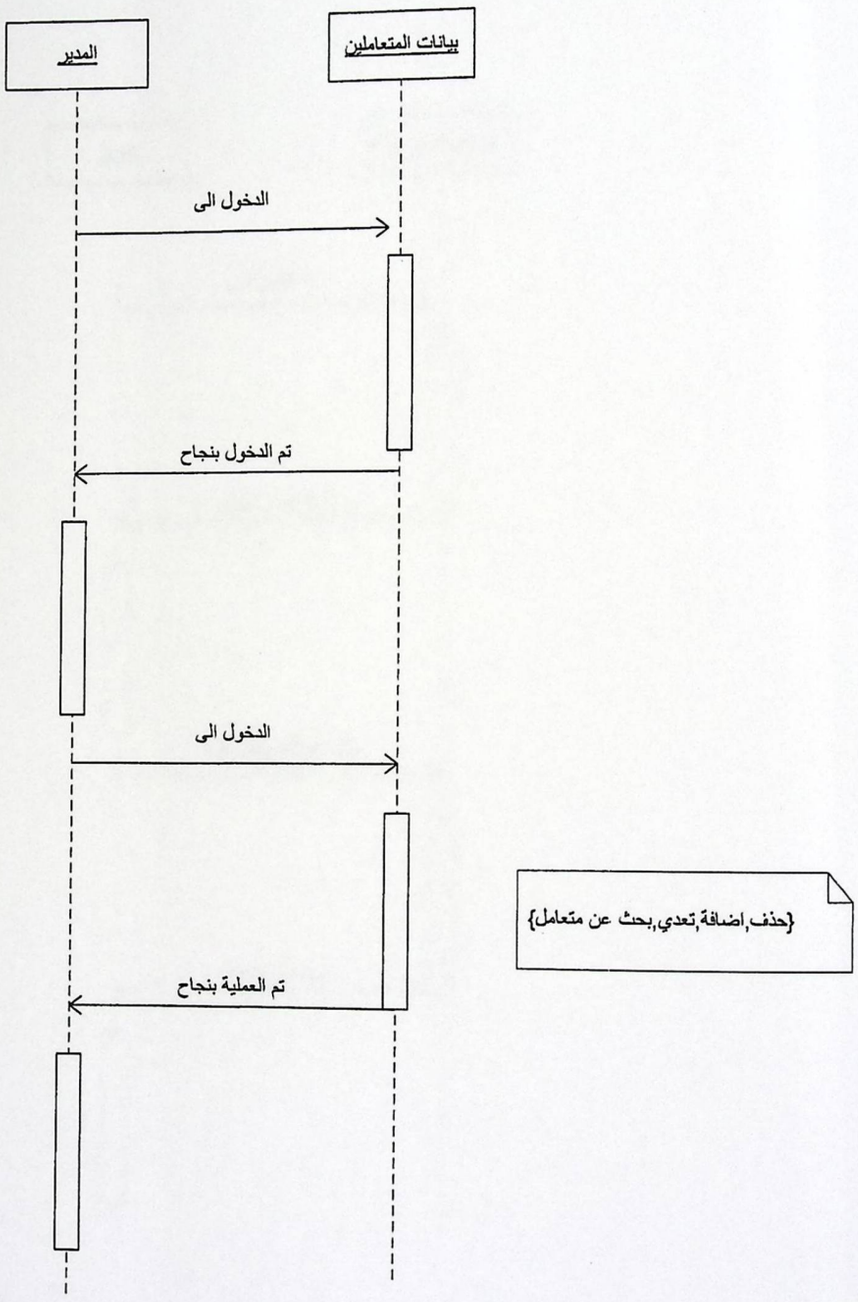
الشكل 3.14 مخطط تتابع عمليات المستخدم مع صفحة بيانات السلع

3. الشكل الثالث : توضح تتابع العمليات التي يقوم بها المستخدم مع الصفحة التابعة للعاملين :



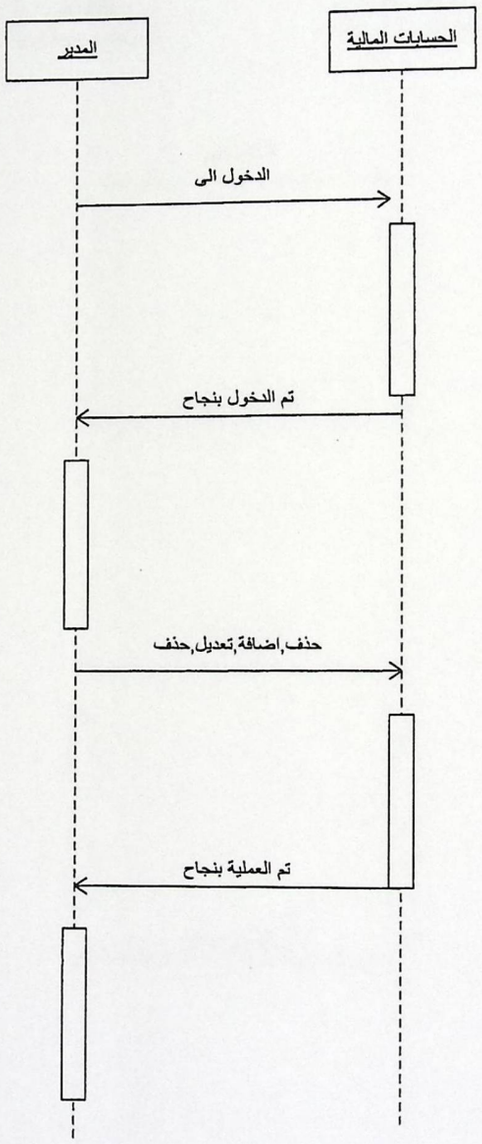
الشكل 3.15 مخطط تتابع عمليات المستخدم مع صفحة بيانات العاملين

4. الشكل الرابع : توضح تتابع العمليات التي يقوم بها المستخدم مع الصفحة التابعة صفحة التابعة لبيانات المتعاملين :



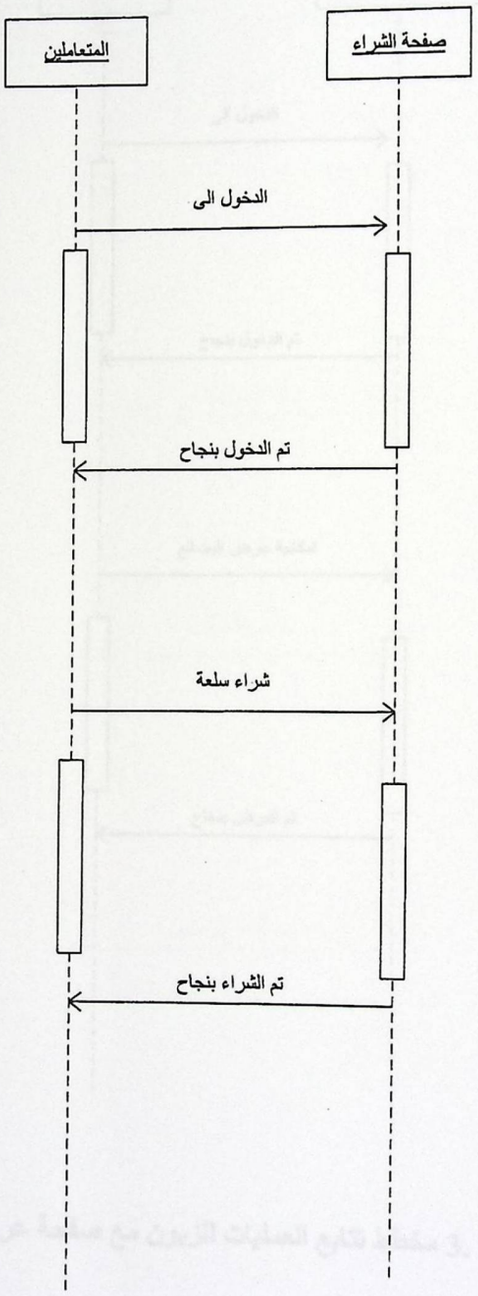
الشكل 3.16 مخطط تتابع العمليات المستخدم مع صفحة لبيانات المتعاملين

5. الشكل الخامس : توضح تتابع العمليات التي يقوم بها المستخدم مع الصفحة التابعة صفحة التابعة للحسابات المالية :



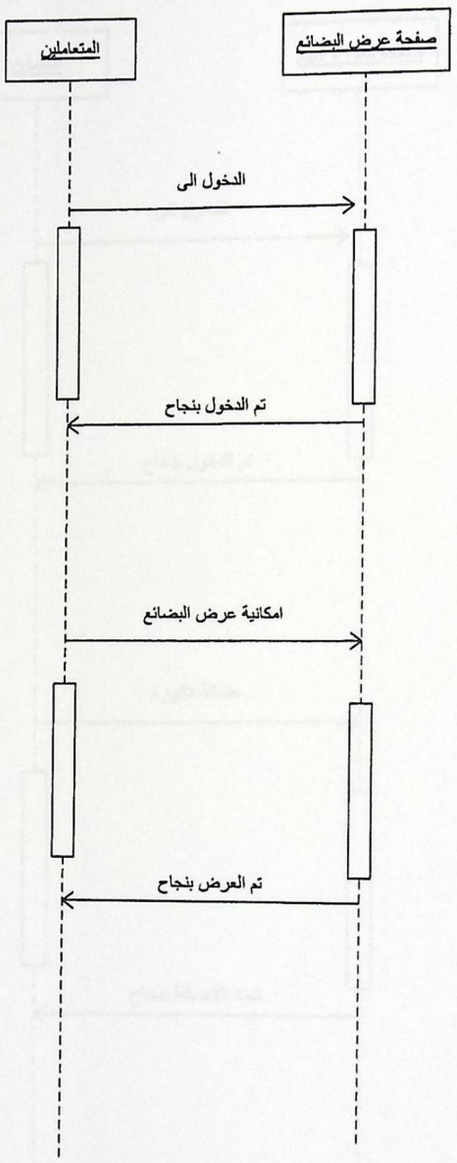
الشكل 3.17 مخطط تتابع العمليات المستخدم مع صفحة العمليات المالية

6. الشكل السادس : توضح تتابع العمليات التي يقوم بها الزبون مع الصفحة التابعة لقسم المشتريات



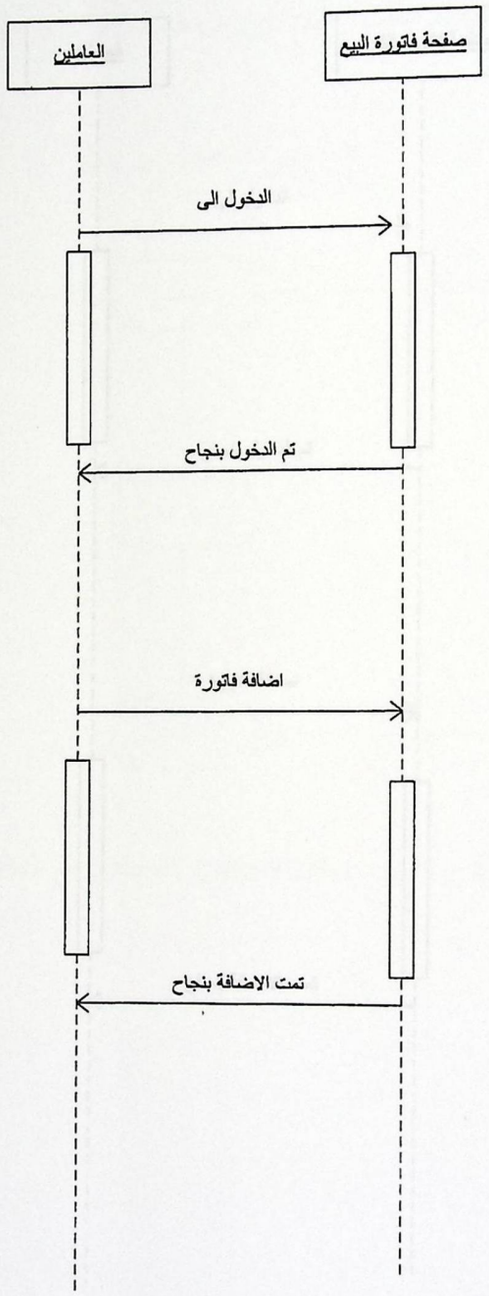
الشكل 3.18 مخطط تتابع العمليات للزبون مع صفحة قسم المشتريات

7. الشكل السابع: توضح تتابع العمليات التي يقوم بها الزبون مع الصفحة لعرض البضائع



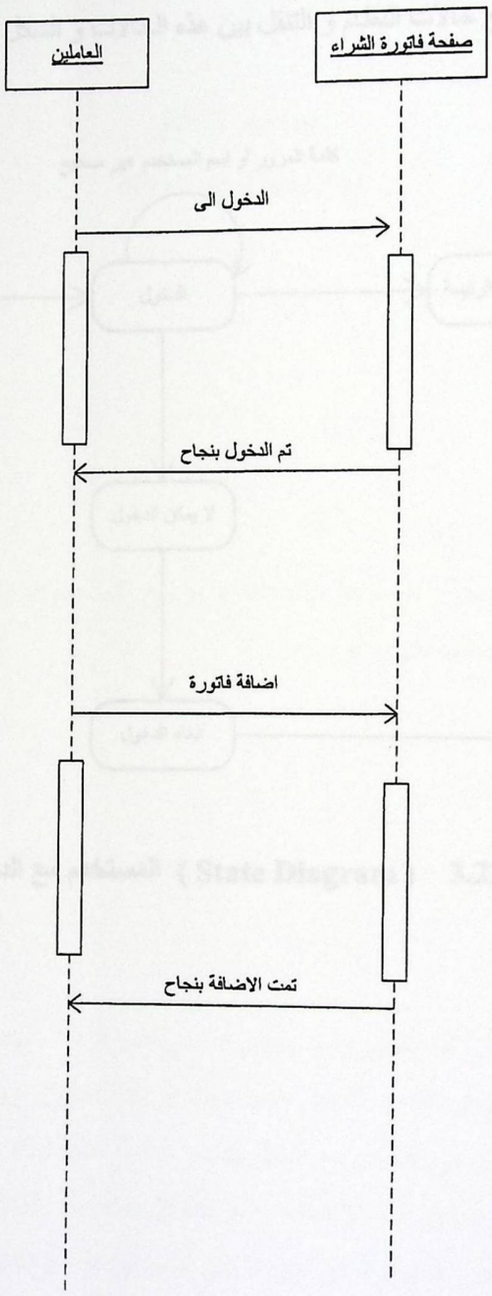
الشكل 19 .3 مخطط تتابع العمليات للزبون مع صفحة عرض البضائع

8. الشكل الثامن: توضح تتابع العمليات التي يقوم بها العامل مع الصفحة فاتورة بيع



الشكل 3.20 مخطط تتابع العمليات للعامل مع صفحة فاتورة بيع

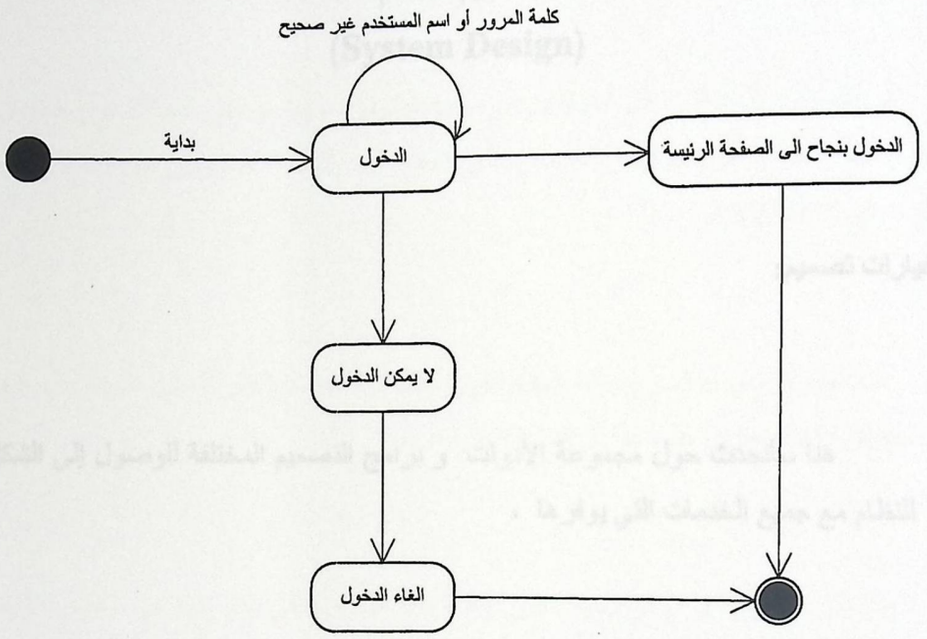
9. الشكل التاسع: توضح تتابع العمليات التي يقوم بها العامل مع الصفحة فاتورة شراء



الشكل 3.21مخطط تتابع العمليات للعامل مع صفحة فاتورة شراء

3.4.4 مخططات التنقل بين حالات النظام (State Diagrams) .

هذا المخطط توضح حالات النظام و التنقل بين هذه الحالات و الشكل موضح لذلك في هذا النظام:



الشكل 3.22 (State Diagram) المستخدم مع الدخول إلى النظام

الوحدة الرابعة

تصميم النظام
(System Design)

4.1 خيارات تصميم:

هنا سأتحدث حول مجموعة الأدوات و برامج التصميم المختلفة للوصول إلى الشكل النهائي للنظام مع جميع الخدمات التي يوفرها ،

4.1.1 نظام إدارة قواعد البيانات Microsoft SQL Server 2000:

إن موقع المدير المالي الشخصي يعتمد بشكل أساسي على قواعد البيانات حيث أن جميع أقسامه بحاجة إلى التعامل مع نظام قاعدة البيانات و التعامل معها حيث أن هذا المشروع يعتمد بشكل أساسي على السرعة في الأداء و السرية و الأمان في التعامل مع المعلومات و فعالية عالية لذلك كان لا بد من اختيار نظام لإدارة قواعد البيانات يعمل على تحقيق هذه الأهداف و هوMicrosoft SQL Server 2000 الذي يدعم بقوة Stored Procedures غير المتوفرة في غيرها من نظم قواعد البيانات القاصر عن هذه الخطوة رغم وجودها مجانية في الأسواق مثل MYSQL فلولا مساوئ هذا المنتج الكبيرة لما طرح بشكل مجاني في الأسواق كما أن الأعمال التي لا بد أن تكون ذات أهداف قوية وكبيرة تحتاج إلى السرية والسرعة لذا فمن الممكن التضحية بمجانية البرامج لتقديم منتج يحظى باحترام المستفيدين منه ويحقق لهم السرعة والأمان،

لهذا السبب والعيوب المتوفرة بكثرة لدى MYSQL عدلت الخيار من MYSQL إلى نظام قواعد بيانات آمن وسريع ومتوفر بأسعار مناسبة مع ما سيقدمه من نتائج.

4.1.2 بيئة البرمجة ASP.NET:

في البداية ترددت كثيرا في الخيارات البرمجية المتوفرة فمنها ما عفي عنها الدهر وأصبحت كلاسيكية PHP ومنها ما هو متطور وملاحق للخدمات التي تحتاج إليها الشبكة العنكبوتين مثل بيئة البرمجة المتكاملة والتي تدعى ASP.NET .

عند المقارنة بين المال و وجودة المنتج تكون الغلبة دائما للمنتج على المال بشرط عدم الخسارة ومن هذا المنطلق وقع اختياري على بيئة ASP.NET .

من اجل ما سيق ذكره وقع الخيار على بيئة ASP.NET والتي تكاد تخلو من العيوب الأساسية التي تعيق تصميم وبرمجة صفحات الانترنت.

4.1.3 لغة النظام (اللغة العربية) :

في الكتاب الموجه من عميد كلية الهندسة والتكنولوجيا إلى الطلبة الخريجين من الجامعة إن اللغة العربية هي اللغة المفضلة لطلبة مشاريع التخرج.

اضافة إلى ما تم ذكره فان استخدام اللغة العربية يقلل من عمليات الاقتباس و السرقة من البرامج و المشاريع الأخرى.

4.1.4 اختيار بناء للمشروع وليس تطوير على مشاريع سابقة:-

إن عملية التطوير على مشاريع موجودة مسبقا ليست بحاجة إلى مشروع تخرج للقيام بها وانما تحتاج الى ايام قليلة للقيام بعملية التطوير وذلك للأسباب التالية:-

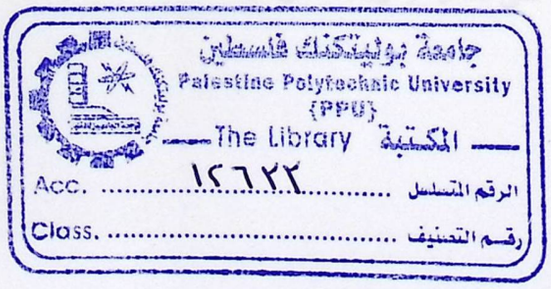
1- أن بيانات المشاريع الموجودة مسبقا عادة لا يتم التعديل عليها أو على بنيتها وهي جاهزة للاستخدام وقد حدث عليها جميع العمليات المسبقة وانما التعديل يكون على نوع قاعدة البيانات المستخدمة .

2- إن التعديل الذي يمكن اجراءه على مشاريع مسبقة هو فقط في لغة البرمجة والتي لا تحتاج الى الكثير من الوقت لتعديلها.

3- أن التعديل يجب أن يشتمل على التطوير إضافة الى التعديل وبعد ذلك ربما يكون المشروع جدير بالنظر اليه،مع العلم بأن الكثير من المبرمجين لا يعتبرون التطوير على المشاريع التي تكون من نوع قواعد البيانات عملا كبيرا.

4.1.5 لماذا تم اختيار المشروع :-

لا بد أن يكون اختيار المشروع واقعيا بحيث يتم دراسة القطاع الذي سوف يستفيد منه وتم اختياره بناء على وجود تصور عن هذا المشرع.



4.2 معلومات تفصيلية عن مكونات النظام :

هذه المرحلة من التصميم للنظام تتضمن شرح مفصل للبيئة التي يعمل بها النظام، وفي هذا الفصل نوضح الشرح التفصيلي لكل الأقسام التي تم برمجتها من ناحية التطبيع للنظام (System Normalization) بشكل دقيق حيث تم إنتاج العديد من الجداول للأقسام المكونة للنظام ، وكانت هذه الجداول الناتجة من عملية التطبيع خالية من أي أعمدة(Attribute) مكررة في أكثر من جدول ،وقبل عملية دمج الأقسام الأربعة الرئيسية للمشروع كان يتواجد القليل من الجداول تتشابه في بعض أو جميع الأعمدة وقد تمت عملية دمج هذه الأقسام بمراعاة أولا عدم تكرار الاعمد في أكثر من جدول ومن ناحية أخرى عدم ضياع الأهداف والخدمات الأساسية لهذه الجداول.

كما أن هذه الوحدة تبين تصميم الكائنات(Classes & Objects) للأقسام الأربعة الأساسية في هذا النظام قبل عملية الدمج بحيث أن هذه الخطوة تحتوي على مجموعة من المراحل المتتالية، حيث أن المرحلة الأولى كانت تصميم الكائنات من دون العناصر التي تحويها والخدمات التي تقدمها وبدون الروابط بين هذه الكائنات علما أن هذه المرحلة كانت قد ذكرت مسبقا، والمرحلة الثانية تم ربط هذه الكائنات الفارغة والعلاقات الموجودة بين هذه الكائنات، والمرحلة الثالثة تصميم الكائنات باحتوائها العناصر من دون الخدمات التي تحويها وبوجود الروابط التي تربط الكائنات ببعضها البعض، والمرحلة الرابعة تصميم الكائنات باحتوائها العناصر والخدمات التي تقدمها والروابط التي تربط بين الكائنات.

4.2.1 مكونات النظام:

4.2.1.1 التطبيع :

في هذه الجزء من الوحدة نوضح الشرح التفصيلي لكل الأقسام التي تم برمجتها من ناحية التطبيع للنظام (System Normalization) بشكل دقيق حيث تم إنتاج العديد من الجداول المكونة للقسم الإداري وكانت النتيجة أن هذه الجداول خالية من أي أعمدة (Attribute) مكررة في أكثر من جدول.

جدول 4.1 التطبيع

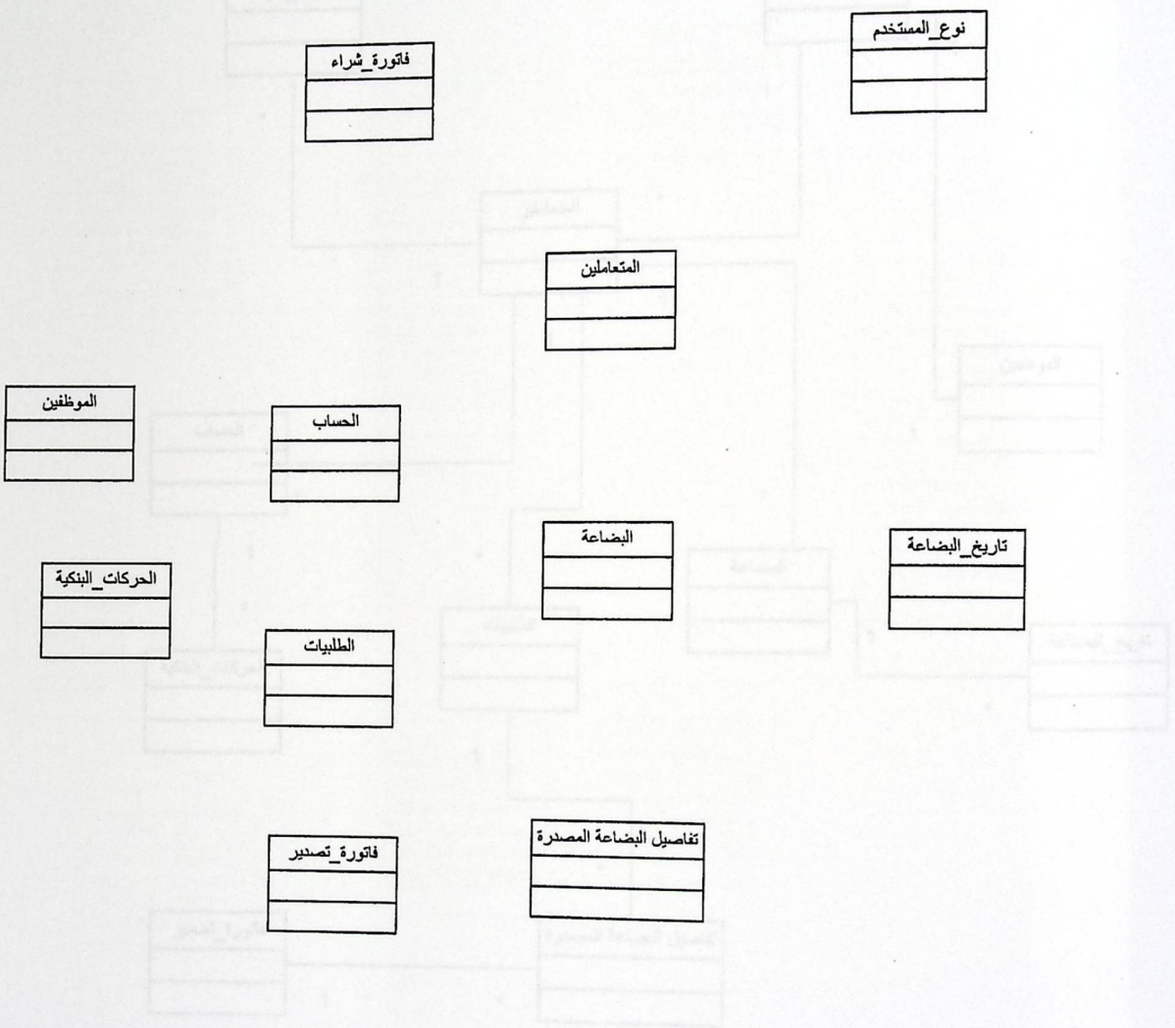
البيانات بدون تطبيع	الشكل الأول للتطبيع	الشكل الثاني للتطبيع	الشكل الثالث للتطبيع	الجدول
رقم_الموظف	رقم_الموظف	رقم_الموظف	رقم_الموظف	بيانات_الموظفين
اسم_الموظف	اسم_الموظف	الاسم_الاول_للموظف	الاسم_الاول_للموظف	
رقم_الهاتف	رقم_الهاتف	الاسم_الثاني_للموظف	الاسم_الثاني_للموظف	
راتب_الموظف	راتب_الموظف	الاسم_الثالث_للموظف	الاسم_الثالث_للموظف	
رقم_الحساب	رقم_الحساب	الاسم_الرابع_للموظف	الاسم_الرابع_للموظف	
الموقع_الوظيفي	الموقع_الوظيفي	وقت_عمل_الموظف	وقت_عمل_الموظف	
رقم_المورد		رقم_الهاتف	رقم_الهاتف	
اسم_المورد	رقم_المورد	عنوان_الموظف	عنوان_الموظف	
رقم_الحساب	اسم_المورد	مؤهل_الموظف	مؤهل_الموظف	
اسم_البنك	رقم_الحساب	راتب_الموظف	راتب_الموظف	
نوع_الحساب		الحالة_الاجتماعية	الحالة_الاجتماعية	
رقم_المستورد	اسم_البنك	عدد_الاولاد	عدد_الاولاد	
اسم_المستورد	نوع_الحساب	عدد_سنوات_الخبرة	عدد_سنوات_الخبرة	
رقم_السلعة		رقم_الحساب_للموظف	رقم_الحساب_للموظف	
اسم_السلعة	رقم_المستورد	الموقع_الوظيفي	الموقع_الوظيفي	
كمية_السلعة	اسم_المستورد			
ثمن_السلعة		رقم_المورد	رقم_المتعامل	المتعاملين
تاريخ_انتهاء_السلعة	رقم_السلعة	اسم_المورد	نوعية_المتعامل	
تاريخ_انتاج_السلعة	اسم_السلعة	رقم_الحساب	الاسم_الاول_المتعامل	
رقم_الفاتورة	كمية_السلعة	عنوان_المورد	الاسم_الثاني_المتعامل	
رقم_البضاعة	ثمن_السلعة	تلفون_المورد	الاسم_الثالث_المتعامل	
كمية_البضاعة			الاسم_الرابع_المتعامل	
سعر_البضاعة	تاريخ_انتهاء_السلعة	رقم_الحساب	عنوان_المتعامل	
تاريخ_التصدير	تاريخ_انتاج_السلعة	اسم_البنك	تلفون_المتعامل	
موقع_التصدير		نوع_الحساب	رقم_حساب_المتعامل	
نوع_المستخدم	رقم_الفاتورة		اسم_شركة_المتعامل	
اسم_المستخدم	رقم_البضاعة	رقم_المستورد	العنوان_الالكتروني_للمتعامل	

كلمة_المرور	كمية_البضاعة	اسم_المستورد		
	سعر_البضاعة	عنوان_المستورد	رقم_الحساب	الحساب
	تاريخ_التصدير	رقم_الحساب	اسم_البنك	
	موقع_التصدير		نوع_الحساب	
		رقم_السلعة		
	نوع_المستخدم	اسم_السلعة		
	اسم_المستخدم	كمية_السلعة	رقم_السلعة	بيانات السلع
	كلمة_المرور	رقم_المورد	اسم_السلعة	
		ثمن_السلعة	كمية_السلعة	
		وحدة_السلعة	رقم_المورد	
			ثمن_السلعة	
		رقم_السلعة	وحدة_السلعة	
		تاريخ_الانتهاء		
		تاريخ_الانتاج	رقم_السلعة	تاريخ_السلع
			تاريخ_الانتهاء	
		رقم_الفاتورة	تاريخ_الانتاج	
		تاريخ_التصدير		
		موقع_التصدير	رقم_الفاتورة	فاتورة_التصدير
		رقم_المستورد	تاريخ_التصدير	
			موقع_التصدير	
		رقم_الفاتورة	رقم_المستورد	
		رقم_البضاعة		
		الكمية	رقم_الفاتورة	تفاصيل_البضاعة_المصدرة
		سعر البضاعة	رقم_البضاعة	
			الكمية	
		اسم_المستخدم	سعر البضاعة	
		نوع_المستخدم		
		كلمة_المرور	اسم_المستخدم	انواع_المستخدمين
			نوع_المستخدم	
			كلمة_المرور	

4.2.1.2 تصميم الكائنات المكونة للنظام

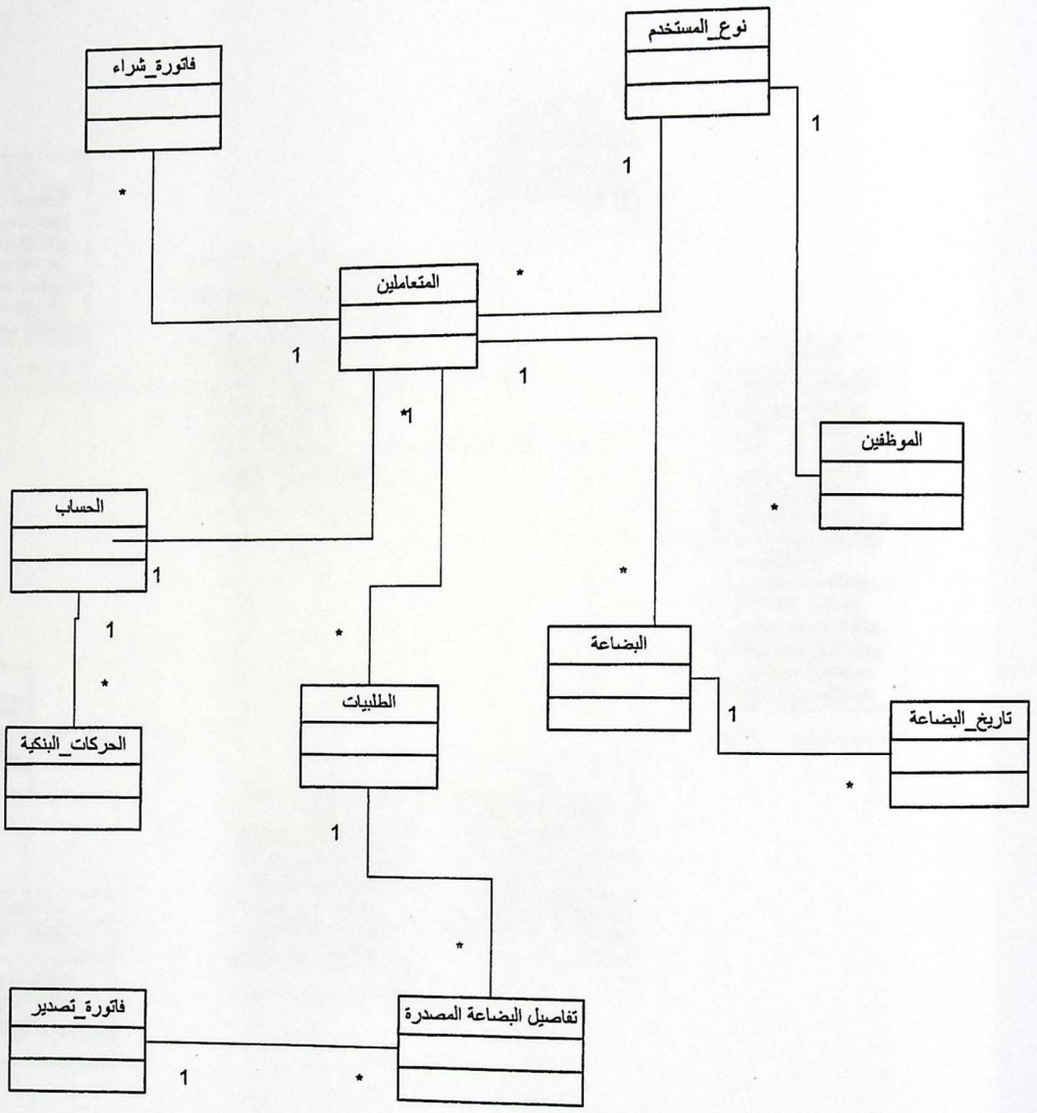
هنا نتناول بشكل تفصيلي تصميم الكائنات(Classes & Objects) حيث أن هذا القسم يحتوي على عدة خطوات متتالية، الخطوة الأولى كانت تصميم الكائنات من دون العناصر التي تحويها والخدمات التي تقدمها وبدون الروابط بين هذه الكائنات علما أن هذه الخطوة كانت قد ذكرت مسبقا، والخطوة الثانية تم ربط هذه الكائنات الفارغة والعلاقات الموجودة بين هذه الكائنات، والخطوة الثالثة تصميم الكائنات باحتوائها العناصر من دون الخدمات التي تحويها وبوجود الروابط التي تربط الكائنات ببعضها البعض، والخطوة الرابعة تصميم الكائنات باحتوائها العناصر والخدمات التي تقدمها والروابط التي تربط بين الكائنات.

4.2.1.2.1 الكائنات المجردة بدون وجود علاقات بينها أو عناصر بداخلها



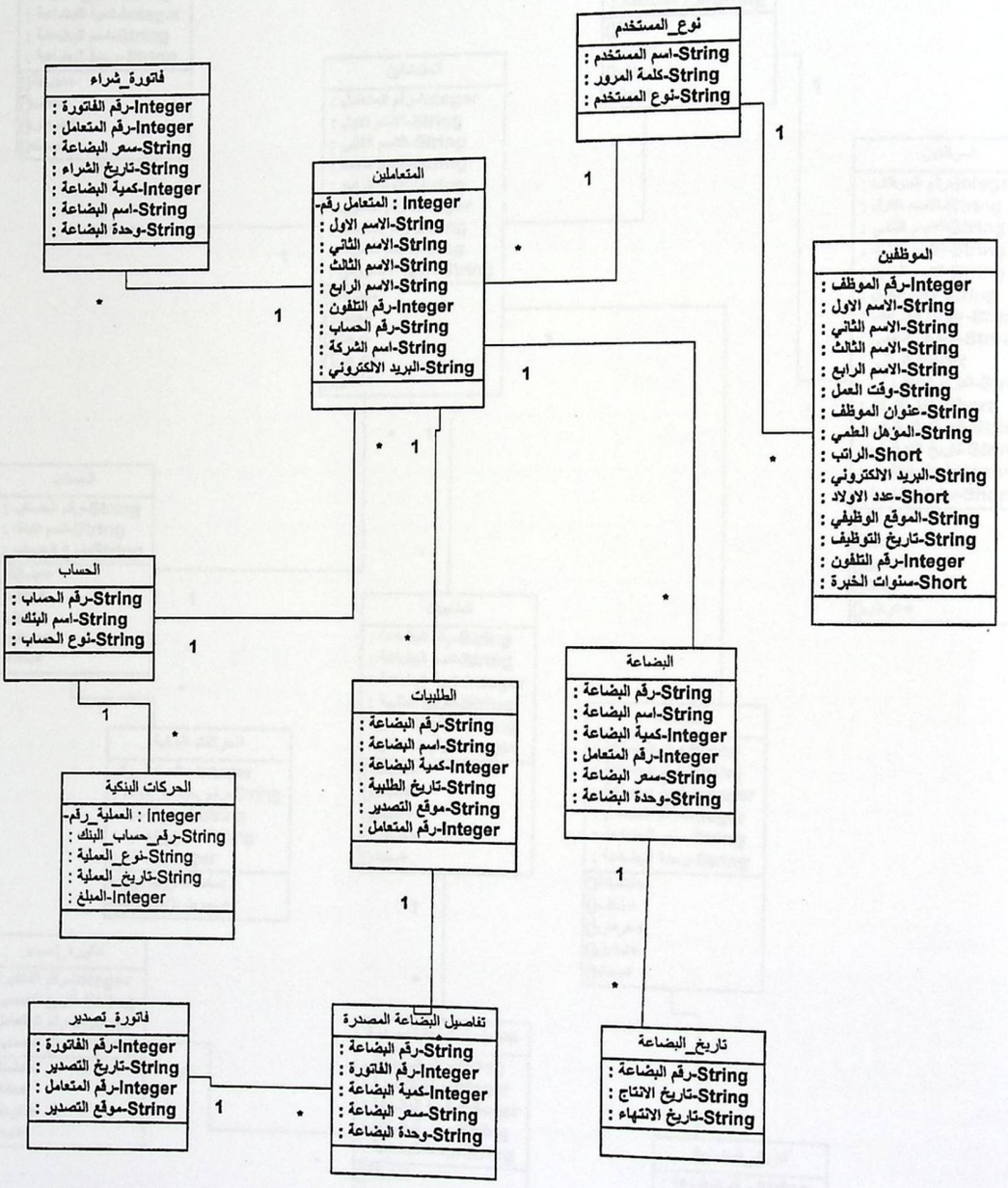
الشكل4.1:- الكائنات المجردة دون وجود علاقات بينها أو عناصر داخلها.

4.2.1.2.2 الارتباطات بين الكائنات



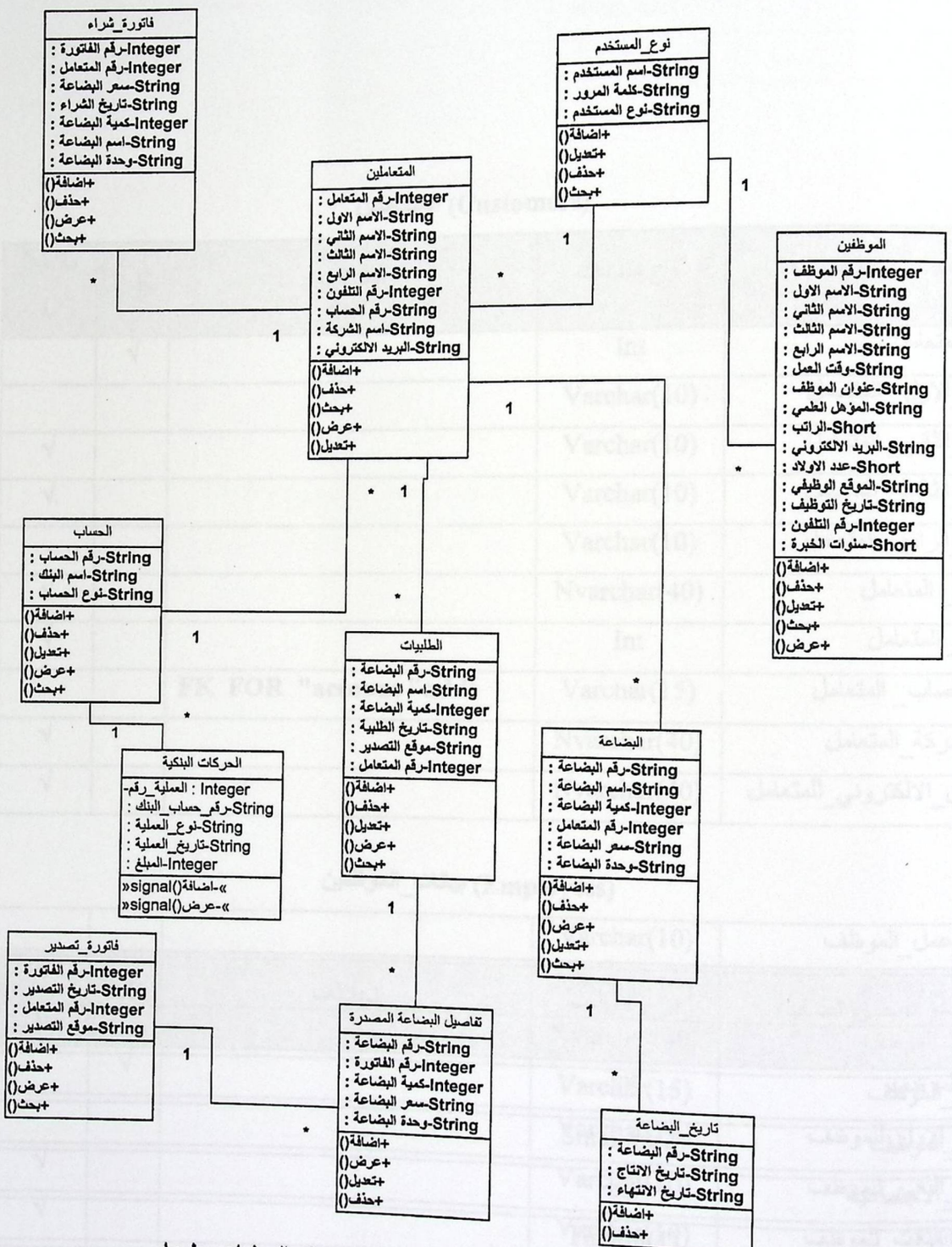
الشكل 4.2: الارتباطات بين الكائنات

4.2.1.2.3 الارتباطات بين الكائنات بوجود العناصر المكونة لهذه الكائنات



الشكل.4.3: الارتباطات بين الكائنات بوجود عناصرها.

4.2.1.2.4 الارتباطات بين الكائنات بوجود العناصر المكونة لهذه الكائنات والعمليات عليها :



الشكل 4.4 : الارتباطات بين الكائنات بوجود عناصرها والعمليات عليها

4.2.1.3.1 جداول قواعد البيانات (Database_Tables)

المتعاملين (Customers)

اسم العنصر(الصف)	نوع البيانات	العلاقات	P.K	NULL
رقم_المتعامل	Int		√	
الاسم_الاول_ المتعامل	Varchar(10)			
الاسم_الثاني_ المتعامل	Varchar(10)			√
الاسم_الثالث_ المتعامل	Varchar(10)			√
الاسم_الرابع_ المتعامل	Varchar(10)			
عنوان_ المتعامل	Nvarchar(40)			√
تلفون_ المتعامل	Int			√
رقم_حساب_ المتعامل	Varchar(15)	FK FOR "account "table		
اسم_شركة_المتعامل	Nvarchar(40)			√
العنوان_الالكتروني_للمتعامل	Nvarchar(40)			√

بيانات_الموظفين (Employess)

اسم العنصر(الصف)	نوع البيانات	العلاقات	P.K	NULL
وقت_عمل_الموظف	Varchar(10)			
رقم_الهاتف	Varchar(10)			
عنوان_الموظف	Nvarchar(40)			
رقم_الموظف	Varchar(15)		√	
الاسم_الاول_للموظف	Int			
الاسم_الثاني_للموظف	Varchar(10)			
الاسم_الثالث_للموظف	Smallint(2)			√
الاسم_الرابع_للموظف	Varchar(10)			√
عدد_الاولاد_للموظف	bit			
عدد_سنوات_الخبرة_للموظف	Varchar(10)			
	Tinyint(1)			
	Varchar(10)			
	Tinyint(1)			

		FK FOR "account_user"table	Varchar(15)	رقم_الحساب_للموظف
			Nvarchar(40)	الموقع_الوظيفي
			Varchar(15)	تاريخ التوظيف

بيانات السلع (goods)

NULL	P.K	العلاقات	نوع البيانات	اسم العنصر(الصف)
	√		Nvarchar(40)	رقم_السلعة
			Nvarchar(40)	اسم_السلعة
			Int	كمية_السلعة
√		Fk for (تفاصيل البضاعة_المصدرة)	Int	رقم_المتعامل
			Nvarchar(20)	ثمن_السلعة
			Nvarchar(20)	وحدة_السلعة

تفاصيل_ البضاعة_المصدرة (exp_invoice_details)

NULL	P.K	العلاقات	نوع البيانات	اسم العنصر(الصف)
	√	Fk for جدول_فاتورة_التصدير	Nvarchar(20)	رقم_الفاتورة
	√	Fk for جدول_البضاعة المصدرة_تاريخ السلعة	Nvarchar(40)	رقم_البضاعة
			Int	الكمية
			Nvarchar(20)	سعر_ البضاعة
			Nvarchar(20)	وحدة_السلعة

فاتورة تصدير (expinvoice)

اسم العنصر(الصف)	نوع البيانات	العلاقات	P.K	NULL
رقم_الفاتورة	Nvarchar(20)		√	
تاريخ_التصدير	Varchar(10)			
موقع_التصدير	Nvarchar(20)			
رقم_المتعامل	Int	FK for جدول_المستوردين		

فاتورة شراء (Buying invoice)

اسم العنصر(الصف)	نوع البيانات	العلاقات	P.K	NULL
رقم_الفاتورة	Nvarchar(20)		√	
تاريخ_التصدير	Varchar(10)			
موقع_التصدير	Nvarchar(20)			
رقم_المتعامل	Int	FK for جدول_المستوردين		

الحساب (Account)

اسم العنصر(الصف)	نوع البيانات	العلاقات	P.K	NULL
رقم_الحساب	Varchar(15)		√	
اسم_البنك	Nvarchar(40)			
نوع_الحساب	Nvarchar(10)			

تاريخ_السلع (GoodsDate)

اسم العنصر(الصف)	نوع البيانات	العلاقات	P.K	NULL
رقم_السلعة	Nvarchar(40)	Fk for جدول_البضاعة المصدرة_تاريخ السلعة	√	
تاريخ_الانتهاء	Nvarchar(15)		√	
تاريخ_الانتاج	Nvarchar(15)			√

نوع_المستخدم (Usertype)

اسم العنصر(الصف)	نوع البيانات	العلاقات	P.K	NULL
اسم_المستخدم	Varchar(50)		√	
نوع_المستخدم	Char(1)			
كلمة_المرور	Varchar(16)			

الطلبيات (Orders)

اسم العنصر(الصف)	نوع البيانات	العلاقات	P.K	NULL
رقم_المتعامل	int	FK for Customers	√	
رقم_البضاعة	Nvarchar(40)	FK for goods		
تاريخ الطلبية	Nvarchar(10)			
موقع التصدير	Nvarchar(40)			
اسم البضاعة	Nvarchar(40)			
كمية البضاعة	int			

فاتورة الشراء (BuyingInvoice)

اسم العنصر(الصف)	نوع البيانات	العلاقات	P.K	NULL
رقم_الفاتورة	Nvarchar(20)	FK for Customers	√	

			int	رقم_المتعامل
			Nvarchar(20)	سعر البضاعة
			varchar(10)	تاريخ الشراء
			int	كمية البضاعة
			Nvarchar(40)	اسم البضاعة
			Nvarchar(10)	وحدة البضاعة

(acoun) الحركات البنكية

NULL	P.K	العلاقات	نوع البيانات	اسم العنصر(الصف)
	√		int	رقم_الحساب
		FK for account	varchar(15)	رقم_حساب_البنك
			varchar(10)	نوع_العملية
			varchar(10)	تاريخ_الحركة
			Money(8)	المبلغ

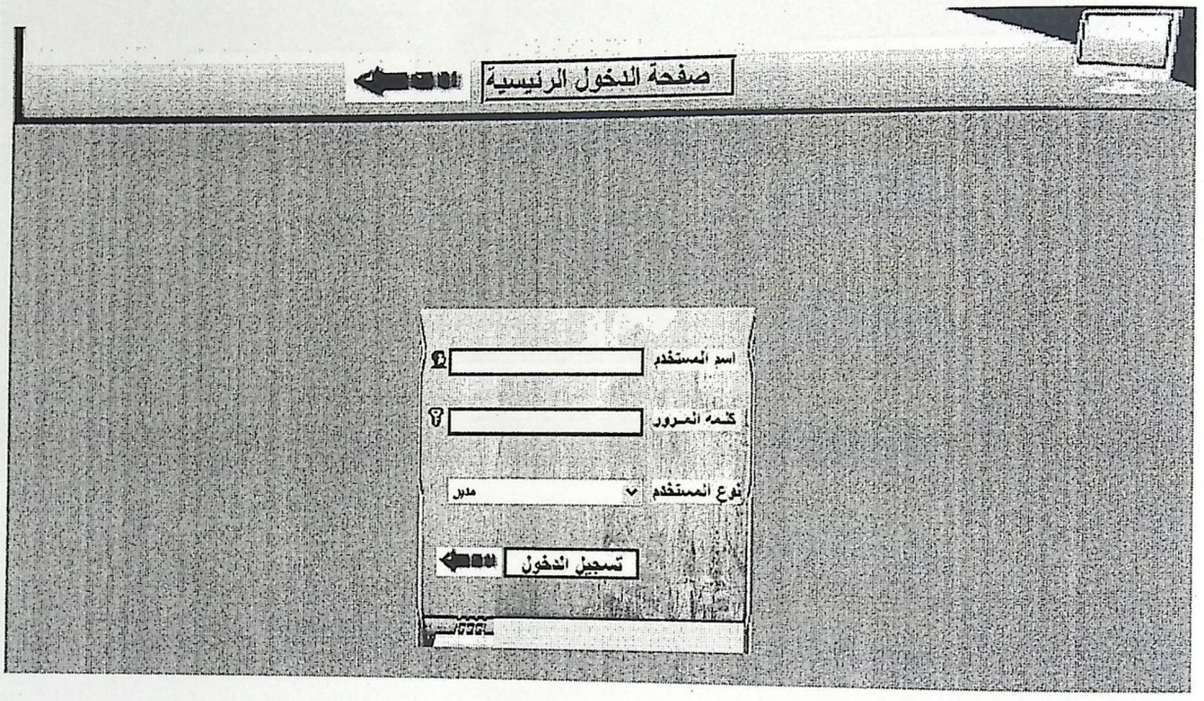
4.3 واجهات التطبيق للمستخدم (GUI) :

في هذا القسم يتم استعراض الواجهات التطبيقية التي سوف يتعامل معها المستخدم للتفاعل مع النظام سواء عرض المعلومات المتواجدة على قاعدة البيانات أو إدخال البيانات الى هذا النظام .

حيث سيتعامل المتعاملون من خلال صفحات الانترنت مع هذا النظام , و سيتم توضيح هذه الآلية من خلال الرسومات والواجهات التطبيقية التي سوف يتعامل معها المستخدم في كل مرحلة من المراحل و هذه

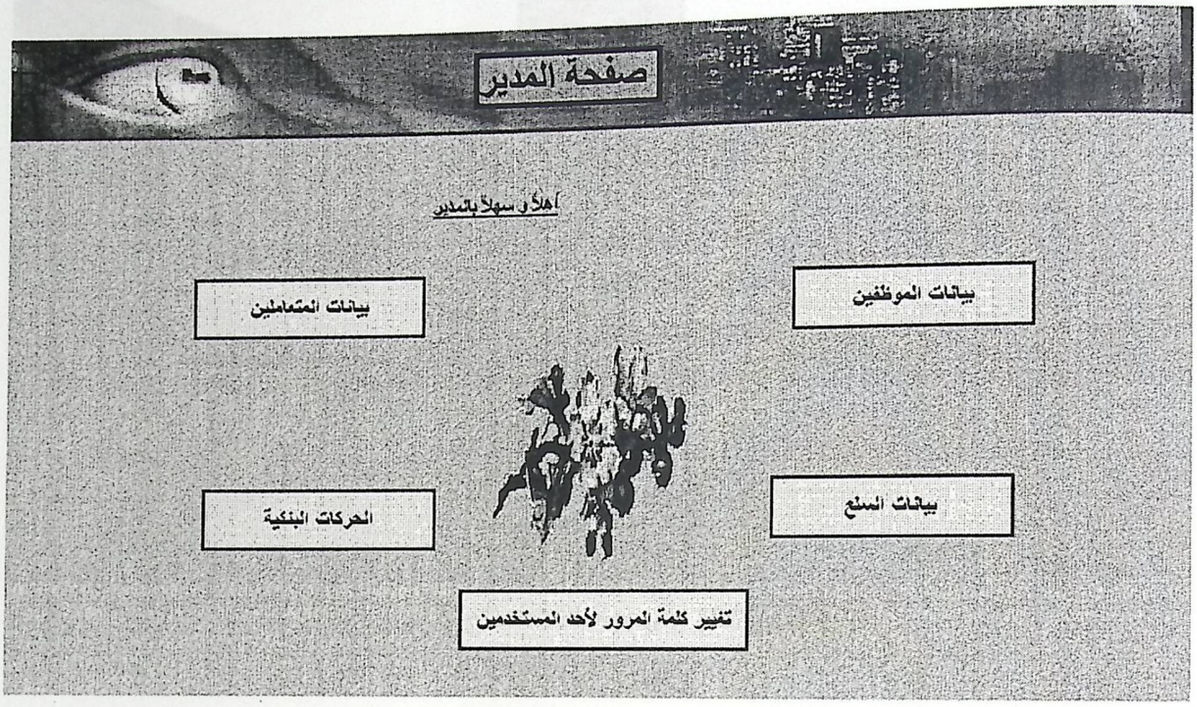
الواجهات سيتم تصميمها باستخدام بيئة البرمجة (ASP.Net) من أجل تطبيقات الانترنت , وهذه الواجهات موضحة كالتالي :

الشكل الاول يبين واجهة الدخول الرئيسية للنظام حيث يفترض أن يكون للمدير والمتعامل والموظف الاسم وكلمة المرور الخاصة به حتى يكون له القدرة و الصلاحية للدخول للنظام والاستفادة من خدماته :



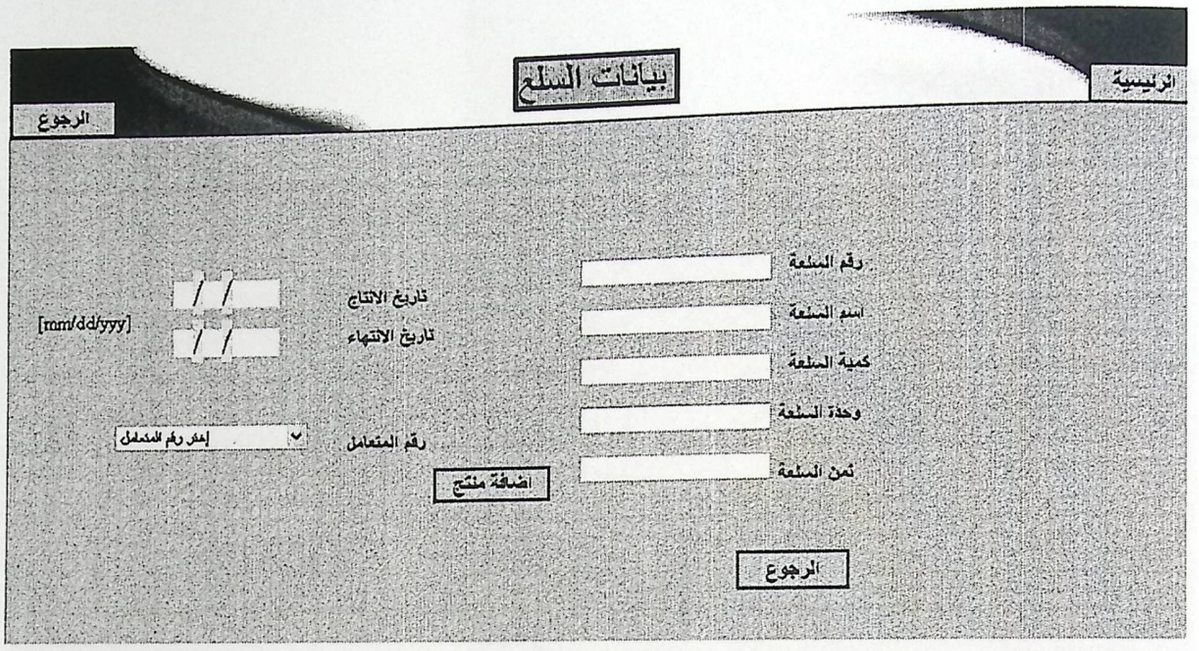
شكل (4.5) : واجهة صفحة الدخول الى النظام

الشكل الثاني يبين واجهة خيارات المدير المختلفة و هي خمسة خيارات كما هو مبين في الشكل التالي:



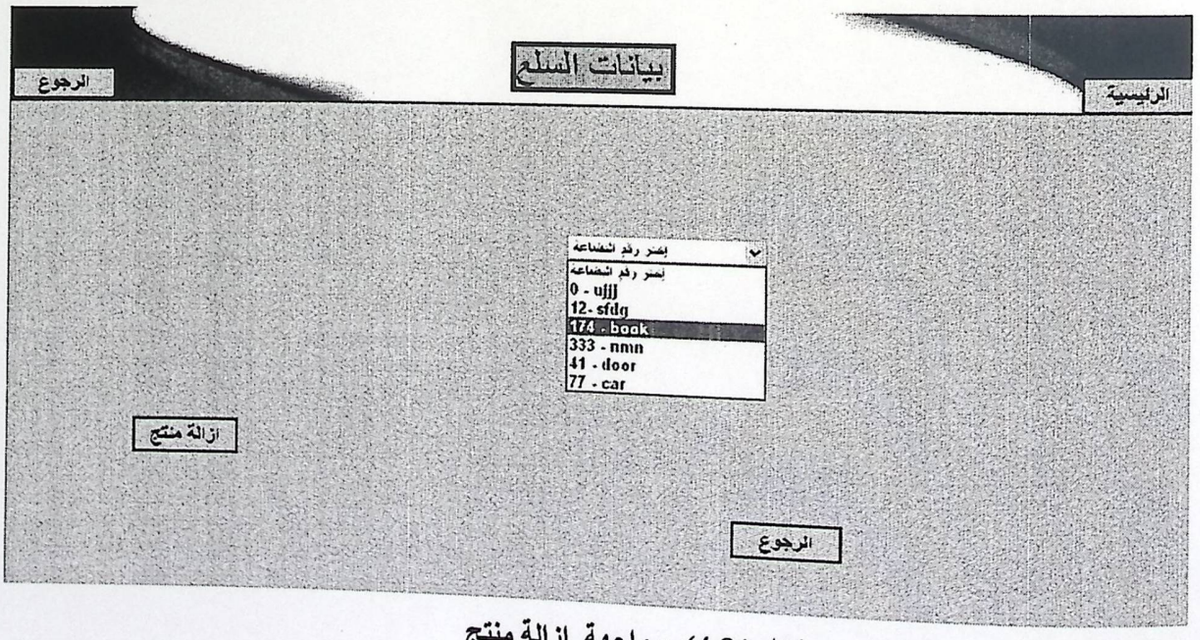
شكل (4.6) : واجهة صفحة المدير

اما الشكل التالي يبين الخيار بيانات السلع و هو عملية اضافة منتج الذي يعطي المدير القدرة على اضافته :



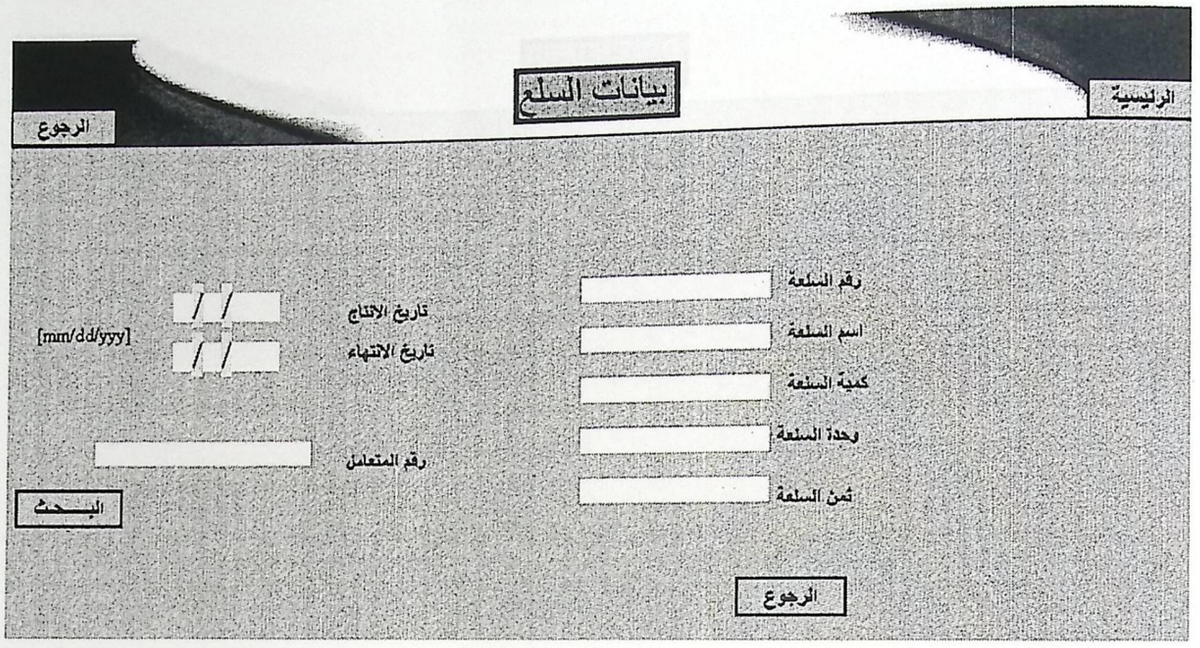
شكل (4.7) : واجهة اضافة منتج

اما الشكل التالي يبين الخيار بيانات السلع و هو عملية حذف منتج الذي يعطي المدير القدرة على حذف أي منتج :



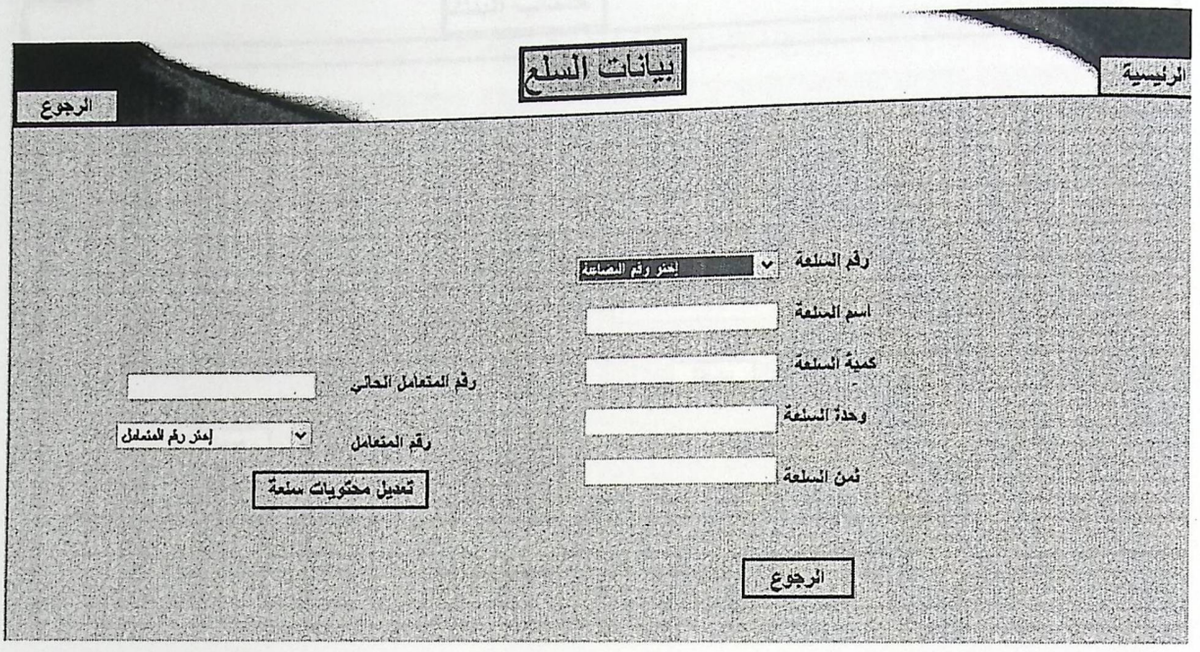
شكل (4.8) : واجهة ازالة منتج

اما الشكل التالي يبين الخيار بيانات السلع و هو عملية بحث عني أي شيء خاص بالمنتج الذي يعطي المدير القدرة على تحديده :



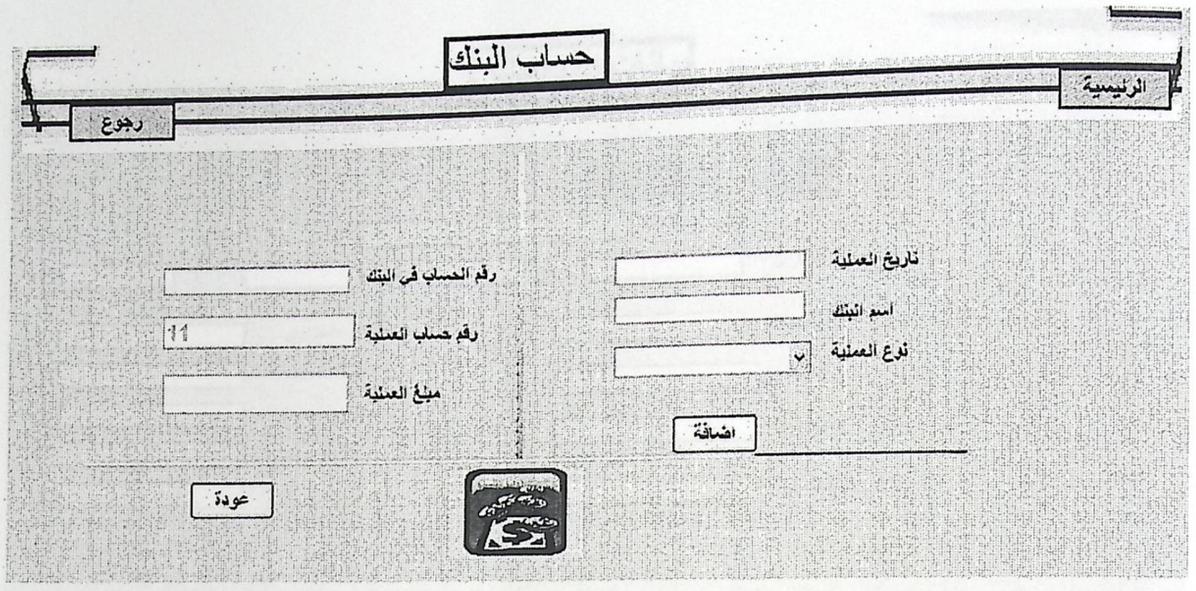
شكل (4.9) : واجهة البحث عن منتج

اما الشكل التالي يبين الخيار بيانات السلع و هو عملية التعديل علي أي شيء خاص بالمنتج الذي يعطي المدير القدرة على تحديده :



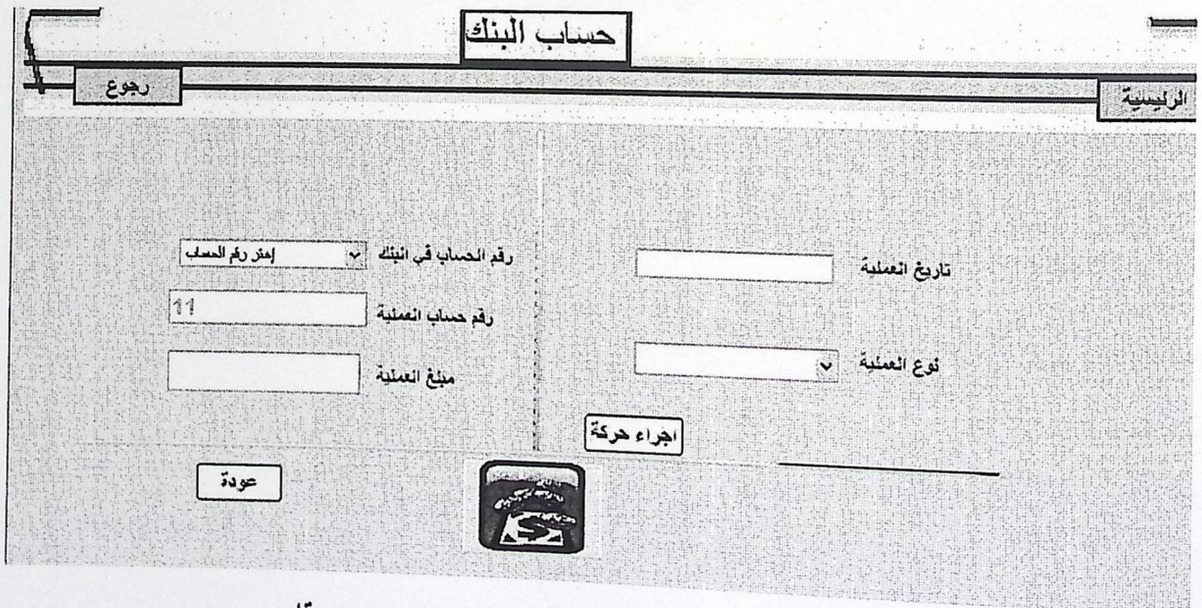
شكل (4.10) :واجهة تعديل محتويات السلعة

أما الشكل التالي يبين الخيار للحركات البنكية و هو عملية إضافة حساب جديد والحركة التي تمت عليه :



شكل (4.11) : واجهة اضافة الحسابات البنكية

أما الشكل التالي يبين الخيار للحركات البنكية و هو عملية حركة بنكية إلى حساب موجود :

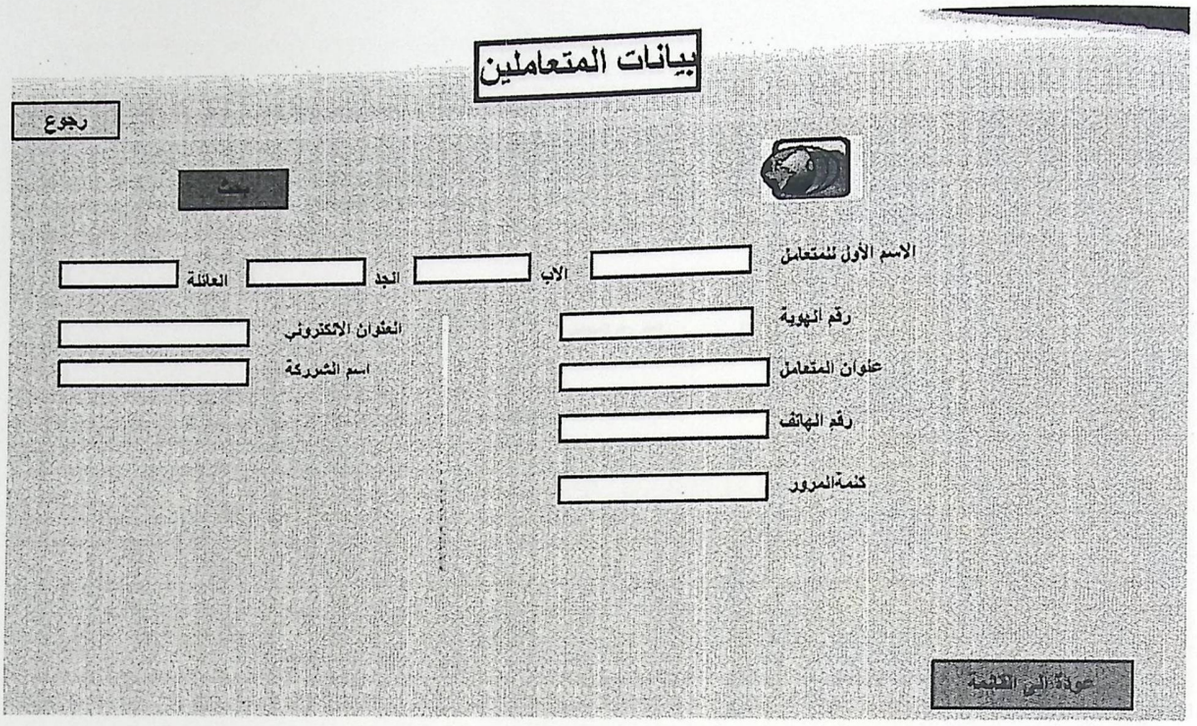


شكل (4.12) : واجهة إدراج حركة بنكية إلى حساب موجود مسبقا

أما الشكل التالي يبين الخيار لبيانات المتعاملين و هو عملية إضافة متعامل جديد إلى قائمة المتعاملين:

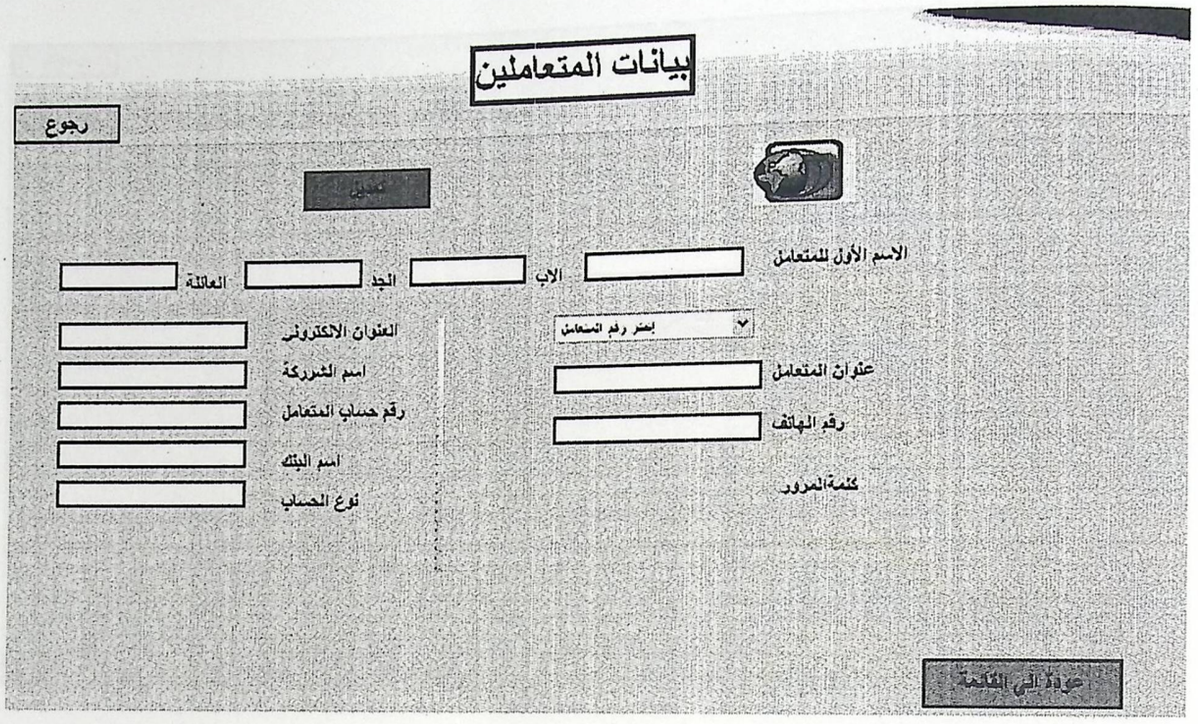
شكل (4.13) : واجهة إضافة بيانات المتعاملين

أما الشكل التالي يبين الخيار لبيانات المتعاملين و هو عملية البحث عن أي شيء خاص بالمتعامل:



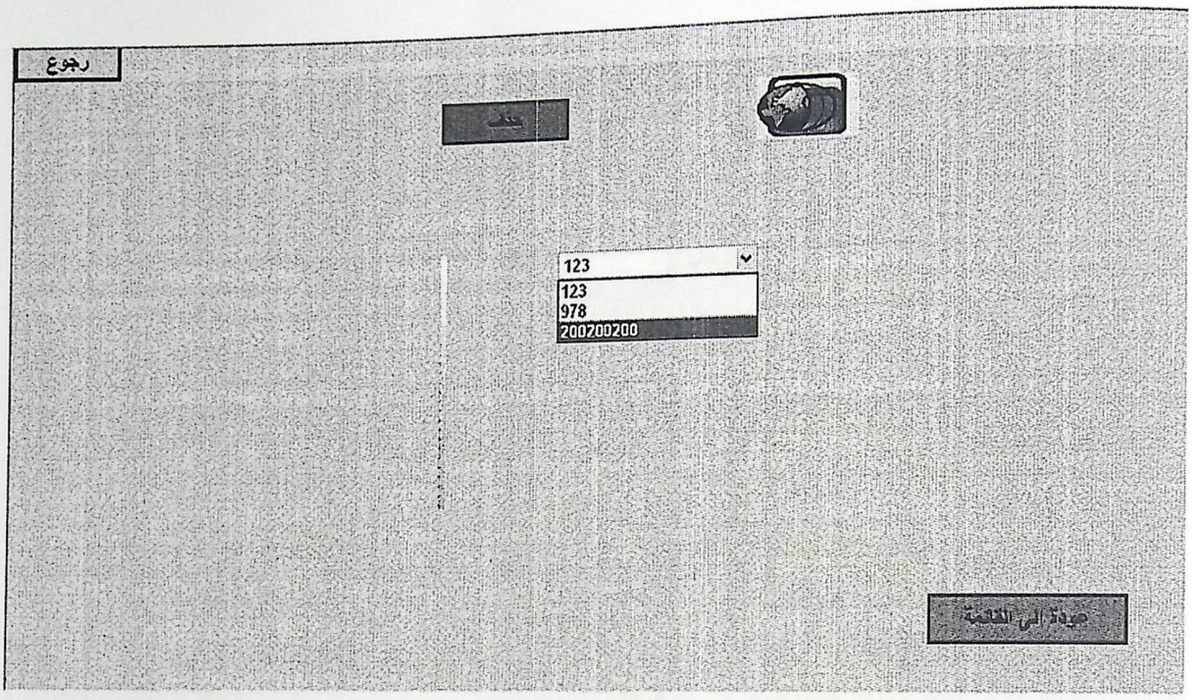
شكل (4.14) : واجهة بحث عن بيانات المتعاملين

أما الشكل التالي يبين الخيار لبيانات المتعاملين و هو عملية التعديل على أي شيء خاص بمتعامل معين:



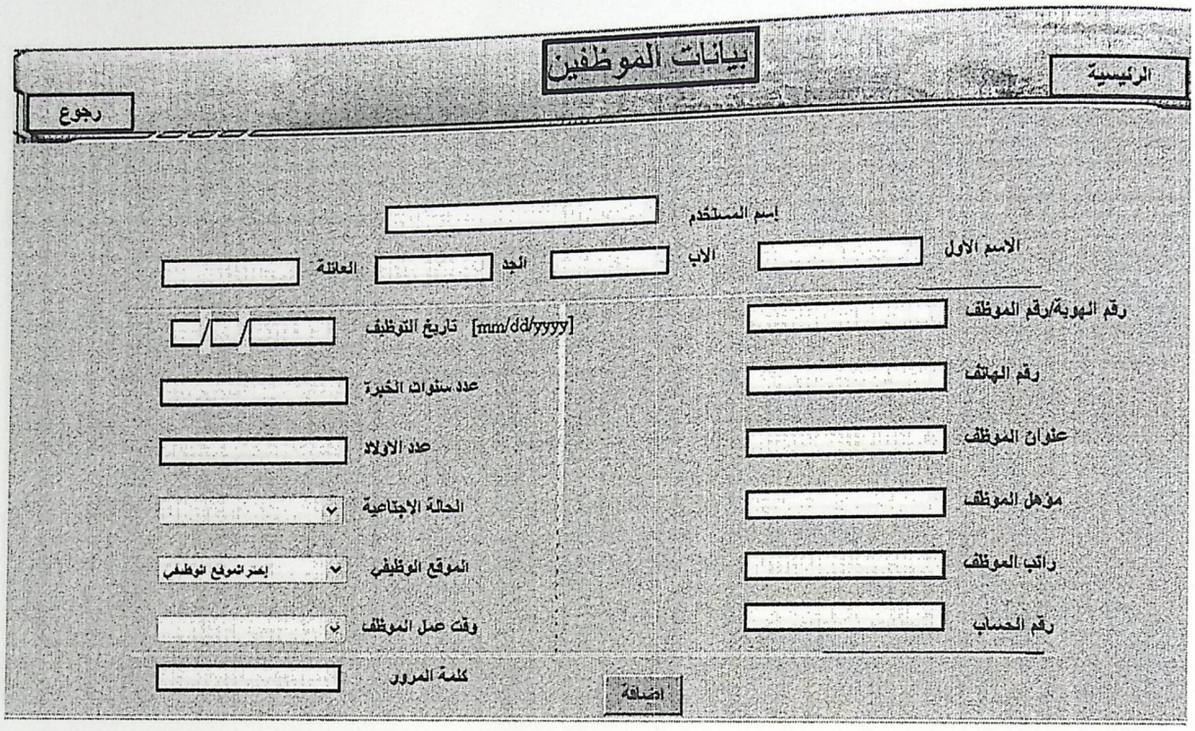
شكل (4.15) :واجهة تعديل بيانات المتعاملين

أما الشكل التالي يبين الخيار لبيانات المتعاملين و هو عملية حذف بيانات متعامل معين:



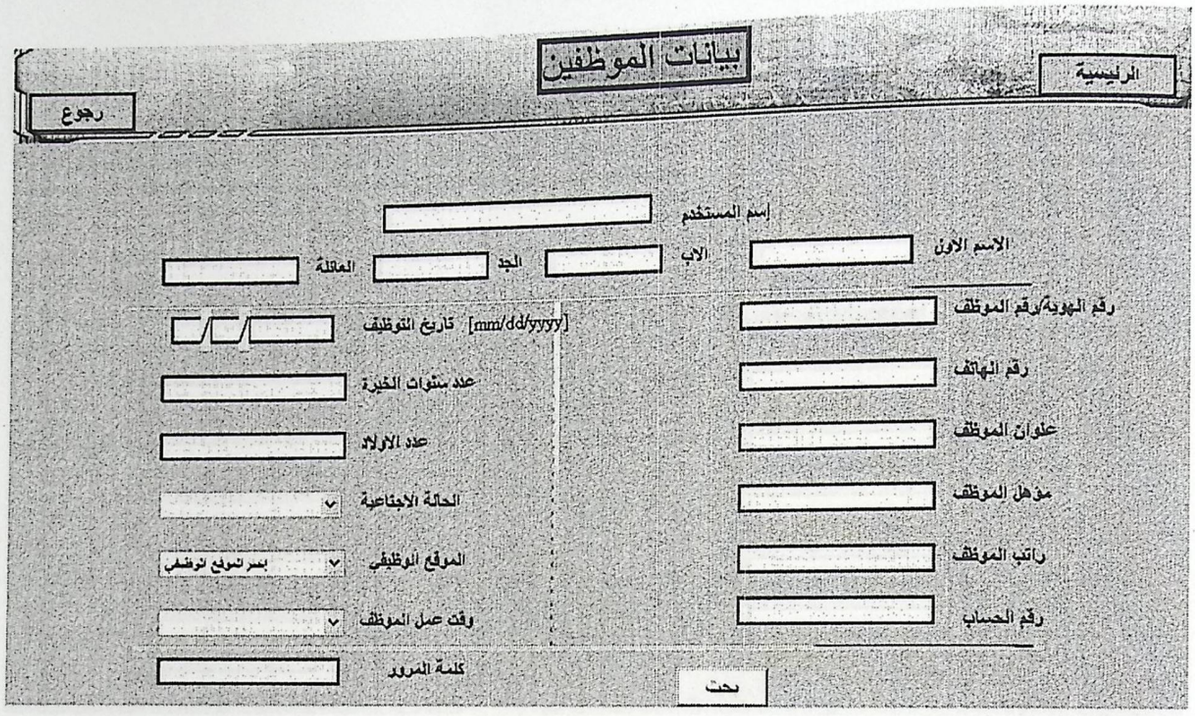
شكل (4.16) : واجهة حذف بيانات المتعاملين

أما الشكل التالي يبين الخيار لبيانات الموظفين و هو عملية إضافة موظف إلى قائمة الموظفين:



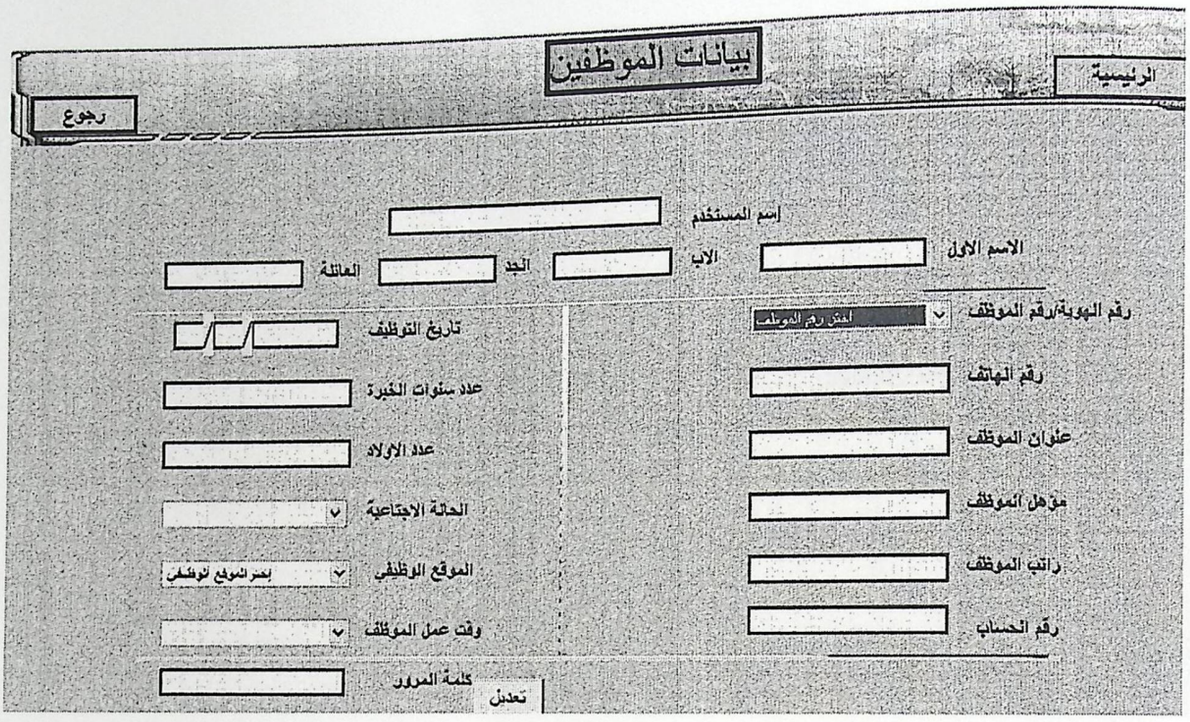
شكل (4.17) :واجهة اضافة بيانات الموظفين

أما الشكل التالي يبين الخيار لبيانات الموظفين و هو عملية البحث عن أي شيء خاص بالموظف:



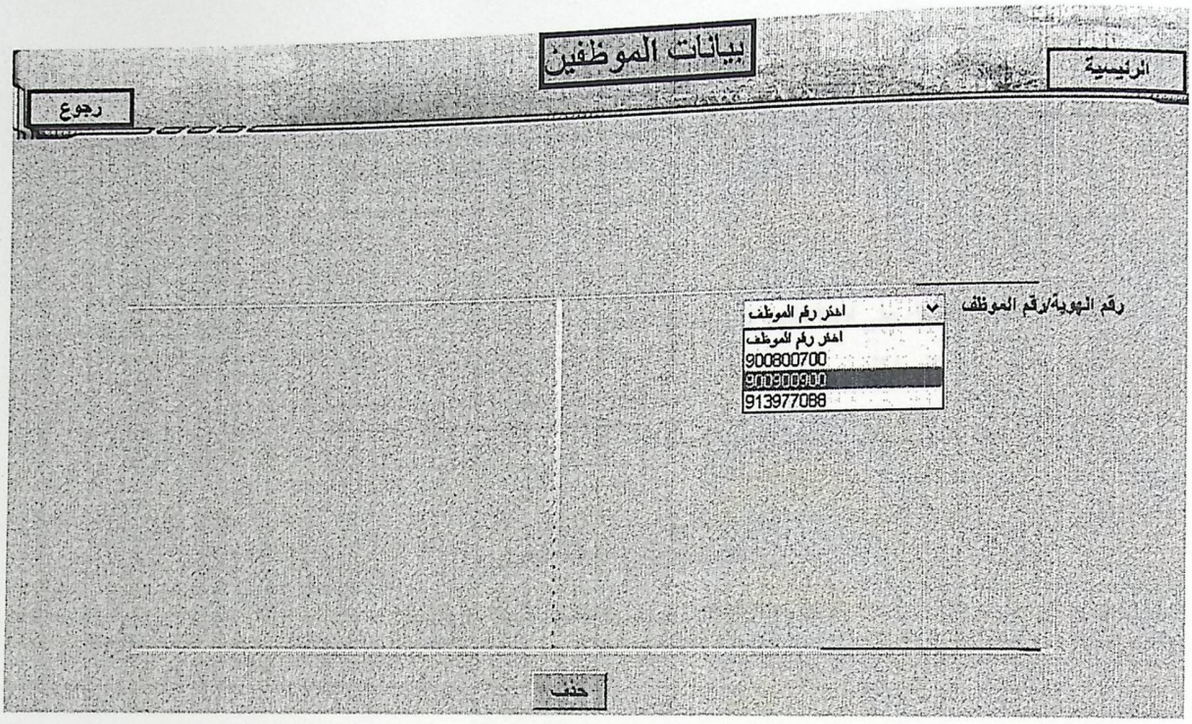
شكل (4.18) :واجهة بحث في بيانات الموظفين

أما الشكل التالي يبين الخيار لبيانات الموظفين و هو عملية التعديل على أي شيء خاص بموظف معين:



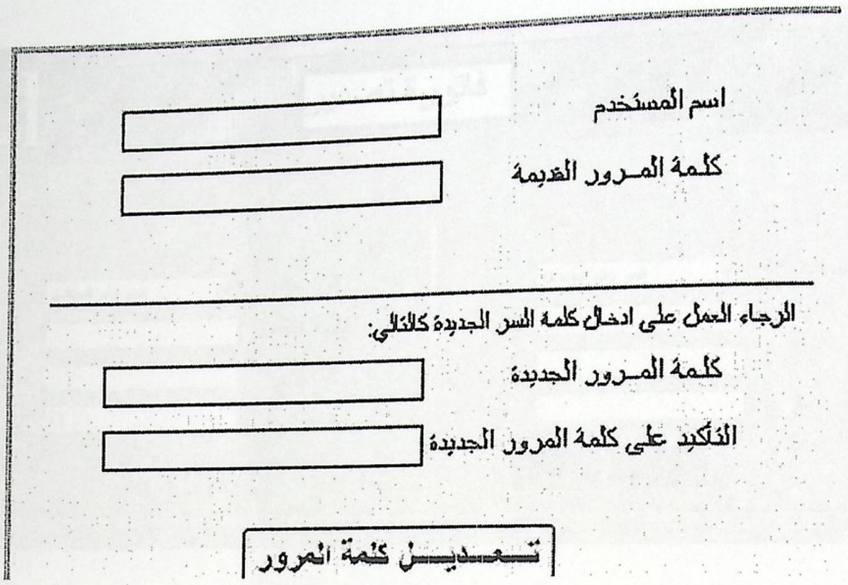
شكل (4.19) :واجهة تعديل بيانات الموظفين

أما الشكل التالي يبين الخيار لبيانات الموظفين و هو عملية الحذف على بيانات موظف معين :



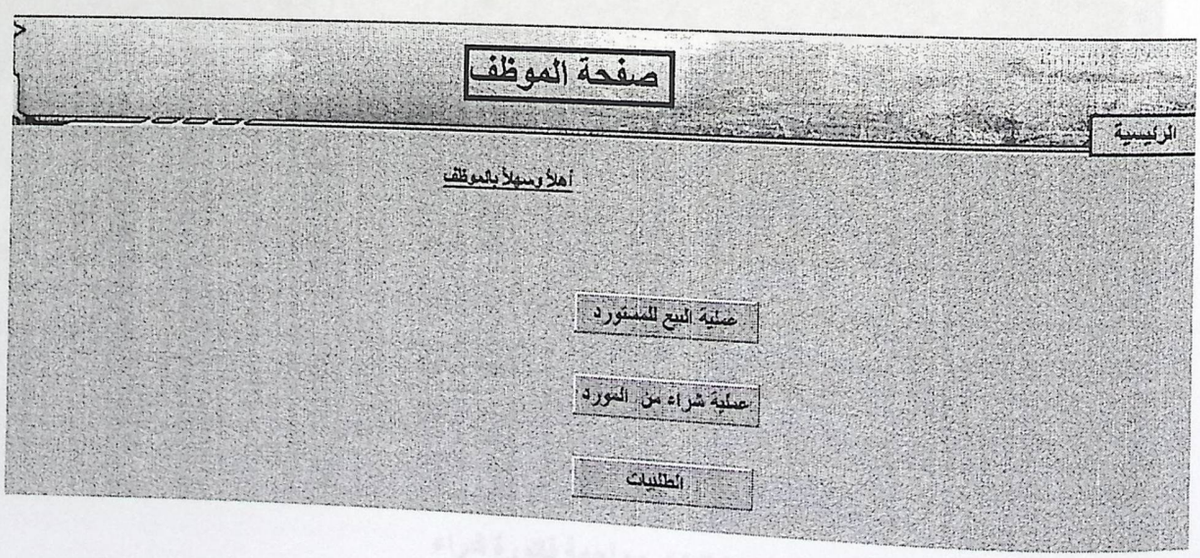
شكل (4.20) : واجهة حذف بيانات الموظفين

أما الشكل التالي يبين الخيار لتعديل كلمة مرور لأي مستخدم :



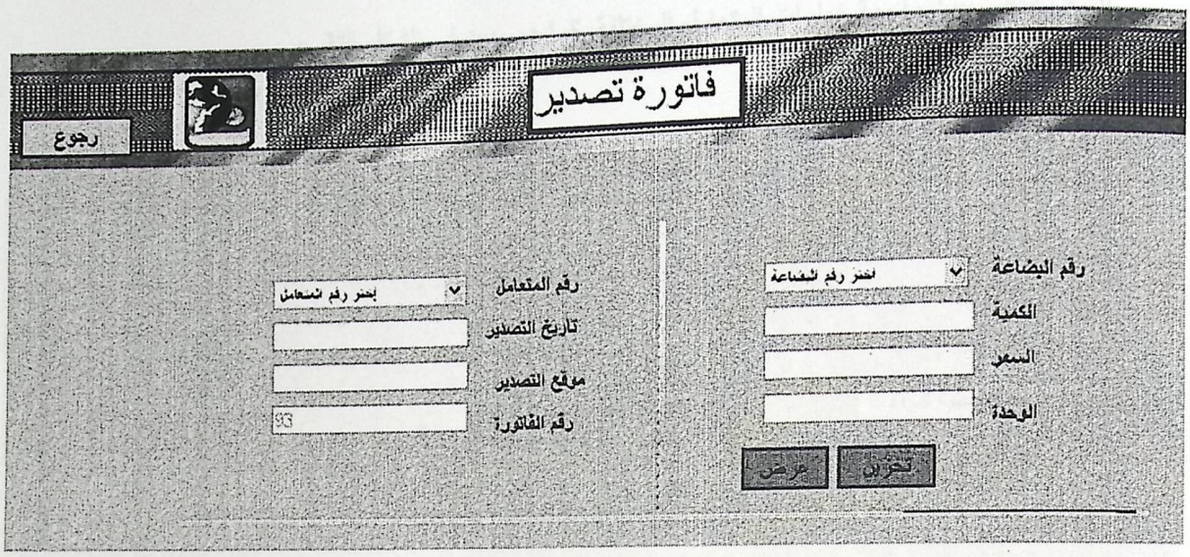
شكل (4.21) :واجهة تعديل كلمة مرور

الشكل الثاني يبين واجهة خيارات الموظف المختلفة و هي ثلاثة خيارات كما هو مبين في الشكل التالي:



شكل (4.22) :واجهة صفحة الموظف

أما الشكل التالي يبين الخيار لعملية البيع للمتعامل و هو عملية إضافة فاتورة تصدير:

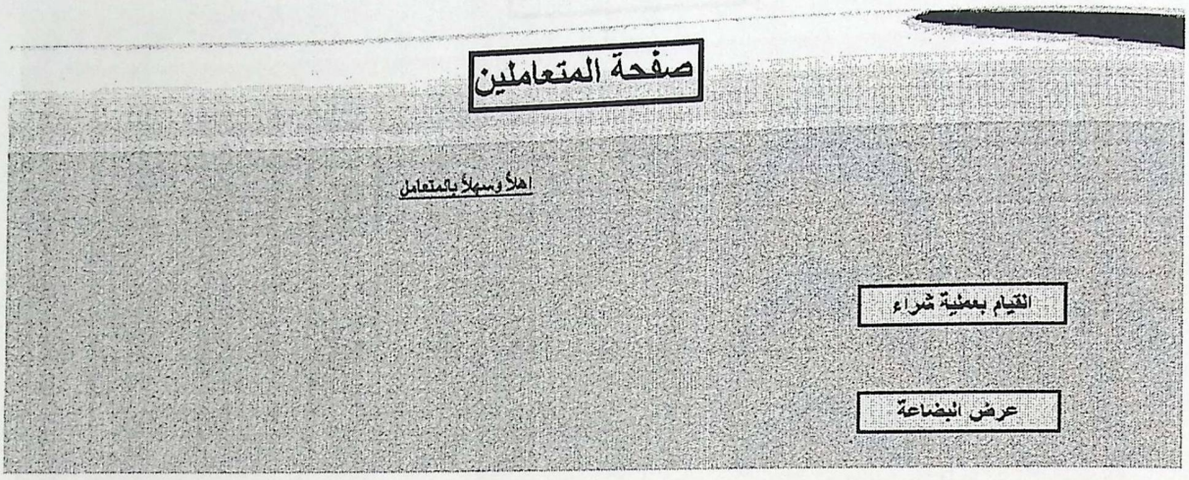


شكل (4.23) :واجهة فاتورة تصدير

أما الشكل التالي يبين الخيار لعملية الشراء من المتعامل و هو عملية إضافة فاتورة شراء:

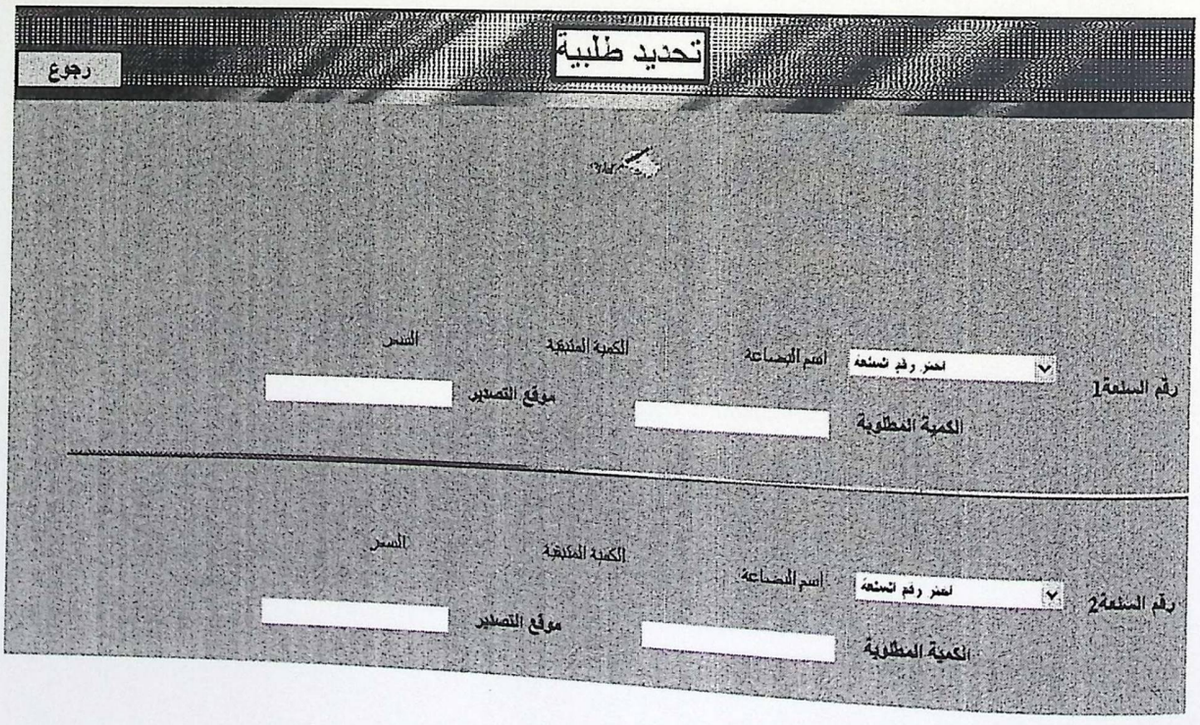
شكل (4.24) : واجهة فاتورة شراء

الشكل الثاني يبين واجهة خيارات المتعامل المختلفة كما هو مبين في الشكل التالي:



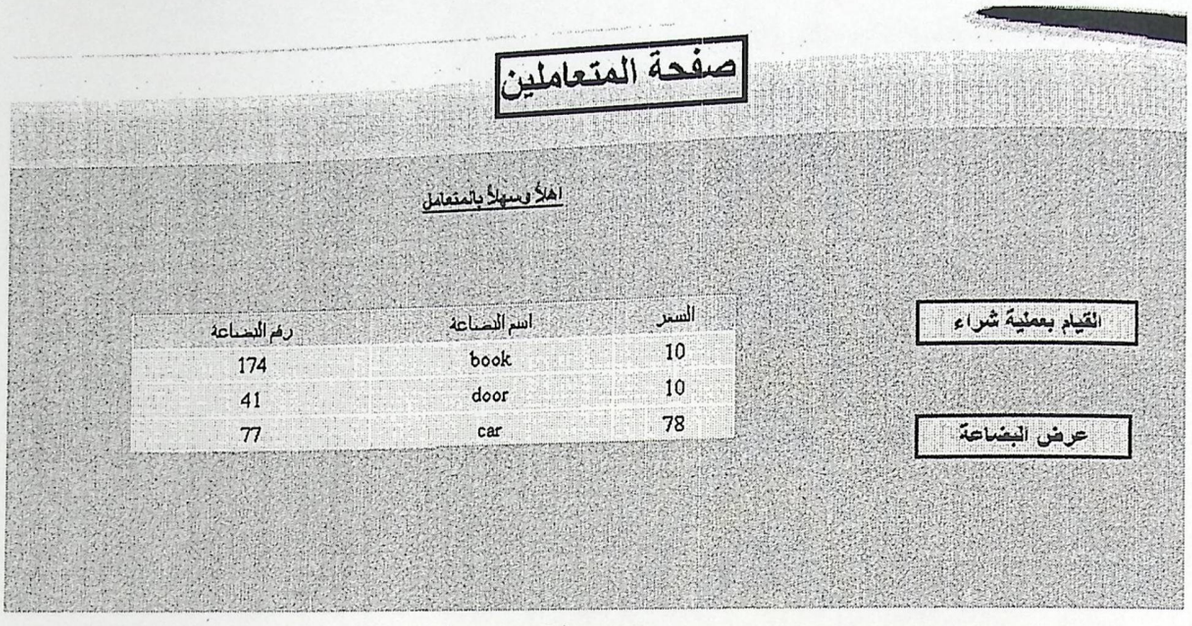
شكل (4.25) :واجهة صفحة المتعاملين

أما الشكل التالي يبين الخيار لعملية الشراء وهي عملية تحديد طلبية:



شكل (4.26) :واجهة تحديد طلبية

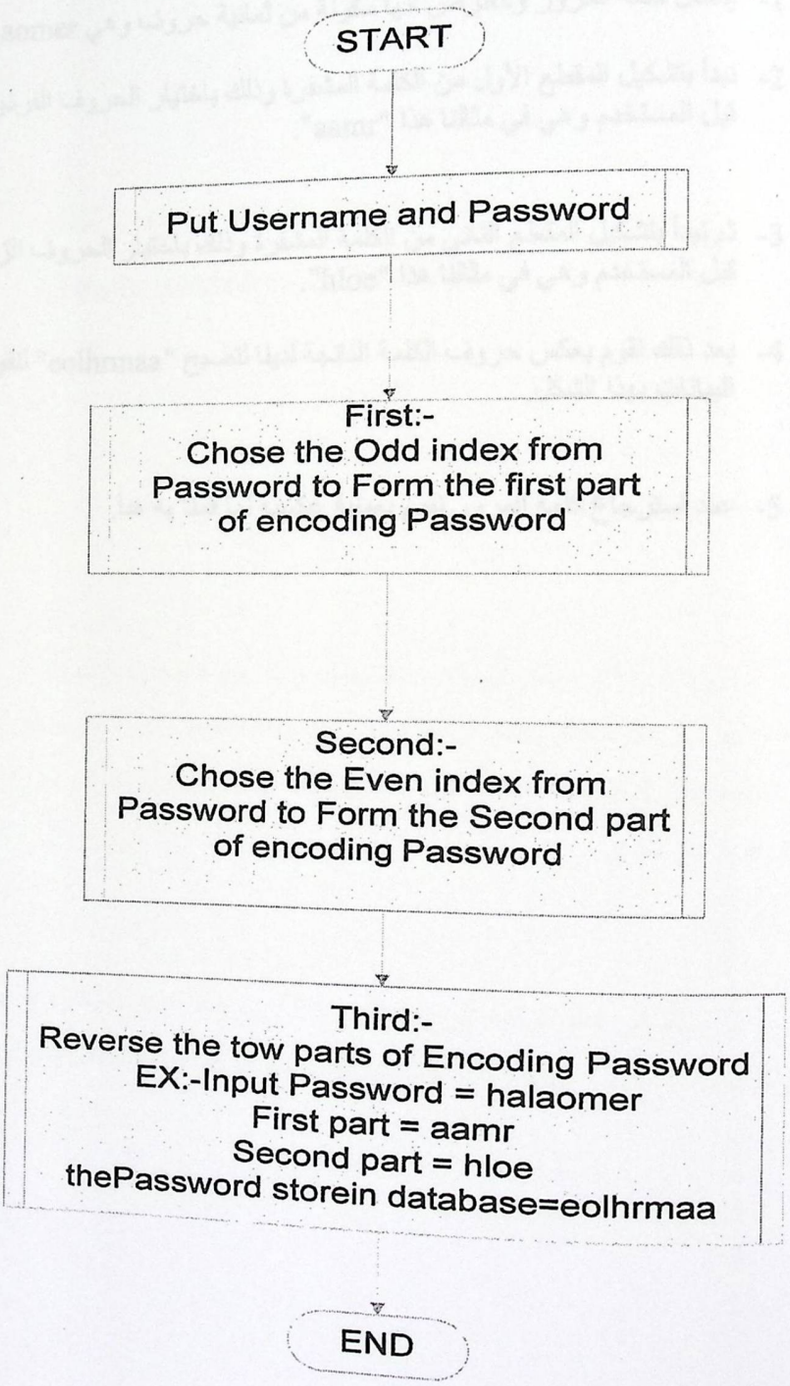
أما الشكل التالي يبين الخيار لعملية عرض منتجات الشركة وما لديها من سلع:



شكل (4.27) : واجهة عرض البضاعة

4.4 المخططات الانسيابية و الخوارزميات :

4.4.1 المخططات الانسيابية على تشفير كلمة المرور المدخلة:



شكل (4.28) المخططات الانسيابية على تشفير كلمة المرور المدخلة

نلاحظ أن هذه الخوارزمية تعمل على تشفير كلمة المرور المدخلة من قبل المستخدم وذلك من خلال الآلية التالية:

1- إدخال كلمة المرور ولنفترض أنها مكونة من ثمانية حروف وهي halaomer.

2- نبدأ بتشكيل المقطع الأول من الكلمة المشفرة وذلك باختيار الحروف الفردية من الكلمة المدخلة من قبل المستخدم وهي في مثالنا هذا "aamr".

3- ثم نبدأ بتشكيل المقطع الثاني من الكلمة المشفرة وذلك باختيار الحروف الزوجية من الكلمة المدخلة من قبل المستخدم وهي في مثالنا هذا "hloe".

4- بعد ذلك نقوم بعكس حروف الكلمة الناتجة لدينا لتصبح "eolhrmaa" لنقوم بتخزينها بداخل قاعدة البيانات بهذا الشكل.

5- عند استرجاع كلمة المرور نقوم بعملية عكسية لما قمنا به هنا.

الفصل الخـــــــامس

البناء التطبيقي للنظام و فحص النظام
(Implementation & Testing)

5.1 مقدمة (Introduction):

أن هذا النظام معتمد في بناءه على بناء و تصميم صفحات الانترنت و التي ترتبط مع قواعد البيانات الخاصة بها ، و هذا النظام يعتمد في بنائه على مجموعة التقنيات التي توفرها شركة Microsoft مثل ال MS SQL Server 2000 لبناء قواعد البيانات ، و كذلك ال VS.Net 2003 و لم يكن هناك حاجة الى مواد برمجية اخرى لبناء هذا النظام .

و سيتم في هذه الوحدة شرح عمليات بناء و وصف وحدات البرمجة و الكود الذي تم استخدامها في عمليات بناء هذا لنظام و كيفية العمل على تحميلها و تحضيرها للشكل النهائي للعمل.

5.2 متطلبات النظام من أنظمة التشغيل و الاجهزة (Hardware & O.S):

إن الاجهزة المطلوب توافرها من أجل ان يعمل النظام بالشكل الصحيح هي كما يلي :

- مجموعة من أجهزة ألـ Pentium IV و ذلك لمستخدمي النظام.
- الأدوات الضرورية لعمل أي شبكة مثل الاسلاك.

أما بالنسبة لانظمة التشغيل التي يجب ان تتوافر على اجهزة كل من االمستخدمين فإن هذا النظام يتطلب وجود نظام التشغيل Windows 2003 Server أو Windows XP .

5.3 أمثلة على بناء اجزاء النظام برمجياً (Implementation Examples):

في هذا القسم سنتناول الحديث عن بعض السطور البرمجية (الكود) التي تمت كتابتها من أجل برمجة هذا النظام .

المثال الاول: يقوم البرنامج المبين أدناه بفحص التاريخ بوضعه بالشكل الصحيح ويمر بعددت عمليات فحص ويفحص تاريخ الإنتاج والانتهاء لمنتج ما حيث إذا تم وضع تاريخ الانتهاء اقل من تاريخ الانتاج فان النظام لا يقبل ذالك :

```
Public Class timetest

    Public Sub New()
    End Sub
```

```
    Public Function testtime(ByRef month As String, ByRef day As
String, ByRef year As String) As Integer
        If year.Trim.Length = 3 Then
            year = "0" & year
        ElseIf year.Trim.Length = 2 Then
            year = "0" & "0" & year
        ElseIf year.Trim.Length = 1 Then
            year = "0" & "0" & "0" & year
        End If
        If month.Trim.Length = 1 Then
            month = "0" & month
        End If
        If day.Trim.Length = 1 Then
            day = "0" & day
        End If
        If Val(month) > 12 Or Val(month) < 1 Then
            Return 0
        End If
        If Val(day) > 31 Or Val(day) < 1 Then
            Return 1
        End If
        If Val(year) > 9999 Or Val(year) < 1994 Then
            Return 2
        End If

        If Val(month) = 2 Then
            If Val(year) Mod 4 <> 0 Then
                If Val(day) > 28 Or Val(day) < 1 Then
                    Return 3
                End If
            ElseIf Val(day) > 29 Or Val(day) < 1 Then
                Return 4
            End If
        ElseIf Val(month) = 4 Or Val(month) = 6 Or Val(month) = 9 Or
Val(month) = 11 Then
            If Val(day) > 30 Then
                Return 5
            End If
        End If

        If month.Chars(0).IsLetter(month.Chars(0)) Or
month.Chars(1).IsLetter(month.Chars(1)) Then
            Return 0
        End If


        If day.Chars(0).IsLetter(day.Chars(0)) Or
day.Chars(1).IsLetter(day.Chars(1)) Then
```

```
            Return 1
        End If
        If year.Chars(0).IsLetter(year.Chars(0)) Or
year.Chars(1).IsLetter(year.Chars(1)) Or
year.Chars(2).IsLetter(year.Chars(2)) Or
year.Chars(3).IsLetter(year.Chars(3)) Then
            Return 2
        End If


        Return 100
    End Function


End Class



If smonth <> "" AndAlso sday <> "" AndAlso syear <> "" Then
            test0 = classtime.testtime(smonth, sday, syear)
            date1 = syear + smonth + sday
        Else
            date1 = ""
        End If
        If emonth <> "" AndAlso eday <> "" AndAlso eyear <> "" Then
            test1 = classtime.testtime(emonth, eday, eyear)
            date2 = eyear + emonth + eday
        Else
            date2 = ""
        End If
        If test0 = 0 Then
            'Me.Label18.Text = "خطأ في تحديد الشهر ضمن تاريخ الانتاج "

            'Me.Label24.Visible = True
            test = False
        ElseIf test0 = 1 Then
            'Me.Label18.Text = "خطأ في تحديد اليوم ضمن تاريخ الانتاج "
            'Me.Label24.Visible = True
            test = False
        ElseIf test0 = 2 Then
            'Me.Label18.Text = "خطأ في تحديد السنة ضمن تاريخ الانتاج "
            'Me.Label24.Visible = True
            test = False
        ElseIf test0 = 3 Then
            'Me.Label18.Text = "خطأ لان شهر 2 في السنة غير الكبيسة أو أكبر
من 28 يساوي أو أقل يوم يساوي "
            'Me.Label24.Visible = True
            test = False
        ElseIf test0 = 4 Then
```

```
أو أكبر الكبيسة السنة في 2 شهر لان خطأ" = 'Me.Label18.Text
            " يوم 29 يساوي أو من وأقل يوم يساوي
            'Me.Label24.Visible = True
            test = False
        ElseIf test0 = 5 Then
            'Me.Label18.Text = "وأقل يوم يساوي أو أكبر الشهر هذا لان خطأ
" يوم 30 يساوي أو
            'Me.Label24.Visible = True
            test = False
        ElseIf test0 = 100 Then
            test = True
            'Me.Label24.Visible = False
            'Me.Label18.Text = ""
        End If
        If test1 = 0 Then
            'Me.Label23.Text = "الانتهاء تاريخ ضمن الشهر تحديد في خطأ"
            'Me.Label25.Visible = True
            test = False
        ElseIf test1 = 1 Then
            'Me.Label23.Text = "الانتهاء تاريخ ضمن اليوم تحديد في خطأ"
            'Me.Label25.Visible = True
            test = False
        ElseIf test1 = 2 Then
            'Me.Label23.Text = "الانتهاء تاريخ ضمن السنة تحديد في خطأ"
            'Me.Label25.Visible = True
            test = False
        ElseIf test1 = 3 Then
            'Me.Label23.Text = "أو أكبر الكبيسة غير السنة في 2 شهر لان خطأ
"يوم 28 يساوي أو من وأقل يوم يساوي
            'Me.Label25.Visible = True
            test = False
        ElseIf test1 = 4 Then
            'Me.Label23.Text = "أو أكبر الكبيسة السنة في 2 شهر لان خطأ
"يوم 29 يساوي أو من وأقل يوم يساوي
            'Me.Label25.Visible = True
            test = False
        ElseIf test1 = 5 Then
            'Me.Label23.Text = "وأقل يوم يساوي أو أكبر الشهر هذا لان خطأ
"يوم 30 يساوي أو
            'Me.Label25.Visible = True
            test = False
        ElseIf test1 = 100 Then
            'Me.Label25.Visible = False
            test = True
            'Me.Label23.Text = ""
        End If
```

```
        End If
    End Sub


Dim val1 As Boolean = False
        Me.Button7.Visible = True
        Dim test1 As Integer
        Dim test0 As Integer
        Dim test As Boolean
        Dim classtime As New timetest
        Dim date1 As String = "1"
        Dim date2 As String = "1"
        Dim smonth As String = Me.Textbox10.Text.Trim
        Dim sday As String = Me.Textbox9.Text.Trim
        Dim syear As String = Me.Textbox5.Text.Trim
        Dim emonth As String = Me.Textbox11.Text.Trim
        Dim eday As String = Me.Textbox12.Text.Trim
        Dim eyear As String = Me.Textbox13.Text.Trim

        If smonth <> "" AndAlso sday <> "" AndAlso syear <> "" Then
            test0 = classtime.testtime(smonth, sday, syear)
            date1 = syear + smonth + sday
        Else
            date1 = ""
        End If
        If emonth <> "" AndAlso eday <> "" AndAlso eyear <> "" Then
            test1 = classtime.testtime(emonth, eday, eyear)
            date2 = eyear + emonth + eday
        Else
            date2 = ""
        End If
        If test0 = 0 Then
            'Me.Label18.Text = "خطأ في تحديد الشهر ضمن تاريخ الانتاج "

            'Me.Label24.Visible = True
            test = False
        ElseIf test0 = 1 Then
            'Me.Label18.Text = "خطأ في تحديد اليوم ضمن تاريخ الانتاج "
            'Me.Label24.Visible = True
            test = False
        ElseIf test0 = 2 Then
            'Me.Label18.Text = "خطأ في تحديد السنة ضمن تاريخ الانتاج "
            'Me.Label24.Visible = True
            test = False
        ElseIf test0 = 3 Then
            'Me.Label18.Text = "خطأ لان 2 شهر في السنة غير الكبيسة أو أكبر من وأقل يوم يساوي
28 يساوي أو من وأقل يوم يساوي "
            'Me.Label24.Visible = True
```

```
            test = False
        ElseIf test0 = 4 Then
            'Me.Label18.Text = "أو أكبر الكبيسة السنة في 2 شهر لان خطأ
" يوم 29 يساوي أو من وأقل يوم يساوي
            'Me.Label24.Visible = True
            test = False
        ElseIf test0 = 5 Then
            'Me.Label18.Text = "وأقل يوم يساوي أو أكبر الشهر هذا لان خطأ
" يوم 30 يساوي أو
            'Me.Label24.Visible = True
            test = False
        ElseIf test0 = 100 Then
            test = True
            'Me.Label24.Visible = False
            'Me.Label18.Text = ""
        End If
        If test1 = 0 Then
            'Me.Label23.Text = "الانتهاء تاريخ ضمن الشهر تحديد في خطأ"
            'Me.Label25.Visible = True
            test = False
        ElseIf test1 = 1 Then
            'Me.Label23.Text = "الانتهاء تاريخ ضمن اليوم تحديد في خطأ"
            'Me.Label25.Visible = True
            test = False
        ElseIf test1 = 2 Then
            'Me.Label23.Text = "الانتهاء تاريخ ضمن السنة تحديد في خطأ"
            'Me.Label25.Visible = True
            test = False
        ElseIf test1 = 3 Then
            'Me.Label23.Text = "أو أكبر الكبيسة غير السنة في 2 شهر لان خطأ
"يوم 28 يساوي أو من وأقل يوم يساوي
            'Me.Label25.Visible = True
            test = False
        ElseIf test1 = 4 Then
            'Me.Label23.Text = "أو أكبر الكبيسة السنة في 2 شهر لان خطأ
"يوم 29 يساوي أو من وأقل يوم يساوي
            'Me.Label25.Visible = True
            test = False
        ElseIf test1 = 5 Then
            'Me.Label23.Text = "وأقل يوم يساوي أو أكبر الشهر هذا لان خطأ
"يوم 30 يساوي أو
            'Me.Label25.Visible = True
            test = False
        ElseIf test1 = 100 Then
            'Me.Label25.Visible = False
            test = True
            'Me.Label23.Text = ""
        End If
```

```
If test = True Then

    val1 = True


End If
```

المثال الثاني: يقوم البرنامج المبين أدناه بالبحث عن أية بضاعة :

```
If date1 = "" AndAlso date2 = "" Then
    val1 = True
End If
Dim str2 As String
Dim str1 As String = " where "

If Me.TextBox1.Text <> "" Then
    str1 = str1 + "gname="
    str1 = str1 + "'" & Me.TextBox1.Text & "' And "
End If
If Me.TextBox2.Text <> "" Then
    str1 = str1 + "gamount="
    str1 = str1 + "" & Me.TextBox2.Text & " And "
End If
If Me.TextBox4.Text <> "" Then
    str1 = str1 + "gpric="
    str1 = str1 + "'" & Me.TextBox4.Text & "' And "
End If
If Me.TextBox3.Text <> "" Then
    str1 = str1 + "gunit="
    str1 = str1 + "'" & Me.TextBox3.Text & "' And "
End If
If Me.TextBox6.Text <> "" Then
    str1 = str1 + "cno="
    str1 = str1 + "" & Val(Me.TextBox6.Text) & " And "
End If

If Me.TextBox8.Text <> "" Then
    str1 = str1 + "gno="
    str1 = str1 + "'" & Me.TextBox8.Text & "' And "
End If
str1 = str1.Substring(0, str1.Length - 5)

If val1 = True Then
    Try
        Me.DataGrid1.Visible = True
        sql = "select * from goods" + str1
        con.Open()
        ad.SelectCommand.CommandText = sql
        ad.SelectCommand.ExecuteNonQuery()
        'Me.DataSet1.Clear()
```

```
            Me.ad.Fill(Me.Dsa1)
            '  Me.Label1.Text = Me.DataSet1.Tables(0).Rows(0)(0)
            Me.DataGrid1.DataBind()
            Me.Label23.Text = "بنجاح البحث عملية تمت"
            Me.Label23.Text = sql
        Catch ex As Exception
            Me.Label4.Text = ex.Message + "    " + sql
            DataSet1.Clear()
            val1 = False

        End Try
    End If
```

المثال الثالث : يقوم البرنامج المبين أدناه باضافة بيانات لمنتج جديد :

```
If date1 <> "" AndAlso date2 <> "" AndAlso test = True AndAlso
Val(date1) < Val(date2) Then

            val1 = True


        End If
        Dim str1 As String = "("
        Dim str2 As String = "values("

        If Me.TextBox1.Text <> "" Then
            str1 = str1 + "gname,"
            str2 = str2 + "'" & Me.TextBox1.Text & " ',"
        End If
        If Me.TextBox2.Text <> "" Then
            str1 = str1 + "gamount,"
            str2 = str2 + "" & Val(Me.TextBox2.Text) & ","
        End If
        If Me.TextBox4.Text <> "" Then
            str1 = str1 + "gpric,"
            str2 = str2 + "'" & Me.TextBox4.Text & " ',"
        End If
        If Me.TextBox3.Text <> "" Then
            str1 = str1 + "gunit,"
            str2 = str2 + "'" & Me.TextBox3.Text & " ',"
        End If
        If Me.DropDownList1.SelectedIndex <> 0 Then
            str1 = str1 + "cno,"
```

```
            str2 = str2 + "" & Me.DropDownList1.SelectedValue & ","
        End If
        str1 = str1 + "gno)"
        str2 = str2 + "'" & Me.TextBox8.Text & " ')"
        If Me.TextBox8.Text = "" Then
            Me.Label4.Text = "السلعة رقم ادخال يجب"
        Else
            Me.Label4.Text = ""
            If val1 = True Then
                Try

                    sql = "insert into goods" + str1 + str2
                    con.Open()
                    ad.SelectCommand.CommandText = sql
                    ad.SelectCommand.ExecuteNonQuery()
                Catch ex As Exception
                    Me.Label4.Text = "الاسماء بنفس سابقة مدخلات يوجد "
                    DataSet1.Clear()
                    val1 = False

                End Try
                If val1 = True Then

                Try
                    date1 = date1.Substring(0, 4) + "-" +
date1.Substring(4, 2) + "-" + date1.Substring(6, 2)
                    date2 = date2.Substring(0, 4) + "-" +
date2.Substring(4, 2) + "-" + date2.Substring(6, 2)
                    sql = "insert into goods_date (
productdate,expiredate,gno)  values ('" & date1 & "','" & date2 & "','"
& Me.TextBox8.Text & "')"
                    'Me.Label4.Text = sql
                    ad2.SelectCommand.CommandText = sql
                    ad2.SelectCommand.ExecuteNonQuery()
                    con.Close()
                        Me.Label4.Text = "بنجاح الاضافة عملية تمت...."


                Catch ex As Exception
                        Me.Label23.Text = ex.Message
                    End Try
                End If
            Else
                Me.Label4.Text = "التواريخ ادخال يجب  [الانتاج تاريخ
صحيحة بصورة  [الانتهاء وتاريخ"
            End If
```

المثال الرابع: يقوم البرنامج المبين أدناه بتغيير كلمة المرور لمستخدم ما:

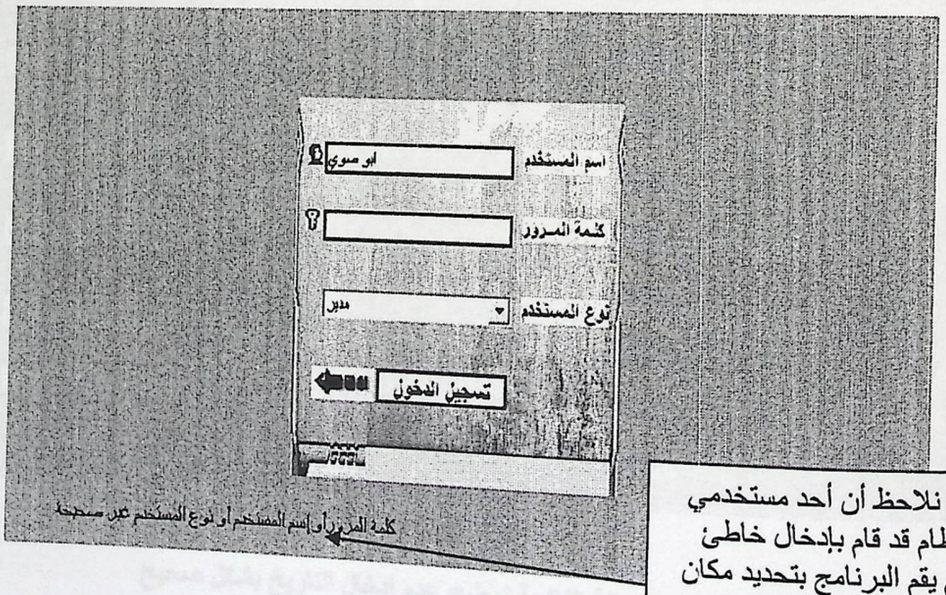
```
Dim str As String
        str = "select upassword from usertype where uname=" &
Val(Me.Textbox4.Text) & ""
        Dim row1 As DataRow
        Try
            Me.SqlConnection1.Open()
            Me.SqlDataAdapter1.SelectCommand.CommandText = str
            Me.SqlDataAdapter1.Fill(Me.DataSet11, "changpass")
            Me.SqlConnection1.Close()
        Catch ex As SqlException
            Me.Label1.Text = ex.Message
        End Try
        Me.Label6.Text = Me.DataSet11.Tables("changPass").Rows.Count
        If Me.DataSet11.Tables("changPass").Rows.Count <> 0 Then
            For Each row1 In Me.DataSet11.Tables("changPass").Rows
                If Me.TextBox1.Text = row1(0) AndAlso Me.Textbox2.Text
= Me.Textbox3.Text Then
                    Try
                        Me.SqlConnection1.Open()
                        Me.SqlDataAdapter1.SelectCommand.CommandText =
"update usertype set upassword=" & Me.Textbox2.Text & " where uname='"
& Val(Me.Textbox4.Text) & "'"
                        Me.SqlDataAdapter1.Fill(Me.DataSet11,
"changpass")
                        Me.SqlConnection1.Close()
                        Me.Label6.Text += "تم تغييركلمة المرور بنجاح..."
                    Catch ex As SqlException
                        Me.Label6.Text += ex.Message
                    End Try
                Else
                    Me.Label6.Text += "اسم المستخدم أو كلمة المرور غير صحيحة"
                End If

            Next
        Else
            Me.Label6.Text += "اسم المستخدم أو كلمة المرور غير صحيحة"
        End If
```

5.4 اختبار آلية عمل النظام (Testing):

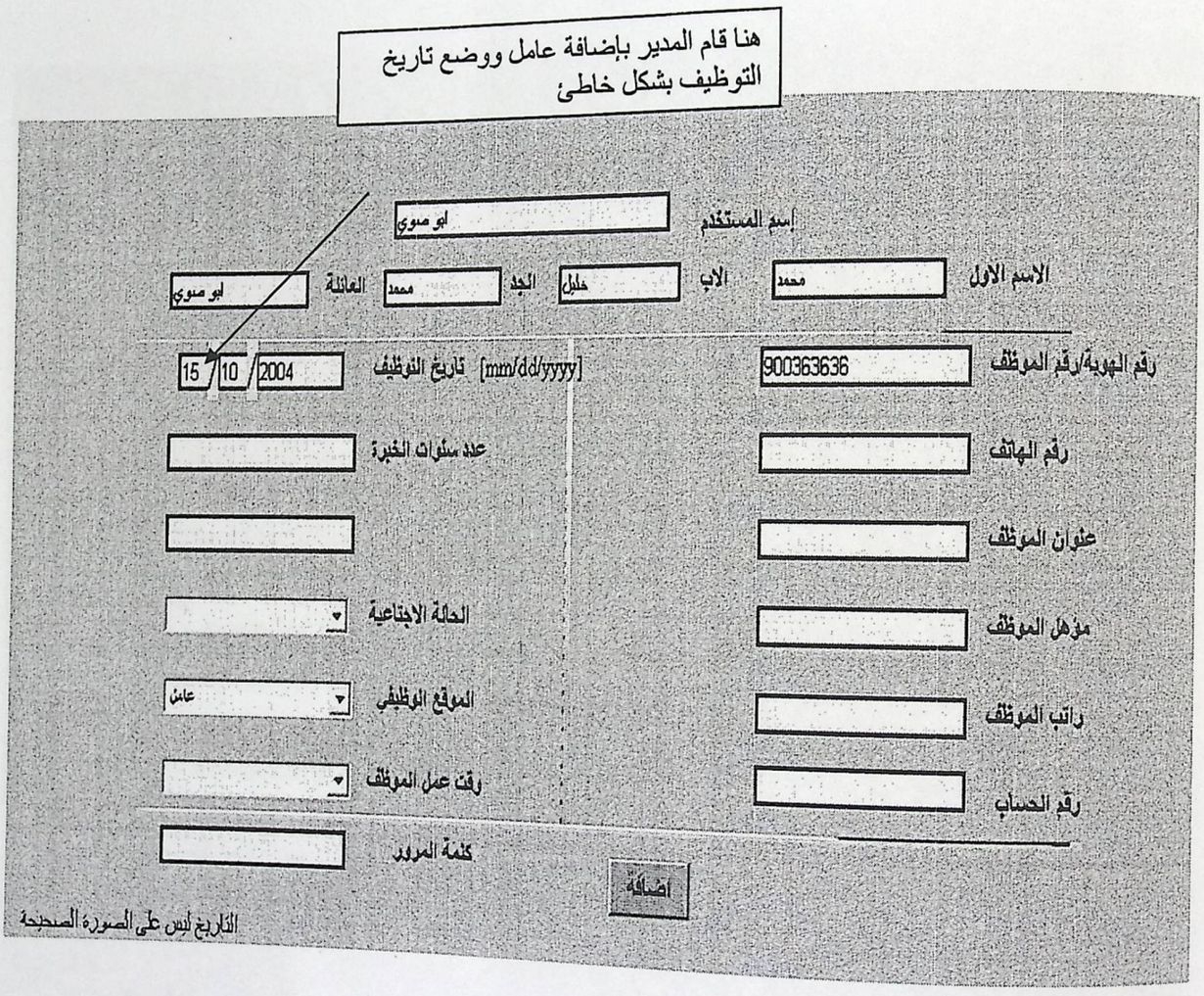
في هذا القسم سيتم شرح عمليات فحص النظام و الاخطاء التي يمكن ان تحصل عند عملية ادخال البيانات في الأقسام المختلفة للنظام ، و سيتم توضيح الرسائل التي تظهر نتيجة حصول مثل هذه الاخطاء و توضح سبب الخطأ و كيفية العمل على اصلاحه ، و سيتم توضيح ذلك حسب الأشكال التالية التي تبين حصول بعض الأخطاء عند بعض أقسام النظام مع الرسالة الناتجة عن ذلك الخطأ ، كما يلي:

- الشكل التالي يوضح الرسالة الناتجة من عدم عملية ادخال كلمة المرور او اسم المستخدم بشكل صحيح لذلك على صفحة الدخول الى النظام، و تظهر الرسالة حتى تلفت انتباه المستخدم لها ، حيث ان الفائدة من ذلك هو اعلام المستخدم عن الخطأ الذي وقع من عدم ادخاله لكلمة المرور او اسم المستخدم بشكل صحيح فلم يستطع الدخول الى النظام ، فكان عليه كتابة كلمة المرور او اسم المستخدم الخاصة به بشكل صحيح لاكمال عملية الدخول:



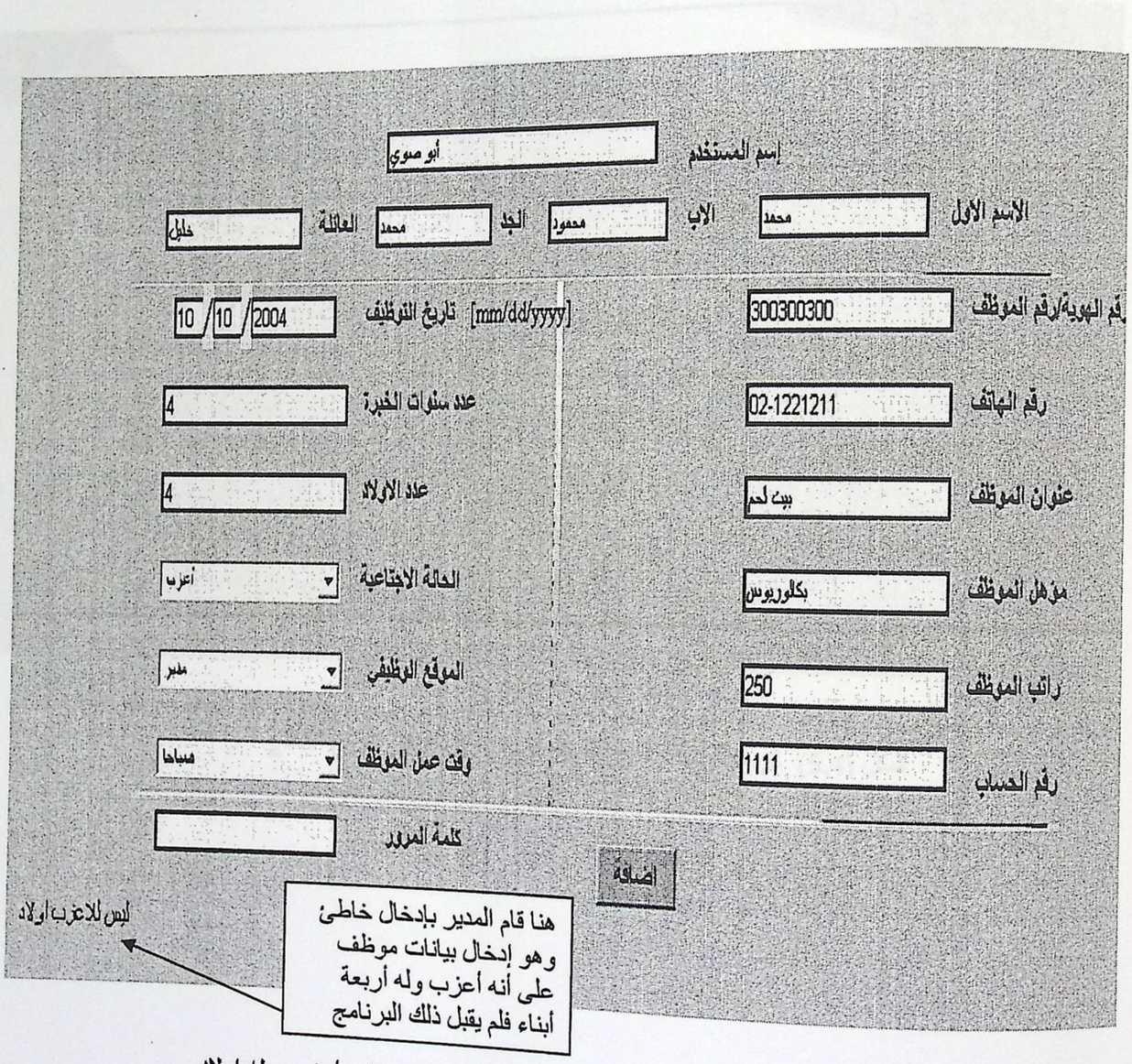
شكل (5.1) الرسالة الناتجة من عدم ادخال كلمة المرور او اسم المستخدم على صفحة الدخول الى النظام

اما الشكل التالي فهو يمثل اضافة بيانات موظف ، حيث ان الرسالة الظاهرة توضح عدم وضع التاريخ بشكل صحيح ، فكان من الضروري تنبيه المستخدم بذلك حتى تستكمل اجراءات الحجز:



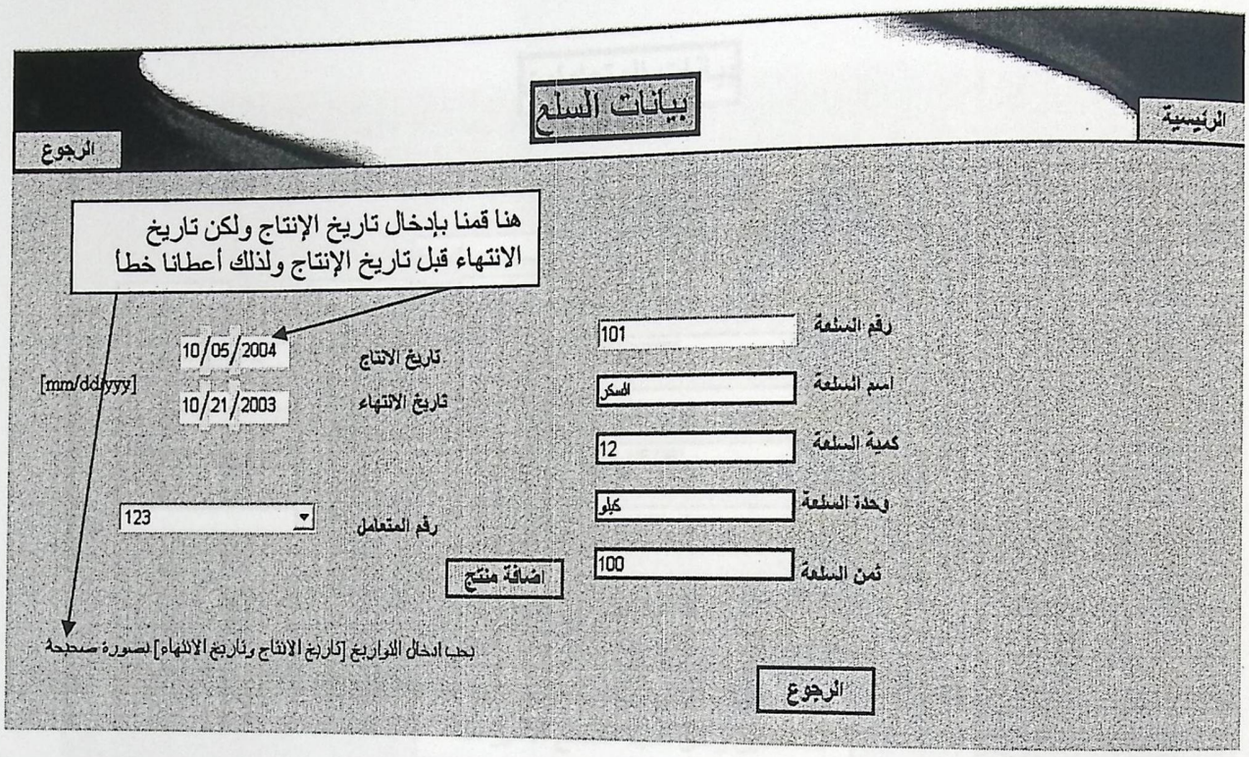
شكل (5.2) الرسالة الناتجة توضح عدم إدخال التاريخ بشكل صحيح

اما الشكل التالي فهو يمثل اضافة بيانات موظف،حيث ان الرسالة الظاهرة توضح اختيار الحالة الاجتماعية للموظف اعزب وله اولاد فتنبه المستخدم بذلك:



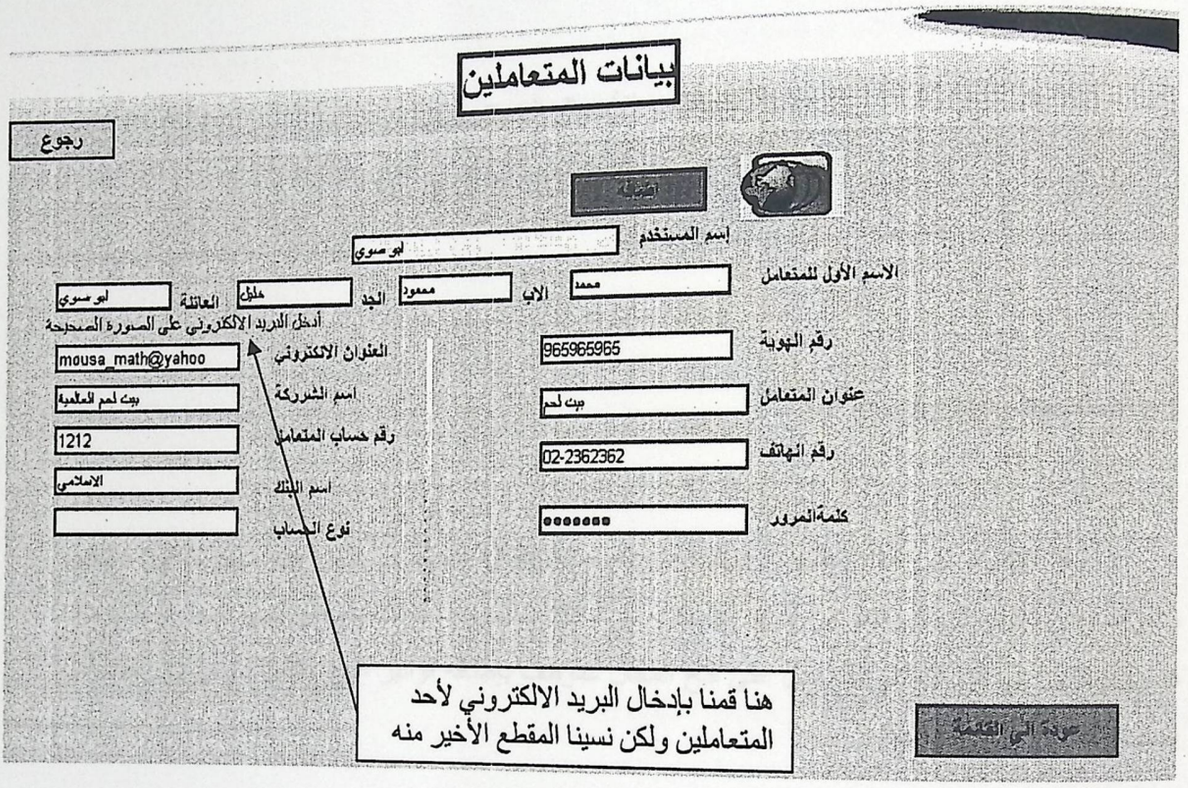
شكل (5.3) الرسالة الناتجة توضح اختيار الحالة الاجتماعية للموظف أعزب وله اولاد

اما الشكل التالي فهو يمثل اضافة بيانات للسلع ، حيث ان الرسالة الظاهرة توضح إدخال تاريخ الإنتاج ولكن تاريخ الانتهاء قبل تاريخ الإنتاج ولذلك أعطانا خطأ.



شكل (5.4) الرسالة الناتجة عن ادخال تاريخ الانتهاء قبل تاريخ الانتاج

أما الشكل التالي فهو يمثل إضافة بيانات متعامل ، حيث ان الرسالة الظاهرة توضح عدم ادخال البريد الالكتروني على الصورة الصحيحة.



شكل (5.5) الرسالة الناتجة عن إدخال البريد الالكتروني بشكل غير صحيح

الفصل الســـــــــادس

الاستنتاجات و الأعمال المستــقبلية
(Conclusions & Future Works)

6.1 الاستنتاجات(Conclusions):

بعد الانتهاء من هذا النظام وملاحظة طريقة عمله وفعالية نتائجه نستخلص منه مجموعة من النتائج والفوائد وهي تظهر في كل قسم من أقسامه المختلفة سواء أكانت للموظف أو للمتعامل أو للمدير حيث نستنتج من قسم الموظفين في هذا النظام أنه يعمل على فتح المجال للموظف بإضافة فواتير البيع والشراء وعرض للطلبيات ومعرفة وضعه داخل الشركة .

أما بالنسبة لقسم المتعاملين فإنه بالإمكان تحديد الطلبية من نفس الموقع الذي يكون جزءا منه وذالك عن طريق شبكة الانترنت وعرض جميع تعاملاته مع الشركة .

أما بالنسبة لقسم المدير فانه بالإمكان إضافة منتج والتعديل عليه وإضافة بيانات الموظفين والمتعاملين وإمكانية التعديل عليها ، وإمكانية إضافة مستخدم جديد للدخول الى صفحة الشركة للقيام بالعمليات التي حددت له.

لذلك فإن هذه الآليات ساعدت على حل هذة المشاكل بشكل كبير مثل فتح الامكانية امام المتعامل بالتعامل مع الشركة عن بعد فلا يتطلب منه الوصول الى موقع الشركة لتحديد طلبيته.

و من الاستنتاجات الاخرى للمشروع انها عادت على القائم على المشروع بجملة من الفوائد ، التي ادت الى اكتسابه للعديد من الخبرات في مجال البرمجة وتصميم الأنظمة المعقدة فمن هذا النظام كانت الفائدة لتعلم مجموعة من بيئات البرمجة ولغاتها ومنها ASP.NET وكذلك MS SQL Server 2000 وأيضاً كان هناك فوائد أخرى مثل تعلم خدمات الانترنت مثل IIS و أيضاً كانت الفائدة بتعلم استخدام أدوات الأمان في هذا النظام وغيرها من الامور المفيدة.

واكتسبت أيضاً الخبرة العملية بشكل كبير وكيفية العمل تحت الضغط الكبير و بالتالي أصبح لدي القدرة على مواجهة مشاريع.

6.2 الأعمال التطويرية والمستقبلية (Future Work) :

يمكن عمل تطوير لهذا النظام بحيث يصبح شامل قسم للمحاسبة يحتوي على جميع أنواع الحسابات الخاصة بشركة ما .

ويمكن تطوير هذا المشروع بربط قاعدة بيانات المشروع مع قاعدة بيانات البنك من اجل استخدام بطاقات الصرف الآلي.

وأيضا يمكن ربط قاعدة بيانات هذا المشروع بالبورصة وذلك من اجل ابقاء قاعدة البيانات متصلة مع آخر التحديثات لأسعار العملات .

المراجـــــع
(References)

John ، Ben SchoellmannS، Kevin McNeish ، Tony Loton ،1. Andrew Filev Published by Wrox ،**Chaur G.Wu .** Professional UML with visual studio.Net ،**Slater** ، b27 6BH، Brimingham، Acocks Green،1102 Warwicked Rosd ،Arden hous ،Press Ltd United Kingdom.

S.Sudarshan. DataBase system Concepts ، **Henry F.Korth،2. Abraham Silberschatz** Beijing China.، Higher Educatio Press ، McGrow-Hill Companies، Fourth Edition ،

، Scott ،Design Ann implementation،**3. Alan L. E.** System Developments ;Analysis USA 1990.،Foresman And company

، **Strachan**. A data Base System ;a practical approach design ، **Begg C،4. Connolly T** 1997.،implementation and management

Jane P. Laudon . management information system and ،**5. Kenneth C.Laudon** ، inc ، prentice Hall ، sixth edition ،organization Technology in the network enterprise higher education press.،person education company

6. Ministry of Education and higher education . statistics in general education in November 2003.، General directorate of educational planning ،palestine 2003-2004

، Macmillan Publishing company ، **7. Raymond M. Jr.** information System conceots 1994.،New york

R. S. press man and associate company . Software ،**8. Roger S. Pressman** Fourth edition McGrow-Hill Companies.،Engineering- practitioner's Approach

Addison-Wesley Publishing ، fifth edition ،**9. Ian Sommerville** . Software engineering 1995.،British Library Cataloguing-in-Publishing Data ،Company

10. http://www.msdn.microsoft.com/vbasic/Default.aspx?pull=/library/enus/dnvs05/html/smartclient_vb2005.asp

11. http://www. programmersheaven.com/zone1/mh49.htm

12 .http://www.palestine-net.com/education/

13.http://www.moe.gov.ps

14. http :// www.pnic.gov.ps/arabic/edu/edua. html

15. http : // www.e.gov.qa /eGovportal/index.jsp?lang=ar

الملحقات
Appendix

```
1.  if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_acoun_account]') and OBJECTPROPERTY(id,
    N'IsForeignKey') = 1)
2.  ALTER TABLE [dbo].[acoun] DROP CONSTRAINT FK_acoun_account
3.  GO
4.
5.  if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_customers_account]') and OBJECTPROPERTY(id,
    N'IsForeignKey') = 1)
6.  ALTER TABLE [dbo].[customers] DROP CONSTRAINT
    FK_customers_account
7.  GO
8.
9.  if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_accounts_bankaccount]') and OBJECTPROPERTY(id,
    N'IsForeignKey') = 1)
10. ALTER TABLE [dbo].[accounts] DROP CONSTRAINT
    FK_accounts_bankaccount
11. GO
12.
13. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_baying_invoice_customers]') and
    OBJECTPROPERTY(id, N'IsForeignKey') = 1)
14. ALTER TABLE [dbo].[baying_invoice] DROP CONSTRAINT
    FK_baying_invoice_customers
15. GO
16.
17. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_goods_customers]') and OBJECTPROPERTY(id,
    N'IsForeignKey') = 1)
18. ALTER TABLE [dbo].[goods] DROP CONSTRAINT FK_goods_customers
19. GO
20.
21. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_order_customers]') and OBJECTPROPERTY(id,
    N'IsForeignKey') = 1)
22. ALTER TABLE [dbo].[orde] DROP CONSTRAINT FK_order_customers
23. GO
24.
```

```
25. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_usertype_customers]') and OBJECTPROPERTY(id,
    N'IsForeignKey') = 1)
26. ALTER TABLE [dbo].[usertype] DROP CONSTRAINT FK_usertype_customers
27. GO
28.
29. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_usertype_employees1]') and OBJECTPROPERTY(id,
    N'IsForeignKey') = 1)
30. ALTER TABLE [dbo].[usertype] DROP CONSTRAINT
    FK_usertype_employees1
31. GO
32.
33. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_exp_invoice_details_expinvoice]') and
    OBJECTPROPERTY(id, N'IsForeignKey') = 1)
34. ALTER TABLE [dbo].[exp_invoice_details] DROP CONSTRAINT
    FK_exp_invoice_details_expinvoice
35. GO
36.
37. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_accounts_fixedaccount]') and OBJECTPROPERTY(id,
    N'IsForeignKey') = 1)
38. ALTER TABLE [dbo].[accounts] DROP CONSTRAINT
    FK_accounts_fixedaccount
39. GO
40.
41. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_accounts_fixedcare]') and OBJECTPROPERTY(id,
    N'IsForeignKey') = 1)
42. ALTER TABLE [dbo].[accounts] DROP CONSTRAINT FK_accounts_fixedcare
43. GO
44.
45. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_exp_invoice_details_goods]') and
    OBJECTPROPERTY(id, N'IsForeignKey') = 1)
46. ALTER TABLE [dbo].[exp_invoice_details] DROP CONSTRAINT
    FK_exp_invoice_details_goods
47. GO
48.
49. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_goods_date_goods]') and OBJECTPROPERTY(id,
    N'IsForeignKey') = 1)
50. ALTER TABLE [dbo].[goods_date] DROP CONSTRAINT
    FK_goods_date_goods
51. GO
```

```
52.
53. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[FK_accounts_mangeacc]') and OBJECTPROPERTY(id,
    N'IsForeignKey') = 1)
54. ALTER TABLE [dbo].[accounts] DROP CONSTRAINT
    FK_accounts_mangeacc
55. GO
56.
57. if exists (select * from dbo.sysobjects where id = object_id(N'[dbo].[account]')
    and OBJECTPROPERTY(id, N'IsUserTable') = 1)
58. drop table [dbo].[account]
59. GO
60.
61. if exists (select * from dbo.sysobjects where id = object_id(N'[dbo].[accounts]')
    and OBJECTPROPERTY(id, N'IsUserTable') = 1)
62. drop table [dbo].[accounts]
63. GO
64.
65. if exists (select * from dbo.sysobjects where id = object_id(N'[dbo].[acoun]') and
    OBJECTPROPERTY(id, N'IsUserTable') = 1)
66. drop table [dbo].[acoun]
67. GO
68.
69. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[bankaccount]') and OBJECTPROPERTY(id, N'IsUserTable')
    = 1)
70. drop table [dbo].[bankaccount]
71. GO
72.
73. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[baying_invoice]') and OBJECTPROPERTY(id,
    N'IsUserTable') = 1)
74. drop table [dbo].[baying_invoice]
75. GO
76.
77. if exists (select * from dbo.sysobjects where id = object_id(N'[dbo].[customers]')
    and OBJECTPROPERTY(id, N'IsUserTable') = 1)
78. drop table [dbo].[customers]
79. GO
80.
81. if exists (select * from dbo.sysobjects where id = object_id(N'[dbo].[employees]')
    and OBJECTPROPERTY(id, N'IsUserTable') = 1)
82. drop table [dbo].[employees]
83. GO
84.
```

```
85. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[exp_invoice_details]') and OBJECTPROPERTY(id,
    N'IsUserTable') = 1)
86. drop table [dbo].[exp_invoice_details]
87. GO
88.
89. if exists (select * from dbo.sysobjects where id = object_id(N'[dbo].[expinvoice]')
    and OBJECTPROPERTY(id, N'IsUserTable') = 1)
90. drop table [dbo].[expinvoice]
91. GO
92.
93. if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[fixedaccount]') and OBJECTPROPERTY(id, N'IsUserTable')
    = 1)
94. drop table [dbo].[fixedaccount]
95. GO
96.
97. if exists (select * from dbo.sysobjects where id = object_id(N'[dbo].[fixedcare]')
    and OBJECTPROPERTY(id, N'IsUserTable') = 1)
98. drop table [dbo].[fixedcare]
99. GO
100.
101.        if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[fixedother]') and OBJECTPROPERTY(id, N'IsUserTable') =
    1)
102.        drop table [dbo].[fixedother]
103.        GO
104.
105.        if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[goods]') and OBJECTPROPERTY(id, N'IsUserTable') = 1)
106.        drop table [dbo].[goods]
107.        GO
108.
109.        if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[goods_date]') and OBJECTPROPERTY(id, N'IsUserTable') =
    1)
110.        drop table [dbo].[goods_date]
111.        GO
112.
113.        if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[mangeacc]') and OBJECTPROPERTY(id, N'IsUserTable') =
    1)
114.        drop table [dbo].[mangeacc]
115.        GO
116.
```

```
117.    if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[orde]') and OBJECTPROPERTY(id, N'IsUserTable') = 1)
118.    drop table [dbo].[orde]
119.    GO
120.
121.    if exists (select * from dbo.sysobjects where id =
    object_id(N'[dbo].[usertype]') and OBJECTPROPERTY(id, N'IsUserTable') = 1)
122.    drop table [dbo].[usertype]
123.    GO
124.
125.    CREATE TABLE [dbo].[account] (
126.        [accountno] [varchar] (15) COLLATE Arabic_CI_AS NOT NULL
    ,
127.        [bankname] [nvarchar] (40) COLLATE Arabic_CI_AS NULL ,
128.        [accounttype] [nvarchar] (10) COLLATE Arabic_CI_AS NULL
129.    ) ON [PRIMARY]
130.    GO
131.
132.    CREATE TABLE [dbo].[accounts] (
133.        [acno] [int] NOT NULL ,
134.        [acdate] [varchar] (10) COLLATE Arabic_CI_AS NOT NULL
135.    ) ON [PRIMARY]
136.    GO
137.
138.    CREATE TABLE [dbo].[acoun] (
139.        [acno] [int] NOT NULL ,
140.        [accountno] [varchar] (15) COLLATE Arabic_CI_AS NOT NULL
    ,
141.        [actype] [varchar] (10) COLLATE Arabic_CI_AS NULL ,
142.        [acdate] [varchar] (10) COLLATE Arabic_CI_AS NULL
143.    ) ON [PRIMARY]
144.    GO
145.
146.    CREATE TABLE [dbo].[bankaccount] (
147.        [acno] [int] NOT NULL ,
148.        [acnobank] [nvarchar] (20) COLLATE Arabic_CI_AS NULL ,
149.        [bankname] [nvarchar] (50) COLLATE Arabic_CI_AS NULL ,
150.        [actyp] [varchar] (10) COLLATE Arabic_CI_AS NULL ,
151.        [bankaddress] [nvarchar] (50) COLLATE Arabic_CI_AS NULL ,
152.        [banktel] [int] NULL ,
153.        [fax] [int] NULL
154.    ) ON [PRIMARY]
155.    GO
156.
157.    CREATE TABLE [dbo].[baying_invoice] (
158.        [cno] [int] NOT NULL ,
```

```
159.            [gprice] [nvarchar] (20) COLLATE Arabic_CI_AS NULL,
160.            [edate] [varchar] (10) COLLATE Arabic_CI_AS NOT NULL,
161.            [gamount] [int] NULL,
162.            [gname] [nvarchar] (40) COLLATE Arabic_CI_AS NULL,
163.            [gunit] [nvarchar] (10) COLLATE Arabic_CI_AS NULL
164.        ) ON [PRIMARY]
165.        GO
166.
167.        CREATE TABLE [dbo].[customers] (
168.            [cno] [int] NOT NULL,
169.            [ctype] [varchar] (10) COLLATE Arabic_CI_AS NULL,
170.            [cname1] [varchar] (10) COLLATE Arabic_CI_AS NULL,
171.            [cname2] [varchar] (10) COLLATE Arabic_CI_AS NULL,
172.            [cname3] [varchar] (10) COLLATE Arabic_CI_AS NULL,
173.            [cname4] [varchar] (10) COLLATE Arabic_CI_AS NULL,
174.            [caddress] [nvarchar] (40) COLLATE Arabic_CI_AS NULL,
175.            [ctel] [int] NULL,
176.            [caccountno] [varchar] (15) COLLATE Arabic_CI_AS NULL,
177.            [ccompanyname] [nvarchar] (40) COLLATE Arabic_CI_AS
    NULL,
178.            [password] [varchar] (50) COLLATE Arabic_CI_AS NULL,
179.            [cemail] [nvarchar] (40) COLLATE Arabic_CI_AS NULL
180.        ) ON [PRIMARY]
181.        GO
182.
183.        CREATE TABLE [dbo].[employees] (
184.            [eno] [int] NOT NULL,
185.            [ename1] [varchar] (19) COLLATE Arabic_CI_AS NULL,
186.            [ename2] [varchar] (10) COLLATE Arabic_CI_AS NULL,
187.            [ename3] [varchar] (10) COLLATE Arabic_CI_AS NULL,
188.            [ename4] [varchar] (10) COLLATE Arabic_CI_AS NULL,
189.            [etimework] [varchar] (10) COLLATE Arabic_CI_AS NULL,
190.            [eadress] [nvarchar] (40) COLLATE Arabic_CI_AS NULL,
191.            [esince] [varchar] (15) COLLATE Arabic_CI_AS NULL,
192.            [eintry] [smallint] NULL,
193.            [efamily] [varchar] (10) COLLATE Arabic_CI_AS NULL,
194.            [enochild] [tinyint] NULL,
195.            [noyear] [tinyint] NULL,
196.            [enoaccount] [varchar] (15) COLLATE Arabic_CI_AS NULL,
197.            [epartwork] [nvarchar] (40) COLLATE Arabic_CI_AS NULL,
198.            [etel] [varchar] (10) COLLATE Arabic_CI_AS NULL,
199.            [datework] [varchar] (15) COLLATE Arabic_CI_AS NULL
200.        ) ON [PRIMARY]
201.        GO
202.
203.        CREATE TABLE [dbo].[exp_invoice_details] (
```

```
204.            [ino] [nvarchar] (20) COLLATE Arabic_CI_AS NOT NULL,
205.            [gno] [nvarchar] (40) COLLATE Arabic_CI_AS NOT NULL,
206.            [gamount] [int] NULL,
207.            [gprice] [nvarchar] (20) COLLATE Arabic_CI_AS NULL,
208.            [gunit] [nvarchar] (20) COLLATE Arabic_CI_AS NULL
209.    ) ON [PRIMARY]
210.    GO
211.
212.    CREATE TABLE [dbo].[expinvoice] (
213.            [ino] [nvarchar] (20) COLLATE Arabic_CI_AS NOT NULL,
214.            [expdate] [varchar] (10) COLLATE Arabic_CI_AS NULL,
215.            [explocation] [nvarchar] (20) COLLATE Arabic_CI_AS NULL,
216.            [cno] [int] NULL
217.    ) ON [PRIMARY]
218.    GO
219.
220.    CREATE TABLE [dbo].[fixedaccount] (
221.            [acno] [int] NOT NULL,
222.            [fnam] [nvarchar] (40) COLLATE Arabic_CI_AS NULL,
223.            [actype] [varchar] (10) COLLATE Arabic_CI_AS NULL,
224.            [fval] [int] NULL
225.    ) ON [PRIMARY]
226.    GO
227.
228.    CREATE TABLE [dbo].[fixedcare] (
229.            [acno] [int] NOT NULL,
230.            [actype] [varchar] (10) COLLATE Arabic_CI_AS NULL,
231.            [cname] [varchar] (10) COLLATE Arabic_CI_AS NULL,
232.            [datefinish] [varchar] (50) COLLATE Arabic_CI_AS NULL,
233.            [carval] [int] NULL,
234.            [typecare] [nvarchar] (40) COLLATE Arabic_CI_AS NULL,
235.            [careno] [int] NULL
236.    ) ON [PRIMARY]
237.    GO
238.
239.    CREATE TABLE [dbo].[fixedother] (
240.            [acno] [int] NOT NULL,
241.            [acdate] [varchar] (10) COLLATE Arabic_CI_AS NULL,
242.            [acname] [nvarchar] (20) COLLATE Arabic_CI_AS NULL,
243.            [actype] [varchar] (10) COLLATE Arabic_CI_AS NULL,
244.            [acval] [int] NULL
245.    ) ON [PRIMARY]
246.    GO
247.
248.
249.    CREATE TABLE [dbo].[goods] (
            [gno] [nvarchar] (40) COLLATE Arabic_CI_AS NOT NULL,
```

```
250.            [gname] [nvarchar] (40) COLLATE Arabic_CI_AS NULL,
251.            [gamount] [int] NULL,
252.            [cno] [int] NULL,
253.            [gpric] [nvarchar] (20) COLLATE Arabic_CI_AS NULL,
254.            [gunit] [nvarchar] (20) COLLATE Arabic_CI_AS NULL
255.        ) ON [PRIMARY]
256.        GO
257.
258.        CREATE TABLE [dbo].[goods_date] (
259.            [gno] [nvarchar] (40) COLLATE Arabic_CI_AS NOT NULL,
260.            [expiredate] [nvarchar] (15) COLLATE Arabic_CI_AS NOT
    NULL,
261.            [productdate] [nvarchar] (15) COLLATE Arabic_CI_AS NULL
262.        ) ON [PRIMARY]
263.        GO
264.
265.        CREATE TABLE [dbo].[mangeacc] (
266.            [acno] [int] NOT NULL,
267.            [actype] [varchar] (10) COLLATE Arabic_CI_AS NULL,
268.            [typeact] [varchar] (20) COLLATE Arabic_CI_AS NULL,
269.            [actval] [int] NULL
270.        ) ON [PRIMARY]
271.        GO
272.
273.        CREATE TABLE [dbo].[orde] (
274.            [gname] [nvarchar] (40) COLLATE Arabic_CI_AS NULL,
275.            [gamount] [int] NULL,
276.            [odate] [nvarchar] (10) COLLATE Arabic_CI_AS NULL,
277.            [explocation] [nvarchar] (40) COLLATE Arabic_CI_AS NULL,
278.            [cno] [int] NOT NULL,
279.            [gno] [nvarchar] (40) COLLATE Arabic_CI_AS NOT NULL
280.        ) ON [PRIMARY]
281.        GO
282.
283.        CREATE TABLE [dbo].[usertype] (
284.            [uname] [nvarchar] (50) COLLATE Arabic_CI_AS NOT NULL,
285.            [upassword] [varchar] (16) COLLATE Arabic_CI_AS NULL,
286.            [utype] [char] (1) COLLATE Arabic_CI_AS NULL,
287.            [cno] [int] NULL,
288.            [eno] [int] NULL
289.        ) ON [PRIMARY]
290.        GO
291.
```