



Faculty of Engineering & Information Technology

Computer Science Department

Graduation Project

Almodeer almohasabi

Presented in partial fulfillment of the requirements for Bachelor degree
in Computer Engineering.

Prepared by:

Yaseen N. Hammouz

Abdulraheem M. Zaqout

Submitted to:

Dr. Nizar Awartani

Submitted on:

May 6th, 2019

Acknowledgment

We would like to express our gratitude for everyone who helped us during the graduation project, starting with our supervisor Dr. Nizar Awartani who didn't keep any effort in encouraging us to do a great job, providing our group with valuable information and advices to do better each time, and we really thank him for understanding, patience, and most importantly, his friendship during the work. Thanks for the continuous support and kind communication which had a great effect regarding enthusiasm about what we are working on. Special thanks to our "Computer science department" and its great doctors and staff.

Disclaimer

This report was written by student(s) at the Computer Science Department, Faculty of Engineering, An-Najah National University. It has not been altered or corrected, other than editorial corrections, as a result of assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of the student(s). An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.

Contents

Acknowledgment	2
Disclaimer	3
Table of Figures	6
Abstract	7
Chapter 1 Introduction	8
1.1 Problem Description	8
1.2 Objectives.....	8
1.3 Scope of work	8
1.4 Glossary.....	8
1.5 Importance	9
1.6 Organization of the documentation	9
Chapter 2 Constraints, Standards and Earlier course work	9
2.1 Constraints	9
2.1.1 Time.	9
2.1.2 Resources	9
2.2 Standards	10
2.3 Earlier Coursework	10
Chapter 3 Literature Review	9
Chapter 4 Methodology	11
4.1 Technical Choices	11
4.1.1 C#	11
4.1.2 DevExpress components	11
4.2 Experimental Methods	12
4.2.1 Database.....	12
4.2.2 program pages structure.....	17
4.2.2.1 Login page:.....	17
4.2.2.2 Home Page:.....	18
4.2.2.3 Add product Page:.....	19
4.2.2.4 Query products page:.....	20
4.2.2.5 Add customer/supplier page:	21
4.2.2.6 Query customer/supplier page:.....	22

4.2.2.7 Sales bill page:.....	23
4.2.2.8 Purchase bill page:	23
4.2.2.9 Return sales page:.....	24
4.2.2.10 Return purchase bill page:.....	24
4.2.2.11 Receipt voucher page:	25
4.2.2.12 Acceptance voucher page.....	26
4.2.2.13 Point of sale (POS) page:	27
4.2.2.14 User's page:	28
4.2.2.15 Query Chart of account page:.....	29
4.2.2.16 Banks page:.....	30
4.2.2.17 Checks page:	30
4.2.2.18 Settings pages:	31
Chapter 5 Results and Discussion.....	36
Chapter 6 Conclusion and Recommendation	38
Chapter 7 References.....	39

Table of Figures

Figure 1-use cases	10
Figure 2-Database diagram.....	13
Figure 3-Database diagram.....	14
Figure 4-login page.....	17
Figure 5- home page.....	18
Figure 6-add product	19
Figure 7-query products	20
Figure 8-add customer.....	21
Figure 9-add supplier.....	21
Figure 10-query customer/supplier	22
Figure 11-sales bill.....	23
Figure 12-purchase bill	23
Figure 13-return sales bill	24
Figure 14-return purchase bill.....	24
Figure 15-receipt voucher.....	25
Figure 16-payment voucher.....	26
Figure 17-POS page	27
Figure 18-user pages	29
Figure 19-query Chart of account page	29
Figure 20-banks page.....	30
Figure 21-checks page	30
Figure 22-customers categories.....	31
Figure 23-price categories	31
Figure 24-Cities and villages.....	31
Figure 25-currencies page	32
Figure 26-initial settings	32
Figure 27-constant settings	33
Figure 28-general settings	33
Figure 29-gold settings	34
Figure 30-improt categories from excel	34
Figure 31-import banks from excel	35
Figure 32-checks exchange rate for all currencies	36
Figure 33-checks the empty fields.....	37
Figure 34-retrieve all processes that have been done before	37

Abstract

Jewelry shops deal with different types of people every day. Even they are customers or dealers. A lot of calculations are made daily with different currencies (JOD, NIS, USD). That makes the manager aware of any loss. But being aware is not enough.

Actually, even with bills and recording every process have been done some times losses appear. So shops like these need an organized system that takes control of all these processes.

Almodeer almohasabi is a responsive desktop application that helps the managers of jewelry shops (specialized for golds) to organize their calculations in a customized manner. Through our application, users will not miss anything matter to their benefits. Everything will be stored in an organized database. The project provides a simple and easy user experience.

Chapter 1 Introduction

1.1 Problem Description

Nowadays, jewelry shops spreaded all over the world. Almost everyone have tried to deal with jewelries whether buying or selling them. That makes jewelry shops deal with too many people daily with different desires.

Unfortunately, dealing with this number of people daily causes incorrect calculations. However, this problem can be solved by having a service that does all these calculations automatically and store all needed information in an organized database. This application helps the manager of the jewelry shop to save his time, avoiding loss, organizing all different bills, and moreover.

Based on all of that, we came up with the idea of building a desktop application (Almodeer almohasabi) that solves this problem and facilitates on people dealing with money and gold stuffs.

1.2 Objectives

The main objective of our project is to build a user friendly, desktop-responsive application that serves as an organizer and helper at the same time. In which users are started using this application to help them with their work.

1.3 Scope of work

This application prompts only for jewelry shops (specialized for golds). Users just need to enter basic information in order to make this application works regarding on it.

1.4 Glossary

C#	Simple, modern, general-purpose, object-oriented programming language developed by Microsoft.
Database	Is an organized collection of <u>data</u> , generally stored and accessed electronically from a computer system. Where databases are more complex they are often developed using formal modeling techniques.
Sqlserver 2014	Is a powerful and reliable free data management system that delivers a rich and reliable data store for lightweight Web Sites and desktop applications.
DevExpress components	User Interface Components for Windows, the Web and Your Mobile World

1.5 Importance

With the variety in the types of jewelry, it would be hard for each of us to be acquaintance with all of these types. So and on the other hand, it would be more helpful for the owners of these shops to have a system that organizes all the work in sufficient database. As a result, the owner will have no loss and keep all his data stored in an organized way.

1.6 Organization of the documentation

The report is organized in six chapters. In chapter 1, we introduce our project briefly. In chapter 2, we describe the constraints we faced before and during the development process, standard we followed and courses that helped us in the project. In chapter 3, we discuss the language and IDE that we used and the most helpful included features. In chapter 4, we describe our methodology and the development of all parts of our project. In chapter 5, we discuss the results we achieved. Finally the project Conclusion, recommendations and future work in chapter 6.

Chapter 2 Overall Description

2.1 Product perspective:

An accounting program that deals with selling and buying operations according to rules pertaining to gold, in addition to storages and terms of use:

- Products information:
 - It includes all information about product properties.
- Customers and supplier information:
 - It includes all information about customer and supplier.
- Bills information:
 - It includes all information about sales and purchases bills.
- Vouchers information:
 - It includes all information about receipt, journal and payment vouchers.
- User information:
 - It includes about information about users (username, password, accounts and terms of use).

2.2 Constraints:

2.2.1 Time:

Learning new programing language such as C# and how to deal with DevExpress components [1] to the user interface.

2.2.2 Resources:

Choosing the best and the right DevExpress components to create a friendly user interface. We chose DevExpress components because its features are suitable for our application and easy to learn.

2.3 Standards:

C# is the fundamental source code of our application, DevExpress components for the user interface and sqlserver 2014 for database.

2.4 Earlier Coursework:

During our studies at the computer science department, we have taken some classes that helped us in the development of this project:

- The Database course: this course was necessary to build the database, storage system and relations in our project.
- The C++ courses: it was the main course that we utilized. It include basics of handling with C# language that we used in the development of the project.

2.5 use cases:

The user will be able to do multi usings we will shown these uses in the following diagram.

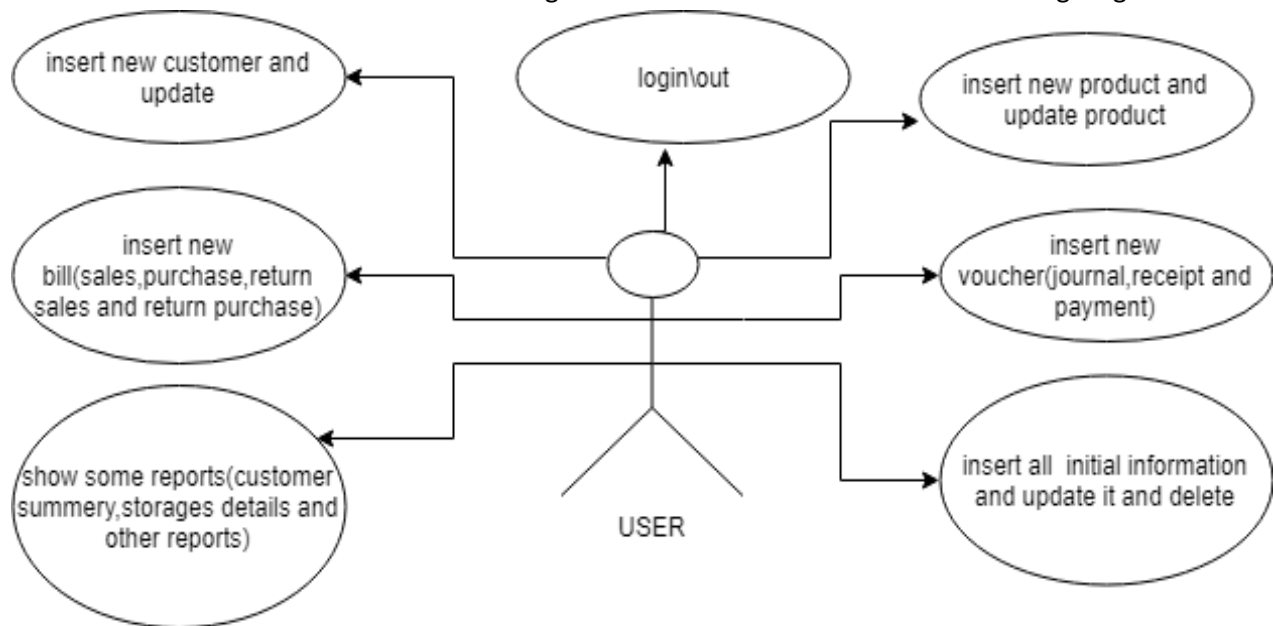


Figure 1-use cases

Chapter 3 Literature Review

There are several accounting program in local region that offer services for accounting and management. These programs are dealing with several sectors like Industrial, commercial, hospitals, etc. However, these applications do not focus on jewelry shops especially gold shops.

Therefore, our application focuses on jewelry shops (specialized for gold). **Almodeer almohasabi** is designed to be a handler for all accounting and calculation stuffs in these shops. Flexibility and friendly use are some of features that **Almodeer almohasabi** characterizes, where users can deal with it without high experience on computers.

Chapter 4 Methodology

4.1 Technical Choices

4.1.1 C# programming language

What is c#?

C# is a hybrid of C and C++, it is a Microsoft programming language developed to compete with Sun's Java language. C# is an object-oriented programming language used with XML-based Web services on the .NET platform and designed for improving productivity in the development of Web applications. [2]

Why C#?

C# has several features that make it a great choice as a source code for developing applications. These are some features that we benefit from mostly in our application:

- C# is a simple, modern, object oriented language derived from C++ and Java.
- The classes and data types are common to all of the .NET languages.
- .NET includes a Common Execution engine and a rich class library.
- In C#, Microsoft has taken care of C++ problems, such as Memory management, pointers etc.
- It supports garbage collection, automatic memory management, and a lot.

4.1.2 DevExpress components

What is DevExpress components?

DevExpress is a custom third party provider of .NET controls. They customize the .NET controls by making it more attractive and more flexible than inbuilt .NET controls.

Why DevExpress?

The most important reasons are the flexibility and easy to use while developing applications.

4.1.3 Sqlserver 2014

What is Sqlserver?

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications—which may run either on the same computer or on another computer across a network (including the Internet).

Why Sqlserver? [3]

There are many advantages helped us to choose the sqlserver:

1. **Lower Cost Of Ownership:** there are express edition of sqlserver it's free.
2. **Security Features Are Better:** SQL Server uses Policy-Based Management to detect security policies that are non-compliant. This feature allows only authorized personnel access to the database. Security audits and events can be written automatically to log files.
3. **Installation Is Streamlined:** It can be installed via a setup wizard and the prerequisite updates are detected and downloaded by the installer automatically.

4.2 Experimental Methods

4.2.1 Database

When we built the database we tried to follow all basics we learned it in the database and advanced database courses we studied. In this section we will show the diagram of database exported using sqlserver and the technique we used to do queries (insert, update, select and delete).

4.2.1.1 Database diagram:

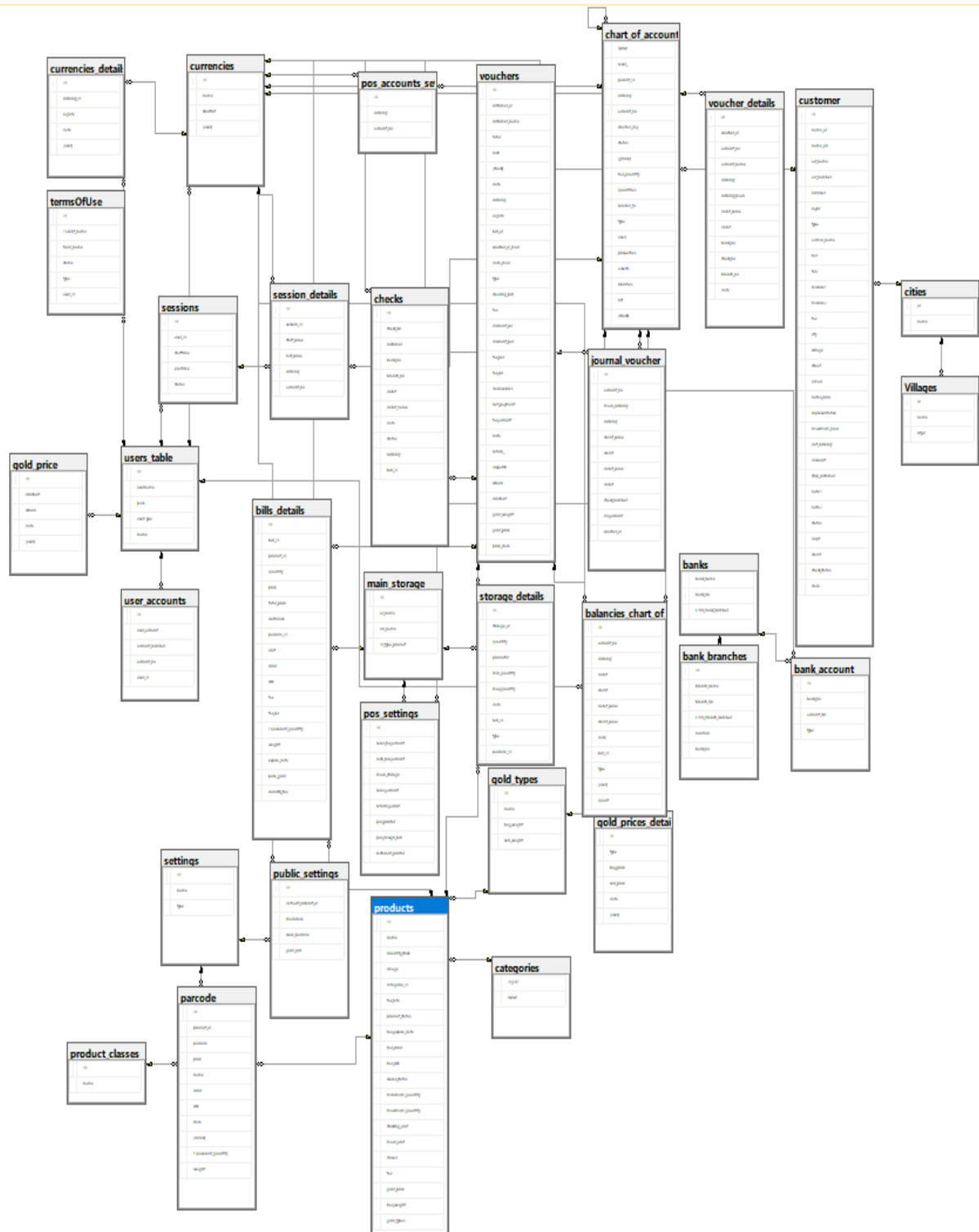


Figure 2-Database diagram

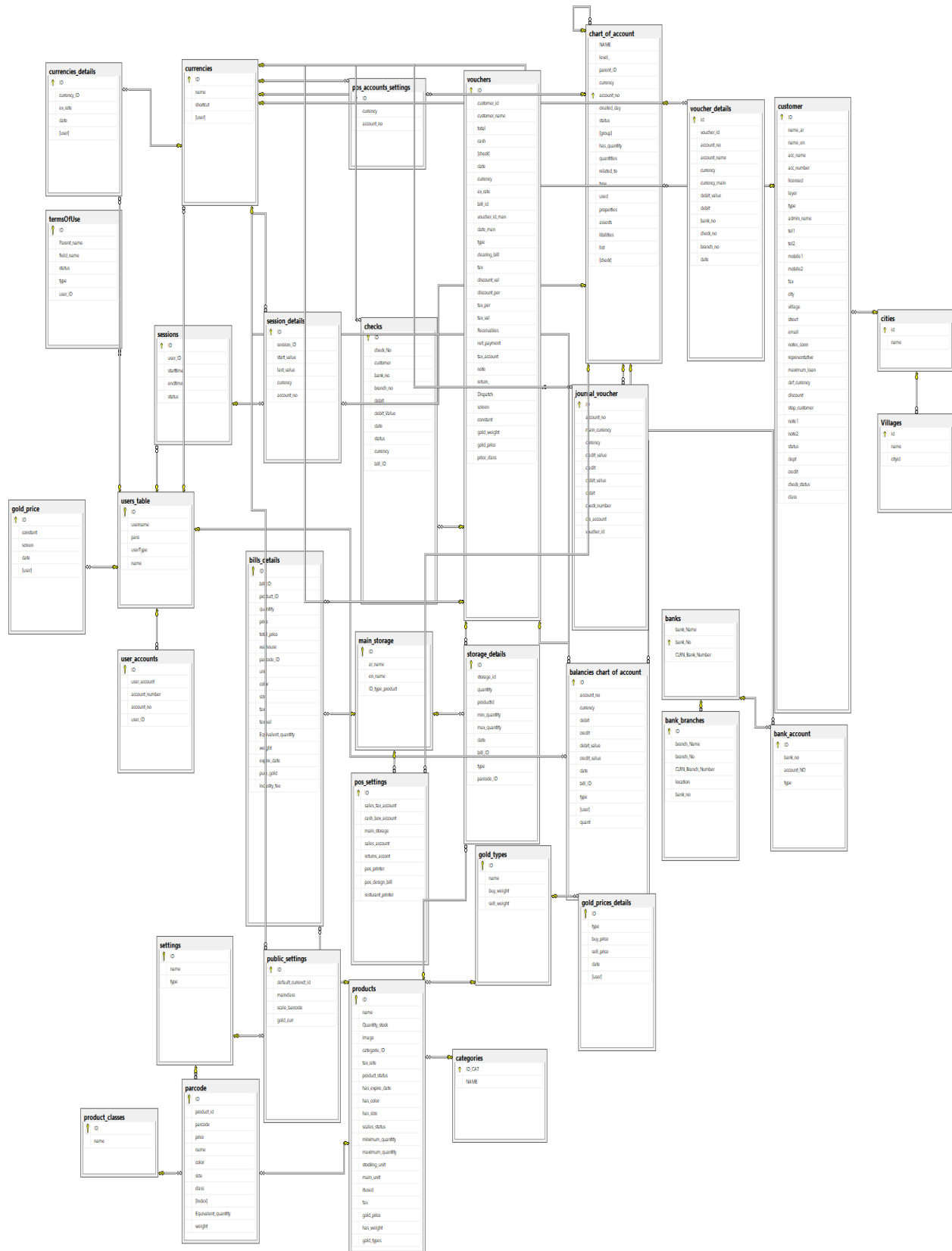


Figure 3-Database diagram

NOTE: Can download the original diagrams from this URL

(https://drive.google.com/open?id=1NkchGsohEjrb_71Nv4_M9KXD3mdSb2-x).

4.2.1.2 Database query technique:

We used **System.Data.SqlClient** library to use sqlcommand class and add query information to sqlcommand object then and execute the sqlcommand object to do the query. We will show some of this queries in this section:

Delete query:

```
public void delete_settings(int id)
{
    DataTable dt = this.get_pos_settings();
    string query = "DELETE FROM[dbo].[settings] where ID = @ID";
    SqlCommand cmd = new SqlCommand(query, conn);
    cmd.Parameters.AddWithValue("@ID", id);
    conn.Open();
    int result = cmd.ExecuteNonQuery();
    conn.Close();
}
```

Insert query:

```
public void insert_into_tax(string name, decimal value)
{
    string query = "INSERT INTO[dbo].[tax]([Name],[value])VALUES(@Name,@value)";

    SqlCommand cmd = new SqlCommand(query, conn);
    cmd.Parameters.AddWithValue("@Name", name);
    cmd.Parameters.AddWithValue("@value", value);
    conn.Open();
    int result = cmd.ExecuteNonQuery();
    conn.Close();
}
```

Select query:

```
public DataTable get_voucher_by_ID(string id)
{
    SqlCommand cmd = new SqlCommand();
    SqlDataReader reader;
    cmd.CommandText = "SELECT *FROM [dbo].[vouchers] where ID=@id";
    cmd.Parameters.AddWithValue("@id", id);
    cmd.CommandType = System.Data.CommandType.Text;
    cmd.Connection = conn;
    conn.Open();
    reader = cmd.ExecuteReader();
    DataTable dt = new DataTable();
    dt.Load(reader);
    conn.Close();
    return dt;
}
```

Update query:

```
public void update_tax(int id,decimal value,string name)
{
    SqlCommand cmd = new SqlCommand();
    SqlDataReader reader;
    cmd.CommandText = "UPDATE [dbo].[tax]SET [Name] = @Name,[value] = @value WHERE ID=@ID";
    cmd.Parameters.AddWithValue("@ID", id);
    cmd.Parameters.AddWithValue("@Name", name);
    cmd.Parameters.AddWithValue("@value", value);
    cmd.CommandType = System.Data.CommandType.Text;
    cmd.Connection = conn;
    conn.Open();
    reader = cmd.ExecuteReader();
    conn.Close();
}
```

Select using procedures:

```
public DataTable get_product_by_type_and_class_and_product_id(int id, int class_, int type)
{
    DAL.DataAccess dal = new DAL.DataAccess();
    SqlParameter[] param = new SqlParameter[3];
    param[0] = new SqlParameter("@type", SqlDbType.Int);
    param[0].Value = type;

    param[1] = new SqlParameter("@class", SqlDbType.Int);
    param[1].Value = class_;

    param[2] = new SqlParameter("@ID", SqlDbType.Int);
    param[2].Value = id;

    DataTable dt = dal.SelectData("get_product_by_type_and_class_and_product_id", param);
    return dt;
}
```


4.2.2 Program pages structure

Here an overview about our pages and the functionality of each page

4.2.2.1 Login page:

Its simple page give the user ability to access to main page and other parts of the system using username and password.the user will be access if he enter the correct user name and password and if he doesnt enter the coorrect data the system wil .be give him message box the user name or password is incorrect

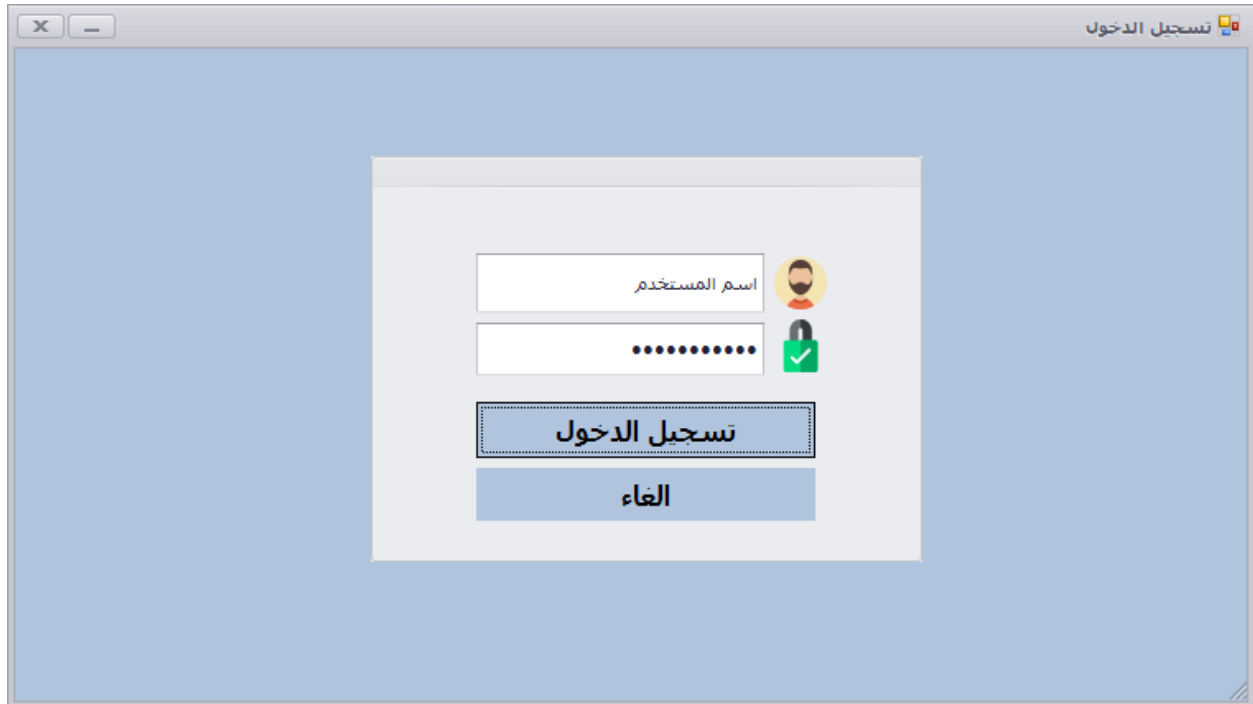


Figure 4-login page

4.2.2.2 Home Page:

This page will be the page that appears to the user once he log in. For every successful login the currencies page will be shown with the ability to change the exchange rate for every currency and the ability to change the screen price and constant value to use them to compute the gold prices. After save button clicked the system will save the exchange rate for every currency and the screen price and the constant value in the database then the system will be compute the gold price for every Gold Caliber (24, 21, 18,...) using this equation $((\text{screen price} + \text{constant value}) * 32.15 * \text{gold caliber weight}) / 1000$, and it gives us the price of one gram of gold for specific caliber and save the price in the database.

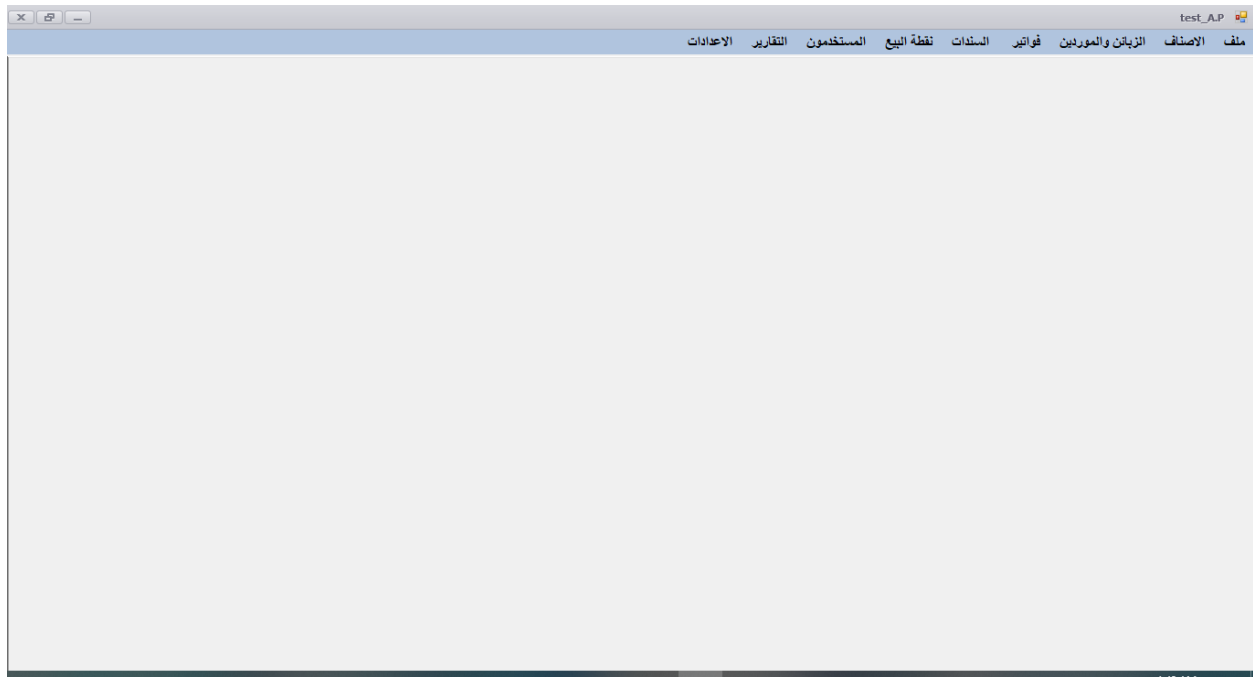


Figure 5- home page

4.2.2.3 Add product Page:

The page contains all properties of the product. It give us the ability to add new product, edit it, add the barcode the system provide multi barcode (multi barcode same product) and print barcodes.

إضافة صنف جديد

بيانات الصنف

رقم الصنف:

اسم الصنف:

المجموعات:

نسبة الضريبة:

الكمية الحالية:

الحد الأدنى:

الحد الأعلى:

الوحدة الرئيسية:

وحدة الجرد:

عيار الذهب:

بياع بالميزان: ☐

بياع بالميزان الالكتروني: ☐

تجميد الصنف: ☐

له تاريخ انتهاء صلاحية: ☐

له لون: ☐

له قياس: ☐

يعامل بسعر الذهب: ☐

له وزن: ☐

الصورة:

تحميل ملف الصورة:

متعدد الباركود و فئات الاسعار | المستودعات والكميات | الاسعار | المورد

الباركود	الوحدة	الكمية المعادلة	مفرق	جملة	جملة الجملة	بيع الشيكات
جدة	جدة					
الرجاء اختيار الوحدة						

حفظ | طباعة | معاينة طباعة | طباعة الباركود

Figure 6-add product

4.2.2.4 Query products page:

From this page we can query about product, open add new product page to add or to edit and from this page we can search about some product.

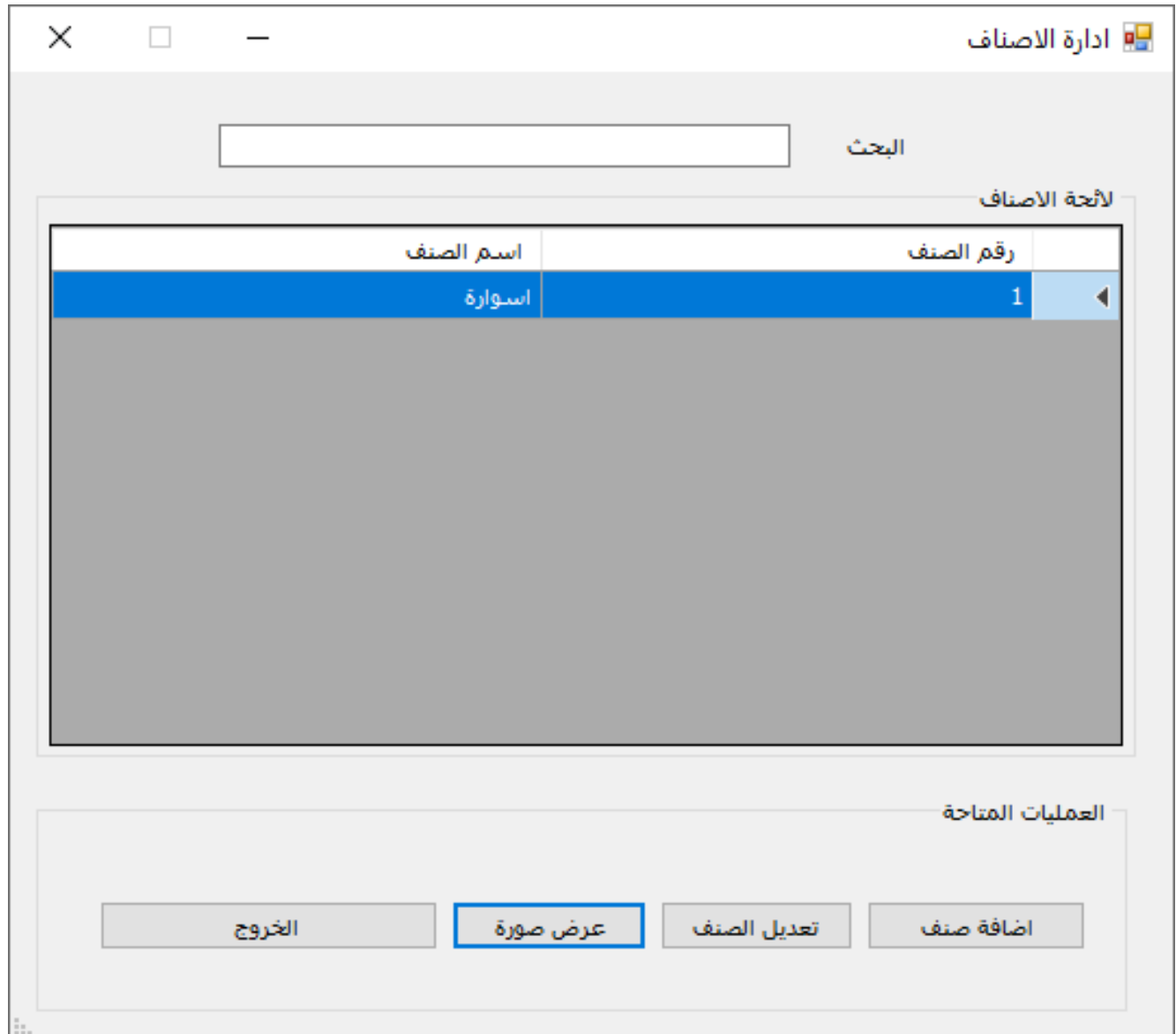


Figure 7-query products

4.2.2.5 Add customer/supplier page:

The page contains all properties of the customer/supplier. It give us the ability to add new customer, edit it.

Figure 8-add customer

Figure 9-add supplier

4.2.2.6 Query customer/supplier page:

From this page we can query about customers/suppliers, open add new customer/supplier page to add or to edit and from this page we can search about some customer/supplier.

الزبائن

البحث

لائحة الزبائن

الرقم	الاسم
1	عبد

العمليات المتاحة

تعديل الزبون

اضافة زبون

اضافة مورد

الخروج

Figure 10-query customer/supplier

4.2.2.7 Sales bill page:

It's give us ability to create new sales bill and its payment.

test_A.P

ملف الإصناف الزبائن والموردين فواتير السندات نقطة البيع المستخدمين التقارير الإعدادات

فاتورة مبيعات

رقم الفاتورة: B1
رقم الفاتورة (يوني):
رقم يدوي:

المورد:
العملة:
الاسم:
نوع الفاتورة:
الضريبة:
سعر الشاشة:

التاريخ: 5/ 6/2019
العملة: شيفل إسرائيلي
الاسم: شيفل إسرائيلي
نوع الفاتورة: فاتورة عادية
الضريبة: 0.000
سعر الشاشة: 1300.000

خيارات الفاتورة: مفرق
قائمة البيع: 1.000
سعر الصرف: شيفل إسرائيلي
فاتورة مقاصة: ☐

حساب صندوق: صندوق شيفل
حساب الصندوق: 2211000002
رقم حساب الزبون: 1114
الحساب الضريبي:

الباركود	رقم الصف	اسم الصف	الكمية	الوحدة	المستودع	قيمة الضريبة	اللون	القياس	الضريبة	الوزن	المصنعة	السعر	إجمالي الد...	صافي اللقب	حذف	عبار البيع
...

إجمالي: شيفل إسرائيلي 0.000
الضريبة: 0.000
قيمة الضريبة: 0.000
نسبة الخصم: 0.000
خصم مبلغ: 0.000
الاجمالي: 0.000

حفظ طباعة معاينة الطباعة حذف الكل حذف السابق التالي إضافة صف تعديل ترحيل استلام

Figure 11-sales bill

4.2.2.8 Purchase bill page:

Its inherited from the sales bill page but when we clicked on save button it give us ability to create new purchase bill and its payment.

test_A.P

ملف الإصناف الزبائن والموردين فواتير السندات نقطة البيع المستخدمين التقارير الإعدادات

فاتورة مشتريات

رقم الفاتورة: S2
رقم الفاتورة (يوني):
رقم يدوي:

المورد:
العملة:
الاسم:
نوع الفاتورة:
الضريبة:
سعر الشاشة:

التاريخ: 5/ 6/2019
العملة: شيفل إسرائيلي
الاسم: شيفل إسرائيلي
نوع الفاتورة: فاتورة عادية
الضريبة: 0.000
سعر الشاشة: 1300.000

خيارات الفاتورة: مفرق
قائمة البيع: 1.000
سعر الصرف: شيفل إسرائيلي
فاتورة مقاصة: ☐

حساب صندوق: صندوق شيفل
حساب الصندوق: 2211000001
رقم حساب الزبون: 1114
الحساب الضريبي:

الباركود	رقم الصف	اسم الصف	الكمية	الوحدة	المستودع	قيمة الضريبة	اللون	القياس	الضريبة	الوزن	المصنعة	السعر	إجمالي الد...	صافي اللقب	حذف	عبار البيع
...

إجمالي: شيفل إسرائيلي 0.000
الضريبة: 0.000
قيمة الضريبة: 0.000
نسبة الخصم: 0.000
خصم مبلغ: 0.000
الاجمالي: 0.000

حفظ طباعة معاينة الطباعة حذف الكل حذف السابق التالي إضافة صف تعديل ترحيل استلام

Figure 12-purchase bill

Its inherited from the sales bill page but when we clicked on save button it give us ability to create new return sales bill and its payment.

ملف الإصناف الزبائن والموردين فواتير السندات نقطة البيع المستخدمون التقارير الإعدادات

فاطمة مبرودات المشتريات

رقم الفاتورة S2 رقم الفاتورة (يومي) رقم الفاتورة (يومي) رقم يدوي

نوع الفاتورة فاتورة عادية فاتورة تجارية سعر الشاشة 1300.000

التاريخ 5/ 6/2019 العملة شيفل اسرائيلي

سعر الصرف 1.000

نسبة الضريبة 0.000

الضريبة 60.000

حساب صندوقي صندوقي شيفل

حساب مبرودات المشتريات 2212

رقم حساب الزبون الحساب الضريبي 1114

البركود	رقم الصنف	اسم الصنف	الكمية	الوحدة	المستوعود	قيمة الضريبة	اللون	القياس	الضريبة	الوزن	المصنعة	السعر	إجمالي الي...	صافي الذهب	حذف	عبار البيع
								...EditValue }	...EditValue }							

المجموع 0.000

صافي الذهب 0

قيمة الضريبة 0

نسبة الخصم 0

إجمالي الخصم 0

حفظ طباعة ممانية الطباعة حذف الكل جديد السابق التالي + اضافة صنف تعديل ترحيل استلام

Figure 13-return sales bill

4.2.2.10 Return purchase bill page:

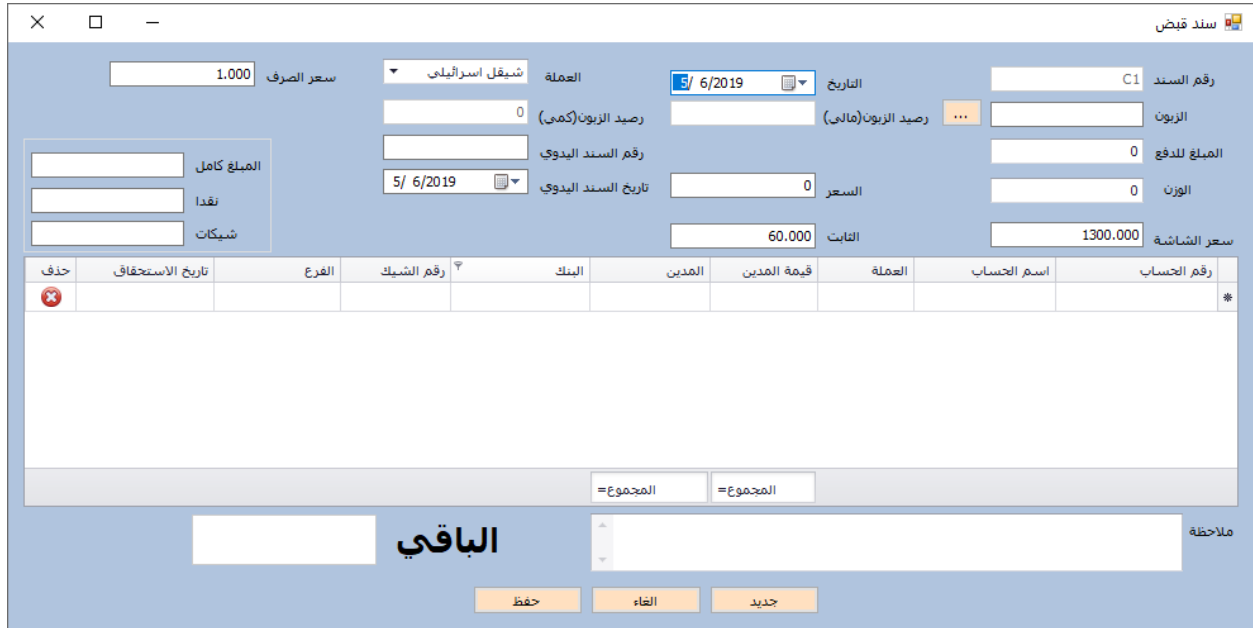
Its inherited from the sales bill page but when we clicked on save button it give us ability to create new return purchase bill and its payment.

[illegible]

Figure 14-return purchase bill

4.2.2.11 Receipt voucher page:

It give us ability to create new receipt voucher and take money from customers or suppliers in different way using cash payment or using checks payment in different currencies.



سند قبض

1.000 سعر الصرف

شيفل اسرائيلي العملة

5/ 6/2019 التاريخ

C1 رقم السند

0 رصيد الزبون (كمي)

0 رصيد الزبون (مالي)

0 المبلغ للدفع

0 الوزن

0 السعر

60.000 الثابت

1300.000 سعر الشاشة

المبلغ كامل

نقدا

شيكات

رقم السند اليدوي

5/ 6/2019 تاريخ السند اليدوي

حذف	تاريخ الاستحقاق	الفرع	رقم الشيك	البنك	المدين	قيمة المدين	العملة	اسم الحساب	رقم الحساب
									*

= المجموع

= المجموع

الباقى

ملاحظة

حفظ

الغاء

جديد

Figure 15-receipt voucher

4.2.2.12 payment voucher page

It give us ability to create new payment voucher and pay money to customers or suppliers in different way using cash payment or using checks payment in different currencies.

The screenshot shows a software interface for creating a payment voucher. The title bar reads 'سند صرف' (Payment Voucher). The form is divided into several sections:

- Top Section:** Contains fields for 'سعر الصرف' (Exchange Rate) set to 1.000, 'العملة' (Currency) set to 'شيفل اسراليلى' (Israeli Sheqel), 'التاريخ' (Date) set to '5/ 6/2019', and 'رقم السند' (Voucher Number) set to 'P1'.
- Left Panel:** A sidebar with buttons for 'المبلغ كامل' (Full Amount), 'نقدًا' (Cash), and 'شيكات' (Checks).
- Right Section:** Contains fields for 'رصيد الزبون (مالي)' (Customer Balance - Financial) set to 0, 'رقم السند اليدوي' (Manual Voucher Number) set to 5/ 6/2019, 'السعر' (Price) set to 0, and 'سعر الشاشة' (Screen Price) set to 1300.000.
- Table:** A table with columns: 'رقم الحساب' (Account Number), 'اسم الحساب' (Account Name), 'العملة' (Currency), 'قيمة المدين' (Debtor Value), 'المدين' (Debtor), 'البنك' (Bank), 'رقم الشيك' (Check Number), 'الفرع' (Branch), 'تاريخ الاستحقاق' (Due Date), and 'حذف' (Delete). The first row is empty, and the second row has a red 'X' icon in the 'حذف' column.
- Bottom Section:** Includes summary buttons labeled '=المجموع' (Total), a large text area for 'ملاحظة' (Note), and action buttons 'حفظ' (Save), 'خروج' (Exit), and 'جديد' (New).

Figure 16-payment voucher

4.2.2.13 Point of sale (POS) page:

It is designed to be comfortable to the users. In this page we can do almost all sales account operation like sales bill, return sales bill, receivables pay and all type of payment except checks payment.

[illegible]

Figure 17-POS page

4.2.2.14 User's page:

It designed using tabbed layout. In this page we can and new user with user name and password, add term of use to the specific user (in the system we use terms of use on main pages only) and we can add specific account from chart of account to the specific user.

رقم المستخدم	الاسم	اسم المستخدم	كلمة المرور	نوع المستخدم	حذف
1	admin	admin	admin	مسؤول	X
12	jamal	jamal	jamal	مستخدم عادي	X
[EditValue is null]	[EditValue is null]	[EditValue is null]	[EditValue is null]	[EditValue is null]	X

ملف	تسجيل الدخول	استعلام الحسابات	انشاء نسخة احتياطية	استعادة نسخة احتياطية	تسجيل الخروج	استلام وتسليم عهدة	الاصناف	الزبائن والموردين	فواتير	السندات	نقطة البيع	المستخدمون	التقارير
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

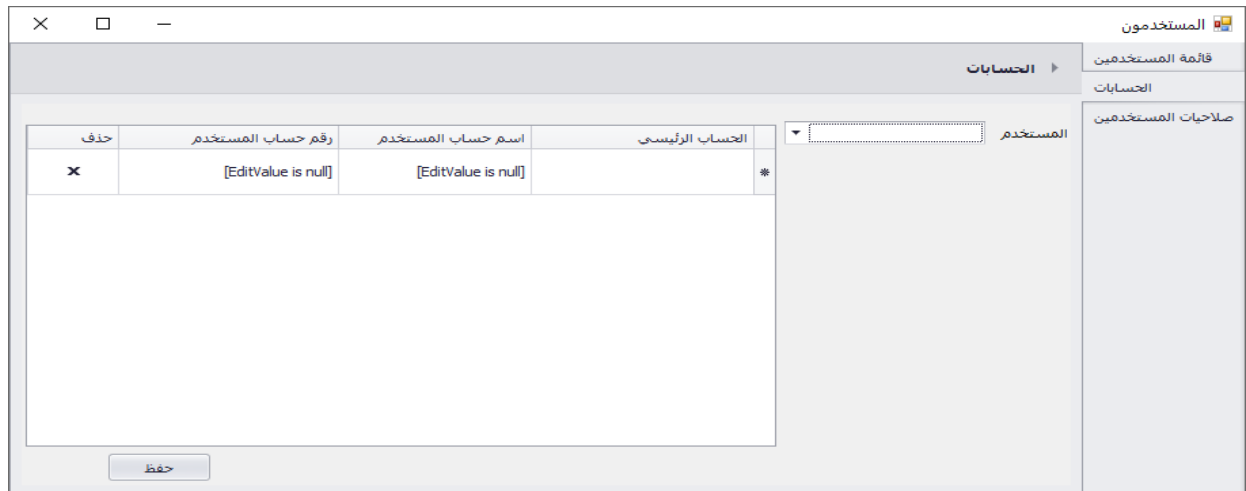


Figure 18-user pages

4.2.2.15 Query Chart of account page:

This page gives the users the ability to query about specific account balances using the tree list to choose the account and the grid view to show data and add new account in the specific location in the tree.

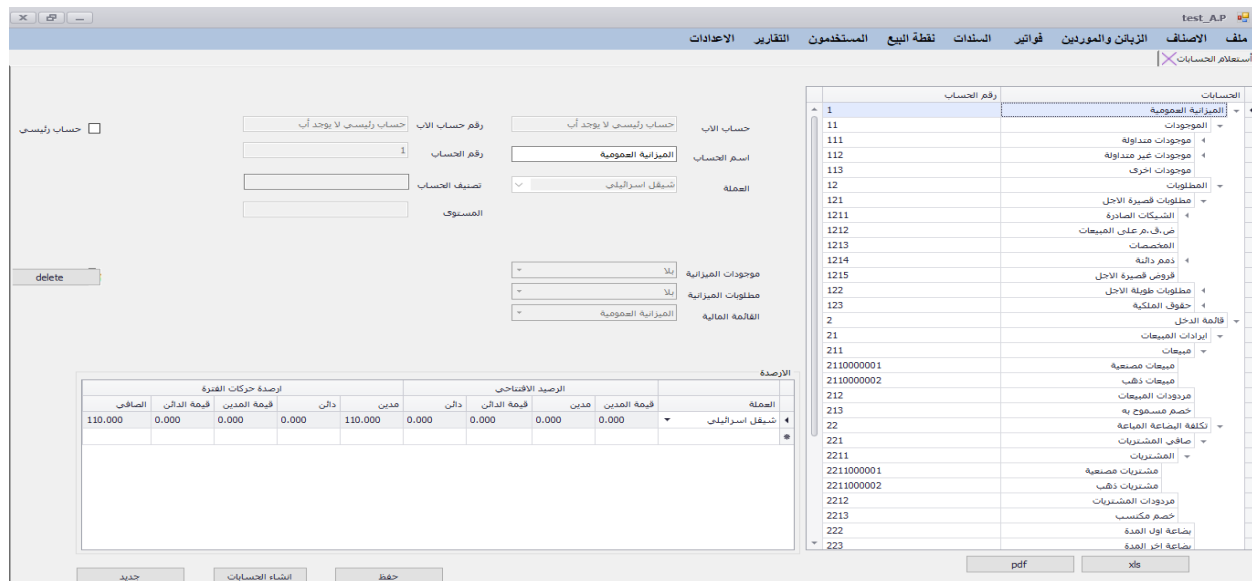


Figure 19-query Chart of account page

4.2.2.16 Banks page:

It gives the user the ability to add banks account to chart of account, add new bank and its branches and query the bank accounts and its branches.

Figure 20-banks page

4.2.2.17 Checks page:

This page gives the user the ability to do all checks process.

Figure 21-checks page

4.2.2.18 Settings pages:

This page let the user make some changes in the application

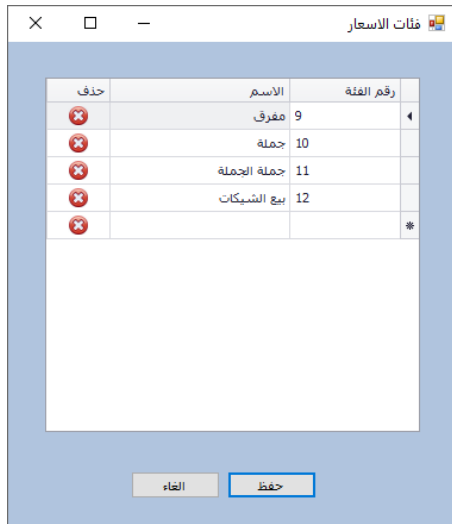


Figure 23-price categories

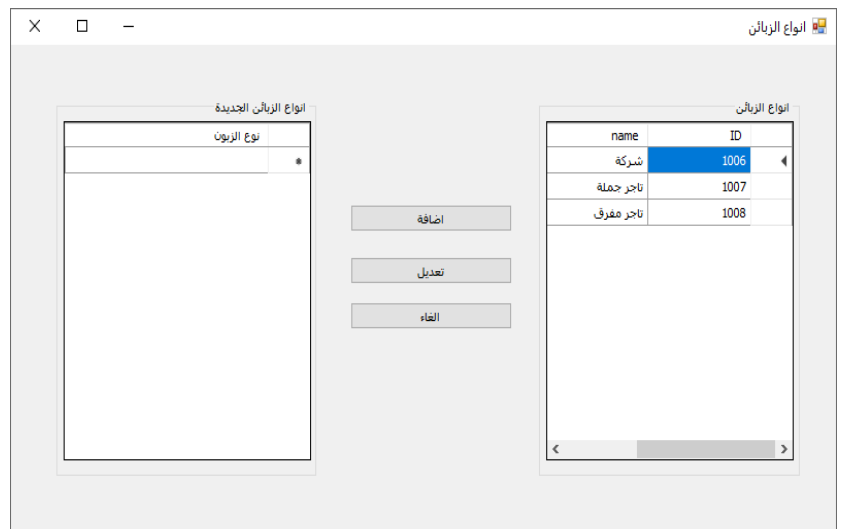


Figure 22-customers categories

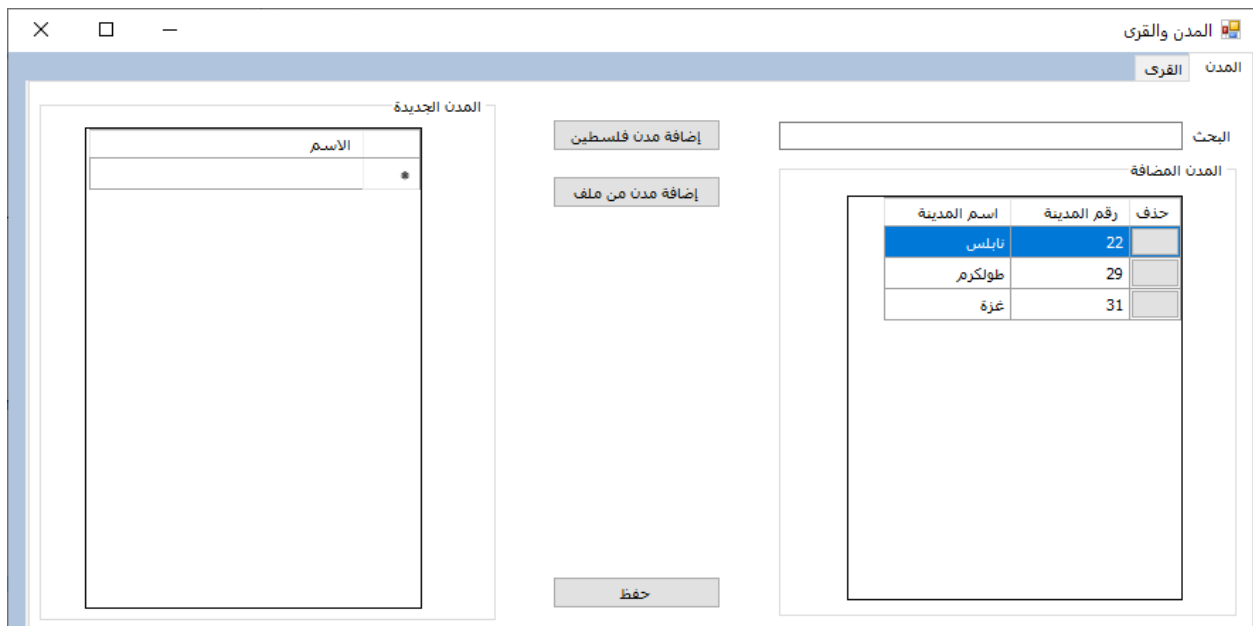


Figure 24-Cities and villages



Figure 25-currencies page

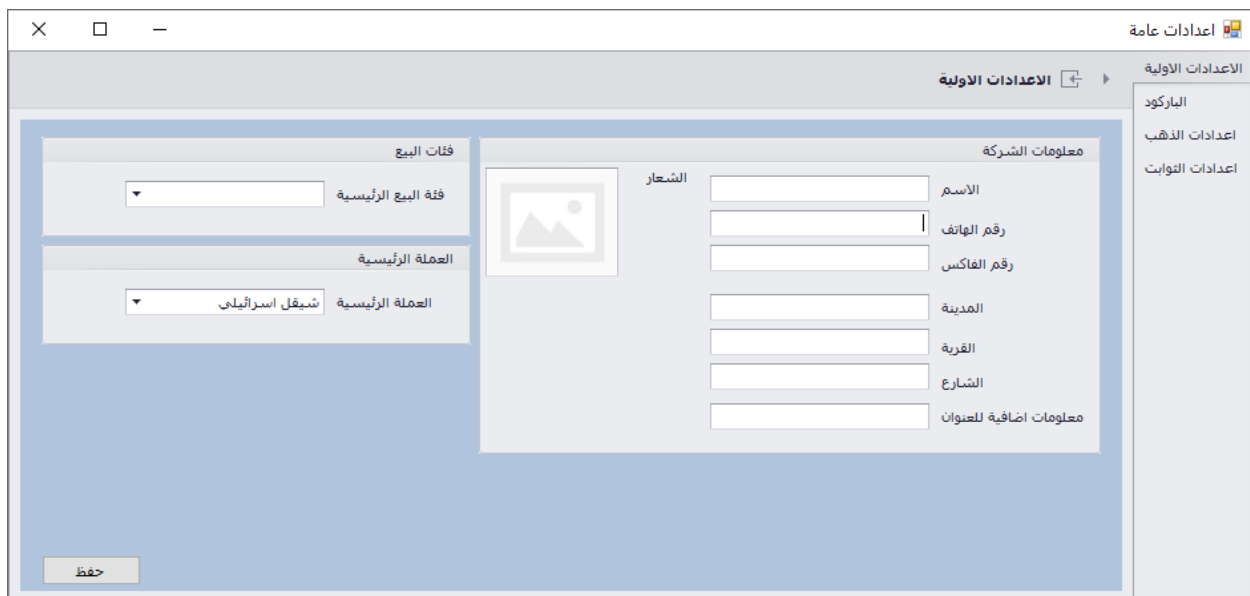


Figure 26-initial settings

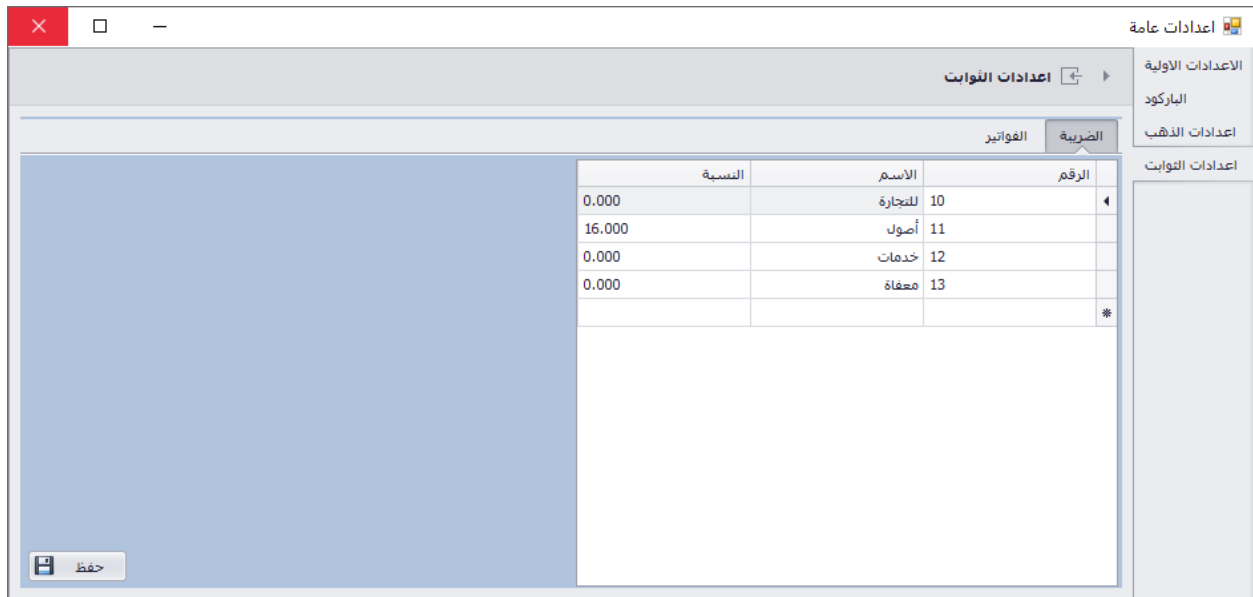


Figure 27-constant settings

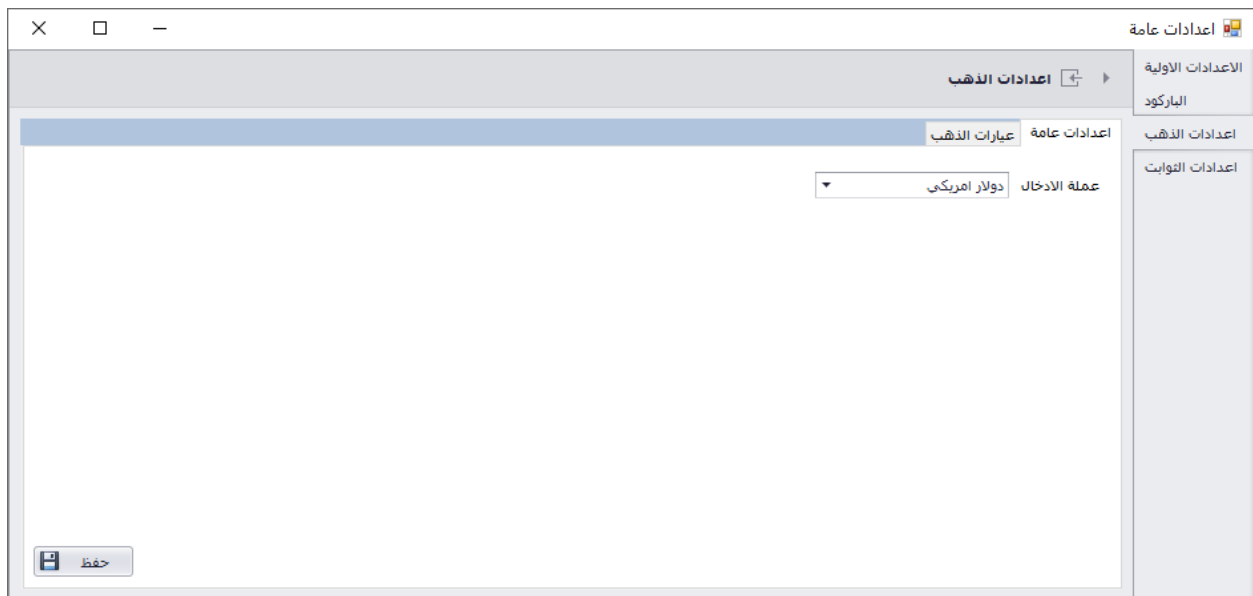


Figure 28-gold settings



Figure 29-gold settings

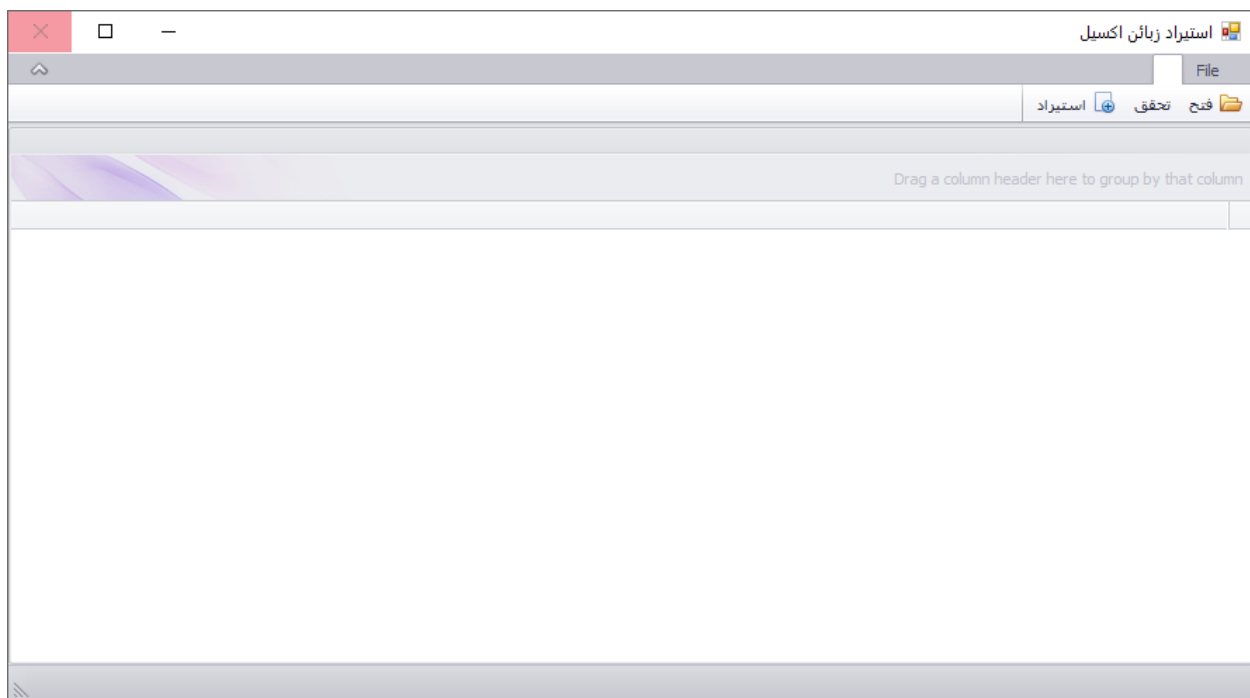


Figure 30-import customers from excel

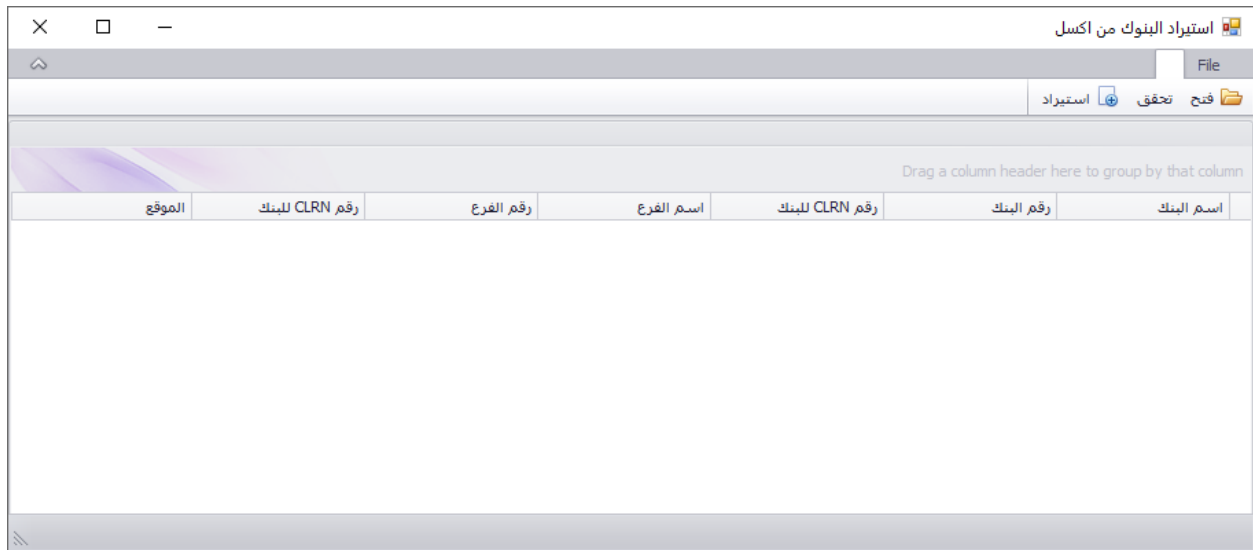


Figure 31-import banks from excel

Chapter 5 Results and Discussion

In order to prove the results of our project, we present the following test scenarios of the various activities on the application:

Checking the currencies, constant and screen price

All jewelry shops work depends mainly on the currencies and its exchange rate. So we checks everything the user might forget or enter incorrect input.

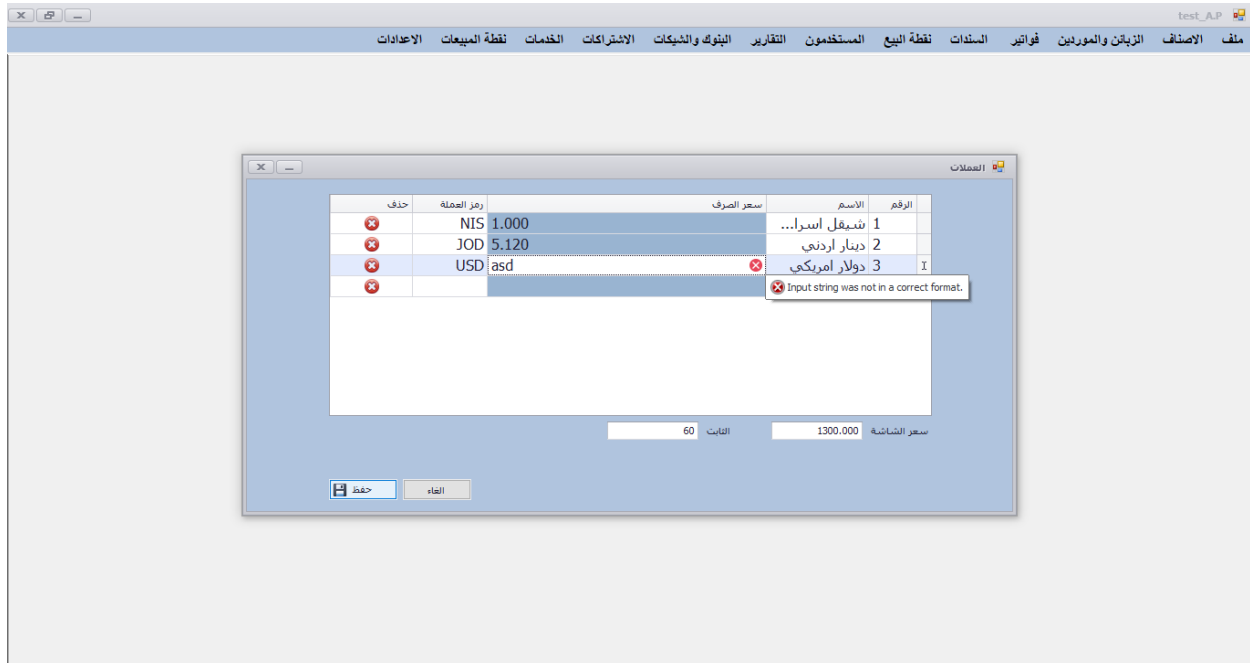


Figure 32-checks exchange rate for all currencies

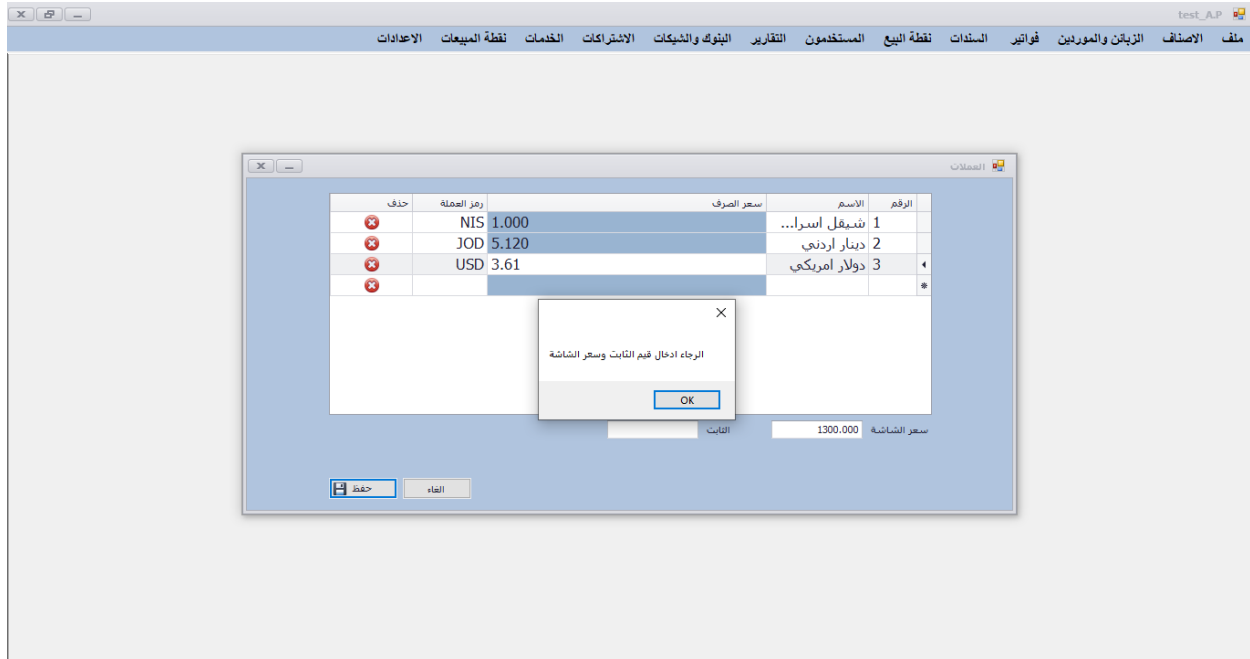


Figure 33-checks the empty fields

The ability to retrieve any needed bill or journal voucher

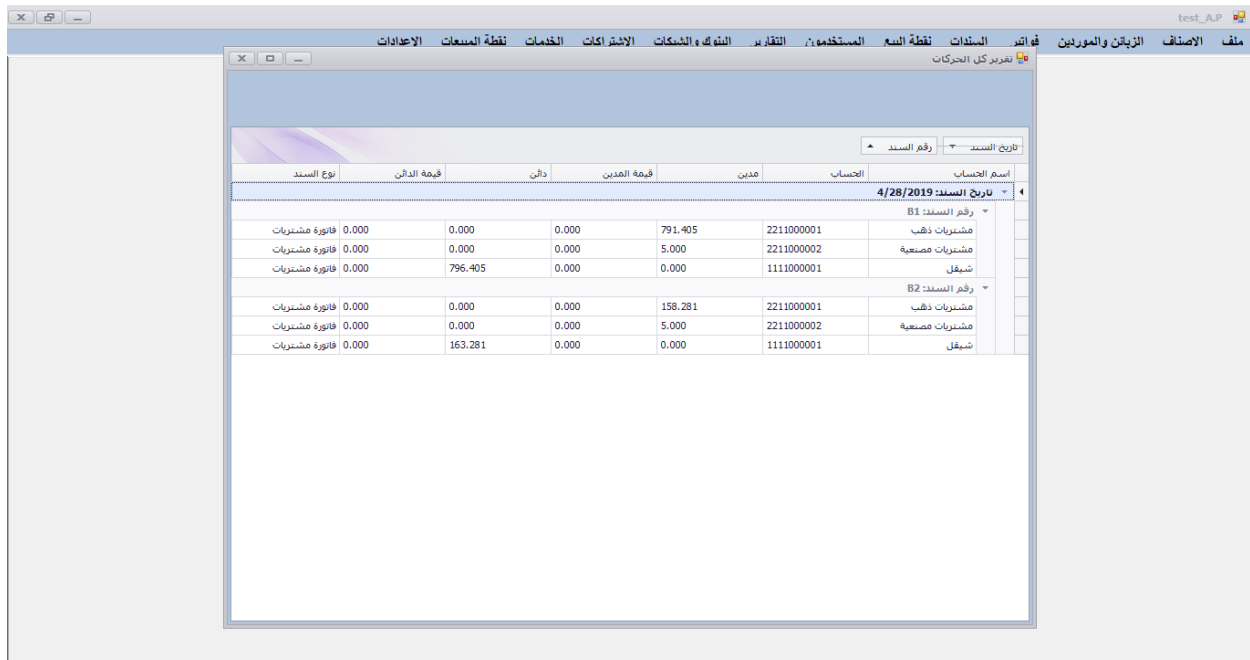


Figure 34-retrieve all processes that have been done before

Chapter 6 Conclusion and Recommendation

In this project, we built an application that aims at organizing various processes in one place, in order to make it easier for user to be acquaintance of all these processes. This will benefit the owners of jewelry shops to not be afraid of any loss.

From this project, we gained an experience in C#, which is considered a new programming language for us.

One of the most achieved works is being familiar of DevExpress component to obtain a friendly user desktop application. We also improve our search skills, since we faced several problem that needs to solve.

Chapter 7 References

[1] devexpress.

[2] webopedia.

[3] sql server.