

Project Write Up

1. Executive Summary

a. Description of the business

Chao's Asian Market is a very well known Asian grocery store in Provo and the surrounding area. They have a large base of loyal customers. Two months ago, Chao's moved to Orem and the owner sold the store to Ernie Silva. The new location that they currently moved too also belongs to Ernie; this location is 4 times larger than the old store was in Provo. However, the old owner didn't have any systems to calculate inventory in stock, keep track of expired items, and other supply chain knowledge.

One day as I shopping in the store after they moved to Orem, I noticed that the store's inventory was very bad. The old owner left a great amount of expired items and products that are not well displayed well at all in the store as they were just randomly placed on shelves.

Therefore, they hired me to control the inventory and make sure all the expired items are identified soon enough to apply discount as to avoid lost revenue due to spoilage. I have to remember and know where the products are in the store. However, the store is not fully expanded and I have brought in over 100 new products to meet customer demand. Remembering the prices and where they are all located is almost impossible. I have been thinking about these problems, looking for a more efficient way to manage the inventory since the first day I started working. I have made this VBA program as an answer to the problems of the store.

b. Overview of the program I built

I have showed the program to Ernie and explained him in details of how it works. He told me, "Lily, I'm impressed! Great job!" He is very happy that the cashiers and any employees won't have to run and look for an item each time they can't find out how much it costs when the customers check out. It will save time for both the customers and cashiers. One of the functions of the program is to find the products; it will also indicate the mark up price, the aisle number, and the arrival date. The arrival date is very important since Chao's carries some unique Asian products, which we can only order once every two weeks. Therefore, the customers will ask for the scheduled arrival date.

As Ernie said, "It's very good that now we can control the inventory and see what will expire soon so we can apply the proper discount." I have built another function in the program, as the users click on expired in "<= 1 month" or "2 month", etc. It will let us see the products which

about to expire at those times, as it loops through the worksheet, comparing the current date and the expiration date. It uses the expiration date minus current date to get the days before expiration to track the inventory. The users also can look at the aisle number to take the products out as they are expired. After that, user can press the “Create Worksheet” button to create a new worksheet which only has the items they need to find (expired, <= 1 month, 2 month, 3 month, 4 month, 5 month). It is very helpful for the employees at Chao’s so they can manage the inventory totaling than 1500 products.

Chao’s Asian Market uses Outlook for emailing and they can also control their employee’s emails. I searched online for a way to automatically open Outlook and send the new expired items worksheet to whoever is in charge of recording and taking care of the expired items. Outlook will put the new worksheet into an email and send it automatically to notify that person.

2. Implementation Documentation

In order to complete this project, I determined and mapped out the 6 elements that this program needed. Below will be the list and their purposes:

- **Product Information:** Gathers all the product information and put in into a new worksheet
- **User Form:** Allows the users to find product information and find the expired products
- **Message Box:** Gives notice users that there are no more products to be found, so they won’t keep pressing the button
- **Generate new worksheet:** Allows the user to separate the important information to another worksheet to analyze
- **Sending message through Outlook:** Allows the users to send the new worksheet from “generate new worksheet” to the procurement quality specialist.
- **Ribbon Control:** Allows the users to run the program without accessing the code. This function helps the user from editing any code or function. It also helps the program to look more professional

I will discuss the implementation of each of these functions in detail in following:

Product Information

In order to gather product information for my database, I have to go to Chao’s to take pictures of product name, expiration date, brand, aisle number, price, country, and other information. I thought it would not take too long. But as I talked to the owner, he would like to see products with as much information as possible. So it turned out that it took much more time than I was expecting. I choose random products from the shelf to put in the database because it would take around 60 hours to get information from all the current inventory in the store.

	A	B	C	D	E	F	G	H	I	J	K
1	Products	Amount	Price	Price Ma	Description	Expiration Date	Categories	Country	In Stock	Arrival Date	Isle
2	Datu Puti White Vinegar	6	2.10	3.15	Spiced Sukang Maasim		Vinegar	Phillipine	Yes		1
3	Silver Swan Coconut Vinegar	6	1.75	2.63			Vinegar	Phillipine	Yes		1
4	Vinegar	6	1.89	2.84	Greatwall Brand		Vinegar	China	Yes		1
5	Rice Vinegar	6	1.65	2.48	Swatow		Vinegar	China	No	12/15/2012	1
6	Apple Vinegar	8	3.15	4.73	500 ml		Vinegar	Korea	Yes		1
7	Sweet Soy Sauce	6	1.15	1.73	kecap manis	12/11/2012	Soy Sauce	China	Yes		2
8	Toyomansi	4	2.25	3.38	Soy sauce with Calamansi		Soy Sauce	China	Yes		2
9	Marca Pina	6	1.79	2.69			Soy Sauce	Phillipine	Yes		2
10	Kim Lan Soy Paste	6	2.95	4.43	Original		Soy Paste	China	Yes		2
11	Kim Lan Chili Soy Paste	6	2.95	4.43	Hot		Soy Paste	China	No	12/15/2012	2
12	Kim Lan Soy Sauce	6	2.60	3.90	Less Salt	12/21/2013	Soy Sauce	China	Yes		2
13	Kim Lan Lou Chau Soy Sauce	6	2.65	3.98	Dark	1/13/2015	Soy Sauce	China	Yes		2
14	Kim Lan Sang Chau Soy Sauce	6	2.99	4.49	Light	2/21/2014	Soy Sauce	China	Yes		2
15	Kadoya	6	8.95	13.43	654 ml		Sesame Oil	Japan	Yes		2
16	Olive Oil	8	5.95	8.93	extra virgin		Olive Oil	America	No	12/21/2012	3
17	Corn Malt Syrup	8	3.15	4.73			Syrup	China	Yes		3
18	Black Bean Sauce	10	2.95	4.43	Lee Kum Kee Brand	11/17/2013	Cooking Sauce	China	Yes		2
19	Kung Pao Stir-Fry Sauce	8	3.85	5.78	Lee Kum Kee Brand	11/15/2013	Cooking Sauce	China	Yes		2
20	Vegetarian Mushroom Flavored Stir-Fry Sauce	8	3.65	5.48	Lee Kum Kee Brand	3/1/2015	Cooking Sauce	China	Yes		2
21	Banana Sauce	10	1.45	2.18	Jufran Brand		Dipping Sauce	China	Yes		2
22	Sweet Chili Sauce	6	2.95	4.43	Mae Ploy Brand for chicken	3/27/2014	Dipping Sauce	Thailand	Yes		2
23	Sweetened Chili Sauce	6	3.35	5.03	For Springroll		Dipping Sauce	Thailand	Yes		2
24	Hot Chili Sauce	8	2.75	4.13	Sriracha Brand	9/1/2013	Dipping Sauce	Vietnam	Yes		2
25	Chili Garlic Sauce	12	2.15	3.23	Sriracha Brand		Dipping Sauce	Vietnam	Yes		2

Implementation 1: Product Information

User Form

From what I learned from Project 4, creating the userform wasn't something too difficult to do. I created 2 user forms: (1) frmProductFind and (2) frmExpire.

1. frmProductFind (Implementation 2.1A)

The user can type in part of or all of the product name to find the price, asile number, row ID#, and the arrival date. After that, user can click on find next, to find the next suitable product. The message box will pop up when there are no more products. The hardest thing in this userform is creating the ComboBox for Categories. A loop runs through the database in categories column (Implementation 2.1B). All the data will be record to a new worksheet. Then, the code will remove the duplicate category's names as there are many products in the same category. After that, it will sort the names into alphabetical order and add the value to the ComboBox. The activesheet will then be deleted.

3.15	Spiced Sukang Maasim
2.63	
2.84	
2.48	
4.73	
1.73	
3.38	
2.69	
4.43	
4.43	
3.90	
3.98	
4.49	
13.43	
8.93	extra virgin
4.73	

Find Product Information X

All or Part of Product Name <input type="text" value="kim"/>	Categories <input style="border: none; border-bottom: 1px solid gray; background-color: #f0f0f0; padding: 2px 5px;" type="text" value="Soy Paste"/> ▼
Product Name <input type="text" value="Kim Lan Soy Paste"/>	Price <input type="text" value="4.425"/>
Isle Number <input type="text" value="2"/>	Row ID # <input type="text" value="10"/>
Arrival Date <input type="text" value="Arrived"/>	

Implementation 2.1A: User Form frmProductFind

```

Private Sub UserForm_Initialize()

    Dim src As Range
    Dim dest As Range

    Set src = Range(Sheets("Chao's").Range("g64").End(xlUp), Sheets("Chao's").Range("g2"))

    Application.ScreenUpdating = False
    Sheets.Add
    Set dest = Range(Range("a2"), Cells(src.Cells.Count, 1))
    dest.Value = src.Value
    dest.RemoveDuplicates Columns:=1, Header:=xlNo
    ActiveSheet.Sort.SortFields.Clear
    ActiveSheet.Sort.SortFields.Add Key:=Range("A2"), SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:=xlSortNormal
    ActiveSheet.Sort.SetRange dest
    ActiveSheet.Sort.Header = xlNo
    ActiveSheet.Sort.MatchCase = False
    ActiveSheet.Sort.Orientation = xlTopToBottom
    ActiveSheet.Sort.SortMethod = xlPinYin
    ActiveSheet.Sort.Apply

    cboCat.AddItem ""
    For Each cell In Range(Range("a2"), Range("a2").End(xlDown))
        cboCat.AddItem cell.Value
    Next
    Application.DisplayAlerts = False
    ActiveSheet.Delete
    Application.DisplayAlerts = False
    Application.ScreenUpdating = True

    rowNum = 2
End Sub

```

Implementation 2.1B: Creating Categories ComboBox

2. frmExpire (Implementation 2.2)

For this userform, I use CheckBox for selection. When user selects “2 month” and clicks “Find Product” CommandButton, the program will loop through all the Expiration Date Column. Then, I use the formula “If Sheets("Chao's").Cells(rowNum, 6).Value <> "" And 31 <= Sheets("Chao's").Cells(rowNum, 6).Value - Date And Sheets("Chao's").Cells(rowNum, 6).Value - Date <= 60 Then” to calculate the number of days before the product will expire. This function also applies to other CheckBox. The “Find Next” CommandButton is use to find the next product which will expire in that time range. The user also sees the aisle number that is very helpful to track, apply discounts, or get rid of the items. Just like the frmProductFind, when there are no more products, the message box will prompt the user that there are no more products. There is also a “Create Worksheet” button, which I will explain more in the next paragraph.

2.48	Swatow		
4.73	500 ml		
1.73			11/201
3.38			
2.69			
4.43			
4.43			
3.90			21/201
3.98			13/201
4.49			21/201
13.43			
8.93			
4.73			17/201
4.43			15/201
5.78			
5.48	LEE KUM KEE BRAND		3/1/201
2.18	Jufran Brand		
4.43	Mae Ploy Brand for chicken		3/27/201

Implementation 2.2: User Form frmExpire

Message Box

The Message Box of “No More Products” (Implementation 3.1) will pop up when there are no more products to be found when the user clicks “Find Next”. It will also help the users to know that there are no more and to stop pressing the button. As the program loops through the database and can’t find any more products, I wrote the code for message box as seen below when “found=false”



Implementation 3. 1: MessageBox “ No More Products!”

Generate New WorkSheet

From the user form frmExpire, I created a CommandButton “Create Worksheet”. The code will loop through Chao’s worksheet, find all the products that fit in the time frame, then it will transfer each value from the Row ID to the new worksheet and name the worksheet, such as “Expire in 5 months (Implementation 4.1). After that, the name of the worksheet is recorded with string “h” and recorded in Sheet3 (which I hide from the worksheets) (Implementation 4.2). The variable “h” will be used in my “SendMessage” Module

A	B	C	D	E	F	G	H	I	J	K
Products	Amount	Price	Price Mark Up	Description	Expiration Date	Categories	Country	In Stock	Arrival Date	Isle
Premium	20	4.7	7.05	Kam Yen Jan Brand. Made with Pork and Chicken	4/6/2013	Sausage	America	Yes		6

Expired Product

Expired In

 Expired
 <= 1 month
 2 month
 3 month
 4 month
 5 month

Product Name

Isle Number Row ID #

Expired in (number of days)

Implementation 4.1: Creating a new WorkSheet

```

If chk5month = True Then
Worksheets.Add.Name = "Expire In 5 Months"
x = 2
Do Until Sheets("Chao's").Cells(rowNum, 1) = ""
  If Sheets("Chao's").Cells(rowNum, 6).Value <> "" And 121 <= Sheets("Chao's").Cells(rowNum, 6).Value - Date And Sheets("Chao's").C
    For y = 1 To 11
      Sheets("Expire In 5 Months").Cells(1, y) = Sheets("Chao's").Cells(1, y)
      Sheets("Expire In 5 Months").Cells(x, y) = Sheets("Chao's").Cells(rowNum, y).Value
    Next
    x = x + 1
  End If
  rowNum = rowNum + 1
  DoEvents
  DoEvents
Loop
End If

h = ActiveSheet.Name
Sheet3.Cells(1).Value = h
End Sub

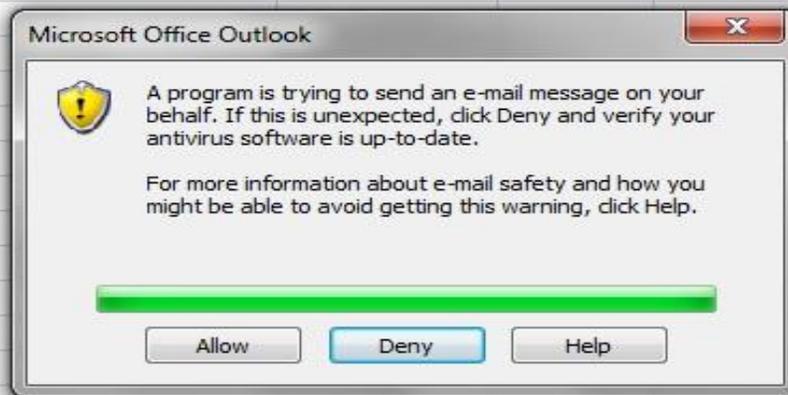
```

Value of variable h is saved in sheet3 to use in "SendMessage" Module

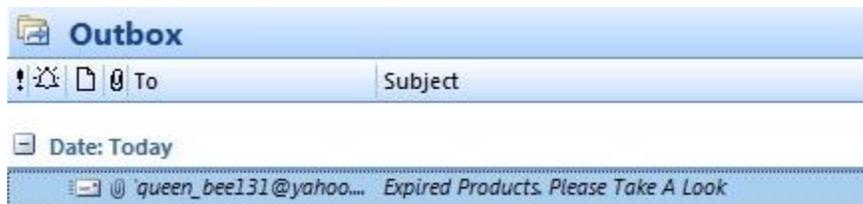
Implementation 4.2: Recording variable h to Sheet3 for "SendMessage" Module

Sending Message Through Outlook

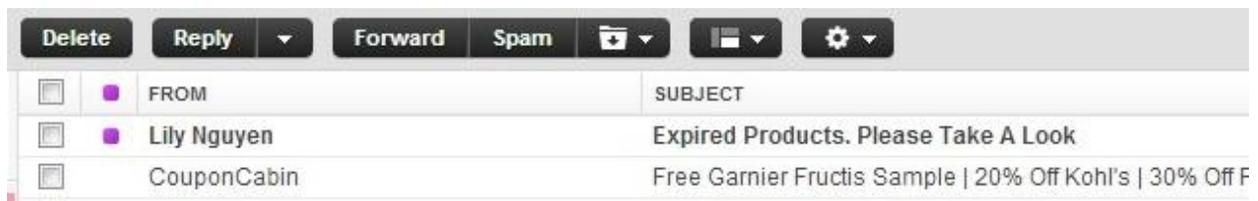
This is the part where I learned the most. I looked online to find how to send an email to Outlook then it can send the email automatically to the Quality Specialist at Chao's. As I pressed the "Send" in the ribbon, there will be a pop up from Outlook (Implementation 5.1). The user clicks on "Allow" this will allow the program to send only that worksheet to their outbox in Outlook. I already put a title for the email and email address of the receiver (Implementation 5.2). The receiver will receive the email automatically from Outlook (Implementation 5.3).



Implementation 5.1: Opening Outlook



Implementation 5.2: Sent to Outbox of Outlook



Implementation 5.3: Email received the Message from Outlook

Here is my code:

```
Dim FileExtStr As String
Dim FileFormatNum As Long
Dim Sourcewb As Workbook
Dim Destwb As Workbook
Dim TempFilePath As String
Dim TempFileName As String
Dim sh As Worksheet
Dim TheActiveWindow As Window
Dim TempWindow As Window
Dim I As Long
Dim g As String
    g = Sheet3.Cells(1).Value

With Application
    .ScreenUpdating = False
    .EnableEvents = False
End With

Set Sourcewb = ActiveWorkbook

With Sourcewb
    Set TheActiveWindow = ActiveWindow
    Set TempWindow = .NewWindow
    .Sheets(g).Copy
End With
TempWindow.Close

Set Destwb = ActiveWorkbook
With Destwb
    If Val(Application.Version) < 12 Then
        FileExtStr = ".xls": FileFormatNum = -4143
    Else

        If Sourcewb.Name = .Name Then
            With Application
                .ScreenUpdating = True
                .EnableEvents = True
            End With
            MsgBox "Your answer is NO in the security dialog"
            Exit Sub
        Else
            Select Case Sourcewb.FileFormat
            Case 51: FileExtStr = ".xlsx": FileFormatNum = 51
            Case 52:
                If .HasVBAProject Then
                    FileExtStr = ".xlsm": FileFormatNum = 52
                Else
                    FileExtStr = ".xlsx": FileFormatNum = 51
                End If
            Case 56: FileExtStr = ".xls": FileFormatNum = 56
            Case Else: FileExtStr = ".xlsb": FileFormatNum = 50
            End Select
        End If
    End If
End With

With Destwb
    .SaveAs TempFilePath & TempFileName & FileExtStr, _
        FileFormat:=FileFormatNum
    On Error Resume Next
    For I = 1 To 3
        .SendMail "queen_bee131@yahoo.com", _
            "Expired Products. Please Take A Look"

        If Err.Number = 0 Then Exit For
    Next I
    On Error GoTo 0
    .Close SaveChanges:=False
End With

Kill TempFilePath & TempFileName & FileExtStr

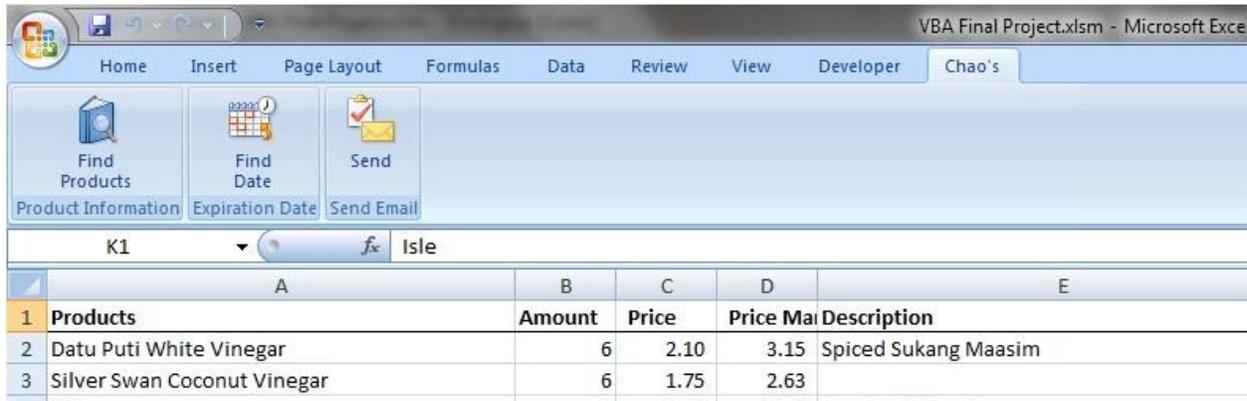
With Application
    .ScreenUpdating = True
    .EnableEvents = True
End With

End Sub
```

Ribbon Control

I used the ribbon because I like how professional it looks (Implementation 6). It helps the users to avoid changing any of the code that I programmed by accident. I created three buttons in a tab that I named "Chao's". Those three buttons are Find Products, Find Date, and Send

- Find Products will run the frmProductFind
- Find Date will run the frmExpire
- Send will run module2 to SendMessage by using Outlook



Implementation 6: Using Ribbon Control

3. Lessons Learned

This has been a great project for me to go through. I like how I had to build everything from the scratch, which was very different from the normal class projects. Because I'm a new programmer, I learned new things in every step I took to build my code. There are some main concepts that I gained from doing this project:

- Always look for the solution before asking the experts because I know that I learned much more from being able to struggle through it. This style of learning also helps me to remember the concepts better.
- VBA Forums were my best resource to look for any answers
- The main problem I was facing was how to run the Combo Box and how to add the value to the box. But from looking at the last project I completed and from researching online, I was able to understand and apply it successfully in my project. I finally feel confident in my ability to work in VBA and the various techniques that we taught in the class.
- Another problem that I was facing was how to create a ribbon. It was taught too fast in the class and I couldn't follow. I ask you again and the answer was to look at the website you recommended which I totally forgot about. I read the website thoroughly and followed the instructions step by step. I created the ribbon and

also improved it by downloading the 2007OfficeIconsGallery to get some good looking icons

- The biggest problem I was facing was how to send an email using VBA. You showed us how to send message and I thought I should do that to send message to the Quality Specialist about which products are about to expire soon or already expired. But after talking to you, I thought I needed to do some thing more creative and more helpful for my employer. So, I tried the code from the class for sending a text message, but it didn't work. So I went online and learned different ways to send a worksheet in an email. It was very challenging but I learned so much and actually really enjoyed it

4. Assistance

I got some code from online sources to help me in sending a worksheet to an email. I also ask Dr. Allen when I really struggled and could not run the program after 2-3 hours of trying. As I remember, he gave me clear directions and the resources to complete the code on my own.