James Omerza
MBA 614 Final Project

Executive Summary

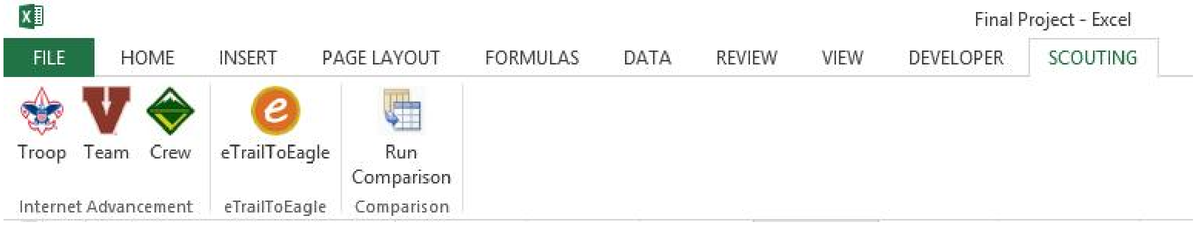
As the Advancement Chair for my Boy Scout Troop, I am responsible for tracking merit badges, rank advancements, courts of honor, leadership positions etc. I currently use three different systems to track this information: (1) etrailtoeagle.com, (2) BSA Internet Advancement, and (3) Excel. The website etrailtoeagle.com is a third-party subscription site that parents, leaders and committee members can access and edit. This is a comprehensive site that tracks almost everything that happens in the troop. Unfortunately, it only allows the user to see one scout's information at a time unless you print a report that is not very easy to read or manipulate. The BSA Internet Advancement is the Boy Scouts of America official advancement site that only the Advancement Chair has access to. The only thing that is entered on this site is rank advancements and merit badges and their respective dates earned. This information is transmitted to the local Boy Scout Council. Finally, in order to view and track all of this information in one location in an easy to read format, I create a separate Excel file that contains a summary of all of the advancement information. Each time I receive advancement information from one of the scouts, I have to update the information in these three places. In addition, because parents and leaders can access and edit etrailtoeagle.com, sometimes merit badges and advancement has been recorded, but I am unware of the change. This problem is faced by almost every Advancement Chair that uses etrailtoeagle.com.

My VBA solution focuses on the scout merit badges and performs three basic steps:

1. Pull and format merit badge information from Internet Advancement
    a. Login and pull Troop Information
    b. Login and pull Team Information
    c. Login and pull Crew Information
2. Login, pull and format merit badge information from eTrailToEagle.com
3. Produce a comparison report

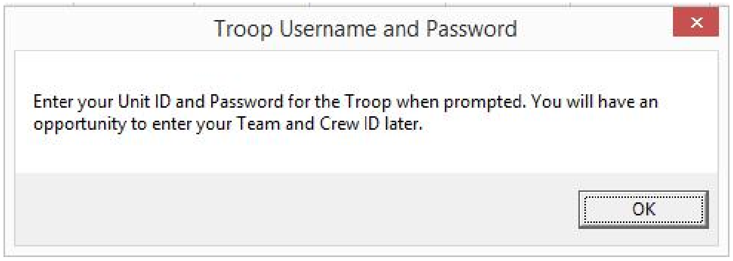
Implementation Documentation

To make the file user friendly, a new tab on the ribbon was created with a button for each of the operations.



1. Pull and format merit badge information from Internet Advancement

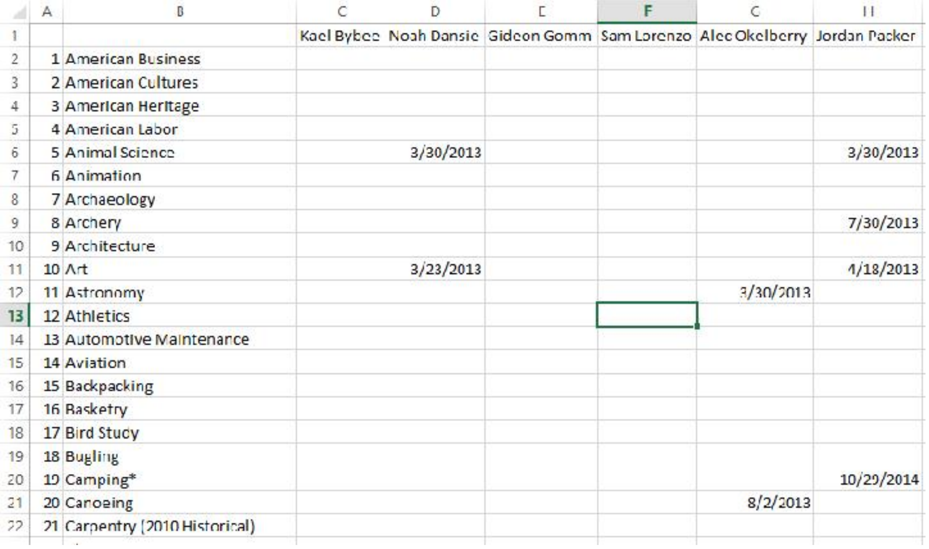
Because Internet Advancement tracks Troop, Team and Crew information separately, a separate username and password must be entered onto the website for each in order for the user to access all information. When the user clicks the Troop button, they are prompted to sign in using the troop username and password first with the following message box:



Troop Username and Password

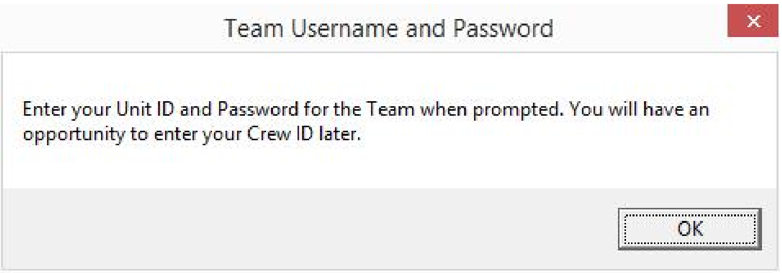
Enter your Unit ID and Password for the Troop when prompted. You will have an opportunity to enter your Team and Crew ID later.

Once the user has clicked the "OK" button, Internet Explorer is opened, made visible and is directed to the Internet Advancement login page. Once the user has signed in, the program automatically manipulates the page through object clicks to get to the first scout's list of merit badges starting with the letters A through C. The program creates a new tab with the name "TroopCouncilRecords" after deleting any files with that name already appearing in the workbook. The program then goes through the HTML file to pull the first scout's dates earned for merit badges A-C while creating a list of the numbered merit badges on the "TroopCouncilRecords" sheet. The program then clicks to the next list of merit badges D-K and performs the same task. This is repeated until all of the dates for the first scout have been placed onto the sheet next to the proper merit badge name. The program finds the name of the first scout and places it at the top of the sheet in the proper column. The program then runs this same set of procedures for each of the scouts listed on the site. The program counts the number of scouts on the site in order to do this. The program then performs an autofit for the rows and columns for this sheet. The result is a list of scouts at the top of the columns and list of merit badges along the rows as shown below:



	A	B	C	D	E	F	G	H
1			Kael Bybee	Noah Dansie	Gideon Gomm	Sam Lorenzo	Alec Okelberry	Jordan Packer
2	1	American Business						
3	2	American Cultures						
4	3	American Heritage						
5	4	American Labor						
6	5	Animal Science		3/30/2013				3/30/2013
7	6	Animation						
8	7	Archaeology						
9	8	Archery						7/30/2013
10	9	Architecture						
11	10	Art		3/23/2013				4/18/2013
12	11	Astronomy					3/30/2013	
13	12	Athletics						
14	13	Automotive Maintenance						
15	14	Aviation						
16	15	Backpacking						
17	16	Basketry						
18	17	Bird Study						
19	18	Bugling						
20	19	Camping*						10/29/2014
21	20	Canoeing					8/2/2013	
22	21	Carpentry (2010 Historical)						

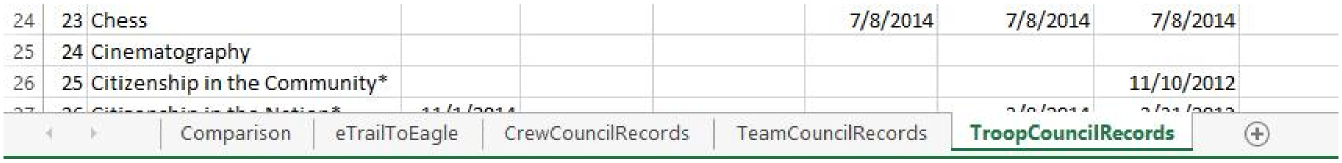
When the user clicks on the Team button, the program prompts the user to enter their Team username and password with a message box as shown below:



Team Username and Password

Enter your Unit ID and Password for the Team when prompted. You will have an opportunity to enter your Crew ID later.

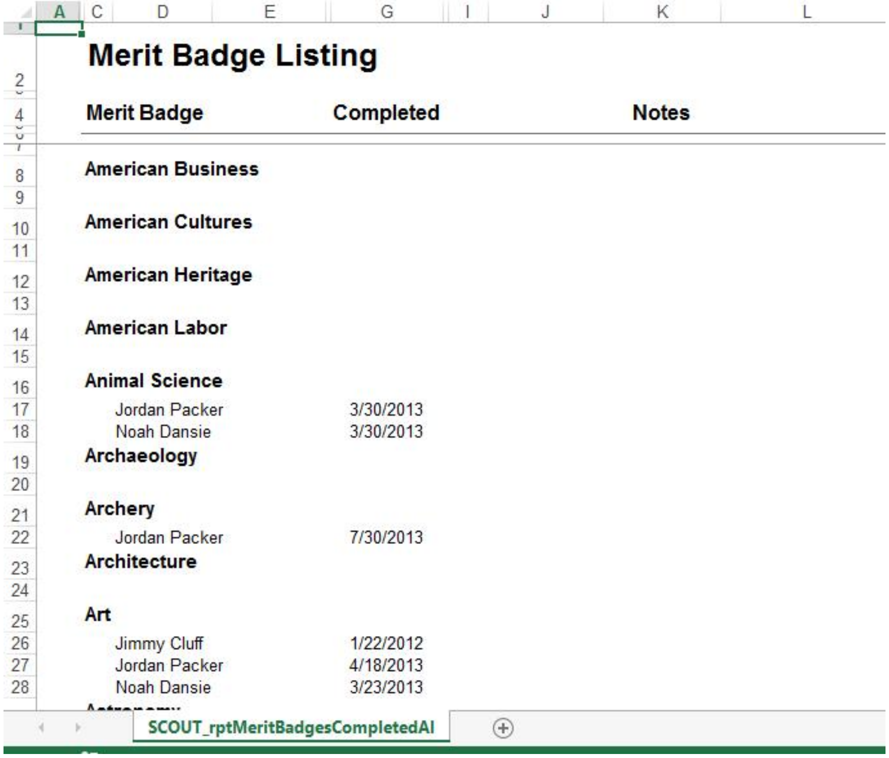
The above process for the Troop is then repeated for the Team and Crew. The result is three tabs in the workbook for the Crew, Team and Troop (in that order) as shown below:



24	23 Chess				7/8/2014	7/8/2014	7/8/2014
25	24 Cinematography						
26	25 Citizenship in the Community*						11/10/2012
27	26 Citizenship in the Nation*	11/1/2014				2/8/2014	2/21/2013

2. Pull and format merit badge information from eTrailToEagle.com

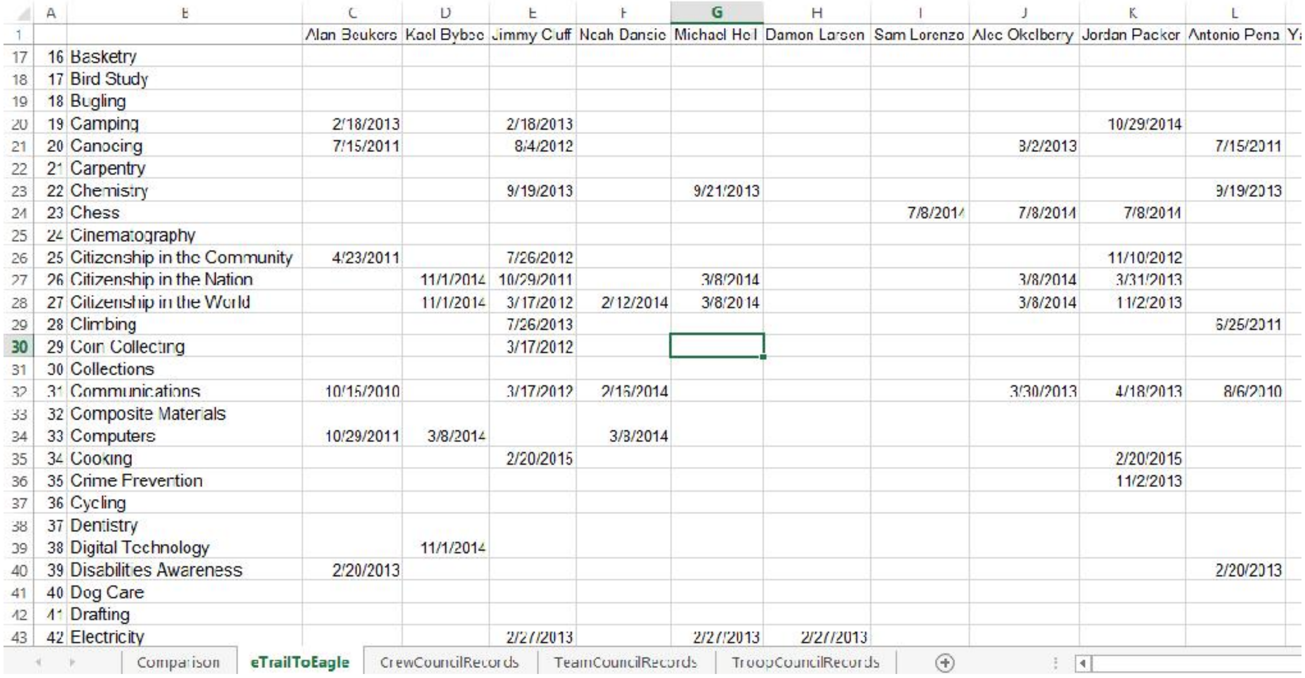
When the user clicks on the eTrailToEagle button, the user is directed to the eTrailToEagle.com login page. Once the user has signed in with their username and password, a series of object clicks will download an Excel file that has all of the merit badge information for all scouts in the Troop, Team and Crew. This list is not user friendly. A sample of the raw report is shown below:



Merit Badge Listing

Merit Badge	Completed	Notes
American Business		
American Cultures		
American Heritage		
American Labor		
Animal Science		
Jordan Packer	3/30/2013	
Noah Dansie	3/30/2013	
Archaeology		
Archery		
Jordan Packer	7/30/2013	
Architecture		
Art		
Jimmy Cluff	1/22/2012	
Jordan Packer	4/18/2013	
Noah Dansie	3/23/2013	
Astronomy		

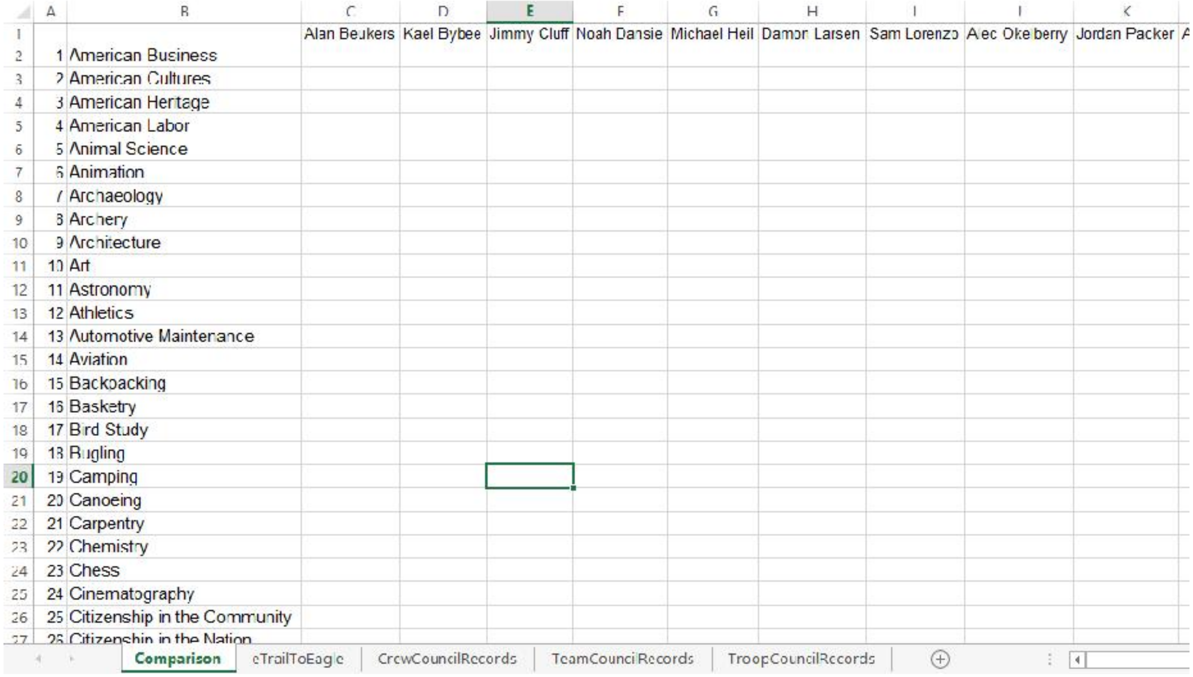
The program then automatically formats this report first by unmerging all of the cells, then deleting unnecessary rows and columns. The program creates a new sheet, copies the list of scouts and pastes them onto the new sheet. It then removes the duplicates since each scout is listed more than once on the main sheet. The program sorts the scouts in alphabetical order and then pastes their names back at the top of the main page, transposing them to the columns. The program then trims the names so they will be comparable to the CouncilRecords names. Once the names are at the top of the page, the program runs through a series of "For-Next" and "If-End If" statements to place the dates of each scout's merit badge date earned next to the proper merit badge and under the proper scout. The program then deletes the old names and dates which are now unnecessary, renames the sheet to "eTrailToEagle" and removes the bold from the list of merit badges. There are some errors on the eTrailToEagle.com report that needed to be fixed. For instance, Internet Advancement has merit badges with the names "Animation", "Cinematography", and "Signs, Signals, and Codes" that are not listed on the eTrailToEagle.com report. Also, the eTrailToEagle.com report lists two merit badges twice. The report is adjusted to add the additional badges and delete the duplicates. Numbers are then inserted next to each merit badge name and the sheet is autofitted. The program then moves the sheet to the main project file after deleting any existing sheet by the same name. The report is now formatted exactly the same as the CouncilRecords sheets as shown below:



	A	B	C	D	E	F	G	H	I	J	K	L
1			Alan Beukers	Kael Bybee	Jimmy Cuff	Noah Dancic	Michael Hel	Damon Larsen	Sam Lorenzo	Alec Okelbery	Jordan Packer	Antonio Pena
17	16	Basketry										
18	17	Bird Study										
19	18	Bugling										
20	19	Camping	2/18/2013		2/18/2013						10/29/2014	
21	20	Canoeing	7/15/2011		8/4/2012					3/2/2013		7/15/2011
22	21	Carpentry										
23	22	Chemistry			9/19/2013		9/21/2013					9/19/2013
24	23	Chess							7/8/2014	7/8/2014	7/8/2014	
25	24	Cinematography										
26	25	Citizenship in the Community	4/23/2011		7/26/2012						11/10/2012	
27	26	Citizenship in the Nation		11/1/2014	10/29/2011		3/8/2014			3/8/2014	3/31/2013	
28	27	Citizenship in the World		11/1/2014	3/17/2012	2/12/2014	3/8/2014			3/8/2014	11/2/2013	
29	28	Climbing			7/26/2013							6/25/2011
30	29	Coin Collecting			3/17/2012							
31	30	Collections										
32	31	Communications	10/15/2010		3/17/2012	2/16/2014				3/30/2013	4/18/2013	8/6/2010
33	32	Composite Materials										
34	33	Computers	10/29/2011	3/8/2014		3/3/2014						
35	34	Cooking			2/20/2015						2/20/2015	
36	35	Crime Prevention									11/2/2013	
37	36	Cycling										
38	37	Dentistry										
39	38	Digital Technology		11/1/2014								
40	39	Disabilities Awareness	2/20/2013									2/20/2013
41	40	Dog Care										
42	41	Drafting										
43	42	Electricity			2/27/2013		2/27/2013	2/27/2013				

3. **Produce a comparison report**

When the user clicks on the Run Comparison button, the program creates a "Comparison" sheet after deleting any existing sheet with the same name. The program copies the list of merit badges and scouts from the eTrailToEagle sheet and runs a comparison between the eTrailToEagle sheet and each of the CouncilRecords sheets using a series of "For-Next" and "If-End If" statements. The report generated will show any dates from the eTrailToEagle sheet that do not match the CouncilRecords sheets. If no council updates are needed, the Comparison sheet should be blank as shown below:



	A	B	C	D	E	F	G	H	I	J	K
1			Alan Beukers	Kael Bybee	Jimmy Cluff	Noah Dansie	Michael Heil	Damon Larsen	Sam Lorenzo	Alec Okelberry	Jordan Packer
2	1	American Business									
3	2	American Cultures									
4	3	American Heritage									
5	4	American Labor									
6	5	Animal Science									
7	6	Animation									
8	7	Archaeology									
9	8	Archery									
10	9	Architecture									
11	10	Art									
12	11	Astronomy									
13	12	Athletics									
14	13	Automotive Maintenance									
15	14	Aviation									
16	15	Backpacking									
17	16	Basketry									
18	17	Bird Study									
19	18	Bugling									
20	19	Camping									
21	20	Canoeing									
22	21	Carpentry									
23	22	Chemistry									
24	23	Chess									
25	24	Cinematography									
26	25	Citizenship in the Community									
27	26	Citizenship in the Nation									

Discussion of Learning and Conceptual Difficulties Encountered

1. **Internet Explorer Manipulation**: I had difficulty understanding how to navigate to websites, login and click on buttons on the page. This was especially challenging because inspecting the elements on the page does not show the actual identification of the object on the page. I had to use the professor's "a.showsource" numerous times in order to see the actual code written for the page to get the proper identification. There were also significant challenges understanding how to interpret the HTML code and use the professor's "a.moveTo", "a.moveBackTo", "a.getText", etc. One particularly difficult part of reading the HTML code was counting the number of scouts on the page. I wrote code to loop through the HTML to check how many scouts were on the page. This may sound easy, but finding unique identifiers is difficult.
2. **Downloading Excel File from Internet Explorer**: I spent many hours trying to figure out how to open a downloaded Excel file from Internet Explorer. There were two issues. First, I could not write VBA code to click the "Open" button to download the file. I discussed this with the professor and discovered I could use the "SendKeys" method to send key strokes to the computer. The second issue was that if the VBA code was running, the file would not open. Once VBA stopped running, the file would open. In order to overcome this, I discussed it with the professor and he suggested the use of the method "Application.OnTime". I had to split my

subprocedure into two and run the second half once the file was downloaded and the first subprocedure finished using the “Application.OnTime” method.

3. **Formatting Issues**: I experienced many formatting issues because I wanted the reports to look a certain way. For instance, I wanted the list of scouts to be in alphabetical order. In order to do this, I had to split the cells using “TextToColumns”, sort the scouts by last name and then concatenate the names back together. I also had to figure out how to use autofit for columns and rows and how to display the gridlines. I wanted the merit badges to be numbered. I had to move a sheet from one workbook to another. Probably the most difficult task related to formatting was placing the merit badge dates earned in the proper place on the eTrailToEagle sheet. I had to play with multiple variables and cell offsets to get just what I needed. I also had to change the format of the cell to “NumberFormat” to be comparable to the other reports.
4. **Ribbon Issues**: The CustomUIEditor and the professor’s Ribbon Wizard have difficulty dealing with updates. For instance, I originally had one button to run through each of the procedures in sequence. When I presented my project to the class on the last day of class, the professor suggested that I add separate buttons for each procedure. The CustomUIEditor and the Ribbon Wizard both had difficulty running this update. I had to create a brand new file and paste all of the code to it and then create a new ribbon. This was time consuming and difficult.

Note: There were no issues that I did not solve with the help of the professor and the internet. Most of the learning occurred because I was forced to apply what we had learned in class (do loops, for loops, if statements, deleting sheets, turning off display alerts, counting cells with data, converting names using LCase for comparison purposes, deleting rows, autofitting, etc. I learned that in order to get through the more difficult concepts, asking someone or checking the internet helps.

Assistance

I did not have any substantial assistance from another person outside of the professor. I went to see the professor in advance of our in-class discussion of the “agent” class module so I could get a jump start on my project. He spent approximately one hour helping me understand what would later be covered in class. The professor also gave me tips on what I could do to overcome some of the difficulties mentioned in the previous section.

Write-Up Detail

The full VBA code for my project is below with descriptions of the purpose of each section.

MODpullAllInfo

```
Option Explicit

Sub pullAllInfo()

getAllCouncilData
downloadETTEData
comparisonSheet

End Sub
```

I created multiple modules with the prefix “MOD” to make the code more user friendly to navigate. This module pulls from each of the other modules to run the full program. Since each subprocedure has its own button, this module is obsolete. However, I kept it in for my own purposes because I like having one button.

MODgetAllCouncilData

```
Option Explicit
Dim a As New agent
Dim url As String
Dim pos  As Long
Dim award As String
Dim awardDate As String
Dim x As Integer
Dim y As Integer
Dim ws As Worksheet
Dim fullName As String
Dim n As Integer
Dim s As Integer
Dim numberOfScouts As Integer
```

This sets module level variables for the subs in the module.

```
Sub getAllCouncilData()
  getTroopCouncilData1
  getTeamCouncilData1
  getCrewCouncilData1
End Sub
```

Because the user has to login three separate times, I separated the three sets of logins with three different subs. These are three separate buttons on the page and this subprocedure will never run. However, I have left it in because I prefer having one button.

```
Sub getTroopCouncilData1()
MsgBox ”Enter your Unit ID and Password for the Troop when prompted. You will have an
opportunity to enter your Team and Crew ID later.”, vbOKOnly, ”Troop Username and
Password”
```

This message box tells the user to enter the user ID and password for the Troop first.

```
Application.DisplayAlerts = False
For Each ws In Worksheets
    If LCase(ws.name) = LCase(”TroopCouncilRecords”) Then
        ws.Delete
    End If
Next
Application.DisplayAlerts = True
Worksheets.Add().name = ”TroopCouncilRecords”

getCouncilData

End Sub
```

This deletes the current TroopCouncilRecords sheet (if any) and adds a new one.

```
Sub getTeamCouncilData1()

MsgBox ”Please log out and enter your Unit ID and Password for the Team when prompted.
You will have an opportunity to enter your Crew ID later.”, vbOKOnly, ”Troop Username
and Password”

Do Until a.document.Location =
”https://scoutnet.scouting.org/IADV/ui/Login/Login.aspx”
  a.waitForLoad
Loop
```

Because the user has to enter a separate user ID and password for the Team, this pauses the program until the user has logged out.

```
Application.DisplayAlerts = False
For Each ws In Worksheets
    If LCase(ws.name) = LCase(”TeamCouncilRecords”) Then
        ws.Delete
    End If
Next
Application.DisplayAlerts = True
Worksheets.Add().name = ”TeamCouncilRecords”
```

This deletes the current TeamCouncilRecords sheet (if any) and adds a new one.

```
getCouncilData

End Sub

Sub getCrewCouncilData1()
```

Because the user has to enter a separate user ID and password for the Team, this pauses the program until the user has logged out.

```
MsgBox ”Please logout and enter your Unit ID and Password for the Crew when
prompted.”, vbOKOnly, ”Troop Username and Password”

Do Until a.document.Location =
”https://scoutnet.scouting.org/IADV/ui/Login/Login.aspx”
  a.waitForLoad
Loop

Application.DisplayAlerts = False
For Each ws In Worksheets
    If LCase(ws.name) = LCase(”CrewCouncilRecords”) Then
        ws.Delete
    End If
Next
Application.DisplayAlerts = True
Worksheets.Add().name = ”CrewCouncilRecords”
```

This deletes the current CrewCouncilRecords sheet (if any) and adds a new one.

```
getCouncilData

End Sub

Sub getCouncilData()

url = ”https://scoutnet.scouting.org/IADV/ui/Home/Main.aspx”
a.visible = True
a.openpage url, True
Do Until a.document.Location = url
  a.waitForLoad
Loop
```

This navigates to the login page of the Council’s Internet Advancement site.

```
a.openpage ”https://scoutnet.scouting.org/IADV/ui/Home/LoadData.aspx”, True
a.document.all(”btnLoadCouncil”).Click
a.waitForLoad

x = 0
For s = 1 To 100
    If a.moveTo(”$BntRankG4”) Then
        x = x + 1
        numberOfScouts = x
    End If
Next
```

This counts the number of Scouts on the page by looking for the unique identifier “$BntRankG4”

```
a.document.all(”dgYouth4$ctl02$BntMeritG4”).Click
a.waitForLoad
```

This clicks on the merit badges for the first scout listed on the page for merit badges A - C

```
x = 0
a.position = 1
getAwardandDateEarned
getFullName1

a.document.all(”DKMerit1”).Click
getAwardandDateEarned
a.document.all(”LRMerit1”).Click
getAwardandDateEarned
a.document.all(”SZMerit1”).Click
getAwardandDateEarned
a.document.all(”btnCancel1”).Click
a.waitForLoad
```

This gets the date earned and list of merit badges for each of the separate pages that lists the merit badges. For instance, “DKMerit1” page lists all of the merit badges that start with the letters D – K.

```
For n = 4 To numberOfScouts + 2
```

```
    a.document.all(”dgYouth4$ctl0” & n - 1 & ”$BntMeritG4”).Click
    x = 0
    getDateEarned
    getFullName2
    a.document.all(”DKMerit1”).Click
    getDateEarned
    a.document.all(”LRMerit1”).Click
    getDateEarned
    a.document.all(”SZMerit1”).Click
    getDateEarned
    a.document.all(”btnCancel1”).Click
    a.waitForLoad
Next
```

This gets the dates earned for each of the additional scouts listed on the page.

```
a.document.all(”btnReset”).Click

Columns(”A:Z”).EntireColumn.autoFit
Rows(”1:1000”).EntireRow.autoFit
```

This autofits the rows and columns.

```
End Sub

Sub getAwardandDateEarned()
a.waitForLoad
a.position = 1
Do While a.moveTo(”id=’DateEarned”)
    pos = a.position
    a.moveBackTo ”<td>”
    award = a.getText(”</td>”)
    If Not a.moveTo(”type=””text”” value=”””) Then Stop
    awardDate = a.getText(””””)
    x = x + 1
    Cells(x + 1, 1).Value = x
    Cells(x + 1, 2).Value = award
    Cells(x + 1, 3).Value = awardDate
    a.position = pos
Loop
End Sub
```

This looks through the HTML for the name of the merit badge and the date earned for the first scout and places those in the proper location on the newly created sheet.

```
Sub getFullName1()
    a.position = 1
    a.moveTo (”FullName=”)
    fullName = a.getText(”&amp;RedirectUrl”)
    Cells(1, 3).Value = fullName
    Cells(1, 3).Value = Replace(Cells(1, 3), ”++”, ” ”)
End Sub
```

This looks through the HTML for the full name of the first scout and replaces the “++” that is in between the first and last name with “ “. Then it places the full name in the proper location on the newly created sheet.

```
Sub getDateEarned()
a.waitForLoad
a.position = 1
    Do While a.moveTo(”id=’DateEarned”)
        pos = a.position
        a.moveBackTo ”<td>”
        If Not a.moveTo(”type=””text”” value=”””) Then Stop
        awardDate = a.getText(””””)
         x = x + 1
        Cells(x + 1, n).Value = awardDate
        a.position = pos
    Loop
```

This looks through the HTML for the name of the date earned for the additional scouts and places those in the proper location on the newly created sheet.

```
End Sub

Sub getFullName2()
    a.position = 1
    a.moveTo (”FullName=”)
    fullName = a.getText(”&amp;RedirectUrl”)
    Cells(1, n).Value = fullName
    Cells(1, n).Value = Replace(Cells(1, n), ”++”, ” ”)
End Sub
```

This looks through the HTML for the full name of the additional scouts and replaces the “++” that is in between the first and last name with “ “. Then it places the full name in the proper location on the newly created sheet.

MODdownloadETTEData

```
Option Explicit
Dim a As New agent

Sub downloadETTEData()

Dim url As String
Dim wbk As Workbook

Application.Wait (Now + TimeValue(”00:00:03”))

url = ”https://etrailtoeagle.com/Welcome/Default.aspx”
a.visible = True
a.openpage url, True
Do Until a.document.Location = url
  a.waitForLoad
Loop
```

This navigates the Internet Explorer browser to the eTrailToEagle.com login site.

```
a.openpage
”https://etrailtoeagle.com/scouts/ReportCenter.aspx”, True
a.document.all(”ctl00_ContentPlaceHolder1_ASPxRoundPanel1_ReportsASPxNavBar_I2i0_T”).C
lick
a.waitForLoad
a.document.all(”ctl00_ContentPlaceHolder1_ASPxRoundPanel1_IReportFormatDropDownList”).
Value = ”Excel”
a.waitForLoad
a.document.all(”ctl00_ContentPlaceHolder1_ASPxRoundPanel1_RunReportASPxButton_CD”).Cli
ck
Application.Wait (Now + TimeValue(”00:00:05”))
```

Through these series of clicks, the program automatically downloads an Excel file with all of the merit badges listed.

```
SendKeys ”{TAB}”, True
SendKeys ”{TAB}”, True
SendKeys ”{ENTER}”, True
```

Because the file won’t open until the “Open” button is clicked, we have to use the “SendKeys” to click the “Open” button.

```
Application.OnTime Now + TimeValue(”00:00:05”),
”formatETTEData”

End Sub
```

Because the file won’t open until the VBA code stops running, we stop the code using the “Application.OnTime” method and then restart it in 5 seconds from now by running the “formatETTEData” subprocedure.

MODformatETTEData

```
Option Explicit

Sub formatETTEData()
    unmergeCells
    deleteRowsColumns
    placeScoutsAtTop
    deleteRows
    deleteSpaces
    placeDates
    moreFormatting
    renameSheet
    deleteExistingSheet
    fixMiscMBs
    insertNumbers
    autoFit
    moveSheet
End Sub
```

These subprocedures make the “formatETTEData” sub more user friendly. Each is detailed below.

```
Sub unmergeCells()
    Workbooks(”MBSummary”).Sheets(”SCOUT_rptMeritBadgesCompletedAl”).Select
    Cells.Select
    Selection.UnMerge
End Sub
```

This unmerges all of the cells in the downloaded file.

```
Sub deleteRowsColumns()
    Range(”1:6”).EntireRow.Delete
    Range(”A:B”).EntireColumn.Delete
    Range(”F:G”).EntireColumn.Delete
    Range(”C:D”).EntireColumn.Delete
End Sub
```

This deletes unnecessary rows and columns on the downloaded file.

```
Sub placeScoutsAtTop()

    Range(”B2:B408”).Select
    Selection.Copy
    Sheets.Add After:=ActiveSheet
    Selection.PasteSpecial Paste:=xlPasteValues
    Application.CutCopyMode = False
    Selection.RemoveDuplicates Columns:=1, Header:=xlNo
```

This copies and pastes the values of the names of the scouts on a new sheet and removes the duplicates.

```
    Range(”A2”).Select
    Range(Selection, Selection.End(xlDown)).Select

    Selection.TextToColumns Destination:=Range(”A2”), DataType:=xlDelimited, _
        TextQualifier:=xlDoubleQuote, ConsecutiveDelimiter:=True, Tab:=True, _
        Semicolon:=False, Comma:=False, Space:=True, Other:=False, FieldInfo _
        :=Array(Array(1, 1), Array(2, 1)), TrailingMinusNumbers:=True
    ActiveCell.Range(”A1:C12”).Select
    ActiveWorkbook.Worksheets(”Sheet1”).Sort.SortFields.Clear
    ActiveWorkbook.Worksheets(”Sheet1”).Sort.SortFields.Add Key:=ActiveCell. _
        Offset(0, 1).Range(”A1:A12”), SortOn:=xlSortOnValues, Order:=xlAscending, _
        DataOption:=xlSortNormal
    With ActiveWorkbook.Worksheets(”Sheet1”).Sort
        .SetRange ActiveCell.Range(”A1:C12”)
        .Header = xlGuess
        .MatchCase = False
        .Orientation = xlTopToBottom
        .SortMethod = xlPinYin
        .Apply
    End With
```

This separates the names into first and last name so we can sort alphabetically by last name.

```
    ActiveCell.Offset(0, 3).Range(”A1”).Select
    ActiveCell.FormulaR1C1 = ”=RC[-3]&”” ””&RC[-2]&”” ”””
    ActiveCell.Select
    ActiveCell.FormulaR1C1 = ”=RC[-3]&”” ””&RC[-2]&”” ””&RC[-1]”
    ActiveCell.Select
    Selection.AutoFill Destination:=ActiveCell.Range(”A1:A12”), Type:= _
        xlFillDefault
    ActiveCell.Range(”A1:A12”).Select
```

This concatenates the names back to a full name now sorted alphabetically by last name.

```
    Range(”D2”).Select
    Range(Selection,
Selection.End(xlDown)).Select
```

This copies the names and pastes them onto the original downloaded sheet, transposed at the top of the sheet.

```
    Selection.Copy
    Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
        :=False, Transpose:=False
    Selection.Copy
    Sheets(”SCOUT_rptMeritBadgesCompletedAl”).Select
    Range(”D1”).Select
    Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
        :=False, Transpose:=True

End Sub

Sub deleteSpaces()
Dim x As Integer

For x = 4 To 15
    Cells(1, x) = Trim(Cells(1, x))
Next
End Sub
```

This trims the names so they don’t have any spaces after the last name. This is important so the names match the Council names.

```
Sub placeDates()
Dim x As Integer
Dim y As Integer

For y = 4 To Cells(1, Columns.Count).End(xlToLeft).Column
    For x = 2 To Cells(Rows.Count, 2).End(xlUp).Row
        If Cells(1, y).Value = Cells(x, y).Offset(1, 2 - y) Then
            Cells(x, y).Select
            Cells(x, y).Value = Cells(x, y).Offset(1, 3 - y)
                If Cells(x, 1).Value = ”” Then
                    Cells(x, y).Cut
                        Cells(x, y).Offset(0, 1 - y).Range(”A1”).Select
                        Selection.End(xlUp).Select
                        ActiveCell.Offset(0, y - 1).Range(”A1”).Select
                        ActiveSheet.Paste
                End If
            Cells(x, y).Offset(1, 3 - y).EntireRow.Delete
            ActiveCell.NumberFormat = ”m/d/yyyy”
        End If
    Next
Next
End Sub
```

This sorts through the names and dates earned for each merit badge for each scout on the downloaded sheet and places the date in the column of the proper scout on the merit badge row. Once the name and date are placed in the proper location, the old row is deleted.

```
Sub autoFit()
Columns(”A:Z”).EntireColumn.autoFit
Rows(”1:1000”).EntireRow.autoFit
End Sub
```

This autofits the rows and columns.

```
Sub deleteRows()

Dim x As Integer
For x = 2 To 1000
    If Cells(x, 1) = ”” And Cells(x, 2) = ”” Then
        Cells(x, 1).EntireRow.Delete
    End If
Next
End Sub
```

This deletes empty rows in between the merit badges.

```
Sub moreFormatting()
    Range(”B:C”).EntireColumn.Delete
    Range(”A:A”).EntireColumn.Font.Bold = False
    ActiveWindow.DisplayGridlines = True
End Sub
```

This deletes the old columns that had the names and dates earned, changes the merit badges to not be bolded and displays the gridlines

```
Sub insertNumbers()
Dim x As Integer
```

```
Columns(”A:A”).Select
Selection.Insert Shift:=xlToRight
For x = 1 To 1000
    If Cells(1 + x, 2) <> ”” Then
        Cells(1 + x, 1) = x
    End If
Next
End Sub
```

This inserts numbers next to the names of the merit badges.

```
Sub renameSheet()

Workbooks(”MBSummary”).Worksheets(”SCOUT_rptMeritBadgesCompletedAl”).name =
”eTrailToEagle”

End Sub
```

This renames the downloaded sheet to “eTrailToEagle”

```
Sub deleteExistingSheet()
Dim ws As Worksheet

Application.DisplayAlerts = False
Workbooks(”Final Project”).Activate
For Each ws In Worksheets
    If LCase(ws.name) = LCase(”eTrailToEagle”) Then
        ws.Delete
    End If
Next
Application.DisplayAlerts = True
Workbooks(”MBSummary”).Activate
End Sub
```

This deletes the existing “eTrailToEagle” sheet in the main workbook (if it exists)

```
Sub moveSheet()
    Application.DisplayAlerts = False
    Workbooks(”MBSummary”).Activate
    Sheets(”eTrailToEagle”).Select
    Sheets(”eTrailToEagle”).Copy Before:=Workbooks(”Final Project.xlsm”).Sheets(1)
    Workbooks(”MBSummary”).Close
    Application.DisplayAlerts = True
End Sub
```

This moves the downloaded and formatted eTrailToEagle sheet to the main workbook.

```
Sub fixMiscMBs()

    Cells(7, 1).EntireRow.Insert
    Cells(7, 1).Value = ”Animation”
    Cells(25, 1).EntireRow.Insert
    Cells(25, 1).Value = ”Cinematography”
    Cells(85, 1).EntireRow.Delete
    Cells(88, 1).EntireRow.Delete
    Cells(120, 1).EntireRow.Insert
    Cells(120, 1).Value = ”Signs, Signals, and Codes”
    Cells(122, 1).EntireRow.Delete

End Sub
```

There are inconsistencies between the eTrailToEagle merit badges and the Council merit badges. This fixes those inconsistencies.

MODComparison

```
Option Explicit

Sub comparisonSheet()

    Dim c As Integer
    Dim ws As Integer
    Dim r As Integer
    Dim cc As Integer
    Dim wsht As Worksheet
    Application.DisplayAlerts = False
    For Each wsht In Worksheets
        If LCase(wsht.name) = LCase(”Comparison”) Then
            wsht.Delete
        End If
    Next
    Application.DisplayAlerts = True
    Worksheets.Add().name = ”Comparison”
```

This deletes the existing “Comparison” tab (if it exists) and adds a new one.

```
    Sheets(”eTrailToEagle”).Select
    Columns(”A:B”).Select
    Selection.Copy
    Sheets(”Comparison”).Select
    Columns(”A:A”).Select
    ActiveSheet.Paste
    Sheets(”eTrailToEagle”).Select
    Rows(1).Select
    Selection.Copy
    Sheets(”Comparison”).Select
    Rows(1).Select
    ActiveSheet.Paste
    Columns(”A:Z”).EntireColumn.autoFit
    Rows(”1:1000”).EntireRow.autoFit
```

This copies the merit badge numbers, merit badge names, and the scout names from the eTrialToEagle sheet and pastes it on the newly created Comparison sheet.

```
For ws = 3 To 5
 For c = 3 To Worksheets(”eTrailToEagle”).Cells(1, Columns.Count).End(xlToLeft).Column
  For cc = 3 To Worksheets(ws).Cells(1, Columns.Count).End(xlToLeft).Column

   If Worksheets(”eTrailToEagle”).Cells(1, c).Value = Worksheets(ws).Cells(1,_
   cc).Value Then

     For r = 2 To Worksheets(”eTrailToEagle”).Cells(Rows.Count, 2).End(xlUp).Row

      If Not Worksheets(”eTrailToEagle”).Cells(r, c).Value = Worksheets(ws).Cells _
      (r,cc).Value Then

        Worksheets(”Comparison”).Cells(r, c).Value = Worksheets(”eTrailToEagle”). _
        Cells(r, c).Value

      End If
     Next
   End If
  Next
 Next
Next

Sheets(”Comparison”).Select

End Sub
```

This looks at each scout listed on the eTrailToEagle sheet and matches the name to the name on each Council sheet. If the name is a match, it goes through the merit badge dates earned to see if they match. If they don’t match, the date on the eTrailToEagle sheet is placed on the Comparison sheet in the proper place.

MODRibbon

This hooks up the new ribbon buttons to the proper subprocedures.

```
Option Explicit

Sub btnTroop_click(control As IRibbonControl)
    getTroopCouncilData1
End Sub

Sub btnTeam_click(control As IRibbonControl)
    getTeamCouncilData1
End Sub
```

```
Sub btnCrew_click(control As IRibbonControl)
    getCrewCouncilData1
End Sub

Sub btnETrailToEagle_click(control As IRibbonControl)
    downloadETTEData
End Sub

Sub btnRunComparison_click(control As IRibbonControl)
    comparisonSheet
End Sub
```