Caleb Brigman

MBA 614 Spreadsheets for Automation

Final Project: Personal Finance Dashboard

The project was to create a personal finance dashboard to track spending relative to budget goals. Currently, my wife and I's finances a spread across four bank accounts and three consumer credit lines which has made it difficult to track. Each of the banks uses a different format for exporting account information which takes hours to standardize and then label according to the spending area. The purpose of the project was to automate this process so it can be completed monthly and create a dashboard to allow us to easily see our spending relative to our goals.

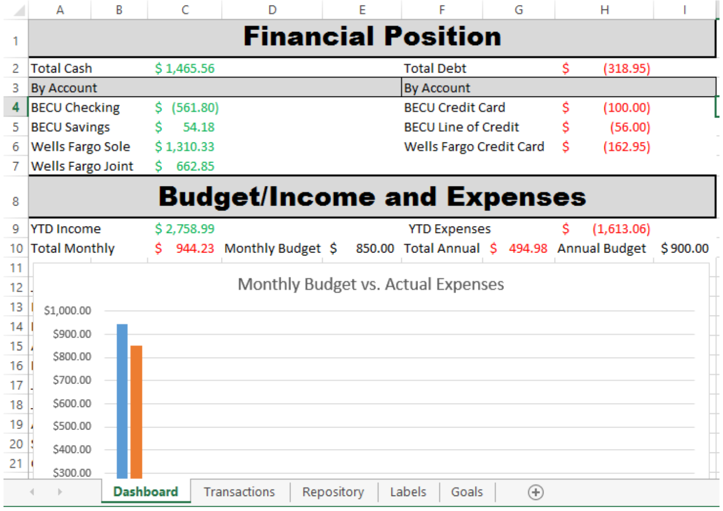
The dashboard tool has three major parts. First, the tool imports account activity from CSV files. Each transaction is standardized and labeled with one of the spending goals. Labels are selected for a type of transaction the first time it is imported into the tool and then is automatically labeled thereafter. Second, the transaction data is compiled and added to the dashboard. The dashboard includes overall numbers for each account as well as monthly numbers by spending area. The compiling is only done for months that are already over and when the numbers have not been compiled previously. Third, only the dashboard is published as a PDF and emailed via Outlook to my wife and me.

Implementation Documentation

The personal finance dashboard has three major parts. The first part loads the bank account information from CSV files in a standardized format and labels each transaction. The second part takes the transactions and compile monthly and overall numbers which are added to the dashboard. The final part publishes the dashboard and emails it to my wife and I. Each part has a corresponding button on the ribbon.

Structure of the Tool

The tool has five sheets. The first sheet is the Dashboard which contains the summary finance information and is the information that gets sent out.



Financial Position

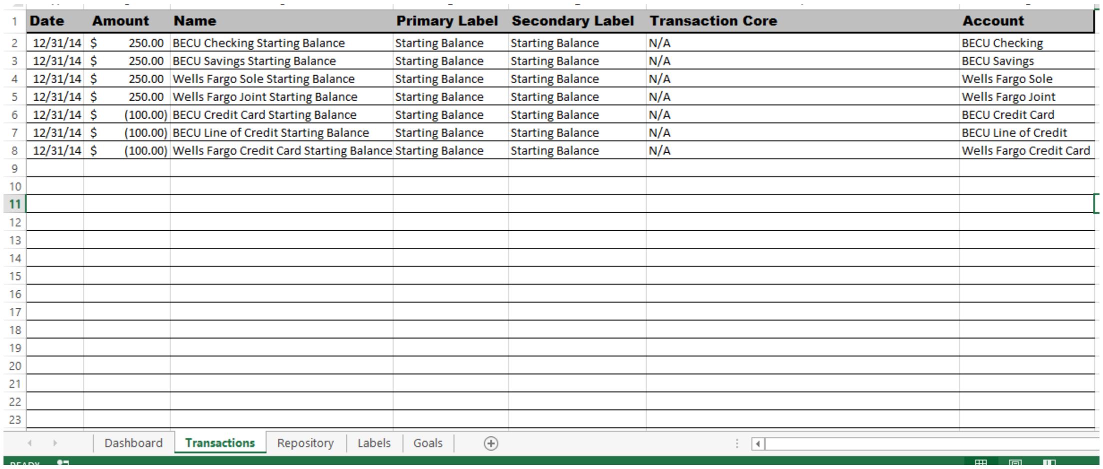
Total Cash	$ 1,465.56			Total Debt	$ (318.95)		
By Account				By Account			
BECU Checking	$ (561.80)			BECU Credit Card	$ (100.00)		
BECU Savings	$ 54.18			BECU Line of Credit	$ (56.00)		
Wells Fargo Sole	$ 1,310.33			Wells Fargo Credit Card	$ (162.95)		
Wells Fargo Joint	$ 662.85						

Budget/Income and Expenses

YTD Income	$ 2,758.99			YTD Expenses	$ (1,613.06)		
Total Monthly	$ 944.23	Monthly Budget	$ 850.00	Total Annual	$ 494.98	Annual Budget	$ 900.00

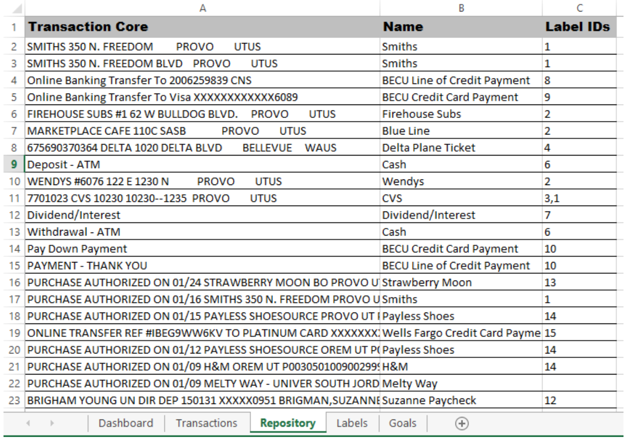
Monthly Budget vs. Actual Expenses

The second sheet is the Transaction which contain each individual transactions for each of the accounts in the standardized format and labeled based on the spending areas.



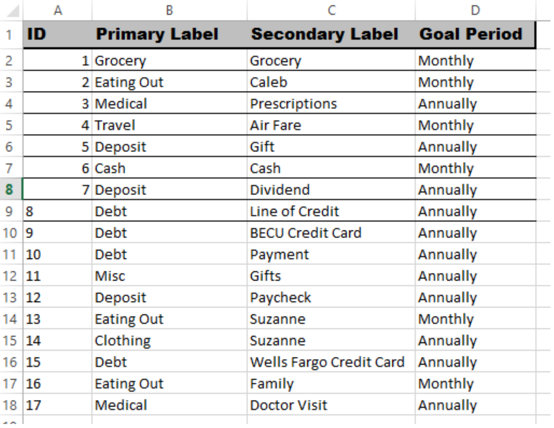
Date	Amount	Name	Primary Label	Secondary Label	Transaction Core	Account
12/31/14	$ 250.00	BECU Checking Starting Balance	Starting Balance	Starting Balance	N/A	BECU Checking
12/31/14	$ 250.00	BECU Savings Starting Balance	Starting Balance	Starting Balance	N/A	BECU Savings
12/31/14	$ 250.00	Wells Fargo Sole Starting Balance	Starting Balance	Starting Balance	N/A	Wells Fargo Sole
12/31/14	$ 250.00	Wells Fargo Joint Starting Balance	Starting Balance	Starting Balance	N/A	Wells Fargo Joint
12/31/14	$ (100.00)	BECU Credit Card Starting Balance	Starting Balance	Starting Balance	N/A	BECU Credit Card
12/31/14	$ (100.00)	BECU Line of Credit Starting Balance	Starting Balance	Starting Balance	N/A	BECU Line of Credit
12/31/14	$ (100.00)	Wells Fargo Credit Card Starting Balance	Starting Balance	Starting Balance	N/A	Wells Fargo Credit Card

The third sheet contains a repository of the “transaction core,” shortened transaction information with the date removed to all labeling to be automated. The repository also contains which label IDs correspond to which transaction cores.



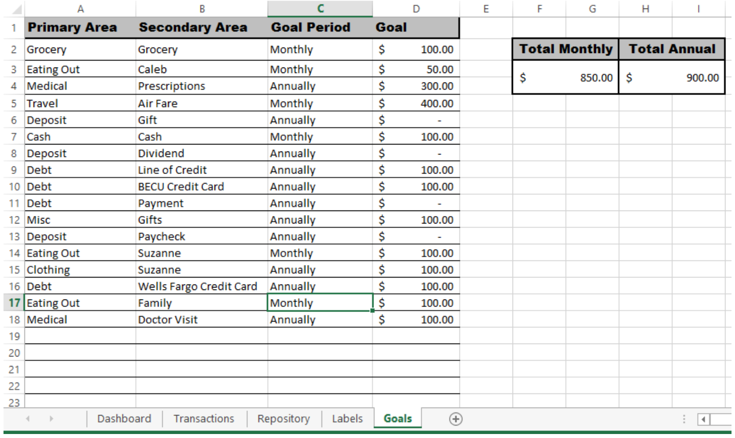
Transaction Core	Name	Label IDs
SMITHS 350 N. FREEDOM PROVO UTUS	Smiths	1
SMITHS 350 N. FREEDOM BLVD PROVO UTUS	Smiths	1
Online Banking Transfer To 2006259839 CNS	BECU Line of Credit Payment	8
Online Banking Transfer To Visa XXXXXXXXXXXX6089	BECU Credit Card Payment	9
FIREHOUSE SUBS #1 62 W BULLDOG BLVD. PROVO UTUS	Firehouse Subs	2
MARKETPLACE CAFE 110C SASB PROVO UTUS	Blue Line	2
675690370364 DELTA 1020 DELTA BLVD BELLEVUE WAUS	Delta Plane Ticket	4
Deposit - ATM	Cash	6
WENDYS #6076 122 E 1230 N PROVO UTUS	Wendys	2
7701023 CVS 10230 10230--1235 PROVO UTUS	CVS	3,1
Dividend/Interest	Dividend/Interest	7
Withdrawal - ATM	Cash	6
Pay Down Payment	BECU Credit Card Payment	10
PAYMENT - THANK YOU	BECU Line of Credit Payment	10
PURCHASE AUTHORIZED ON 01/24 STRAWBERRY MOON BO PROVO U	Strawberry Moon	13
PURCHASE AUTHORIZED ON 01/16 SMITHS 350 N. FREEDOM PROVO U	Smiths	1
PURCHASE AUTHORIZED ON 01/15 PAYLESS SHOESOURCE PROVO UT	Payless Shoes	14
ONLINE TRANSFER REF #IBEG9WW6KV TO PLATINUM CARD XXXXXXX	Wells Fargo Credit Card Payme	15
PURCHASE AUTHORIZED ON 01/12 PAYLESS SHOESOURCE OREM UT P	Payless Shoes	14
PURCHASE AUTHORIZED ON 01/09 H&M OREM UT P003050100900299	H&M	14
PURCHASE AUTHORIZED ON 01/09 MELTY WAY - UNIVER SOUTH JORD	Melty Way	
BRIGHAM YOUNG UN DIR DEP 150131 XXXXX0951 BRIGMAN,SUZANNE	Suzanne Paycheck	12

The fourth sheet contains the labels and label IDs.



ID	Primary Label	Secondary Label	Goal Period
1	Grocery	Grocery	Monthly
2	Eating Out	Caleb	Monthly
3	Medical	Prescriptions	Annually
4	Travel	Air Fare	Monthly
5	Deposit	Gift	Annually
6	Cash	Cash	Monthly
7	Deposit	Dividend	Annually
8	Debt	Line of Credit	Annually
9	Debt	BECU Credit Card	Annually
10	Debt	Payment	Annually
11	Misc	Gifts	Annually
12	Deposit	Paycheck	Annually
13	Eating Out	Suzanne	Monthly
14	Clothing	Suzanne	Annually
15	Debt	Wells Fargo Credit Card	Annually
16	Eating Out	Family	Monthly
17	Medical	Doctor Visit	Annually

The final sheet is the goals and contains the spending areas and the associated goals for each area. Each sheet plays a specific part in the tool



Primary Area	Secondary Area	Goal Period	Goal
Grocery	Grocery	Monthly	$ 100.00
Eating Out	Caleb	Monthly	$ 50.00
Medical	Prescriptions	Annually	$ 300.00
Travel	Air Fare	Monthly	$ 400.00
Deposit	Gift	Annually	$ -
Cash	Cash	Monthly	$ 100.00
Deposit	Dividend	Annually	$ -
Debt	Line of Credit	Annually	$ 100.00
Debt	BECU Credit Card	Annually	$ 100.00
Debt	Payment	Annually	$ -
Misc	Gifts	Annually	$ 100.00
Deposit	Paycheck	Annually	$ -
Eating Out	Suzanne	Monthly	$ 100.00
Clothing	Suzanne	Annually	$ 100.00
Debt	Wells Fargo Credit Card	Annually	$ 100.00
Eating Out	Family	Monthly	$ 100.00
Medical	Doctor Visit	Annually	$ 100.00

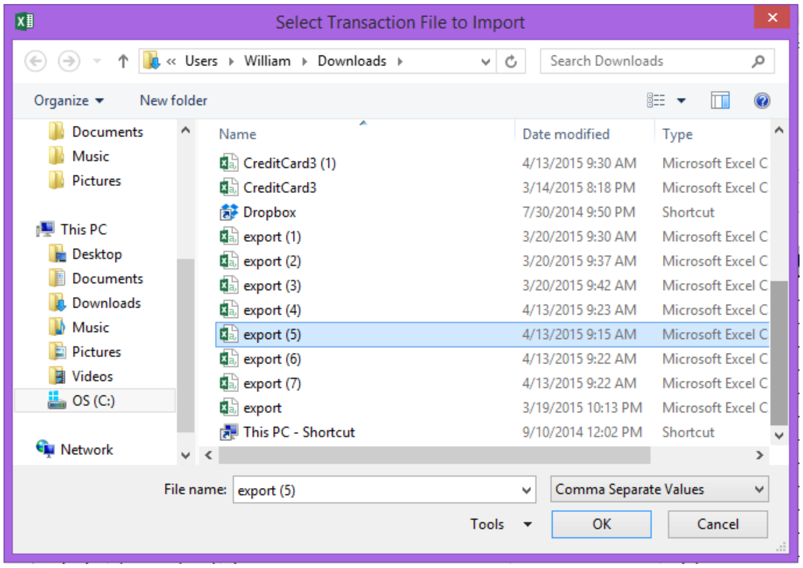
Total Monthly	Total Annual
$ 850.00	$ 900.00

Load Statements

The first part of the tool is loading the transactions from the back statement. Loading bank statements is initiated by selecting the “Load Statement” button on the “Finances” tab. The macro loads the bank statements into the tool through four steps.

Step 1: Select the Statement to Load

The macro first loads a file dialog box to select the statement to be loaded. The file select is filtered for CSV files and only CSV files can be selected.



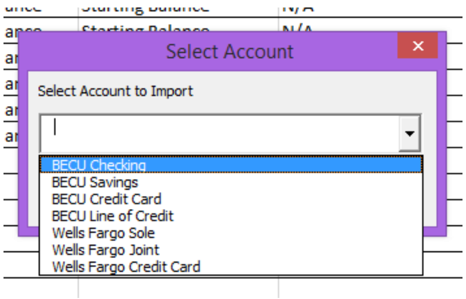
Name	Date modified	Type
CreditCard3 (1)	4/13/2015 9:30 AM	Microsoft Excel C
CreditCard3	3/14/2015 8:18 PM	Microsoft Excel C
Dropbox	7/30/2014 9:50 PM	Shortcut
export (1)	3/20/2015 9:30 AM	Microsoft Excel C
export (2)	3/20/2015 9:37 AM	Microsoft Excel C
export (3)	3/20/2015 9:42 AM	Microsoft Excel C
export (4)	4/13/2015 9:23 AM	Microsoft Excel C
export (5)	4/13/2015 9:15 AM	Microsoft Excel C
export (6)	4/13/2015 9:22 AM	Microsoft Excel C
export (7)	4/13/2015 9:22 AM	Microsoft Excel C
export	3/19/2015 10:13 PM	Microsoft Excel C
This PC - Shortcut	9/10/2014 12:02 PM	Shortcut

File name: export (5)

Comma Separate Values

Step 2: Select Account Associated with Statement

After the file has been selected, a user form appears with a dropdown to select which account the statement is for. There are seven different account and the account type selected determines how the statement is processed and triggers one of seven different sub procedures.



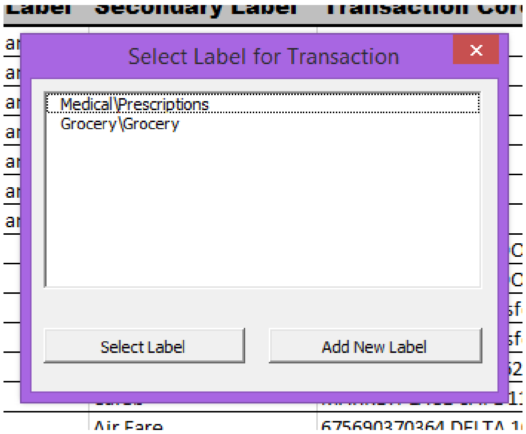
Step 3: Record the Data from the Statement into the Tool

From the statement, the date, value and name for each transaction is added into the tool on the "Transaction Sheet." The location of the information on the statement varies by account so a separate sub is called depending on the account selected. The name of the account is also record with each transaction. Once the values have been recorded, the transaction name is edited to remove the non-essential information, such as "POS Withdrawal," and date specific information to create the "transaction core" which is used later. Each account has multiple formats for the transaction name with different formulas to edit them.

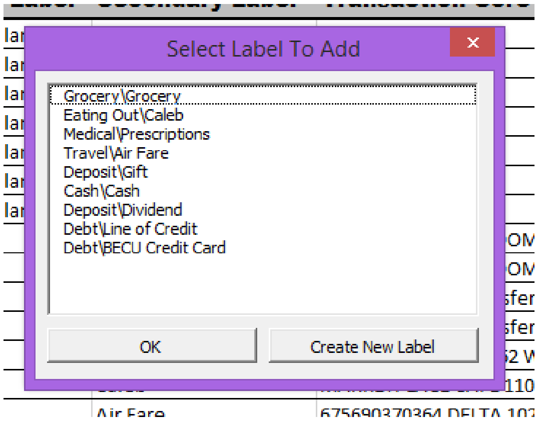
Step 4: Label the Transaction

Once the information has been recorded and the "transaction core" is created, the transaction is labeled. The tool compared the transaction core to the transaction cores listed in the repository. When the matching transaction core is found, it records the common name for the transaction, such as Smiths, to the Transaction sheet. If there is only one Label ID associated with the transaction core, the tool looks about the ID on the label sheet and record the Primary and Secondary labels, such as "Eating out" and "Family," for the transaction.

If multiple IDs are associated with the transaction, a user form will appear to select which label to use.



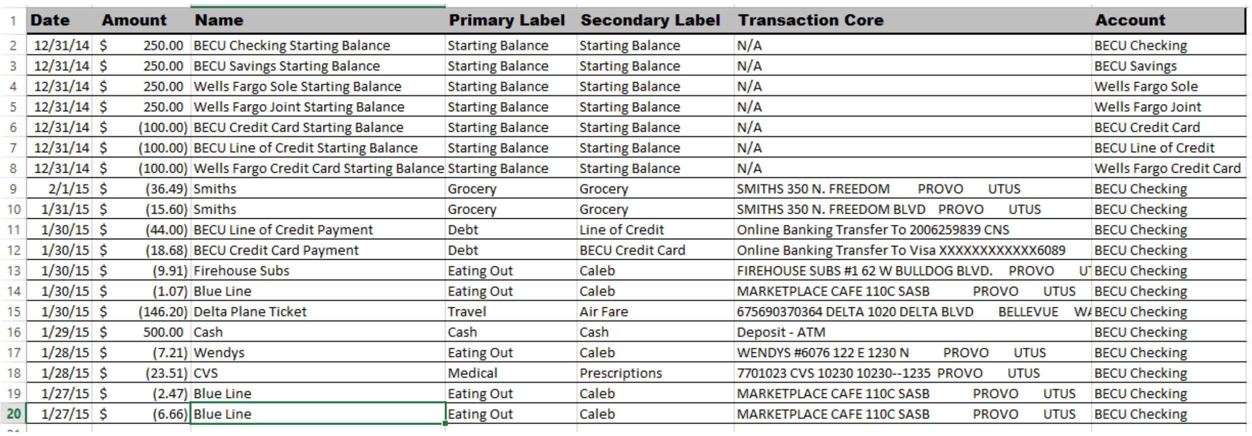
The user can also choose to associate an existing label with this transaction core by selecting "Add New Label." This opens another user form to select from the existing labels. When the label is selected, the ID is added to the Label ID column on the Repository sheet.



Grocery\Grocery
Eating Out\Caleb
Medical\Prescriptions
Travel\Air Fare
Deposit\Gift
Cash\Cash
Deposit\Dividend
Debt\Line of Credit
Debt\BECU Credit Card

Additionally, the user could create an entirely new label by selecting "Create New Label." This will open a series of input boxes to create the Primary and Secondary label and to select the time period for the label, either "Monthly" or "Annually." These values are added to the next available row on the Labels sheet and assigned the next sequential number ID.

When the labels have been added, the transaction is now standardized for all accounts and ready to be used to compile the dashboard.



1	Date	Amount	Name	Primary Label	Secondary Label	Transaction Core	Account
2	12/31/14	$ 250.00	BECU Checking Starting Balance	Starting Balance	Starting Balance	N/A	BECU Checking
3	12/31/14	$ 250.00	BECU Savings Starting Balance	Starting Balance	Starting Balance	N/A	BECU Savings
4	12/31/14	$ 250.00	Wells Fargo Sole Starting Balance	Starting Balance	Starting Balance	N/A	Wells Fargo Sole
5	12/31/14	$ 250.00	Wells Fargo Joint Starting Balance	Starting Balance	Starting Balance	N/A	Wells Fargo Joint
6	12/31/14	$ (100.00)	BECU Credit Card Starting Balance	Starting Balance	Starting Balance	N/A	BECU Credit Card
7	12/31/14	$ (100.00)	BECU Line of Credit Starting Balance	Starting Balance	Starting Balance	N/A	BECU Line of Credit
8	12/31/14	$ (100.00)	Wells Fargo Credit Card Starting Balance	Starting Balance	Starting Balance	N/A	Wells Fargo Credit Card
9	2/1/15	$ (36.49)	Smiths	Grocery	Grocery	SMITHS 350 N. FREEDOM PROVO UTUS	BECU Checking
10	1/31/15	$ (15.60)	Smiths	Grocery	Grocery	SMITHS 350 N. FREEDOM BLVD PROVO UTUS	BECU Checking
11	1/30/15	$ (44.00)	BECU Line of Credit Payment	Debt	Line of Credit	Online Banking Transfer To 2006259839 CNS	BECU Checking
12	1/30/15	$ (18.68)	BECU Credit Card Payment	Debt	BECU Credit Card	Online Banking Transfer To Visa XXXXXXXXXXXX6089	BECU Checking
13	1/30/15	$ (9.91)	Firehouse Subs	Eating Out	Caleb	FIREHOUSE SUBS #1 62 W BULLDOG BLVD. PROVO UT	BECU Checking
14	1/30/15	$ (1.07)	Blue Line	Eating Out	Caleb	MARKETPLACE CAFE 110C SASB PROVO UTUS	BECU Checking
15	1/30/15	$ (146.20)	Delta Plane Ticket	Travel	Air Fare	675690370364 DELTA 1020 DELTA BLVD BELLEVUE WA	BECU Checking
16	1/29/15	$ 500.00	Cash	Cash	Cash	Deposit - ATM	BECU Checking
17	1/28/15	$ (7.21)	Wendys	Eating Out	Caleb	WENDYS #6076 122 E 1230 N PROVO UTUS	BECU Checking
18	1/28/15	$ (23.51)	CVS	Medical	Prescriptions	7701023 CVS 10230 10230--1235 PROVO UTUS	BECU Checking
19	1/27/15	$ (2.47)	Blue Line	Eating Out	Caleb	MARKETPLACE CAFE 110C SASB PROVO UTUS	BECU Checking
20	1/27/15	$ (6.66)	Blue Line	Eating Out	Caleb	MARKETPLACE CAFE 110C SASB PROVO UTUS	BECU Checking

Compile Dashboard

The second part of the tool is compiling the dashboard. This takes the information from the Transactions and Goals and uses this to compile monthly finance numbers and the overall current financial position. The macro is activated by selecting the “Compile Dashboard” button on the “Finances” ribbon. The numbers are compiled in seven steps.

Step 1: Check Label and Goal Match

The first step is to ensure that all labels have a corresponding goal. This ensures that no transactions are unintentionally left out of the dashboard. The tool compared each label to the goals to ensure each label has a corresponding goal and that the period, Monthly or Annually, matches between the label and the goal. If the label and goal match, the tool moves to the next label. If the period, does not match, an input box displays for the user to indicate the appropriate period and this period is changed on the Label and Goal sheet. If a goal does not exist for the label, the label is added to the next available row on the goal sheet and the user is asked to input a goal for the area in an input box. Once all of the labels and goals have been verified, the tool moves to the next step.

Step 2: Determine Months to Compile

The next step is to identify which months to compile the data for. Because some goals will change over the course of the year, the tool is designed to only compile the numbers for a month once against on the goals as they existed for the month. To accomplish this, the tool looks at the transactions and the dashboard. A month is only compiled if two conditions are met. First, the month needs to over. This is determined by finding the highest transaction date and only compiling the data for the month is the end of the months is before the highest transaction date. Second, the data for the month cannot have been previously compiled. This is checked on the dashboard in the fields reserved for that month. Each month is compared against these criteria and if it meets both, the variable for the month is changed to true and signals later subs to compile the data for that month.

Step 3: Create and Load the Arrays

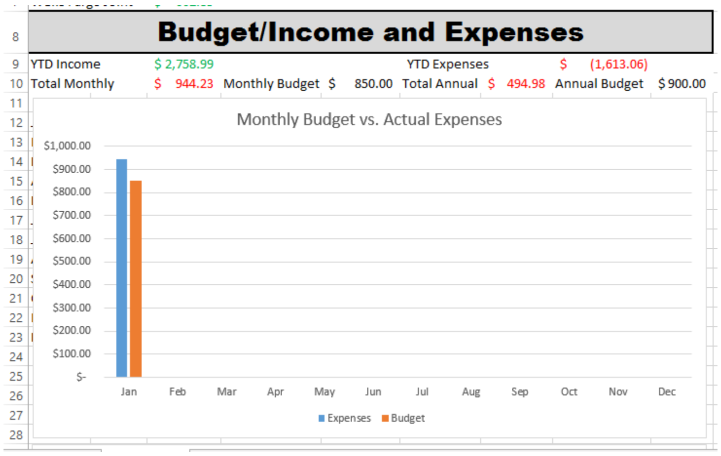
The tool creates two arrays. One array holds the Primary and Secondary labels for each goal. The other array holds the goal amount for each goal and then a place for the actual spending for the month in that area. These arrays are created by reading the goals from the Goals sheet. All goal which have been given a goal greater than $0 are included. (Some goals have a $0 value because they are connected with transfers between accounts, such as using the checking account to pay a credit card. These are excluded to avoid double counting.) These arrays are used to compile the numbers.

Step 4: Compile the Monthly Numbers

The tool then goes line by line through the transactions. If the transaction occurred in the month to be compiled, then it findings the corresponding label in the first array with the label names. Once a match is found, the value of the amount for the transaction is added the value for that goal in the second array. In this way a running total for the month for each goal is kept. This continues until all of the transactions have been reviewed.

Step 5: Record the Budget and Expense Totals

The tool then adds up all of the monthly goal amounts and actual expenses for the area to create a grand total for the month of budget versus expenses and adds it to cells connected to a chart on the dashboard. The same thing is done with the annual goals.

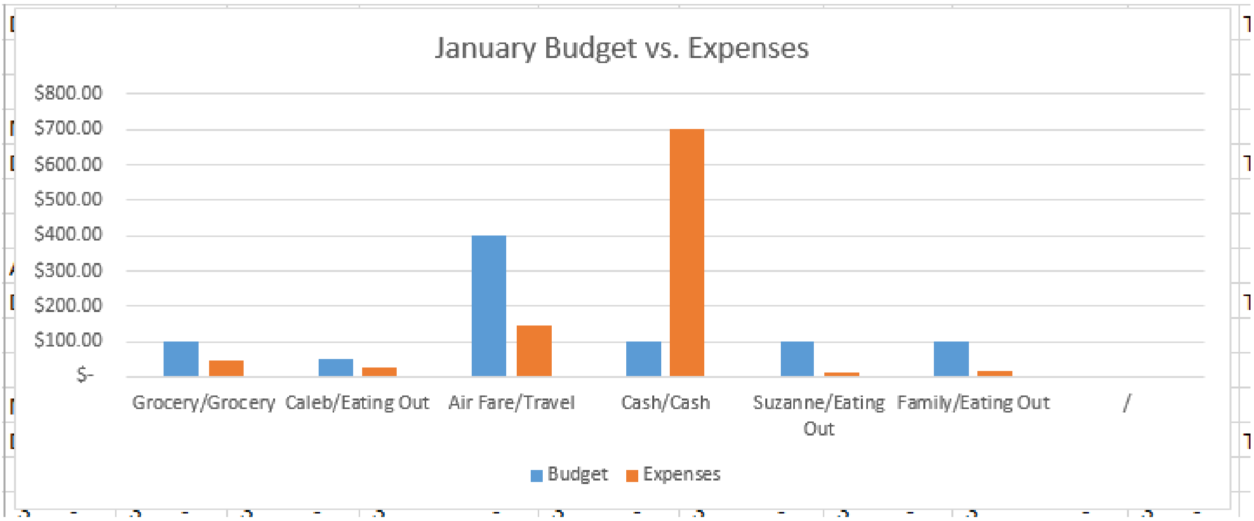


Budget/Income and Expenses

YTD Income	$ 2,758.99			YTD Expenses	$ (1,613.06)		
Total Monthly	$ 944.23	Monthly Budget	$ 850.00	Total Annual	$ 494.98	Annual Budget	$ 900.00

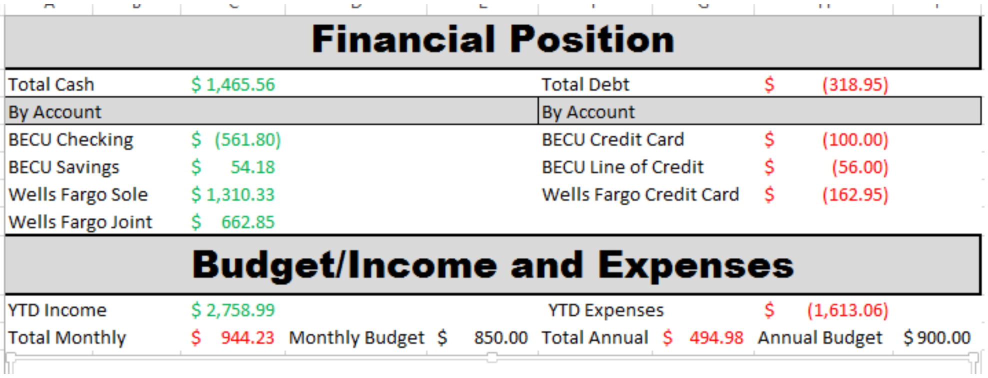
Step 6: Update Monthly Charts

The tool then takes each of the monthly budget goals and actual expenses and adds the label, budget amount and expense amount to cells connected to a chart for the months on the dashboard. The source data for the chart is resized appropriately for the number of goals for the month to create a final budget versus actual chart for the month.



Step 7: Compute the Current Financial Position

The final step in compiling the dashboard is computing the current financial position. This includes the current cash between all accounts and the current debt for all accounts. Additionally, it includes the cash and debt by account. Finally it includes the total year to date income and expenses. This is computed by reviewing each of the transactions on the Transaction sheet and then adding up all of the inputs and outputs for each account on a running basis to create a total number. These values are recorded on the dashboard. Once this has been completed, the dashboard is ready to be sent out.



Financial Position

Total Cash	$ 1,465.56	Total Debt	$ (318.95)
By Account		By Account	
BECU Checking	$ (561.80)	BECU Credit Card	$ (100.00)
BECU Savings	$ 54.18	BECU Line of Credit	$ (56.00)
Wells Fargo Sole	$ 1,310.33	Wells Fargo Credit Card	$ (162.95)
Wells Fargo Joint	$ 662.85		

Budget/Income and Expenses

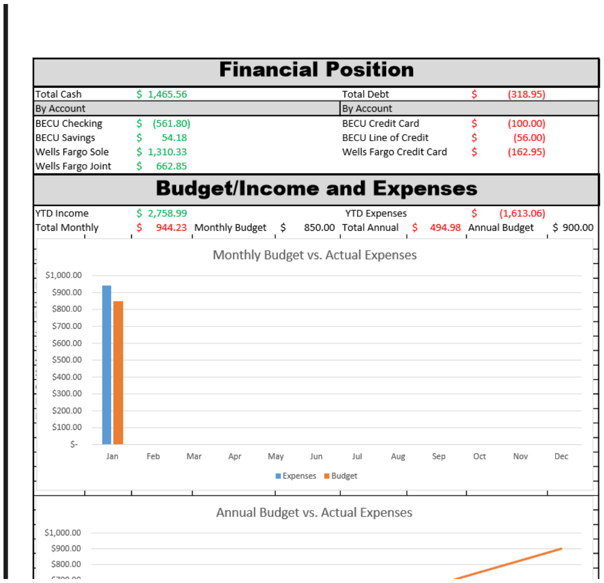
YTD Income	$ 2,758.99			YTD Expenses	$ (1,613.06)		
Total Monthly	$ 944.23	Monthly Budget $	850.00	Total Annual	$ 494.98	Annual Budget	$ 900.00

Email the Dashboard

The last part of the tool is publishing the Dashboard sheet and sending it out via email to my wife and I. This is activated by selecting the “Email” button on the “Finances” ribbon. This occurs in three step, of which two are handled in VBA.

Step 1: Publish the Dashboard

The tools publishes only the Dashboard sheet in a PDF. This is done by having a set print area. This print area is used as the guide for what will be included in the published PDF.



Financial Position

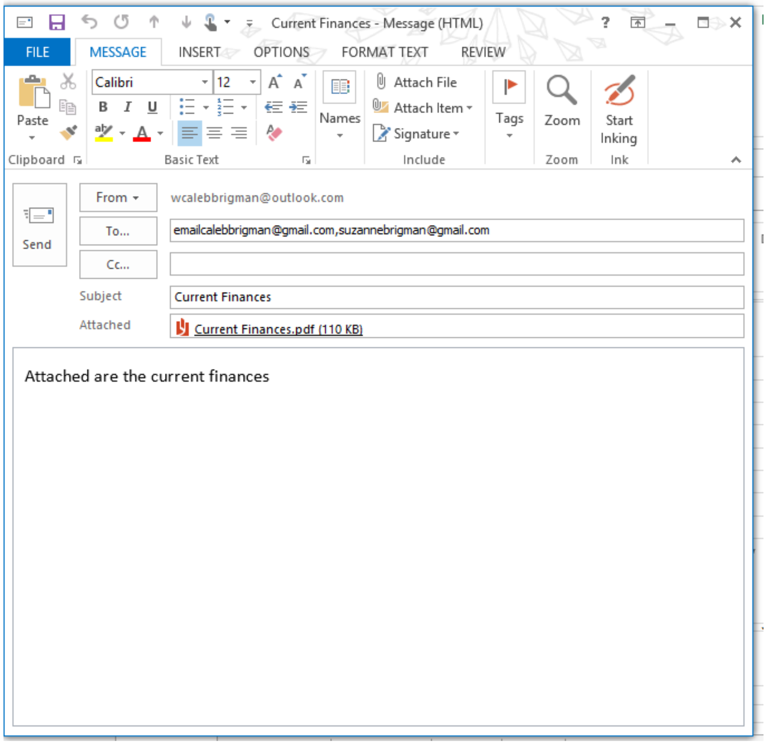
Total Cash	$ 1,465.56	Total Debt	$ (318.95)
By Account		By Account	
BECU Checking	$ (561.80)	BECU Credit Card	$ (100.00)
BECU Savings	$ 54.18	BECU Line of Credit	$ (56.00)
Wells Fargo Sole	$ 1,310.33	Wells Fargo Credit Card	$ (162.95)
Wells Fargo Joint	$ 662.85		

Budget/Income and Expenses

YTD Income	$ 2,758.99			YTD Expenses	$ (1,613.06)		
Total Monthly	$ 944.23	Monthly Budget	$ 850.00	Total Annual	$ 494.98	Annual Budget	$ 900.00

Step 2: Create Outlook Message

Once the PDF has been created, the tool creates an Outlook message to be used to send out the Dashboard. The message is created already set with my wife and I’s email addresses, the subject line and the PDF attached. The body of the message is intentionally left open so that I can include in personal comments I would like to communicate to my wife about our finances. The tool creates the message and opens the email message window in Outlook.



From: wcalebbrigman@outlook.com

To...: emailcalebbrigman@gmail.com,suzannebrigman@gmail.com

Cc...:

Subject: Current Finances

Attached: Current Finances.pdf (110 KB)

Attached are the current finances

Step 3: Send the Email

Finally, I add my comments to the email in the Outlook message window and click send. This is the only part done outside of VBA and Excel.

Difficulties Encountered

I encountered four major difficulties during this project. Three of the issues I encountered were the result of my limited exposure to that area in VBA. The last was the result of changes outside of my control.

<u>Charts</u>

One of the areas I struggled with was creating charts in VBA. My original idea was to only create the monthly budget versus actual charts when the numbers for that month were being compiled. However, I had to abandon that for two reasons. The biggest was because of how publishing is done in Excel. The second issues I encountered was trying to create a chart based on an array. I tried several different options, but was not successful. After extensive research in VBA blogs and forums, I discovered it was actually impossible and no one seemed to be able to do it without recording the values to a cell somewhere. Given this, I had to follow suit and instead record my values to a cell and have those cells connected to the chart and only change the source data used.

Publishing

Another area I struggle with was publishing the dashboard to a PDF. While it is very easy to publish an entire workbook using VBA, publishing just one part of one sheet is much harder. I originally tried to make a dynamic way to adjust the area to be published. However, I discovered that even for more experienced coders this was fraught with errors. Instead I resorted to preselecting the print area and then using the print area as the guide for what to publish.

Creating Emails in Outlook

This area was only difficult because Outlook has very specific code to be used in VBA and I did not know what it was not was it covered in class. I had research this online to final several sites that could teach me how to code for Outlook messages in VBA.

Bank Changes

My biggest difficulty was with changes that both banks made to their statements. I had completed programing the first part of the tool to load the statements only to discover that both banks had recently upgraded their systems and create new formats for their bank statements. I had to go back and reprogram the entire first part of the tool to align to these new changes.

Learnings

I took away three major learnings from this project: scope creep, balancing user design and functional design. These learnings are less specific to VBA and more applicable to any technology project.

Scope Creep

Scope creep is a very real thing in a technology project. I was constantly tempted, and often succumbed, to want to add more features to what I was doing. Often it felt like I could easily add them by just doing an hour more here and an hour more there. However, overtime these small additions added up to an enormous additional workload. Finally, I got into the habit at the end of asking “Will this additional feature really help me do personal finances better?” and it most cases the answer was no. In future projects I need to more clearly define the scope to avoid this issue.

Balancing User Design

Similar to scope creep, I was often confronted with the issues of adding more time to the project to have a more “fool proof” tool or acknowledging some responsibility on the user’s side to pay attention to what I was doing. Adding safeguards to prevent users from making mistakes takes a significant investment in additional code. It was a trade-off I often had to think through and make.

Functional Design

I did not do any extensive functional design work prior to starting to code. I had a general idea of how the tool should work and then just jumped in. However, as I progressed through the project this approach came back to bite me. I found that after getting into later parts of the tool that the way I had programmed other sections was a barrier to the current section. I was often faced with deciding whether to go back and change past sections or program around those issues. I could have avoided most of this and used far less code if I had done a more extensive functional design to begin with.

Assistance

I used no assistance from another individual. The only help I received was from the book required for the class and by reviewing VBA blogs and forums.