Executive Summary

Project Objective:

Far too often people arrive home after grocery shopping only to realize that they have forgotten to purchase one or more essential grocery items for the upcoming week. Too often, I have found myself in this exact predicament. Forgetting an item is especially aggravating when it is an essential ingredient in meals you were hoping to make that week. The only solutions are to either go back to the store or revise your meal plans for the upcoming week.

In an effort to solve this problem I have created a VBA grocery list which ensures that users will always go to the grocery store with a complete list of the ingredients they will need based on what they want to make that week.

System Overview:

The VBA grocery list achieves the project objectives by first offering users the ability to enter in all of the recipes that they love to make. These recipes are stored in a worksheet. When it is time to go to the grocery store, the user clicks on the "make a list" button and is prompted with the names of the recipes they entered into the system. Now the user only needs to choose which recipes they want to make for the upcoming week and the program will automatically compile a list of all necessary ingredients (and their quantity) that the user will need to make their selected meals.

Before finalizing the list, the program allows the user to go through and uncheck the ingredients that they already have or modify the ingredient amounts. Finally the program will group the items based on the food type (eg. dairy, vegetables, etc…) and the user can click a button to print out the complete grocery list.

Implementation and documentation

Adding Recipes

When users open the VBA grocery list they are faced with the main page which has three buttons- "Add Recipe," "Make a Grocery List," and "Print." The main page worksheet is named "UI." The first thing a user must do is add their recipes to the program. If the program has no recipes, the user will not be able to make a grocery list.

Figure 1: The main page

To add a recipe, the user clicks on the "Add Recipe" button (the top button on figure 1). At this point, the user is prompted with the add recipe user form. This form can be seen in figure 2. To

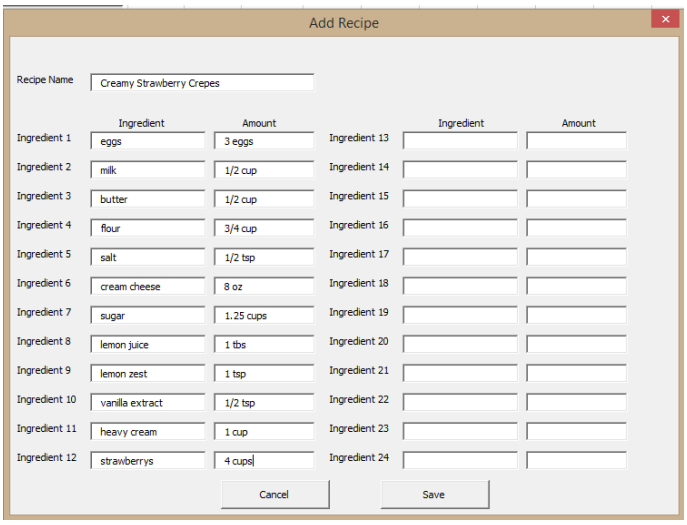

Figure 2: Add Recipe Form

use the form, the user simply writes in the recipe name, the ingredients, and the amount of those ingredients. In the case of figure 2, the user is entering a recipe for Creamy Strawberry Crepes. After entering the recipe, the user simply clicks the "Save" button and the recipe will be saved in the recipe sheet in the workbook. After clicking "Save" the Add Recipe user form goes away and the user is once again looking at the main page with the three buttons. The user also has the option to click the "cancel" button on the "add recipe" user form which will immediately close the user form and returned the user to the main page.

The user can save as many recipes as desired. As stated above, the saved recipes are stored in the "Recipes" tab of the workbook. They can be looked at, edited, or deleted from this worksheet if needed. Figure 3 shows some of the recipes in this worksheet.

Asparagus Parmesan		
	butter	1 tbs
	olive oil	1/4 cup
	asparagus	1 lb
	parmesan cheese	3/4 cups
	salt	1 pinch
	pepper	1 pinch
end		
Creamy Strawberry Crepes		
	eggs	3 eggs
	milk	1/2 cup
	butter	1/2 cup
	flour	3/4 cup
	salt	1/2 tsp
	cream cheese	8 oz
	sugar	1.25 cups
	lemon juice	1 tbs
	lemon zest	1 tsp
	vanilla extract	1/2 tsp
	heavy cream	1 cup
	strawberrys	4 cups
end		

UI | Recipes | Categories | Lists

Figure 3: The recipe worksheet

Generating a Grocery List

When it is time to go to grocery shopping, the user can quickly use this workbook to make and print out a perfect grocery list. First the user clicks on the "Make a List" button on the main page (see figure 1). After clicking on this button, the user is instantly prompted with the "Choose Recipes" user form. This form is displayed in figure 4. The List Box on this form is populated with the names of all the recipes that the user has entered and saved using the "Add Recipe" button discussed above. At this point the user either chooses to click the "cancel" or "Okay" button. The cancel button will simply hide the "Choose Recipes" user form and return the user to the main page. The "Okay" button will begin to compile the list by doing the following:

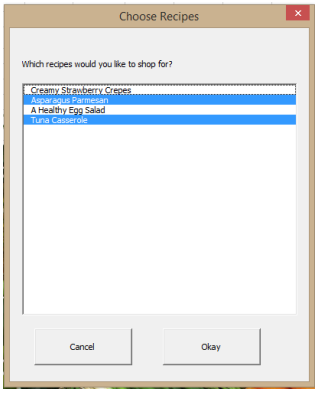

Figure 4: Choose Recipes UF

1) The program searches the “Recipe” worksheet for the highlighted recipes.
2) The program compiles a list of the ingredients of the selected recipes. This list is placed on the “lists” worksheet (see figure 4). The lists tab only exists as a place for the macro to edit the text. In this example, all of the recipes on the “add a recipe” user form were highlighted.

eggs	3 eggs
milk	1/2 cup
butter	1/2 cup
flour	3/4 cup
salt	1/2 tsp
cream cheese	8 oz
sugar	1.25 cups
lemon juice	1 tbs
lemon zest	1 tsp
vanilla extract	1/2 tsp
heavy cream	1 cup
strawberrys	4 cups
butter	1 tbs
olive oil	1/4 cup
asparagus	1 lb
parmesan cheese	3/4 cups
salt	1 pinch
pepper	1 pinch
hardboiled eggs	8 eggs
mayonaise	1/4 cup
fresh chives	1/4 cup
dijon mustard	1 tsp
celery salt	1/2 tsp
paprika	1/2 tsp
kosher salt	1/2 tsp
black pepper	1/2 tsp
egg noodles	12 ounce
frozen green peas	2 cups
c of mush soup	2 cans
tuna	2 cans
onion	1 onion
american cheese	10 slices

Figure 5: Ingredients worksheet

3) The program searches each ingredient name (not the amount) in the list to see if there are more than one of that particular ingredient.
4) If more than one of the ingredient exists, the program pulls all instances of that ingredient out of the list (see figure 6).

	M	N
	butter	1/2 cup
	butter	1 tbs

Figure 6: Two of same ingredient

5) Next the program breaks up the amount column of the pulled out section so it can see if the measurements are the same (see column O and P in figure 7). The program removes pluralization from the measurements so it will see things like cups and cup as the same.

M	N	O	P
butter	1/2 cup	0.50	cup
butter	1 tbs	1.00	tbs

Figure 7: split up amount

6) If the measurements are the same (eg, cup and cup) then the program adds the amounts of those measurements together. If they are different (eg, cup and tbs) then the program just concatenates them (in figure 8 they are being concatenated because they are different measurements).

	X	Y
	butter	0.5 cup + 1 tb

Figure 8: combine ingredients

7) The combined plural ingredients and the single ingredients are now moved to a new list. At this point, there is only one entry per ingredient (figure 9).

AA
eggs- 3 eggs
milk- 1/2 cup
butter- 0.5 cup + 1 tb
flour- 3/4 cup
salt- 0.5 tsp + 1 pinch
cream cheese- 8 oz
sugar- 1.25 cups
lemon juice- 1 tbs
lemon zest- 1 tsp
vanilla extract- 1/2 tsp
heavy cream- 1 cup
strawberrys- 4 cups
olive oil- 1/4 cup
asparagus- 1 lb
parmesan cheese- 3/4 cups
pepper- 1 pinch
hardboiled eggs- 8 eggs
mayonaise- 1/4 cup
fresh chives- 1/4 cup
dijon mustard- 1 tsp
celery salt- 1/2 tsp
paprika- 1/2 tsp
kosher salt- 1/2 tsp
black pepper- 1/2 tsp
egg noodles- 12 ounce
frozen green peas- 2 cups
c of mush soup- 2 cans
tuna- 2 cans
onion- 1 onion
american cheese- 10 slices

Figure 9: combined list

8) Now the user is prompted with the "remove items you already have" user form. Here the user can look at all of the items in the list and select those that they don't need to buy at the store because they already own them (figure 10). In this case, the user is removing the flour and the sugar because they already have them in their kitchen. Clicking the "cancel" button will terminate the list making. Clicking the "Okay" button will move the user to the next step.

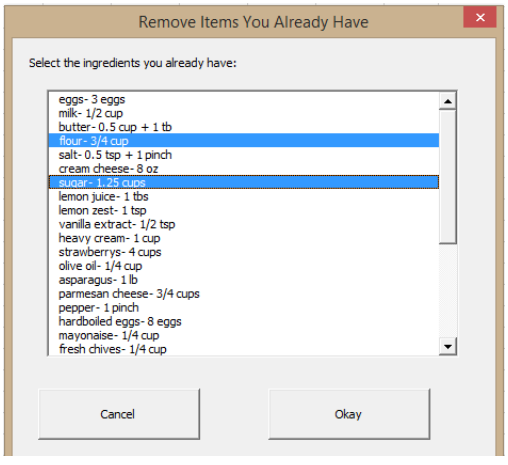

Figure 10: Remove items you have UF

9) After the user hits "okay" the highlighted items and amounts are removed from the grocery list.

10) Now the user is prompted with the "edit ingredients" user form which has a list box populated with the updated list created in steps 9 and 10. Here the user can choose to edit any of the ingredients on their list. This is useful if they need less of any of the ingredients because they already have some of that ingredient at home. To edit an ingredient the user highlights that ingredient and clicks edit (figure 11).

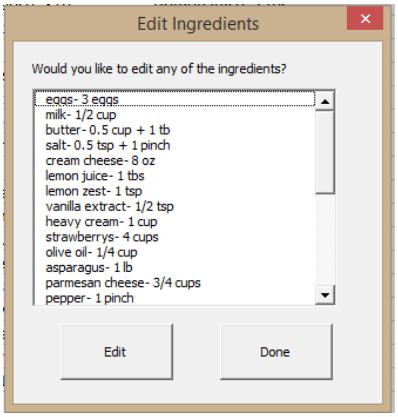

Figure 11: Edit Ingredients

11) After clicking edit, the user is prompted with a user form to edit that particular ingredient (figure 12). In this example, the user is going to decrease the amount of strawberries from 4 cups to 2 cups because they already have some at home. To do this they simply change the text and click done.

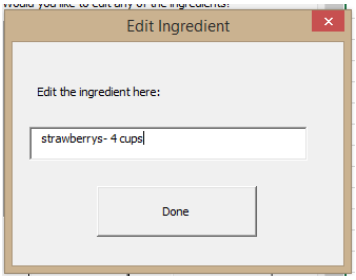

Figure 12: Edit ingredient box

12) Once the desired ingredients are edited, the user clicks the "done" button (see figure 11).

13) Once "done" is clicked, the program splits the food from the amount again and checks each food for a match on the "categories" worksheet in the workbook. The categories worksheet has a bunch of common foods and the section that they fall under in the grocery store (eg, fruits, vegetables, dairy, etc…). Figure 13 shows the categories worksheet. These categories can be edited to include more foods if the user wishes.

Vegetables	Fruits	Dairy Products	Meats	Spices	Grains
Asparagus	Apple	Cream	Bacon	Basil	pasta
Avocado	Apricot	Cheese	Beef	Black pepper	noodles
Beets	Banana	Milk	Chicken	Cilantro	Bagels
Bell pepper	Blueberries	Yogurt	Ground beef	Cinnamon	Buns
Broccoli	Cantaloupe	Ice Cream	Ham / Pork	Garlic	Cake
Brussels sprout	Cranberries	Butter	Hot dogs	Ginger	Pastries
Cabbage	Fig	Sour Cream	Lunchmeat	Mint	Fresh bread
Carrot	Grapefruit	Buttermilk	Turkey	Oregano	Pie
Cauliflower	Grapes	Creme Fraiche	Catfish	Paprika	Pita bread
Celery	Kiwi fruit	Clotted Cream	Crab	Parsley	bread
Collard greens	Lemons and limes	Kefir	Lobster	Red pepper	Rolls
Cucumber	Orange	Condensed and Evapor	Mussels	Salt	cookies
Eggplant	Papaya	Powdered Milk	Oysters	Vanilla extract	Donuts
Fennel bulb	Pear, Bartlett	Mascarpone	Salmon	cayenne pepper	croissants
Garlic	Pineapple	Bleu cheese	Shrimp	pepper	rice
Green bean	Plum	Cheddar	Tilapia	ground pepper	egg noodles
Green pea	Prune	Cottage cheese	Tuna	kosher salt	
Kale	Raisins	Cream cheese	ground turkey		
Leeks	Raspberries	Feta	sausage		
Mushrooms, Crimini	Strawberries	Goat cheese	ribs		
Mushrooms, Shiitake	Watermelon	Mozzarella			
Mustard greens		Parmesan			
Olive		Provolone			
Onion		Ricotta			
Parsley		Sandwich slices			
Potato		Swiss cheese			
Romaine lettuce		parmesan cheese			
Sea vegetables		american cheese			
Spinach					
Squash, summer					
Squash, winter					
Sweet potato, with skin					
Swiss chard					
Tomato, fresh					
Turnip Greens					
Yam					

Figure 13: Categories Worksheet

14) If the food is found in the category worksheet, the program assigns that food a number based on which column it fits in (see figure 14). If the food cannot be found the category worksheet, the program assigns that food a 7 and will put it in the "other" category of the grocery list.

3.00	american	10 slices
1.00	asparagus	1 lb
5.00	black pep	1/2 tsp
3.00	butter	0.5 cup + 1 tb
7.00	c of mush	2 cans
7.00	celery salt	1/2 tsp
3.00	cream che	8 oz
7.00	dijon mus	1 tsp
6.00	egg noodl	12 ounce
7.00	eggs	3 eggs
7.00	fresh chiv	1/4 cup
7.00	frozen gre	2 cups
5.00	ground pe	1 pinch
7.00	hardboile	8 eggs
7.00	heavy cre	1 cup
5.00	kosher sal	1/2 tsp
7.00	lemon jui	1 tbs
7.00	lemon zes	1 tsp
7.00	mayonais	1/4 cup
3.00	milk	1/2 cup
7.00	olive oil	1/4 cup
1.00	onion	1 onion
5.00	paprika	1/2 tsp
3.00	parmesan	3/4 cups
5.00	pepper	1 pinch
5.00	salt	0.5 tsp + 1 pinch
7.00	strawberr	2 cups
4.00	tuna	2 cans

Figure 14: Ingredients with Categories

15) At this point all the ingredients have been combined and are matched to a food category. Now the program goes back to the main page and creates the grocery list next to the buttons. The ingredients are placed in different categories in the list based on what number they were assigned (figure 15).

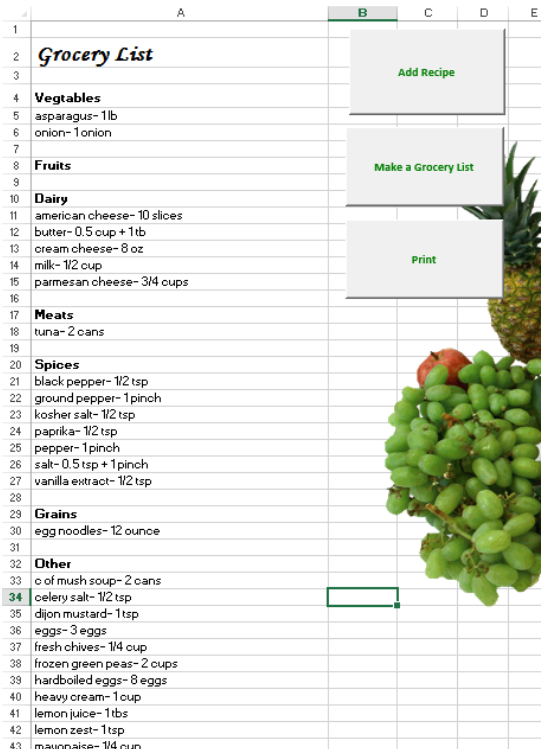

Figure 15: Final Grocery List

Printing the Grocery List

Now the user only needs to click on the "print" button on the main page (see figure 15) and the grocery list will automatically print. The picture and buttons do not print with the list.

Learning and Conceptual Difficulties

What I learned

Above all else, I learned just how much time it takes to do a project of this magnitude. Before starting the project I was confident that I would be able to complete the program after just a couple days of work. I was totally wrong. This project took me weeks to finish and gave me a new respect for just how much programing it takes to accomplish a relatively simple task.

Along the same lines, doing this project taught me the importance of weighing the pain of the current solution vs the pain involved in programing a better solution. For example, my program makes creating a grocery list very easy and will probably save me a return trip to the grocery store every month. I have to ask myself though, did I spend more time on the program than I would have spent making lists and returning to the grocery store? It will probably take a few years of using this worksheet for me to actually have saved myself time. Sometimes the imperfect current solution is better than the final solution.

While working on the project I learned a ton about the importance of breaking up sub procedures and not just trying to make one giant sub procedure. Breaking up the sub procedures makes it much easier to troubleshoot the program and find the errors. It also makes the program less overwhelming and easier to follow.

Conceptual Difficulties

Thankfully, I was able to build all of the elements in my initial proposal and they all worked like I wanted. Some things did take quite a while for me to figure out. I often had to build loops within loops within loops when checking ingredients for duplicates. This required a lot of thought and concentration. I found the easiest way to do this is to start with the inner-most loop and move outwards.

One of my biggest problems was finding an efficient way to move the entered recipes from the "add a recipe" user form to the spreadsheet. I ended up just having to write the specific name of each and every textbox and directing its value to a spot in an array because I was unable to loop through the textboxes. I also found the user forms a little tedious and difficult to work with. Every time that I accidentally double-clicked on an object in the user form it would open a sub procedure for that action.

The last and most challenging problem that I had while building this program was following all of the code that I wrote. As I mentioned above, breaking up the sub procedures helped, but I still had a hard time following my code through all the user forms. It would start in a macro and then bring up a user form and then go through the okay button to another user form and then go through that user form's okay button and so on. It became really difficult to find certain parts of the code for editing purposes.

Assistance

I received no assistance as I built my project.

Appendix

Blake Laidlaw 2/25/15

VBA Project Proposal

Everyday hundreds of Americans get home after grocery shopping only to realize that they have forgotten to purchase one or more essential items for the upcoming week. Too often, I have found myself in this exact predicament. The only solutions are to either go back to the store or revise your meal plans for the upcoming week.

I want to solve this problem for myself and others by creating a spreadsheet that will eliminate this problem by doing the following:

1. Users will input their favorite meals and the required ingredients into a user form
2. When it is time to go to the grocery store, users will be prompted with the list of the meals they entered into the user form. They will check those that they wish to make for the upcoming week.
3. The spreadsheet will use VBA to compile a list of all necessary ingredients (and their quantity) that the user will need to make their selected meals.
4. At this point the user can go through and uncheck the ingredients that they already have or modify the ingredient amounts
5. The spreadsheet will then group all of the items based on the food type (eg. dairy, produce, etc…)
6. The spreadsheet will print out a finalized grocery list