

# Recipe Shopping List Creator

---

Sam Morton

4/13/2011

# Executive Summary

---

For some, cooking is an undesirable activity necessitated by the laws of nature and the rules of scarcity. Worse yet, the act of cooking is preceded by the preparation as to what to cook and a subsequent trip to the grocery store. Preparation for the grocery store trip involves perhaps the most distasteful act of the whole process: choosing recipes and rereading each over and over while trying to determine if the required ingredients are in stock or need to be purchased.

The Recipe Shopping List Creator is meant to make easier the preparation for the trip to the grocery store. The user can choose from a list of previously entered recipes, enter a new recipe, or import a recipe from AllRecipes.com. Once the recipes to make have been chosen, one click will aggregate the ingredients into an easily readable list. The program will even add amounts of the same ingredients together. After making the shopping list, printing and going to the grocery store are all that stand in the way of practicing to become the next Emeril.

# Documentation

---

The program consists of four main parts:

- Worksheets
- Ribbon
- Add Recipe
- Search\Create Shopping List

## Worksheets

The worksheet has 3 worksheets that hold data necessary for the program's function:

- Recipes
- Measurements
- Equivalents

The Recipe worksheet contains the list of recipes that the user has entered or imported. The first column displays the recipe name, the second holds the recipe directions, and the remaining rows hold the ingredient information. The ingredient columns are laid out as follows: the first column contains the quantity and unit of measure, such as "1 cup", while the next column holds the ingredient name, "flour". This pattern is repeated for as many ingredients there are in the recipe. The screenshot immediately following shows the "Recipe" sheet. In order to remove a recipe, the user must delete the row from this worksheet.

	A	B	C	D	E	F	G	H
1	Recipe Name	Directions	Qty1	Ingredient1				
2	Magic Beans	testset	1 cup	chicken	3 cup	beans		
3	Magic Salsa	testset233	1 cup	magic	1 cup	salsa		
4	Salsa Chicken	Put it in the crock pot	1 lb	chicken	1 cup	salsa	1 can	corn

The Measurements worksheet contains a list of measurement names and an equivalent name that the former will be replaced by. For example, seen in the figure below, the measurement "c" meaning cup, will be recognized by the program and be changed to "cup."

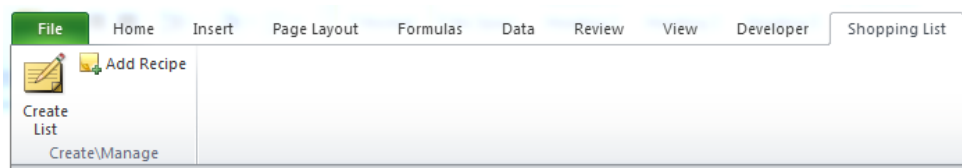
	A	B
1	Measurement	Conversion
2	c	cup
3	cups	cup

The Equivalents worksheet shows measurements and their equivalents in other units of measure. For example, 16 ounces equals 1 pound. The first column shows the smaller unit of measure, the next column shows the amount of the smaller unit, and the third column shows the equivalent larger unit of measurement. It is important to note here, as later, that the program's use of this

worksheet is somewhat limited at present time due to the nature of solid versus liquid measurements and ingredients. For instance, a pint may indeed be equal to two cups, but when buying a liquid ingredient at the store, one would probably not see it measured in cups, but rather it would be measured in pints, quarts, fluid ounces, gallons, or some other liquid measurement. To avoid having to hard-program in all the liquid versus solid ingredients, this functionality is limited to only the most obvious conversions.

## Ribbon

The ribbon tab entitled “Shopping List” is the entry point for the program and consists of the Search\Create Shopping List and Add Recipe buttons.



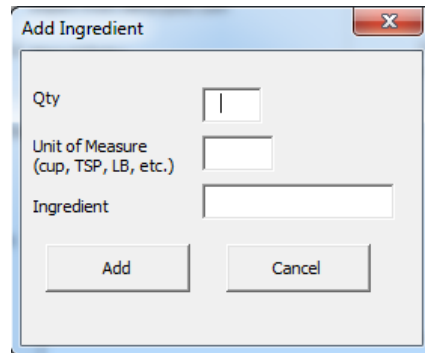
## Add Recipe

The “Add Recipe” dialog, shown below, gives the user two choices for adding new recipe “Manual Entry” and “Import from AllRecipes.com”

If the user leaves it on the default setting, “Manual Entry,” the user can enter the name of the recipe and the directions for the recipe in the respective boxes.

To add ingredients to the recipe, the user clicks the “+” icon, which will pop up another input screen, enabling the input of an ingredient quantity, a unit of measure (cup, teaspoon, tablespoon,

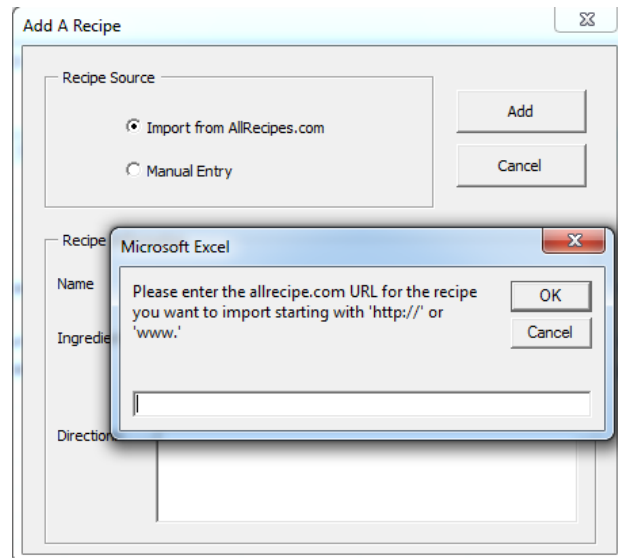
etc), and the name of the ingredient (chicken, mozzarella cheese, flour, etc). This screen is seen below.

A dialog box titled "Add Ingredient" with a close button (X) in the top right corner. It contains three input fields: "Qty" with a vertical line cursor, "Unit of Measure (cup, TSP, LB, etc.)", and "Ingredient". Below the fields are two buttons: "Add" and "Cancel".

To remove ingredients, select an ingredient in the box and press the “-“ button.

Finally, to add the recipe, click the “Add” button and the recipe will be added to the worksheet.

Selecting the “Import from AllRecipes.com” choice immediately brings up a dialog where the user can enter the URL of the recipe he or she wishes to import. Selecting “OK” closes the dialog and adds the name, ingredients, and directions from the recipe on AllRecipes.com to the “Add Recipe” dialog. The user then reviews and edits where needed using the methods described previously.

A screenshot showing two overlapping dialog boxes. The background dialog is titled "Add A Recipe" and has a "Recipe Source" section with two radio buttons: "Import from AllRecipes.com" (selected) and "Manual Entry". It also has "Add" and "Cancel" buttons. Below this are fields for "Recipe Name", "Ingredients", and "Directions". Overlaid on top is a smaller "Microsoft Excel" dialog box with the text: "Please enter the allrecipe.com URL for the recipe you want to import starting with 'http:/' or 'www.'" and "OK" and "Cancel" buttons.

## Create List

The Search\Create Shopping List dialog has four major components and is shown in a picture below:

- Search

- Recipe List
- Recipes To Make
- Recipe Details

## Search

The list of recipes in the “Recipe List” box can be searched using the text boxes at the top: one searches by name and the other by ingredient.

## Recipe List

Each recipe in the worksheet is populated in the “Recipe List” box by the recipe’s name using the row source property. The Recipe List is shown in the left box in the screenshot above.

## Recipes To Make

The box on the right of the dialog is the recipes to make box. Recipes can be added to the “Recipes to Make” box by using the “>” and “<” buttons located between the “Recipe List” box and the “Recipes To Make” box. Selecting a recipe in the “Recipe List” box and clicking the “>” button will add the recipe to the “Recipes To Make” box on the right. Recipes in this box will have their ingredients included in the shopping list when the user clicks end.

## Recipe Details

Once a recipe is selected, its details will display in the name, directions, and ingredients boxes at the bottom of the dialog. The recipe can be edited by clicking the “Edit” button on the right. Clicking the “Edit” button will unlock all the fields, enable the “Add Ingredient” and “Remove Ingredient” buttons, and change the text of the “Edit” button to “Save”. Editing the name,

directions, or adding or removing ingredients and clicking the “Save” button will save the changes to the worksheet.

## Creating the Shopping List

Finally, the user can click the “End and Create Shopping List” button to end the recipe choosing process and create the shopping list. The shopping list creation involves collecting the ingredients from the chosen recipes, reconciling the units of measure with known units of measure found on the “Measurements” worksheet, adding measurements for like ingredients together, and finally outputting the results to a worksheet. When the shopping list is created, a message will pop up informing the user the name of the sheet containing the shopping list, which will include the recipes chosen, the date the shopping list was created, and the list of ingredients. A shopping list created for “Fabulous Wet Burritos” is shown below.

1	Shopping List - 4/12/2011	
2		
3	Recipes	
4	1x Fabulous Wet Burritos (Row 7)	
5		
6	Ingredients	
7	1 lb. ground beef	
8	0.5 cup chopped onion	
9	1 whole clove garlic, minced	
10	0.5 tsp cumin	
11	0.25 tsp salt	
12	0.125 tsp pepper	
13	4.5 oz. can diced green chile peppers	
14	16 oz. can refried beans	
15	15 oz. can chili without beans	
16	10.75 oz. can condensed tomato soup	
17	10 oz. can enchilada sauce	
18	6 whole inch flour tortillas, warmed	
19	2 cup shredded lettuce	
20	1 cup chopped tomatoes	
21	2 cup shredded Mexican blend cheese	
22	0.5 cup chopped green onions	

## Lessons Learned

The most difficult part of this project was all the string manipulation that was required. When importing from AllRecipes.com, the program has to determine from the list of ingredients where the quantities, units of measure, and ingredients are. Determining where each of these is in the ingredient string requires a variety of instr, left, right, and mid, among other techniques. On top of that, sometimes an ingredient doesn’t have an explicit unit of measurement as in the case of “1 garlic clove.” Additionally, because of a design choice made early on regarding the columns for the recipes, a similar, though less intense, process has to be done when assembling the shopping list. In the end, however, it all works somewhat well, and I definitely learned some tricks for working with strings and decided it isn’t the most fun thing to do.

I also learned ever more about user interfaces and particularly about the listbox. I tried a variety of methods for populating the listbox before deciding on one that met the needs of the project and was flexible to various things that a user could do. For example, in the event that a user has several recipes for the same basic dish, and names them the same thing, I needed a way to keep track of which was which in both of the listboxes. Also, when the user edited the name of a recipe I wanted a quick, easy way to propagate that change back into the listbox displaying the recipes.

A final interesting lesson learned was how to use a custom dialog to return information on many variables. In creating a user form for ingredient input, I had three pieces of information I wanted returned. An inputbox will only return one, so I made my own. I included a sub called “popup” that showed the ingredient input box, with its three inputs, and took several variables ByRef. Once the user pressed add or cancel, the inputs would be set and “returned” to the sub that popped up the add ingredient dialog box.

Overall it was a fun, challenging, and definitely time-consuming project. It enabled me to put to use concepts learned in the class in a project of my choosing. These concepts were reinforced as I constantly had to think how to apply them to my project and in the most effective way possible.