Dinnertime!

Project Report VBA Winter 2015

Courtney Llera

4/11/2015

Table of Contents

Executive Summary…… 3

Implementation

Main Execution…… 4

Dining In…… 4

Dining Out…… 6

Lessons Learned and Difficulties Defined…….. 8

Assistance Received…… 8

Executive Summary

The user of this project is not a business, but anyone who has ever asked the question, “What’s for dinner?” Whether the user is single or married, after a long day, it’s the last question we want to make a decision about. And so, I created Dinnertime to enable users to either select an existing recipe through the Dine In functionality or query Groupon.com or KSL.com for restaurant deals in the area through the Dine Out interface.

To dine in, users must first enter their favorite recipes into the program on a designated sheet in the workbook. The user can either enter ingredients and ethnic category they are in the mood for or they can leave the selection process up to the user. Three random options can be generated in the event that no categories are chosen. Once a recipe is selected, additional details (instructions, ingredients, and other information from the list of recipes) will be shown in a new userform.

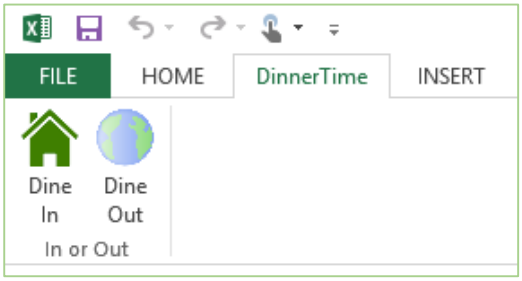
To dine out, a userform appears that enables searching of Groupon.com by location, then by genre (or ethnicity) according to what is available in that area. This functionality is accomplished through a web query, followed by a subprocedure that opens the appropriate link in an IE browser. Alternatively, if the user lives in Utah, they can enter their zip code to find local restaurant deals through KSL in an IE browser.

Implementation

Creation of the Dinnertime program required three development phases: (A) main execution, (B) Dining In, and (C) Dining Out.

Main Execution

Before writing functional code, the Custom UI Editor for Excel was downloaded and installed from Ron de Bruin's website to create two buttons on the excel menu ribbon. The new buttons and Dinnertime tab are pictured at the right.



Dining In

The "Dining In" feature lets users select a dinner option from a list of recipes (entered manually by the user) in the program either randomly or purposefully. A "Recipes" tab was created to store variables such as title, cook-time, notes, and more. A screenshot of the table is shown in Figure 1.

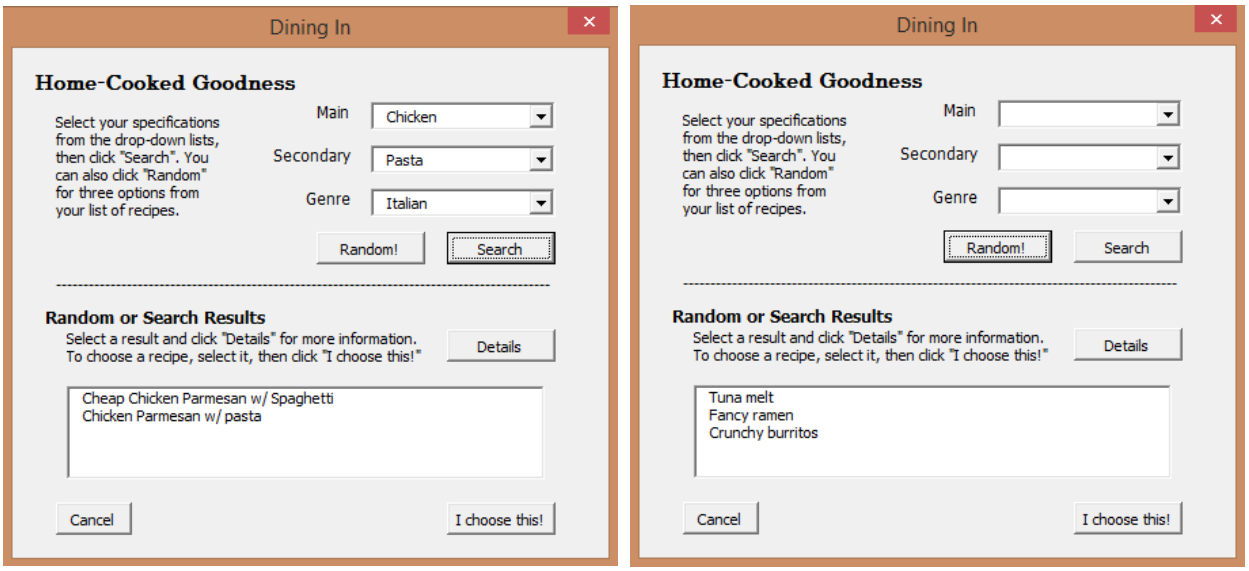
Figure 1: Screenshot of Recipes Tab with existing data (simulates user's entries)

	A	B	C	D	E	F	G	H	I	J
1	Recipe Title	Main ingredien	Secondary Ingr	Genre	Favorite	Time to	Last Cooked	Ingredients	Instructio	Notes
2	Tortellini w/ Peas & Mushrooms	Pasta	Vegetables	Italian	Yes	20 min	1/20/2015	1 pkg frozen	Melt butt	Boil the pe
3	Cheap Chicken Parmesan w/ Spagh	Chicken	Pasta	Italian	No	20 min	3/2/2015	Chicken patt	Cook chicl	Real parm
4	Crunchy burritos	Beef (ground)	Vegetables	Mexican	Yes	30 min	2/14/2015	1 lb ground	Heat vege	Have salsa
5	Beef lo mein	Beef	Pasta	Asian	Yes	60 min	3/27/2015			Add extra
6	Chicken and broccoli ring	Chicken	Vegetables	American	Yes	30 min	3/27/2015	2 cups brocc	mix aaaaaaaaaall	the
7	Fancy ramen	Pasta	None	Asian	Yes	15 min	12/30/2014	2 pkg Ramer	Chop gree	Can exper
8	Tuna melt	Tuna (canned)	None	American	No	15 min	4/10/2014	1 can tuna, s	Mix desired amount	
9	Pierogies with Sausage	Sausage	Pasta	Polish	Yes	20 min	4/20/2015	2 boxes froz	Cook pierogies accor	

Instructions | Recipes | GrpOnLoc | webQuery | FlavorPLoc

After clicking the "Dine In" button on the ribbon, a userform (Figure 2) will appear with options for narrowing the dinner choices. Users select options for main ingredients, secondary ingredients, and food type. The screenshots below show the results for a normal search (with criteria) and a random search.

Figure 2: User's View of Dining In userform after clicking "Search" (left) or "Random" (right)



Each element type, name, and purpose (except labels) in the Dining In userform is explained in Table 1.

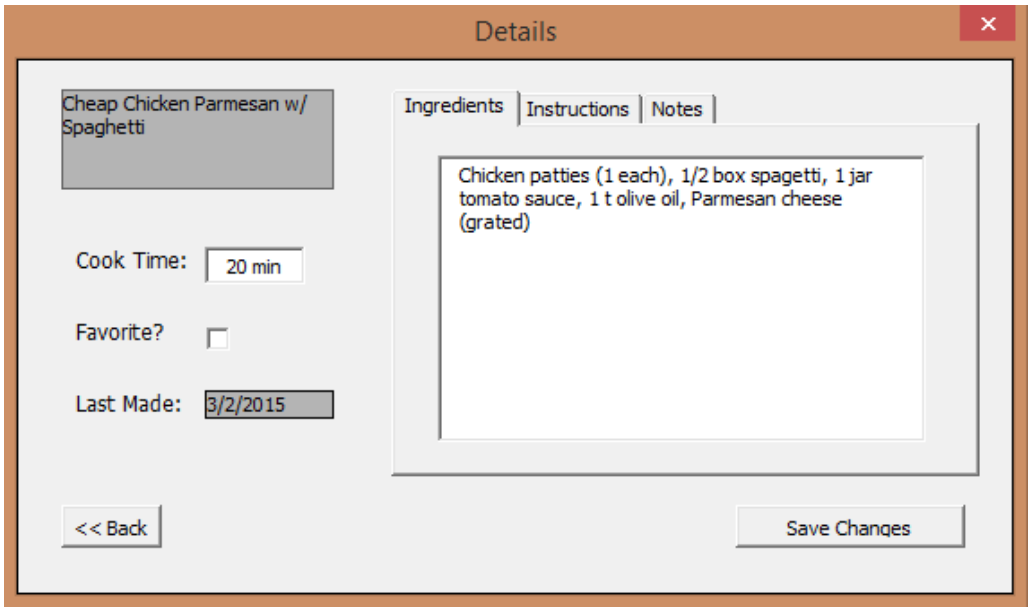
Table 1: Dining In Userform Elements and Uses

Element Type	Element Name	Purpose
Combo box (3)	Main	Enables user to select main ingredient (usually meat) for search
	Secondary	Enables user to select secondary ingredient for search
	Genre	Enables user to select food type (ethnicities) for search
List box	Results	Lists results of searches both random or set
Button (5)	"Random!"	Generates 3 different digits between 1 and the number of occupied rows in "Recipes" tab; the entries that correspond with the 3 numbers are shown in the results box
	"Search"	Identifies all existing recipes that match criteria set by the user in the main, secondary, and genre combo boxes
	"Details"	Displays recipe information (cook time, last made, ingredients, instructions, etc.) for the selected option from the results box
	"I choose this!"	Executes a subroutine that sets the "last made" variable to today's date; this action can be done before or after the recipe's details have been viewed or saved by the user
	"Cancel"	Closes the "Dining In" window and ends the program

To view information on a specific recipe, then clicks the "Details" button. A "Details" userform (Figure 3) appears and enables users to view or change all information except the title

of the recipe and the date on which that recipe was most recently cooked. When the user clicks "Save Changes", alterations made to any editable field in the Details userform is saved and a messagebox confirms the action. The user is then instructed to keep the window open as a reference while cooking the dish.

Figure 3: Details Userform



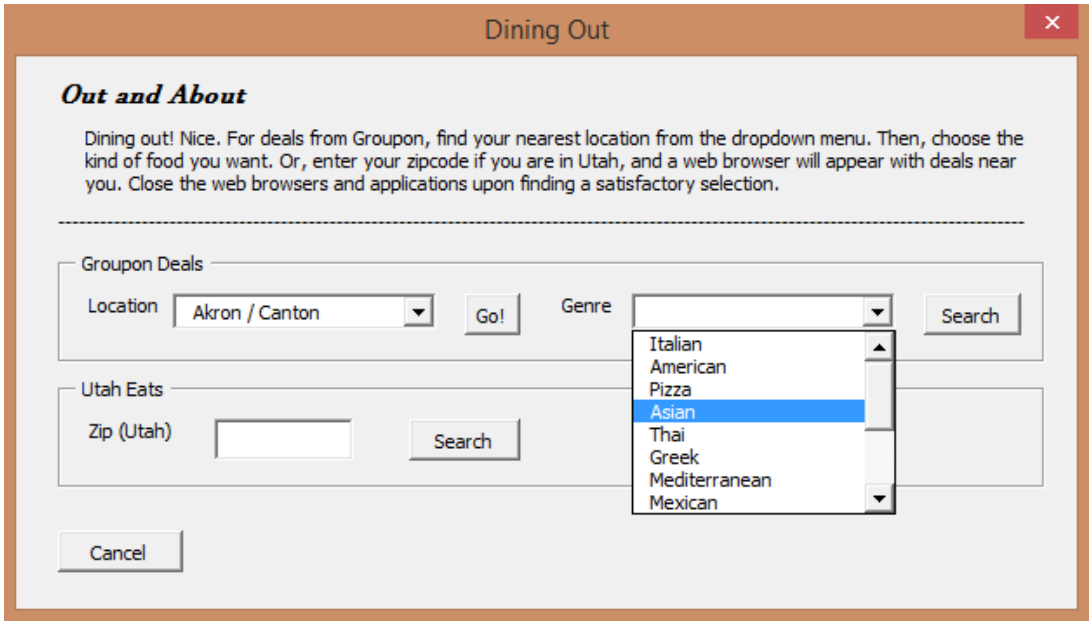
Recipe title is not editable in order to protect accidental deletions from occurring; if the field is left blank, users may not see it as an option in the results box of the Dining In userform. The "last made" variable is static in the form because it is automatically set when the user selects it as dinner for that night with the "I choose this!" button.

Dining Out

Originally, two web queries were to be used in retrieving information from Groupon.com and another restaurant-finding website; the information was to be shown in the Dinnertime program. The finished program includes the Groupon.com query, but swaps retrieval for display in IE browsers. For Groupon, the user selects a location, then presses "Go!" to execute the web query. The query retrieves data from Groupon.com about the restaurant types available in that area. The resulting list is parsed and loaded into the Genre combo box. Then, the user chooses the type of food they want and clicks "Search". The subprocedure that executes upon click

action takes the selected location and genre and places the values into a url which opens to the Groupon page. The userform is shown in Figure 4 below.

Figure 4: Dining Out Userform (After first Groupon webquery)



Instead of utilizing two different interfaces to accomplish similar web queries, it seemed more prudent to consolidate both Groupon and KSL search functionalities onto one space. Each element in the Dining Out form is detailed by type, name, and purpose in Table 2. Groupon items are highlighted in green and KSL in blue; general elements remain colorless.

Table 2: Dining In Userform Elements and Uses

Element Type	Element Name	Purpose
Text box	Zip (Utah)	Enables user to enter their zip code to find local food deals on KSL
Combo box (2)	Location	Enables user to choose from a preset location list on the "GrpOnLoc" sheet; location is used to query Groupon.com for available genres in that area
	Genre	Enables user to select a genre from the list pulled from the location query; value chosen in genre combo box is used in new webquery to open IE browser
Button (4)	"Go!"	Activates first web query that retrieves Groupon information for the location chosen by user
	"Search"	Activates second web query that opens IE browser to the link that uses location and genre chosen by user in combo boxes
	"Search" (2)	Opens web browser that navigates to ksl restaurants page for the zip code entered
	"Cancel"	Closes the "Dining Out" window and ends the program

Lessons Learned and Difficulties Defined

Difficulties Defined

Difficulties encountered in this process are listed in Table 3 below with the method used to handle them.

Table 3: List of Difficulties Experienced Through Project Development

Problem	Method Used to Solve
Groupon.com does not permit importing of specific page elements	Imported entire page of data into "webQuery" worksheet within project for additional pre-parsing in "FlavorPLoc" (flavor per location) sheet
Groupon.com restaurant categories appear in inconsistent order per each location	Set the beginning of selection to be parsed to the keyword "Restaurants", then checked to see if "Beer, Wine, & Liquor" existed. If so, I set it to be the end of the selection. If not, "Location" became the end and further non-category items were parsed out of the category list.
Groupon.com does not permit some locations to be viewed upon opening the IE browser to correct link	Attempted to alter the link to go to the proper page instead of a splash screen for user email entry, but it was not successful.
Data retrieved from Groupon required removal of empty lines and parens	Looped through the list of categories and removed each empty line with an if statement. Used "Instr" to remove all parenthesis.

Even after project completion, there is one outstanding error that I cannot fix: Groupon's inconsistent placement of categories in the left-hand navigation per location prevents the category list generated in Dinnertime from being completely non-category-free. Although I have written a large If-Elseif-Else statement to handle removal, I only was able to account for about 10 extraneous cases.

Lessons Learned

I wish I had known about Groupon.com's background structure before committing to using it as a resource; I also wish I would have paid attention to its growing unpopularity and unreliability. Several peers with whom I spoke about my project doubt that Groupon will still be operational once its competitors prove to be dependable (Amazon and Google). So, this code may require updating if Groupon becomes bankrupt.

Assistance Received

I received no substantial assistance from others, but I did receive some error handling suggestions from Jordan Carlson, a teammate in two other IS classes. I had difficulties trying to troubleshoot Groupon's web query for locations in which "Beer, Wine, and Liquor" were not available. He helped me find a website which suggested three different methods of error handling. Aside from the resource website Jordan suggested, I completed the assignment alone.