**The Party Planning People’s Great Gathering GUI: User’s Manual**

**By Rebecca Scanlon and Cecille Yang**

Our program is called *The Party Planning Peoples’ Great Gathering GUI.*We have decided for our final project to create a party food planner.  The program will allow users to add in people to their contact list and then add specific people to a particular party.  Each person will contain a list of foods that they like and foods they are allergic to.  Once the party planner adds people to her party, the program will produce a list of foods that are recommended to have based on how many people like the food while making sure that if anyone is allergic to the food the user is notified.  In addition, the party planner will have a search feature that allows the user to access recipes for favored food items in each party via google search.

Features:

* ability to save contacts and load them, to save time
* ability to save a set of parties for loading, editing, and reference in the future
* ability to select suggested foods and search for them on google search



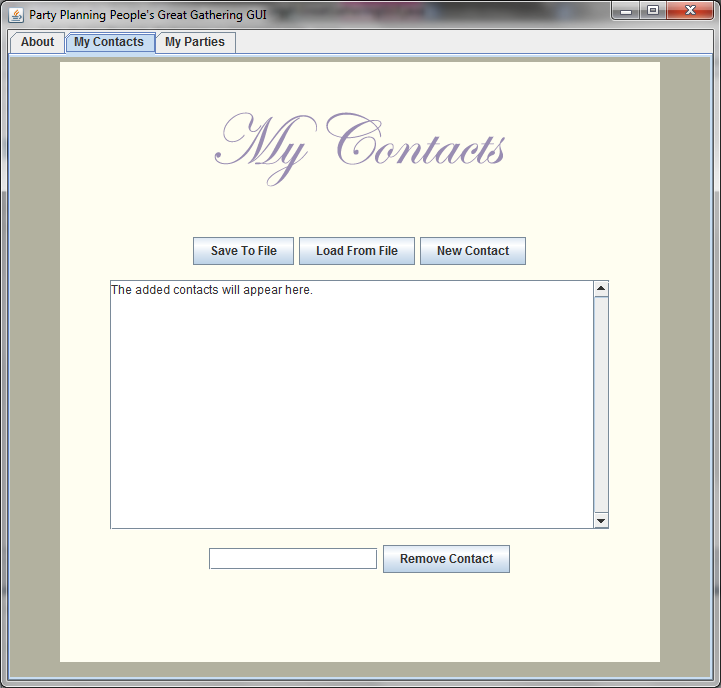
**Description:**

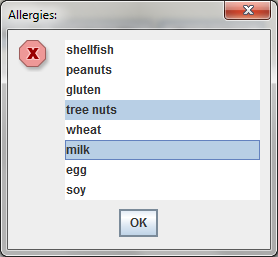
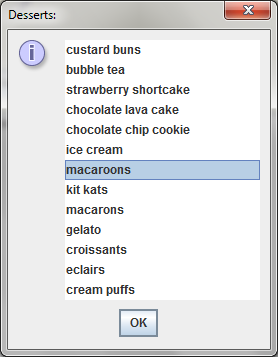
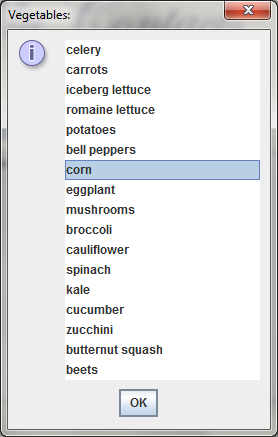
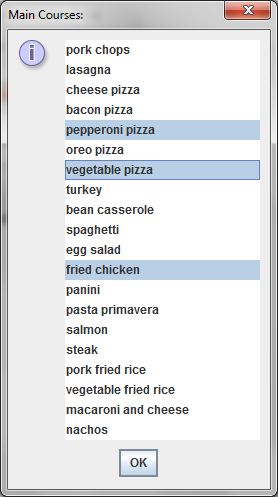
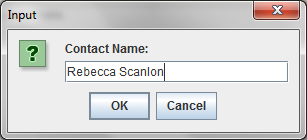
The party planning people’s great gathering GUI is run through two main tabs: contacts and my parties.

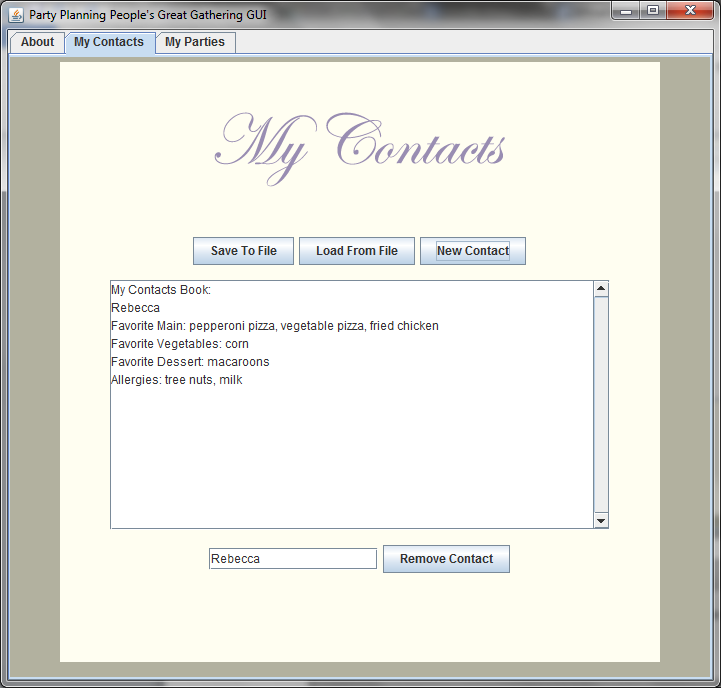
**Contacts Tab:**

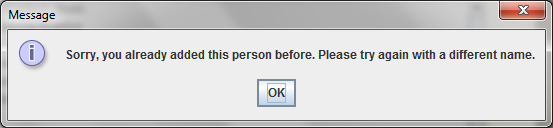
The contacts tab allows the user to create new contacts by clicking the “New Contact” button. After pressing this button the user will be prompted to add a name of a friend and then will be given lists of main dishes, then vegetables, then desserts of which the user can choose all of the friend’s favorite foods. If the user tries to add two friends with the same name, they will be prompted to add the second friend with a different name. On the contacts tab, the user can save all of the contacts they added so that the next time they enter the program they can load this file and all of their contacts will be loaded.

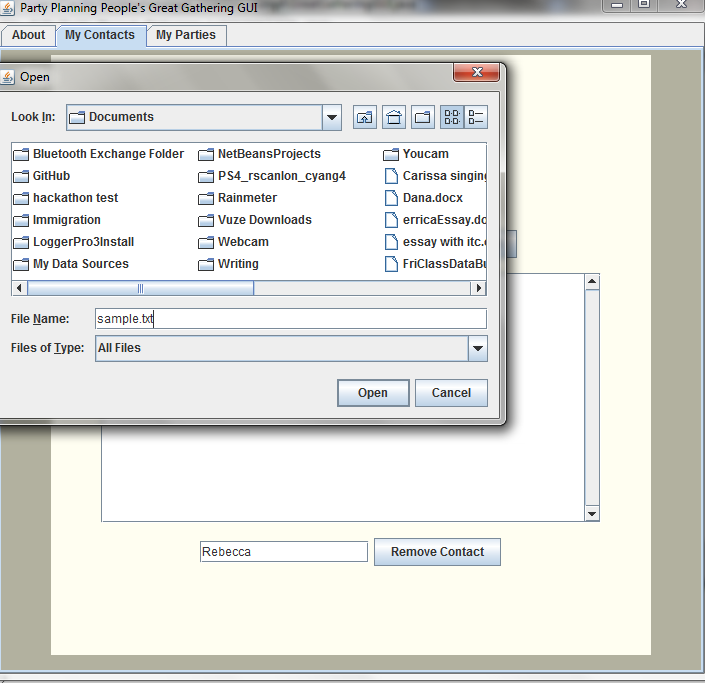
You can also delete a contact by typing in the person’s name and hitting “Remove”.

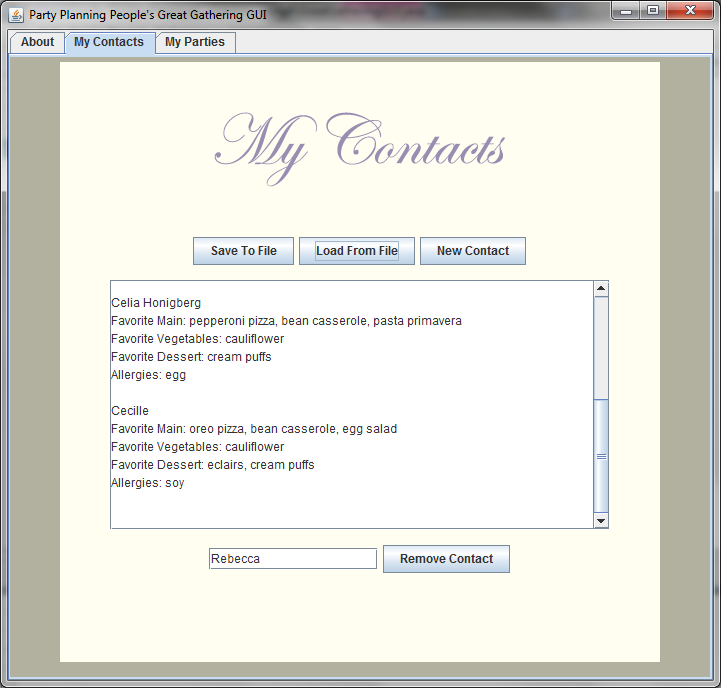






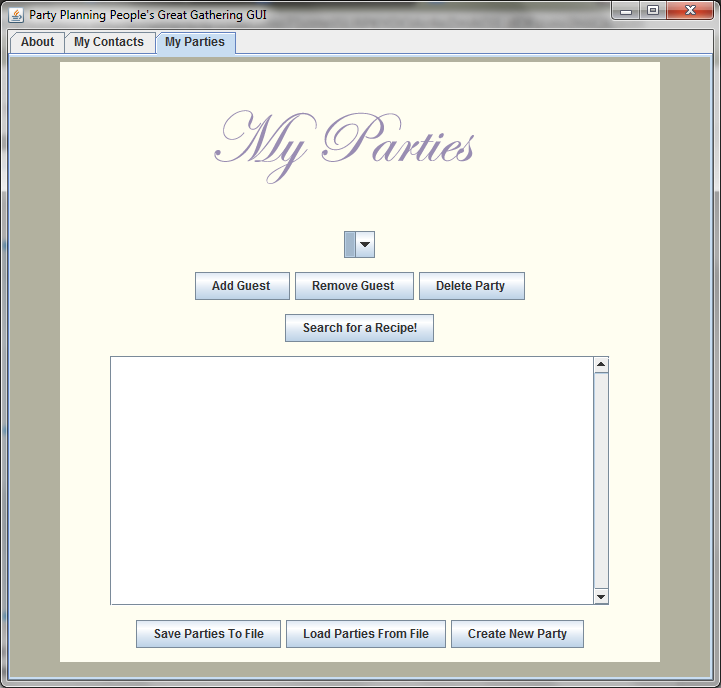


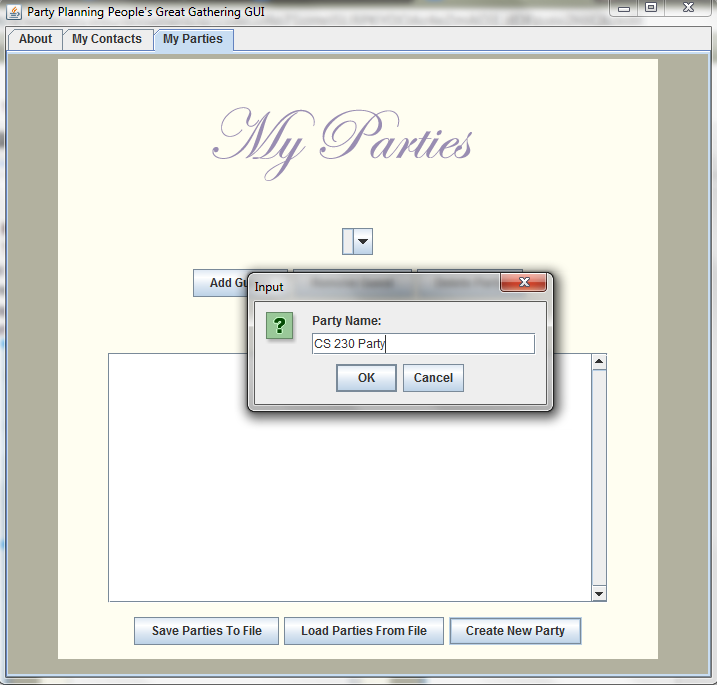




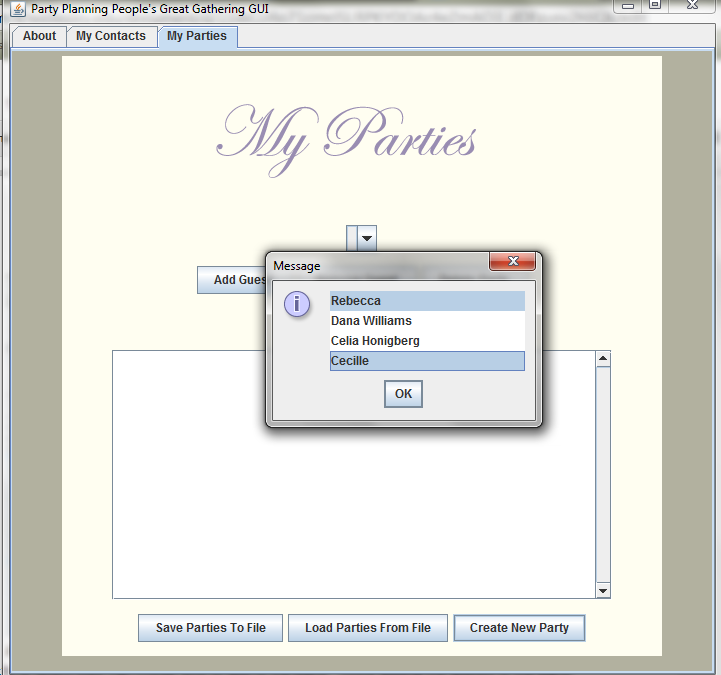
**Parties Tab:**

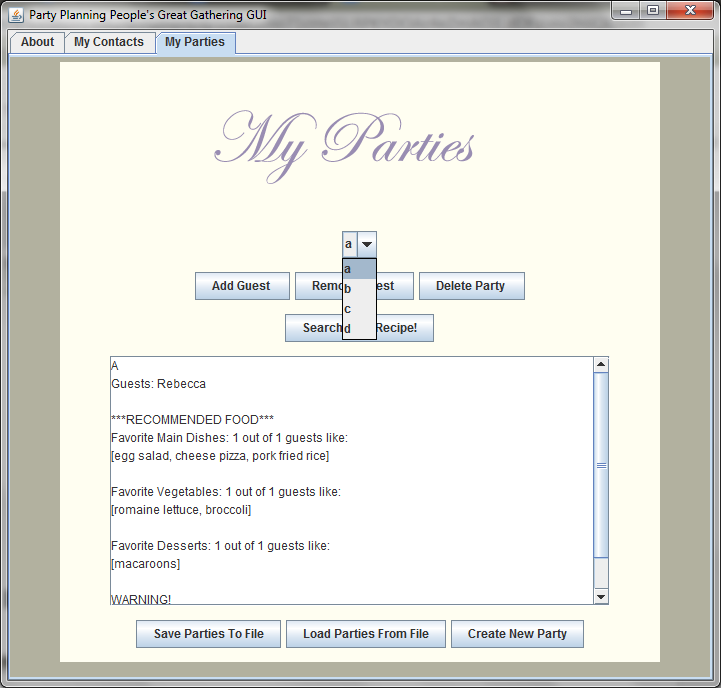
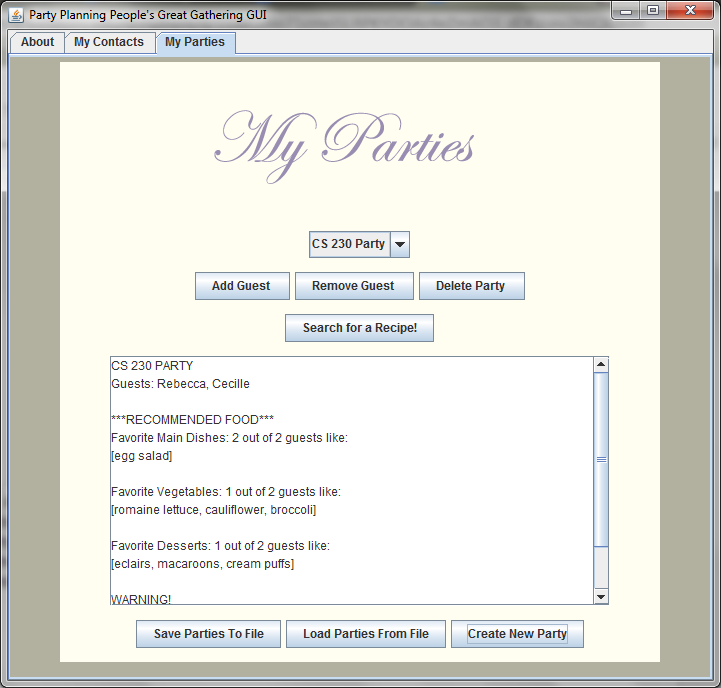
The my parties tab keeps track of all of the parties the user has made with the suggested recipes for each party. Similarly to the contacts tab, the user can create a new party by click on the “Create a New Party” button. Once clicked, the user will be prompted for the name of the party (again, parties must have different names and the user will be notified if they try to make two parties with the same name).



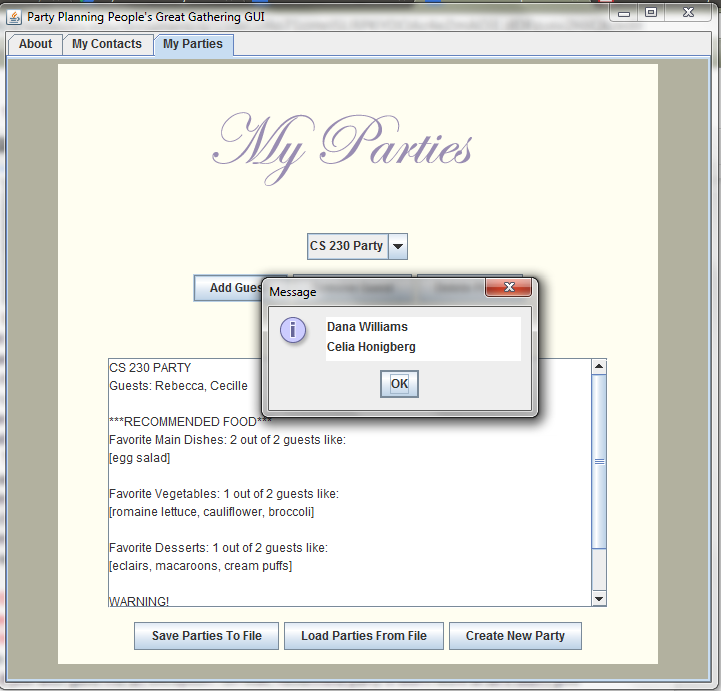


After entering the name of the party, the user will be presented with a list of all of their contacts of which they can chose who to add to the party. Once adding the guests to the party, the party will appear in the combo box to allow the user to access and change the information about the party and the suggested menu will appear in the text area. Similarly, to the contacts, the parties can be saved and loaded from a file by clicking on the respective buttons.

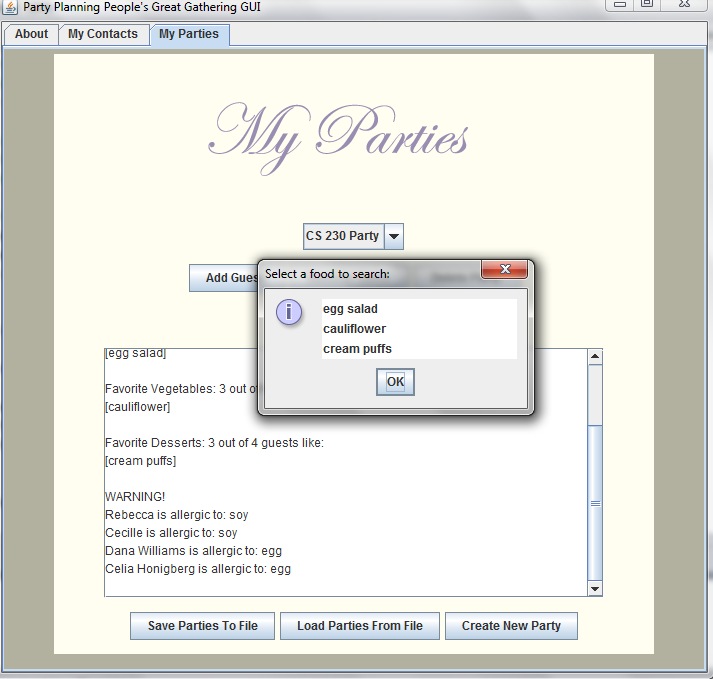
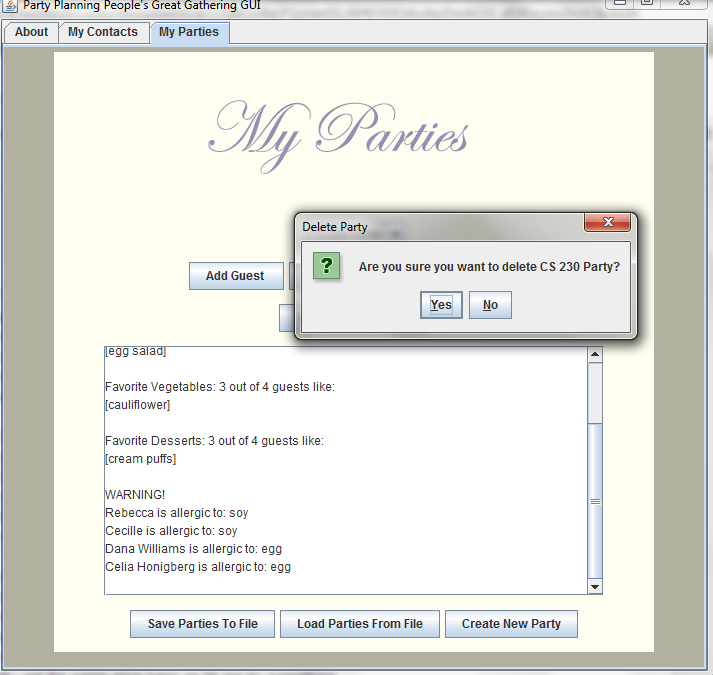


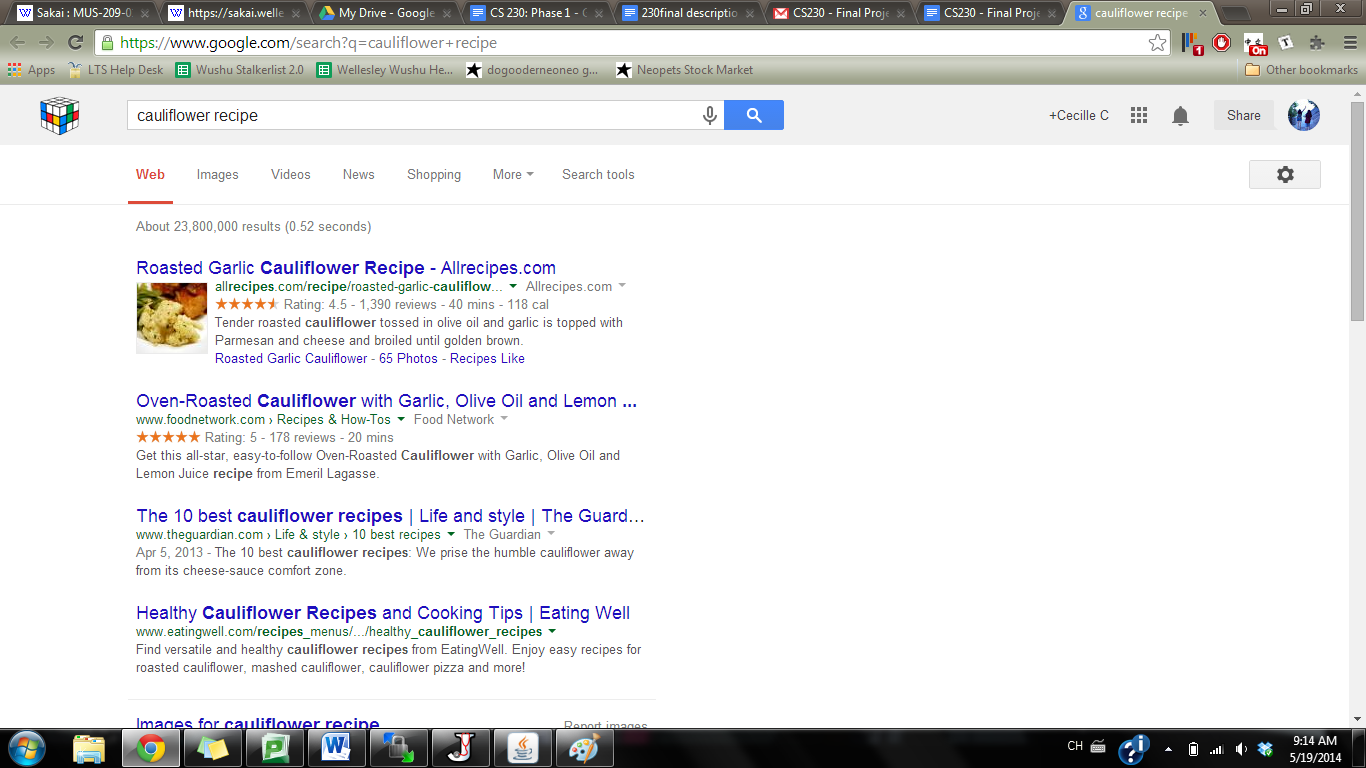


Users can add and remove guests from their parties and clicking on these buttons will present the user will the names of the guests that have been added/ or haven’t been added to their party according to the button pushed. However, if the user has already added all of their guests to the party, clicking on the “Add a Guest” button will prompt the user to add more friends to their contact list before they can add more to their party. Similarly, if there are no guests in a party and the user clicks on “Remove a Guest” the user will be prompted that no guests have been added to the party so no guests can be removed from the party.



The user can delete a party by clicking on the “Delete” button which will confirm that the user wants to delete the party. Finally, the user can search for a recipe for one of the recommended foods for a party, by selecting the party they wish to search for a food for from the combo box and then clicking “Search for a Recipe”. This button will then present a list of all the recommended foods for the party and clicking on one of the foods will open up a web browser with a recipe search for that food.





**Room for Improvement:**

* Allow user to modify contacts (change likes, allergies)

**Technical Report**

**Basic Classes**

**Person Class**

Represents an individual and their food preferences and allergies.

This class utilizes *linked lists* to keep track of preferred foods (one for each: main, vegetable, dessert) and a linked list to keep track of allergies, if there are any.

Methods:

* constructor to create a person w/ name
* constructor to create a person w/ name and linked list of preferred foods and allergies
* add preferred food (main, vegetable, dessert)
* add allergy
* a “likes” method to check if a person likes a specific food
* an isAllergic method to check if a person is allergic to a specific food
* a getter for name
* a getter for the linked list of main, vegetable, and dessert
* a getter for the allergies list
* toString

**Contacts Class**

Stores a collection of people classes that can be accessed quickly for information on a specific person’s food preferences/ allergies. Gives the option to load a previously constructed contacts list or save a contacts list you made.

This class utilizes a hash table with a key (person’s name) and value (person class itself) to quickly access a person’s information when given a name.

Methods:

* constructor to read from a formatted file of people
  + helper method to convert strings from file to linked lists
  + helper method to create a person based on linked lists
* constructor to create an empty contacts class
* a write to file method to “save” contacts for future loading purposes
  + helper method to convert linked lists to strings
* an addPerson method (returns true if successfully added)
* a removePerson method (returns true if successfully removed)
* a getter method for getting a specific person (can then be used to get allergies/ preferences w/ person class methods)
* a getter method for getting the hash table
* a getter method for getting total number of contacts
* a toString method

**Party Class**

The party class represents a party and all the people invited.

Methods:

* A constructor that only takes in the name of the party and makes an empty guests list
* A constructor that takes in a name and a linked list of guests
* Method to add a person
* Method to remove a person
* A getter for the guests list (returns LinkedList<String>)
* A getter for the party name
* Methods to return the most commonly liked dishes (most frequent, could be more than one)
  + main
  + vegetable
  + dessert
* A method to return a string of all guests and their allergies
* A method that returns all common foods (main, vegetable, and dessert), for the search function
* A method that checks if a person is a guest at the party
* Methods that return the number of people who like the recommended main dishes
  + main
  + vegetable
  + dessert

**MyParties Class**

The MyParties class represents a collection of parties that a user can load or save for future reference.

* A constructor that creates an empty set of parties
* A constructor that reads from a file and takes in a contacts class to create parties based on the contacts
* A constructor that takes in one party and creates an empty contacts list
* A constructor that takes in a contacts list and makes a MyParties class with no parties and that contacts class
* A method to check whether all guests read from file were added (all appeared in corresponding contacts class).
* Method to add a party to the parties list (checks for same name)
* Method to delete a party
* Method to retrieve a party
* Method to set contacts to a Contacts class passed in
* Getter for contacts
* Getter for a String array of all names in contacts book (for use in JComboBoxes)
* Method to write a MyParties class to a string
  + Method to then take the string and write it to file

**GUI Classes**

The GUI classes visually represent the program.

* Main GUI (constructs other panels and MyParties class and passes in appropriate parameters)
* About Panel
* Contacts Panel
* MyParties Panel