CptS - 451 Introduction to Database Systems
Spring 2021

Project Description

In your semester long CptS 451 course project you would develop a data search application for "Yelp.com's business review data". The emphasis would be on the database infrastructure of the application.

Learner Objectives:

At the conclusion of this assignment you will gain experience in:

- ✓ Database modeling and design
- ✓ Populating the database with large datasets
- ✓ Querying large databases
- ✓ Optimizing query performance through indexes
- ✓ JSON parsing
- ✓ Database Application Development

Overview:

In 2013, Yelp.com has announced the "Yelp Dataset Challenge" and invited students to use this data in an innovative way and break ground in research. In your project you would query this dataset to extract useful information for local businesses and individual users.
The Yelp data is available in JSON format. The original Yelp dataset includes and **1.32M** tips by **1.97M** users for **209.39K** businesses from United States, Canada, UK, and Germany. (https://www.yelp.com/dataset) In your project you will use a smaller dataset that your instructor created. This simplified dataset includes only **19,983** businesses, **189,298** users, **287,288** tips written for those businesses, and **3,786,310** check-ins to those businesses.

You will be given sample (Python) code to parse some of the Yelp JSON files (available on Canvas). The Yelp JSON files that you will use in this project are available at the instructor's website at:
https://eecs.wsu.edu/~arslanay/CptS451/project/yelp_dataset/yelp_CptS451_2020.zip
(Note: Please make sure to use the dataset available on the above link, not the one from the Yelp.com website)

See Appendix-B for an overview of the Yelp Academic Dataset.

Requirements:

You will develop a target application which runs queries on the Yelp data and extracts useful information. The primary users for this application will be potential customers seeking for businesses.
Using this application the users can gather information about:

- the businesses in a particular state, city, and/or zipcode,
- the businesses that belong to certain categories,
- detailed information about a business,
- ratings and popularity of businesses

You may design your application either as a standalone or a web-based application.

A detailed description of the application and example screenshots are available in **Appendix-A.** In evaluating your work instructor will primarily focus on how you design your database and how efficiently you can search

the database. However, your GUI should provide the basic functionality for easy search of the business. Creativity is encouraged! Additional functionality will be considered for extra credit.

Submission Instructions:

You will submit the deliverables for milestones on **Canvas (canvas.wsu.edu)**. For each milestone you will create a .zip files that contains all deliverables for that milestone, name the .zip files as *<yourteamname>_milestoneX.zip*, and submit it to the corresponding milestone dropbox on Canvas. Specific submission details for each milestone will be provided in milestone descriptions.

Below is a summary of the milestone tasks. We will post detailed descriptions of the milestones when they are assigned.

Project Milestones:

I. Milestone-0: (no submission required)

Download and install PostgreSQL Database Server. You may download the latest version from the link https://www.postgresql.org/ (Download and install the latest version of PostgreSQL Core Distribution.)

II. Milestone-1:

1) Parse JSON Data:

   Download the Yelp dataset from: https://eecs.wsu.edu/~arslanay/CptS451/project/yelp_dataset/yelp_CptS451_2020.zip. Look at each JSON file and understand what information the JSON objects provide. Pay attention to the data items in JSON objects that you will need for your application.

   Download the sample program from Canvas (*Project/ Sample JSON Parsing Code*). The sample code:
   - reads JSON objects form a file and extracts certain key and value pairs from JSON objects, and
   - writes the extracted data into a text file.

   Please note that the sample code includes examples of extracting simple key, value pairs from business JSON objects. In a JSON object the key value can be another JSON object (for example: categories and attributes in business data), therefore you need to recursively parse those objects until you extract all data stored in JSON objects. You will write the code for parsing business, user, tip, and check-in JSON objects.

2) i) Design a database schema that models the database for the described application scenario in Appendix-A and provide the ER diagram for your database design. Your schema should be precise but complete. It should be designed in such a way that all queries/data retrievals on/from the database run efficiently and effectively. In Milestone2 you will revise your ER model.

   ii) Translate your ER model into relations and produce DDL SQL statements for creating the corresponding tables in a relational DBMS. Note the constraints, including key constraints, referential integrity constraints, not NULL constraints, etc. needed for the relational schema to capture and enforce the semantics of your ER design.

2) Build a very simple database application (either web or standalone) which runs simple queries on the given simple database. The goal of this exercise is to get you started in database programming early on.

   The instructor will provide a video which explains how to establish connectivity with PostgreSQL in C# using Npgsql. Instructor will provide the queries you need to run on your table (see Milestone 1 specification).

Milestone-1 Deliverables:

1. (25%) Source code for parsing all JSON data. Only submit your source code, **not the data files**.
2. (40%) The E-R diagram and relations (CREATE TABLE statements) for your database design. To create your ER diagram, I suggest you use draw.io tool (https://www.draw.io/). You may also use your favorite drawing tool (e.g., Visio, Word, PowerPoint). Should be submitted in .pdf format. Name the diagram "*<your-team-name>_ER_v1.pdf*" and the SQL statements "*<your-team-name>_schema.sql*".
3. (35%) Source code for your application. Only submit your source code, **not the data files**.

III. Milestone-2:

1) Revise your database schema (ER model and relations).
2) Populate your database with the Yelp data. Generate INSERT statements for your tables and run those to insert data into your DB. You will also write and additional scripts to update the information stored in your database.

   Write triggers to ensure the validity and consistency of the information stored in your database. Details will be available in Milestone2 specification.
3) Build the alpha-prototype of your application.

Milestone-2 Deliverables:
(Weights of the deliverables are TBA)

1. The revised E-R diagram. **Should be submitted in .pdf format.** Name this file *"<your-team-name>_ER_v2.pdf"*
2. SQL script file containing all SQL statements (i.e., CEATE TABLE statements, UPDATE statements, and TRIGGERS) . Name this file *"<your-team-name>_SQL.sql"*
3. Source code for parsing/inserting Yelp data into the database.
4. Alpha version of your Yelp application.

Check the Milestone2 specification for additional deliverables.

You will demonstrate your Milestone2 to the instructor and the TA.

IV. Milestone-3:

In this milestone you will complete your Yelp application A detailed description of the application requirements is provided in Appendix-A.

Milestone-3 Deliverables:
The source code of your application. **Please only upload your source code, not your DB files.**

You will demonstrate your final project to the instructor and the TA. The demonstration schedule will be announced in mid-April.

References:

1. Yelp Dataset Challenge, https://www.yelp.com/dataset
2. Samples for users of the Yelp Academic Database, https://github.com/Yelp/dataset-examples
3. Yelp Challenge, University of Washington Student Paper 1 http://courses.cs.washington.edu/courses/cse544/13sp/final-projects/p08-fants.pdf
4. Yelp Challenge, University of Washington Student Paper 2, http://courses.cs.washington.edu/courses/cse544/13sp/final-projects/p10-michelmj.pdf

Appendix-A

Application Specification

The primary users for this application will be potential customers seeking for businesses. Using this application the users can gather information about:

- the businesses in a particular state, city, and/or zipcode,
- the businesses that belong to certain categories,
- detailed information about businesses,
- ratings and popularity of businesses,
- the businesses that their friends visited and reviewed, etc.

You may design your application either as a standalone or a web-based application. Below you will find screenshots to help you visualize the required functionality.

The application will have 2 main windows:

A. **User Information:**

Users can view their own information, the list of their own friends, the latest tips that each friend has provided, etc.

Use Case:

1. The user enters their name and chooses their own user id (among the users who has the same name). The system displays the following for the selected user:
   - user's profile information (including, their *name, average stars, the date he/she joined yelp, number of fans, average stars, count of votes, total tip count, total tip likes, and user's location (lat/long coordinates*)).
   - user's friends (name and star rating of each friend and the date he/she yelps since)
   - the latest tip that each of those friends posted. Note that this is different than the list of most recent reviews by friends. You need to return the latest tip that each friend has posted. (See Figure-1 for the properties you should include for each friend tip.)

   Note: The user's location (lat/long) , total tip count, and total tip likes are not part of the JSON data, but are calculated/collected and stored in the databases as the application is used. The latitude and longitude coordinates are entered by the user in the "User" tab. "Tip count" is the count of tips user provided for businesses and the "total tip like" is the sum of all like courts for user's tips.

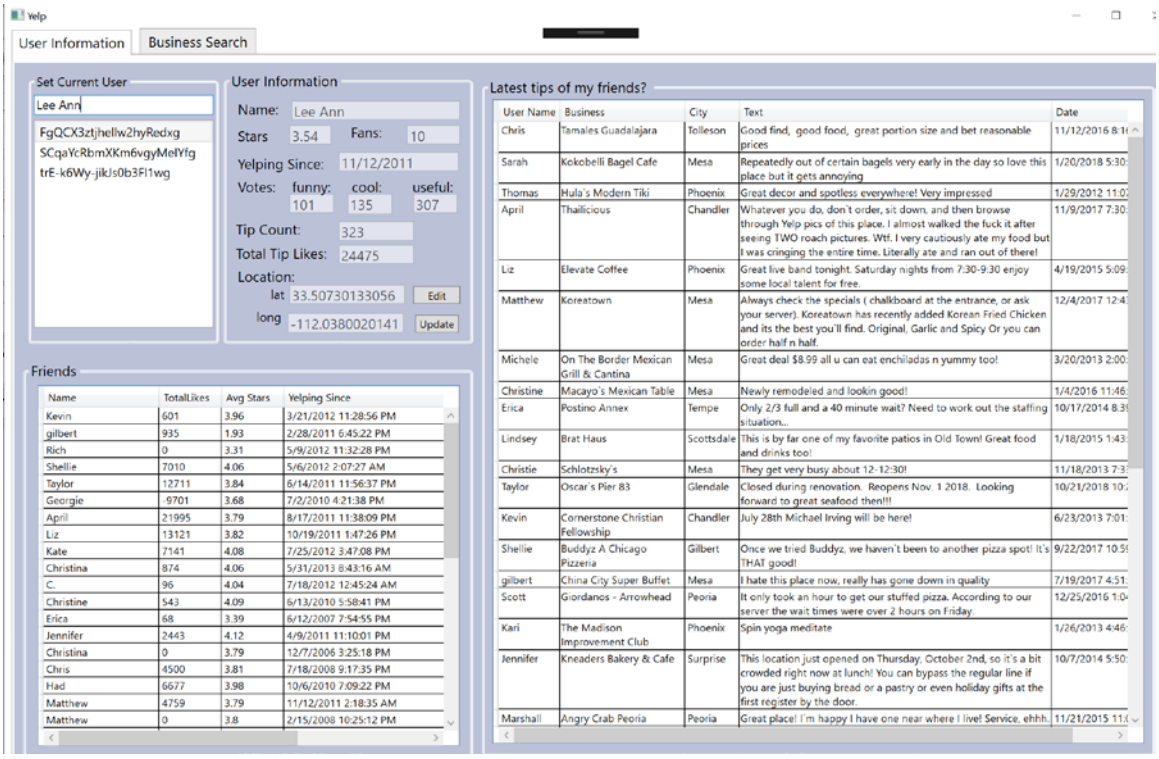


Figure 1 – User information window

B. **Business Search:**

Users can search for businesses which are within a certain state, city, and zip and which belong to the selected categories and/or attributes. The application allows users to retrieve and display various information about a selected business.

Use Cases:

1. User selects a state, city, and zipcode. When search button is pressed, the businesses in that state/city/zipcode are retrieved (see Figure-2). The following information is displayed for each business in the search result.
   - Business name
   - Address, city, state
   - Distance to user's location.
   - Business rating (stars)
   - Number of tips provided for the business
   - Number of check-ins to the business

   Notes:
   1. To get the number of tips and check-ins, you should
      (i) *query the tips table to calculate the number of tips;*
      (ii) *query the check-in table to calculate the number of check-ins for each business;* and
      (iii) *update those attribute values in the business table*.
   2. Distance: You should calculate the earth distance between the lat/long coordinated of the user and each business using the Haversine Formula. **The distance calculation should be implemented in SQL (as a function)**.

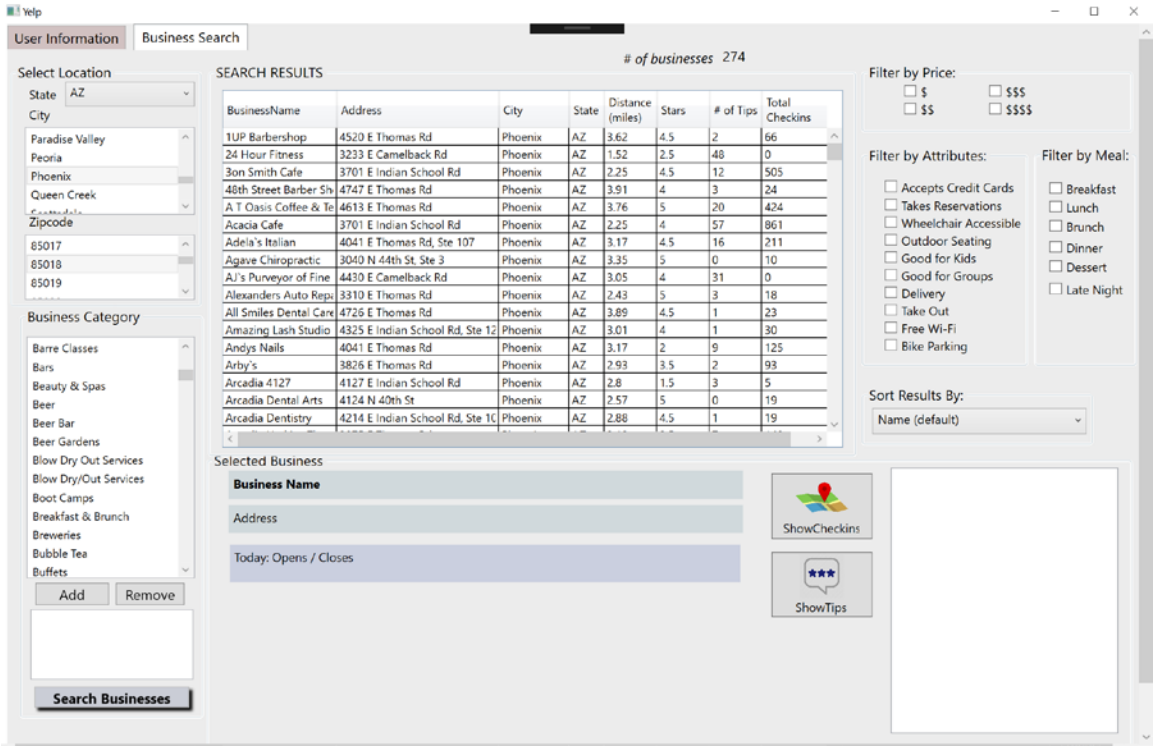


Figure 2 – Searching for the businesses in a Phoenix, AZ, 85018

3. The user might refine the results by specifying one or more business categories. The search will return the businesses which belong **to the ALL categories** specified by the user (i.e., AND condition) Note that, the more categories are selected the more restrictive the search will be. (see Figures-3 and -4)

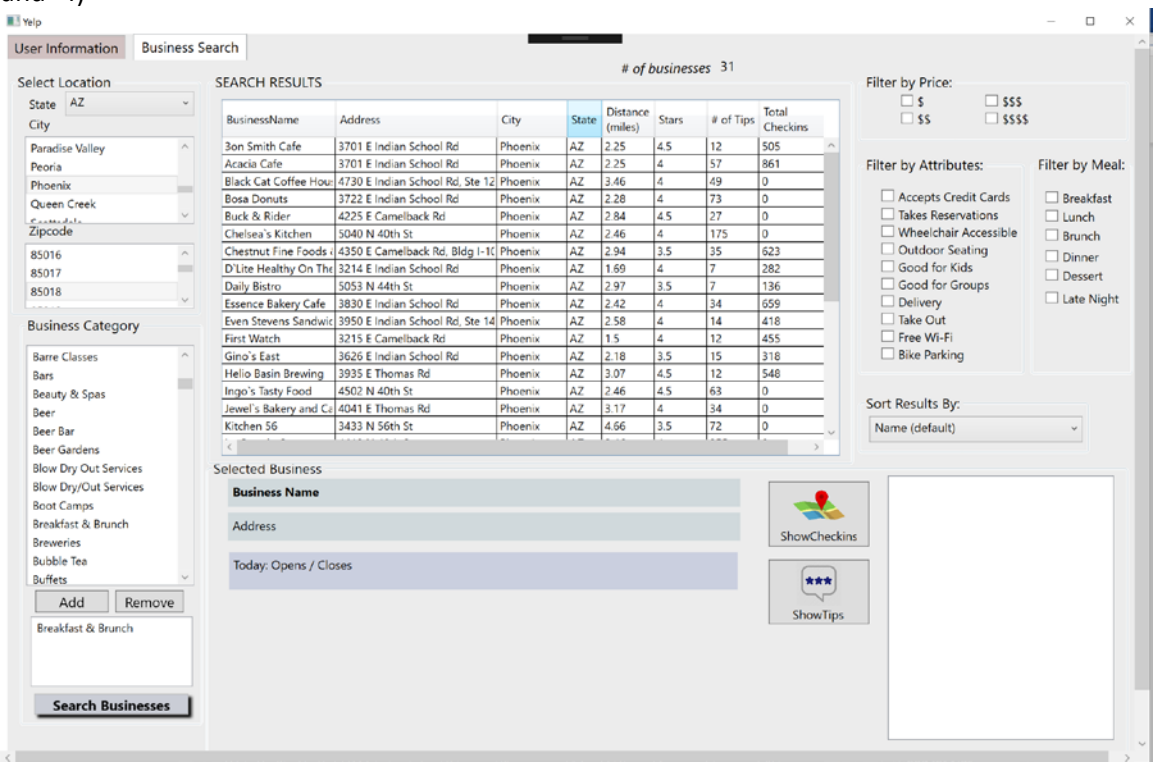
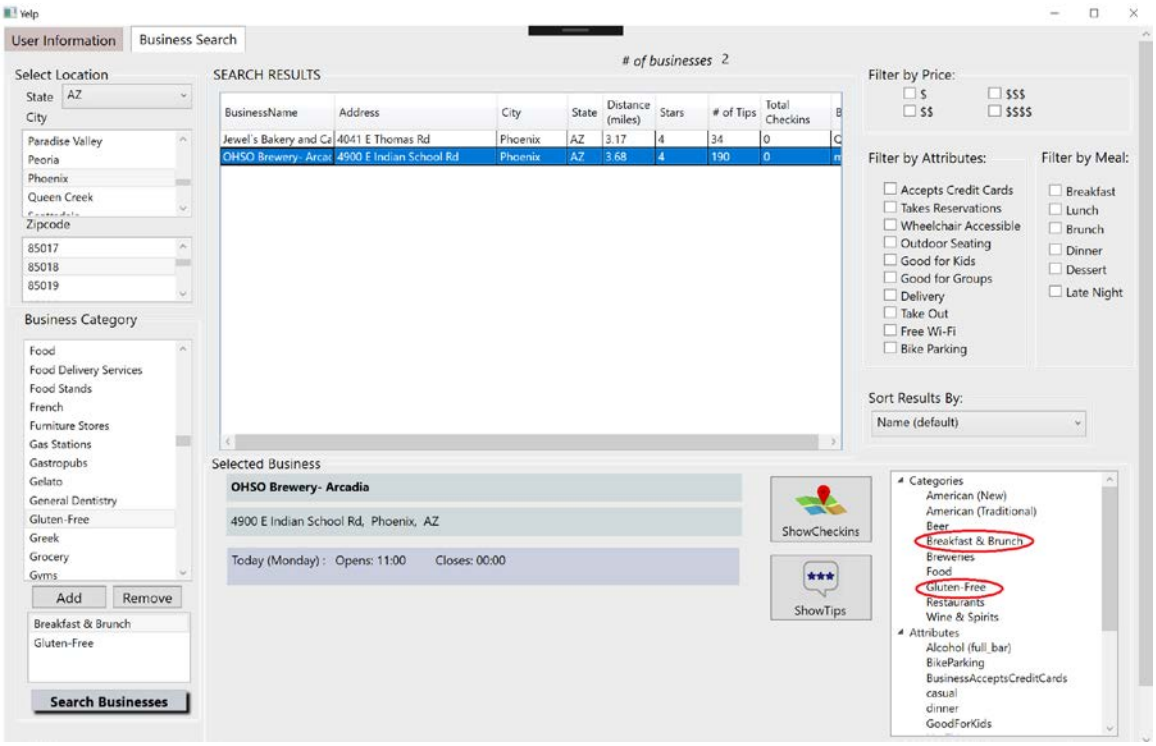
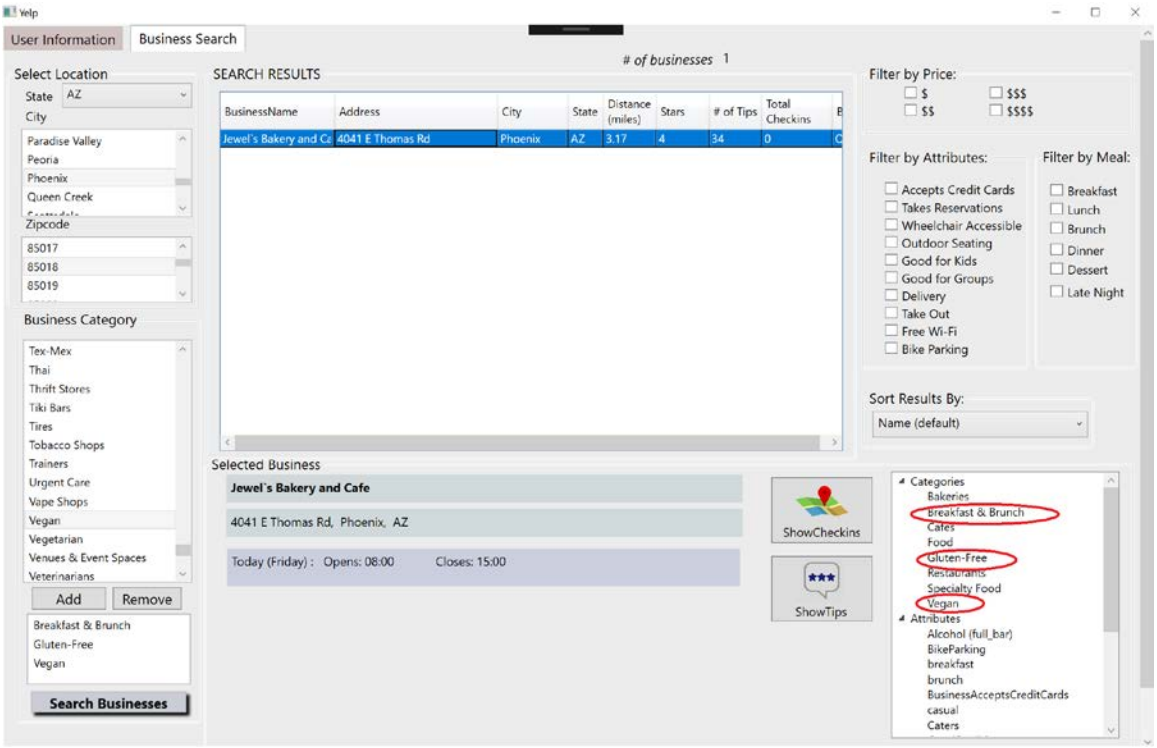


Figure 3 – Searching for the businesses with category 'Breakfast & Brunch' in Phoenix, AZ, 85018. The search result includes 31 businesses.



(a) Searching for the businesses with categories 'Breakfast & Brunch' and 'Gluten-Free' in Phoenix, AZ, 85018. The search result includes 2 businesses.



(b) Searching for the businesses with categories 'Breakfast & Brunch', 'Gluten-Free', and 'Vegan' in Phoenix, AZ, 85018. The search result includes a single business.

Figure 4 – Searching for the businesses that match multiple categories.

4. When the user selects a business in the search results, the following are displayed for the selected business:
    i. the name, address of the business
    ii. the open/close times of the business for the current day of the week;
    iii. the categories and attributes of the business

   (see Figures 4, 5, and 6 for examples)

5. The user may also refine results by specifying various attributes including:
    i. Price range (1 to 4) – (see RestaurantsPriceRange2 attribute)
    ii. Accepts credit cards (see BusinessAcceptsCreditCards attribute)
    iii. Takes reservations (see RestaurantsReservations attribute)
    iv. Wheelchair accessible (see WheelchairAccessible attribute)
    v. Outdoor seating (see OutdoorSeating attribute)
    vi. Good for kids (see GoodForKids attribute)
    vii. Good for groups (see RestaurantsGoodForGroups attribute)
    viii. Delivery (see RestaurantsDelivery attribute)
    ix. Take out (see RestaurantsTakeOut attribute)
    x. Wifi (free Wifi only) – (see WiFi attribute)
    xi. Bike parking (see BikeParking attribute)
    xii. Meals (breakfast, brunch, lunch, dinner, desert, and latenight attributes)

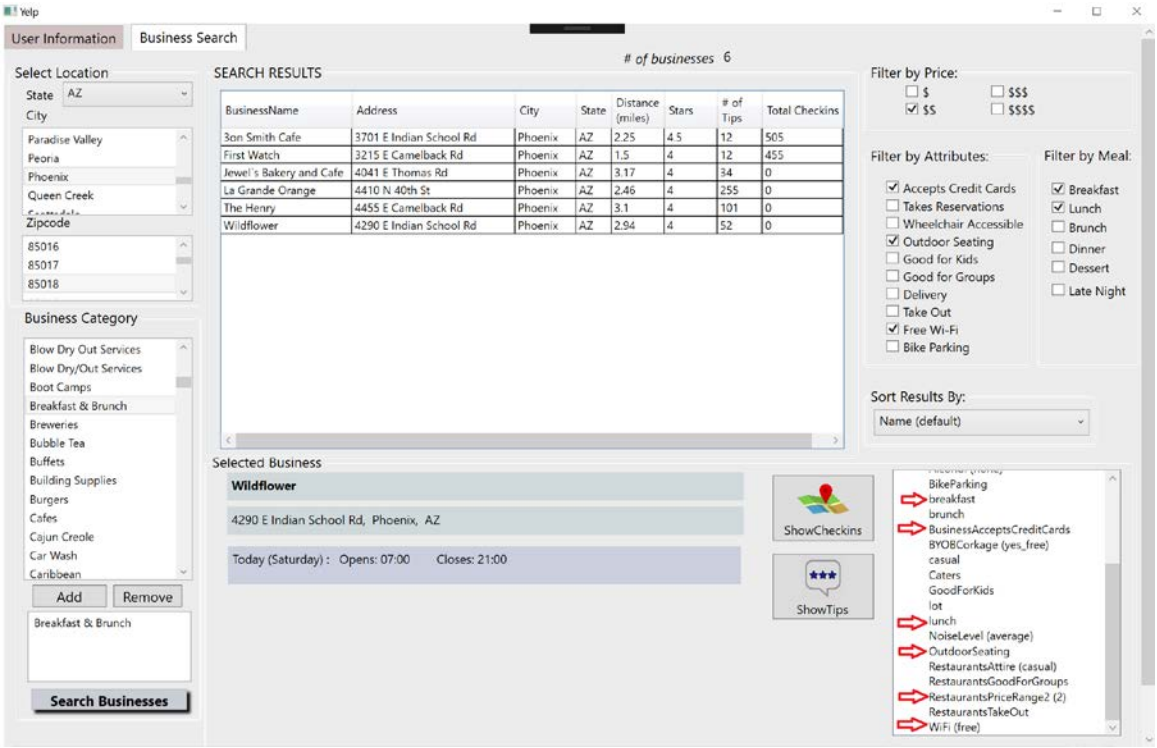


Figure 5 – Searching for the businesses with category 'Breakfast & Brunch' in Phoenix, AZ, 85018 which has 'price range 2', accepts credit cards, has outdoor seating, has free Wi-Fi, and serves breakfast and lunch.
The details of the selected business (*Wildflower*) are displayed.

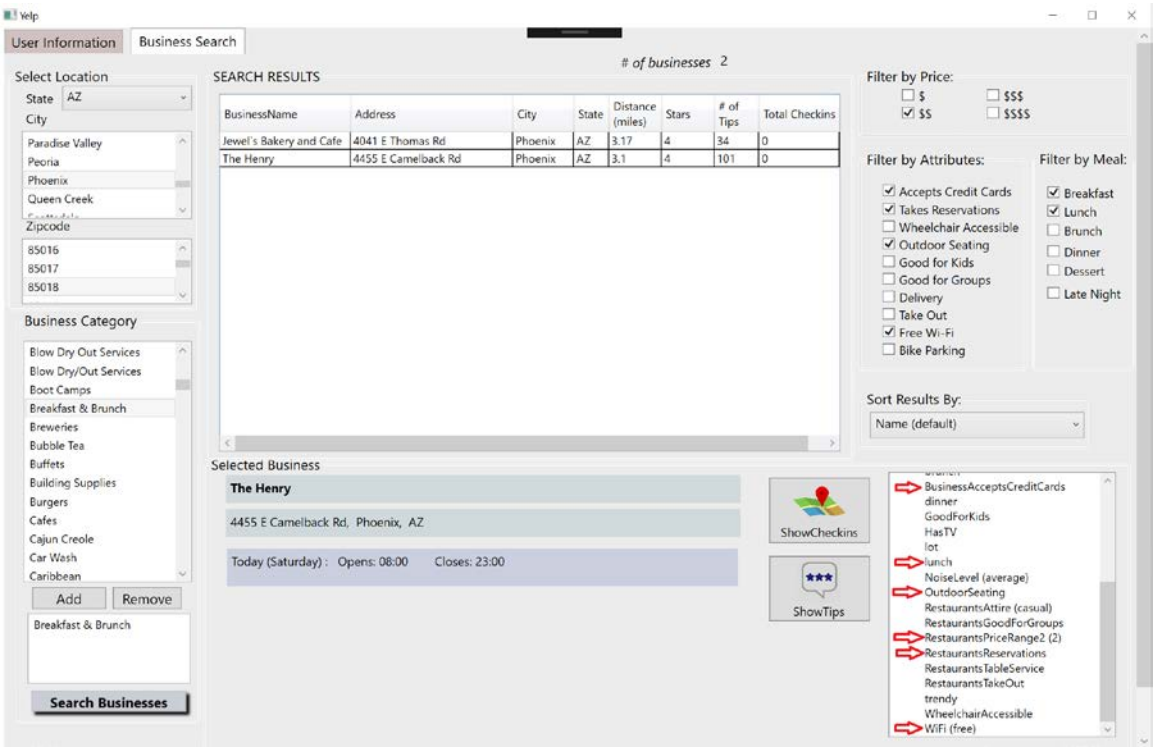


Figure 6 – Searching for the businesses with category 'Breakfast & Brunch' in Phoenix, AZ, 85018 which has 'price range 2', accepts credit cards, takes reservations, has outdoor seating, has free Wi-Fi, and serves breakfast and lunch. Compare this to Figure-5: adding an additional search attribute (takes reservations) narrows down the results.

6. The user may sort the results based on the following attribute values.
    a. Business name (default sort order)
    b. Highest rating (stars)
    c. Most number of tips
    d. Most check-ins
    e. Nearest

In Figure-7 the business search results are sorted by number of check-ins (in descending order). All sorting should be done in the SQL query (you can't use the sorting features of the data-grid.)

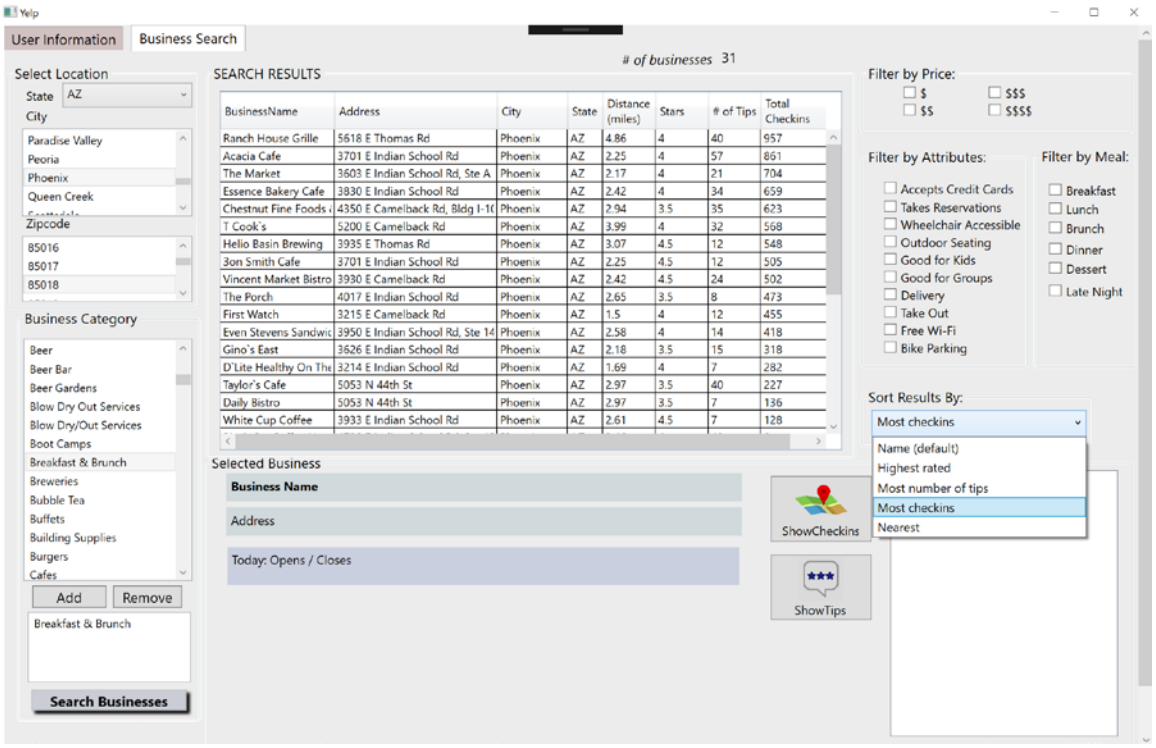
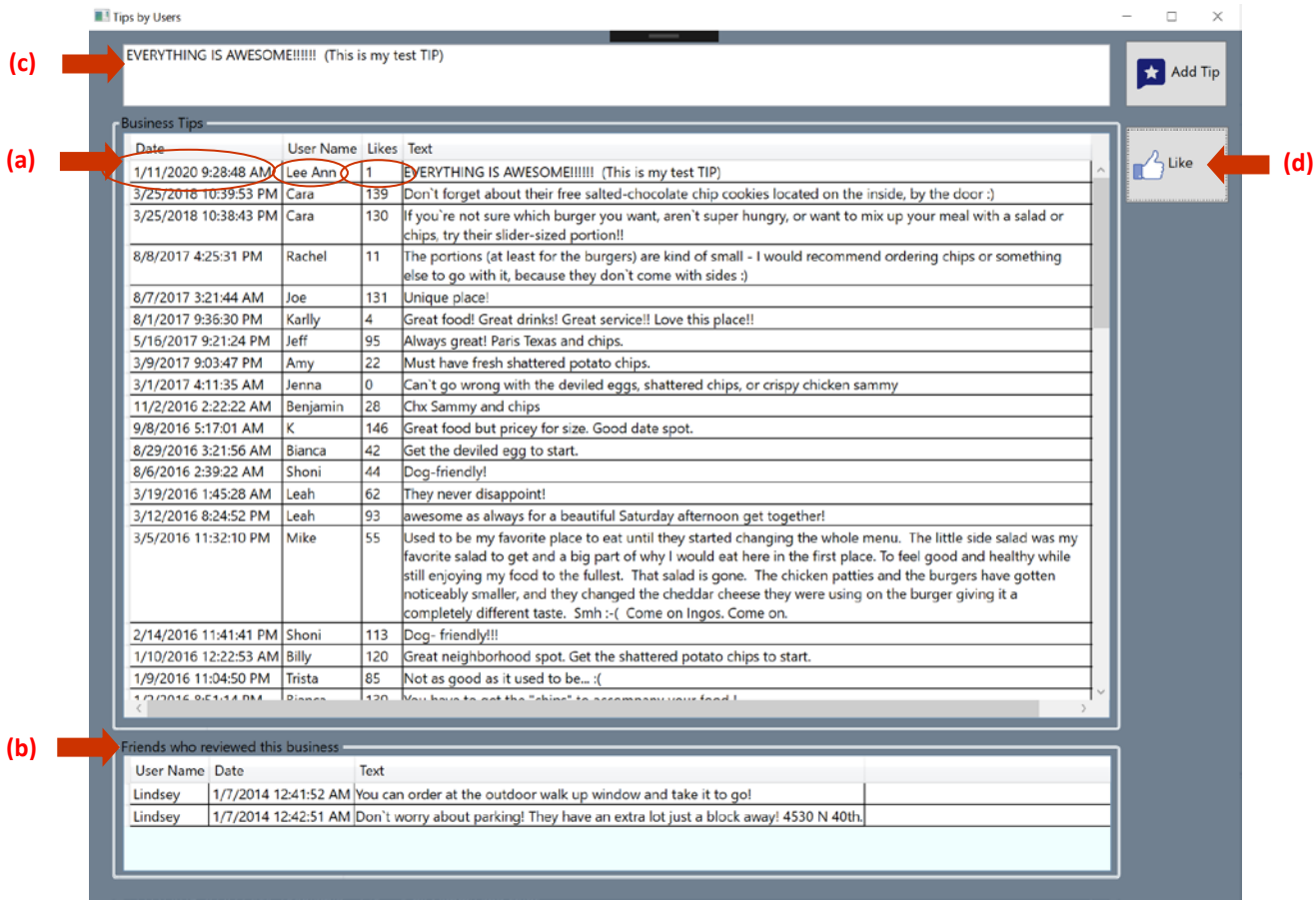


Figure 7 – Business search results can be sorted by various values. In this figure, results are sorted by the number of check-ins (in descending order).

7. The user may select a certain business in the search results (by simply clicking on a business) and
   (a) display the tips provided for the business;
   (b) display the tips that users friends have provided;
   (c) add a new tip for the selected business. (Note: We assume that the tip is posted by the user selected in the 'User Information' tab. The timestamp of the new tip should be the time that the tip is added.) (See Figure-8)
   (d) like a tip (in Figure-8 the user liked their own tip and the like count was incremented to 1)



Date	User Name	Likes	Text
1/11/2020 9:28:48 AM	Lee Ann	1	EVERYTHING IS AWESOME!!!!!! (This is my test TIP)
3/25/2018 10:39:53 PM	Cara	139	Don`t forget about their free salted-chocolate chip cookies located on the inside, by the door :)
3/25/2018 10:38:43 PM	Cara	130	If you`re not sure which burger you want, aren`t super hungry, or want to mix up your meal with a salad or chips, try their slider-sized portion!!
8/8/2017 4:25:31 PM	Rachel	11	The portions (at least for the burgers) are kind of small - I would recommend ordering chips or something else to go with it, because they don`t come with sides :)
8/7/2017 3:21:44 AM	Joe	131	Unique place!
8/1/2017 9:36:30 PM	Karlly	4	Great food! Great drinks! Great service!! Love this place!!
5/16/2017 9:21:24 PM	Jeff	95	Always great! Paris Texas and chips.
3/9/2017 9:03:47 PM	Amy	22	Must have fresh shattered potato chips.
3/1/2017 4:11:35 AM	Jenna	0	Can`t go wrong with the deviled eggs, shattered chips, or crispy chicken sammy
11/2/2016 2:22:22 AM	Benjamin	28	Chx Sammy and chips
9/8/2016 5:17:01 AM	K	146	Great food but pricey for size. Good date spot.
8/29/2016 3:21:56 AM	Bianca	42	Get the deviled egg to start.
8/6/2016 2:39:22 AM	Shoni	44	Dog-friendly!
3/19/2016 1:45:28 AM	Leah	62	They never disappoint!
3/12/2016 8:24:52 PM	Leah	93	awesome as always for a beautiful Saturday afternoon get together!
3/5/2016 11:32:10 PM	Mike	55	Used to be my favorite place to eat until they started changing the whole menu. The little side salad was my favorite salad to get and a big part of why I would eat here in the first place. To feel good and healthy while still enjoying my food to the fullest. That salad is gone. The chicken patties and the burgers have gotten noticeably smaller, and they changed the cheddar cheese they were using on the burger giving it a completely different taste. Smh :-(Come on Ingos. Come on.
2/14/2016 11:41:41 PM	Shoni	113	Dog- friendly!!!
1/10/2016 12:22:53 AM	Billy	120	Great neighborhood spot. Get the shattered potato chips to start.
1/9/2016 11:04:50 PM	Trista	85	Not as good as it used to be... :(

Figure 8 – The tips for the business 'Ingo`s Tasty Foods' in Phoenix, AZ, 85018. The most recent tips appear at the top. For each tip, you should display the name of the user who provided the tip, the date tip is provided, the likes for the tip, and the tip text.

8. The user may select a certain business in the search results (by simply clicking on a business) and
   (a) display the total number of check-ins for each month as a chart;
   (b) check-in to the selected business.

Please see Figure-10 for an example

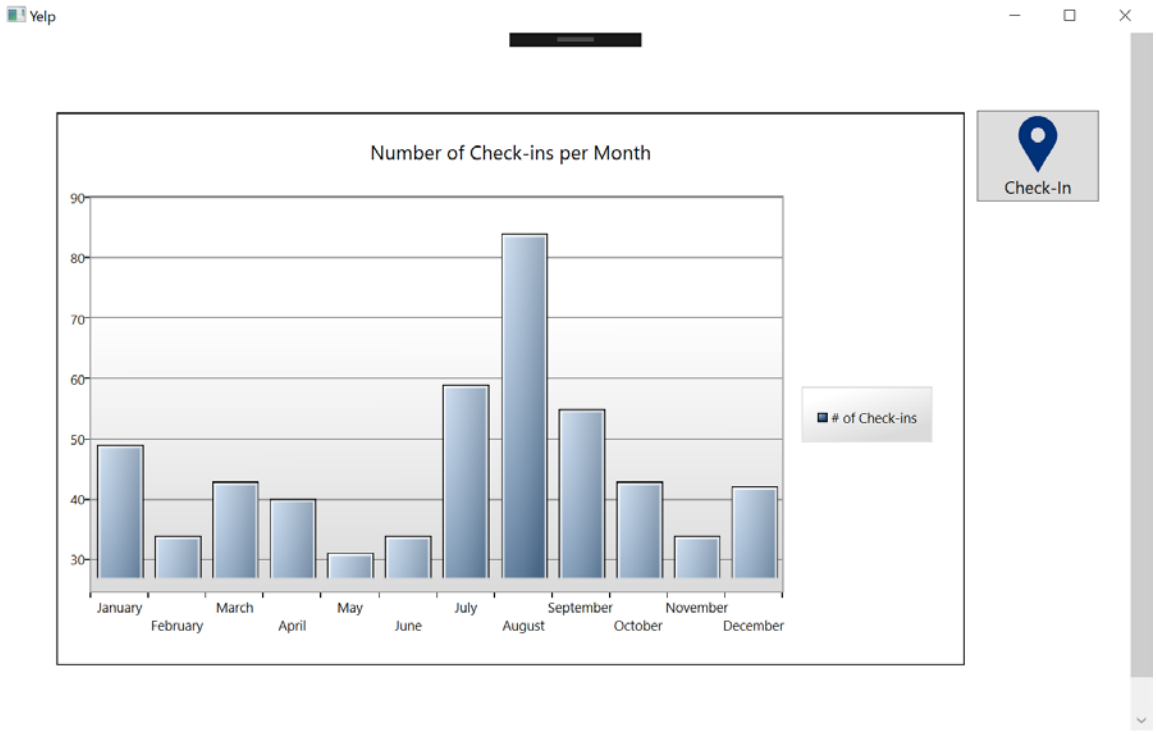


Figure 8 – The check-in chart for 'Helio Basin Brewing' in Phoenix, AZ, 85018. The graph shows the number of check-ins per month. When the user checks-in, the graph will be updated (the count for current month will be increased by one).

9. After a check-in to a business, the check-in count of the business should be updated (by a trigger). Similarly, when a new tip is provided for a business, the total tip count of the business should be incremented. (See Figue-9)
   You will write the triggers that increment the check-in and tip count in milestone-2.

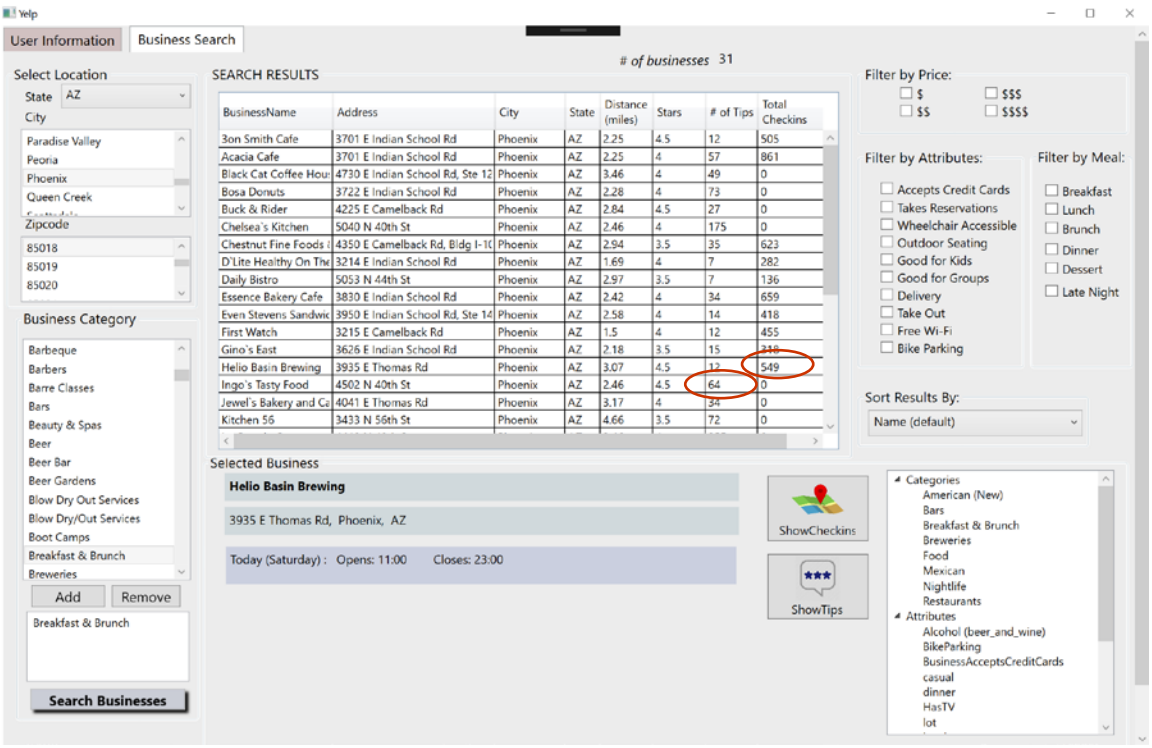


Figure 8 – After check-in, the check-in and tip counts are automatically updated. Compare the circled counts to those in Figure-3. (Check-in count increased from 63 to 64, and the tip count increased from 548 to 549).

Please note that all data displayed on the GUI should be kept in the database and should be retrieved from it when needed. You are not allowed to create internal data structures to store data.

You may design your application either as a standalone or a web-based application.

Appendix-B

Yelp's Academic Dataset

Yelp has made available a dataset which contains user reviews for **192.6K** businesses from United States, Canada, UK, and Germany. The purpose was to provide a real-world data set to promote research in various areas of research. The dataset includes 6 types of data objects: *business*, *review*, *user*, *tip*, *check-in, and photos*. Every object contains a 'type' field, which tells whether it is a *business*, a *user*, or a *review*. *Business* objects contain basic information about local businesses. *Review* objects contain the details of the reviews by users for the businesses. *Review*'s user_id associates the reviews with the *user* objects. Similarly, *review*'s business_id associates each review with the *businesses*.

Detailed description of the data objects is available at:
https://www.yelp.com/dataset/documentation/main

In your project, you will only parse *business*, *user*, *tip,* and *check-in objects.*

Usage of this dataset is governed by the Academic Dataset Terms of Use.