

CptS -451 Introduction to Database Systems Spring 2017

Project Milestone-3

Due Date: Thursday April 20th, by 11:59pm
Milestone-3 Demos: April 24th and April 25th

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:

Use Case:

1. The user enters his/her own user id and retrieves his/her user profile information including, name, average stars, date he/she joined yelp, number of fans, average stars, and count of votes. The list of the user's friends and the latest tip each friend posted are displayed. User may rate one of his/her friends or remove a friend. (See Figure-1)

The screenshot shows a web application interface for Yelp. It has a top navigation bar with 'User Information' and 'Business Search' tabs. The main content area is divided into three sections:

- Set Current User:** A form with 'User ID:' and a text input field containing 'KtNT8biTJlbAFh-APgO7-w'. A 'Set User' button is below it.
- User Information:** A summary of the user's profile:

Name:	Debbie	Yelping Since:	2006-06
Stars:	4	Votes:	funny: 557 cool: 557 useful: 557
Fans:	35		
- Friends:** A table listing the user's friends with columns for Name, Avg Stars, and Yelping Since.

Name	Avg Stars	Yelping Since
Dan	3	2007-07
Corine	4	2012-12
Sophia	4	2014-06
Julie	4	2010-05
Nate	4	2011-11
Sheila	4	2007-07
Michelle	4	2012-12
Chad	4	2011-06
Marc	4	2013-07
Manuel	3	2013-03
Rob	4	2008-01
Avner	4	2011-02
Tiana	3	2007-07
Reggie	4	2006-09
James	4	2011-05
Alexis	4	2013-12
Mark	4	2013-06
Christie	4	2012-06
Terri	4	2009-10
Marc	4	2008-08
Clifton	4	2011-08
Levinia	4	2012-10
Zurii	4	2006-03

 Below the table are buttons for 'Rate Friend' and 'Remove Friend', along with a 'rate score (0-5)' input field.
- Tips by Friends:** A table showing the latest tips posted by the user's friends.

User Name	Business	City	Text
Jaime	Cili	Las Vegas	Super quiet on the weekends. You may even get the patio to yourself ^
Jennifer	The Golden Door Spa & Cafe	Cave Creek	My husband loved loved loved the ashiatsu massage.
Jennifer	Carefree Station	Carefree	Order the flaming saganaki. You don't need a description. Just do it.
David	Vovomeena	Phoenix	Coffee beans for their japanese cold brew come from Cave Creek Ro
Miyuki	Kirkland's	Las Vegas	Great deals but awful customer service.
Miyuki	Bonito Michoacan	Las Vegas	House margarita free with check in.
Alexis	All About Catering	Las Vegas	Don't listen to Nacho B. They were clients of a competitor!
Jason	Farmer Boys	Las Vegas	Breakfasting.
Jason	Grouchy John's Coffee Shop	Las Vegas	First timer
Wynn	Raku	Las Vegas	Delicious
Britney	Hakkasan Nightclub	Las Vegas	Sucks ass
Steve	Seven Tattoo Studio	Las Vegas	Opening party tomorrow, lots of food trucks, music, beverages, and I
Queenie	Pho Bosa	Las Vegas	Try Bun bo hue!
April	Farmer Boys	Las Vegas	No blackberry jam. :-(((
Lisa	Desigual	Las Vegas	Looks closed- gone and off map too :(
Kaui	Kahuku Poke and Hawaiian Barbecue	Las Vegas	Good good, great service.
Julie	Angara Indian Spice Grill	Las Vegas	The samosa here is very delicious! I usually avoid them at other India
Tiana	CC Nail Spa	Las Vegas	Kim and Mia are the best!!! Best nails in Vegas!!!
Tiana	J T Nails	Las Vegas	\$30 French tip pedicure
Tiana	Modern Nails - Las Vegas	Las Vegas	Closed? Called at 11am Saturday no answer
Tiana	Nails Design	Las Vegas	\$20 men's pedi \$25 French tip pedicure
Russ	Yama Sushi	Las Vegas	"No Name" - should be named "Something Awesome"!
Russ	Du-Par's Restaurant and Bakery	Las Vegas	Best (Standard) Pancakes I've had! Just Plain Buttery, Fluffy and that E
Eric	Little Shop of Magic	Las Vegas	Love this place
Rodney	Spago	Las Vegas	Spago is celebrating their 22nd anniversary with their original menu

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. The application allows users to display some statistics about the businesses in the search results and to retrieve various information about a selected business (See Figure-2)

BusinessName	Address	#ofTips	TotalCheckins
Zoë's Kitchen	521 W McDowell Rd Phoenix, AZ 85003	43	541
My Florist Cafe	530 W McDowell Rd Phoenix, AZ 85003	4	20
Sticklers	30 N 3rd Ave Phoenix, AZ 85003	17	270
Court House Cafe	111 W Monroe St Ste 300 Phoenix, AZ 85003	1	11
Cibo	603 N 5th Ave Phoenix, AZ 85003	177	2778
The Coffee Conspiracy	Conspire 901 N 5th St Phoenix, AZ 85003	3	19
The Hero Factory	114 W Adams St 102 Phoenix, AZ 85003	18	216
Jimmy John's	11 W Washington St Ste 175 Phoenix, AZ 85003	10	213
Jersey Mike's Subs	1515 N. 7th Avenue Ste 140 Phoenix, AZ 85003	16	275
The Habit Burger Grill	530 W McDowell Rd Phoenix, AZ 85003	40	560
Potbelly Sandwich Shop	534 W McDowell Rd Phoenix, AZ 85003	18	351

Figure 2 - Search Businesses**Use Cases:**

1. User selects a state, city, and/or zipcode and the business categories for the business in that state/city/zipcode are displayed (you may be deducted points if you just list all categories). User specifies one or more categories from that list; when search button is pressed the businesses in that state/city/zipcode which belong **to ALL specified categories** will be returned (i.e., AND condition). The following information should be provided for each business:
 - Business name
 - Address
 - # of tips provided for the business
 - Total number of check-ins
 (Note: You should query the tips table to calculate the number of tips and number of check-ins for each business and update those attributes in the business table.)
2. The user might refine the results according to the times the business is open/closed on a certain day of the week. User specifies a day-of-week and a start and end time. All businesses that are closed during the given time-slot are excluded from the results. Please note that if a business is closed during part of the slot (but open during the rest), it should be excluded from the result. All filtering on the results need to be implemented in SQL queries. No points will be given if you filter results in the GUI when you display them in the list-view.

If no time-slot is given, no filtering should be done.

3. The user may select a certain business in the search results (by simply clicking on a business) and display various information about the business, including:
 - a. *Show Check-ins*: All check-ins for the business which are grouped by the day-of-the-week and the time-of-the-day. For simplicity, you are asked to aggregate the check-in information into morning (6am-12noon), afternoon (12noon-5pm), evening (5pm-11pm), night (11pm-6am) intervals. (Assume start time of each interval is inclusive and end time is exclusive.) Your application should visualize the number of check-ins for each day-of-the-week as a chart. Please see Figure-3 for an example.

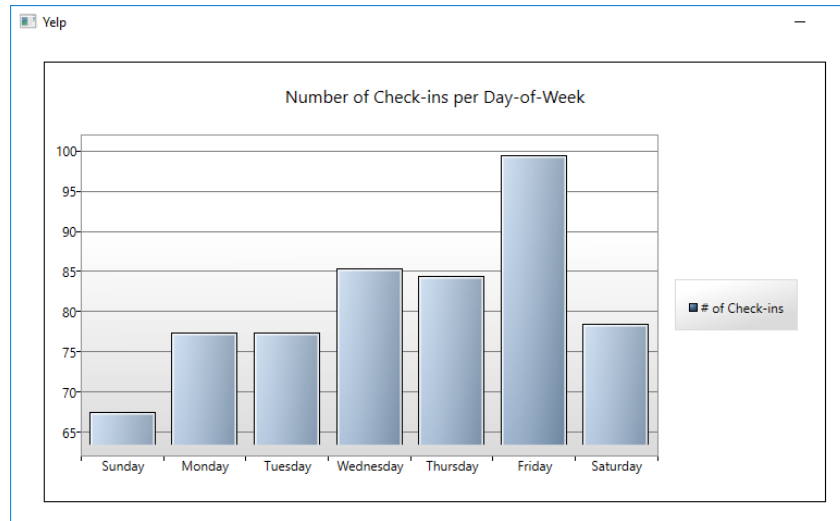


Figure 3 – Number of check-ins for the selected business

- b. *Show Tips*: The tips provided for the selected business. For each tip, you should display the name of the user who provided the tip, the date tip is provided, the number likes for the tip and the tip text. You should display this information as a list (or table).
- c. *#of Business per Category*: Number of business per category for the businesses that appear in the search results. You should display this information as a chart. (see Figure-4)

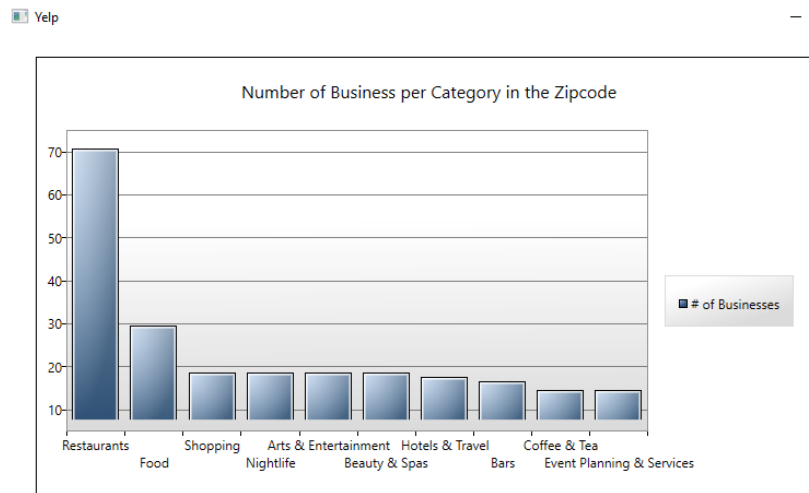


Figure 4 – Number of businesses per category

- d. *Avg Stars per Category*: Average number of star ratings per category. Your application should display this information as a graph.

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.

Milestone-3 Details:

- Establish connectivity with the DBMS
- Write the SQL queries to search your database tables for the use cases described above.
- Embed/execute queries in/from the code. Retrieve query results and parse the returned results to generate the output that will be displayed on the GUI.
- Implement the GUIs for the described use cases.

In evaluating your work instructor's primary focus will be primarily on how efficiently you can search the database and pull out the information. However your GUI should provide the basic functionality for easy browsing of the business categories and attributes (as illustrated in the images). Creativity is encouraged! Additional functionality will be considered for extra credit.

The response time for searches should be reasonably fast (in the order of seconds). You will need to create indexes to speed up the searches on big tables.

Suggestions:

- Please start this milestone a.s.a.p. (if you haven't done so). Please reach out to the instructor if you have any issues.
- PostgreSQL creates indexes for primary key and foreign key attributes. You might need to create additional indexes to improve your query response times. (PostgreSQL syntax for creating indexes: <https://www.postgresql.org/docs/9.1/static/sql-createindex.html>)
Try running your queries before you add any new index. If the response time for a particular query is longer than 30-40secs, then create index(s) on join attribute(s) and/or search attributes of that query.

Testing your Queries:

You may test the correctness of the queries in your application as follows:

- Consider the input values for the query and choose the value(s) which will return a small number of results.
- Open the raw JSON file in an editor, and search for the objects that have the query search value(s).
- If the result you estimated from the JSON file matches the result of the SQL query, then your query should be correct.
- Otherwise you should first make sure your database is populated correctly for the values you are testing for. If your database is correct but there is a problem with the query itself, then you should break down your query into multiple steps and verify that the intermediate results between those steps are correct.

Examples:

Query: "Given a zipcode and one or more categories, find businesses which are located in that zipcode and which are associated with all the specified categories."

- To test this, decide on zipcode and some category name(s) that will give you a small list of results. First try the query with a single category selection. Compare the number of businesses you count in the JSON file to the number of businesses your query returns. Then try for two or more categories.
- Pick a few of the businesses your application lists in the results and search for them in the JSON file. Make sure that they satisfy all the search and filter criteria, including categories, zipcode, open/close times (if specified).

Query: “Given a user find the friends of the user and list the latest tips provided by each friend. “

- To test this, pick a user who has a reasonable number of friends (up to 10). For each of the friends listed in the user JSON object, check the tips JSON objects and look for the tip(s) by that friend that has the most recent date (substitute friend id for userid when you search tips) Repeat it for each of the friends. Note that if the user provided more than one tip on the same day and if those are the latest tips by the user, all of those should be in the result.

Milestone-3 Deliverables:

1. The source code of your application. **Please only upload your source code, not your DB files.**
2. SQL script file that contains your main SELECT query for searching businesses and reviews. Also include the CREATE INDEX statements for the indexes you used.

Create a zip archive “<your-last-name>_milestone3.zip” that includes your source code and the SQL script file. Upload your milestone-3 submission on Blackboard until the deadline.

You will demo your final code to the instructor on April 24th or April 25th. The demo schedule will be posted before the deadline.

References:

1. Yelp Dataset Challenge, http://www.yelp.com/dataset_challenge/
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>