Team 1

12/6/2016

CIS444, Web Programming

Tukwut chow

A social restaurant search engine

Contents

[Introduction 3](#_Toc468726639)

[Team Members 3](#_Toc468726640)

[Project Description 3](#_Toc468726641)

[Potential users 3](#_Toc468726642)

[Our advantage over competitors 3](#_Toc468726643)

[User’s Manual 4](#_Toc468726644)

[Anonymous Users 4](#_Toc468726645)

[Listing Restaurants 4](#_Toc468726646)

[Searching Restaurants 4](#_Toc468726647)

[Viewing a Restaurant’s page 4](#_Toc468726648)

[Signing Up for an Account 6](#_Toc468726649)

[Registered Users 6](#_Toc468726650)

[Logging In 6](#_Toc468726651)

[Submitting New Restaurants 7](#_Toc468726652)

[Voting 7](#_Toc468726653)

[Adding a Comment 8](#_Toc468726654)

[Admin Users 8](#_Toc468726655)

[Listing and Promoting Users 8](#_Toc468726656)

[Enhanced moderation powers 8](#_Toc468726657)

[Web Pages and Their Interrelationships 9](#_Toc468726658)

[Index.php 9](#_Toc468726659)

[addRestaurant.php 9](#_Toc468726660)

[restaurant.php 9](#_Toc468726661)

[login.php, logout.php, login.php, loginVerify.php, signUpVerify.php 9](#_Toc468726662)

[loginFailure.php 9](#_Toc468726663)

[listUsers.php 9](#_Toc468726664)

[Misc. Form Handler Pages 9](#_Toc468726665)

[Database Design and Implementation 10](#_Toc468726666)

[Table Descriptions 10](#_Toc468726667)

[User 10](#_Toc468726668)

[Restaurant 10](#_Toc468726669)

[Vote 10](#_Toc468726670)

[Comment 10](#_Toc468726671)

[Tag 10](#_Toc468726672)

[Tag Restaurant Mapping 10](#_Toc468726673)

[ERD 11](#_Toc468726674)

[Web Design Issues such as resolved difficulties, possible future improvements and enhancements 12](#_Toc468726675)

[Implementation Details 12](#_Toc468726676)

[User Contributions 12](#_Toc468726677)

[Rajak Jamal 12](#_Toc468726678)

[Curtis Robinson 12](#_Toc468726679)

[Shane Sharareh 13](#_Toc468726680)

[Richard Stelly 13](#_Toc468726681)

[Justin Van Bibber 13](#_Toc468726682)

[Travis Weil 13](#_Toc468726683)

# Introduction

## Team Members

* Rajak Jamal
* Curtis Robinson
* Shane Sharareh
* Richard Stelly
* Justin Van Bibber
* Travis Weil

## Project Description

Our team is unique in that every one of our members, at some point, has been a commuter student. Each of us knows the difficulty of finding a good place to eat around campus when one is new to the area. San Marcos provides so many great restaurant options, it’s easy to get lost in all the choices. We were inspired by this struggle to create a means for Cal State San Marcos students to list their favorite restaurants, vote on other’s suggestions, and discover new and interesting finds they may have never heard of before. Our initial vision resembled something like Yelp – user submitted pages for each and every restaurant with a 5-star rating system. As we started to sketch up plans for the site, the complexity grew. A fundamental question appeared dealing with the rating system – Why 5 stars? What differentiates a two-star rating from a three-star? Or what’s the difference between a three and a four? Obviously a one-star rating is vastly different than a five-star rating, but what about the in-between?

Our answer to this is a simple upvote/downvote scoring method with a ranking by aggregate votes. You either like the restaurant and upvote it, are indifferent and ignore it, or hate it and downvote it. This is very similar to the way that Reddit’s scoring works, and we took that as inspiration for our scoring algorithm.

All content is user-submitted. Restaurants are submitted and voted upon by the community, and users are free to comment on a restaurant, maybe to provide a recommendation or a warning to potential visitors. Anonymous users are free to browse the site, search for restaurants, and read comments, but submitting a restaurant or voting requires an account. Finally, administrative users play the role of moderators, removing potentially abusive comments or restaurant submissions with their delete powers.

### Potential users

Anyone from CSUSM is welcome to use this site, from faculty to staff to students. Even parents, alumni, or friends of students, we envision, may want to use this site.

### Our advantage over competitors

We can provide anonymity for our users so they can speak their minds without fear of persecution. The site is also really simple to use and we think it looks very pretty as well.

# User’s Manual

## Anonymous Users

### Listing Restaurants

The first thing a visitor is greeted by is the home page, which provides a listing of every restaurant in the system, sorted by total score of upvotes or downvotes. Clicking the name of the restaurant takes the user to the restaurant’s website, in another tab. Clicking “view comments” will allow the user to read the comments on this restaurant.

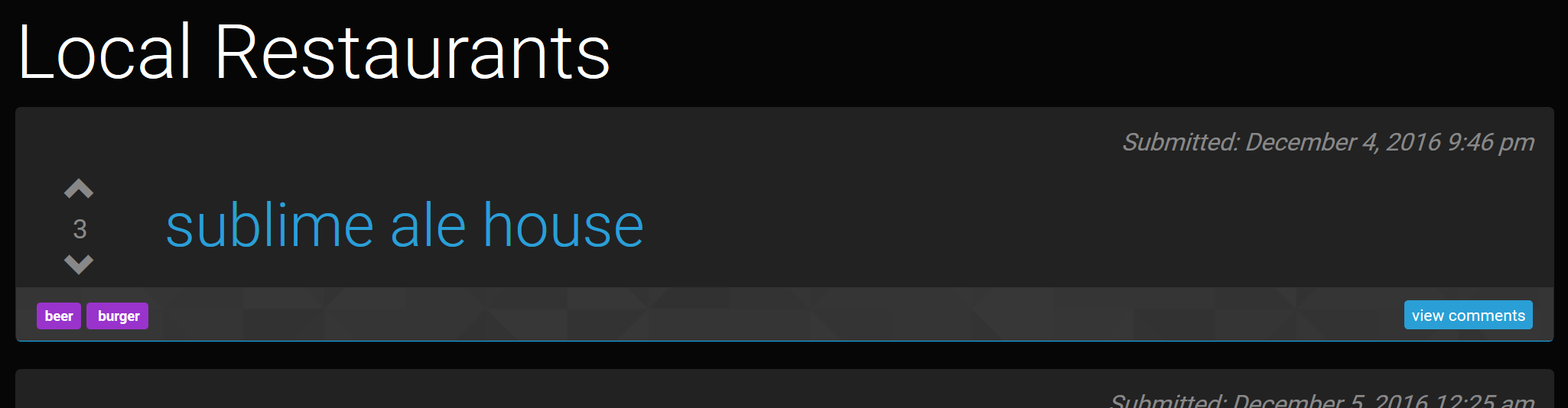


Figure - listing restaurants

### Searching Restaurants

From the homepage, any visitor may also search for restaurants using the provided search field. Search works in a unique way, as a discovery tool. Instead of searching for a specific restaurant such as Panda Express, one searches for an attribute of that restaurant, called a Tag (more on tags later). For example, If I was looking for Chinese food, I would search “Chinese” and the system would provide all the restaurants tagged as Chinese, sorted by aggregate score of votes. Essentially this returns the top Chinese restaurants, as voted by the users.

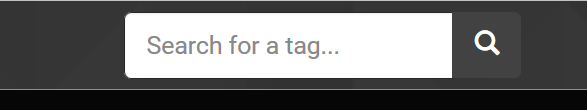


Figure - searching restaurants

### Viewing a Restaurant’s page

Restaurant pages are really simple. Why provide more info than what the restaurant’s website provides? Here is where users can view comments about a restaurant. This page is accessed by clicking the “view comments” link at the bottom of each restaurant post in the home page.

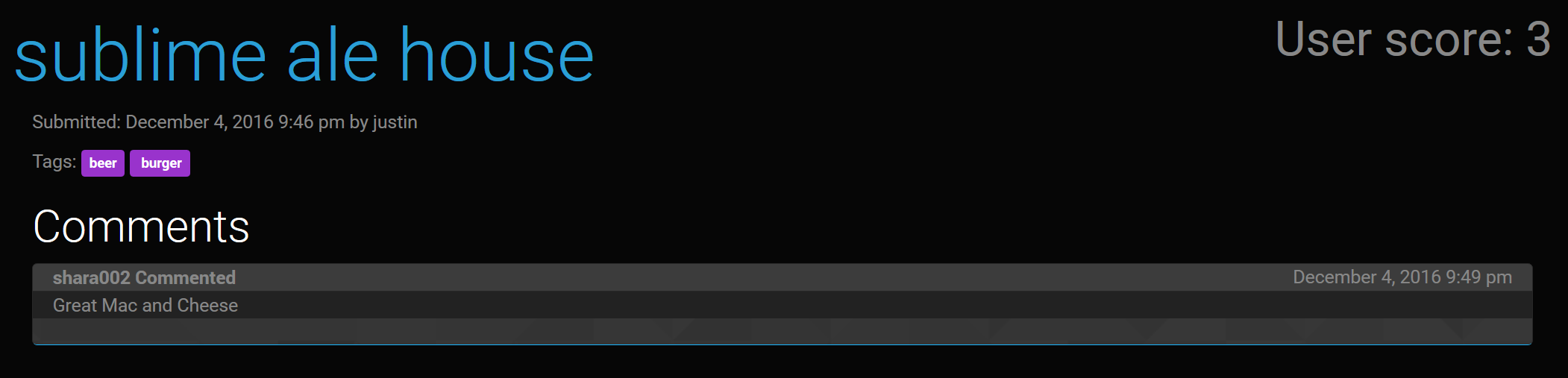


Figure - viewing comments as an anonymous user

### Signing Up for an Account

Signing up is dead simple. We only require a username and password. This is done to ensure absolute anonymity for our users. No email addresses are recorded. We believe that any user should be able to speak their mind about any restaurant (so long as they are not attacking anyone in particular) without fear of the consequences of, say, writing a negative review for a restaurant. Usernames are enforced unique.

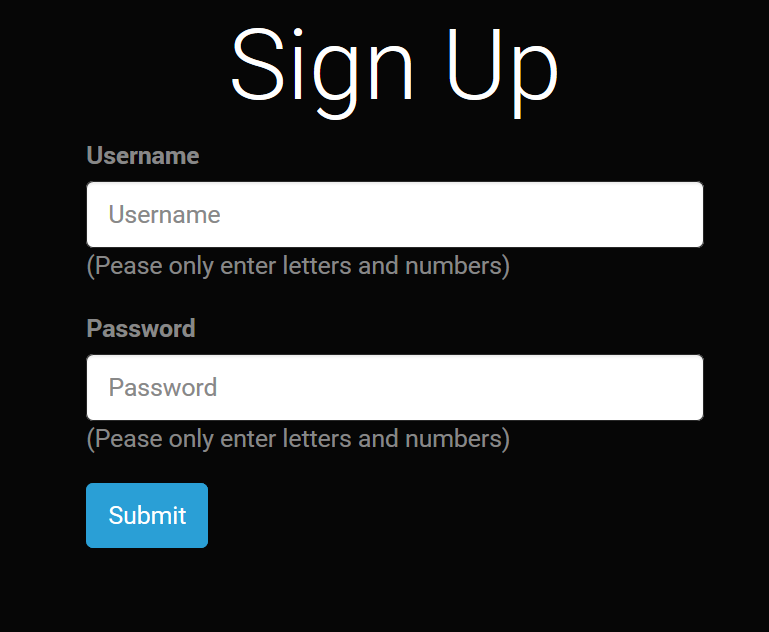


Figure - signing up

## Registered Users

Registered users have all the powers of anonymous users, plus the ones listed below. In addition, displaying restaurants changes slightly. The system remembers which restaurants the user has voted on in the past, so it highlights them. If one has upvoted a restaurant in the past, the up arrow and score are highlighted in green to indicate a positive vote was made in the past. Downvotes are indicated by highlighting the down arrow in red.

### Logging In

Logging in is just as easy as signing up, just provide a username and password.



Figure - logging in

### Submitting New Restaurants

Submitting a restaurant is easy. Three fields are provided, two are required. Restaurant name is as it sounds, and is required. URL is a link to the restaurant’s web site, also required and must be unique. Lastly, Tags are provided by the user. A tag is a one-word descriptor for the restaurant like ‘beer’ if the restaurant is known for serving beer, or ‘pizza’ or ‘wings.’ These attributes are user-defined on the fly and are used when searching for a restaurant. Users are permitted up to 5 tags at 16 characters each, and this field is entirely optional. Users may also delete a restaurant they have submitted.



Figure - submitting a restaurant

### Voting

Users are allowed to vote for their favorite or least favorite restaurants, which affects the aggregate score for a restaurant, and its rank within search. Each user is allowed one vote per restaurant, but a vote can be changed at any time from positive to negative or vice-versa.

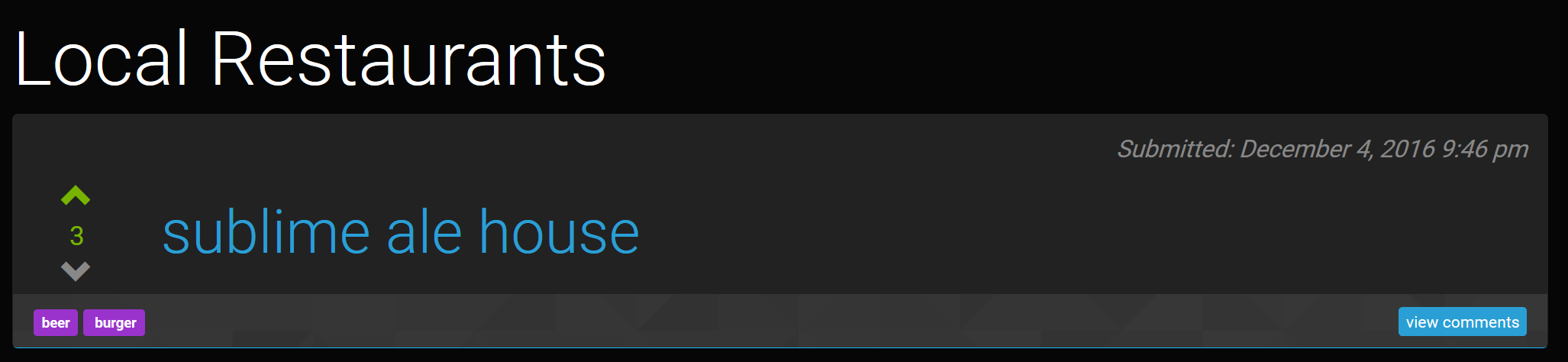


Figure - a restaurant that's been upvoted by the current user

### Adding a Comment

Users are free to speak their minds about any restaurant by means of commenting. Users may also delete their own comments.



Figure - submitting a comment on a restaurant

## Admin Users

Admin users have all the powers of regular users, plus the following.

### Listing and Promoting Users

Admins have the ability to make more admins. This is done by promoting regular users. An admin may also be demoted back to a regular user by another admin. An admin may not change his or her own admin status.

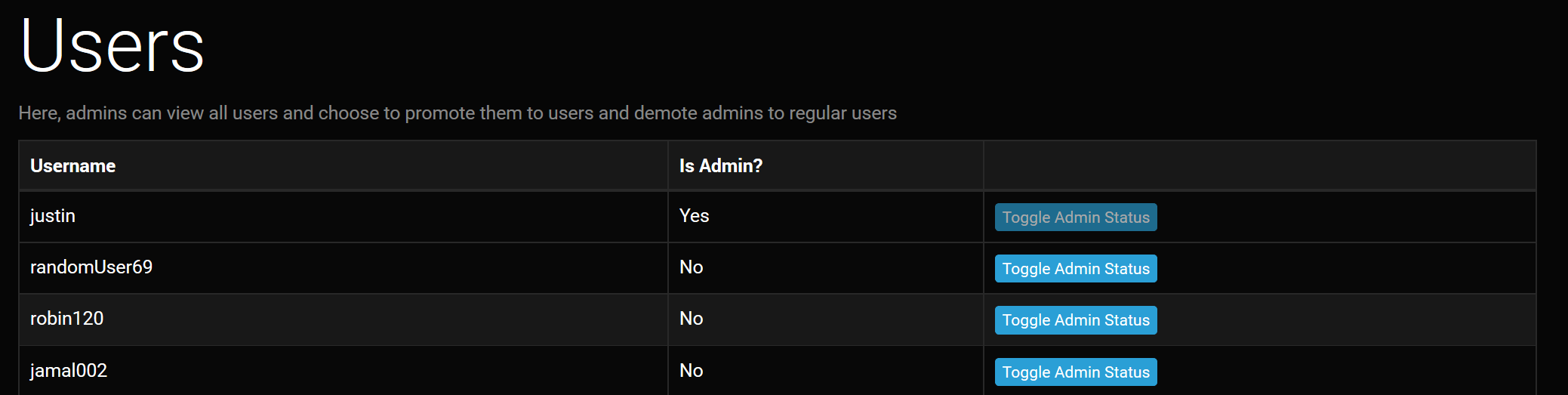


Figure - listing users and toggling their admin status

### Enhanced moderation powers

As an administrator, one can delete any restaurant or comment. This is done to help mitigate abusive actions by regular users.

# Web Pages and Their Interrelationships

Our site is relatively simple. There are only a few pages that really matter.

## Index.php

This page handles displaying all restaurants and search results. It sorts results by total score and provides a means for registered users to vote on restaurants.

## addRestaurant.php

Allows users to submit new restaurants. This page is reached by clicking “add restaurant” in the navbar as a registered user. Upon submitting a restaurant, the user is redirected to the restaurant.php page which corresponds to this new restaurant. The form handler for this page is addRestaurantVerify.php, which is invisible to the user but does the legwork for actually entering a restaurant into the database.

## restaurant.php

Displays a single restaurant and its comments. Also allows registered users to submit new comments and delete their own comments. The original submitter of this restaurant is also allowed to delete this restaurant at their discretion. This page is visited by clicking “view comments” for any given restaurant on index.php or by submitting a new restaurant.

## login.php, logout.php, login.php, loginVerify.php, signUpVerify.php

These pages handle logging in, signing up, and logging out. They create and destroy PHP session data to track user sessions. loginVerify and signUpVerify are form handlers for login and sign up, respectively. They do the dirty work for checking/entering user data into the database and also creating session data.

## loginFailure.php

If a user tried to login with a bad username/password, they are taken to this nice “try again” page. The same happens if a new user tries to register with bad credentials.

## listUsers.php

This page is accessible by admins only. It allows admins to promote or demote users to admin status. It also allows admins to view all registered users for the site. This page is accessible via the top navbar if signed in as an administrator.

## Misc. Form Handler Pages

Other pages not mentioned are largely invisible to the user and work to manipulate the database. For example, upvote and downvote each get their own php files to handle the associated up/downvote logic. The same can be said for addComment, deleteComment, deleteRestaurant, etc.

# Database Design and Implementation

Our database is very simple, consisting of only six tables. The creation scripts and sample data can be found in the SQL folder alongside the rest of our code for this project.

## Table Descriptions

### User

Includes user data such as usernames or passwords. Usernames are enforced unique.

### Restaurant

Includes restaurant data such as name, url, when it was submitted, and the id of who submitted it. Restaurant URLs are unique.

### Vote

A vote is an entity in the database which contains the following: the user id of who voted, to prevent a user form voting twice for the same restaurant, the restaurant id, and a bool value which determines whether the vote is positive or not. The sum of all these votes for a restaurant determines its score. The combination of a user id and a restaurant id are unique.

### Comment

A comment is a bit of text written by a user about a restaurant. This table contains a restaurant id to which a comment belongs, the user id of who wrote it, the time it was written, and the text itself.

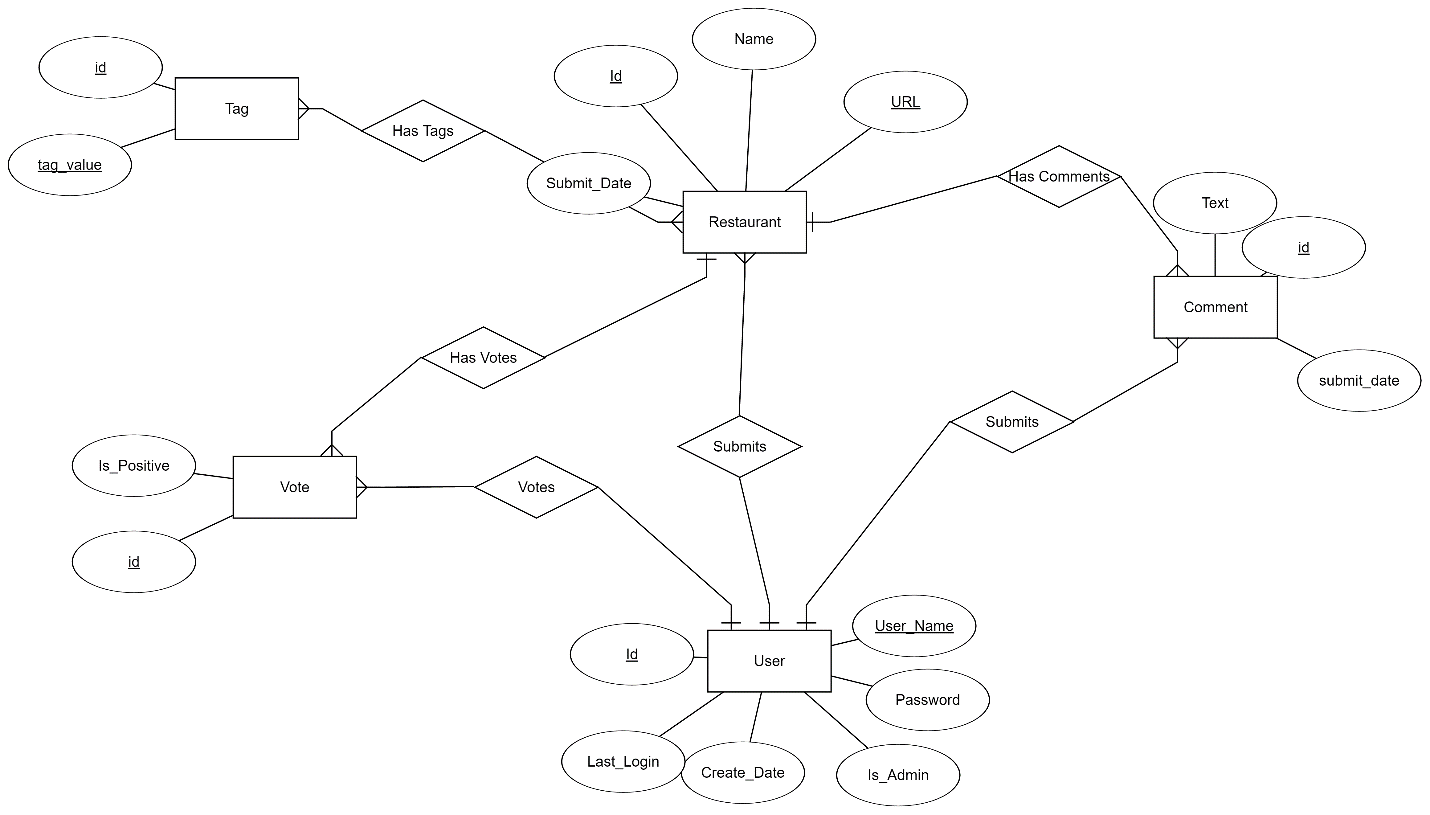
### Tag

A tag is a one-word descriptor for a restaurant. This is all we store in the database, alongside a unique id for this tag.

### Tag Restaurant Mapping

Tags have a many-to-many relationship with restaurants. This allows a restaurant to have many tags, and allows a tag to be reused for many restaurants. Each row is simply a restaurant id to which a tag belongs, and the tag id for a particular tag.

## ERD



# Web Design Issues such as resolved difficulties, possible future improvements and enhancements

We encountered a few issues when designing this site. The first being that we are using PHP 5.5, which was released over 10 years ago. PHP has changed a lot in the past 10 years. For example, a lot of support is missing for JSON. We had originally aimed to use a lot of AJAX to dynamically load and create content for the site. However, without the json\_encode and json\_decode functions provided in modern PHP, this wasn’t feasible. We were able to use some AJAX but not to the originally planned extent.

As far as future improvements or enhancements go, one is: why stop at restaurants? We have a robust means for cataloging, listing, and voting on objects. Why not use the same logic for something like… Tukwut Cars? Or Tukwut Computers? Users could submit cars, for example, and tag them with attributes like “safe” or “crossover” or “convertible” and users can vote for their favorite cars. Then when shopping for a car, a user could search for a certain feature of a car and find the best car that meets that critera.

We’d also like to fix our search. Currently one can only search by one tag at a time, due to a bug in PHP’s string concatenation functions. Ideally we’d allow an arbitrary number of search terms.

Otherwise, we wish to keep the site simple. The main draw to the site is the content and potential community, not flashy features. The content is the feature.

# Implementation Details

We had an issue trying to access our team1 database. As such, we just elected to use one of our member’s databases – vanbi002. The username and password for this db are both simply vanbi002. Dr. Wu should have full permissions on this database.

The rest of the code resides within the team1 folder on the cis444 webserver.

You are free to create your own username and use the site, or log in as an admin:

Username: Justin

Password: password

# User Contributions

## Rajak Jamal

Rajak wrote a lot of the initial html for the site and helped conceive the initial design and layout. He also wrote the PHP which enabled the display and submission of comments.

## Curtis Robinson

Curtis wrote the documentation for the site and built a majority of the final presentation. Curtis was also respoinsible for the administrative powers on the site – listing and promoting users; deleting comments; and deleting restaurants

## Shane Sharareh

Shane, like Rajak, was instrumental in creating the initial design for the site. He also created the logic for displaying and submitting a restaurant.

## Richard Stelly

Richard handled up/down voting and scoring. He also helped to construct the database.

## Justin Van Bibber

Justin was the project manager, advisor, and wrote a lot of documentation for the site. He was involved in the initial design and layout as well.

## Travis Weil

Travis handled the searching logic and also most of the index.php. He, along with Richard, worked to design the database for the site.