Restaurant Rating System:
DB Proposal

Team members

Farid Jafri (fjafri@stevens.edu)
Ryan Gray (rgray1@stevens.edu)
Leo Schaffner (lschaffn@stevens.edu)
Ziyou Shang (zshang3@stevens.edu)

Users

The Users collection will contain information about a user. The user's account information, and the references to the user's reviews and comments will be stored here.

```
{
        _id: Automatically generated,
        "username": "xX_R35TAURANT_N0SC0P3R_Xx",
        "firstName": "John",
        "lastName": "Doe",
        "reviews": ["objectIdOfReview100", "objectIdOfReview101"],
        "comments": ["objectIdOfComment200", "objectIdOfComment201"],
        "email": "email@compuserve.net",
        "password": "random password",
        "isOwner": true,
        "profilePicture": BSON object,
        "bio": "Something about me…"
}
```

Name	Type	Description
_id	ObjectID	Global identifier specific to the user, generated when the user is created
username	string	The username of the user
firstName	string	The first name of the user
lastName	string	The last name of the user
reviews	array[string]	An array containing the id's of

		reviews which the user has posted
comments	array[string]	An array containing the id's of comments which the user has posted to a review
email	string	The email which is linked to the account
password	string	The password which is linked to the account
isOwner	boolean	Determines whether the user is a customer or a restaurant owner, cannot be changed once set
profilePicture	BSON	Profile picture of the user
bio	string	A short biography of the user

* Password will be encrypted, method not currently known

Restaurants

The Restaurants collection will contain all details about the restaurant. These details include name, owner's email, category, featured items, menu, methods of service, location, nearby restaurants, and hours. Reviews will be attached to the restaurant's page, and will be used to determine the restaurant's rating and frequently used tags to describe it. Restaurants can only be made by "owner" accounts.

{
  _id: Automatically generated id,
  "name": "Starbucks",
  "owner": "email@compuserve.net",
  "category": "Fast Food",
  "rating": 3.6
  "reviews": ["1k2j312j4o1ij24", "4gh5v64n56gn"],
  "featuredItems": [{"Coffee": "$1.99"}, {"Espresso", "$1.99"}, {"Cappuccino", "$2.99"}],
  "menu": "link@menus.com",
  "serviceMode": ["Dine-in", "Takeaway"],
  "location": "1 Castle Point Terr, Hoboken, NJ",
  "nearbyRestaurants": ["22k3j5n2kjn35", "32k5j23i5j2ilj5"],
  "hours": {"Sunday": "8:00 AM - 6:00 PM", "Monday": "6:00 AM - 6:00 PM", "Tuesday": "6:00 AM - 6:00 PM", "Wednesday": "6:00 AM - 6:00 PM", "Thursday": "6:00 AM - 6:00 PM", "Friday": "6:00 AM - 6:00 PM", "Saturday": "10:00 AM - 6:00 PM"},
  "isOpen": true,

```
  “frequentTags”: [“Awaiting more reviews”],
  “longitude”: 50.33,
  “latitude”: 40.75
}
```


Name	Type	Description
_id	ObjectID	The id of the restaurant
owner	string	The id of the user from Users collection
name	string	Name of the restaurant
category	string	The type of establishment the restaurant is, could be Fast Food, Family, Fine Dining, Ethnic, and so on.
rating	number	The average rating of the restaurant
reviews	array[string]	Array of id from Reviews collection
featuredItems	array[object{string:string}]	Array of objects of featured dishes from the restaurant, with prices
menu	string	A link to the menu document (i.e. pdf)
serviceMode	array[string]	Type of services - Dine-in, Takeaway, Delivery
location	string	Street, city, and state the restaurant is located in
nearbyResturants	array[string]	Brief list of nearby restaurants, will be automatically generated based on the location input, and updated with new restaurant additions
hours	object	Times of the day when the restaurant is open. An object that stores days as keys and hours as values

isOpen	boolean	Compares the user's local date/time to the hours object to determine if the restaurant is currency open
frequentTags	array[string]	Array of the most used tags from reviews. If it has less than three reviews, it will simply be ["Awaiting more reviews"]
longitude	number	Longitude of the location of the restaurant
latitude	number	Latitude of the location of the restaurant

*Alternatively, the menu can be displayed directly on the page if applicable

Reviews

The Reviews collection contains all the reviews for corresponding restaurants. Every review stored in the Reviews collection contains the restaurant being reviewed, the user that creates the review, the content of the review, the set of comments from users about the review, and also a criteria for potential malicious reviews that might need to be deleted. Reviews can only be made by "customer" accounts.


```
{
  "_id":ObjectId("5f9c6b12191487f5b311c460"),
  "restaurantReviewed":"5f9c6b3c31d9aaef6cfcfd4a",
  "user":"5f9c6b4dbec7a3c553c47d21",
  "rating":5,
  "dateOfReview": "10/30/2020",
  "content":"This is a good restaurant.",
  "sReview":0,
  "comments":[
     "5f9c6b5fe30a794b97dfae00",
     "5f9c6b6a97f2c2e4263733e5",
     "5f9c6b753e21a219b35c32c3"
  ],
 "tags": ["cheap", "healthy"]
}
```


Name	**Type**	**Description**
_id	ObjectId	The object id automatically

		generated by MongoDB
restaurantReviewed	string	The id of restaurant being reviewed
user	string	The id of user that creates the review
rating	number	The overall rating for the restaurant
dateOfReview	string	Date of the review created
content	string	The content of the review
sReview	number	A criteria for potential malicious reviews
comments	array	Set of user comments of the review
tags	array	A list of words that describes the review. Can be used for the search feature.

Comments

The Comments collection will contain the comment itself, which will be posted automatically to the review page the user is on, and the username of whoever posted the comment. It will also include the date the comment was posted, which will be fetched at the time of posting. Any user can make comments.

{
_id: Automatically generated ObjectId,
“commenter”: “5f9c6b4dbec7a3c553c47d21”,
“reviewId”: “bd8wa289-3v7v-4478-a831-e36a02de13w3”,
“comment”: “This was a really great review. I’ll never eat food the same way again.”,
“date”: “10/20/2003”
}

Name	**Type**	**Description**
_id	ObjectId	The object id automatically generated by MongoDB
commenter	string	ID of the user who posted

reviewId	string	reviewId will be automatically filled in for the user based on the review they are currently looking at
comment	string	Body of comment
date	string	Date comment is being posted