# **The Cocktail Lounge**

**Goal**

A go-to website to search for cocktails, get recipes, curate a list of favorites and much more.

**Users**

Anyone above the age of 18 who is interested to know more about cocktails. A place for amateurs to search for cocktail recipes, experts to quickly create new drinks with at-hand ingredients, with guidance on the best look for a drink.

**Data**

API Choice: The Cocktail DB

<https://www.thecocktaildb.com/api.php?ref=apilist.fun>

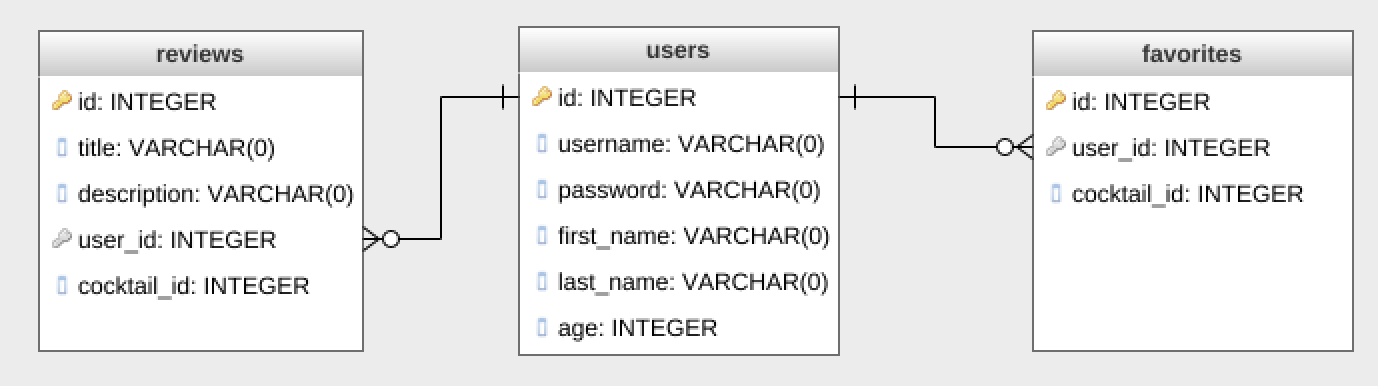
The API provides cocktail information by name, ingredients, alcohol level and even glass type.

**Outline**

Database Schema:

The schema will consist of a

1. Users table : id, username, password,age,first\_name, last\_name
2. Reviews table: id, title, description,user\_id, *cocktail\_id*
3. Favorites table: id,user\_id,*cocktail\_id*

**

User Flow:

* The website homepage will display a random list of cocktails with images. Users can also view the links to search for cocktails by various categories (search by name,ingredients, alcoholic/non alcoholic). The page also displays the login/logout/signup links.
* To further access the app,each user will need to sign up and login. Only adults above the age of 18 years will be allowed to signup.Users can see the list of cocktails that match the search criteria and link to the individual cocktail page that includes the recipe and user reviews.
* Option to save favorite drinks to the user account and post reviews if interested. There will also be provision to edit /delete the user account, reviews and to unfavorite drinks.

Clarifications

* API data vs postgres data: Since the external api provides the data, do I need to create a cocktails table? Is it possible to enable users to add new cocktail recipe and if so, where would i store that data?

We cannot add new cocktails to the main api database but since our api response consists of required cocktail data, we can create a small table in our server with the new recipes that we add and then check both api db and our table for results

.

* How to define the favorites table since this would be a subset of the cocktails database?

Since the api response includes the cocktail id, we can use that to create cocktail\_id in our favorites table.

Approach 1:when user favorites a drink, save only the cocktail id and each time you need to display it, send a api request.

Approach2: Create a cocktail table and when user favorites a drink, save all the required data (like id,ingredients,recipe etc) to our version of cocktail table and show the results from our table.