

# Whiskedin



**15<sup>th</sup> February 2019**

**Geraldo A. Vera Perez**

**Javier Bustillo Hernandez**

Prof. Bienvenido Velez

INSO4117-126

---

## Introduction

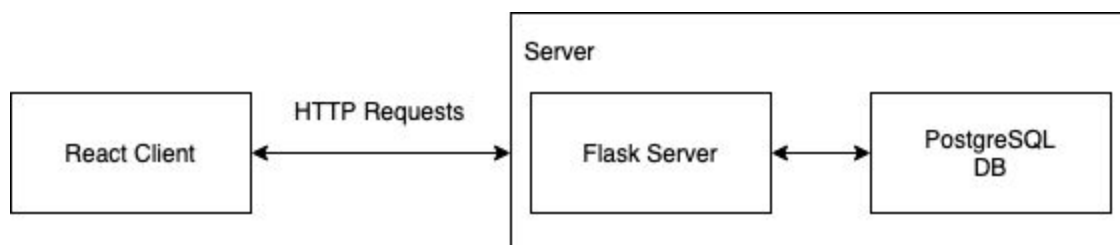
As a whiskey collector with a growing collection of bottles it gets hard to remember all the bottles you own and those you wish to add to your collection. It's hard to brag to your buddies about your collection if you can't remember it. It's also hard to remember every suggestion made by fellow friends.

Whiskedin is a web application for users to keep track of their current whiskey collection. Users can save their bottles by adding relevant information and storing an image of the bottle. The bottles are shown as “cards” to the user. Users can also share their whiskeys with other users, making this web application a kind of trading card platform for whiskey aficionados.

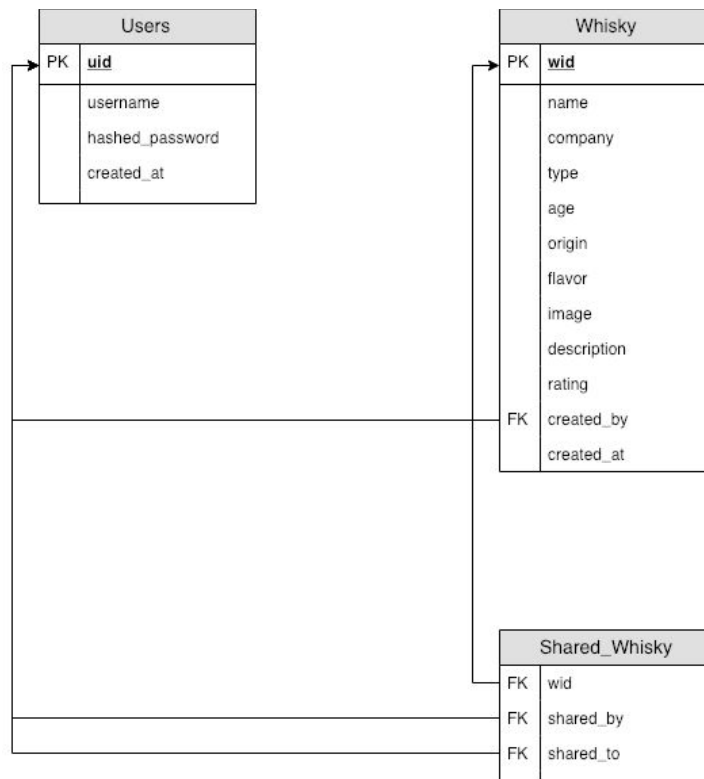
## Technical Description

System architecture:

This webapp will be a react client that communicates with a REST API that will be developed using flask, a web framework for python. This REST API communicates with a postgresql database and sends responses back to the react client.



## ER Diagram



# Wireframe

Log In

Sign Up

Whiskedin

A place to track your whiskey  
collection and discover  
new bottles

Whiskers

Logout

+

White Walker

Johnnie Walker

Image Here

Share

←

→

Username

Password

Log In

Username

Password

Confirm Password

Create Account

---

## Project schedule:

For each phase the following user stories are expected to be completed with their appropriate tests..

### Phase 1 Report and Demo Friday March 8

- As an unregistered user I can visit the app main page and learn about what the app can do for me and create a new account.
- As a registered user I can visit the app main page and login with my username and password.
- As a registered user I can sequentially scan through my list of collectible items forward and backward by clicking on right and left buttons respectively.

### Phase 2 Report and Demo Friday March 29

- As a registered user I can always look at and edit the information associated with the collectible item that I am currently viewing.
- As a registered user I can search for collectible items by specifying terms. The app will move to the first item beyond the current item that contains these terms in any of its information fields.

### Phase 3 Report and Demo Friday April 12

- As a registered user I can share my collectible items with other registered users
- As a registered user I can browse through all collectible items shared with me and add any of them to my list of items.

### Final Phase Report and Demo Friday April 26

- Extensive testing
- Complete any additional user stories identified throughout sprints

---

## Testing plan:

The tests for this web application will be divided in the following axis:

- User interface elements
- Input space
- Output space
- External interface
- Functionality

Once development of the web application begin, specific tests will be identified and developed in order of possible risk.

Tests will be organized by the following structure:

- Unit tests
- Integration tests
- System tests
- Acceptance test

Unit tests will be designed to test the input, output space of any applicable method. Unit tests will also cover the external interface of the webapp which is the REST API that will be developed to communicate the client to the database. Once sufficient unit tests have been developed, integration tests will be made to test the interaction between different modules in the each layer of the web application.

System tests will be designed to test the user interface elements and general functionality of the webapp, these tests will be developed primarily with selenium. The acceptance tests will cover all functionality that is expected to be working after every sprint.