The project uses Flask as the backend, and uses JS+Html as the frontend.

The frontend uses jquery and selectize.js (for the multi-select dropdown).

All requests between the webpages and the server are asynchronous, using post method, using json as datatype.

Python files

app.py	Create a Flask object
run_server.py	This file is used to run the server. Don’t combine “run_server.py” and “app.py” into a single file.
listener.py	File for rest api.
service.py	The service layer, this file is used to generate and store images
DAO.py	Data access object
model.py	File for the model
networks.py	File for the model
utils.py	File for the model

File Structure

According to the default Flask blue print. Static content such as webpages, JavaScrip files, images, CSS files are in ”static” folder. Browsers can visit files in this folder through urls.

css	css files
js	Currently empty folder
generated_images	generated images
index.html	Website main page
assists	Examples for gallery
model.png	Image for the main page

Sqlite database structure
Table name:food

ID	Integer	auto increment primary key
food_type	Text	not null
ingredients	Text	not null
image_name	Text	not null
feedback	Integer	Integer from 1 to 5
geo_location	Text	not null
browse_name	Text	not null

Some existing problems:

1. Although the multi-select dropdown tool works perfectly on Safari, Chrome and Firefox, it is not compatible with some unpopular web browser.
This tool doesn't designed to be compatibility to all browsers. Other multi-select dropdown tools have the same problem. I think there is no need to fix it since the tool works fine in most cases.

Potential problems when number of visitors increase:

1. The system cannot have too much files in a single folder (generated_image folder)
2. The Flask server provides multi-thread function. But it cannot handle too much connections
3. The file name of generated images may duplicate