MANUAL TEST PLAN

By Tianli Ding

➔

PREREQUISITES:

In order to use the web scraper and test the code, you could download a text editor, for example, Visual Studio Code. (Fig.1), or PyCharm CE. (Fig.2)

Fig.1

Fig.2

Eclipse Download Link:

https://code.visualstudio.com/download

PyCharm Download Link:

https://www.jetbrains.com/student/

Below we take PyCharm as an example.

➔

Operations and the Results (screen capture with description):

1. START

To run the web scraper program, you should put in “python main.py -h” in terminal to start.

Then the usage and help information will show up in your terminal as follows:

```
(.venv) (base) Tianlis-MBP:sp20-cs242-project tianliding$ python main.py -h
usage: main.py [-h] N URL

Take one integer as the number of recipes to be scrapped

positional arguments:
  N           an integer for the number of recipes
  URL         a recipe URL to start scrapping

optional arguments:
  -h, --help  show this help message and exit

Please enter one integer, and an recipe url
```

Therefore, the main.py takes one integer and a recipe URL from www.chinasichuanfood.com

The first integer is the number of books to be scrapped, which should be less than 2000.

The URL is a recipe URL as the start page of scrapping.

2. WARNINGS

If you have entered an integer that is larger than 2000, a warning will show up:

```
(.venv) (base) Tianlis-MBP:sp20-cs242-project tianliding$ python main.py 2001 'https://www.chinasichuanfood.
com/spicy-crispy-potatoes/'
=========================
You are requesting 2001 recipes to be scraped
https://www.chinasichuanfood.com/spicy-crispy-potatoes/ as a starting recipe url

========WARNING==========
please enter an integer that is less than 2000
========WARNING==========
```

Therefore, please make sure to enter integer that is smaller than 2000.

3. VALIDITY

If you have entered a valid number for recipes, and a valid URL, then you will see a message as below:

```
(.venv) (base) Tianlis-MBP:sp20-cs242-project tianliding$ python main.py 101 'https://www.chinasichuanfood.c
om/spicy-crispy-potatoes/'
=========================
You are requesting 101 recipes to be scraped
https://www.chinasichuanfood.com/spicy-crispy-potatoes/ as a starting recipe url
```

4. PROCESS

If the scraper start to scape, the process will be shown in the terminal, , as the scrapping going on.

```
Now we start scraping the recipes
Scraped 25% already
Scraped 50% already
Scraped 75% already
Finished scrapping
Now writing into json format
Finished!
```

It may take some time if you have requested a large number of recipes.

5. JSON FILE

The result of the scrape will be saved into json files, namely “recipes.json”.


```
[
    {
        "dish_name": "Spicy Crispy Potatoes",
        "dish_image": [
            "https://www.chinasichuanfood.com/wp-content/uploads/2017/11/Spicy-potatoes_-2.jpg"
        ],
        "prep_time": "no details",
        "cooking_time": "PT10M",
        "total_time": "PT25M",
        "ratings": "5 out of 5",
        "calories": "442 kcal",
        "ingredients": [
            "500 g small potatoes or large potatoes",
            "2 tbsp. cooking oil",
            "2 scallions ( ,white part and green part separated)",
            "2 tbsp. chopped garlic",
            "1 tbsp. chopped ginger",
            "1/4 red onion",
            "pinch of salt",
            "1 small bunch of coriander",
            "1 tbsp. Chinese chili oil ( ,optional)",
            "2 tbsp. red chili pepper powder",
            "1 tbsp. freshly ground cumin powder"
        ],
        "cooking_methods": [
            "Peel and cut large potatoes into one bite size pieces or if you use small baby potato
```

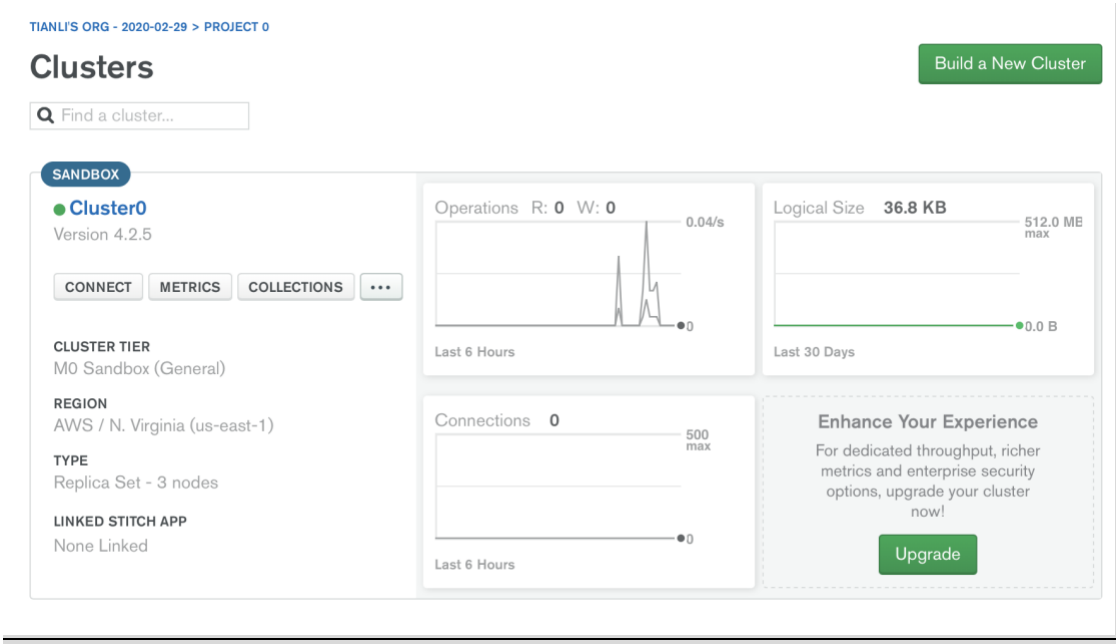

➔ INFORMATION INCLUDED

The information included in the json file are:

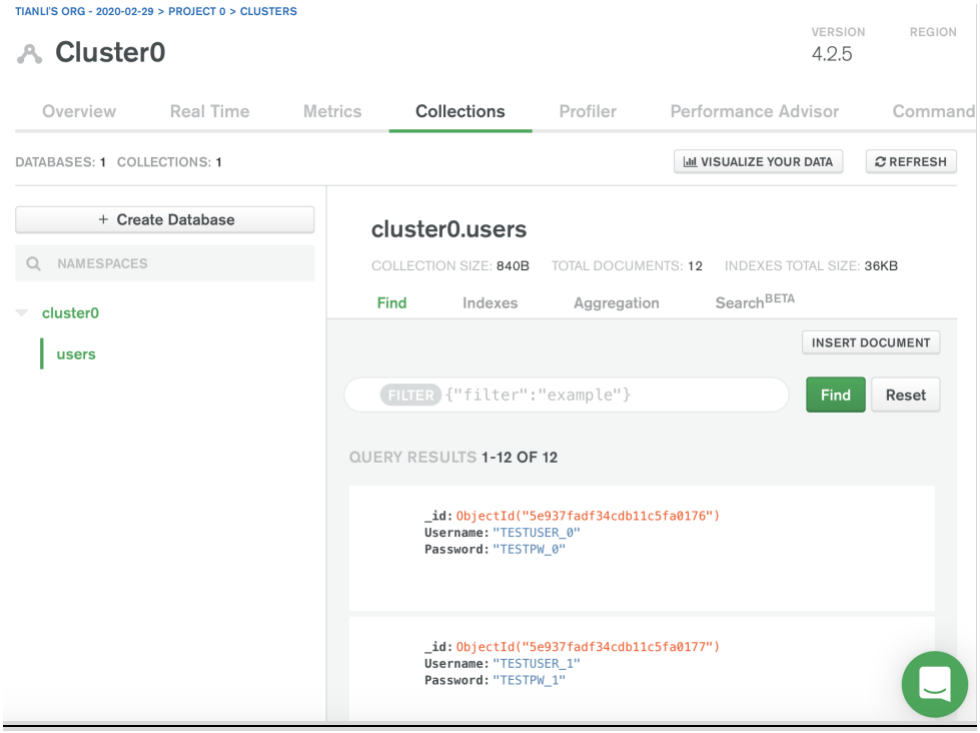
- Dish name
- Dish image
- Preparation time
- Cooking time
- Total time
- Ratings
- Calories
- Ingredients
- Cooking methods
- Recommendations

6. DATABASE

For this project, we use MongoDB to store the user information we scraped.



In the page below, the cluster0 contains one collection of users, storing user information.



7. UNITTEST

To further test the functionality of the scraper, there is unit test designed for the scraper class and user class.

In “logic_test.py”, you could see all the test for each function as well as tests for json file and database.

To run the logic_test.py, you could type “python logic_test.py” to view the overall test result.

In order to view the details of each test, you can type “python logic_test.py -v”.

If you see “OK” for each of the test, then the scraper and user class should be function fine (see below).

```
test_read_from_database (__main__.Test) ... ok
test_write_to_database (__main__.Test) ... ok

----------------------------------------------------------------------
Ran 10 tests in 4.398s

OK
```

SECTION 2

1. TEST API

Before running the API, you have to run the test in order to make sure the API functions well. Therefore, you should first run “pytest” in the folder “test”.

There are six tests to evaluate the functionality of the API. All of them need to be passed to make sure the whole program is functional.

```
(.venv) (base) Tianlis-MBP:test tianliding$ pytest
========================================= test session starts =========================================
platform darwin -- Python 3.7.4, pytest-5.4.1, py-1.8.1, pluggy-0.13.1
rootdir: /Users/tianliding/sp20-cs242-project/test
collected 6 items

api_test.py ......                                                                            [100%]

========================================== 6 passed in 1.62s ==========================================
```

To see the details of each test, you can run “pytest -v”.

```
(.venv) (base) Tianlis-MBP:test tianliding$ pytest -v
========================================= test session starts =========================================
platform darwin -- Python 3.7.4, pytest-5.4.1, py-1.8.1, pluggy-0.13.1 -- /Users/tianliding/sp20-cs242-proj
ect/.venv/bin/python3
cachedir: .pytest_cache
rootdir: /Users/tianliding/sp20-cs242-project/test
collected 6 items

api_test.py::test_index_get PASSED                                                            [ 16%]
api_test.py::test_home_page_get PASSED                                                        [ 33%]
api_test.py::test_user_get PASSED                                                             [ 50%]
api_test.py::test_user_post PASSED                                                            [ 66%]
api_test.py::test_user_delete PASSED                                                          [ 83%]
api_test.py::test_user_put PASSED                                                             [100%]

========================================== 6 passed in 1.30s ==========================================
```

If you have already passed all six test, then you are free to go through information below.

2. RUN API

In order to run API, you have to type “export FLASK_APP=web_api.py” then “flask run” in terminal.

```
^C(.venv) (base) Tianlis-MBP:sp20-cs242-project tianliding$ flask run
 * Serving Flask app "web_api.py"
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

Then you can copy paste the website above on a web browser.

3. RUN WEBSITE

Then, you can start the website by typing “yarn start” in a new window in terminal.

If everything is correct, you will see it output these messages.

```
Compiled successfully!

You can now view app in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.10.15:3000

Note that the development build is not optimized.
To create a production build, use yarn build.
```

Then you can copy paste the website above on a web browser.

Note the ports for website and the API are different, one 5000, another 3000.

4. NAVIGATION

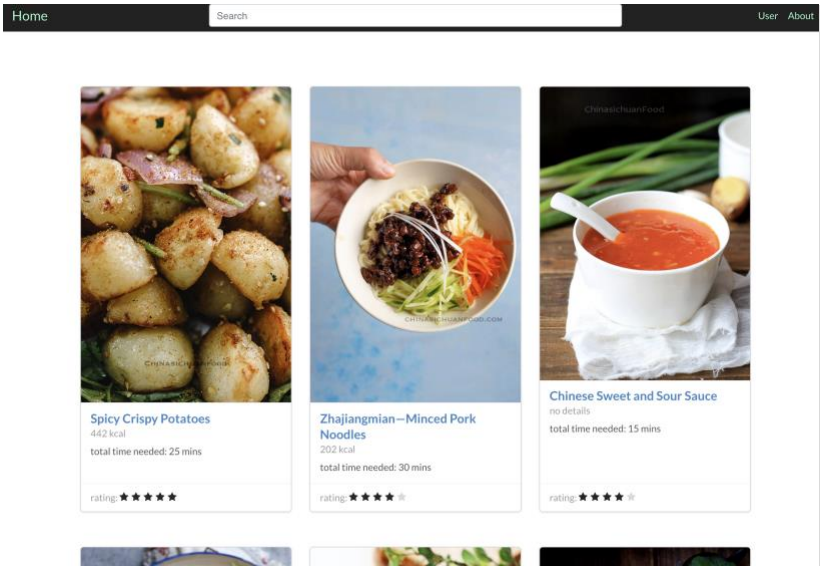
In “localhost:3000”, you can find home page.

Each page has a navigation bar, so that you can click “home” to go to the home page, “user” to go to the user page.

The search function will be introduced in the future.

5. HOME PAGE

In home page, except the navigation bar that we have introduced, it also includes all the recipe previews:

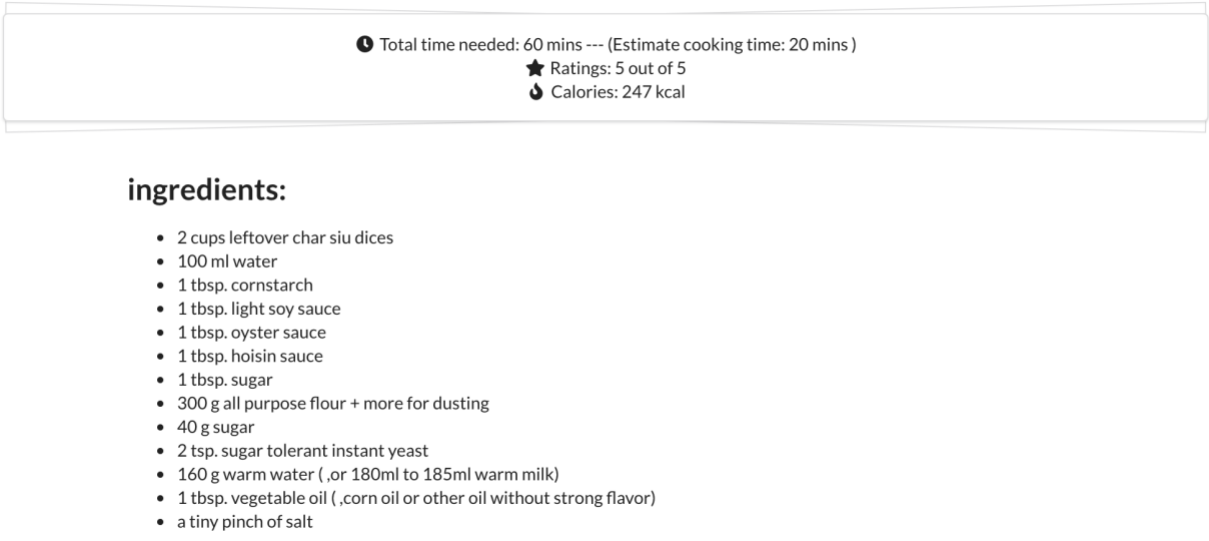


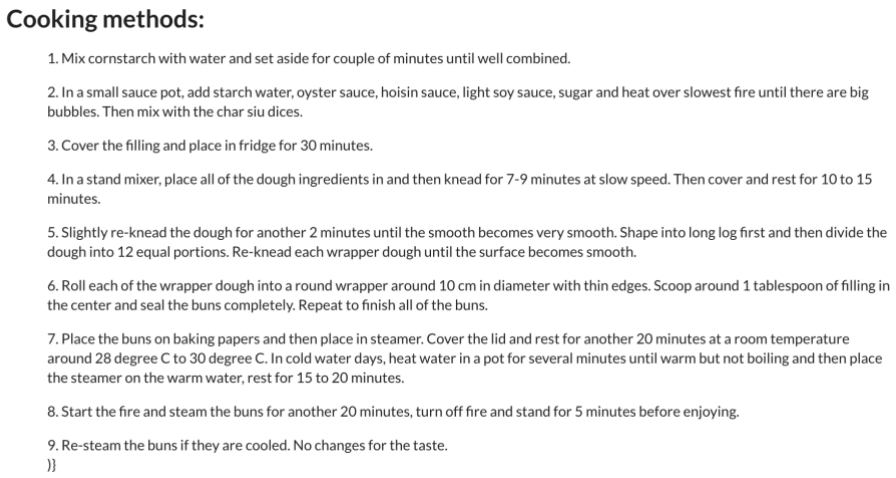
Each recipe preview will include a picture of the dish, the dish name, calories of the dish, the total time needed for cooking, and the rating. See example below:

If you click on the name of the dish, you will be directed to recipe page.

6. RECIPE PAGE

In recipe page, you will see more information about the dish, including the ingredients needed for the dish and the cooking methods.





Cooking methods:

1. Mix cornstarch with water and set aside for couple of minutes until well combined.

2. In a small sauce pot, add starch water, oyster sauce, hoisin sauce, light soy sauce, sugar and heat over slowest fire until there are big bubbles. Then mix with the char siu dices.

3. Cover the filling and place in fridge for 30 minutes.

4. In a stand mixer, place all of the dough ingredients in and then knead for 7-9 minutes at slow speed. Then cover and rest for 10 to 15 minutes.

5. Slightly re-knead the dough for another 2 minutes until the smooth becomes very smooth. Shape into long log first and then divide the dough into 12 equal portions. Re-knead each wrapper dough until the surface becomes smooth.

6. Roll each of the wrapper dough into a round wrapper around 10 cm in diameter with thin edges. Scoop around 1 tablespoon of filling in the center and seal the buns completely. Repeat to finish all of the buns.

7. Place the buns on baking papers and then place in steamer. Cover the lid and rest for another 20 minutes at a room temperature around 28 degree C to 30 degree C. In cold water days, heat water in a pot for several minutes until warm but not boiling and then place the steamer on the warm water, rest for 15 to 20 minutes.

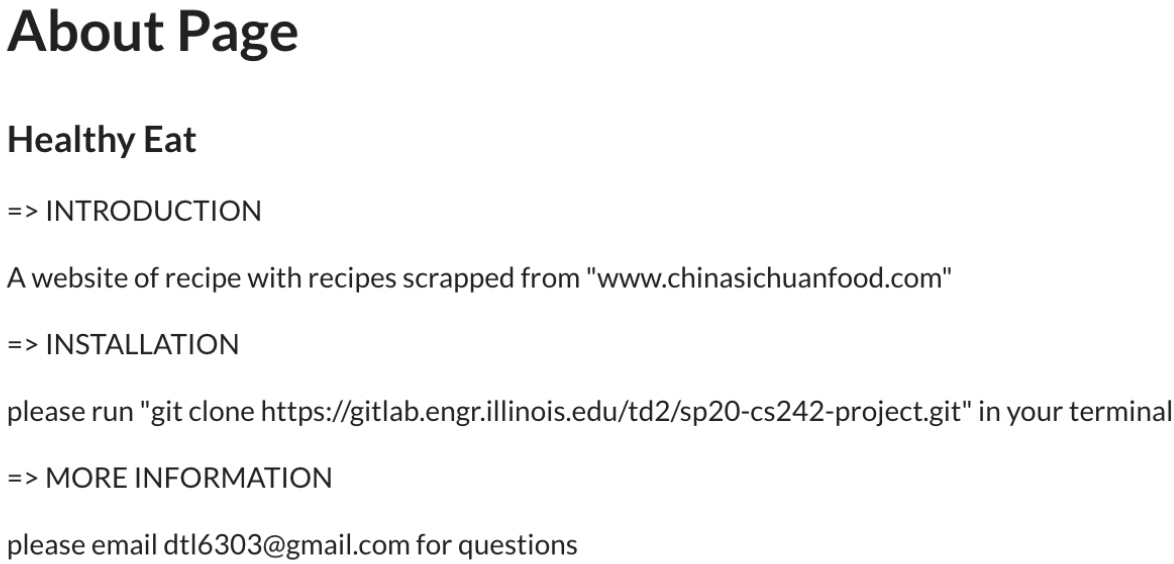
8. Start the fire and steam the buns for another 20 minutes, turn off fire and stand for 5 minutes before enjoying.

9. Re-steam the buns if they are cooled. No changes for the taste.
)}

7. ABOUT PAGE

In about page, there are some information related to this website.

Information includes the introductions, way to fork, email for questions.



About Page

Healthy Eat

=> INTRODUCTION

A website of recipe with recipes scrapped from "www.chinasichuanfood.com"

=> INSTALLATION

please run "git clone https://gitlab.engr.illinois.edu/td2/sp20-cs242-project.git" in your terminal

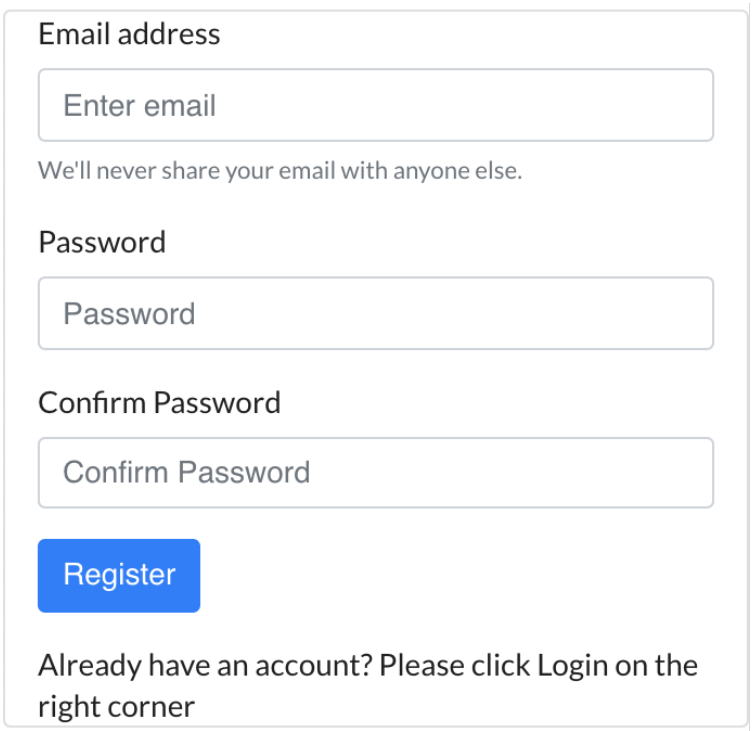
=> MORE INFORMATION

please email dtl6303@gmail.com for questions

Section 3

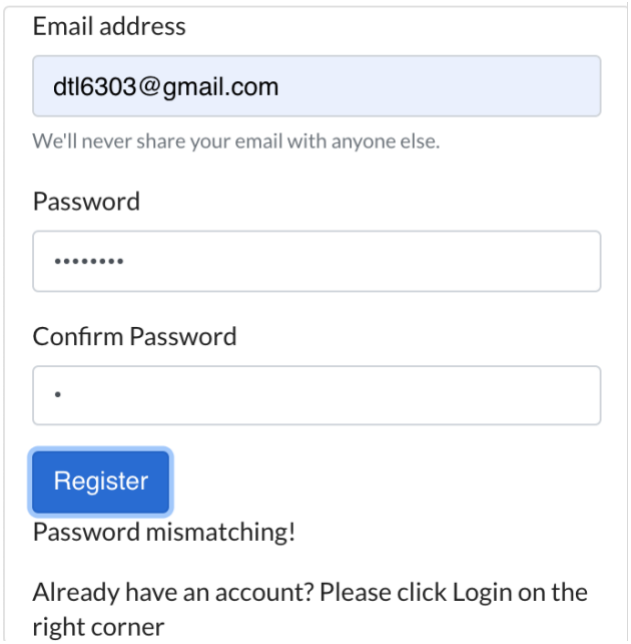
➔ REGISTRER

1. Register page



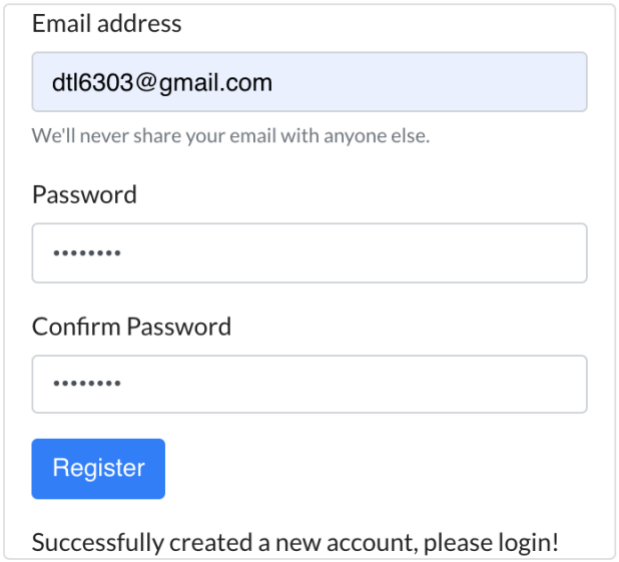
If you do not have an account, then you have to register first. By typing in your email address, and password twice, you will have a new account.

2. Password and confirm password not match



If they are mismatching, then a warning message will show.

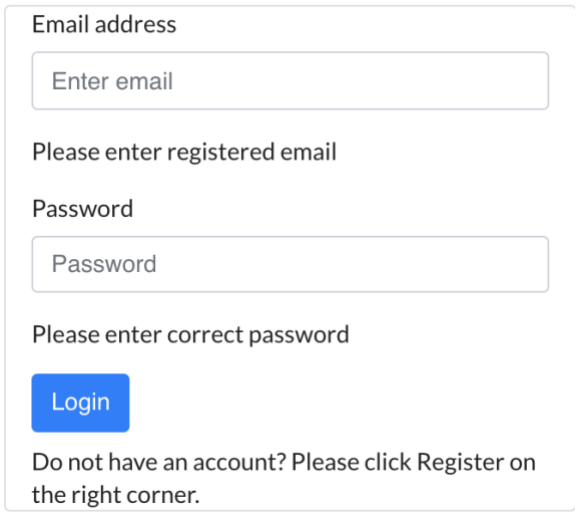
3. Successfully registered



If you see the message “successfully created a new account, please login!” then you have already successfully created a new account.

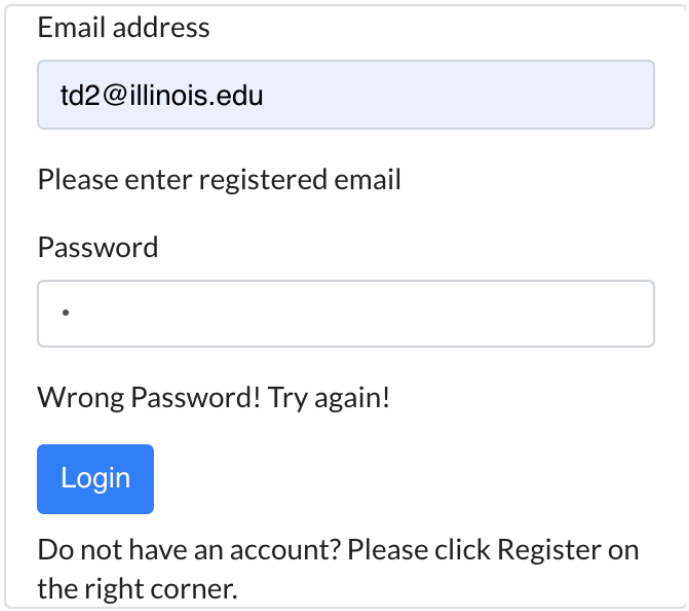
➔ LOGIN

4. Login page



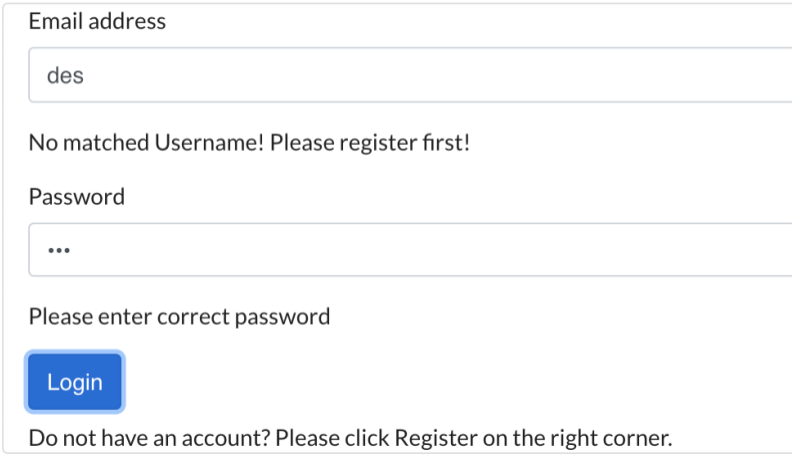
To login to the account and see the data, you have to input correct email address and password.

5. Wrong password



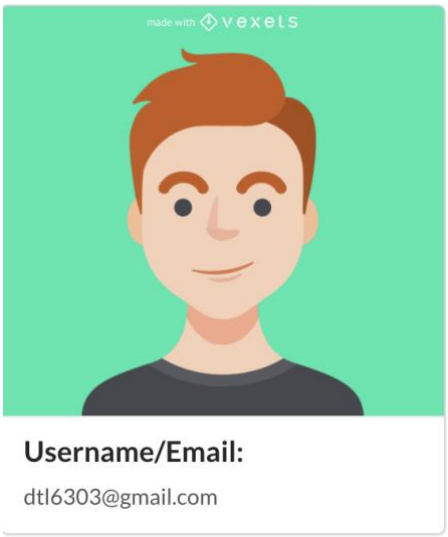
If the password you entered is wrong, but the email address is registered, you will see “Wrong password! Try again!” on the page. Therefore, you have to input a correct password in order to login

6. Wrong Email



If you have entered an unregistered email address, you will see the warning message “No matched Username! Please register first!”. Therefore, you have to enter a registered email. If you have not registered yet, please register first.

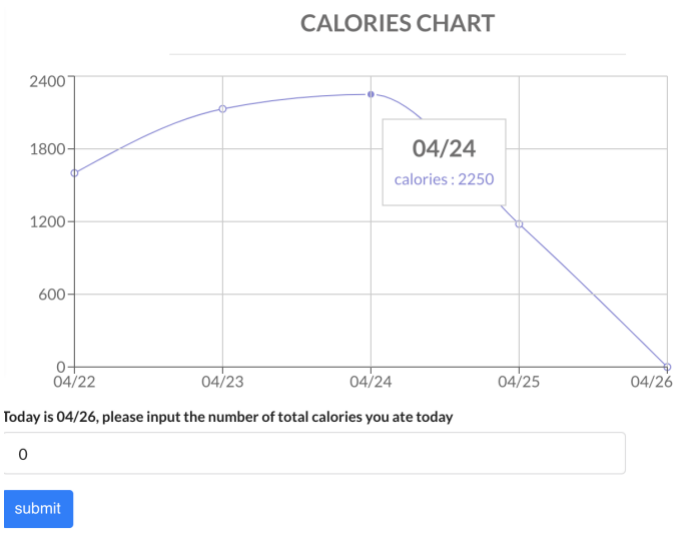
7. <u>User information</u>



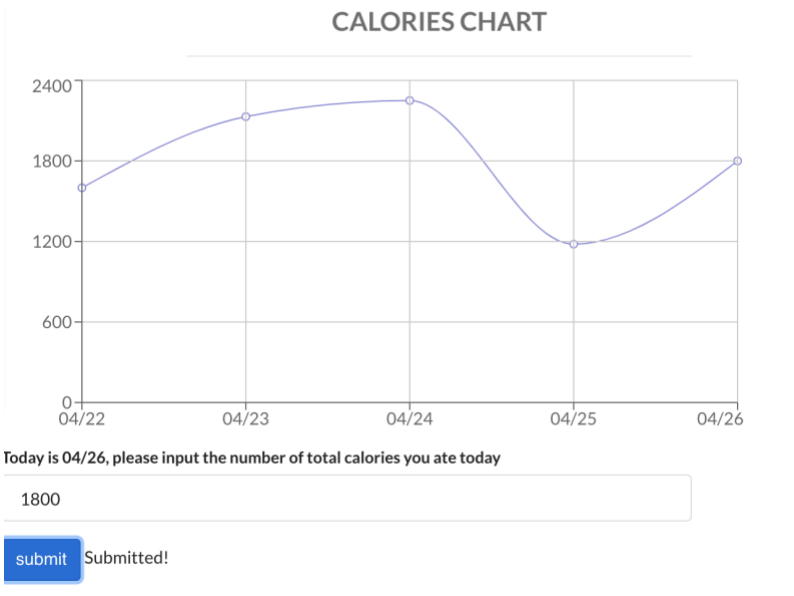
On the left of the user page, you can see a card will default image and username.

➔ Data analysis

8. Calories graph & input

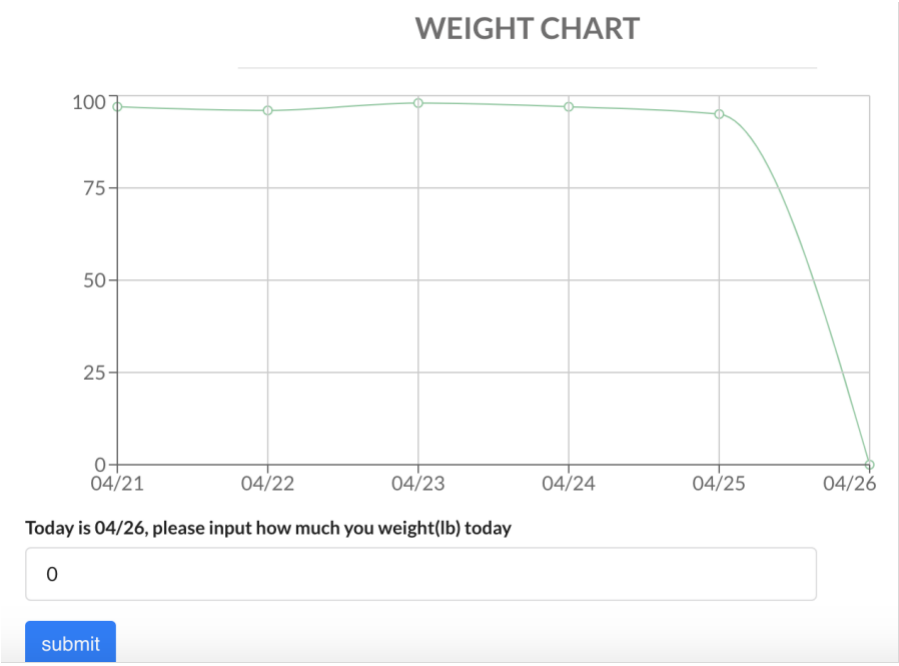


In calories section, you will see the date and corresponding calories you recorded in a graph. Below the graph, there is a form to input the calories you have eaten today. The default is 0.



If you have successfully submitted the number, you can see “submitted!” as a reminder, and corresponding graph.

9. <u>weight graph & input</u>



The weight chart is similar to calories graph. By submitting your weight, you will get a complete graph as follows: