

Application Manual

Team Name: RunTime Terror

Project Name: Food Planner2

Github Link: <https://github.gatech.edu/gt-cs6440-hit-fall2019/FoodPlanner2>

Team Member

- Andrew Lo - Developer
- Youjung Kim - Developer
- Vaneesh Bahl - Quality Analyst/Manager
- Dale Park - Project Manager
- Donald Ford - Developer

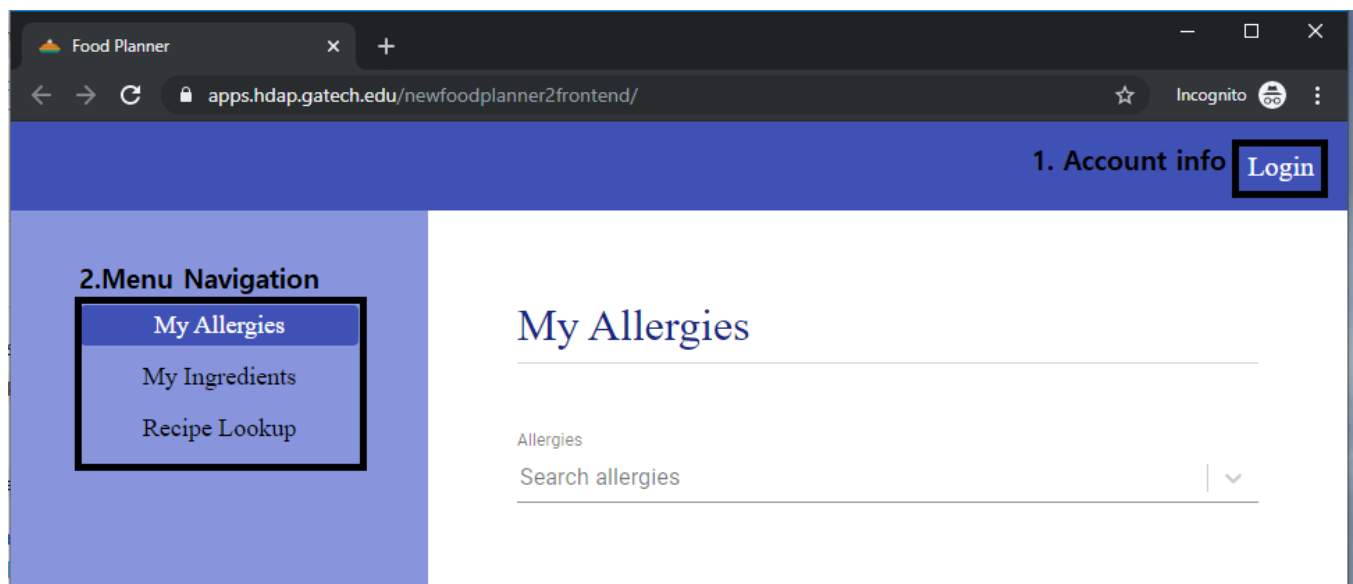
Contents

- 1. Application Access
- 2. Account Management
 - 2.1. Log in
 - 2.2. Sign up
 - 2.3. Log out
- 3. My Allergies
- 4. My Ingredients
- 5. Recipe Look up
- 6. Local Application Deployment
 - 6.1. How To Run Local Build
 - 6.2. How To Access Local Database
- 7. Production Application Deployment
- 8. Checks To See If The Servers Are Alive
 - 8.1. Production Servers
 - 8.2. Local Servers
 - 8.3. Postman Script
- 9. Version

1. Application Access

Access our Food Planner application using the below url. It directs you to the application with Login in the upper right corner and the Menu Navigation on the left side.

URL: <https://apps.hdap.gatech.edu/newfoodplanner2frontend/>



2. Account Management

2.1. Log In By clicking Login from the initial page, you can access the login page. Using the test credential below, we can sign into the application.

Username: test, Password: test

My Food Planner

Username

Password

LOGIN

Don't have an account [Sign up!](#)

Login in with example creds:
Username: test, Password: test

2.2. Sign Up Clicking “Sing up” button takes you to the Sign Up page. Enter the required information as well as current allergies. It will create an account on FoodPlanner as well as create a patient on HAPI FHIR with the allergy conditions .

Sign up

First Name
Eliot

Last Name
Ryan

Username
starbaker

Creates an account on FoodPlanner.

Password
....

Creates a Patient on HAPI FHIR with the allergy conditions.


Email
test@test.edu

Allergies

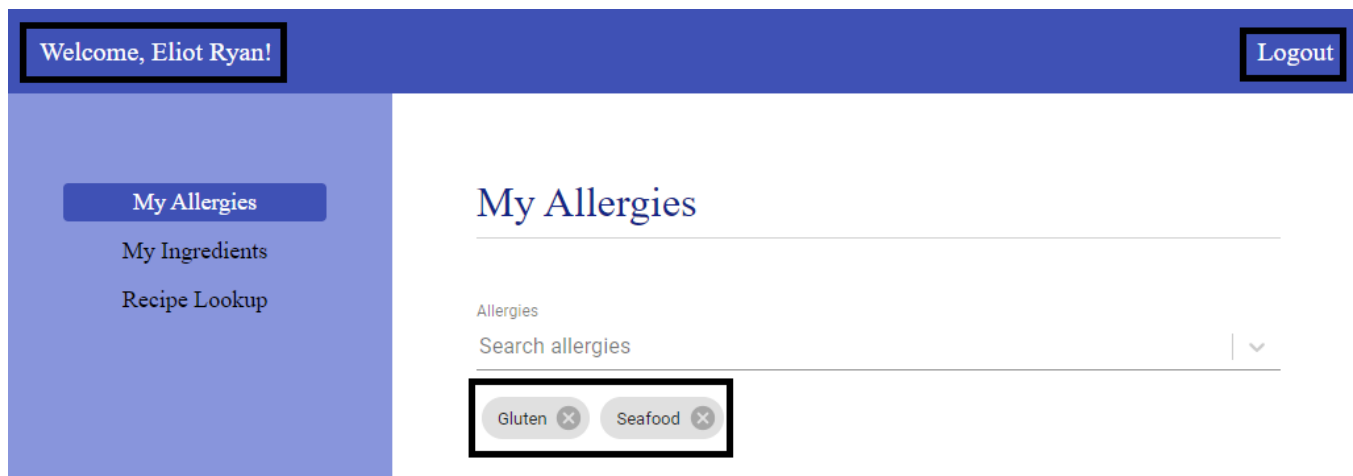
Search allergies

Seafood ✕ Gluten ✕

SIGN UP



2.3. Log out After signing into the application, you can see the user's name displayed on the left hand corner and the entered allergy conditions are saved on the bottom of the allergy search bar. Lastly, by clicking "LogOut" in the top right corner, it will sign you out.



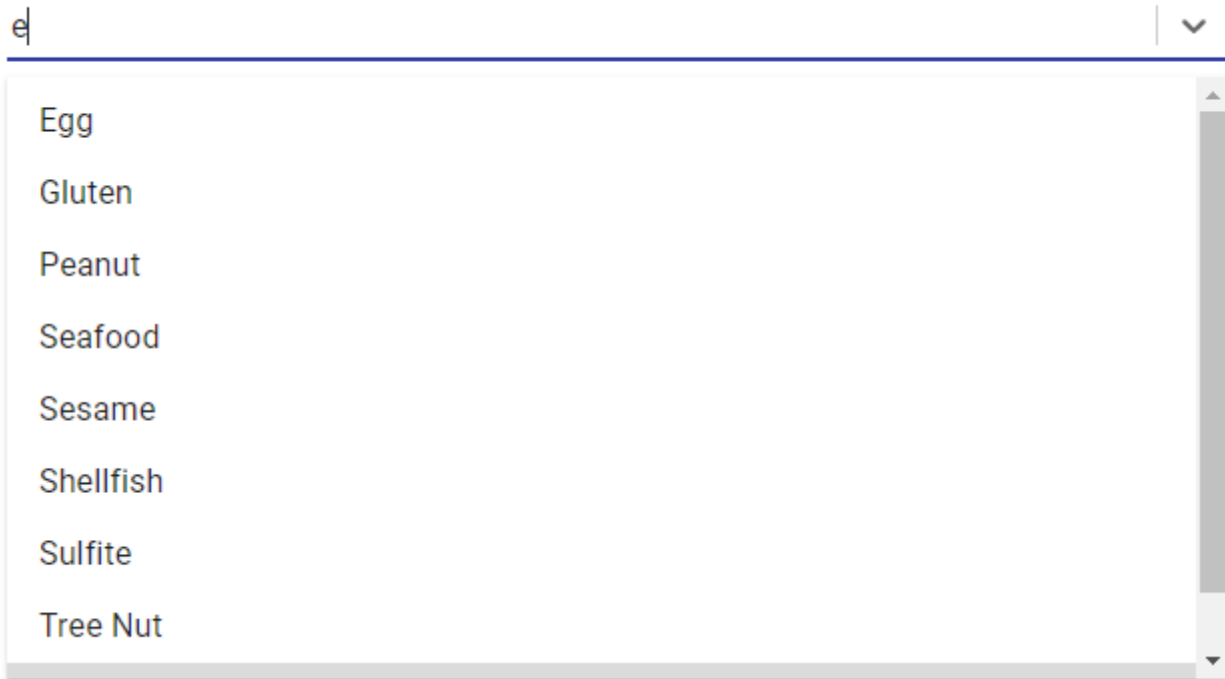
3. My Allergies

Through the left side bar, you will be able to find “My Allergies” page. This will show you the current allergies and allow you to edit them. The allergy information is also received via HAPI FHIR communication using FHIR patientID and maintained by user input. This allergy data will be preserved when you refresh the page.



* Search allergies: Typing a few characters of the allergen on the search bar will populate a list of matching items.

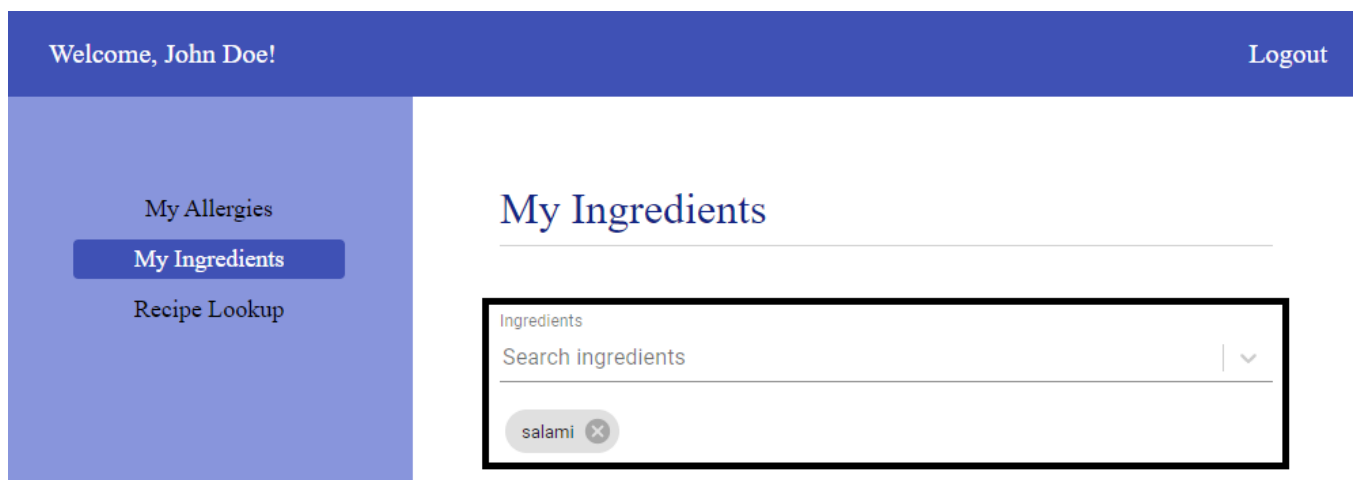
Allergies

A screenshot of a web application's 'Allergies' section. At the top, there is a search bar with the letter 'e' entered. Below the search bar is a dropdown menu with a list of allergens: Egg, Gluten, Peanut, Seafood, Sesame, Shellfish, Sulfite, and Tree Nut. The dropdown menu has a scrollbar on the right side. To the right of the dropdown menu, there is a small asterisk symbol (*).

Add allergies: By selecting the one from the list, the allergen will be registered on the bottom of the search bar. *
Delete allergies: Clicking X button next to the saved allergen will delete it from your allergies. *

4. My Ingredients

Similarly, using the left side bar you will be able to find “My Ingredients” page. This will show you previously added ingredients and allow you to edit them. The ingredient search results are pulled from Spoonacular API and the saved list of ingredients are maintained by user. In the same way with allergies, the ingredient data will be preserved when you refresh the page.

A screenshot of a web application's 'My Ingredients' page. The page has a dark blue header with the text 'Welcome, John Doe!' on the left and 'Logout' on the right. On the left side, there is a light blue sidebar with three menu items: 'My Allergies', 'My Ingredients' (which is highlighted with a dark blue background), and 'Recipe Lookup'. The main content area has a title 'My Ingredients' in a large, dark blue font. Below the title is a search bar with the text 'Search ingredients' and a dropdown arrow. Below the search bar, there is a list of ingredients, with 'salami' visible and a small 'X' button next to it to delete it. The search bar and the list of ingredients are enclosed in a black rectangular box.

- * Search ingredients: Typing a few characters of the ingredient on the search bar will populate a list of matching items.
- * Add ingredients: By selecting the one from the list, the ingredient will be registered on the bottom of the search bar.
- * Delete ingredient: Clicking X button next to the saved ingredient will delete the item from the saved list.

5. Recipe Look up

Lastly, on the left side bar, you will find the “Recipe Look up” menu. This page will take your allergies and ingredients as input and recommend the meal planning options. The recipes are received from Spoonacular API.

* Search recipes: You can search by type of “Cuisine” and “Recipe Keyword”. If you do not enter any information into these search fields, the query will return all categories. The user can also filter results by their “Allergies” and their “Ingredients”.

As shown in the example, a cuisine of American and keyword of pizza was used and resulted in three hits that matched those inputs without the “Allergy” or “Ingredient” filter set.

Welcome, John Doe!Logout

My Allergies

My Ingredients

Recipe Lookup

Recipe Lookup

Cuisine (Optional)
American

☐ Filter by my allergies☐ Filter by my ingredients


Recipe Keywords
Pizza

SEARCH

(Required)
Results

Sloppy Joe pizza breads

Ready In: 30 min.




VIEW RECIPE

VIEW NUTRITION

Cobb Salad Pizza

Ready In: 30 min.




VIEW RECIPE

VIEW NUTRITION

Salami Pizza with Garlic and Chili Oil

Ready In: 45 min.



VIEW RECIPE

VIEW NUTRITION

Once the allergy filter is enabled, two pizzas that contain seafood and eggs are removed from the results based on the users saved allergies in their profile.

6

Recipe Lookup

Cuisine

American



☒ Filter by my allergies

☐ Filter by my ingredients

Recipe Keywords

Pizza

SEARCH

Results

Sloppy Joe pizza breads

Ready In: 30 min.



VIEW RECIPE

VIEW NUTRITION

When the “Ingredients” filter is enabled, the pizza that contains salami is the only result based on the saved ingredients for the user profile.

Recipe Lookup

Cuisine

American



☐ Filter by my allergies

☒ Filter by my ingredients

Recipe Keywords

Pizza

SEARCH

Results

Salami Pizza with Garlic and Chili Oil

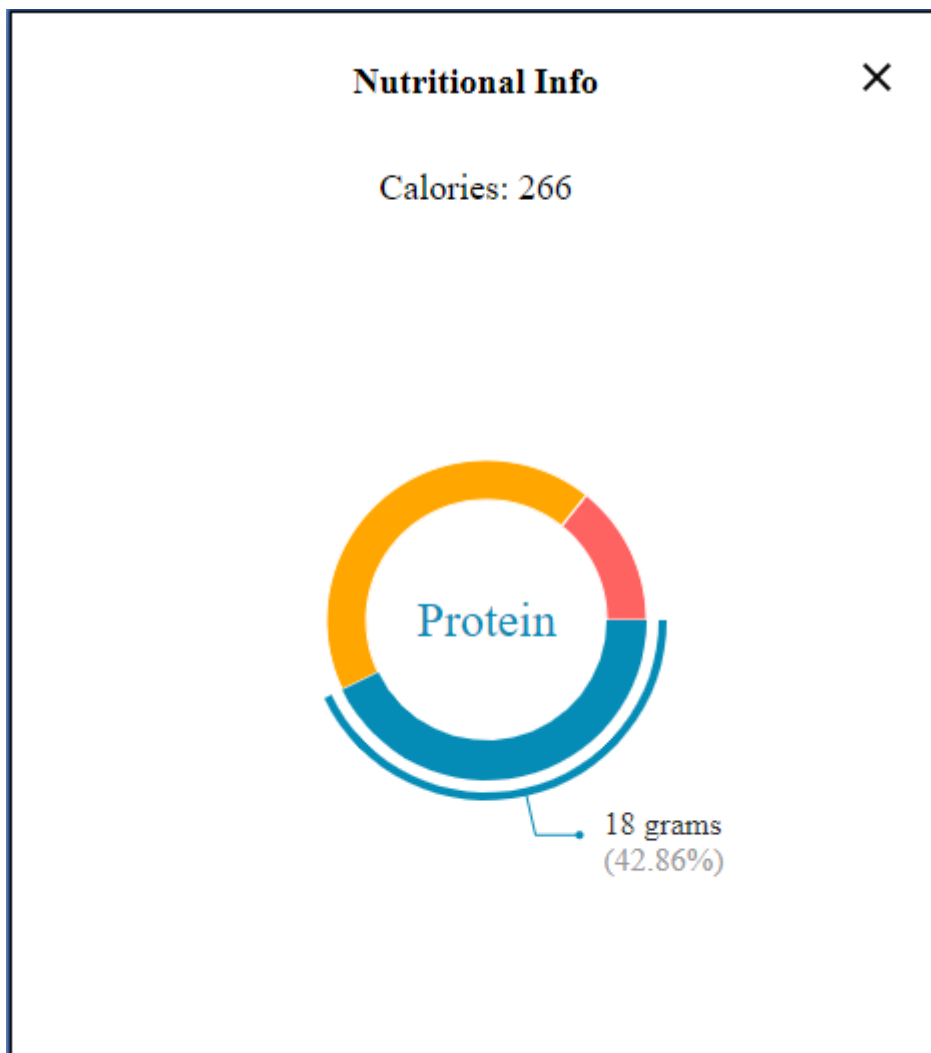
Ready In: 45 min.



VIEW RECIPE

VIEW NUTRITION

- View recipes: By clicking “View Recipe” button, it will direct you to a web page with the recipe detail.
- View Nutrition: By clicking “View Nutrition” button, the application populates a page with the total calories of the recipe and a pie chart of nutritional information.



6. Local Application Deployment

6.1. How To Run Local Build To deploy all three dockers (frontend, backend, database) download and install docker here. URL: <https://www.docker.com/> Next, open up terminal or PowerShell, and cd to FoodPlanner2 project folder and run

- build tagged services "docker-compose build"
- start all services "docker-compose up"
- stopping all services "docker-compose down"

All in one command: "docker-compose down; docker-compose build; docker-compose up;"

Frontend URL: <http://localhost:3000/> Backend URL: <http://localhost:8080/>

6.2. How To Access Local Database Database access, please download pgAdmin 4 and connect to localhost:5432 using username 'postgres' and no password.

7. Production Application Deployment

Please refer to section 2 of the Special Instructions document.

URL: <https://github.gatech.edu/gt-cs6440-hit-fall2019/FoodPlanner2/blob/master/Final%20Delivery/Special%20Instructions.pdf>

8. Checks To See If The Servers Are Alive

8.1. Production Servers Frontend Application URL:

`curl -I https://apps.hdap.gatech.edu/newfoodplanner2frontend/` Response:

```
HTTP/1.1 200 OK
Date: Sun, 24 Nov 2019 20:54:07 GMT
Server: envoy
content-type: text/html
content-length: 2611
last-modified: Wed, 20 Nov 2019 04:56:31 GMT
etag: "5dd4c77f-a33"
accept-ranges: bytes
x-envoy-upstream-service-time: 0
Access-Control-Allow-Origin: *
```

Backend Application URL:

`curl https://apps.hdap.gatech.edu/newfoodplanner2backend/` Response:

Hello Runtime Terror

Database URL:

`https://apps.hdap.gatech.edu/pgadmin/login?next=%2Fpgadmin%2Fbrowser%2F`

Database Name: foodplanner2database

Login: student@hdap.gatech.edu

Password: hdap2019!

Database Login: test

Password: test

8.2. Local Servers Frontend Application URL:

`curl -I http://localhost:3000/` Response:

```
HTTP/1.1 200 OK
Server: nginx/1.17.4
Date: Sun, 24 Nov 2019 21:14:06 GMT
Content-Type: text/html
Content-Length: 2611
Last-Modified: Fri, 22 Nov 2019 07:50:48 GMT
Connection: keep-alive
ETag: "5dd79358-a33"
Accept-Ranges: bytes
```

Backend Application URL: `curl http://localhost:8080/` Response:

Hello Runtime Terror

Database access, please download pgAdmin 4.

Database URL: localhost:5432

Database Login: postgres

No Password

8.3. Postman Script Please refer to section 1 of the Special Instructions document.

URL: `https://github.gatech.edu/gt-cs6440-hit-fall2019/FoodPlanner2/blob/master/Final%20Delivery/Special%20Instructions.pdf`

Script URL: `https://github.gatech.edu/gt-cs6440-hit-fall2019/FoodPlanner2/blob/master/FoodPlanner.postman_collection.json`

9. Version

- Front End
 - Node.js v10.16.3

- yarn v1.19.1
 - npm 6.12.0
- Docker 2.1.0.3
- Back End
 - Java 1.8
 - SpringBoot 2.1.6.RELEASE
 - Maven 3.6.2
 - HAPI FHIR 3.1.0
- Database
 - postgres 12.0