

SWE573 SOFTWARE DEVELOPMENT PRACTICE

FALL 2023

Project Final Deliverables

Instructor: Asst. Prof. Suzan Üsküdarlı

Ahmadreza Samiei

Project Name: Recipe

GitHub Repository: github.com/ahmadrezasam/SWE-573-Project

Git Tag Version: v0.9

Deployment URL: <http://shark-app-iq8vp.ondigitalocean.app/>

January 02, 2024

TABLE OF CONTENTS

TABLE OF CONTENTS	i
HONOR CODE	ii
Credentials to Login to the App.....	iii
Overview	iii
Software Requirements Specification	iii
Functional Requirements	iii
Mockups Design	v
Home page	v
Recipe page	vi
Registration Page.....	vii
Profile page	viii
Deployment.....	viii
System manual	viii
Dockerfile	viii
Docker-compose	ix
User manual	ix
Home Page:	ix
Recipe Details Page:	ix
User Registration:.....	x
User Profile:.....	x
Add Recipe Page:.....	x
Profile Page Table:.....	x
Food Allergens Warning:	x
Test results	x
Unit Test	x
User test	xi
Video link presentation	xi

HONOR CODE

Related to the submission of all the project deliverables for the Swe573 2023 Fall semester project reported in this report, I Ahmadreza declare that:

- I am a student in the Software Engineering MS program at Bogazici University and am registered for Swe573 course during the Fall 2023 semester.
- All the material that I am submitting related to my project (including but not limited to the project repository, the final project report, and supplementary documents) have been exclusively prepared by myself.
- I have prepared this material individually without the assistance of anyone else with the exception of permitted peer assistance which I have explicitly disclosed in this report.

Ahmadreza Samiei

Credentials to Login to the App

Although the user can register for the app with their own credentials, one user example's credentials are added below.

Email: ahmadreza@email.com

Password: r674wBc7^%%P5qMb

Overview

This project has been developed for the SWE 573 (Software Development Practice) course and focuses on a recipe application. The web application allows users to view a list of recipes, along with detailed information such as title, description, ingredients, instructions, nutrition facts (calculated using the FoodData Central API), preparation and cooking time, comments, and average ratings for each recipe. Registered users have the ability to create recipes, interact with other recipes by rating, commenting, and bookmarking. These interactions can be viewed, edited, or deleted on the user's profile page. Additionally, users can manage their personal information on the profile page, select food allergens from a provided list, and receive warnings on recipe pages containing those allergens.

Software Requirements Specification

Functional Requirements

1. User Registration:

1. Users shall create an account with a unique email and password
 1. Password shall be minimum 5 characters
 2. Password shall include at least 1 number
2. User shall specify personal information
 1. User shall enter name and last name
 2. User must choose gender as either male or female
 3. User must enter height and weight
 1. System shall allow user to enter height in cm
 2. System shall allow user to enter weight in kg
 4. User shall choose country of residence
 5. User must choose dietary preference as either omnivore or vegetarian or vegan
 6. User shall choose their diet goals as either weight loss or weight gain or healthy food

2. User login:

1. Users shall login with their email and password
2. Users shall recover their password

3. User Profile:

1. User shall manage personal information from profile settings
 1. User shall have the ability to change their password

2. User shall edit height and weight
 1. System shall calculate and display user's BMI
3. User shall edit country of residence
4. User shall edit dietary preference as either omnivore or vegetarian or vegan
5. **User shall upload profile image**
6. User shall change their diet goals
7. Users shall provide descriptions about themselves
8. Users shall choose food allergens from allergens list
2. User shall manage activities from profile account
 1. Users shall view/edit/delete their saved/bookmarked recipes in their profile activities
 2. Users shall view/edit/delete their ratings for recipes in their profile activities
 3. Users shall view/edit/delete their comments for recipes in their profile activities
 4. **Users shall view/edit/delete their questions for recipes in their profile activities**
 5. Users shall view/edit/delete their recipes in their profile activities
3. User shall create menu from their profile account
 1. Users shall add their recipes
 1. Users must provide titles for their recipes
 2. Users must provide required ingredients and their amounts for their recipes
 3. Users must provide required time to preparation and cook for their recipes
 1. System shall select time in (hh:mm:ss) format. **Times are in minutes**
 4. Users shall select meal types including breakfast, lunch, dinner, dessert, snack **(dessert, snack not included)**
 5. Users shall upload images for their recipes
 6. **Users shall upload videos for their recipes**
 7. **Users shall provide difficulty level**
 2. Users shall edit and delete their recipes
 3. Users shall see others' the reviews/suggestions for their recipes
 4. Users shall see others' ratings' average for their recipes
 5. **Users shall see others' questions for their recipes and answer their questions**
 6. Users shall see their menus list
 7. Users shall edit and delete their menus list
4. **Recipes list (Home page)**
 1. User shall view recipes' list
 2. User shall view recipes' ratings and total time
 1. System shall display total time in minutes and hours
 3. User shall search for recipes by ingredients, dietary restrictions, cooking time, and difficulty level **(title is added) (dietary restrictions, cooking time, and difficulty level not completed)**
5. **Recipe details**

1. User shall view recipe details including ingredients, descriptions, instructions, photos, videos, nutrition facts, preparation time and cooking time
2. User shall view recipe rating, reviews, creator of recipe
3. Recipe interaction while signed in
 1. User shall rate and bookmark the recipe
 1. The system shall display rating out of 0 to 5 with 0.5 increment **completed (with 1 increment)**
 2. User shall leave review and/or ask questions about the recipe
 3. User shall be informed about food allergen if selected allergens from user profile
 4. **User shall add an alternative recipe for the current food recipe**
 5. **Users shall view top 3 high-rated alternative recipes**
 6. **Users shall see other recipe recommendations recommended by the system.**
6. **Admin Profile: Django default admin panel is used for this purpose**
 1. **Admin shall ban any user who violates the website's terms of service**
 2. **Admin shall delete any comment/question/answer that violates the website's terms of service**

Mockups Design


Home page

LOGO
 [Item1](#)
[Item2](#)
[Item3](#)
[Item4](#)

Log in


Register

Search
 Filters




Category
Food Title

Total Time
 Rating



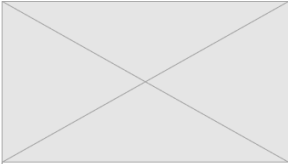
Category
Food Title

Total Time
 Rating



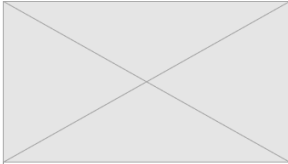
Category
Food Title

Total Time
 Rating



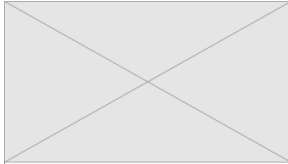
Category
Food Title

Total Time
 Rating



Category
Food Title

Total Time
 Rating



Category
Food Title

Total Time
 Rating

Recipe page

LOGO

Item1

Item2

Item3

Item4

Log in

Register

Title

Food description

Cooking time

Preparation time

Total time

Ingredients:

Amount - Item 1

Amount - Item 2

Amount - Item 3

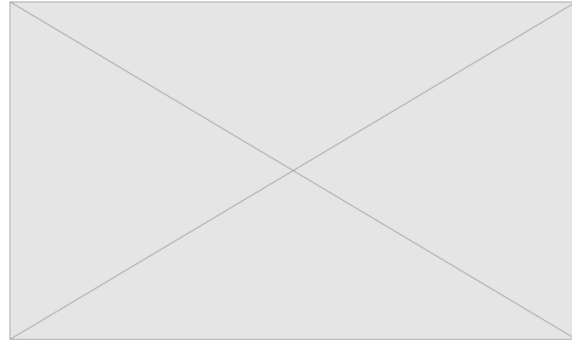
Amount - Item 4

Preparation Steps:

Step 1

Step 2

Step 3



★ Out of 5

Bookmark

Rating

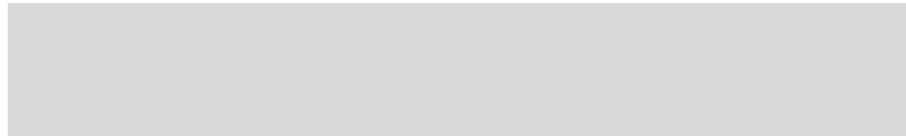
Recommendations



Add alternative recipe

Add

Nutrition Facts:



Reviews

Q/A



Registration Page

LOGO

[Log in](#)**Sign up for a free account***Required

*Gender:

Male ☐Female ☐*Dietary Preference: omnivore ☐ vegetarian ☐ vegan ☐Diet Goals: lose weight ☐ gain weight ☐ healthy food ☐[Register](#)

Profile page

LOGO

Sign Out

▼ Name Surname

User Information

First name

Last name

*Height(cm)

*Weight(kg)

BMI

Description

*Gender: Male ☐ Female ☐

Country of Residence

*Dietary Preference: omnivore ☐ vegetarian ☐ vegan ☐

Diet Goals: lose weight ☐ gain weight ☐ healthy food ☐

Upload Image: Upload Image

Change Password

New Password

Food Allergens

Food Allergens List

Save

Bookmarks - Ratings - Comments - Q/A

Recipe 1: Food Title - Description - user created - User's Interaction

Edit

Delete

Deployment

The project is deployed on DigitalOcean, a cloud hosting service. The deployment URL is shark-app-ig8vp.ondigitalocean.app. Docker is utilized for deployment, and the Dockerfile settings are accessible on the GitHub repository.

System manual

Dockerfile

A Dockerfile is used to build a Docker image. It contains a set of instructions that Docker follows to create an image. These instructions define the base image, set up the environment, copy files into the image, install dependencies, and configure the application within the image.

You run the docker build command, pointing it to the directory where your Dockerfile is located. This command creates a Docker image based on the instructions in the Dockerfile. The image is recipeApp itself without any additional packages including database.

docker build -t <image-name> .

-t: This flag is used to tag the image with a name.

``.``: The ``.`` refers to the current directory

Docker-compose

`docker-compose.yml` is used to define a multi-container Docker application. It allows you to define services, networks, and volumes in a declarative way. It simplifies the process of running multiple containers and their configurations. For my project, the file includes a Docker Compose configuration for a Django application along with a PostgreSQL database. Unlike Dockerfile `docker-compose` initialize the database with the application

Version: Specifies the version of the Docker Compose file format.

Services:

db: Defines a PostgreSQL service using the `postgres:15` image. It sets up environment variables for the PostgreSQL database, in port 5432, and mounts a volume (`pg_data`) for data storage.

web: Defines a service for your Django application. It builds the image from the current directory (`.`), sets the container name to `'recipeApp'`, exposes port 8000, mounts the current directory as a volume, and sets environment variables for Django and database configuration. It also depends on the `my-postgres` service.

Volumes:

pg_data: Defines a named volume for storing PostgreSQL data.

Commands:

docker-compose build Builds or rebuilds the Docker images.

docker-compose up Starts the services

docker-compose down Stops and removes the containers, networks, and volumes

User manual

Home Page:

When a user enters the recipe web application, he/she can see a list of recipe cards. Each card contains an image of the food, title, description, creator, rating, and total cooking time. Users can utilize the search functionality on the home page to find recipes by title or ingredients separated by commas.

Recipe Details Page:

Upon selecting a recipe card, users are directed to the recipe details page. Here, they can find information such as the recipe title, creation time, description, ingredients (with amounts and names),

instructions, preparation and cooking time, average rating, and nutrition facts calculated using the FoodData Central API. A comment section allows users to view comments from others.

User Registration:

Unregistered users cannot interact with recipes. To do so, users can navigate to the navbar and register by providing a unique email address, a unique username, and a password. After registration, users can log in using their email and password.

User Profile:

Once logged in, users can access their profile by clicking the profile button in the navbar. In this section, users can view and manage their information. They can add, edit, or delete information, excluding email and username. The profile card on the left also allows users to add a new recipe or change their password. Moreover, they user can logout using “logout” button.

Add Recipe Page:

Clicking on "Add Recipe" button on the profile page directs the user to the recipe creation page. Here, users can input the title, category, description, ingredients, steps, preparation and cooking time, and an image for their recipe. All fields on this page are required.

Profile Page Table:

On the profile page, users can find tables listing their recipes, comments, ratings, and bookmarks. They have the option to delete or modify entries from the table.

Food Allergens Warning:

In the food allergens section of the profile page, users can select allergens from a provided list. If any allergens are chosen, a warning will be displayed to the logged-in user on recipe pages containing the selected allergens.

Test results

Unit Test

File Location: The unit tests are available in the "tests.py" file within the GitHub repository.

Tested Models: Unit tests have been conducted on the following models:

- User Profile
- Recipe
- User Interaction

Views Testing - Recipe: The following views related to recipes have been tested:

- Home view
- Recipe view
- Add Recipe view
- Edit Recipe view
- Delete Recipe view

Views Testing - Profile: The profile views have been thoroughly tested, including:

- Profile view
- Password Change view
- Register view
- Login view
- Logout view

User Interaction Views: User interaction views on both profile and recipe pages have been tested, covering:

- User Comments
- User Bookmarks
- User Ratings

User test

As demonstrated during lecture hours, the project possesses the main functionalities required for the program to execute. Minor bugs may be present, but they do not impact the core functionality of the project.

Video link presentation

<https://drive.google.com/file/d/1Cgmy6GZVqh2de8EK9NZlvqcWtRYAlaRB/view?usp=sharing>