BE Capstone Project

Ahmed Amin - Egypt

Title Slide

Title: Recipe Management API

Subtitle: API for managing and organizing recipes

Author: Ahmed Amin

Date: October 2024

Logo:

Introduction

- **What is Recipe Management API ?**
  - An API for managing recipes, ingredients and categories.
  - Provides endpoints for creating, reading, updating and deleting recipes.
- **Purpose:**
  - To make recipe management efficient for developers and users alike.

Project Goals

- **Goals:**
  - Create a user-friendly API for managing recipes.
  - Enable searching and filtering by titles, categories, ingredients, preparation time, cooking time and servings.
  - Provide secure user authentication for recipe management.

Key Features

CRUD operations for recipes:

User can perform Create, Read, Update and Delete operations on recipes.

This allows for easy management and maintenance of recipes.

User Authentication:

The API includes user authentication to ensure that only authorized users can manage recipes.

Search & Filter:

Filter recipes by ingredients, categories and titles

Technologies Used

Backend: Django, Django REST Framework

Database: MySQL

Authentication: Token-based Authentication (DRF tokens) and

JWT (JSON Web Tokens)

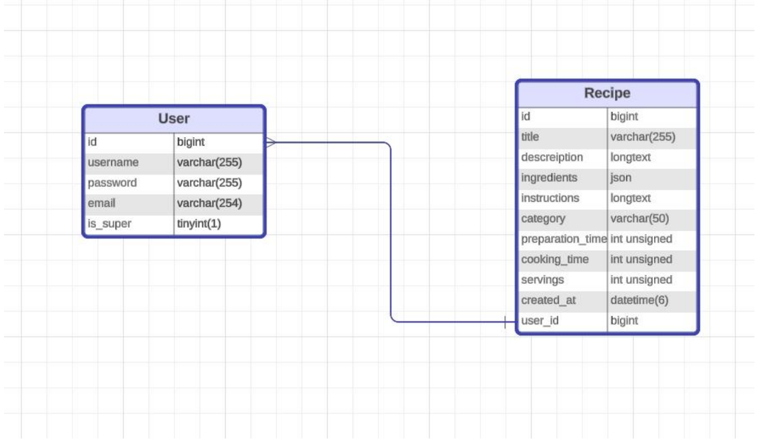
Tools: VS Code, Postman, Git, Github

Hosting: Deployed on pythonanywhere,

Data Models

- Recipe Model:
  - title [varchar(255)]
  - description [longtext]
  - ingredients [json]
  - instructions [longtext]
  - category [varchar(50)]
  - preparation_time [int unsigned]
  - cooking_time [int unsigned]
  - servings [int unsigned]
  - created_at [datetime(6)]
  - user [foreignkey => User]

- User Model:
  - email [varchar(255)]
  - username [varchar(254)]
  - password [varchar(255)]



User	
id	bigint
username	varchar(255)
password	varchar(255)
email	varchar(254)
is_super	tinyint(1)

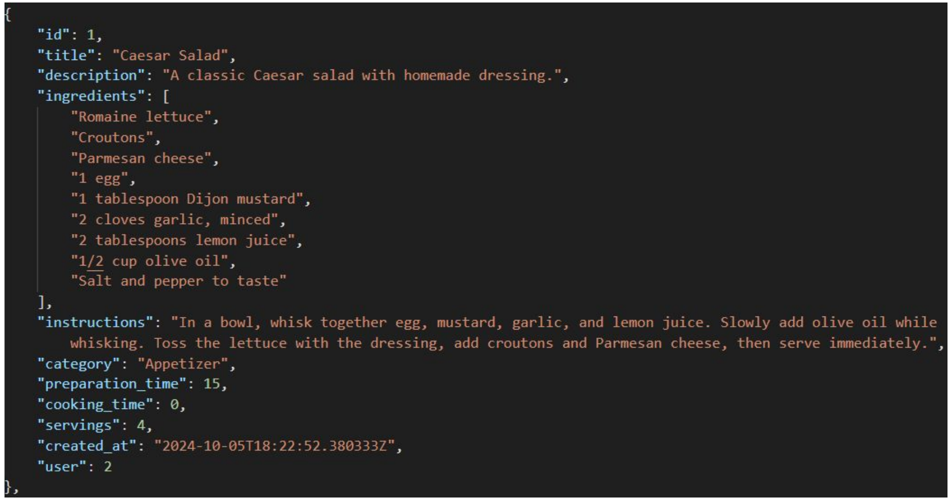
Recipe	
id	bigint
title	varchar(255)
descreiption	longtext
ingredients	json
instructions	longtext
category	varchar(50)
preparation_time	int unsigned
cooking_time	int unsigned
servings	int unsigned
created_at	datetime(6)
user_id	bigint

API Endpoints

- `GET /api/recipes/` - display all recipes in the database (only admin can add, delete or update on it)
- `GET /api/recipes/{id}` - display specific recipes in the database (only admin can add, delete or update on it)
- `GET /api/users/` - display all users (only admin can use it)
- `POST /register/` - register new user
- `POST /logout/` - logout from the current user's account
- `GET /user/` - display the current user's profile
- `GET /user/recipes/` - display the current user's recipes
- `POST /user/recipes/` - add new recipe in the current user's recipes
- `GET /user/recipes/{id}/` - display specific recipe of the user's recipes
- `PUT /user/recipes/{id}/` - update specific recipe of the user's recipes
- `DELETE /user/recipes/{id}/` - delete specific recipe of the user's recipes

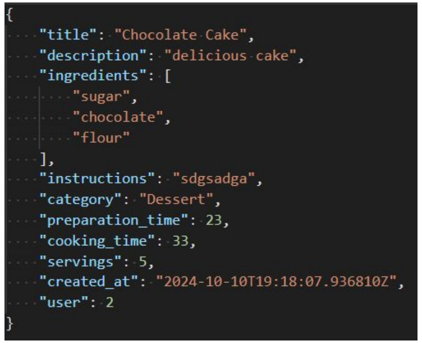
Key API Calls

- List Recipe:
  - **GET**/user/recipes/



```
{
    "id": 1,
    "title": "Caesar Salad",
    "description": "A classic Caesar salad with homemade dressing.",
    "ingredients": [
        "Romaine lettuce",
        "Croutons",
        "Parmesan cheese",
        "1 egg",
        "1 tablespoon Dijon mustard",
        "2 cloves garlic, minced",
        "2 tablespoons lemon juice",
        "1/2 cup olive oil",
        "Salt and pepper to taste"
    ],
    "instructions": "In a bowl, whisk together egg, mustard, garlic, and lemon juice. Slowly add olive oil while
        whisking. Toss the lettuce with the dressing, add croutons and Parmesan cheese, then serve immediately.",
    "category": "Appetizer",
    "preparation_time": 15,
    "cooking_time": 0,
    "servings": 4,
    "created_at": "2024-10-05T18:22:52.380333Z",
    "user": 2
},
```

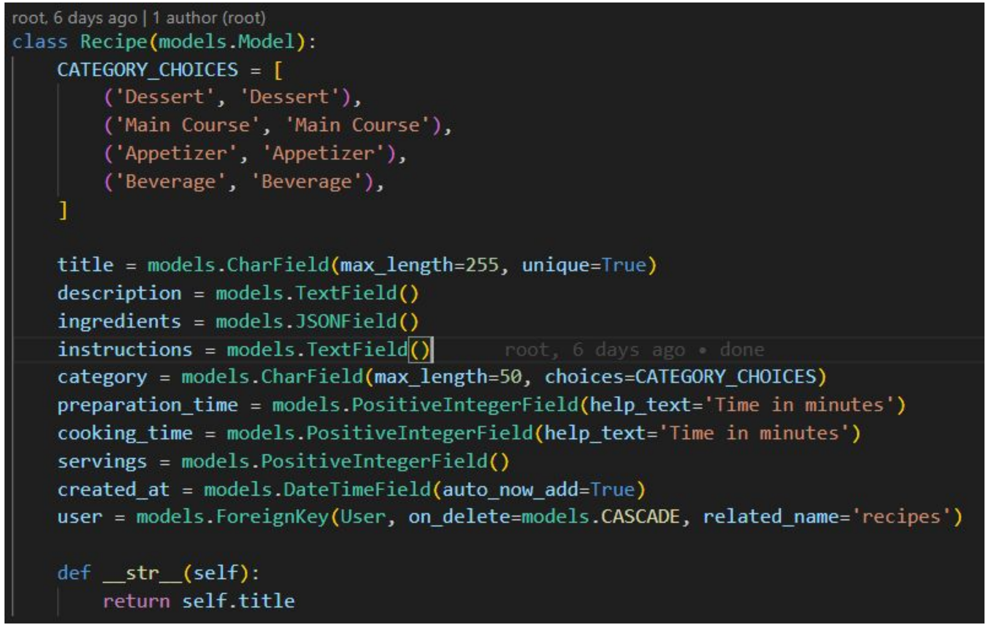
- Create a Recipe:
  - **POST** /user/recipes/



```
{
    "title": "Chocolate Cake",
    "description": "delicious cake",
    "ingredients": [
        "sugar",
        "chocolate",
        "flour"
    ],
    "instructions": "sdgsadga",
    "category": "Dessert",
    "preparation_time": 23,
    "cooking_time": 33,
    "servings": 5,
    "created_at": "2024-10-10T19:18:07.936810Z",
    "user": 2
}
```

Code Walkthrough

- Models
    - Code Snippet (Recipe Model):



```
root, 6 days ago | 1 author (root)
class Recipe(models.Model):
    CATEGORY_CHOICES = [
        ('Dessert', 'Dessert'),
        ('Main Course', 'Main Course'),
        ('Appetizer', 'Appetizer'),
        ('Beverage', 'Beverage'),
    ]

    title = models.CharField(max_length=255, unique=True)
    description = models.TextField()
    ingredients = models.JSONField()
    instructions = models.TextField()       root, 6 days ago • done
    category = models.CharField(max_length=50, choices=CATEGORY_CHOICES)
    preparation_time = models.PositiveIntegerField(help_text='Time in minutes')
    cooking_time = models.PositiveIntegerField(help_text='Time in minutes')
    servings = models.PositiveIntegerField()
    created_at = models.DateTimeField(auto_now_add=True)
    user = models.ForeignKey(User, on_delete=models.CASCADE, related_name='recipes')

    def __str__(self):
        return self.title
```

❖ Explanation:

➢ Each recipe has title, description, ingredients, instructions, category, preparation time, cooking time, servings and relationship with users.

➢ The `ForeignKey` links recipes with users.

Error Handling & Validation

- Error Codes:
  - **400 Bad Request:** Invalid input or missing fields.
  - **401 Unauthorized:** User is not authenticated.
  - **404 Not Found:** Resource does not exist.

- Validation:
  - username, email must be unique.
  - recipe's title must be unique.
  - preparation_time, cooking_time and servings in recipe must be positive integer.
  - validations to check if the user in cookies token didn't logout.

Deployment

The Recipe management API can be deployed to platforms PythonAnywhere. Additional settings such as ALLOWED_HOSTS and DEBUG can be configured based on the deployment platform and project requirements.

Conclusion

- The Recipe Management API simplifies the management of recipes and users.
- It is designed for developers building recipe-related applications or websites.

Q&A

- Thank You!
- Contact Information:
  - Email: programmer82253@gmail.com
  - GitHub Repo: Estoda/Recipe_Management_API: the capstone project for alx (github.com)