**RECIPE APP**

**GROUP MEMBERS**

* **FAHAMA FAROOQ**
* **ASIFA FAROOQ**
* **FAIZA ASGHAR**

**LEADER**

* **FAHAMA FAROOQ**

**WHAT IS RECIPE APP?**

A recipe app is a software application designed to help users discover, organize, and follow cooking recipes. Users can typically search for recipes based on various criteria such as ingredients, cuisine type, dietary preferences, or cooking time. They can also save their favorite recipes, create shopping lists based on recipe ingredients, and sometimes even track their cooking progress.

**WHY WE USE THIS APP?**

Here's why people might use recipe apps and a real-life example:

1. Convenience: Recipe apps provide a convenient way to access a wide variety of recipes from different sources without the need for physical cookbooks. Users can have all their favorite recipes accessible on their smartphones or computers.

2.Meal Planning: Users can plan their meals in advance by browsing through recipes, saving them for later, and organizing them into meal plans for the week. This helps in ensuring a balanced and varied diet.

3. Shopping List: Many recipe apps offer a feature to generate shopping lists based on the ingredients required for selected recipes. This makes grocery shopping more efficient and helps users avoid missing ingredients while cooking.

4. Cooking Inspiration: Recipe apps can be a source of inspiration for users who want to try new dishes or explore different cuisines. They often include user-generated content, reviews, and ratings, which can help users discover popular and highly-rated recipes.

**Real-life example:**

Suppose someone wants to explore plant-based recipes to incorporate more vegetables into their diet. They can use a recipe app like "Yummly" or "Allrecipes" to search for vegetarian or vegan recipes, save their favorites, create a meal plan for the week, and generate a shopping list based on the ingredients needed for those recipes. This helps them streamline their cooking process and stay organized while trying out new plant-based dishes.

**CODE**

import tkinter as tk

from tkinter import messagebox, scrolledtext

import os

class RecipeAppGUI:

    def \_\_init\_\_(self, master):

        self.master = master

        self.master.title("Recipe App")

        self.recipes = {}

        self.note\_filename = "recipe\_notes.txt"

        self.create\_widgets()

    def create\_widgets(self):

        self.master.configure(bg='light green')

        # Recipe Name

        self.recipe\_name\_label = tk.Label(self.master, text="Recipe Name:", font=("Times New Roman", 12), bg='light grey')

        self.recipe\_name\_label.grid(row=0, column=0, sticky="e")

        self.recipe\_name\_entry = tk.Entry(self.master, font=("Times New Roman", 12))

        self.recipe\_name\_entry.grid(row=0, column=1, padx=5, pady=5)

        # Ingredients

        self.ingredients\_label = tk.Label(self.master, text="Ingredients:", font=("Times New Roman", 14), bg='light grey')

        self.ingredients\_label.grid(row=1, column=0, sticky="e")

        self.ingredients\_entry = tk.Entry(self.master, font=("Times New Roman", 12))

        self.ingredients\_entry.grid(row=1, column=1, padx=5, pady=5)

        # Instructions

        self.instructions\_label = tk.Label(self.master, text="Instructions:", font=("Times New Roman", 14), bg='light grey')

        self.instructions\_label.grid(row=2, column=0, sticky="e")

        self.instructions\_entry = tk.Text(self.master, height=5, width=30, font=("Times New Roman", 12))

        self.instructions\_entry.grid(row=2, column=1, padx=5, pady=5)

        # Buttons

        self.add\_button = tk.Button(self.master, text="Add Recipe", command=self.add\_recipe, font=("Arial", 12), bg="green", fg="white")

        self.add\_button.grid(row=3, column=0, columnspan=2, pady=10)

        self.view\_button = tk.Button(self.master, text="View Recipe", command=self.view\_recipe, font=("Arial", 12), bg="green", fg="white")

        self.view\_button.grid(row=4, column=0, columnspan=2, pady=5)

        self.search\_button = tk.Button(self.master, text="Search Recipe", command=self.search\_recipe, font=("Arial", 12), bg="green", fg="white")

        self.search\_button.grid(row=5, column=0, columnspan=2, pady=5)

        self.notebook\_button = tk.Button(self.master, text="Open Notebook", command=self.open\_notebook, font=("Arial", 12), bg="green", fg="white")

        self.notebook\_button.grid(row=6, column=0, columnspan=2, pady=5)

    def add\_recipe(self):

        name = self.recipe\_name\_entry.get()

        ingredients = self.ingredients\_entry.get()

        instructions = self.instructions\_entry.get("1.0", tk.END)

        self.recipes[name] = {'ingredients': ingredients, 'instructions': instructions}

        messagebox.showinfo("Success", "Recipe added successfully!")

        # Clear the input fields after adding the recipe

        self.recipe\_name\_entry.delete(0, tk.END)

        self.ingredients\_entry.delete(0, tk.END)

        self.instructions\_entry.delete("1.0", tk.END)

    def view\_recipe(self):

        view\_window = tk.Toplevel(self.master)

        view\_window.title("All Recipes")

        if self.recipes:

            text\_area = scrolledtext.ScrolledText(view\_window, width=40, height=10, wrap=tk.WORD, font=("Arial", 12))

            text\_area.grid(row=0, column=0, padx=10, pady=10)

            for name, recipe in self.recipes.items():

                text\_area.insert(tk.END, f"Recipe: {name}\nIngredients: {recipe['ingredients']}\nInstructions: {recipe['instructions']}\n\n")

        else:

            no\_recipes\_label = tk.Label(view\_window, text="No recipes available.", font=("Arial", 12))

            no\_recipes\_label.grid(row=0, column=0, padx=10, pady=10)

    def search\_recipe(self):

        keyword = self.recipe\_name\_entry.get()

        found\_recipes = []

        for name, recipe in self.recipes.items():

            if keyword.lower() in name.lower():

                found\_recipes.append(name)

        if found\_recipes:

            messagebox.showinfo("Search Result", f"Found {len(found\_recipes)} recipes containing '{keyword}':\n{', '.join(found\_recipes)}")

        else:

            messagebox.showerror("Error", "No recipes found containing the keyword.")

    def open\_notebook(self):

        notebook\_window = tk.Toplevel(self.master)

        notebook\_window.title("Recipe Notebook")

        text\_area = scrolledtext.ScrolledText(notebook\_window, width=40, height=10, wrap=tk.WORD, font=("Arial", 12))

        text\_area.grid(row=0, column=0, padx=10, pady=10)

        # Load notes from file if it exists

        if os.path.exists(self.note\_filename):

            with open(self.note\_filename, 'r') as f:

                notes = f.read()

                text\_area.insert(tk.END, notes)

        # Save note to file function

        def save\_note():

            with open(self.note\_filename, 'w') as f:

                notes = text\_area.get("1.0", tk.END)

                f.write(notes)

            messagebox.showinfo("Success", "Note saved successfully!")

        # Button to save note

        save\_button = tk.Button(notebook\_window, text="Save Note", command=save\_note, font=("Arial", 12), bg="green", fg="white")

        save\_button.grid(row=1, column=0, pady=5)

def main():

    root = tk.Tk()

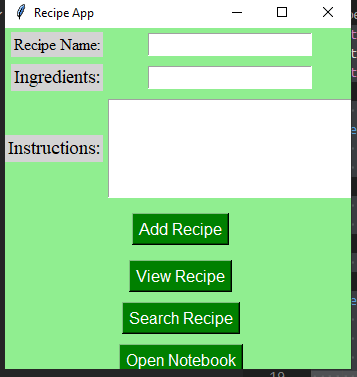
    app = RecipeAppGUI(root)

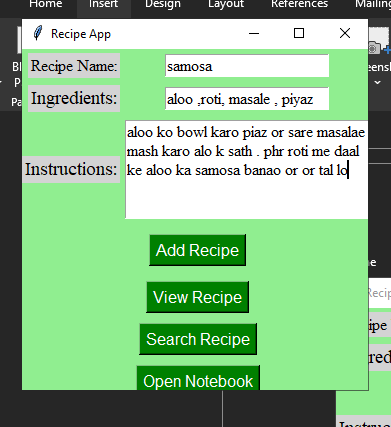
    root.mainloop()

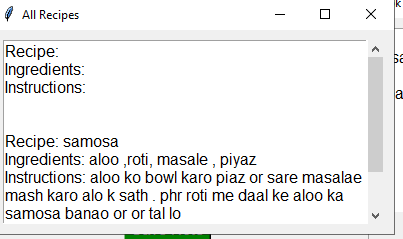
if \_\_name\_\_ == "\_\_main\_\_":

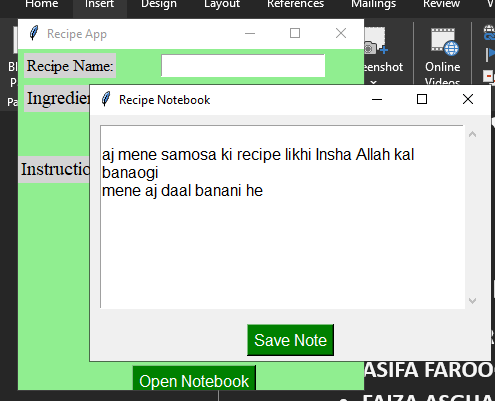
    main()

**OUTPUT:**









**REVIEW About Class:**

**Classes offered by "Bano Qabil 2.0" with Miss Sumbul were beneficial in enhancing my Python skills from a beginner's level. I would recommend these classes to anyone looking to learn Python in a structured and engaging manner, especially for those starting with the basics.** **Every learning environment is beneficial. I gained a lot of knowledge .**