SmartCook - AI-Powered Recipe Generator

Politechnika Poznańska Project Report

Birgül Yüksel ER1922

June 2025

Ingredients

Project Overview

SmartCook is a Python-based AI-powered recipe generator designed to create personalized cooking recipes using natural language generation. Users input available ingredients and choose a desired tone (style) and language (English or Turkish), and the system generates a complete recipe including preparation details, estimated calories, and even a health tip. The project enhances user experience with features such as text-to-speech and clipboard copy.

Objectives

- To create a fun and intelligent assistant that transforms ingredients into unique recipes.
- To offer a bilingual (English/Turkish) interface for broader accessibility.
- To experiment with OpenAI's language models for natural-sounding, creative output.
- To incorporate text-to-speech for enhanced interactivity and accessibility.

Technologies Used

Library	Purpose
openai	For accessing GPT models to generate text.
pyttsx3	Converts recipe text into speech.
pyperclip	Enables clipboard operations.
random	Used for generating surprise ingredients.
builtins	Standard input/output functions for interaction

Key Features

🗣️ Multilingual Support

- Supports **English and Turkish** interfaces via MESSAGES dictionaries.
- User selects the language at the beginning of the session.

🎭 Recipe Style Personalization

- Users choose one of four recipe styles:
    1. **Fun** – Conversational and cheerful.
    2. **Professional** – Technical and precise.
    3. **Grandma** – Warm and nostalgic.
    4. **Blogger** – Story-rich and casual.

🍳 Smart Recipe Generation

- Based on GPT-3.5 Turbo Instruct.
- Includes title, ingredients, prep/cook time, calorie estimate, and health tip.

🔊 Text-to-Speech (TTS)

- Uses pyttsx3 to read the recipe aloud.
- Detects available voices depending on system and language.

📋 Clipboard Integration

- Option to copy the entire recipe for later use via pyperclip.

Workflow Diagram

[User Input] → [Language Selection] → [Style Selection]

↓

[Ingredient Input] → [Prompt Building]

↓

[OpenAI Model Response] → [Display Recipe]

↓

[Text-to-Speech (optional)] → [Clipboard Copy (optional)]

Installation & Requirements

To run **SmartCook**, ensure you have the following Python environment and libraries installed:

Requirements

- Python 3.8 or above
- pip (Python package manager)

Required Libraries

“ *pip install openai pyttsx3 pyperclip* ”

Additional Notes

For voice output (TTS), system TTS engines must be available:

- Windows: SAPI5 voices such as "Microsoft David" or "Microsoft Tolga"
- macOS: Built-in voices like "Samantha", "Yelda"
- Linux: May require espeak or festival, compatibility not guaranteed

You need a valid OpenAI API key:

- Sign up at https://platform.openai.com/
- Copy your API key and insert it into the script:

  “ *openai.api_key = "your-api-key-here"* ”

How to Run

1. Clone or download the script to your local machine.
2. Ensure your Python environment is active and dependencies are installed.
3. Set your OpenAI API key in the openai.api_key field.
4. Run the program via terminal or command prompt: “ *python generator.py* ”

How to Use

Once the program starts:

1. **Choose Language**: Enter en (English) or tr (Turkish).
2. **Select Style**: Choose 1–4 for different recipe personalities:
    - 1: Fun
    - 2: Professional
    - 3: Grandma
    - 4: Blogger
3. **Enter Ingredients**: Provide a comma-separated list of ingredients like: “ *chicken, tomato, rice* ” Or type *surprise* to get a random set.
4. **View Recipe**: Wait while your recipe is generated.
5. **Extras**:
    - Choose to have the recipe **read aloud**.
    - Choose to **copy** it to your clipboard.

Unit Testing

To ensure the correctness of the recipe generation logic, a unit test was added for the generate_recipe() function.

How to Run the Test

1. Make sure generator.py and test_smartcook.py are in the same directory.
2. Run the following command in the terminal:

    “ *python -m unittest test_smartcook.py* ”

What It Tests

- That the generate_recipe() function returns a string.
- That it responds correctly to both custom ingredients and "surprise" mode.
- That it handles OpenAI integration using mocked responses, without sending real API requests.

This test confirms the core functionality works as expected and satisfies the project’s unit testing requirement.

Challenges Faced

- **Voice Matching per Language**: Matching correct TTS voices for Turkish on all OS platforms required conditional handling.
- **Prompt Engineering**: Achieving stylistic tone consistency in GPT responses demanded fine-tuned instructions.
- **Clipboard Encoding**: Some systems required manual encoding adjustments for Turkish characters.

Future Improvements

- GUI version using Tkinter or PyQt for broader user base.
- Add more languages (e.g., French, Spanish).
- Package as a mobile app with voice interaction.
- Local AI fallback (e.g., fine-tuned LLM with llama.cpp).
- Ingredient recognition from uploaded photos using image classification.

Conclusion

SmartCook combines the power of GPT-based language models with user-friendly features to make AI-generated recipe creation delightful, educational, and accessible. It showcases how natural language interfaces can enrich everyday experiences like cooking, even for beginners or those with limited ingredients.