

# My food buddy - Foodie

By Sonia Wadhvani

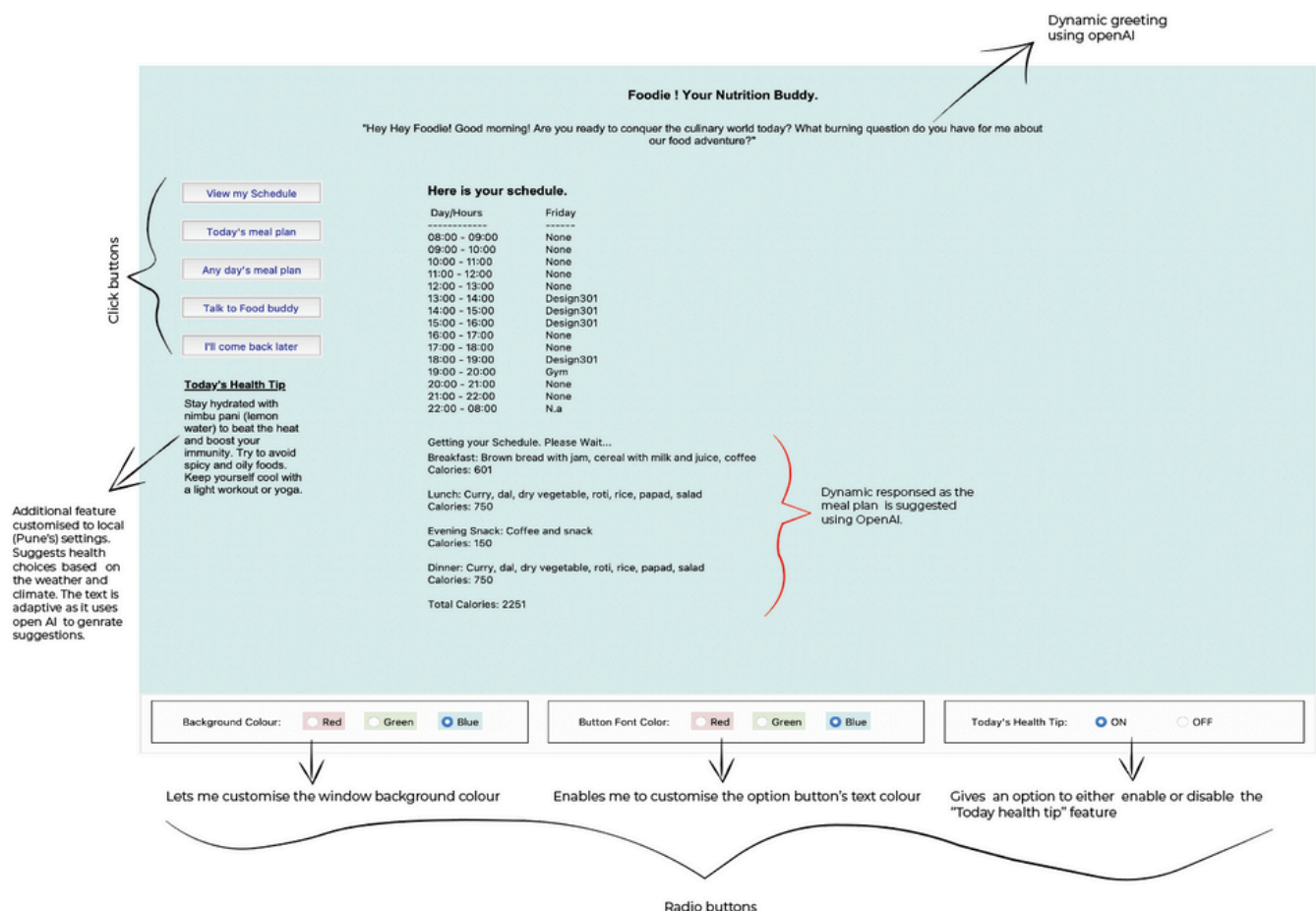
## Problem Statement

The daily schedule at FLAME University varies for each student, leading to missed meals on days with morning classes, lunch conflicts, or delayed dinners due to sports or exercise activities.

This may cause health issues.

## Program Solution

The program uses standard development practices and universally popular third party API libraries (openai, tkinter etc.). The data source is a static schedule which feeds in the students schedule in form of a static array. There are several menu options to View week or day's schedule or talk to "My chatbot- Foodie". The data (schedule and menu options) is passed to AI to fetch meal preference based on schedule. The meal options are customised to the calorie needs of the individual and tuned to lecture times. The UI window is self-adjusted to size of the monitor, the data is formatted and displayed on the UI. Additional features include location based health tips and customization of look & feel of the UI in terms of colours and content.



## Logic Used

The program uses standard development practices and universally popular third party API libraries. (openai, tkinter etc.). The data source is a static schedule which feeds in the students schedule in form of a static array. There are several menu options to View week or day's schedule or talk to "My chatbot- Foodie". The data (schedule and menu options) is passed to AI to fetch meal preference based on schedule. The meal options are customised to the calorie needs of the individual and tuned to lecture times. The UI window is self-adjusted to size of the monitor, the data is formatted and displayed on the UI.

Additional features include location based health tips and customization of look and feel of the UI in terms of colours. Some content can be shown and hidden. Dynamic greeting and tip presented to the user . The chatbot is restricted to food related conversation. The Ui window picks up the size of the monitor dynamically.

Technically, the flow starts from with the menu being displayed, based on the data from schedule being read, appropriate days menu is read and sent to OpenAI for meal preference . The response is shown back to user. The buttons are made visible or invisible based on choice . The program is neatly written into proper methods and commented for quick referencing .



## Problems faced

Challenges faced were interesting, the first one was to move from console to a UI, tkinter library helped fix it. Second the content was never same size so the new content used to show the residual content so to refresh the content was a challenge however enabling and disabling window children helped me achieve that .

Yet another was a using the fixed position for window elements created problems when the different Monitor size was used or window was resized . To fix this is used relative positioning by identifying screen size and appropriately sizing my window . Also putting restriction on window resizing helped.

Another challenge was the dark and light mode of the device . The looks and feel wasn't consistent and caused UI to look shabby which I fixed by giving a default colour to all the window elements .

Overall it was a great leaning experience helping me to learn to use third party APIs , built complex logic and thinking outside the boundaries.