Assignment 3

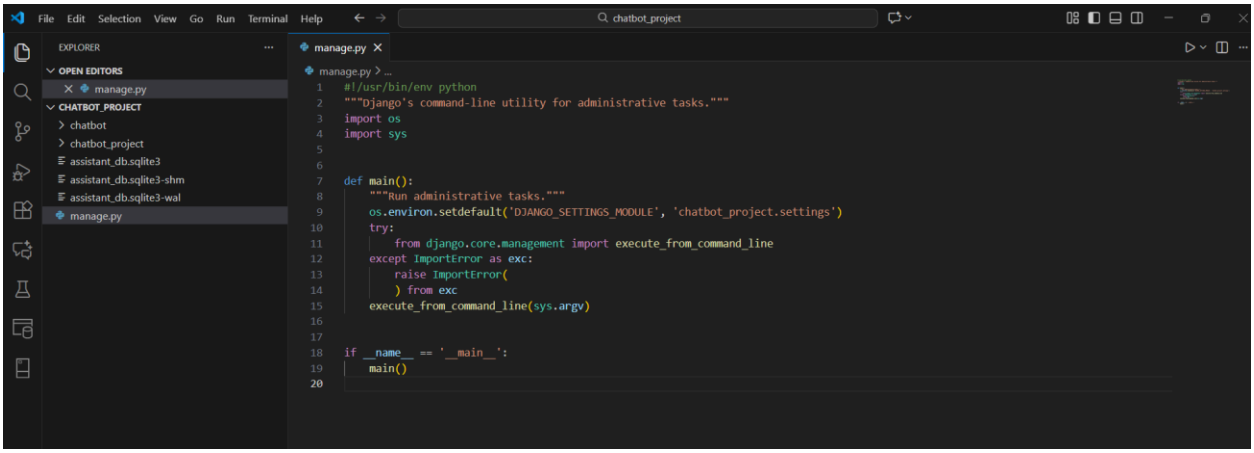
Prashanna Acharya

Advance Artificial Intelligence (MSCS-633-B01)

University of the Cumberlands

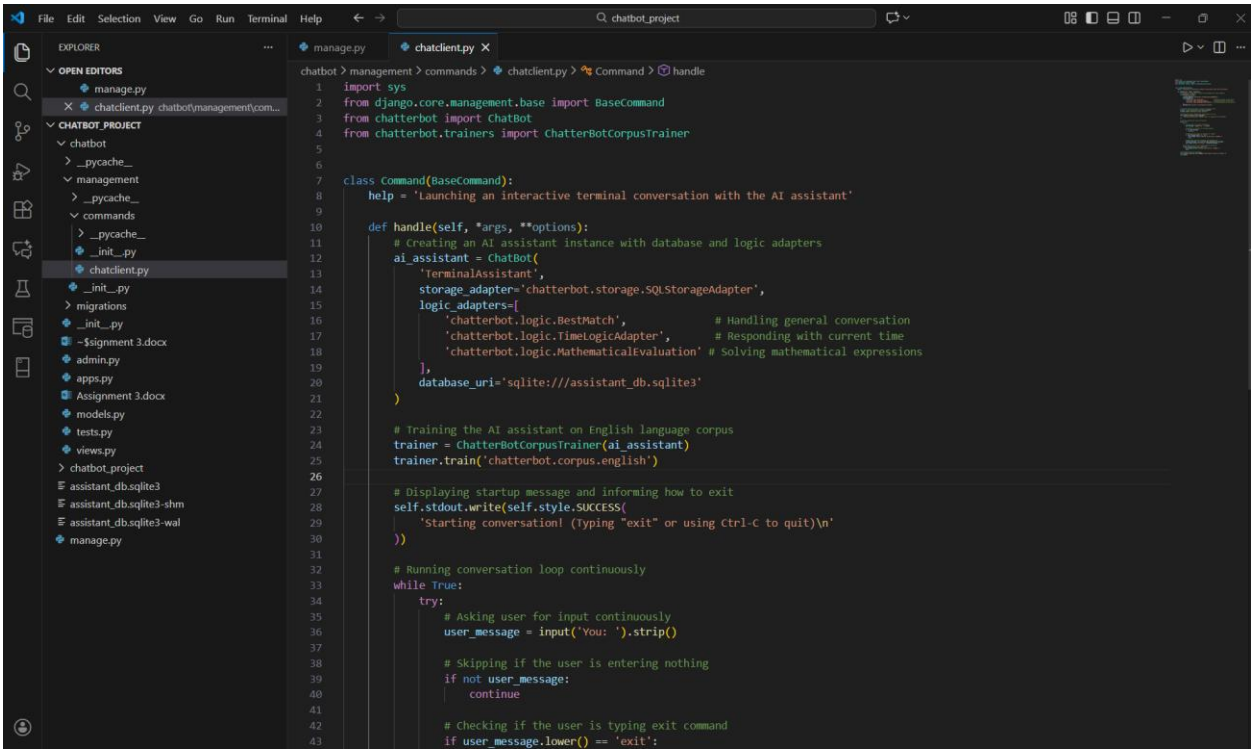
Dr. Ulrich Vouama

manage.py



The manage.py file is a script that Django automatically generates when a new project is created. Its role is to provide a command-line tool for controlling and working with the Django application. When executed, it sets the appropriate environment variable, so Django knows which settings configuration to load. After that, it brings in Django's command-handling machinery and passes control to it. This setup enables to run various project-related commands, such as launching the development server, managing database migrations, creating apps, or executing custom commands. Essentially, manage.py acts as the main gateway for performing administrative operations in a Django project and ensures everything is properly configured before those tasks run.

chatclient.py



The chatclient.py file is implemented as a Django management command that is intended to operate a ChatterBot-driven chatbot within the terminal. A Command class is defined by extending Django's BaseCommand, allowing the script to be detected and executed through the command python manage.py chatclient. Inside the command, a ChatBot instance is configured with a storage adapter, multiple logic adapters, and an SQLite database in which conversational data is stored. Training is then carried out using ChatterBot's English corpus so that a basic level of conversational ability is provided. Once training has been completed, an interactive loop is initiated, through which user input is continuously received and responses from the chatbot are generated in real time. The session is maintained until the user types "exit" or interrupts the program. Through this design, the Django project is enabled to function as a terminal-based chatbot system controlled by a custom management command.

Output on Terminal