

Project Title: Study Buddy Chatbot

Project Summary:

This project introduces students to Artificial Intelligence with basic machine learning concepts through a fun and interactive study assistance. Students will build a python based chatbot that connects to an AI API and learns from the user over time. The chatbot tracks study sessions, recognizes subjects like math or science, and detects simple emotions (positive or negative words). Using this data it adapts its feedback and motivation to support each student's learning habits.

Students gain hands-on experience with core AI ideas such as data driven personalization, user interaction, and improving responses based on past behavior, all while creating a helpful tool that makes studying more enjoyable.

Target Audience

- **Grade range:** 5-12
- **Skill level:** Intermediate
- **Tools/Tech:** Python (with Tkinter or Streamlit) for the interface and integrates and AI (OpenAI/Hugging Face/**Mistral AI**)
- Note API costs money ex: 30 students for one month-2 cad

Learning Objectives

- Understand how AI responds to user input - describe how chat bot uses AI model and prompts to generate conversational responses.
- Modify and extend Python code - edit and customize a pre-built chatbot program to change behaviours, visuals, and personality.
- Design and customize a user interface - build a simple chat window including buttons, input boxes, and theme elements.
- Implement basic data logging - track study sessions, timestamps, simple storage
- Apply prompt engineering techniques - adjust chatbots instructions to improve tone, clarity, and motivational language.
- Explore introductory machine learning concepts- use simple rule based personalization to help chatbot learn from data, study time, mood, subject keywords and improve responses over time.

Project Outcomes / Deliverables

- A working study buddy chatbot application
- A chatbot that logs study time and adapts feedback
- A chatbot that recognizes mood and subjects
- A theme UI customization by each student
- A short explanation of how their bot learns
- Saved data showing the bot improving over time

Core Features (MVP)

- Chat interface: input box, send button, conversation window
- AI powered responses
- Personalized greeting and session start command
- Study session timer/logger
- Basic machine learning personalization
- Personalized greetings and feedback
- Clear chat/reset button

Session Breakdown / Timeline

- **Session 1:** Introduction to AI and ML - demo study buddy, plan theme/personality under restrictions (code template)
- **Session 2:** Backend Code-Understand how python calls the AI API and returns text
- **Session 3:** UI design and Customization-build chat window, buttons, theme
- **Session 4:** Customizing UI+AI+Timer- Send input to AI, log study time, display response
- **Session 5:** Machine learning personalization
- **Session 6:** Catch up session/ testing+Showcase

Risks & Mitigation Strategies

- **Risk:** Students find API confusing
Mitigation: Provide simplified pre-built API wrappers
- **Risk:** Algorithm complexity may confuse some students
Mitigation: Simplify explanation, include diagrams
- **Risk:** Uneven coding speeds in group
Mitigation: Pair fast learners to help peers, give extra stretch tasks

- **Risk:** UI layout errors or handling bugs
Mitigation: Provide a working UI template and checkpoint files. Mentors demonstrate each change step by step to prevent indentation or layout issues

GitHub Structure

- Week 1 - Week 6 Github resources
- Create a Github
- Technical Plan - code wise
- Developer timeline
- Streamlit
- Choose what API we need for study buddy project
- Create github

Make a github

Finalize what we're gonna use Streamlit, AI API

Github: SidMcStarter

Henry Tang or Proff to TA class