AI Customized Learning Recommendation Chatbot

PROJECT POLICY DEFINITION

Version: 2025/01

Status: Draft

Issued: 2025/08/17

# 1. Policy Statement

This project aims to develop an AI-powered educational tutor chatbot using Python, Streamlit, and LangChain for a hackathon. The chatbot will present a small set of predefined problems, provide guided explanations, allow users to attempt quizzes, and generate feedback with recommendations tailored to the difficulty level of the question. The project will not record user progress or store data, ensuring privacy and lightweight implementation.   
The policy serves as a guiding framework before development begins.

# 2. Purpose and Scope

Purpose: To create an interactive, AI-driven learning tool that demonstrates innovation, usability, and educational value within hackathon constraints.  
  
Scope Includes:  
- Streamlit-based chatbot interface (minimal but functional).  
- Predefined learning problems (MVP: 1 unit with 5 problems).  
- Quizzes with auto-evaluated feedback.  
- Recommendations tailored to learner level and difficulty.  
- AI-generated feedback via LangChain prompt templates.  
  
Scope Excludes:  
- User accounts or authentication.  
- Persistent data storage (no database).  
- Progress tracking across sessions.

# 3. Principles

- Simplicity: Keep the system lightweight and hackathon-feasible.  
- Accessibility: Easy to use in any browser.  
- Transparency: Clearly state limitations (no history, no tracking).  
- Educational Value: Provide meaningful feedback and recommendations.  
- Conversation Flow: Greeting → Ask level → Recommend quiz → Provide feedback.

# 4. Outcomes

- A working chatbot demo accessible via web.  
- One learning unit with five problems and quizzes.  
- Feedback with AI-generated recommendations based on difficulty.  
- Minimal, functional interface for hackathon judging.

# 5. Technology Stack

- Python (core development language)  
- Streamlit (UI and web app framework)  
- LangChain (AI logic, prompt handling, and feedback generation)

# 6. Roles and Responsibilities

- STEAM & ED/IT Team:  
 • Design learning scenarios and conversation flows.  
 • Create quiz and feedback content.  
  
- AI & CSE Team:  
 • Implement chatbot interface and UI using Streamlit.  
 • Connect LangChain for logic and recommendations.  
 • Ensure rule-based branching for quiz recommendations.

# 7. Risk Management

- Time Constraint → Focus on minimum viable product (1 unit, 5 problems).  
- AI Inaccuracy → Use curated content + rule-based branching.  
- Expectation Gap → Disclaimers about no progress saving.  
- Deployment Risk → Use Streamlit Cloud / simple hosting for reliability.

# 8. Implementation Approach

- Phase 1: Define 1 unit with 5 problems.  
- Phase 2: Build chatbot UI (Streamlit).  
- Phase 3: Implement conversation flow (greeting → ask level → recommend quiz → feedback).  
- Phase 4: Integrate LangChain for explanations and recommendations.  
- Phase 5: Test MVP and deploy demo.

# 9. Q&A Clarifications

- Database use? Not required for MVP; no user record saving.  
- Feedback type? AI-generated feedback using LangChain is acceptable.

# 10. Closure

At hackathon end, the project will be closed by:  
- Delivering a live demo.  
- Documenting features, risks, and lessons learned.  
- Suggesting future improvements (e.g., database, progress tracking, more content units).