Scholar AI Assistant — User Guide

1. Introduction

The Scholar AI Assistant is a conversational chatbot designed to help users discover academic resources from the CSUSB Library. It uses AI-powered search and filtering to return credible, scholarly, and peer-reviewed materials based on user queries.

Scope: Helps users find scholarly materials through an interactive Streamlit interface.

Audience: Students, researchers, and faculty using CSUSB library tools.

Requirements: Python 3.10+, Streamlit, Groq API Key, requests, urllib3, pydantic.

2. Installation and Setup

To install and run the application, follow one of the two methods below.

Option 1 — Docker Setup:

- Clone the repository: git clone
- Build Docker image: docker build -t scholar-ai .
- Run container: docker run -p 8501:8501 -e GROQ_API_KEY= scholar-ai
- Access via browser: http://localhost:8501

Option 2 — Local Setup:

- Clone repository: git clone
- Install dependencies: pip install -r requirements.txt
- Set API Key: export GROQ_API_KEY='your_api_key'
- Run app: streamlit run app.py
- Access via: http://localhost:8501

3. Application Overview

The application has three layers — Frontend (Streamlit UI), Core Logic (services and handlers), and External APIs (Groq LLM and CSUSB Primo).

4. User Interface Overview

- Header: Displays app title and description.
- Chat Area: Input field for user queries using natural language.
- Sidebar: Contains filters for peer-reviewed content, resource type, and date range.
- Results Table: Displays organized search results with details.

5. Features and Functionalities

- Conversational AI — Understands natural language queries.

- Peer-review filtering — Retrieves credible scholarly content.
- Date & Type filters — Narrow search by publication date or resource type.
- Dynamic display — Renders results in structured tables.
- Session Management — Maintains conversation context.

6. Example Use Case

Example: A student searches for peer-reviewed articles on climate change. 1. Enter query: “Find peer-reviewed articles on climate change.” 2. Enable Peer-reviewed filter in the sidebar. 3. View results displayed in the results section. 4. Start new search to reset session.

7. Troubleshooting Guide

- Missing API Key → Set GROQ_API_KEY before running the app.
- App not loading → Ensure Streamlit or Docker is running properly.
- No results found → Refine query or check internet connection.
- Missing modules → Reinstall dependencies with pip install -r requirements.txt.

8. Best Practices for Users

- Use specific, concise research queries.
- Apply filters for accurate results.
- Check API key validity before running.
- Restart session if search results seem inconsistent.

9. Credits

Developed by: Scholar AI Assistant Team Institution: California State University, San Bernardino (CSUSB) Tools: Streamlit, Python, Groq API, CSUSB Primo API Version: 1.0.0