

Project Proposal: NUML InsightBot

1. Project Overview

NUML InsightBot is an intelligent, secure, and user-friendly chatbot designed specifically for the National University of Modern Languages (NUML). It leverages advanced AI to answer user queries by scraping and processing information from the NUML official website. The application features user authentication, protected routes, and a responsive chatbot interface with streaming responses. The system automatically refreshes its knowledge base every two months to ensure information remains current.

2. Technical Architecture

Frontend (React with React Router DOM)

- Pages:**
 - Login:** User authentication.
 - Signup:** User registration.
 - Home:** Overview of the chatbot's capabilities.
 - About:** Details about the project and team.
 - Contact:** Contact information and form.
 - Services:** Description of services offered.
 - Support:** Protected route hosting the chatbot interface.
- Protected Routes:** The `/support` route requires authentication.
- Streaming Responses:** Real-time, ChatGPT-like response generation.

Backend (FastAPI)

- RESTful API** for handling authentication and chatbot queries.
- Web Scraping** via Firecrawl.
- Vector Database** (Pinecone) for storing embeddings.
- AI Models:**
 - Open-source models (Ollama/Groq) for response generation.
 - Gemini embeddings for text processing.

Database (PostgreSQL)

- User Management:** Stores user credentials and profiles.
- Chat History:** Maintains conversation history with memory.
- Scraping Logs:** Tracks scraping activities and timestamps.

Additional Features

- Automated Scraping:** Scheduled every two months.
- Environment Variables:** Secure configuration via `.env`.
- User State Management:** Persistent chat sessions.

3. Page Content and Design

Login Page

- Content:** Login form (email, password), "Forgot Password" link, and redirect to Signup.
- Design:** Clean and minimalist with NUML branding.

Signup Page

- Content:** Registration form (name, email, password, confirm password).
- Design:** Consistent with Login page.

Home Page

- Content:** Introduction to NUML InsightBot, key features, and call-to-action for authentication.
- Design:** Engaging with visuals and concise text.

About Page

- Content:** Project description, mission, and team information.
- Design:** Professional and informative.

Contact Page

- **Content:** Contact form, email, phone number, and social media links.
- **Design:** User-friendly form with validation.

Services Page

- **Content:** List of services provided by NUML InsightBot.
- **Design:** Card-based layout.

Support Page (Protected)

- **Content:** Chatbot interface with message history and input box.
- **Design:** Similar to ChatGPT, with real-time streaming.

4. Database Schemas

User Schema

```
class User(BaseModel):
    id: int = Field(primary_key=True)
    name: str
    email: str = Field(unique=True)
    password: str
    created_at: datetime
    last_login: datetime
```

Chat History Schema

```
class ChatHistory(BaseModel):
    id: int = Field(primary_key=True)
    user_id: int = ForeignKey("users.id")
    query: str
    response: str
    timestamp: datetime
```

Scraping Log Schema

```
class ScrapingLog(BaseModel):
    id: int = Field(primary_key=True)
    timestamp: datetime
    status: str # Success/Failure
    pages_scraped: int
```

5. File Structure

```
numl-insightbot/
├── frontend/
│   ├── public/
│   ├── src/
│   │   ├── components/
│   │   ├── pages/
│   │   │   ├── Login.js
│   │   │   ├── Signup.js
│   │   │   ├── Home.js
│   │   │   ├── About.js
│   │   │   ├── Contact.js
│   │   │   ├── Services.js
│   │   │   └── Support.js
│   │   ├── App.js
│   │   └── index.js
│   └── package.json
├── backend/
│   ├── app/
│   │   ├── api/
│   │   │   ├── endpoints/
│   │   │   └── auth.py
│   │   ├── core/
│   │   ├── models/
│   │   ├── services/
│   │   └── main.py
│   ├── requirements.txt
│   └── Dockerfile
├── database/
│   └── migrations/
└── .env
```

6. API Routes

- **POST /api/register:** User registration.
- **POST /api/login:** User authentication.
- **GET /api/content:** Retrieve website content.
- **POST /api/chat:** Handle chatbot queries.
- **GET /api/scrape-status:** Check scraping status.
- **POST /api/admin/scrape:** Initiate manual scrape (admin).

7. Implementation Plan

Milestone 1: Project Setup (Days 1-2)

- Initialize React and FastAPI projects.
- Configure PostgreSQL database.
- Set up environment variables.
- Create basic routing structure.

Milestone 2: Authentication System (Days 3-4)

- Implement user registration and login.
- Create protected routes.
- Set up session management.

Milestone 3: Web Scraping Module (Days 5-6)

- Implement Firecrawl integration.
- Design data processing pipeline.
- Set up initial scraping of NUML website.

Milestone 4: AI Integration (Days 7-9)

- Configure Ollama/Groq API connections.

- Implement Gemini embeddings.
- Set up Pinecone vector store.
- Create chatbot query handling system.

Milestone 5: Chatbot Interface (Days 10-12)

- Build React chat components.
- Implement streaming responses.
- Add chat history and memory.
- Create support page with protected access.

Milestone 6: Additional Pages & Polish (Days 13-14)

- Complete Home, About, Contact, and Services pages.
- Implement responsive design.
- Conduct testing and bug fixes.

Milestone 7: Deployment & Documentation (Day 15)

- Deploy application.
- Create user documentation.
- Final testing and validation.

8. Potential Premium Features

1. **Advanced Query Handling:** Priority processing for paid users.
2. **Extended Chat History:** Longer conversation memory.
3. **PDF Export:** Ability to download conversations.
4. **Customized Alerts:** Notifications for specific content updates.
5. **API Access:** Direct access to the chatbot API.

9. Risks and Mitigation

- **Website Structure Changes:** Regular monitoring and adaptive scraping.
- **Model Performance:** Multiple fallback options for AI services.
- **Authentication Security:** Regular security audits and token rotation.
- **Data Freshness:** Strict adherence to the 2-month scraping schedule.

10. Conclusion

NUML InsightBot will serve as a valuable resource for the NUML community, providing instant access to accurate and up-to-date information. The 15-day timeline is aggressive but achievable with focused effort. The architecture ensures scalability, security, and maintainability.
