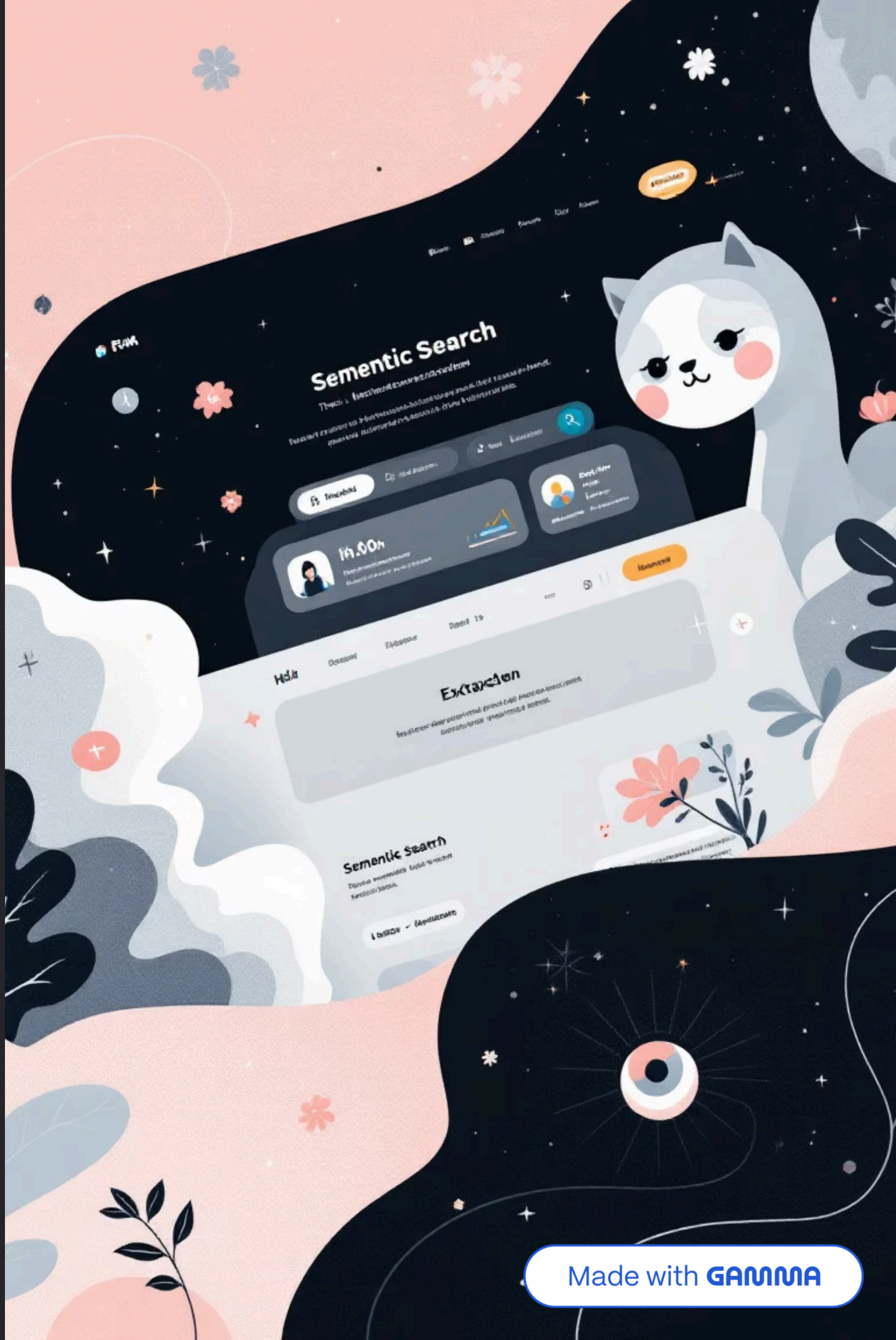
📊 Slide Deck Content (5 slides)

Objective: Build an SPA that extracts HTML from a given website and allows semantic search.

Approach:

- React frontend for input & results.
- FastAPI backend for HTML parsing & search.
- Vector DB (FAISS + Weaviate) for semantic similarity.



Frontend Design

Built with React (Vite)

Modern development setup with fast build times and hot module replacement for efficient development workflow.

Simple form: URL + Search Query

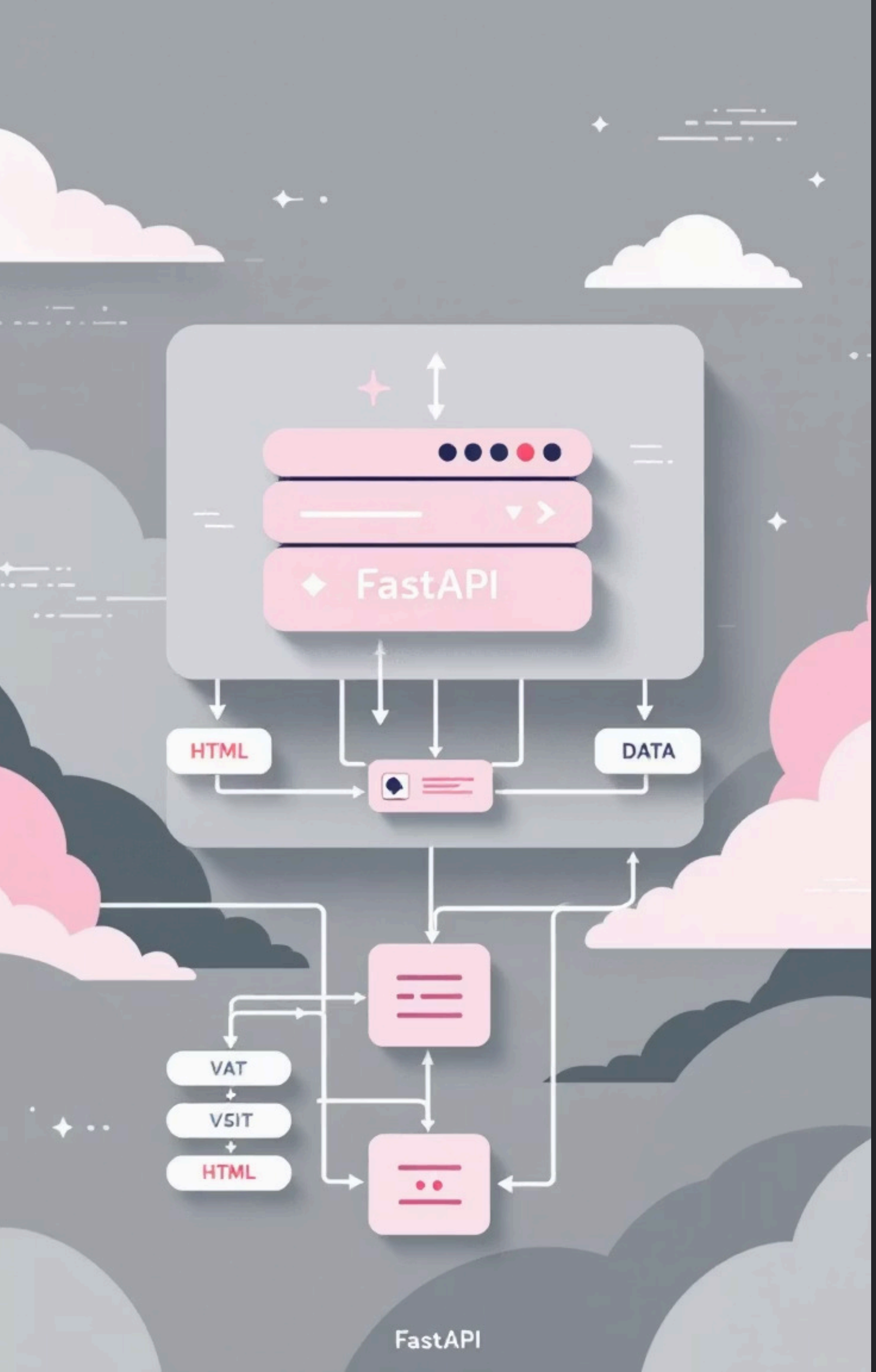
Clean, intuitive interface that accepts website URLs and search terms from users with minimal complexity.

Calls FastAPI backend with Axios

Reliable HTTP client integration for seamless communication between frontend and backend services.

Displays top 10 results in styled cards with Euclidian Distance

Professional presentation of search results with visual similarity scores for user understanding.



Backend Logic

FastAPI endpoints: /searchFAISS and /searchWeaviate.

Steps:

01

Fetch & clean HTML (BeautifulSoup)

Extract and sanitize HTML content from target websites using robust parsing libraries.

02

Tokenize into 500-token chunks

Break down content into manageable segments for efficient processing and search operations.

03

Generate embeddings (all-MiniLM-L6-v2)

Convert text chunks into high-dimensional vectors using state-of-the-art language models.

Outputs: JSON list of top 10 matches.

Vector Database

FAISS: Local, lightweight, fast in-memory search

Perfect for development and small-scale deployments with minimal setup requirements.

Weaviate: Docker-based, persistent vector search with REST API

Enterprise-ready solution with advanced features and scalable architecture.

Both tested in project.

Schema: Each chunk stored with {text, html}.

Conclusion

Challenges:

- Handling token chunking correctly.
- Running Weaviate with Docker.

Lessons Learned:

- Trade-offs between FAISS (simple) and Weaviate (enterprise-ready).
- Importance of embedding model choice.

Future Work: Deploy online, add caching, extend multi-URL search.