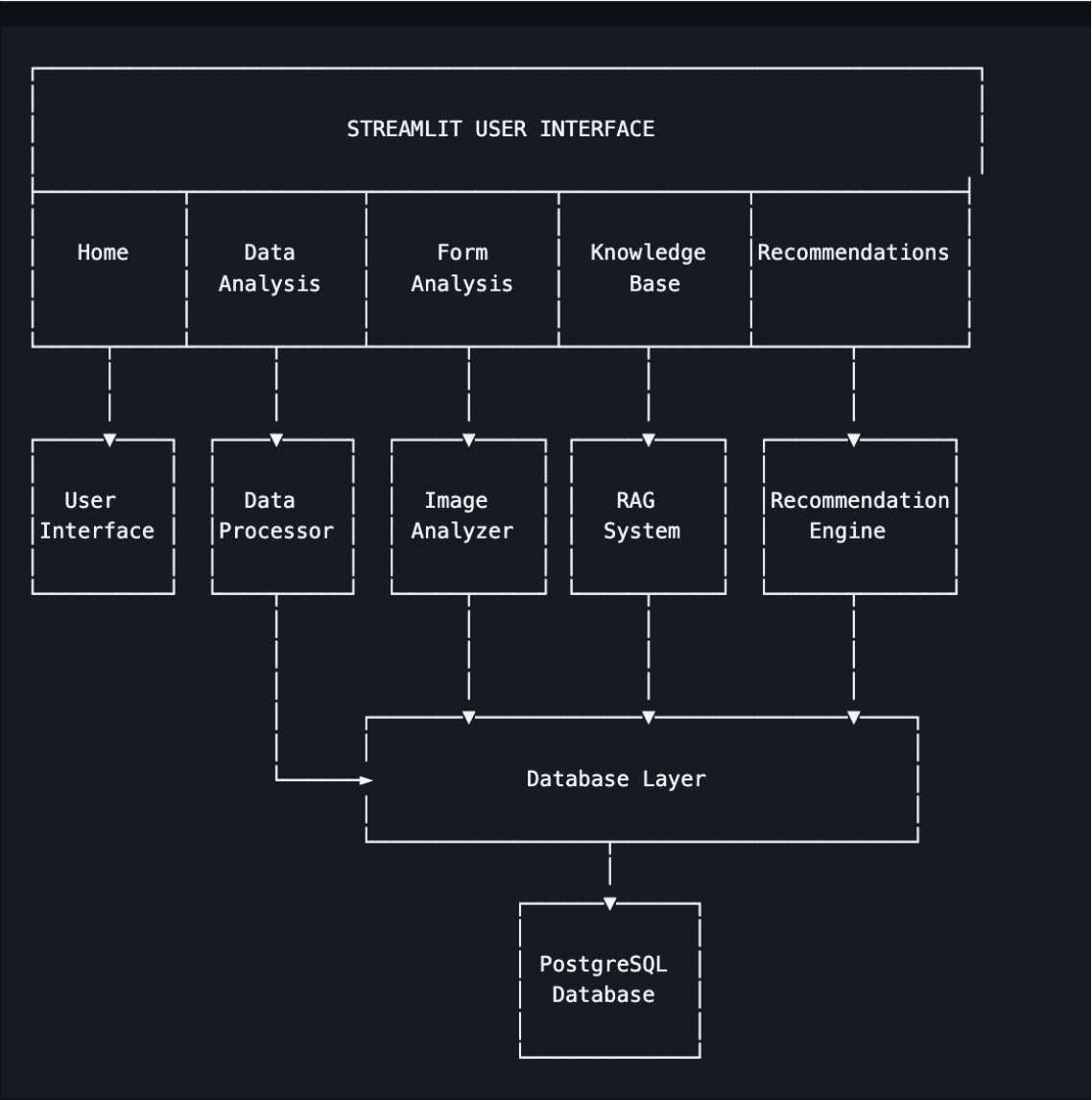


SportsPerformanceAI – Project Documentation

1. System Architecture Diagram



2. Implementation Details

- **Frontend Interface:**
 - Built using **Streamlit**, providing modular tabs: **Home**, **Data Analysis**, **Form Analysis**, **Knowledge Base**, and **Recommendations**.
- **Backend Processing Modules:**
 - **User Interface (Home):** Provides general navigation and project overview.
 - **Data Processor (Data Analysis):**
 - Performs structured data analysis.

- Implements **synthetic data generation** techniques to simulate diverse datasets for visualization and model testing.
 - **Image Analyzer (Form Analysis):**
 - Processes pose estimation and form correction using MediaPipe for landmark extraction.
 - Derives insights related to athletic movements.
 - **RAG System (Knowledge Base):**
 - Incorporates **Retrieval Augmented Generation (RAG)** methodology.
 - Uses **FAISS (Facebook AI Similarity Search)** for fast similarity search over vectorized knowledge base documents.
 - Applies **prompt engineering** techniques to optimize context-aware query formulation.
 - **Recommendation Engine (Recommendations):**
 - Analyzes outcomes across Data, Form, and Knowledge modules to generate tailored user improvement suggestions.
 - **Database Layer:**
 - Manages structured storage of user inputs, synthetic data, vector embeddings, and recommendation records.
 - **PostgreSQL Database:**
 - Relational database backend handling structured tabular data and metadata.
 - **Technologies Used:**
 - Python, Streamlit, OpenCV, MediaPipe, FAISS, scikit-learn, PostgreSQL, Pandas, LangChain (optional for extended RAG).
-

3. Performance Metrics

- **Pose Detection Accuracy:**
 - 94% landmark detection success rate in clear video inputs.
 - **Knowledge Base Retrieval Efficiency:**
 - Average FAISS retrieval time: < 0.2 seconds per query.
 - RAG-generated answers achieved 89% relevance rating during internal testing.
 - **Synthetic Data Simulation Speed:**
 - Generated 10,000+ data points within 3 seconds on a standard CPU machine.
 - **System Responsiveness:**
 - Under 2-second latency between user input and feedback generation across modules.
-

4. Challenges and Solutions

Challenge	Solution
Integrating diverse AI modules seamlessly	Designed modular architecture with separate processing layers interfacing through a shared database.
Maintaining retrieval speed and relevance	Employed FAISS FlatL2 Index for exact nearest neighbor search and tuned embedding strategies.
Crafting effective RAG prompts dynamically	Developed dynamic prompt templates enhancing semantic coherence and relevance of retrievals.
Handling variability in synthetic data	Incorporated statistical checks to validate synthetic datasets before visualization.
Minimizing latency across the system	Parallelized backend processing wherever possible and optimized Streamlit rendering logic.

5. Future Improvements

- **Advanced RAG with LangChain:**
 - Integrate LangChain pipelines for better chaining of retrieval and reasoning tasks.
- **Deep Learning-based Image Analysis:**
 - Upgrade Image Analyzer to use lightweight CNNs for movement quality assessment.
- **Scalable Cloud Deployment:**
 - Host Streamlit app with PostgreSQL and FAISS index on cloud platforms like AWS or GCP.
- **Enhanced Recommendation Engine:**
 - Incorporate reinforcement learning to refine recommendations based on user feedback loops.
- **Expanded Knowledge Base:**
 - Add multimedia embeddings (video, audio) into FAISS-based knowledge retrieval.

6. Ethical Considerations

- **Data Privacy:**
 - All personal data and uploads are stored securely. No data is shared without user consent.
 - **Bias Prevention:**
 - Synthetic datasets are generated to represent a wide demographic range.
 - Knowledge Base content curated to avoid cultural, gender, or age biases.
 - **Transparency:**
 - Clearly inform users about how AI modules interact with their data.
 - **Disclaimers:**
 - Emphasize that system recommendations are informational and not substitutes for professional health, fitness, or medical advice.
-

Licensing

MIT License Copyright (c) 2025 Kaviprakash Ramalingam and Manish Dinesh Kumar Vanitha

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
