

KidsSmart+ Educational Database

Prepared by: Kavin Nanthakumar & Mohamed Imaad

Introduction & Background



- AUTOMATES COLLECTION & MANAGEMENT OF EDUCATIONAL PROGRAM DATA

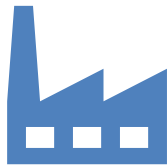


- SOURCES: GOVT WEBSITES, EVENTBRITE, MEETUP, LIBRARIES



- CENTRALIZED PLATFORM FOR INSTITUTIONS & PARENTS

Industry Context & Market Trends



- EdTech industry growing at 20% annually



- 60% of institutions adopting automated program management



- Shift to cloud-based, real-time data processing

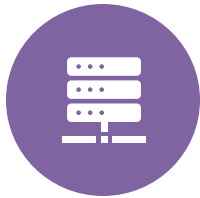
Unique Value Proposition



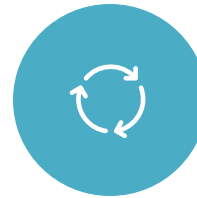
- Competitors:
Brightwheel,
HiMama, Procare



- Advantages:



- Real-time multi-source data integration



- Automated deduplication & validation



- User-friendly dashboard & analytics



Project Aims & Objectives



- Automate Data Collection: Web scraping & API integration



- Ensure Data Integrity: Validation & standardization



- Enhance Decision-Making: Interactive dashboard

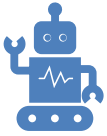


- Secure Data Storage: Encryption & access controls



- Optimize Performance: Fast API response times

Functional Requirements



- Data Collection:
Automated web
scraping



- Data Processing:
Standardization &
duplicate removal



- Dashboard:
Interactive UI with
filtering & visualization



- Security: AES-256
encryption, MFA,
secure APIs



- Performance: <2s API
response time, 99.9%
uptime

System Architecture



- Front-End: Streamlit UI



- Back-End: Flask API



- Database: SQLite (Scalable options available)



- Security: Encryption, authentication, API security

Use Case Analysis



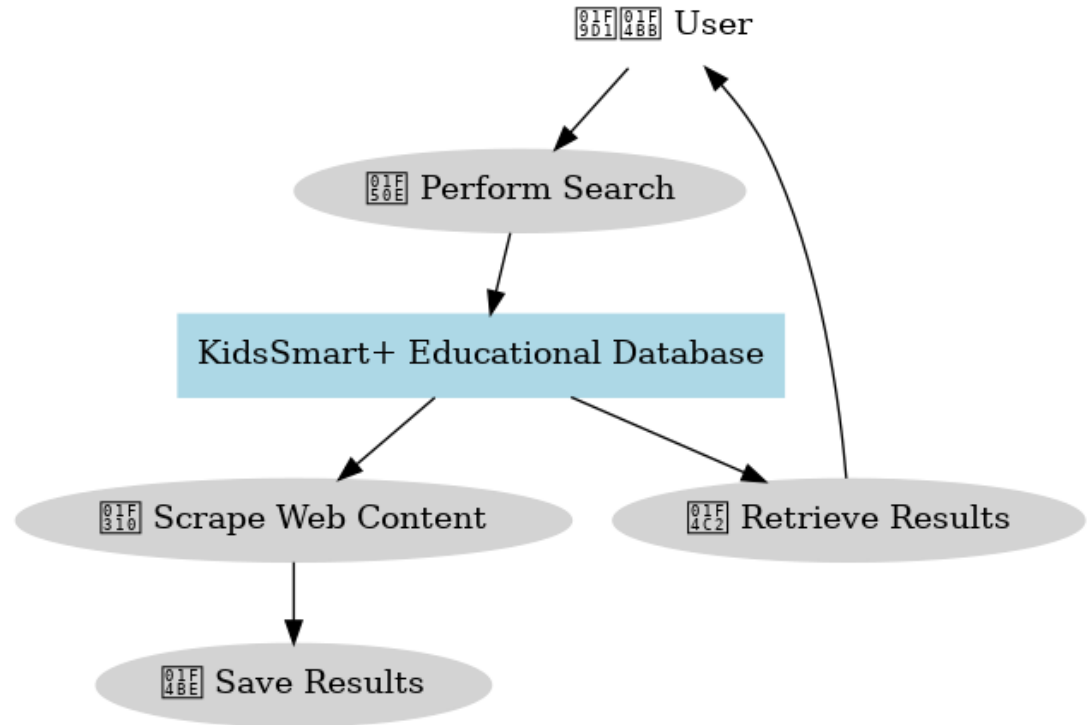
- Data Collection:
Admin/API initiates
scraping



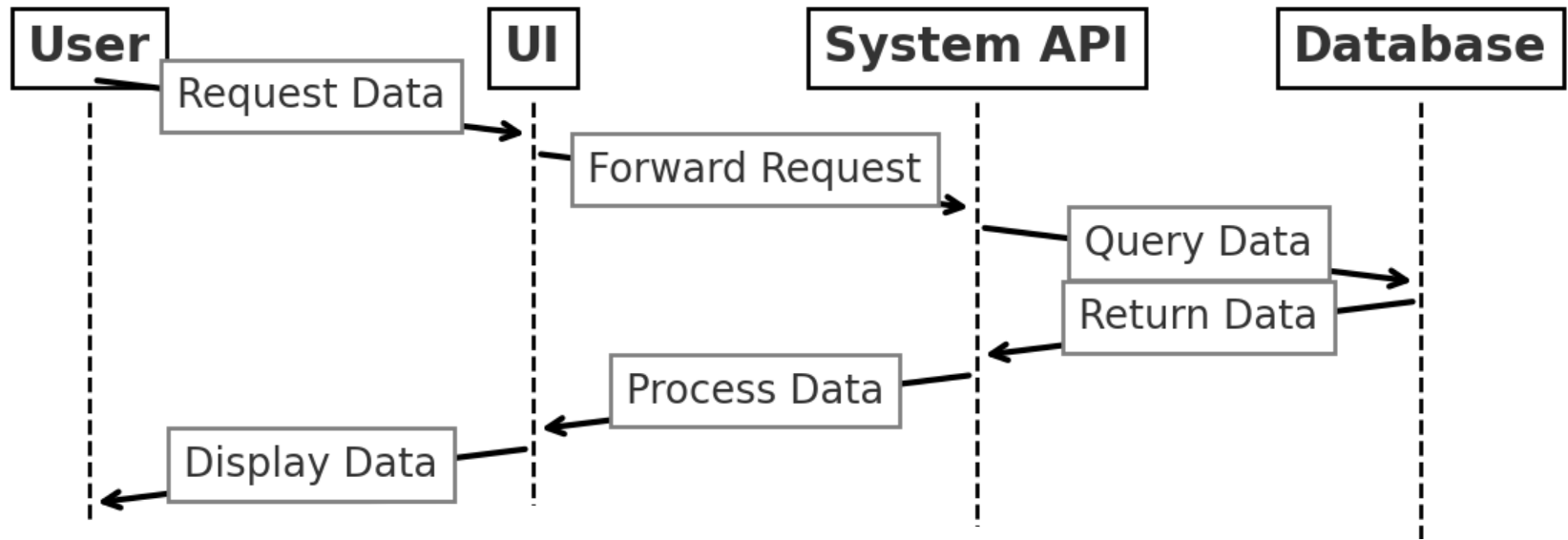
- User Login:
Authentication &
dashboard access



- Report
Generation: Filters
& exports data



Simple and Clear Sequence Diagram



Sequence Diagram

- User requests data → API fetches data → Database returns data → API processes & sends data → User receives data

Resource Management

- Cloud Storage: AWS S3/Azure Blob

- API Services: Google Search API

- Development Tools: Flask, Streamlit

- Security Measures: AES-256 encryption, MFA

Pseudocode & Implementation



- STREAMLIT UI & SEARCH FUNCTIONALITY



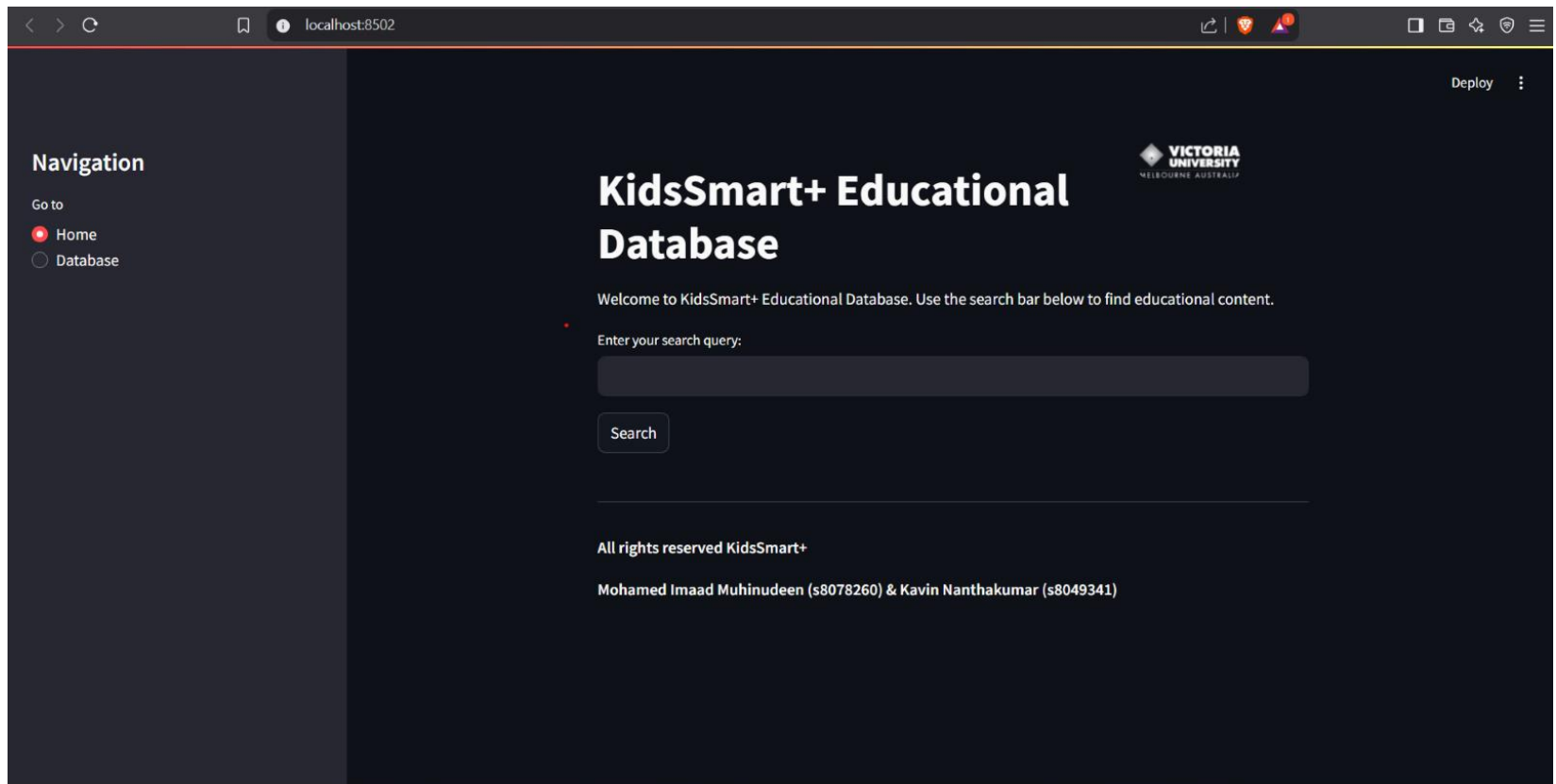
- WEB SCRAPING WITH BEAUTIFULSOUP



- DATABASE MANAGEMENT USING SQLITE

UI Design Features

- User-Friendly Interface: Intuitive navigation
- Responsive Design: Desktop, tablet, mobile
- Accessibility Standards: Optimized color contrast



Risk Management



- Data Breach:
Encryption & access
control



- API Rate Limits:
Caching & API key
rotation



- Scraping Failures: Error
handling & fallback
mechanisms

Timeline & Milestones

- Phase | Duration | Key Deliverables
- Requirement Gathering | 2 weeks | Documentation, competitor analysis
- Development | 4 weeks | Backend, frontend, database setup
- Testing | 3 weeks | Security & bug fixes
- Deployment | 1 week | System launch & monitoring



Conclusion



- EFFICIENT EDUCATIONAL DATA RETRIEVAL SYSTEM



- SECURE, SCALABLE, AND AUTOMATED



- FUTURE ENHANCEMENTS: AI-DRIVEN SEARCH, SMART FILTERING, IMPROVED UI

A large, solid red oval with a slight white highlight on the left side, serving as a background for the text.

Thank You!

Questions & Discussion