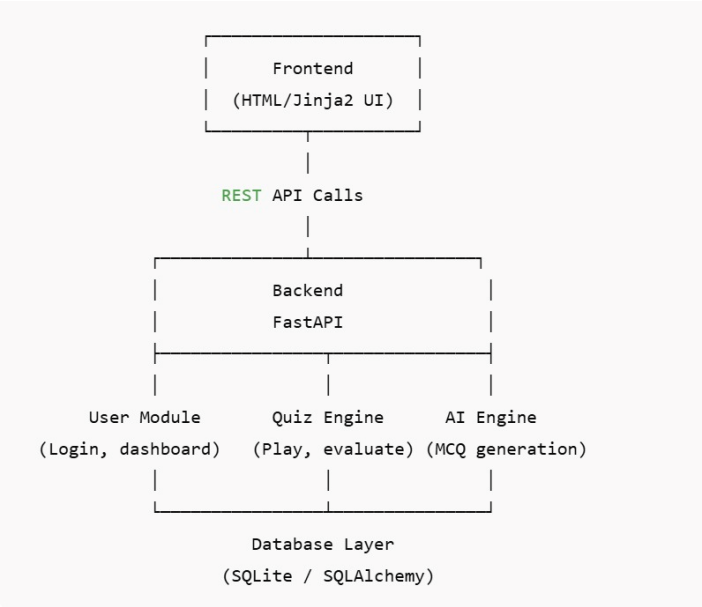
RetainIQ – System Design Document

Overview

RetainIQ is an AI-powered revision & quiz engine that generates MCQs from uploaded documents, identifies weak concepts, and schedules personalized revision using spaced repetition. The system is built using FastAPI, SQLite, Jinja2 templates, and integrates OpenAI LLMs for quiz creation.

High-Level Architecture



Component Descriptions

A. Frontend Layer

- Built using Jinja2, HTML, CSS, JS
- Pages include:
- Upload documents
- Flashcards viewer
- Quiz settings & quiz player
- Dashboard (progress tracking)

B. Backend Layer

Module	Responsibilities
User Management	login, logout, sessions
Document Module	upload PDFs, extract text, store content
AI Quiz Generator	generate MCQs using OpenAI
Quiz Runtime Engine	show questions, track answers, score
Weak Concept Detector	analyze incorrect answers
Spaced Repetition Scheduler	schedule Day1 → Day3 → Day7 revisions

Database Design

Tables

Users

| id | name | email | password |

Documents

| id | user_id | filename | content | created_at |

Quizzes

| id | user_id | document_id | difficulty | questions_json |

Quiz Attempts

| id | quiz_id | user_id | score | attempt_data | weak_topics

AI Module Design

Input

Raw text extracted from PDF

Pipeline

Text Cleanup

Topic Extraction

MCQ Generation

Difficulty Classification

Validation

Output

List of MCQs:

```
{
  "question": "...",
  "options": ["A", "B", "C", "D"],
  "correct_index": 2,
  "difficulty": "easy"
}
```

Quiz Runtime Flow

User chooses difficulty (easy / medium / hard)

Platform loads questions from session

User selects answer → submitted to backend

System:

stores attempt

identifies weak concepts

queues revision session if needed

Spaced Repetition Logic

Cycle	After Attempt
Review 1	Day 1
Review 2	Day 3
Review 3	Day 7
Review 4	Day 14

Deployment Overview (we are still planning to**)**