ZuAI Backend Engineer Evaluation Task

Sample Paper API with Gemini Integration

This task focuses on building a robust and efficient API for managing and processing sample papers, with a specific emphasis on integrating Gemini for PDF and text extraction. This task is designed to evaluate a candidate's ability to build a production-ready feature leveraging advanced language models.

Core Requirements:

1. Data Modeling and Validation:

   - Design comprehensive Pydantic models for the sample paper JSON structure, including nested models for sections and questions.

   - Ensure these models handle validation (e.g., data types, required fields, custom validation rules).

2. API Endpoints:

   - POST /papers: Creates a new sample paper from JSON input (validated by your Pydantic models). Returns the created paper's ID.

   - GET /papers/{paper_id}: Retrieves a sample paper by ID. Returns the JSON representation. Implement Redis caching for this endpoint.

   - PUT /papers/{paper_id}: Updates an existing sample paper (partial updates supported).

   - DELETE /papers/{paper_id}: Deletes a sample paper.

   - POST /extract/pdf: Accepts a PDF file upload. Uses Gemini to extract information and convert it to the sample paper JSON format.

   - POST /extract/text: Accepts plain text input. Uses Gemini to extract information and convert it to the sample paper JSON format.

   - GET /tasks/{task_id}: Checks the status of a PDF extraction task.

Gemini Integration and PDF/Text Processing:

Integrate Gemini for both PDF and text extraction. Provide clear instructions in your README on how to set up the Gemini environment and any necessary API keys.

1. PDF Processing (Asynchronous): Use asynchronous processing for PDF extraction.

2. Text Processing (Synchronous): Text extraction can be synchronous.

3. Handle potential errors during Gemini interactions gracefully.

Database Integration (MongoDB):

Store sample papers and task status/results in MongoDB. Design an efficient schema and consider appropriate indexing strategies.

Caching (Redis):

Implement Redis caching for the GET /papers/{paper_id} endpoint. Include cache invalidation logic.

API Documentation:

Generate and provide interactive API documentation using Swagger UI/Redoc (via FastAPI).

Brownie Points (Optional):

1. Enhanced Gemini Prompts: Experiment with different prompt engineering techniques to improve the accuracy of extraction.

2. Search Functionality: Implement full-text search capabilities on the question and answer fields.

3. Rate Limiting: Implement rate limiting to prevent API abuse.

4. Schema Validation and Error Handling: Implement advanced schema validation and error handling.

5. Security Considerations: Implement basic security measures.

Technology Stack:

Python 3, FastAPI, Pydantic, MongoDB, Redis, Gemini Python library, Asynchronous processing

libraries, pytest for testing.

Evaluation Criteria:

1. Completeness: Implementation of all core requirements.

2. Correctness: API endpoints function correctly and produce expected results.

3. Gemini Integration: Effective integration with Gemini for PDF and text extraction.

4. Code Quality: Clean, well-structured, and maintainable code.

5. Efficiency: Efficient database queries, appropriate caching, and asynchronous processing.

6. Error Handling: Robust error handling and informative error messages.

7. Testing: Comprehensive test coverage.

8. Documentation: Clear and concise documentation, including Gemini setup instructions.

9. Brownie Points: Implementation of optional features, demonstrating initiative.

Appendix: Sample Paper JSON Structure

```
{
  "title": "Sample Paper Title",
  "type": "previous_year",
  "time": 180,
  "marks": 100,
  "params": {
    "board": "CBSE",
    "grade": 10,
    "subject": "Maths"
  },
  "tags": [
    "algebra",
    "geometry"
  ],
  "chapters": [
    "Quadratic Equations",
```

    "Triangles"
  ],
  "sections": [
   {
     "marks_per_question": 5,
     "type": "default",
     "questions": [
      {
        "question": "Solve the quadratic equation: x^2 + 5x + 6 = 0",
        "answer": "The solutions are x = -2 and x = -3",
        "type": "short",
        "question_slug": "solve-quadratic-equation",
        "reference_id": "QE001",
        "hint": "Use the quadratic formula or factorization method",
        "params": {}
      },
      {
        "question": "In a right-angled triangle, if one angle is 30°, what is the other acute angle?",
        "answer": "60°",
        "type": "short",
        "question_slug": "right-angle-triangle-angles",
        "reference_id": "GT001",
        "hint": "Remember that the sum of angles in a triangle is 180°",
        "params": {}
      }
     ]
   }
  ]
}

Comprehensive Sample Paper Creator

Paper Title

| Previou... ∨ | 180 | 0 |

| CBSE ∨ | 10 ∨ | Maths ∨ |

Questions

Question 1 ⌃ ⊖

Question

Answer

| Short ∨ |

Question Slug

Reference ID

Hint (optional)

⊕ Add Question

Create Sample Paper