

AI 기반 강의 요약 시스템 개발 (백엔드)

20204080 김병환

목차

1. 구현 계획 및 흐름
2. 개발에 필요한 기술 스택
3. 데이터 구조 설계(RDB & NoSQL)
4. DB 테스트 결과

1. 구현 계획 및 흐름

1. 전체 백엔드 구조 설계, 기술 스택 학습 (Spring Boot, MySQL, MongoDB 등)
2. Spring Boot 프로젝트 세팅, REST API 설계 , DB설계(RDB & NoSQL)및 테스트
3. 정형 데이터는 MySQL, 유동적인 문서 구조는 MongoDB에 저장(8월 초 마무리 예정)
4. JWT + Spring Security 인증 구현, 클라우드 배포 테스트(GCP or AWS), 에러 처리 및 기능 점검 (8월 중순 마무리 예정)
5. 추가적인 기능 보완 및 버그 수정(8월 말 마무리 예정)

2. 개발에 필요한 필요한 기술 스택(백엔드)

1. Spring Boot

백엔드 프레임워크로 사용, REST API 설계, 의존성 주입, 웹 서버 구동, DTO, Controller, Service 등

2. Spring Data JPA (MySQL 사용)

객체와 관계형 DB(MySQL) 간 매핑을 처리, CRUD 작업을 쉽게 처리할 수 있도록 지원, DB: MySQL 사용

3. Spring Data MongoDB(MongoDB 사용)

MongoDB와 Java 객체 간 매핑 처리, 요약/질문처럼 유동적인 데이터는 문서형 DB인 MongoDB에 저장

4. HTTP Request / REST API

클라이언트가 서버에 데이터를 요청하는 표준 방식

5. WebSocket

서버 ↔ 클라이언트 간 양방향 실시간 통신 가능

6. JWT (JSON Web Token) + Spring Security

사용자 인증 및 인가 처리에 사용

7. Cloud (예: AWS, GCP 등)

백엔드 서버, 데이터베이스, 스토리지 등을 클라우드에 배포 가능

3. 데이터 구조 설계(RDB & NoSQL)

users			
PK	id	BIGINT AUTO INCREMENT	사용자 고유 ID
	password	VARCHAR(255)	비밀번호
	email	VARCHAR(255)	이메일
	role	VARCHAR(50)	권한

uploads			
PK	id	BIGINT AUTO INCREMENT	업로더 고유 ID
FK	user_id	BIGINT	업로더 참조
	file_name	VARCHAR(255)	저장된 파일명
	upload_time	TIMESTAMP DEFAULT CURRENT TIMESTAMP	업로드 일시
	status	VARCHAR(50)	상태 구분(UPLOADED, PROCESSING, DONE)

stt_results			
PK	id	BIGINT AUTO INCREMENT	STT 결과 고유 ID
FK	upload_id	BIGINT	어떤 파일의 STT 결과인지
	raw_text	TEXT	STT로 변환된 전체 텍스트
	language	VARCHAR(50)	언어 (예: ko, en)
	confidence	FLOAT or DECIMAL(4,3)	인식 신뢰도

MongoDB ↔ MySQL 간의 논리적 연결

컬렉션 : summaries

```

{
  _id: ObjectId("abc123"),
  sttResultId: 1,
  summaryText: "이것은 요약문입니다.",
  modelUsed: "gpt-4",
  questions: [
    {
      questionType: "OBJECTIVE",
      questionText: "질문 내용입니다.",
      answer: "정답 또는 해설"
    },
    {
      questionType: "DESCRIPTIVE",
      questionText: "서술형 질문입니다.",
      answer: "서술형 답변"
    }
  ]
}

```

MongoDB

SummaryDocument : 하나의 요약 결과, STT 결과 1개당 대응,
그에 따른 예상문제 리스트(QuestionDocument)를 배열로 포함

QuestionDocument : 하나의 요약 결과의 여러 개의 문제들

4. DB 테스트 결과(swagger로 테스트)



user-controller

GET /api/users 모든 사용자 목록 조회

POST /api/users 사용자 저장

GET /api/users/email 이메일로 사용자 조회

upload-controller

GET /api/uploads 전체 업로드 목록 조회

POST /api/uploads 업로드 저장

summary-controller

POST /api/summaries 요약 저장

GET /api/summaries/stt-result/{sttResultId} STT 결과 ID로 요약 조회

stt-result-controller

POST /api/stt-results

GET /api/stt-results/upload/{uploadId} 업로드 ID로 STT 결과 조회

4. DB 테스트 결과(swagger로 테스트)

user-controller

GET /api/users 모든 사용자 목록 조회

Code Details

200

Response body

```
[
  {
    "id": 1,
    "name": "병비더",
    "email": "ddd@naver.com",
    "role": "user"
  },
  {
    "id": 2,
    "name": "병지노",
    "email": "aaa@naver.com",
    "role": "user"
  }
]
```

id : 1, 2인 유저 존재

upload-controller

GET /api/uploads 전체 업로드 목록 조회

Code Details

200

Response body

```
[
  {
    "id": 1,
    "fileName": "test-file.mp3",
    "uploadTime": "2025-07-22T05:43:04.427Z",
    "status": "UPLOADED",
    "user": {
      "id": 2,
      "name": "병지노",
      "email": "aaa@naver.com",
      "role": "user"
    }
  },
  {
    "id": 2,
    "fileName": "test-file2.mp3",
    "uploadTime": "2025-07-22T05:43:04.427Z",
    "status": "UPLOADED",
    "user": {
      "id": 2,
      "name": "병지노",
      "email": "aaa@naver.com",
      "role": "user"
    }
  }
]
```

id : 2인 사람이 파일 업로드 2개

stt-result-controller

Code Details

200

Response body

```
[
  {
    "id": 1,
    "uploadId": 2,
    "rawText": "테스트",
    "language": "ko",
    "confidence": 0.92
  }
]
```

id : 2인 사람이 파일 업로드 2개 한 것중에
2번째 파일에 STT로 변환된 텍스트 존재
("test-file2.mp3" 파일에 대해 STT 결과가 생성)

4. DB 테스트 결과(swagger로 테스트)

stt-result-controller

Code Details

200

Response body

```
[
  {
    "id": 1,
    "uploadId": 2,
    "rawText": "테스트",
    "language": "ko",
    "confidence": 0.92
  }
]
```

summary-controller

Code

Details

200

Response body

```
[
  {
    "id": "687f306c3ab1c03bd77004f3",
    "sttResultId": 1,
    "summaryText": "요약된 결과",
    "modelUsed": "gpt",
    "questions": [
      {
        "questionType": "OBJECTIVE",
        "questionText": "객관식 문제",
        "answer": "정답, 해설"
      },
      {
        "questionType": "DESCRIPTIVE",
        "questionText": "서술형",
        "answer": "답"
      }
    ]
  }
]
```

STT로 변환된 텍스트의(id: 1) 요약+ 문제 -> id : "687f306c3ab1c03bd77004f3"에 저장(MongoDB)

8월 계획

~8월 10일 : 음성 데이터를 텍스트 변환 기능 연동, MongoDB에 원본 데이터 저장 구조 완성, 음성 데이터를 텍스트 변환 기능 연동

~8월 17일 : 요약/문제 생성용 비동기 처리 흐름 구현, MySQL에 정형 결과 저장

~8월 24일 : JWT + Spring Security로 인증 구현, 1차 전체 흐름 통합 및 테스트

~8월 31일 : 클라우드(GCP 또는 AWS) 배포 테스트, 추가적인 기능 보완 및 버그 수정

감사합니다
