

백엔드 엔지니어 사전과제

래티스 백엔드 엔지니어에 지원해주셔서 감사합니다 🙏

래티스 사전과제는 이 페이지 아래의 문제를 해결하는 api를 작성하는 것입니다. 문제를 해결하기 전에 사전과제 진행 방법을 읽어주시고 유의해 주세요!

사전과제 진행 방법

- Node 환경에서 프레임워크는 Express나 NestJs를 자유롭게 선택하셔도 됩니다. root의 readme.md에 완성해주신 api를 로컬 환경에서 띄울 수 있는 방법을 적어주세요.
macOS 또는 Windows 환경에서 실행 가능해야 한다는 것을 기억해 주세요.
- 사전과제 해결을 위해서는 db가 필요합니다. sqlite in-memory db를 사용하는 것을 권장합니다. 익숙하지 않으시다면 어떤 방식으로든 db와 비슷한 동작을 하는 것으로 충분합니다.
예) aws rds 사용, memory에 저장, 로컬 파일로 저장
- 명세하지 않은 응답에 대해서는 자유롭게 처리해주시면 됩니다. 예러 응답의 경우 적합하다고 생각하는 방식으로 response를 보내주시면 됩니다.
- 퍼포먼스와 코드 퀄리티도 확인합니다. 선택하신 프레임워크나 db의 한계로 발생하는 것들은 평가하지 않습니다.
- 코드는 **이메일을 통해 zip 형식으로 제출**해주시면 됩니다. 제출해주신 api 프로젝트는 로컬 환경에서 띄워서 평가합니다. localhost:80/ 으로 요청을 보내고 응답이 잘 오는지 확인합니다.

문제

래티스는 계약을 체결한 후에 계약서를 관리할 수 있는 프로젝트 기능을 제공하고 있습니다. 프로젝트는 다음과 같은 기능이 필요합니다.

- 사용자는 프로젝트를 여러개 만들 수 있습니다.
- 사용자는 특정 프로젝트에 계약서 파일을 업로드 할 수 있습니다.
- 사용자는 프로젝트의 정보를 확인 할 수 있습니다.

- 사용자는 특정 프로젝트에 여러 메모를 남길 수 있습니다.

Project

사용자가 프로젝트를 생성할 때에는 프로젝트의 제목을 입력합니다. 샘플 데이터는 다음과 같습니다.

endpoint	/projects
method	POST
content-type	application/json
body	{ "title": "NBA 중계 협상 건" }

프로젝트가 생성되면 200 응답과 함께, 작성된 프로젝트의 유니크한 id 혹은 uuid를 응답으로 돌려줍니다.

body는 클라이언트에서 json으로 파싱해서 사용합니다.

http status	200
body	{ "uuid": "abc" }

생성된 프로젝트의 정보를 확인 할 수 있습니다.

endpoint	/projects/{uuid}
method	GET
content-type	application/json

샘플 응답은 다음과 같습니다. 프로젝트 정보를 내려줍니다.

http status	200
body	{ "uuid": "abc", "title": "NBA 중계 협상 건", "memos": [], "fileUrl": null, }

	"createdAt": 1646380534, }
--	-------------------------------

사용자는 생성한 모든 프로젝트를 확인할 수 있습니다. 아무런 query parameter 없이 요청하면 최대 100개까지만 내려줍니다. page, size가 쿼리 파라미터로 요청에 포함될 수 있습니다. page가 0, size가 20이면 1번째부터 20번째까지의 프로젝트를 내려줍니다. page가 1, size가 20이면 21번째부터 40번째의 프로젝트를 내려줍니다.

프로젝트는 **최신순**으로 정렬되어야 합니다. 첫 번째 프로젝트가 가장 최신의 프로젝트이고, 두 번째 프로젝트는 그 직전 생성된 프로젝트입니다.

endpoint	/projects
method	GET
content-type	application/json
query parameter	page=0&size=20

http status	200
body	[{ "uuid": "abc", "title": "NBA 중계 협상 건", "fileUrl": "file/test.pdf", "memos": [], "createdAt": 1646380534 }, { "uuid": "def", "title": "임대계약", "fileUrl": null, "memos": [{ "id": 123, "content": "계약 만료일 확인하기" }], "createdAt": 1646380539 }]

File

사용자는 특정 프로젝트에 계약서를 업로드 할 수 있습니다.

endpoint	/projects/{uuid}/file
method	POST
content-type	multipart/form-data
body	{ "file": File }

http status	200
body	{ "uuid": "abc" }

Memo

사용자는 특정 프로젝트에 메모를 남길 수 있습니다.

endpoint	/memos
method	POST
content-type	application/json
body	{ "projectUuid": "def", "content": "계약 만료일 확인하기" }

샘플 응답은 다음과 같습니다. 답변의 id를 내려줍니다.

http status	200
body	{ "id": 123 }

사용자는 메모를 확인할 수 있습니다. projectUuid 를 parameter 로 넣어서 해당 프로젝트에 관련된 메모만 받아오는 것도 가능합니다.

endpoint	/memos
method	GET
content-type	application/json
query parameter	page=0&size=20&projectUuid=def

(이 요청은 projectUuid=def에 해당하는 메모들만 20개 내려줍니다.)