

# FastAPI와 MySQL을 활용한 웹 백엔드 구축

20214072 황주연



# 사용 기술 및 도구

🐍 백엔드 프레임워크: FastAPI

🔗 ORM: SQLAlchemy

🐘 데이터베이스: MySQL (Workbench로 확인)

📦 기타: Pydantic (데이터 검증), Python

# 핵심 구현 내용

기 DB 모델 정의 (User)

기 Pydantic 스키마 정의 (UserCreate, UserRead)

기 API 기능 구현

- POST /users → 사용자 생성
- GET /users → 전체 조회
- GET /users/{id} → 개별 조회
- PUT /users/{id} → 수정
- DELETE /users/{id} → 삭제

기 예시

```
@app.post("/users", response_model=UserRead)
def create_user(user: UserCreate):
    db = SessionLocal()
    db_user = User(name=user.name, email=user.email)
    db.add(db_user)
    db.commit()
    db.refresh(db_user)
    db.close()
    return db_user
```

# 실행화면

localhost:8000/docs/

## FastAPI 0.1.0 OAS 3.1

/openapi.json

### default

- GET** /users Get Users
- POST** /users Create User
- GET** /users/{user\_id} Get User
- PUT** /users/{user\_id} Update User
- DELETE** /users/{user\_id} Delete User

### Schemas

- HTTPValidationError** > Expand all **object**
- UserCreate** > Expand all **object**
- UserRead** > Expand all **object**
- ValidationError** > Expand all **object**

# 실행화면

MySQL Workbench

work x

File Edit View Query Database Server Tools Scripting Help

Navigator: SCHEMAS

Filter objects

- sys
- testdb
  - Tables
  - Views
  - Stored Procedures
  - Functions

Query 1

```
1 • SELECT * FROM users;
```

Limit to 1000 rows

Result Grid

	id	name	email	created_at
▶	1	홍길동	hong@example.com	2025-07-21 00:05:55
	2	홍길동	dasdsa@naver.com	2025-07-21 00:09:26
	3	장주연	wmdus50@naver.com	2025-07-21 00:11:02
▲	more	more	more	more

Administration: Schemas

# 테스트

POST

/users Create User

Parameters

Cancel

Reset

No parameters

Request body required

application/json

Edit Value | Schema

```
{
  "name": "아무개",
  "email": "amugae@naver.com"
}
```

Result Grid | Filter Rows: | Edit:

	id	name	email	created_at
▶	1	홍길동	hong@example.com	2025-07-21 00:05:55
	2	홍길동	dasdsa@naver.com	2025-07-21 00:09:26
	3	홍주연	windus50@naver.com	2025-07-21 00:11:02
	4	아무개	amugae@naver.com	2025-07-21 22:54:51
•	NULL	NULL	NULL	NULL

Execute

Clear

# 앞으로의 계획

기프론트엔드 연동하여 실제 웹페이지에서 API 호출

기사용자 인증 추가 (JWT)

기비동기 처리 및 에러 핸들링 고도화

기배포 환경 구축 (e.g. Docker)

# 8월 계획

1주차	캡스톤 디자인 - MongoDB로 비정형 데이터 저장하는 법 근감소증 연구 - 7월 25일 연구 미팅 내용에 대해 공부하고 제 대로된 계획은 미팅 후 다시 계획할 예정
2주차	캡스톤 디자인 - 외부 AI 서버 연동 근감소증 연구 - 공부
3주차	캡스톤 디자인 - WebSocket으로 실시간 알림 근감소증 연구 - 공부
4주차	캡스톤 디자인 - Swagger 문서화 + Postman 테스트 근감소증 연구 - 공부

일단 계획은 이렇고 더 빨리 끝낼 수 있다면 빨리  
끝내고 남은 것을 공부할 예정입니다.