

# Cloud App - Kelarin

---

Kelarin merupakan sistem manajemen tugas berbasis cloud yang dirancang untuk membantu mahasiswa dalam mengelola tugas akademik secara terstruktur dan kolaboratif. Dengan Kelarin, pengguna dapat menambahkan tugas, menentukan deadline, serta membagikannya kepada anggota tim sehingga seluruh anggota dapat memantau dan mengerjakan tugas secara bersama-sama.

Kelarin ditujukan untuk mahasiswa yang sering kali mengalami permasalahan kesulitan dalam mengatur tugas bersama, seperti tugas yang tidak terdokumentasi dengan baik, pembagian peran yang tidak jelas, atau keterlambatan penyelesaian tugas. Melalui penggunaan sistem ini seluruh data tugas disimpan secara terpusat pada cloud, sehingga dapat diakses kapan saja dan di mana saja. Dengan demikian, proses kerja kelompok menjadi lebih terorganisir, transparan, dan efisien.

## ETHEREAL TEAM

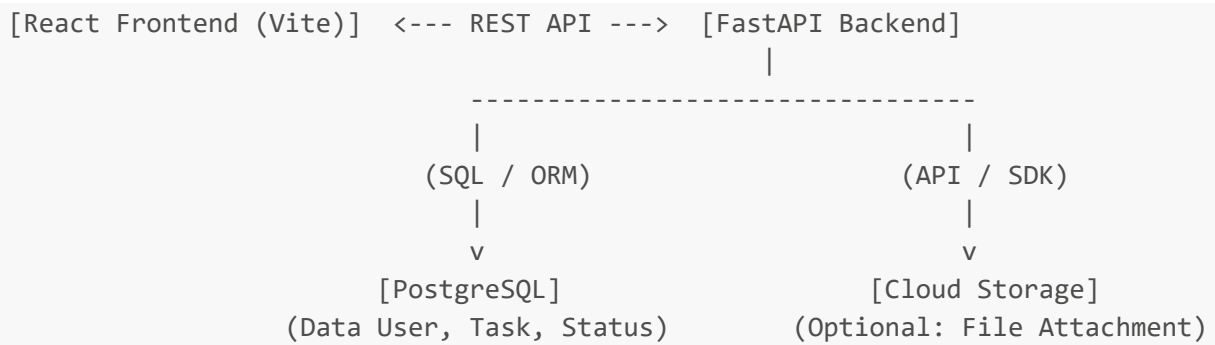
Nama	NIM	Peran
Tiya Mitra Ayu	10231088	Lead Backend
Amazia Devid Saputra	10231013	Lead Frontend
Alsha Dwi Cahya	10231011	Lead DevOps
Andini Permata Sari	10231015	Lead QA & Docs
Ansellma Tita Pakartiwuri P	10231017	Lead CI/CD & Deploy

## Tech Stack

Teknologi	Fungsi
FastAPI	Backend REST API
React	Frontend SPA
PostgreSQL	Database
Docker	Containerization
GitHub Actions	CI/CD
Railway/Render	Cloud Deployment

## Architecture

```
[Client / User]
  |
  (HTTPS)
  |
  v
```



## Penjelasan Arsitektur

### Client / User

Pengguna mengakses aplikasi Kelarin melalui browser menggunakan protokol HTTPS untuk memastikan komunikasi yang aman.

### React Frontend (Vite)

Frontend berfungsi sebagai antarmuka pengguna yang menangani tampilan dashboard, manajemen tugas, serta interaksi user. Frontend berkomunikasi dengan backend melalui REST API.

### FastAPI Backend

Backend bertanggung jawab dalam logika bisnis aplikasi seperti autentikasi, pengelolaan tugas (CRUD), pembaruan status, serta pengolahan data sebelum disimpan ke database atau cloud storage.

### PostgreSQL

Digunakan untuk menyimpan data terstruktur seperti data pengguna, daftar tugas, status, dan deadline.

### Cloud Storage

Digunakan jika terdapat fitur upload file seperti lampiran tugas atau bukti penyelesaian.

## Getting Started

### Prasyarat

- **Python 3.10+**
- **Node.js 18+**
- **Git**

---

## Tech Stack

### Backend

Komponen	Teknologi	Deskripsi
Language	Python 3.10+	Bahasa pemrograman utama
Framework	FastAPI	REST API server (High performance)
Database	PostgreSQL	Penyimpanan data relasional

Komponen	Teknologi	Deskripsi
API Gateway	Nginx	Reverse proxy & routing

## Frontend

Komponen	Teknologi	Deskripsi
Library	React.js	User Interface library
Build Tool	Vite	Frontend tooling & bundling (SPA)

## DevOps & Deployment

Komponen	Teknologi	Deskripsi
Container	Docker	Packaging aplikasi
Orchestration	Docker Compose	Multi-container management
CI/CD	GitHub Actions	Automated test & deploy
Cloud	Railway	Hosting & deployment platform

## Backend

```
cd backend
pip install -r requirements.txt
uvicorn main:app --reload --port 8000
```

## Frontend

```
cd frontend
npm install
npm run dev
```



## Roadmap

Minggu	Target	Status
1	Setup & Hello World	<input checked="" type="checkbox"/>
2	REST API + Database	<input type="checkbox"/>
3	React Frontend	<input type="checkbox"/>
4	Full-Stack Integration	<input type="checkbox"/>

Minggu	Target	Status
5-7	Docker & Compose	<input type="checkbox"/>
8	UTS Demo	<input type="checkbox"/>
9-11	CI/CD Pipeline	<input type="checkbox"/>
12-14	Microservices	<input type="checkbox"/>
15-16	Final & UAS	<input type="checkbox"/>