



## RANCANG BANGUN APPLICATION PROGRAMMING INTERFACE (API) MENGGUNAKAN GAYA ARSITEKTUR GRAPHQL UNTUK PEMBUATAN SISTEM INFORMASI PENDATAAN ANGGOTA UNIT KEGIATAN MAHASISWA (UKM) STUDI KASUS UKM STARLABS

Nanda Khoirul Akmal<sup>1</sup>, M Novinarsyah Dasaprawira<sup>2</sup>

<sup>1,2</sup> Universitas Nahdlatul Ulama Al Ghazali Cilacap

---

### Article Info:

Dikirim: 26 Mei 2022

Direvisi: 9 Juni 2022

Diterima: 29 Juni 2022

Tersedia Online: 30 Juni 2022

### Penulis Korespondensi:

Nanda Khoirul Akmal

Universitas Nahdlatul Ulama Al

Ghazali Cilacap

Email:

nanda.19572011009@students.unugh  
a.id

**Abstrak:** *Unit Kegiatan Mahasiswa (UKM) Science Technology And Computers Laboratories (STARLABS) adalah sebuah organisasi intra kampus yang berada di Universitas Nahdlatul Ulama Al Ghazali Cilacap yang bergerak di bidang IT. Akan tetapi organisasi ini belum memiliki suatu sistem informasi untuk manajemen rantai aktivitas yang berada di organisasi ini. Hal ini yang menyebabkan pendataan aktivitas organisasi dengan cara manual. Perancangan aplikasi ini diharapkan bisa membangun suatu API berarsitektur GraphQL untuk sistem informasi organisasi.*

**Kata kunci:** *graphql; API; organisasi.*

---

**Abstract:** *Student Activity Unit (UKM) Science Technology And Computers Laboratories (STARLABS) is an intra-campus organization located at Nahdlatul Ulama Al Ghazali University Cilacap which is engaged in IT. However, this organization does not yet have an information system to manage the chain of activities in this organization. This causes the data collection of organizational activities in a manual way. The design of this application is expected to be able to build an API with GraphQL architecture for organizational information systems.*

**Keywords:** *graphql; API; organization.*

---

## PENDAHULUAN

### 1.1 Latar Belakang

Perkembangan teknologi saat ini begitu cepat, hampir semua aspek kehidupan manusia menggunakan teknologi. Kebutuhan akan suatu alat untuk manajemen organisasi berbasis IT semakin dibutuhkan, seperti halnya organisasi dikampus yang disebut Unit Kegiatan Mahasiswa (UKM) yang setidaknya membutuhkan aplikasi pengelolaan data anggota, dll. Salah satunya adalah UKM Starlabs yaitu sebuah UKM yang berdiri sejak November 2015 di Universitas Nahdlatul Ulama Al Ghazali Cilacap. Walau berdiri sejak 2015, akan tetapi sampai saat tulisan ini dibuat Maret 2022 masih belum memiliki sistem pendataan anggota secara digital. Dahulu pernah menggunakan cara konvensional dengan catatan di kertas dan sirna ketika pergantian pengurus. Adapun ketika melakukan pembuatan suatu aplikasi, kita bisa menggunakan beberapa gaya arsitektur salah satunya adalah *Application Programming Interface* (API).

*Application Programming Interface* (API) adalah antarmuka yang dibangun oleh pengembang sistem sehingga beberapa atau seluruh fungsi sistem dapat diakses secara terprogram [1]. API juga biasa dianggap sebagai suatu kumpulan teknik yang jelas untuk menciptakan suatu komunikasi antara perangkat lunak yang berbeda beda komponen. Fungsi API adalah untuk memudahkan penggunaan teknologi tertentu ketika membangun perangkat lunak atau aplikasi bagi pengembang.

API memiliki beberapa arsitektur, ada REST (*Representational State Transfer*), GraphQL, dll. Disini penulis menggunakan arsitektur GraphQL. GraphQL adalah bahasa kueri untuk API dan runtime untuk memenuhi kueri tersebut dengan data yang ada [2].

Tujuan perancangan sistem informasi pendataan anggota UKM ini adalah agar seluruh UKM di lingkungan UNUGHA bisa melakukan pendataan anggotanya secara digital, dan data anggota tertata rapih.

### 1.2 Tinjauan Pustaka

#### 1.2.1 Underfetching

*Under-fetching* menjadi masalah pada sistem informasi karena berdampak menurunkan performa suatu web. Penurunan itu terja karena proses ‘meminta kembali’ data pada *backend di endpoint* yang lain.

Contoh misal dalam suatu sistem terdapat halaman yang meminta request di 2 *endpoint* yang berbeda, misal */user/id* untuk mendapatkan data pengguna dan */user/id/jurusan* untuk data jurusan yang dimiliki pengguna, maka sistem memiliki 2 pekerjaan yang harus dikerjakan, yaitu meminta data pada *endpoint /user* dan */user/jurusan*.

Masalah ini yang bisa diselesaikan oleh *GraphQL*. *GraphQL* bisa menentukan data apa yang perlu diambil berdasarkan *query* yang ada dari backend. *Backend* yang menentukan data apa yang hanya diperbolehkan diambil dan *frontend* yang menentukan data apa yang perlu diambil dari apa yang disediakan *backend*.

## 2. METODOLOGI PENELITIAN

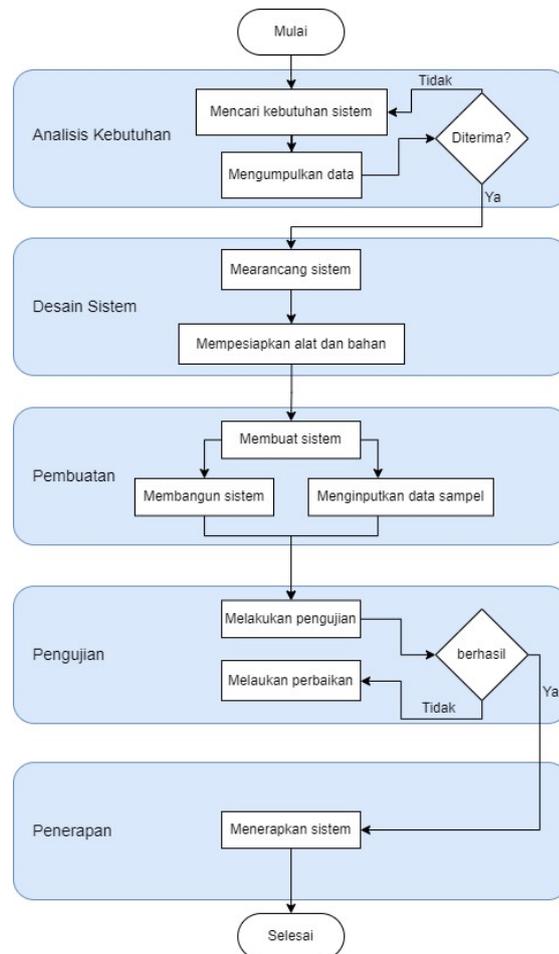
Penelitian ini dibagi menjadi 2 tahap, yaitu tahap pengambilan data dan pengembangan sistem.

### 2.1 Metode Pengambilan Data

Pengambilan data dilakukan dengan cara menginputkan secara manual dengan mengisi data anggota melalui form yang sudah disediakan oleh sistem. Form tersebut memiliki field antara lain nama, nim, prodi, email, whatsapp, alamat, tanggal lahir, angkatan, dan divisi.

### 2.2 Metode Pengembangan Sistem

Metode pengembangan sistem yang penulis gunakan adalah metode SDLC (*System Development Life Cycle*). Ditunjukkan pada gambar 1.



**Gambar 1. Metode SDLC**

Uraian dari langkah-langkah penelitian yang ada di gambar 1 sebagai berikut:

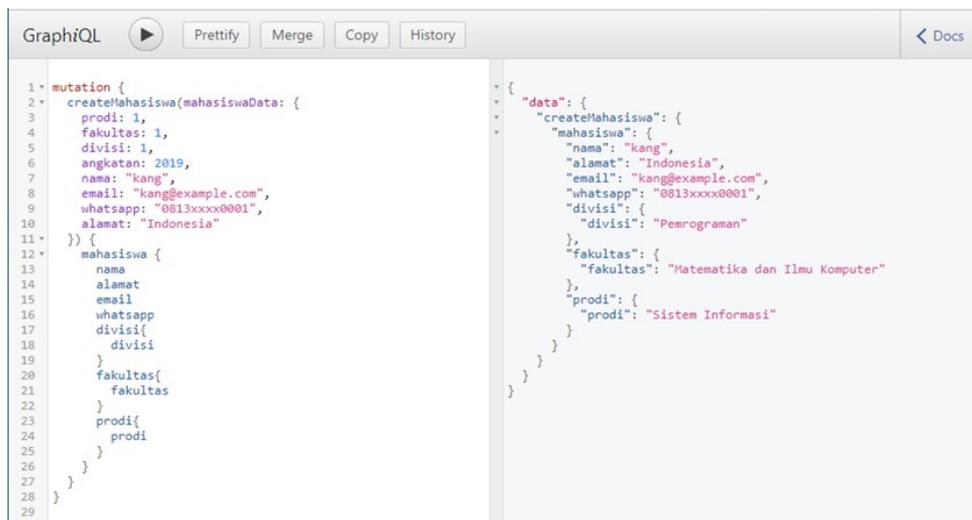
- 1) Analisis Kebutuhan  
Pada tahap ini yaitu mencari kebutuhan apa saja yang diperlukan guna membangun sistem dan mengumpulkan data untuk menentukan kebutuhan sistem yang akan dibangun.
- 2) Mendesain Sistem  
Pada tahapan ini menentukan unsur - unsur yang diperlukan untuk dimasukkan ke dalam sistem yang akan dibuat. Penentuannya ini berdasarkan hasil dari analisis kebutuhan yang dilakukan sebelumnya. Langkah pada tahapan perancangan ini meliputi perancangan desain sistem dan mengumpulkan alat dan bahan yang akan digunakan.
- 3) Proses Pembuatan  
Pada tahapan ini adalah proses mewujudkan sistem dengan cara menulis kode program berdasarkan hasil dari perancangan yang sudah dilakukan.
- 4) Proses Pengujian  
Pada tahap ini adalah melakukan pengujian sistem yang telah dibuat. Ini dilakukan untuk mengurangi resiko terjadinya kesalahan sebelum diterapkan. Dan juga tahapan pengujian ini bertujuan untuk mengetahui kesesuaian sistem dengan analisis yang dibuat. Langkah - langkah tahap pengujian antara lain:
  - a. Menjalankan sistem diperangkat yang cocok dengan kebutuhan sistem
  - b. Mencatat dan memperbaiki kekurangan - kekurangan yang ada sebelum diterapkan.
- 5) Penerapan  
Pada tahap ini adalah menggunakan sistem yang telah selesai dibuat. Langkah pada tahap penerapan adalah memastikan sistem berjalan dan data dapat berkomunikasi dengan baik sesuai harapan.

### 3. HASIL DAN PEMBAHASAN

Hasil penelitian ini adalah bisa menambahkan data.

- 1) Proses Tambah Data Mahasiswa

Dalam proses tambah data mahasiswa ini, kita memasukan data seperti Nama, Email, WhatsApp, Alamat, Angkatan, Divisi, Fakultas, Prodi. Proses tambah data mahasiswa ditunjukkan pada gambar 2.



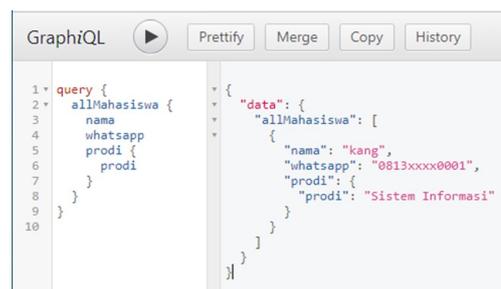
```
GraphQL
mutation {
  createMahasiswa(mahasiswaData: {
    prodi: 1,
    fakultas: 1,
    divisi: 1,
    angkatan: 2019,
    nama: "kang",
    email: "kang@example.com",
    whatsapp: "0813xxxx0001",
    alamat: "Indonesia"
  }) {
    mahasiswa {
      nama
      alamat
      email
      whatsapp
      divisi {
        divisi
      }
      fakultas {
        fakultas
      }
      prodi {
        prodi
      }
    }
  }
}
```

```
{
  "data": {
    "createMahasiswa": {
      "mahasiswa": {
        "nama": "kang",
        "alamat": "Indonesia",
        "email": "kang@example.com",
        "whatsapp": "0813xxxx0001",
        "divisi": {
          "divisi": "Pemrograman"
        },
        "fakultas": {
          "fakultas": "Matematika dan Ilmu Komputer"
        },
        "prodi": {
          "prodi": "Sistem Informasi"
        }
      }
    }
  }
}
```

Gambar 2. Proses Tambah Data Mahasiswa

## 2) Proses Lihat Data Semua Mahasiswa

Dalam proses lihat data mahasiswa ini, kita memasukan query yang ingin kita ketahui dari data semua mahasiswa seperti Nama, No. WhatsApp, Prodi, dll. Proses lihat data semua mahasiswa ditunjukkan pada gambar 3.



```
GraphQL
query {
  allMahasiswa {
    nama
    whatsapp
    prodi {
      prodi
    }
  }
}
```

```
{
  "data": {
    "allMahasiswa": [
      {
        "nama": "kang",
        "whatsapp": "0813xxxx0001",
        "prodi": {
          "prodi": "Sistem Informasi"
        }
      }
    ]
  }
}
```

Gambar 3. Proses Lihat Data Semua Mahasiswa

## 4. KESIMPULAN

### 4.1. Kesimpulan

- 1) Metode GraphQL dapat diterapkan kedalam perancangan sistem informasi berbasis API. Dalam penelitian ini, metode graphql berhasil diterapkan di sistem informasi UKM.
- 2) Pengujian sistem di penelitian ini mendapat keberhasilan menyeluruh pada pengujian yang direncanakan.

### 4.2. Saran

Agar fungsi *crud* (*create, read, update, delete*) bisa terlaksana, maka akan lebih baik jika menambahkan kategori untuk menghapus data dan mengupdate data, bukan hanya bisa untuk membuat data dan melihat data saja.

## DAFTAR PUSTAKA

- [1] Afriansyah, R., Sholeh, M., Andayati, D. (2021). Perancangan Aplikasi Pemrograman Antarmuka Berbasis Web Menggunakan Gaya Arsitektur Representasi Untuk Sistem Presensi Sekolah, 9(1), 84-93.
- [2] GraphQL 2022. A query language for your API. Diakses pada 22 Maret 2022, dari <https://graphql.org/>
- [3] Kumia, J. S. & Risyda, F. (2021). Rancang Bangun Penerapan Model Prototype Dalam Perancangan Sistem Informasi Pencatatan Persediaan Barang Berbasis Web, 9(2), 229.
- [4] Firdausi, A. T., Hormansyah, D. S., Ervansyah, F. (2021). Implementasi GraphQL Untuk Mengatasi Under-Fetching pada Pengembangan Sistem Informasi Pelacakan Alumni Politeknik Negeri Malang, 7(4), 74-75.