

**PERANCANGAN DAN OPTIMALISASI LAYANAN DATA GRAPHQL
PADA APLIKASI TRIPPERSPACE**

(Skripsi)



Oleh

**Joni Setiawan
1615061031**

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2023**

ABSTRAK

PERANCANGAN DAN OPTIMALISASI LAYANAN DATA GRAPHQL PADA APLIKASI TRIPPERSPACE

Oleh

Joni Setiawan

Industri pariwisata dan ekonomi kreatif Indonesia mengalami penurunan pada tahun 2020 dikarenakan pandemi COVID-19. Sejak Februari 2020 wisatawan mancanegara yang masuk ke Indonesia mengalami penurunan yang sangat drastis dan April 2020 adalah puncaknya dengan jumlah wisatawan hanya sebanyak 158 ribu. Untuk mendukung perkembangan industri pariwisata di Indonesia dikembangkan aplikasi mobile Tripperspace sebagai media informasi dan media sosial bagi wisatawan untuk berbagi informasi dan mengeksplorasi pariwisata yang ada di Indonesia. Aplikasi masih dalam tahap pengembangan prototipe interaktif yang digunakan untuk membuat representasi nyata dan mendekati produk final dengan menggunakan teknologi dan alat yang sama dengan produk akhir. Pada tahap pengembangan prototipe interaktif dibutuhkan layanan data yang akan menghubungkan aplikasi dengan basis data sehingga data pengguna dapat tersimpan dan terkelola. Pengoptimalan layanan data juga dilakukan untuk meningkatkan kinerja dan keandalan sistem. Penelitian dilakukan dengan Design Science Research Method (DSRM). Hasil penelitian menunjukkan bahwa sistem layanan data GraphQL memiliki sifat *versatile* dapat menangani perubahan kebutuhan data baru tanpa memperbarui keseluruhan selama data dalam entitas yang sudah didefinisikan. Sementara itu hasil perancangan sistem layanan data graphql untuk aplikasi Tripperspace menghasilkan 16 *Query* dan 3 *Mutation* yang dapat digunakan. Optimalisasi dilakukan dengan penggunaan *cache* dan pembatasan kedalaman *query* menunjukkan waktu respon server dari 37 detik menjadi 30 detik.

ABSTRACT

DESIGN AND OPTIMIZATION OF GRAPHQL DATA SERVICES IN THE TRIPPERSPACE APPLICATION

By

Joni Setiawan

Indonesia's tourism industry and creative economy experienced a decline in 2020 due to the COVID-19 pandemic. Since February 2020 foreign tourists entering Indonesia have experienced a very drastic decline and April 2020 was the peak with the number of tourists only 158 thousand. To support the development of the tourism industry in Indonesia, the Tripperspace mobile application has been developed as an information medium and social media for tourists to share information and explore tourism in Indonesia. The application is still in the interactive prototype development stage which is used to make a real representation and approximate the final product using the same technology and tools as the final product. At the interactive prototype development stage, a data service is needed that will connect the application to the database so that user data can be stored and managed. Optimization of data services is also carried out to improve system performance and reliability. The research was conducted using the Design Science Research Method (DSRM). The results show that the GraphQL data service system has a versatile nature that can handle changes in new data requirements without updating the entire data as long as the data is in an entity that has been defined. Meanwhile, the results of designing a GraphQL data service system for the Tripperspace application produce 16 Queries and 3 Mutations that can be used. Optimization is done by using cache and limiting query depth, showing server response time from 37 seconds to 30 seconds.

Keywords : *Application Programming Interface, GraphQL, Design Science Research Method*

**PERANCANGAN DAN OPTIMALISASI LAYANAN DATA GRAPHQL
PADA APLIKASI TRIPPERSPACE**

Oleh

Joni Setiawan

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA TEKNIK**

Pada

**Program Studi Teknik Informatika
Jurusan Teknik Elektro
Fakultas Teknik Universitas Lampung**



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2023**

**Judul Skripsi : PERANCANGAN DAN OPTIMALISASI
LAYANAN DATA GRAPHQL PADA
APLIKASI TRIPPERSPACE**

Nama Mahasiswa : Joni Setiawan

Nomor Pokok Mahasiswa : 1615061031

Program Studi : Teknik Informatika

Jurusan : Teknik Elektro

Fakultas : Teknik



1. Komisi Pembimbing

Wahyu Eko Sulistiono, S.T., M.Sc.

Ir. Meizano Ardhi Muhammad, S.T., M.T.

NIP 19741201200112 1 001

NIP 19810528201212 1 001

2. Mengetahui

**Ketua Jurusan
Teknik Elektro**

**Ketua Program Studi
Teknik Informatika**

Herlinawati, S.T., M.T.

Mona Arif Muda, S.T., M.T.

NIP 19710314 199903 2 001

NIP 19711112 200003 1 002

MENGESAHKAN

1. Tim Penguji

Ketua : Wahyu Eko Sulistiono, S.T., M.Sc.

Sekretaris : Ir. Meizano Ardhi Muhammad, S.T., M.T.

Penguji : Dr. Eng. Ir. Mardiana, S.T., M.T., IPM.

2. Dekan Fakultas Teknik

Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc.

NIP 19750928 200112 1 002

Tanggal Lulus Ujian Skripsi : 05 Mei 2023



SURAT PERNYATAAN

Saya yang bertandatangan dibawah ini, menyatakan bahwa skripsi yang berjudul “Perancangan Dan Optimalisasi Layanan Data GraphQL Pada Aplikasi Tripperspace” dibuat oleh saya sendiri. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung.

Apabila di kemudian hari terbukti bahwa skripsi ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung, 5 Mei 2023

Yang membuat pernyataan,



Joni Setiawan

NPM. 1615061031

RIWAYAT HIDUP



Penulis dilahirkan di Buluwangi, 17 Januari 1998. Penulis merupakan anak pertama dari tiga bersaudara dari pasangan Bapak Sukardi dan Ibu Tumiaty. Penulis diterima di Program Studi Teknik Informatika Universitas Lampung pada tahun 2016.

Selama menjadi mahasiswa penulis aktif menjadi *Freelance Engineer*. Beberapa kali terlibat dalam proyek besar diantaranya pengembangan HAMS (*Human & Asset Management System*) dan SOP (*Standard Operational Procedure*) di PT. Rindang 31 sebagai *Operation Engineer*, pengembangan SISTER (Sistem Informasi Sumberdaya Terintegrasi) di Universitas Lampung sebagai *Technical Writer*, pengembangan Aplikasi Koran Mobile di Radar Lampung sebagai *Backend Programmer* dan beberapa proyek lain yang tidak bisa penulis sebutkan.

Pada tahun 2018 penulis dan teman-teman mendirikan sebuah komunitas dengan nama Gradien. Tujuan dari pembentukan komunitas Gradien adalah memberikan wadah bagi mahasiswa yang ingin belajar mengenai dunia teknologi.

Kupersembahkan karyaku ini untuk orang-orang yang telah memberikanku doa dan semangat,

Untuk kedua orang tuaku tercinta, yang tak pernah henti memberi doa dan dukungan.

Seluruh Dosen dan teman – teman seperjuangan Teknik Informatika Universitas Lampung yang turut membantuku, Serta Universitas Lampung sebagai tempat menuntut ilmuku selama masa perkuliahan ini.

Terima kasih.

SANWACANA

Puji syukur kehadirat Allah SWT. yang telah melimpahkan rahmat serta hidayah-Nya kepada penulis sehingga dapat menyelesaikan penyusunan tugas akhir ini dengan judul “Perancangan dan Optimalisasi Layanan Data GraphQL Pada Aplikasi Tripperspace”. Shalawat serta salam senantiasa tercurahkan kepada Nabi Muhammad SAW.

Dalam pelaksanaan penelitian dan penyusunan skripsi ini penulis menerima banyak sekali dukungan baik secara moril maupun materil dari berbagai pihak. Oleh karena itu, penulis ingin mengucapkan terimakasih kepada seluruh pihak yang telah membantu, khususnya kepada :

1. Allah SWT Tuhan yang maha kuasa berkat rahmat dan karunia-Nya di dalam kehidupan.
2. Herlinawati, S.T., M.T. Selaku Ketua Jurusan Teknik Elektro, Universitas Lampung;
3. Bapak Mona Arif Muda, S.T., M.T. Selaku Ketua Program Studi Teknik Infomatika.
4. Bapak Wahyu Eko Sulistiono, S.T., M.Sc. Selaku dosen pembimbing utama yang telah memberikan kesempatan dan pembelajaran yang akan selalu saya kenang.

5. Bapak Ir. Meizano Ardhi Muhammad, S.T., M.T., selaku Dosen Pembimbing kedua atas kesediaan waktunya dalam memberikan saran, bimbingan dan arahan;
6. Ibu Dr. Eng. Ir. Mardiana, S.T., M.T., IPM selaku dosen penguji yang membantu memberikan saran dan arahan pada penyelesaian penelitian ini.
7. Bapak dan Ibu Dosen beserta Civitas Akademika Jurusan Teknik Elektro dan Informatika yang telah mendidik, membimbing, dan memberikan ilmu pengetahuan kepada saya.
8. Kedua orang tua tercinta, pemberi motivasi terbesar, yang selalu memberikan doa dan dukungan dalam hidup saya.
9. Rekan-rekan PSTI angkatan 2016 yang tidak dapat saya sebutkan satu persatu, selaku teman angkatan yang menemani masa perkuliahan saya.

Bandar Lampung, 5 Mei 2023
Penulis

Joni Setiawan
1615061031

DAFTAR ISI

	Halaman
ABSTRAK.....	ii
DAFTAR ISI.....	xii
DAFTAR GAMBAR.....	xv
DAFTAR TABEL	xvii
I. PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	5
1.3. Tujuan Penelitian	5
1.4. Manfaat Penelitian	5
1.5. Batasan Masalah	6
1.6. Sistematika Penulisan	6
II. TINJAUAN PUSTAKA.....	8
2.1. Tripperspace.....	8
2.2. Application Programming Interface (API).....	9
2.3. GraphQL	10
2.4. Basis Data	11
2.5. PostgreSQL	13
2.6. PHP	13
2.7. Laravel	13
2.8. Penelitian Terkait	14
III. METODE PENELITIAN	16

3.1.	Waktu dan Tempat Penelitian	16
3.2.	Alat dan Bahan.....	16
3.3.	Tahap Penelitian.....	16
3.3.1.	Identifikasi Masalah.....	17
3.3.2.	Objektif dari Solusi.....	18
3.3.3.	Desain & Pengembangan.....	18
3.3.4.	Demonstrasi	18
3.3.5.	Evaluasi.....	19
3.3.6.	Komunikasi	19
IV.	HASIL DAN PEMBAHASAN.....	20
4.1.	Identifikasi Masalah dan Objektif dari Solusi.....	20
4.2.	Desain dan Pengembangan	27
4.2.1.	Schema dan Resolver	28
4.2.2.	Query dan Mutation	38
4.3.	Demonstrasi	41
4.3.1.	Data User	41
4.3.2.	Data Provinsi.....	43
4.3.3.	Data Kabupaten.....	44
4.3.4.	Data Kecamatan	45
4.3.5.	Data Desa.....	46
4.3.6.	Data Post	47
4.3.7.	Data Lokawisata.....	49
4.4.	Evaluasi.....	50
4.4.1.	Waktu Respon dan <i>Error Rate Query</i>	50
4.4.2.	Penggunaan <i>Cache</i> Pada Respon <i>Query GraphQL</i>	52
4.4.3.	Perbedaan Relasi Tabel Database Terhadap Kecepatan Respon	54
4.5.	Komunikasi.....	56
V.	SIMPULAN DAN SARAN	57

5.1. Simpulan	57
5.2. Saran	58
DAFTAR PUSTAKA.....	59

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Tripperspace.....	8
Gambar 2.2 GraphQL.....	11
Gambar 2.3 PostgreSQL.....	13
Gambar 3.1 <i>Design Science Research Process (DSRP) model</i>	17
Gambar 4.1 <i>Empathy Observation Worksheet</i>	21
Gambar 4.2 <i>Entity User</i>	22
Gambar 4.3 <i>Entity Lokawisata</i>	22
Gambar 4.4 <i>Entity Paket Wisata</i>	23
Gambar 4.5 <i>Entity Desa</i>	23
Gambar 4.6 <i>Entity Kecamatan</i>	24
Gambar 4.7 <i>Entity Kabupaten</i>	24
Gambar 4.8 <i>Entity Provinsi</i>	25
Gambar 4.9 <i>Entity Post</i>	25
Gambar 4.10 <i>Entity Order</i>	26
Gambar 4.11 <i>Entity Relationship Diagram (ERD)</i>	26
Gambar 4.12 <i>Desain Sistem</i>	27
Gambar 4.13 <i>Schema Data User</i>	28
Gambar 4.14 <i>Resolver Data User</i>	28
Gambar 4.15 <i>Schema Data Lokawisata</i>	29
Gambar 4.16 <i>Resolver Data Lokawisata</i>	30
Gambar 4.17 <i>Schema Data Post</i>	30
Gambar 4.18 <i>Resolver Data Post</i>	31
Gambar 4.19 <i>Schema Data Provinsi</i>	31
Gambar 4.20 <i>Resolver Data Provinsi</i>	32
Gambar 4.21 <i>Schema Data Kabupaten</i>	33
Gambar 4.22 <i>Resolver Data Kabupaten</i>	33

Gambar 4.23 <i>Schema</i> Data Kecamatan.....	34
Gambar 4.24 <i>Resolver</i> Data Kecamatan	35
Gambar 4.25 <i>Schema</i> Data Desa.....	36
Gambar 4.26 <i>Resolver</i> Data Desa	36
Gambar 4.27 <i>Schema</i> Data Paket Wisata.....	37
Gambar 4.28 <i>Resolver</i> Data Paket Wisata.....	37
Gambar 4.29 Query dan Mutation	38
Gambar 4.30 Contoh <i>request</i> untuk <i>create user</i>	41
Gambar 4.31 Hasil respon <i>request create user</i>	42
Gambar 4.32 Contoh <i>request</i> data <i>users</i>	42
Gambar 4.33 Potongan respon <i>request</i> data <i>users</i>	43
Gambar 4.34 Contoh <i>request</i> data provinsi	43
Gambar 4.35 Potongan respon <i>request</i> data provinsi.....	44
Gambar 4.36 Contoh <i>request</i> data kabupaten	44
Gambar 4.37 Potongan respon <i>request</i> data kabupaten	45
Gambar 4.38 Contoh <i>request</i> data kecamatan	45
Gambar 4.39 Potongan respon <i>request</i> data kecamatan.....	46
Gambar 4.40 Contoh <i>request</i> data desa	46
Gambar 4.41 Potongan respon <i>request</i> data desa.....	47
Gambar 4.42 Contoh <i>request</i> data post.....	47
Gambar 4.43 Potongan respon <i>request</i> data post.....	48
Gambar 4.44 Contoh <i>request</i> data lokawisata	49
Gambar 4.45 Potongan respon <i>request</i> data lokawisata.....	50
Gambar 4.46 <i>Postman Runner</i>	51
Gambar 4.47 Hasil <i>Postman Runner</i>	51
Gambar 4.48 Query data Desa	52
Gambar 4.49 Hasil <i>query</i> data desa tanpa <i>cache</i>	53
Gambar 4.50 Hasil <i>query</i> data desa dengan <i>cache</i>	53
Gambar 4.51 <i>Query</i> data lokawisata dengan <i>post</i>	54
Gambar 4.52 Hasil <i>query</i> data lokawisata dengan post tanpa relasi database	54
Gambar 4.53 Hasil <i>query</i> data lokawisata dengan post dengan relasi <i>one to many</i>	55
Gambar 4.54 Hasil <i>query</i> data lokawisata dengan post dengan relasi <i>many to many</i>	55

DAFTAR TABEL

	Halaman
Tabel 1 Alat – Alat Penelitian.....	16
Tabel 2 Bahan - Bahan Penelitian.....	16
Tabel 3 Query dan Mutation	39

I. PENDAHULUAN

1.1. Latar Belakang

Program Magang Bersertifikat dan Studi Independen Bersertifikat (MSIB) Merdeka Belajar Kampus Merdeka (MBKM) merupakan bentuk pembelajaran perguruan tinggi mandiri, dan fleksibel sehingga dapat menciptakan budaya belajar yang inovatif dan sesuai dengan kebutuhan mahasiswa. Dengan adanya Program Magang Bersertifikat dan Studi Independen Bersertifikat (MSIB) Merdeka Belajar Kampus Merdeka (MBKM), diharapkan mahasiswa dapat termotivasi untuk berpartisipasi dalam pembangunan nasional. PT Cipta Konsultan Internasional meluncurkan sebuah program pembelajaran berbasis proyek yaitu *Agile Innovation Project-based Learning* (AIPBL). Program *Agile Innovation Project-based Learning* (AIPBL) bertujuan untuk memampukan mahasiswa mengembangkan solusi inovatif untuk memecahkan permasalahan kompleks yang ditemukan di lingkungannya. Dalam program pembelajaran berbasis proyek ini, mahasiswa akan diberikan proyek atau tantangan kelompok (5 orang) untuk memecahkan satu permasalahan yang ada di lingkungannya dengan mengembangkan sebuah solusi atau produk inovatif yang *desirable* atau diinginkan pengguna (pemilik masalahnya), *feasible* atau mampu diimplementasikan dengan baik, dan *viable* atau menguntungkan dari segi bisnis atau finansial [1].

Pada program Magang Bersertifikat dan Studi Independen Bersertifikat (MSIB) Merdeka Belajar Kampus Merdeka (MBKM) terdapat tugas proyek besar dimana harus membuat inovasi berdasarkan permasalahan yang terjadi. Permasalahan yang diangkat adalah sektor pariwisata yang ada di Indonesia. Industri pariwisata dan ekonomi kreatif Indonesia mengalami penurunan pada tahun 2020 dikarenakan pandemi COVID-19. Sejak Februari 2020 wisatawan mancanegara yang masuk ke Indonesia mengalami penurunan yang sangat drastis dan April 2020 adalah puncaknya dengan jumlah wisatawan hanya sebanyak 158 ribu. Sepanjang 2020 total dari jumlah wisatawan mancanegara yang berkunjung ke Indonesia hanya sekitar 4,052 juta wisatawan. Total tersebut hanya sekitar 25% dari total wisatawan mancanegara yang berkunjung pada tahun 2019. Dampak pandemi COVID-19 terhadap rencana bepergian 82% orang yang berencana bepergian dalam 6 bulan terakhir mengubah rencana. Dalam kelompok tersebut, pelancong menyesuaikan rencana mereka. 50% membatalkan kepergian, 11% mengubah dari internasional menjadi domestik, 45% mengurangi penerbangan ke jalur darat, dan 22% mengubah dari penerbangan ke jalur darat [2]. Untuk mendukung perkembangan industri pariwisata di Indonesia dikembangkan aplikasi mobile Tripperspace sebagai media informasi dan media sosial bagi wisatawan untuk berbagi informasi dan mengeksplorasi pariwisata yang ada di Indonesia. Dengan adanya Tripperspace diharapkan dapat mempermudah mengakses informasi pariwisata secara terpusat tanpa harus mencari di laman pencarian Google atau di media sosial seperti Facebook atau Instagram. Pengembangan dilakukan dalam beberapa tahap. Tahap pengembangan Prototipe Statik yang digunakan untuk membuat representasi visual dari produk atau sistem, namun tidak memiliki fungsionalitas yang lengkap. Dalam

Prototipe Statik, desain dan tampilan visual produk atau sistem diwakili oleh gambar statis atau mockup. Dan tahap pengembangan Prototipe Interaktif yang digunakan untuk membuat representasi nyata dan mendekati produk final dengan menggunakan teknologi dan alat yang sama dengan produk akhir. Prototipe Interaktif memiliki tingkat detail dan kompleksitas yang tinggi, termasuk fungsionalitas, fitur, tampilan visual, interaksi, dan pengalaman pengguna yang lengkap. Lalu setelah itu tahap perilis secara publik jika pengujian dari tahap-tahap sebelumnya sudah memenuhi kebutuhan yang sudah ditentukan.

Pada tahap pengembangan prototipe interaktif dibutuhkan layanan data yang akan menghubungkan aplikasi dengan basis data sehingga data pengguna dapat tersimpan dan dikelola. Layanan data yang biasa digunakan saat ini adalah REST API. Tetapi layanan data seperti ini tidak fleksibel dan butuh waktu pengembangan yang cukup lama sehingga dapat menghambat pengembangan prototipe Tripperspace. Oleh karena itu, untuk mencegah hal tersebut, maka dibutuhkan layanan data yang fleksibel dan dapat dikerjakan dalam waktu yang lebih singkat.

Kebutuhan layanan data yang fleksibel dan dapat dikerjakan dalam waktu yang singkat dapat diatasi dengan GraphQL. Fleksibilitas dari cara kerja GraphQL dapat mengatasi beberapa kebutuhan sekaligus dibandingkan dengan REST (*Representational State Transfer*) yang harus membuat satu-persatu *endpoint* untuk setiap kebutuhan data[3]. Salah satu permasalahan REST API adalah *Under-fetching* dan *over-fetching*. *Under-fetching* terjadi ketika klien harus melakukan beberapa permintaan untuk mendapatkan semua data yang dibutuhkan, sedangkan *over-fetching* terjadi ketika klien meminta lebih banyak data dari yang dibutuhkan.

Dalam kedua kasus tersebut, terdapat pemborosan sumber daya. GraphQL dirancang untuk mengatasi masalah *under-fetching* dan *over-fetching*. GraphQL memungkinkan klien untuk meminta hanya data yang dibutuhkan dan memungkinkan server mengembalikan hanya data yang diminta. GraphQL juga memungkinkan klien untuk menentukan struktur data yang diinginkan, sehingga memungkinkan pengembang layanan data untuk mengubah struktur data tanpa memengaruhi klien. Hal tersebut dibutuhkan aplikasi Tripperspace karena aplikasi masih dalam tahap prototipe sehingga ada kemungkinan kebutuhan data akan terus berubah. Dengan fleksibilitas yang dimiliki GraphQL memudahkan pengembang layanan data yang mudah dipelihara, karena pengembang hanya perlu menentukan tipe data yang didukung oleh layanan data dan memungkinkan klien untuk menentukan data yang diinginkan. Kelebihan lain dari GraphQL adalah fitur *Subscription* dimana klien memungkinkan untuk mengambil data secara *real-time* dari server GraphQL. Klien dapat menerima pembaruan data otomatis tanpa perlu melakukan permintaan baru[4].

Pengoptimalan layanan data juga dilakukan untuk meningkatkan kinerja dan keandalan sistem. Pengoptimalan performa dapat membantu mengurangi waktu respon server sehingga pengguna dapat memperoleh hasil yang diinginkan dengan lebih cepat. Layanan data dapat melayani lebih banyak permintaan dari pengguna secara bersamaan, sehingga meningkatkan skalabilitas sistem. Dengan meningkatkan efisiensi kinerja layanan data, biaya operasional yang terkait dengan pemeliharaan dan pengoperasian server dapat dikurangi. Pengoptimalan performa juga dapat membantu meningkatkan keandalan server dengan memastikan bahwa

layanan data dapat beroperasi secara konsisten dan dapat diakses oleh pengguna dengan baik. Optimalisasi dilakukan dengan penggunaan *cache* pada hasil permintaan data yang sama secara berulang sehingga akan mengurangi beban server untuk melakukan pengambilan data ke database. Selain itu pembatasan kedalaman *query* dilakukan untuk membatasi permintaan yang sangat dalam dan rumit sehingga membebani dan menghambat kinerja server. Batasan kedalaman *query* juga dapat membantu mendorong penggunaan GraphQL yang lebih baik dan terstruktur. Dengan membatasi kedalaman *query*, pengembang dan klien dipaksa untuk memikirkan secara lebih cermat tentang struktur data dan bagaimana data akan diminta dan dikembalikan.

1.2. Rumusan Masalah

Rumusan masalah penelitian adalah bagaimana perancangan dan optimalisasi sistem layanan data dengan *Case Tools* GraphQL dan PostgreSQL.

1.3. Tujuan Penelitian

Tujuan penelitian adalah merancang dan mengoptimisasi layanan data GraphQL untuk aplikasi Tripperspace.

1.4. Manfaat Penelitian

Manfaat penelitian adalah sebagai berikut :

1. Melalui perancangan yang efisien pengembangan aplikasi Tripperspace akan lebih fleksibel.
2. Optimalisasi layanan data dapat meningkatkan kinerja dan keandalan sistem. Pengoptimalan performa dapat membantu mengurangi waktu respon server

sehingga pengguna dapat memperoleh hasil yang diinginkan dengan lebih cepat.

1.5. Batasan Masalah

1. Dalam penelitian hanya berfokus pada perancangan dan optimalisasi sistem layanan data GraphQL.
2. Bidang penelitian hanya menggunakan data-data yang diperoleh saat kegiatan Kampus Merdeka Studi Independen di Corporate Innovation Asia (CIAS) dan data *dummy* untuk kebutuhan pengujian.

1.6. Sistematika Penulisan

Dalam pembahasan laporan penelitian ini, sistematika penulisan yang digunakan adalah sebagai berikut :

I. PENDAHULUAN

Pada Bab Pendahuluan ini berisi tentang latar belakang penelitian pengembangan aplikasi, rumusan masalah penelitian, tujuan penelitian, manfaat penelitian, batasan masalah dan sistematika penulisan penelitian.

II. TINJAUAN PUSTAKA DAN LANDASAN TEORI

Pada Bab ini berisi tentang penelitian serupa yang telah dilakukan sebelumnya, pengetahuan tentang basis data, ERD (*Entity Relationship Diagram*), *Conceptual Data Model*, *Physical Data Model*, PostgreSQL, dan API (*Application Programming Interface*).

III. METODE PENELITIAN

Pada Bab ini berisi tentang mengenai metode yang digunakan yaitu *Design-Science Research Methode* (DSRM).

IV. PEMBAHASAN

Pada bab ini menjelaskan tentang hasil pengujian Perancangan dan Optimalisasi Layanan Data GraphQL.

V. PENUTUP

Pada Bab ini berisi tentang kesimpulan dari Perancangan dan Optimalisasi Layanan Data GraphQL dan saran pengembang hasil penelitian lanjutan untu hasil yang lebih baik lagi.

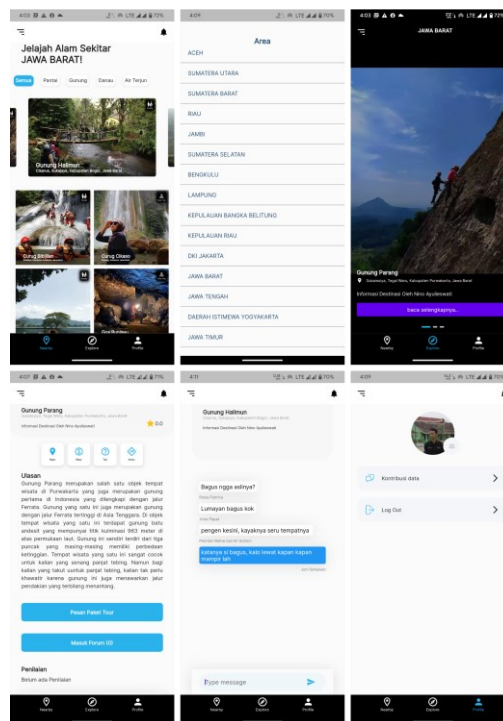
DAFTAR PUSTAKA

LAMPIRAN

II. TINJAUAN PUSTAKA

2.1. Tripperspace

Tripperspace adalah platform yang menyediakan informasi seputar wisata alam di Indonesia dari wisatawan untuk wisatawan agar mendapatkan kemudahan dan kenyamanan saat mengakses informasi destinasi wisata alam di Indonesia. Tripperspace dikembangkan berdasarkan penelitian yang dilakukan pada program Magang Bersertifikat dan Studi Independen Bersertifikat (MSIB) Merdeka Belajar Kampus Merdeka (MBKM).



Gambar 2.1 Tripperspace

Tripperspace masih dalam tahap pengembangan prototipe yang dikembangkan menggunakan *framework* Flutter dengan bahasa pemrograman Dart dan masih menggunakan data statis dan belum memiliki layanan data. Fitur yang dikembangkan adalah sebagai berikut :

1. “*Nearby*”, fitur ini menampilkan rekomendasi tempat wisata berdasarkan provinsi yang di pilih. Data yang ditampilkan juga dapat difilter berdasarkan kategori.
2. “*Post*”, fitur ini menampilkan konten yang dibagikan oleh pengguna lain.
3. “Ulasan Tempat Wisata”, fitur ini menampilkan informasi dan ulasan dari pengguna yang pernah mengunjungi tempat wisata yang di pilih pada halaman explore. Terdapat informasi titik koordinat pada peta digital, biaya yang dikeluarkan, informasi akses jalan dan tips perjalanan.

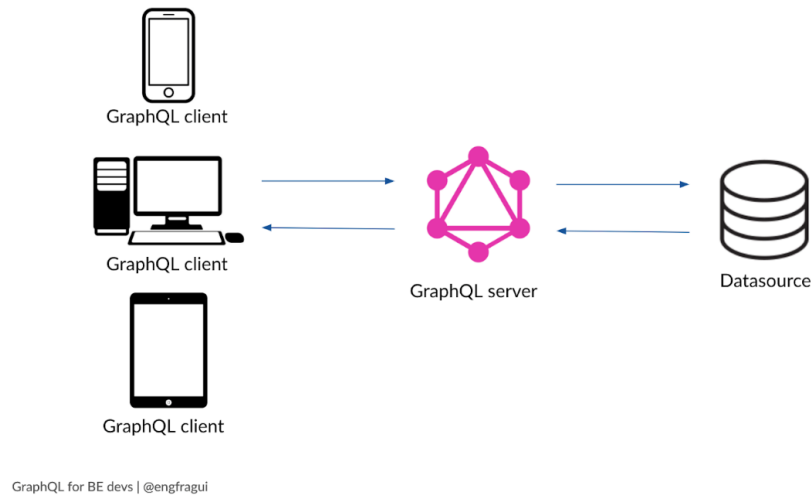
2.2.Application Programming Interface (API)

Application Programming Interface (API) atau Antarmuka Pemrograman Aplikasi merupakan sekumpulan fungsi, perintah dan protokol yang dapat digunakan saat membangun perangkat lunak untuk sistem operasi tertentu. Memungkinkan pengembang untuk mengintegrasikan dua bagian dari aplikasi atau dengan aplikasi yang berbeda secara bersamaan[5]. Tujuan penggunaan API adalah untuk mempercepat proses pengembangan dengan menyediakan fungsi secara terpisah sehingga pengembang tidak perlu membuat fitur yang serupa. Penerapan API akan sangat terasa jika fitur yang diinginkan sudah sangat kompleks, tentu membutuhkan waktu untuk membuat yang serupa dengannya.

2.3. GraphQL

Setelah pengembangan dan menggunakannya secara internal selama tiga tahun, pada tahun 2016 Facebook merilis GraphQL. *Framework* GraphQL memperkenalkan tipe baru dari antarmuka akses data berbasis Web yang menyajikan alternatif untuk konsep antarmuka berbasis REST (*Representational State Transfer*). Sejak dirilis, GraphQL memperoleh momentum yang signifikan dan telah diadopsi oleh semakin banyak pengguna. GraphQL adalah *query language* untuk API (*Application Programming Interface*) dan waktu proses untuk memenuhi *query* tersebut dengan data yang sudah ada. Dalam pengembangan sistem layanan data GraphQL terdapat elemen-elemen penting yaitu *Schema*, *Resolver*, *Query* dan *Mutation*. *Schema* merupakan struktur data dari GraphQL yang menentukan tipe data apa saja yang tersedia di server, serta bagaimana tipe data tersebut terhubung satu sama lain. *Schema* juga menentukan bagaimana permintaan *query* dan *mutation* akan diproses oleh server. *Resolver* adalah sebuah fungsi yang digunakan dalam GraphQL untuk mengolah permintaan *query* dan *mutation* yang dikirimkan ke server. Setiap permintaan *query* atau *mutation* harus memiliki sebuah *resolver* yang menentukan bagaimana permintaan tersebut akan diproses oleh server. *Resolver* memiliki akses ke sumber data yang mendasari, seperti database atau API lain, dan menggunakan sumber data tersebut untuk mengambil atau memperbarui data yang diinginkan. *Resolver* juga dapat digunakan untuk melakukan validasi terhadap data yang dikirimkan ke server melalui *mutation*, atau mengolah data yang diterima dari server sebelum dikirim ke *client*. *Query* merupakan permintaan yang dikirimkan ke server untuk mengambil data dari server. *Query* terdiri dari nama tipe data yang diinginkan, serta properti apa saja

yang ingin diambil dari tipe data tersebut. *Mutation* merupakan permintaan yang dikirimkan ke server untuk mengubah data di server, seperti menambah, mengubah, atau menghapus data. *Mutation* juga dapat digunakan untuk mengirimkan data ke server dan memperbarui data di server[6].



Gambar 2.2 Graphql

2.4. Basis Data

Basis data merupakan kumpulan data yang menjabarkan suatu aktivitas dari satu atau beberapa entitas yang berhubungan. Sebagai contoh basis data sebuah universitas memiliki entitas seperti mahasiswa, staf pengajar, mata kuliah, dan ruang kelas. Hubungan antara entitas tersebut seperti mahasiswa yang mendaftar untuk mengikuti mata kuliah, staf pengajar yang mengajarkan mata kuliah, dan penggunaan ruang kelas untuk suatu mata kuliah.

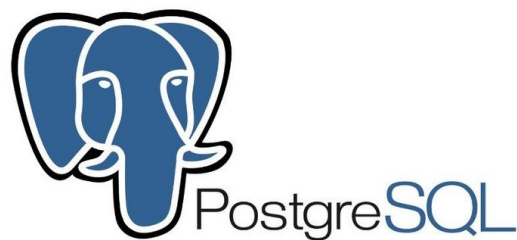
Sistem manajemen basis data diolah melalui dua bahasa yaitu DDL (*Data Definition Language*) dan DML (*Data Manipulation Language*). DDL merupakan bahasa yang digunakan untuk menentukan skema basis data sesuai dengan

kumpulan ekspresi DDL. Perintah DDL di kompilasi, dan menghasilkan kumpulan tabel yang tersimpan pada sebuah berkas yang disebut data dictionary yang juga berisi metadata. Struktur dan metode akses yang digunakan oleh sistem database. Sedangkan DML merupakan bahasa yang digunakan untuk melakukan manipulasi terhadap database seperti melakukan pengambilan data, memasukkan data, menghapus data, dan memodifikasi data, tujuannya adalah mempermudah manusia untuk berinteraksi dengan sistem. Terdapat 2 tipe DML yakni procedural dimana pengguna menentukan data apa yang diperlukan dan bagaimana data tersebut didapat dan nonprocedural saat pengguna hanya menentukan data apa yang diperlukan. Sebuah bahasa kueri adalah bagian dari DML yang hanya melibatkan pengambilan informasi. Aturan DML dan bahasa kueri biasanya hampir sama.

Database Life Cycle merupakan tahapan atau siklus proses dibuat database. Proses pertama merupakan *system definition* yaitu penjelasan batasan-batasan dan cakupan database dari sudut pandang *user*. Kemudian desain database, pada tahap ini menerjemahkan analisa kebutuhan *user* ke dalam bentuk rancangan. Tahap selanjutnya adalah implementasi database, pada tahap ini dilakukan pembuatan konseptual database untuk menentukan tabel-tabel pada database dengan sebuah *tools*. Kemudian konversi aplikasi, dilakukan konversi konseptual database sebelumnya kedalam sistem yang baru (pembuatan database nya). Lalu dilakukan pengujian dan validasi, pada tahap ini dilakukan pengujian database dengan tujuan memeriksa fungsi database dan menemukan kesalahannya. Setelah dilakukan pengujian, maka dapat melanjutkan tahap pengoperasian dan pengawasan (pemeliharaan)[7].

2.5. PostgreSQL

PostgreSQL merupakan sebuah *Obyek-Relasional Data Base Management System* (ORDBMS) yang dikembangkan oleh Berkeley Computer Science Departement. PostgreSQL menawarkan kebutuhan aplikasi data, tambahan-tambahan yang cukup signifikan seperti *class, inheritance, type, dan function*. Tambahan keistimewaan lain yang tidak dimiliki *database management system* yang lain berupa *constraint, triggers, rule, dan transaction integrity*, dengan adanya *feature* (keistimewaan) tersebut maka para pemakai dapat dengan mudah mengimplementasikan dan menyampaikan sistem ini[8].



Gambar 2.3 PostgreSQL

2.6. PHP

PHP adalah bahasa pemrograman sederhana namun kuat yang dirancang untuk membuat konten HTML (*Hypertext Markup Language*). PHP dapat berjalan pada banyak sistem operasi, dari operasi sistem Linux, sampai Windows dan MacOS. Reamus Lerdorf menyusun PHP pertama pada 1994[9].

2.7.Laravel

Laravel merupakan sebuah framework PHP yang menggunakan konsep MVC (*Model View Controller*). Laravel dirancang untuk meningkatkan kualitas

perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, serta meningkatkan pengalaman bekerja dengan aplikasi yang menyediakan sintaks yang ekspresif, jelas, dan hemat waktu. Selain itu, laravel juga menyediakan web server lokal sehingga para pengembang web dapat melihat tampilan dari aplikasi web yang dibuatnya secara langsung[10].

2.8. Penelitian Terkait

Penelitian terkait dengan penelitian ini adalah sebagai berikut :

1. Penelitian yang berjudul Pemanfaatan Sistem Informasi Geografi dalam Pengembangan Pariwisata oleh Riwayatiningih dan Hartuti Purnaweni. Penelitian ini membahas tentang pemanfaatan sistem informasi geografi yang berperan penting dalam pengembangan pariwisata. Dengan adanya sistem informasi geografi, pengguna dapat memahami dan menganalisa dengan mudah dalam bentuk grafik atau peta [11]. Pada penelitian ini terdapat kemiripan dimana pemanfaatan sistem informasi dalam pengembangan pariwisata sehingga menjadi referensi dalam penentuan data yang akan dikelola pada penelitian ini.
2. Penelitian yang berjudul Analisis Dan Perancangan Sistem Informasi Pariwisata Berbasis Web Sebagai Media Promosi Pada Kabupaten Tebo oleh Raysa Puteri Ardhiyani dan Herry Mulyono. Penelitian ini membahas tentang perancangan sistem informasi di kabupaten Tebo yang akan dimanfaatkan sebagai media promosi pariwisata yang ada di kabupaten Tebo [12]. Pada penelitian ini terdapat kemiripan pada proses perancangan sistem.

3. Penelitian yang berjudul *Migrating to GraphQL: A Practical Assessment* oleh Gleison Brito, Thais Mombach, dan Marco Tulio Valente. Penelitian ini membahas mengenai migrasi API ke GraphQL. Penelitian menunjukkan bahwa GraphQL dapat mengurangi ukuran dokumen JSON yang dikembalikan oleh REST API sebesar 94% (dalam jumlah field/bidang) dan 99% (dalam jumlah *byte*), keduanya merupakan hasil median. Ketika dibandingkan hasil dari respon REST sebesar 9.8 MB sedangkan dengan GraphQL menjadi 86 KB[13]. Penelitian ini menjadi referensi dalam optimalisasi layanan data yang dikembangkan.

III. METODE PENELITIAN

3.1. Waktu dan Tempat Penelitian

Waktu dan tempat penelitian ini adalah:

Waktu Penelitian : September 2021 – Februari 2022

Tempat Penelitian : Universitas Lampung & PT Cipta Konsultan Internasional

3.2. Alat dan Bahan

Alat yang digunakan dalam penelitian ini dapat dilihat pada Tabel 1.

Tabel 1 Alat – Alat Penelitian

No	Perangkat	Spesifikasi	Kegunaan
1	Laptop	AMD Athlon 300U, RAM 8GB, Windows 10 Pro	Perangkat pembuatan dan pengujian sistem
3	VS Code	VS Code 1.78	<i>Text Editor</i> untuk menulis kode program
5	Postman	Postman v10.13	Tools untuk pengujian API

Bahan yang digunakan dalam penelitian ini dapat dilihat pada Tabel 2.

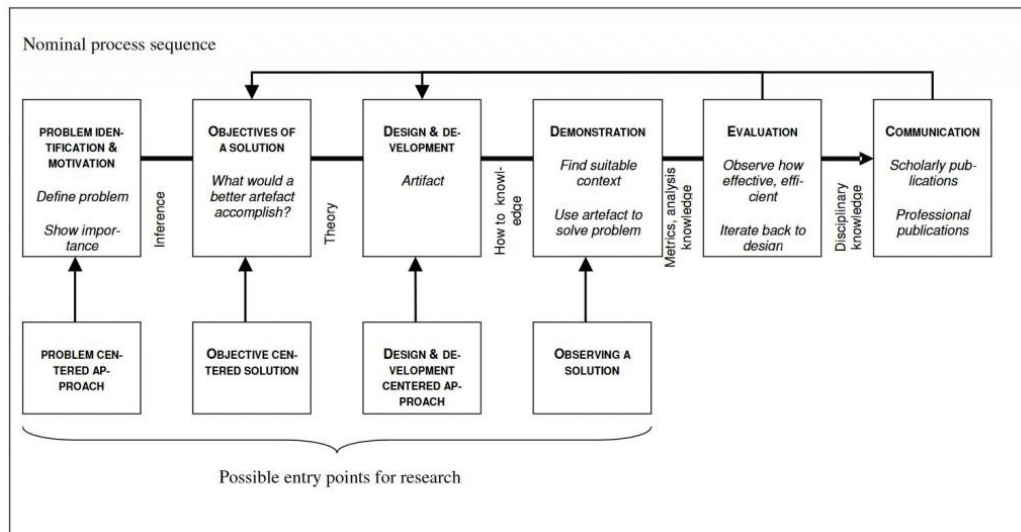
Tabel 2 Bahan - Bahan Penelitian

No	Nama	Keterangan
1	Data Wawancara Pengguna	Data-data yang diperoleh saat kegiatan Kampus Merdeka Studi Independen di Corporate Innovation Asia (CIAS)
3	Data Dummy	Data-data palsu yang digunakan untuk proses pengujian sistem.

3.3. Tahap Penelitian

Metode yang digunakan pada penelitian ini adalah *Design-Science Research Methode* (DSRM). DSRM merupakan metode penelitian yang bertujuan untuk

membuat produk baru dan mengujinya apakah produk berguna sesuai tujuannya. DSRM akan memandu peneliti ke dalam enam buah tahapan untuk mendesain, merancang, menguji, mendemokan ke pengguna dan pembuatan dokumentasi untuk sebuah sistem layanan data [14] [15].



Gambar 3.1 *Design Science Research Process (DSRP) model*

3.3.1. Identifikasi Masalah

Tahap pertama, Identifikasi Masalah (*Problem identification*). Bertujuan untuk mengidentifikasi permasalahan yang akan dipecahkan dan nilai pentingnya solusi dari permasalahan lalu membuat latar belakang dari masalah tersebut. Identifikasi dan analisa dilakukan untuk mengetahui kebutuhan dan harapan terhadap solusi yang akan dikembangkan. Pada penelitian ini identifikasi permasalahan dilakukan dengan cara studi literatur yang berhubungan dengan konsep layanan data dan bagaimana proses optimalisasi yang baik untuk diterapkan dari berbagai sumber yang berkaitan, juga dilakukan dengan wawancara dan survey lapangan sampai pada akhirnya dapat merumuskan permasalahan yang ada dilapangan.

3.3.2. Objektif dari Solusi

Tahap kedua, Objektif dari solusi (*Objectives of a solution*). Ketika latar belakang sudah teridentifikasi selanjutnya ditetapkan tujuan yang ingin dicapai dari solusi yang di buat. Solusi yang baru diharapkan mendukung solusi yang belum terpecahkan sebelumnya. Menetapkan tutujuan yang jelas untuk solusi yang akan dikembangkan bertujuan agar dapat diukur apakah solusi tersebut sukses atau tidak. Solusi dikembangkan berdasarkan hasil dari identifikasi permasalahan yang sudah dilakukan.

3.3.3. Desain & Pengembangan

Tahap ketiga, Desain & Pengembangan (*Design & Development*). Dalam tahap ini dilakukan pembuatan rencana dan spesifikasi teknis untuk solusi yang dikembangkan. Mengembangkan solusi dalam bentuk produk termasuk didalamnya penentuan fungsi-fungsi dan arsitektur sistem. Lalu membuat prototipe dan melakukan pengujian awal untuk menguji apakah solusi tersebut dapat berfungsi dengan baik. Setelah itu melakukan iterasi dan perbaikan pada prototipe hingga solusi yang dihasilkan sesuai dengan spesifikasi teknis yang telah ditetapkan. Pengembangan layanan data dimulai dengan membuat skema data dari setiap entitas yang sudah ditentukan. Dari skema-skema data tersebut dibuatlah *query* dan *mutation* sesuai dengan kebutuhan aplikasi Tripperspace.

3.3.4. Demonstrasi

Tahap keempat, Demonstrasi (*Demonstration*). Mengujicoba solusi yang sudah dikembangkan apakah dapat menyelesaikan masalah atau tidak. Pengujian solusi yang telah dikembangkan dengan menggunakan data atau kasus yang realistis dan

melakukan pengujian kinerja untuk memastikan bahwa solusi yang dikembangkan berjalan dengan baik dan tidak menimbulkan masalah lainnya. Pengujian setiap *query* dan *mutation* pada layanan data GraphQL akan diujicoba dengan aplikasi Postman.

3.3.5. Evaluasi

Tahap kelima, Evaluasi (*Evaluation*). Mengobservasi dan mengukur seberapa efektif produk dalam menyelesaikan masalah, apabila hasil evaluasi belum memuaskan dapat dilakukan iterasi tahap 3,4,5 hingga diperoleh hasil evaluasi yang diinginkan. Hasil ujicoba dari aplikasi Postman akan di observasi apakah ada *query* atau *mutation* yang memberika respon *error* atau tidak sesuai dengan skema data yang sudah ditentukan. Perbaikan akan terus dilakukan hingga respon sesuai dengan skema data yang di inginkan.

3.3.6. Komunikasi

Dan tahap keenam, Komunikasi (*Communication*). Setelah hasil evaluasi yang di peroleh memuaskan, selanjutnya mengkomunikasikan permasalahan, pentingnya permasalahan dan solusi tersebut, produk yang dibuat, fungsi atau fitur dan keterbaruannya, validitas desainnya, dan efektifitas produk dalam menyelesaikan masalah. Salah satu bentuk komunikasi adalah Publikasi Ilmiah.

V. SIMPULAN DAN SARAN

5.1. Simpulan

Simpulan yang dapat diambil dari hasil penelitian adalah sebagai berikut:

1. Perancangan sistem layanan data graphql untuk aplikasi Tripperspace menghasilkan 16 *Query* dan 3 *Mutation* yang dapat digunakan.
2. Optimalisasi sistem layanan data dilakukan dengan penggunaan *cache* pada hasil *query* dan pembatasan maksimal kedalaman *query* yang dapat dilakukan. Dilihat dari hasil pengujian penggunaan *cache* dapat mengurangi waktu respon server dari 37 detik menjadi 30 detik.
3. Hasil pengujian menunjukkan perbedaan relasi database tidak mempengaruhi secara signifikan terhadap kecepatan waktu respon server. Sehingga dalam pengembangan layanan data dengan GraphQL dapat menggunakan relasi data sesuai kebutuhan tanpa memikirkan relasi data yang digunakan.

5.2. Saran

Berdasarkan penelitian yang telah dilakukan dalam penelitian ini, saran untuk pengembangan selanjutnya yang berlandaskan sebagai berikut:

1. Diharapkan pada penelitian selanjutnya dapat dilakukan *Integration Testing* pada aplikasi tripperspace untuk uji validitas apakah rancangan layanan data sudah sesuai dengan kebutuhan aplikasi Tripperspace.
2. Pada penelitian ini penggunaan *cache* dilakukan pada seluruh *query*, pada penelitian selanjutnya diharapkan dapat memilah data mana yang memerlukan *cache* atau tidak. Karena jika *cache* diterapkan pada data yang memiliki perubahan data yang cepat maka data yang diterima *client* tidak sesuai dengan data terbaru.

DAFTAR PUSTAKA

- [1] Kemendikbud, “Merdeka Belajar: Kampus Merdeka,” *kampusmerdeka.kemendikbud.go.id*, 2021.
<https://kampusmerdeka.kemendikbud.go.id/activity/active/detail/686486> (diakses 29 Mei 2022).
- [2] Kemenparekraf, *Buku Tren Pariwisata 2021*. Kemenparekraf, 2021. Diakses: 4 Juni 2023. [Daring]. Tersedia pada:
<https://www.kemenparekraf.go.id/pustaka/Buku-Tren-Pariwisata-2021>
- [3] G. Brito dan M. T. Valente, “REST vs GraphQL: A controlled experiment,” dalam *Proceedings - IEEE 17th International Conference on Software Architecture, ICOSA 2020*, 2020, hlm. 81–91. doi: 10.1109/ICSA47634.2020.00016.
- [4] S. Buna, *Learning GraphQL and Relay*, vol. 1. Packt Publishing, 2016. [Daring]. Tersedia pada: www.packtpub.com
- [5] R. T. Fielding dan R. N. Taylor, “Principled Design of the Modern Web Architecture,” *ACM Trans Internet Technol*, vol. 2, no. 2, hlm. 115–150, 2002, doi: 10.1145/514183.514185.
- [6] O. Hartig dan J. Pérez, “An initial analysis of facebook’s GraphQL language,” *CEUR Workshop Proc*, vol. 1912, 2017.
- [7] C. Carlos, M. Steven, dan R. Peter, *Database System: Design, Implementation, and Management*. Cengage Learning, Inc. ALL, 2018.
- [8] PostgreSQL Global Development Group, “PostgreSQL: Documentation: 15: 1. What Is PostgreSQL?,” *www.postgresql.org*, 2022.
<https://www.postgresql.org/docs/current/intro-what-is.html> (diakses 18 Desember 2022).
- [9] “PHP: Hypertext Preprocessor.” <https://www.php.net/> (diakses 16 April 2023).
- [10] “Laravel - The PHP Framework For Web Artisans.” <https://laravel.com/> (diakses 16 April 2023).
- [11] H. Purnaweni dan Riwayatiningih, “Pemanfaatan Sistem Informasi Geografi dalam Pengembangan Pariwisata Geographic Information System Utilization in Tourism Development,” *Proceeding Biology Education Conference*, vol. 14, hlm. 154–161, 2017.
- [12] R. P. Ardhiyani dan H. Mulyono, “Analisis Dan Perancangan Sistem Informasi Pariwisata Berbasis Web Sebagai Media Promosi Pada Kabupaten Tebo,” *Jurnal Manajemen Sistem Informasi*, vol. 3, no. 1, hlm. 952–972, 2018.
- [13] G. Brito, T. Mombach, dan M. T. Valente, “Migrating to GraphQL: A Practical Assessment,” *SANER 2019 - Proceedings of the 2019 IEEE 26th International Conference on Software Analysis, Evolution, and Reengineering*, hlm. 140–150, 2019, doi: 10.1109/SANER.2019.8667986.

- [14] S. Gregor dan A. R. Hevner, “Positioning and presenting design science research for maximum impact,” *MIS Q*, vol. 37, no. 2, hlm. 337–355, 2013, doi: 10.25300/MISQ/2013/37.2.01.
- [15] Tony D. Susanto, “Metode PENELITIAN SAINS-DESAIN (Design-Science Research) – Tony D. Susanto, Ph.D. (ITIL, COBIT, TOGAF),” *notes.its.ac.id*, 2020. <https://notes.its.ac.id/tonydwisusanto/2020/09/12/metode-penelitian-sains-desain-design-science-research/> (diakses 28 Mei 2022).
- [16] Kin Lane, *API Metrics: What to Measure, Why, and How*. San Francisco: API Evangelist, 2016.
- [17] Avinash Tiwari, *API Performance Testing: How to Test and Tune Your Web APIs for Maximum Performance*. California: O’Reilly Media, 2020.