

**ANALISIS KOMPARATIF REST API DAN GRAPHQL MENGGUNAKAN
RUNTIME ENVIRONMENT NODEJS**

SKRIPSI



disusun oleh

Fery Reza Aditya

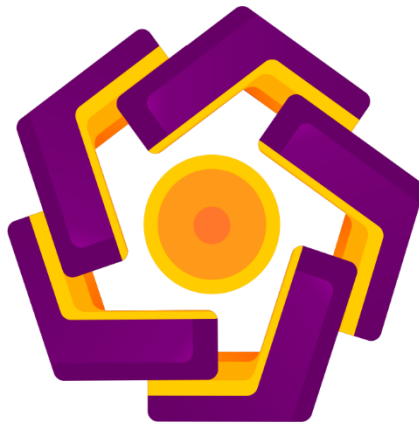
17.11.1731

**PROGRAM SARJANA
PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2022**

**ANALISIS KOMPARATIF REST API DAN GRAPHQL MENGGUNAKAN
RUNTIME ENVIRONMENT NODEJS**

SKRIPSI

untuk memenuhi sebagian persyaratan
mencapai gelar Sarjana
pada Program Studi Informatika



disusun oleh

Fery Reza Aditya
17.11.1731

**PROGRAM SARJANA
PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA**

2022

PERSETUJUAN

SKRIPSI

**ANALISIS KOMPARATIF REST API DAN GRAPHQL MENGGUNAKAN
RUNTIME ENVIRONMENT NODEJS**

yang dipersiapkan dan disusun oleh

Fery Reza Aditya

17.11.1731

telah disetujui oleh Dosen Pembimbing Skripsi
pada tanggal 19 April 2021

Dosen Pembimbing,

Lukman, M.Kom

NIK. 190302151

PENGESAHAN

SKRIPSI

**ANALISIS KOMPARATIF REST API DAN GRAPHQL MENGGUNAKAN
RUNTIME ENVIRONMENT NODEJS**

yang dipersiapkan dan disusun oleh

Fery Reza Aditya

17.11.1731

telah dipertahankan di depan Dewan Penguji
pada tanggal 19 April 2021

Susunan Dewan Penguji

Nama Penguji

Tanda Tangan

M. Rudyanto Arief, S.T, M.T
NIK. 190302098

Lukman, M.Kom
NIK. 190302151

Uyock Anggoro Saputro, M.Kom
NIK. 190302419

Skripsi ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana Komputer
Tanggal 25 September 2022

DEKAN FAKULTAS ILMU KOMPUTER

Hanif Al Fatta, M.Kom
NIK. 190302096

PERNYATAAN

Saya yang bertandatangan dibawah ini menyatakan bahwa, skripsi ini merupakan karya saya sendiri (ASLI), dan isi dalam skripsi ini tidak terdapat karya yang pernah diajukan oleh orang lain untuk memperoleh gelar akademis di suatu institusi pendidikan tinggi manapun, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis dan/atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Segala sesuatu yang terkait dengan naskah dan karya yang telah dibuat adalah menjadi tanggung jawab saya pribadi.

Yogyakarta, 31 Desember 2021



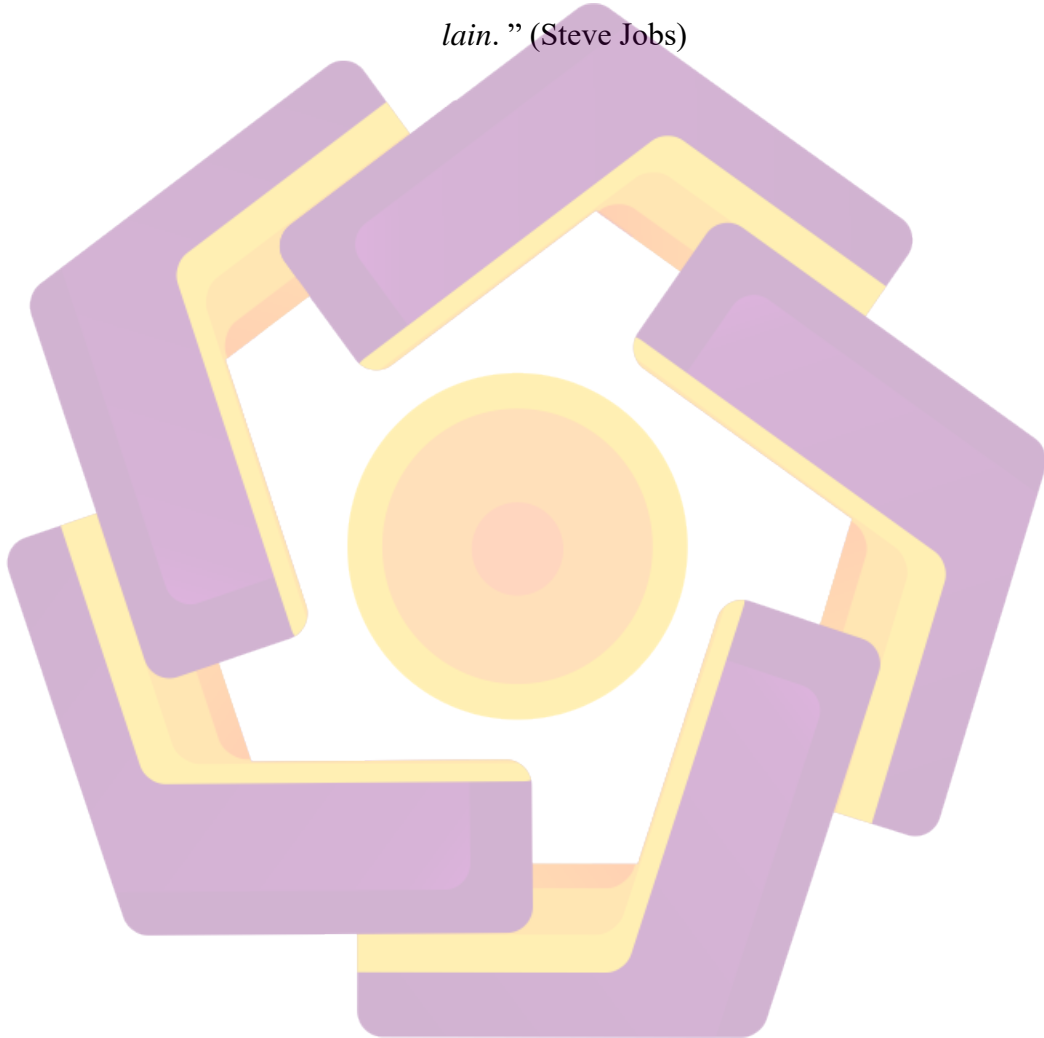
Fery Reza Aditya

17.11.1731

MOTTO

"Menyesali nasib tidak akan mengubah keadaan. Terus berkarya dan bekerjalah yang membuat kita berharga." (GusDur).

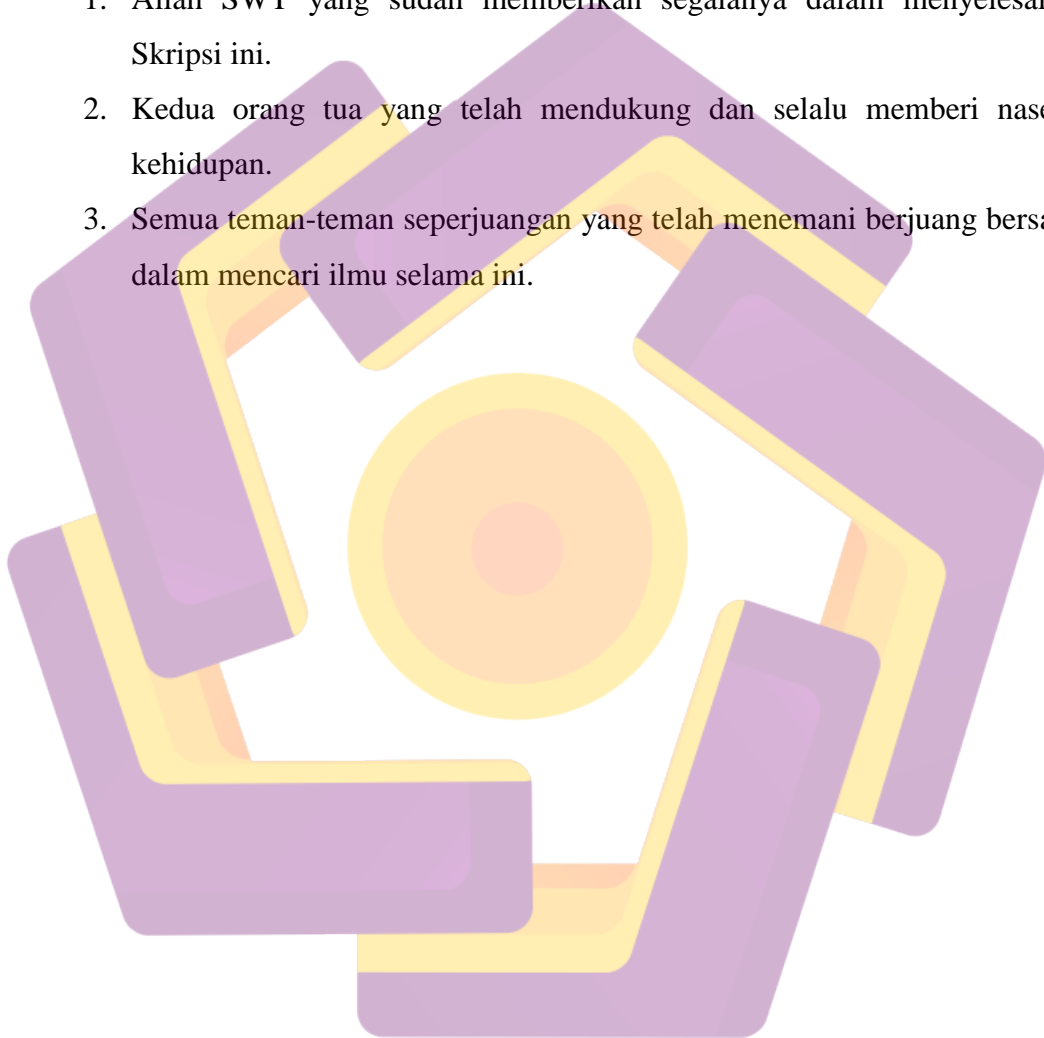
"Waktumu terbatas. Jangan menyia-menyiakan dengan menjalani hidup orang lain." (Steve Jobs)



PERSEMBAHAN

Alhamdulillah, Puji syukur saya panjatkan kehadiran Allah SWT atas terselesaikannya Skripsi ini dengan baik dan lancar, dan skripsi ini saya persembahkan untuk :

1. Allah SWT yang sudah memberikan segalanya dalam menyelesaikan Skripsi ini.
2. Kedua orang tua yang telah mendukung dan selalu memberi nasehat kehidupan.
3. Semua teman-teman seperjuangan yang telah menemani berjuang bersama dalam mencari ilmu selama ini.



KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan penulisan skripsi ini dengan judul “Analisis Komparatif Rest Api Dan GraphQL Menggunakan Runtime Environment Nodejs”.

Penulis menyadari sepenuhnya bahwa terselesaikannya skripsi ini karena adanya bantuan dan bimbingan dari berbagai pihak. Dalam pembuatan skripsi ini masih banyak sekali kekurangan-kekurangan dan kelemahan-kelemahannya. Oleh karena itu penulis berharap kepada semua pihak agar dapat menyampaikan kritik dan saran yang membangun untuk menambah kesempurnaan skripsi ini, namun penulis tetap berharap skripsi ini akan bermanfaat bagi semua pihak yang membacanya.

Yogyakarta, 31 Desember 2021

Penulis

Fery Reza Aditya

17.11.1731

DAFTAR ISI

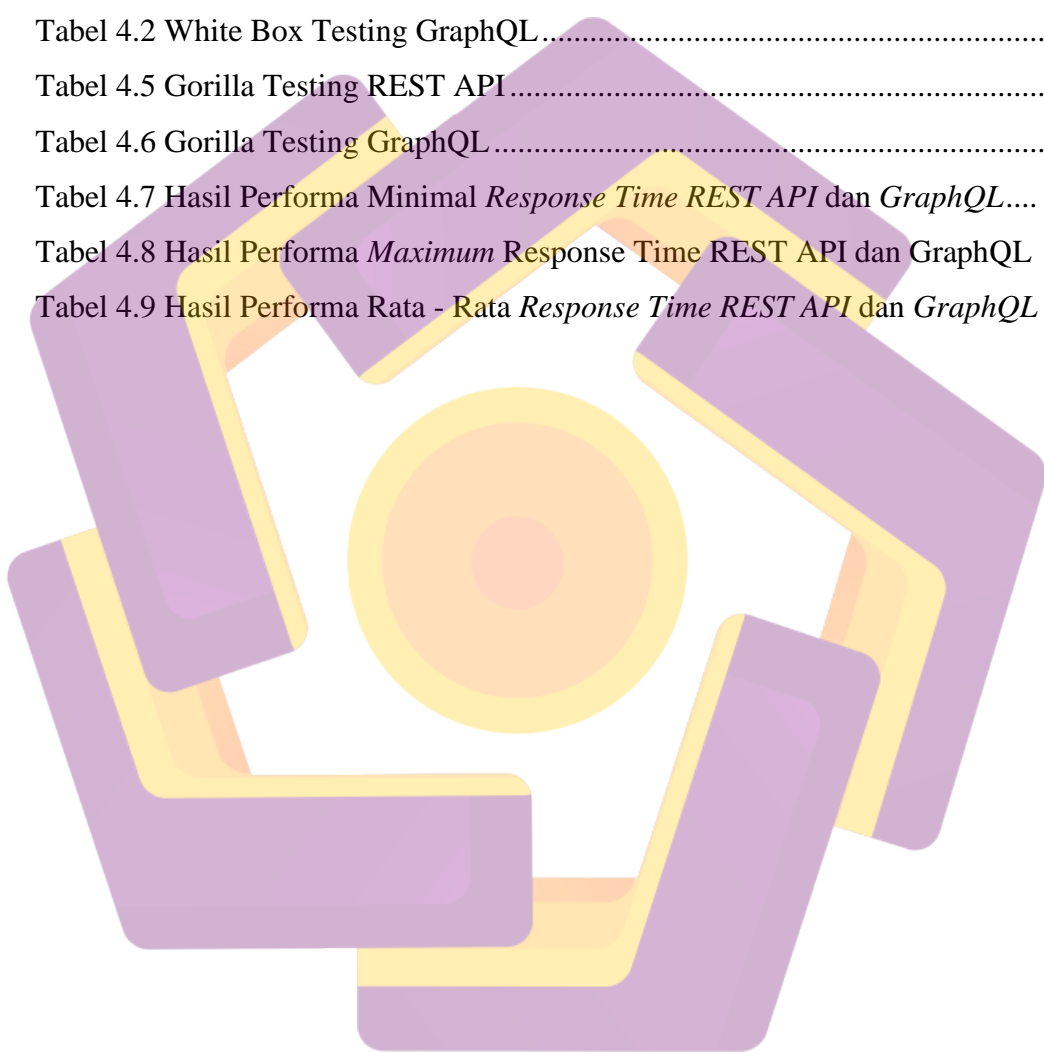
JUDUL	I
PERSETUJUAN.....	III
PENGESAHAN.....	IV
PERNYATAAN.....	V
MOTTO	VI
PERSEMBAHAN.....	VII
KATA PENGANTAR.....	VIII
DAFTAR ISI.....	IX
DAFTAR TABEL	XII
DAFTAR GAMBAR.....	XIII
INTISARI	XVIII
ABSTRACT	XIX
BAB I PENDAHULUAN.....	1
1.1 LATAR BELAKANG MASALAH.....	1
1.2 RUMUSAN MASALAH.....	2
1.2 BATASAN MASALAH.....	2
1.4 TUJUAN PENELITIAN	2
1.5 MANFAAT PENELITIAN.....	3
1.6 METODE PENELITIAN	3
1.7 SISTEMATIKA PENULISAN	3
BAB II LANDASAN TEORI.....	5
2.1 TINJAUAN PUSTAKA.....	5
2.2 ANALISIS.....	7
2.3 KOMPARATIF.....	7
2.4 DEFINISI <i>WEB SERVICE</i>	7

2.4.1	<i>REST API</i>	8
2.4.2	<i>GraphQL</i>	9
2.5	<i>JAVASCRIPT</i>	10
2.6	<i>RUNTIME ENVIRONMENT</i>	10
2.7	<i>NODEJS</i>	11
2.7.1	<i>Expressjs</i>	11
2.8	<i>MONGODB</i>	11
2.9	<i>JSON</i>	12
2.10	<i>APACHE JMETER</i>	13
2.11	<i>GORILLA TEST</i>	14
BAB III METODOLOGI PENELITIAN		15
3.1	<i>ALAT DAN BAHAN PENELITIAN</i>	15
3.1.1	<i>Kebutuhan Perangkat Keras</i>	15
3.1.2	<i>Kebutuhan Perangkat Lunak</i>	15
3.2	<i>SAMPLE DATA</i>	16
3.3	<i>ALUR PENELITIAN</i>	16
3.4	<i>STRUKTUR COLLECTION</i>	19
3.5	<i>PERANCANGAN ROUTE</i>	19
3.5.1	<i>Tabel Route REST API</i>	19
3.5.2	<i>Tabel Route GraphQL</i>	21
3.5.3	<i>Tabel Penyetaraan REST API dan GraphQL</i>	23
BAB IV HASIL DAN PEMBAHASAN		26
4.1	<i>INSTALASI DAN KONFIGURASI SOFTWARE</i>	26
4.1.1	<i>Instalasi Nodejs dan NPM</i>	26
4.1.2	<i>Instalasi MongoDB dan MongoDB Compass</i>	32
4.1.3	<i>Instalasi Apache JMeter</i>	38
4.1.4	<i>Inisialisasi Project REST API dan GraphQL</i>	40
4.1.5	<i>Instalasi dan konfigurasi Packages REST API</i>	40
4.1.6	<i>Instalasi Packages GraphQL</i>	42
4.2	<i>IMPORT DATA USERS KE MONGODB</i>	44

4.3	RANCANGAN ROUTE	46
4.3.1	<i>Rancangan route REST API</i>	46
4.3.2	<i>Rancangan route GraphQL</i>	52
4.4	SKENARIO PENGUJIAN.....	59
4.4.1	<i>White box Testings</i>	60
	<i>Tabel 4.1 White Box Testing REST API</i>	61
4.4.2	<i>Black box Testings</i>	66
	<i>Tabel 4.3 Black Box Testing REST API</i>	66
	<i>Tabel 4.4 Black Box GraphQL</i>	68
4.4.3	<i>Gorilla Testing</i>	70
4.5	HASIL PENGUJIAN	72
4.5.1	<i>Hasil Pengujian REST API</i>	72
4.5.2	<i>Hasil Pengujian GraphQL</i>	88
4.6	ANALISA PERFORMA	104
BAB V PENUTUP		106
5.1	KESIMPULAN.....	106
5.2	SARAN.....	106
DAFTAR PUSTAKA		107

DAFTAR TABEL

Tabel 2.1 Keunggulan dan perbedaan penelitian sebelumnya.....	6
Tabel 3.1 Route URL REST API.....	19
Tabel 3.2 Route URL GraphQL.....	21
Tabel 3.3 Penyetaraan <i>REST API</i> dan <i>GraphQL</i>	23
Tabel 4.2 White Box Testing GraphQL.....	64
Tabel 4.5 Gorilla Testing REST API.....	71
Tabel 4.6 Gorilla Testing GraphQL.....	72
Tabel 4.7 Hasil Performa Minimal <i>Response Time REST API</i> dan <i>GraphQL</i>	104
Tabel 4.8 Hasil Performa <i>Maximum Response Time REST API</i> dan <i>GraphQL</i>	104
Tabel 4.9 Hasil Performa Rata - Rata <i>Response Time REST API</i> dan <i>GraphQL</i>	104



DAFTAR GAMBAR

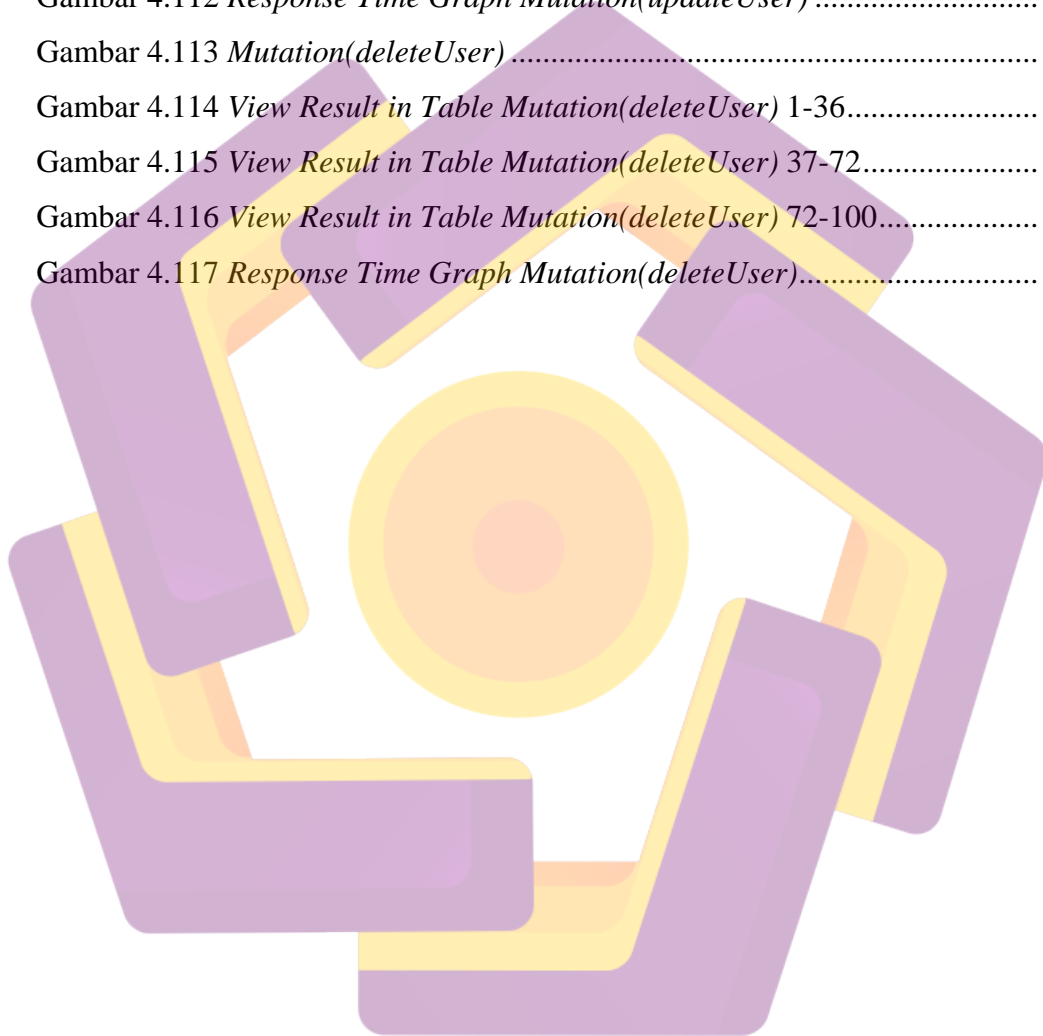
Gambar 2.1 <i>Web Services</i> (https://medium.com/@codequsdotcom/web-services-api-learn-web-services-from-scratch-c325d9fa34da)	8
Gambar 2.2 <i>REST API</i> (http://mfikri.com/artikel/restful-api).....	9
Gambar 2.3 <i>GraphQL</i> (https://www.netlify.com/blog/2020/01/21/advice-from-a-graphql-expert/)	10
Gambar 2.4 <i>Java Runtime Environment vs Nodejs Runtime Environment</i> (https://medium.com/free-code-camp/what-exactly-is-node-js-ae36e97449f5)....	11
Gambar 2.5 <i>Mongodb vs RDBMS</i> (https://beginnersbook.com/2017/09/mapping-relational-databases-to-mongodb/).....	12
Gambar 2.6 <i>Struktur JSON</i> (https://medium.com/@julianam.tyler/what-is-json-91df906ed27b)	13
Gambar 2.7 <i>GUI Apache JMeter</i> (https://en.wikipedia.org/wiki/Apache_JMeter)	14
Gambar 3.1 <i>Total Data</i>	16
Gambar 3.2 <i>Flowchart penelitian</i>	19
Gambar 3.3 <i>users collection</i>	19
Gambar 4.1 <i>Nodejs Setup Wizard</i>	26
Gambar 4.2 <i>End-User License Agreement</i>	27
Gambar 4.3 <i>Destination folder</i>	27
Gambar 4.4 <i>Custom Setup</i>	28
Gambar 4.5 <i>Tools for Native Modules</i>	29
Gambar 4.6 <i>Ready to Install Nodejs</i>	30
Gambar 4.7 <i>Nodejs Setup Wizard Finish</i>	31
Gambar 4.8 <i>command version Nodejs dan npm</i>	31
Gambar 4.9 <i>Welcome to the MongoDB</i>	32
Gambar 4.10 <i>End-User License Agreement</i>	33
Gambar 4.11 <i>Choose Setup Type</i>	34
Gambar 4.12 <i>Service Configuration</i>	35
Gambar 4.13 <i>Install MongoDB Compass</i>	36

Gambar 4.14 <i>Ready to install MongoDB</i>	37
Gambar 4.15 <i>Completed the MongoDB</i>	38
Gambar 4.16 <i>jmeter.bat</i>	39
Gambar 4.17 <i>Apache JMeter GUI</i>	39
Gambar 4.18 <i>NPM init</i>	40
Gambar 4.19 <i>REST API Depedencies</i>	41
Gambar 4.20 <i>REST API Dependencies</i>	42
Gambar 4.21 <i>REST API Devdependencies</i>	42
Gambar 4.22 <i>GraphQL Dependencies</i>	43
Gambar 4.23 <i>GraphQL Dependencies</i>	43
Gambar 4.24 <i>GraphQL Devdependencies</i>	44
Gambar 4.25 <i>MongoDB Compass Create Connection</i>	44
Gambar 4.26 <i>MongoDB Compass Create database and Collection</i>	45
Gambar 4.27 <i>MongoDB Compass import file</i>	45
Gambar 4.28 <i>MongoDB Compass Select File</i>	46
Gambar 4.29 <i>Total documents</i>	46
Gambar 4.30 <i>REST API Server</i>	47
Gambar 4.31 <i>Get All User Routes</i>	47
Gambar 4.32 <i>Get All User Controller</i>	48
Gambar 4.33 <i>Get All User Service</i>	48
Gambar 4.34 <i>Get User By Id Routes</i>	48
Gambar 4.35 <i>Get User By Id Controller</i>	48
Gambar 4.36 <i>Get User By Id Services</i>	49
Gambar 4.37 <i>Create user Routes</i>	49
Gambar 4.38 <i>Create user Controller</i>	49
Gambar 4.39 <i>Create user Service</i>	50
Gambar 4.40 <i>Update user by id Routes</i>	50
Gambar 4.41 <i>Update user by id Controller</i>	51
Gambar 4.42 <i>Update user by id Service</i>	51
Gambar 4.43 <i>Delete user by id Routes</i>	51
Gambar 4.44 <i>Delete user by id Controller</i>	51

Gambar 4.45 <i>Delete user by id Service</i>	52
Gambar 4.46 <i>GraphQL Server</i>	52
Gambar 4.47 <i>GraphQL Schema</i>	53
Gambar 4.48 <i>GraphQL Schema Input</i>	54
Gambar 4.49 <i>GraphQL Query users</i>	55
Gambar 4.50 <i>GraphQL Query users Controller</i>	55
Gambar 4.51 <i>GraphQL Query users Service</i>	55
Gambar 4.52 <i>GraphQL Query users(userId) Service</i>	56
Gambar 4.53 <i>GraphQL Mutation createUser</i>	57
Gambar 4.54 <i>GraphQL Mutation createUser Controller</i>	57
Gambar 4.55 <i>GraphQL Mutation createUser Service</i>	57
Gambar 4.56 <i>GraphQL Mutation updateUser</i>	58
Gambar 4.57 <i>GraphQL Mutation updateUser Controller</i>	58
Gambar 4.58 <i>GraphQL Mutation updateUser Service</i>	58
Gambar 4.59 <i>GraphQL Mutation deleteUser</i>	59
Gambar 4.60 <i>GraphQL Mutation deleteUser Controller</i>	59
Gambar 4.61 <i>GraphQL Mutation deleteUser Service</i>	59
Gambar 4.62 <i>Server performance</i>	60
Gambar 4.63 <i>Result White box Test REST API</i>	61
Gambar 4.64 <i>Result White box Test GraphQL</i>	64
Gambar 4.65 <i>Gorilla Test Schema</i>	71
Gambar 4.66 <i>Get All User HTTP Request</i>	73
Gambar 4.67 <i>View Result in Table Get All User 1 - 36</i>	73
Gambar 4.68 <i>View Result in Table Get All User 37 - 72</i>	74
Gambar 4.69 <i>View Result in Table Get All User 73 - 100</i>	74
Gambar 4.70 <i>Response Time Graph Get All User</i>	75
Gambar 4.71 <i>HTTP Request Get User By Id</i>	75
Gambar 4.72 <i>View Result in Table Get User By Id 1 - 36</i>	76
Gambar 4.73 <i>View Result in Table Get User By Id 37 - 72</i>	76
Gambar 4.74 <i>View Result in Table Get User By Id 72 - 100</i>	77
Gambar 4.75 <i>Response Time Graph Get User By Id</i>	77

Gambar 4.76 <i>HTTP Request Create User</i>	78
Gambar 4.77 <i>HTTP Header Create User</i>	78
Gambar 4.78 <i>View Result in Table Create User 1 - 36</i>	79
Gambar 4.79 <i>View Result in Table Create User 37 - 72</i>	80
Gambar 4.80 <i>View Result in Table Create User 72 - 100</i>	81
Gambar 4.81 <i>Response Time Graph Create User</i>	81
Gambar 4.82 <i>HTTP Request Update User By Id</i>	82
Gambar 4.83 <i>View Result in Table Update User By Id 1 - 36</i>	83
Gambar 4.84 <i>View Result in Table Update User By Id 36 - 71</i>	83
Gambar 4.85 <i>View Result in Table Update User By Id 71 - 100</i>	84
Gambar 4.86 <i>Response Time Graph Update User By Id</i>	84
Gambar 4.87 <i>HTTP Request Delete User By Id</i>	85
Gambar 4.88 <i>View Result in Table Delete user By Id 1-36</i>	85
Gambar 4.89 <i>View Result in Table Delete user By Id 37-72</i>	86
Gambar 4.90 <i>View Result in Table Delete user By Id 72-100</i>	87
Gambar 4.91 <i>Response Time Graph Delete user By Id</i>	87
Gambar 4.92 <i>GraphQL HTTP Header</i>	88
Gambar 4.93 <i>Query User</i>	88
Gambar 4.94 <i>View Result in Table Query User 1-36</i>	89
Gambar 4.95 <i>View Result in Table Query User 37-72</i>	90
Gambar 4.96 <i>View Result in Table Query User 72-100</i>	91
Gambar 4.97 <i>Response Time Graph Query User</i>	91
Gambar 4.98 <i>Query User(userId)</i>	92
Gambar 4.99 <i>View Result in Table Query User(userId) 1-36</i>	92
Gambar 4.100 <i>View Result in Table Query User(userId) 37-72</i>	93
Gambar 4.101 <i>View Result in Table Query User(userId) 72-100</i>	94
Gambar 4.102 <i>Response Time Graph Query User(userId)</i>	94
Gambar 4.103 <i>Mutation(createUser)</i>	95
Gambar 4.104 <i>View Result in Table Mutation(createUser) 1-36</i>	96
Gambar 4.105 <i>View Result in Table Mutation(createUser) 37-72</i>	96
Gambar 4.106 <i>View Result in Table Mutation(createUser) 72-100</i>	97

Gambar 4.107 <i>Response Time Graph Mutation(createUser)</i>	97
Gambar 4.108 <i>Mutation(updateUser)</i>	98
Gambar 4.109 <i>View Result in Table Mutation(updateUser)</i> 1-36	98
Gambar 4.110 <i>View Result in Table Mutation(updateUser)</i> 37-72	99
Gambar 4.111 <i>View Result in Table Mutation(updateUser)</i> 72-100	100
Gambar 4.112 <i>Response Time Graph Mutation(updateUser)</i>	100
Gambar 4.113 <i>Mutation(deleteUser)</i>	101
Gambar 4.114 <i>View Result in Table Mutation(deleteUser)</i> 1-36.....	101
Gambar 4.115 <i>View Result in Table Mutation(deleteUser)</i> 37-72.....	102
Gambar 4.116 <i>View Result in Table Mutation(deleteUser)</i> 72-100.....	103
Gambar 4.117 <i>Response Time Graph Mutation(deleteUser)</i>	103



INTISARI

Perbandingan performa dari kedua teknologi *REST API* dan *GraphQL* menggunakan *Nodejs* sebagai *runtime environment*-nya dan *Apache JMeter* sebagai *tools benchmark* pengujian pada penelitian ini. Skema pengetesan pada penelitian ini yaitu *White box test* menggunakan *mocha* dan *chai*, *Black box test* menggunakan *Postman*, dan *Gorilla test* menggunakan *Apache JMeter*. Pada penelitian ini terdapat beberapa variable penelitian yaitu performa dalam menerima *request* dan mengirimkan *response* yang diminta oleh *user*.

Alur penelitian dalam penelitian ini adalah membandingkan kedua teknologi tersebut dengan melayani *request* dan memberikan *response* dari *user*. *Database* yang digunakan yaitu *MongoDB* karena melewati proses eksekusi *query sql*.

Pada Penelitian ini *REST API* memiliki *response time* yang lebih cepat dari *GraphQL* dalam berbagai skenario pengujian. Akan tetapi *GraphQL* memiliki kelebihan yaitu hanya memerlukan satu buah *route* atau *URL* untuk setiap request oleh *user*.

Kata Kunci: *REST API, GraphQL, Apache JMeter, MongoDB, Nodejs.*



ABSTRACT

Comparison of the performance of both REST API and GraphQL technologies using Nodejs as the runtime environment and Apache JMeter as the benchmark testing tool in this study. The test scheme in this study is White box test using mocha and chai, Black box test using Postman, and Gorilla test using Apache JMeter. In this study there are several research variables, namely performance in receiving requests and sending responses requested by the user.

The research flow in this study is to compare the two technologies by serving requests and providing responses from users. The database used is MongoDB because it goes through the SQL query execution process.

In this study, REST API has a faster response time than GraphQL in various test scenarios. However, GraphQL has the advantage that it only requires one route or URL for each request by the user.

Keywords: REST API, GraphQL, Apache JMeter, MongoDB, Nodejs.

