

**OPTIMALISASI QUERY DATABASE E LEARNING JTI  
POLITEKNIK NEGERI JEMBER MENGGUNAKAN INDEX**

**SKRIPSI**



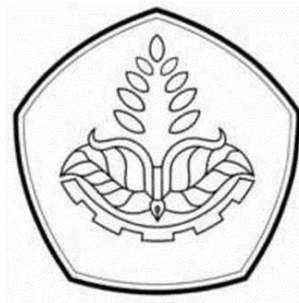
oleh

**Ega Kustian Pratama  
NIM E41171590**

**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI JEMBER  
2022**

**OPTIMALISASI QUERY DATABASE E LEARNING JTI  
POLITEKNIK NEGERI JEMBER MENGGUNAKAN INDEX**

**SKRIPSI**



Sebagai salah satu syarat untuk memperoleh gelar Sarjana Sains Terapan (S.ST)  
di Program Studi Teknik Informatika  
Jurusan Teknologi Informasi

oleh

**Ega Kustian Pratama  
E41171950**

**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI JEMBER  
2022**

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,  
RISET DAN TEKNOLOGI  
POLITEKNIK NEGERI JEMBER  
JURUSAN TEKNOLOGI INFORMASI

---

OPTIMALISASI QUERY DATABASE JTI POLITEKNIK NEGERI  
JEMBER MENGGUNAKAN INDEX

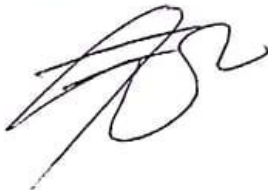
Diuji pada tanggal : 24 Januari 2022

Ketua,



I Putu Dodv Lesmana, ST, MT  
NIP. 19790921 200501 1 001

Sekretaris,



Ery Setiyawan JulevAtmaji, S.Kom, M.C  
NIP 19890710 201903 1 010

Anggota,



I Gede Wiryawan, S.Kom, M.Kom  
NIP 19880117 201903 1 008

Mengesahkan  
Ketua Jurusan Teknologi Informasi



Heandriant Rishawan, S.Kom, M.Cs  
NIP 19830203 200604 1 003



## SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Ega Kustian Pratama

NIM : E41171590

menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam Skripsi saya yang berjudul "Optimalisasi Query Database JTI Politeknik Negeri Jember menggunakan Index" merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing, dan belum pernah diajukan dalam bentuk apapun pada perguruan tinggi manapun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan dari penulis lain telah disebutkan dalam naskah dan dicantumkan dalam daftar pustaka di bagian akhir Skripsi ini.

Jember, 24 Januari 2022



Ega Kustian Pratama  
NIM E41171590







**PERNYATAAN  
PERSETUJUAN PUBLIKASI  
KARYA ILMIAH UNTUK KEPENTINGAN  
AKADEMIS**

Yang bertanda tangan di bawah ini, saya :

Nama : Ega Kustian Pratama  
NIM : E41171590  
Program Studi : Teknik Informatika  
Jurusan : Teknologi Informasi

Demi pengembangan ilmu pengetahuan, saya menyetujui untuk memberikan kepada UPT. Perpustakaan Politeknik Negeri Jember, Hak Bebas Royalti Non- Eksklusif (Non-Exclusive Royalty Free Right) atas Karya Ilmiah berupa Laporan Skripsi saya yang berjudul :

**OPTIMALISASI QUERY DATABASE JTI POLITEKNIK NEGERI  
JEMBER MENGGUNAKAN INDEX**

Dengan Hak Bebas Royalti Non-Eksklusif ini UPT. Perpustakaan Politeknik Negeri Jember berhak menyimpan, mengalih media atau format, mengelola dalam bentuk pangkalan data (Database), mendistribusikan karya dan menampilkan atau mempublikasikannya di internet atau media lain untuk kepentingan akademis tanpa perlu meminta ijin dari saya selama tetap mencantumkan nama saya sebagai penulis atau pencipta.

Saya bersedia untuk menanggung secara pribadi tanpa melibatkan pihak Politeknik Negeri Jember, Segala bentuk tuntutan hukum yang timbul atas pelanggaran hak cipta dalam karya ilmiah ini.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Jember  
Tanggal : 06 Juni 2021  
Diyatakan,  
  
9B5AJX670628306

Nama : Ega Kustian Pratama  
NIM. : E41171590

## **MOTTO**

“The only way to do great work is to love what you do”

*(steve job)*

“Aspek penting dari kreativitas adalah tidak takut untuk gagal”

*(Dr. Edwin Land)*

“Cara terbaik untuk memprediksi masa depan anda adalah dengan menciptakannya”

*(Abraham Lincoln)*



## **PERSEMBAHAN**

Dengan penuh rasa syukur, sebuah karya kecil laporan skripsi ini saya persembahkan kepada orang-orang yang tercinta:

1. Almarhum Ayahku tercinta Kusno dan ibuku tercinta Tutik Puji Hariani yang telah memberikan kasih sayang dan semangat yang tidak pernah surut dengan doa serta selalu membangkitkanku dalam setiap keterpurukan.
2. Much Nesha Adinata, Fahrizal Azi Ferdiansyah yang sudah membantu dan memberikan dukungan dalam pengerjaan skripsi ini.
3. Selfi Sulistyو Ningsih yang memberikan support dalam pengerjaan skripsi ini.
4. Sahabat dan teman-teman seperjuangan Teknik Informatika 2017 yang mengisi keseharianku dengan penuh ceria, terima kasih atas kebersamaan dan persahabatan yang tidak terlupakan.
5. Almamater tercinta Politeknik Negeri Jember

# OPTIMALISASI QUERY DATABASE JTI POLITEKNIK NEGERI JEMBER MENGGUNAKAN INDEX

**Ega Kustian Pratama**

Program Studi Teknik Informatika  
Jurusan Teknologi Informasi

## ***ABSTRACT***

*E-Learning JTI is a place of online learning program especially for student of Information technology department at Politeknik Negeri Jember. Sometimes, this system has an error in extremely low response time. Therefore, with the help of query index in the several tables, hopefully will help this system being optimal. The initial database upload was using bigdump which is useful for uploading the database locally in mysql. Also used Moodle as online learning platform from Moodle can find out several important database flows used in e-learning. Moodle has about 64 tables which has important role in e-learning with total 419 tables used. The main database was given a query index to connect among cables that will be useful for the more optimal response time and try to execute the relation among cables without index. From the result we enter the data in excel to compare which response time is faster. The next step is the result of the query index and without query index, a phyton program is created which is useful for comparing which respond time is more optimal.*

**Keyword:** *E – Learning JTI, Moodle , Query index, Phyton*

# **Optimalisasi Query Database JTI Politeknik Negeri Jember menggunakan Index**

**Ega Kustian  
Pratama**

Program Studi Teknik  
Informatika Jurusan  
Teknologi Informasi

## **ABSTRACT**

E – Learning JTI yaitu suatu tempat pembelajaran online khususnya untuk mahasiswa jurusan teknologi informasi yang ada di Politeknik Negeri Jember. Dari suatu sistem ini terkadang memiliki kendala berupa respon time yang cukup lambat maka dari itu. Dengan bantuan query index pada beberapa tabel yang nantinya akan membuat sistem ini berjalan lebih optimal. Awal upload database menggunakan bigdump yang berguna untuk upload database secara local di mysql. Penggunaan moodle pun tak lupa yakni sebuah platform pembelajaran secara online dari moodle dapat mengetahui beberapa aluran database yang penting digunakan dalam e – learning itu. Dari moodle ada sekitar 64 tabel yang berperan penting dalam e – learning dari total 419 tabel yang digunakan. Database yang utama tadi diberi query index untuk direlasikan antar tabel yang nantinya akan berguna untuk respon time lebih optimal dan mencoba exsecusi relasi antar tabel tanpa index. Dari hasil tersebut data kita masukkan di excel untuk membandingkan respon time mana yang lebih cepat. Langkah selanjutnya hasil dari query index dan tanpa query index itu dibuatkan program phyton yang berguna untuk membandingkan respon time mana yang lebih optimal.

***Key word : E – Learning JTI, Moodle , Query index, Phyton***

## RINGKASAN

**Optimalisasi Query Database JTI Politeknik Negeri Jember menggunakan Index**, Ega Kustian Pratama. Nim E4117190, Tahun 2021, Teknologi Informasi, Politeknik Negeri Jember, Ery Setiyawan Julev Atmaji, S.Kom,M.Cs (Pembimbing I).

Politeknik Negeri Jember adalah Politeknik ternama di Jawa Timur yang bertempat di kabupaten Jember dengan memiliki banyak jurusan salah satunya jurusan Teknologi Informasi ada tiga program studi yakni Teknik komputer, Manajemen informatika, dan Teknik informatika. Pada jurusan ini juga memiliki portal *system* informasi yang dikelola oleh jurusan sendiri dan *E-Learning*. Namun *system* tersebut belum berjalan efektif karena masih terpisah portal yakni JTI.polije.ac.id dan E-Learningnya menambahkan garis miring *elearning* untuk masuk portal *E-Learning*. Pada *E-Learning* JTI juga masih menggunakan *database* MYSQL yang dinilai kurang efisien dalam mengelola data yang cukup besar. Dari hal itu sistem informasi ini terkadang sering mengalami kendala yakni *respond time* yang cukup lama jika diakses oleh banyak *user*.

Berdasarkan masalah tersebut maka, penelitian ini melakukan seleksi tabel yang ada di *database e-learning* JTI Politeknik Negeri Jember menggunakan query *index*. Yang melalui tahap *upload database* dengan *bigdump*, cek relasi dengan *moodle*, pemberian *query index* ke beberapa tabel yakni 64 tabel dari 419 tabel total keseluruhan. Data hasil *query index* lalu dimasukkan ke dalam *excel* yang nantinya akan dihitung dan dibuat grafik untuk mengetahui perbandingan antara dengan *query* dan tanpa *query*.

Hasil Penelitian dengan menggunakan *query index* pada *database e – learning* JTI ini akan menghasilkan suatu *load* data dan *respon time* lebih optimal dari sebelumnya. *Query index* ini diharapkan dapat mempermudah para mahasiswa untuk belajar.



## PRAKATA

Puji Syukur Alhamdulillah penulis panjatkan atas limpahan rahmat serta karunia Allah SWT sehingga penulis dapat menyelesaikan Skripsi yang berjudul “Optimalisasi Query Database JTI Politeknik Negeri Jember menggunakan Index” Skripsi ini disusun guna memenuhi syarat untuk menyelesaikan studi di Politeknik Negeri Jember Program Studi D-IV Teknik Informatika.

Penyusunan Skripsi ini tak lepas dari bantuan berbagai pihak, oleh karena itu penulis ingin menyampaikan ucapan terima kasih kepada semua pihak yang telah membantu:

1. Bapak Saiful Anwar, S.TP, MP selaku direktur Politeknik Negeri Jember
2. Hendra Yufit Riskiawan, S.Kom, M.Cs selaku Ketua Jurusan Teknologi Informasi
3. Trismayanti Dwi P, S.Kom, M.Cs selaku Ketua Prodi Teknik Informatika.
4. Bapak Ery Setiyawan Jullev Atmadji, S.Kom, M.Cs atas bimbingan, kesabaran serta ilmu yang telah diberikan selama ini.
5. Almarhum Ayahku tercinta Kusno dan ibuku tercinta Tutik Puji Hariani yang telah memberikan kasih sayang dan semangat yang tidak pernah surut dengan doa serta selalu membangkitkanku dalam setiap keterpurukan.
6. Sahabat dan teman-teman seperjuangan Teknik Informatika 2017 yang mengisi keseharianku dengan penuh ceria, terima kasih atas kebersamaan dan persahabatan yang tidak terlupakan.
7. Semua pihak yang turut membantu sehingga skripsi ini dapat selesai

Penulis menyadari masih banyak kekurangan dan ketidaksempurnaan dalam penulisan skripsi ini. Untuk itu, kritik dan saran yang membangun demi kesempurnaan penulisan selanjutnya. Semoga skripsi ini dapat memberikan manfaat bagi kita semua. Amin.

Jember, 5 Juli 2021

Penulis



## DAFTAR ISI

	Halaman
<b>COVER</b> .....	<b>i</b>
<b>HALAMAN JUDUL</b> .....	<b>ii</b>
<b>HALAMAN PENGESAHAN</b> .....	<b>iii</b>
<b>HALAMAN PERNYATAAN</b> .....	<b>ii</b>
<b>HALAMAN PUBLIKASI</b> .....	<b>v</b>
<b>HALAMAN MOTTO</b> .....	<b>vi</b>
<b>HALAMAN PERSEMBAHAN</b> .....	<b>vii</b>
<b>ABSTRACT Bahasa Inggris</b> .....	<b>viii</b>
<b>ABSTRACT Bahasa Indonesia</b> .....	<b>ix</b>
<b>RINGKASAN</b> .....	<b>ix</b>
<b>PRAKATA</b> .....	<b>xi</b>
<b>DAFTAR ISI</b> .....	<b>xii</b>
<b>DAFTAR GAMBAR</b> .....	<b>xiii</b>
<b>DAFTAR TABEL</b> .....	<b>xiv</b>
<b>BAB 1. PENDAHULUAN</b> .....	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	2
1.3. Batasan Masalah.....	3
1.4. Tujuan.....	3
1.5. Manfaat.....	3
<b>BAB 2. TINJAUAN PUSTAKA</b> .....	<b>4</b>
2.1. State Of The Art .....	4
2.2. Pengertian Sistem .....	6
2.3. Pengertian Informasi .....	7
2.4. Pengertian Sistem Informasi .....	7
2.5. Definisi Data .....	9
2.5.1. Sejarah <i>E-Learning</i> .....	9
2.5.2. Pengertian <i>E-Learning</i> .....	10
2.5.3. Manfaat Perkuliahan Online .....	11
2.6. Pengenalan <i>Database</i> , <i>DBMS</i> dan <i>RDBMS</i> .....	12

2.6.1. Hierarki <i>Database</i> .....	13
2.6.2. Mysql .....	14
2.7. Mengenal SQL ( <i>Structured Query Language</i> ).....	15
2.8. Optimasi <i>Query</i> .....	16
2.9. Index.....	18
2.9.1. <i>Column Indexes</i> (Index Satu Kolom) .....	18
2.9.2. <i>Multiple-Column Indexes</i> (Index Beberapa Kolom).....	18
2.10. Pemrograman Bahasa <i>Python</i> .....	19
<b>BAB 3. METODOLOGI PENELITIAN .....</b>	<b>20</b>
3.1. Tempat dan Waktu Penelitian .....	20
3.1.1. Tempat Penelitian .....	20
3.1.2. Waktu Penelitian.....	20
3.2. Alat dan Bahan .....	20
3.2.1. Alat .....	20
3.2.2. Bahan .....	20
3.3. Tahapan Penelitian .....	21
3.3.1. Tahapan tahapan metode <i>waterfall</i> .....	21
3.3.2. Pengumpulan Data.....	22
3.4. Rancangan Sistem .....	23
3.4.1. Flowchart .....	23
<b>BAB 4. HASIL DAN PEMBAHASAN .....</b>	<b>27</b>
4.1. Data Percobaan.....	27
4.2. Proses Data .....	28
4.3. Tahapan Metode <i>Waterfall</i> .....	29
4.3.1. <i>Analysis</i> .....	29
4.3.2. <i>Design</i> .....	29
4.3.3. <i>Implementation</i> .....	32
4.3.4. <i>Testing</i> .....	37
4.3.5. <i>Maintenance</i> .....	42
<b>BAB 5. KESIMPULAN DAN SARAN .....</b>	<b>45</b>
5.1. Kesimpulan.....	45
5.2. Saran.....	45
<b>DAFTAR PUSTAKA .....</b>	<b>47</b>
<b>LAMPIRAN.....</b>	<b>Error! Bookmark not defined.</b>

## DAFTAR GAMBAR

Gambar 2.1 Hirarki Database.....	13
Gambar 3.1 Model Waterfall (Bassil,2012).....	20
Gambar 3.2 Flowchart proses index.....	23
Gambar 3.3 Rancangan flowchart testing system optimasi query database ...	24
Gambar 4.3 ERD database E – learning JTI .....	30
Gambar 4.4 Database E – learning JTI .....	31
Gambar 4.5 Upload database .....	32
Gambar 4.6 query index.....	35
Gambar 4.7 Query index .....	35
Gambar 4.8 Drop index.....	38
Gambar 4.9 Export 64 tabel e -learing dengan indeks.....	38
Gambar 4.10 export 64 tabel e -learing_kw dengan tanpa index.....	37
Gambar 4.11 Import data oleh python .....	38
Gambar 4.12 Pembuatan fungsi input ke database mysql .....	38
Gambar 4.13 Pembuatan fungsi perbandingan .....	39
Gambar 4.14 Proses membandingkan data .....	39
Gambar 4.15 Hasil dari perbandingan data.....	40
Gambar 4.16 Pengecekan data didalam database mysql.....	40
Gambar 4.17 Grafik hasil Loop 1 .....	40
Gambar 4.18 Grafik hasil Loop 2 .....	40
Gambar 4.19 Grafik hasil Loop 3 .....	41
Gambar 4.20 Grafik hasil Loop 4 .....	42
Gambar 4.21 Grafik hasil Loop 5 .....	42
Gambar 4.22 Grafik rata-rata hasil keseluruhan .....	43
Gambar 4.23 Coding phyton untuk menampilkan grafik perbandingan.....	44

## DAFTAR TABEL

Tabel 2. 1 Studi Literatur .....	5
----------------------------------	---

## BAB 1. PENDAHULUAN

### 1.1. Latar Belakang

Perkembangan teknologi informasi dan komunikasi sampai dengan saat ini begitu pesatnya seiring waktu berjalan, bahkan begitu banyak orang di seluruh dunia bergantung pada teknologi yang berkembang. Sistem informasi perkuliahan *online* adalah cara baru dalam proses belajar mengajar yang merupakan dasar dan konsekuensi logis dari perkembangan teknologi informasi dan komunikasi. Dengan perkuliahan *online*, peserta ajar (*learner* atau mahasiswa) tidak perlu duduk di ruang kelas untuk menyimak setiap ucapan dari seorang dosen secara langsung. Perkuliahan *online* juga dapat mempersingkat jadwal target waktu perkuliahan, dan tentu saja menghemat biaya yang harus dikeluarkan oleh sebuah program studi atau program pendidikan.

Sebagaimana yang disebutkan di atas, perkuliahan *online* telah mempersingkat waktu pembelajaran dan membuat biaya studi lebih ekonomis. Perkuliahan *online* mempermudah interaksi antara mahasiswa dengan bahan/materi, mahasiswa dengan dosen/instruktur maupun sesama mahasiswa. Mahasiswa dapat saling berbagi informasi dan dapat mengakses bahan-bahan belajar setiap saat dan berulang-ulang, dengan kondisi yang demikian itu dapat lebih memantapkan penguasaannya terhadap materi pembelajaran. (Sudarmaji, 2016).

Politeknik Negeri Jember adalah Politeknik ternama di Jawa Timur yang bertempat di kabupaten Jember dengan memiliki banyak jurusan salah satunya jurusan Teknologi Informasi ada tiga program studi yakni Teknik komputer, Manajemen informatika, dan Teknik informatika. Pada jurusan ini juga memiliki portal sistem informasi yang dikelola oleh jurusan sendiri dan *E-Learning*. Namun sistem tersebut belum berjalan efektif karena masih terpisah portal yakni JTI.poliije.ac.id dan *E-Learning*nya menambahkan garis miring E-learning untuk masuk portal *E-Learning*. Pada *E-Learning* JTI juga masih menggunakan *database* MYSQL yang dinilai kurang efisien dalam mengelola data yang cukup besar.

Tujuan dari optimasi *query* adalah menemukan jalan akses yang paling baik untuk meminimalkan total waktu pada saat proses sebuah *query*. Untuk mencapai tujuan tersebut, maka diperlukan *optimizer* untuk melakukan analisa *query* dan untuk melakukan pencarian jalan akses. SQL adalah *transform-oriented language* atau bahasa yang dirancang dengan penggunaan relasi untuk mengubah masukan menjadi keluaran yang dibutuhkan. Sebagai sebuah bahasa, standar internasional SQL menetapkan dua komponen pokok yaitu *Data Definition Language* (DDL) untuk mendefinisikan struktur basis data dan akses kontrol data dan *Data Manipulation Language* (DML) untuk mengembalikan dan perbarui data. Dimana optimasi *query* memberikan sebuah cara pemecahan masalah dengan mencoba menggabungkan teknik-teknik yang meliputi transformasi logika dan mempresentasikan teknik tersebut dalam sebuah masalah. Pemecahan masalah dengan membandingkan dua bentuk algoritma yang paling banyak digunakan untuk mengetahui mana yang terbaik yang dapat digunakan pada kondisi tertentu. Sedangkan *Query* merupakan kemampuan untuk menampilkan suatu data dari *database* dimana mengambil dari tabel-tabel yang ada pada *database* tersebut, namun tabel tersebut tidak semua ditampilkan, hanya sesuai data yang diinginkan atau data yang ingin ditampilkan. (Bismo Raharjo et al., 2018)

Berdasarkan uraian diatas pentingnya pengelolaan *database* untuk mengoptimalkan **system** informasi *E-Learning* agar lebih optimal jika diakses atau digunakan oleh banyak *user*. Oleh karena itu dalam menampilkan data, melakukan pencarian data, dan mengelola data membutuhkan *response time* yang memakan waktu sekitar 3- 5 menit maka dari itu dibutuhkannya Teknik *index* yang dapat mempercepat proses data yaitu dengan Teknik optimasi *query* menggunakan *index* sehingga dapat mempercepat kinerja *database*.

## 1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dibuat, maka rumusan masalah pada penelitian ini adalah :



1. Bagaimana cara mengoptimalkan kinerja *query database* dengan menggunakan Teknik optimasi *query* menggunakan *index*?
2. Bagaimana cara mempercepat pengolahan data dan menampilkan data agar lebih optimal?

### **1.3. Batasan Masalah**

Adapun pada penelitian ini terdapat batasan masalah yaitu :

1. Menganalisis *database system* informasi *E-Learning* pada jurusan teknologi informasi politeknik negeri jember
2. Data yang digunakan dalam analisis ini terdapat tiga prodi yakni Teknik komputer, manajemen informatika dan Teknik informatika.

### **1.4. Tujuan**

Adapun tujuan penelitian pada penelitian ini adalah sebagai berikut :

1. Dapat mempermudah kinerja *query database* sehingga kinerja *E- Learning* dapat optimal.
2. Meningkatkan kinerja pengolahan *database* pada *E- Learning* JTI Politeknik Negeri Jember.
3. Meningkatkan kualitas *query database E- Learning* JTI Politeknik Negeri Jember.

### **1.5. Manfaat**

Adapun manfaat pada penelitian ini adalah sebagai berikut :

1. Dapat mengoptimalkan kinerja *query database* sesuai Teknik *index* yang tepat.
2. Dapat optimal dalam menampilkan data kepada mahasiswa maupun dosen.

## BAB 2. TINJAUAN PUSTAKA

### 2.1. State Of The Art

Pada penelitian yang dilakukan, penulis mengacu pada studi literatur atau penelitian sebelumnya. Berikut ini merupakan penjabaran dari penelitian sebelumnya :

1. Optimalisasi *Query* dalam Basis Data My SQL Menggunakan Index (Ridho Pamungkas, 2018)

Pada penelitian yang dilakukan oleh Ridho Pamungkas diulas mengenai optimalisasi *query* sistem informasi menggunakan *index*. Pada *database* sistem informasi, indeks bermanfaat untuk mempercepat pencarian, jika *index* digunakan sesuai dengan kebutuhan. Penggunaan *index* dalam *database* secara sembarangan membuat kinerja dari *query* berat. Metode yang dipilih pada penelitian adalah metode eksperimental. Pada metode eksperimental memungkinkan sekali untuk dilakukan manipulasi dan meruban variabel serta melakukan mencari indeks yang paling optimal. Hasil dari penelitian menunjukkan indeks yang dirancang mampu meningkatkan kinerja dalam pencarian dalam baris data yang banyak.

2. Analisis Kinerja Optimasi *Query Cross Join, Natural Join Dan Full Outer Join* (Bismo Raharjo et al., 2018).

Pada penelitian yang dilakukan oleh Yohanes Aryo Bismo Raharjo dkk membahas analisis kinerja optimalisasi *query* pada *sintaks cross join, natural join* dan *full outer join*. Terdapat 3 tahapan dalam optimasi *query*, yaitu tahap persiapan, tahap pengujian dan tahap analisis hasil pengujian. Tahap persiapan dimulai dari pengumpulan data dan perancangan basis data. Pada Tahap pengujian penelitian dilakukan optimasi ketiga *sintaks query*. Algoritma untuk optimasi pada penelitian adalah tahap pengujian adalah *Query Hash Join* dan *Query Nested Join*. Pengujian data dilakukan pada baris data 1, 2, 4, 8,16, 32, 64, 128, 256, 512, 1024, 2048, 4096 dan 5000. Pada baris data 1, 4, 64, 128, 256, 512, 1024, 2048, 4096, dan 5000 menunjukkan *respons time sintaks natural join* lebih cepat dibanding *sintaks cross join* dan *full outer join*. Dan pada baris data 2, 8, 16, dan

32, *sintaks cross join* lebih unggul dibandingkan dari sintaks *natural join* dan *full outer join* .

3. Migrasi dan Optimalisasi *Database* Sistem Informasi berbasis *E-Learning* Program Diploma III Manajemen Informatika Universitas Muhammadiyah Metro (Sudarmaji, 2016)

Penelitian ini mengambil studi kasus dari Sistem Informasi berbasis E-learning ini merupakan hasil dari analisis dan observasi yang penulis lakukan terhadap sistem informasi yang sedang berjalan dan hasil perancangan yang baru dari perancangan Sistem Informasi Perkuliahan Online Pada Diploma-III Universitas Muhammadiyah Metro Kota Metro yang menggunakan *database*, dan menghasilkan sebuah antarmuka perkuliahan antara dosen dan mahasiswa. Secara garis besar penulis dapat mengemukakan bahwa sebagian sistem informasi yang sedang berjalan belum cukup baik, karena terdapat kekurangan dalam sistem tersebut, yaitu sistem SIAKAD dan e-learning yang tidak menjadi satu, sehingga dinilai kurang efisien dalam sistem perkuliahan sebuah universitas Muhammadiyah Metro.

Tabel 2. 1Studi Literatur

NO	Nama Peneliti	Judul	Hasil Penelitian
1	Ridho Pamungkas, 2018	Optimalisasi <i>Query</i> dalam Basis Data My SQL Menggunakan Index	Metode : <i>index</i> Variabel : basis data mysql Hasil : menentukan Keuntungan Menggunakan Skenario Index View
2	Bismo Raharjo et al., 2018	Analisis Kinerja Optimasi <i>Query Cross Join, Natural Join</i> Dan <i>Full Outer Join</i>	Metode : <i>Query Cross Join, Natural Join</i> Dan <i>Full Outer Join</i> Variabel : <i>database</i> penerbit buku.

---

			Hasil : membandingkan <i>time respon Query Cross Join, Natural Join Dan Full Outer Join</i> untuk menghasilkan yang paling optimal.
			Metode : MariaDB
			Variabel : <i>Database</i> Sistem Informasi berbasis <i>E-Learning</i> Program Diploma III Manajemen Informatika Universitas Muhammadiyah Metro
3	Sudarmaji, 2016	Migrasi dan Optimalisasi <i>Database</i> Sistem Informasi berbasis <i>E-Learning</i> Program Diploma III Manajemen Informatika Universitas Muhammadiyah Metro	Hasil : menggabungkan suatu sistem <i>E-Learning</i> menjadi satu bagian yang nantinya akan mempermudah kedua belah pihak mahasiswa dan dosen.
			Metode : Index
			Variabel : <i>Database E – Learning</i> JTI Politeknik Negeri Jember
4	Ega Kustian P. 2020	Optimalisasi <i>Query Database</i> JTI Politeknik Negeri Jember menggunakan Index	Hasil : Mengoptimalkan suatu <i>query database</i> pada sistem informasi <i>E-Learning</i> agar respons <i>time</i> lebih optimal.

---

## 2.2. Pengertian Sistem

Suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan

strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. Rohmat Taufiq, ST.,M.KOM (2013:1) ” Sistem berasal dari bahasa Latin (*systēma*) dan bahasa Yunani (*sustēma*) adalah suatu kesatuan yang terdiri komponen atau elemen”. Menurut Stephen A. Moscovice dan Mark G. Simkin dalam Jogiyanto H.M (1988:1) “Suatu sistem adalah suatu kesatuan yang terdiri dari interaksi subsistem yang berusaha untuk mencapai tujuan (*goal*) yang sama”.

Menurut Cole / Neuschel dalam Zaki Baridwan (1989:3) menyimpulkan sistem sebagai berikut : Sistem adalah suatu dari suatu kerangka dari prosedur-prosedur yang berhubungan yang di susun sesuai dengan suatu skema yang menyeluruh (terintegrasikan) untuk melaksanakan suatu kegiatan atau fungsi utama dari perusahaan. Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu.

### **2.3. Pengertian Informasi**

Telah diketahui bahwa informasi merupakan hal yang sangat penting bagi manajemen dalam pengambilan keputusan. Beberapa ahli mendefinisikan informasi sebagai berikut: Menurut Jogiyanto H.M, (2005:8) menyimpulkan informasi sebagai : Informasi adalah sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian yang nyata digunakan untuk pengambilan keputusan. Menurut Agus Mulyanto (2009:12) menyimpulkan informasi sebagai : Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya, sedangkan data merupakan sumber informasi yang menggambarkan suatu kejadian yang nyata.

### **2.4. Pengertian Sistem Informasi**

Sistem informasi merupakan suatu perkumpulan data yang terorganisasi beserta tata cara penggunaannya yang mencakup lebih jauh dari pada sekedar

penyajian. Istilah tersebut menyiratkan suatu maksud yang ingin dicapai dengan jalan memilih dan mengatur data serta menyusun tata cara penggunaannya. Keberhasilan suatu sistem informasi yang diukur berdasarkan maksud pembuatannya tergantung pada tiga faktor utama, yaitu : keserasian dan mutu data, pengorganisasian data, dan tata cara penggunaannya untuk memenuhi permintaan penggunaan tertentu, maka struktur dan cara kerja sistem informasi berbeda-beda bergantung pada macam keperluan atau macam permintaan yang harus dipenuhi. Suatu persamaan yang menonjol ialah suatu sistem informasi menggabungkan berbagai ragam data yang dikumpulkan dari berbagai sumber.

Menurut A-bahra bin Ladjamudin (2013:13) “sistem informasi adalah suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi”. Menyangkut pemahaman tentang pengertian sistem informasi ini, dalam bukunya, Agus Mulyanto (2009:29) mengutipkan beberapa pendapat para ahli, diantaranya:

1. Menurut James alter, sistem informasi adalah “kombinasi antar prosedur kerja, informasi, orang dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi”.
2. Menurut Bodnar dan Hopwood, sistem informasi adalah “kumpulan perangkat keras dan perangkat lunak yang dirancang untuk mentransformasikan data ke dalam bentuk informasi yang berguna”.
3. Menurut Gelinas, Oram dan Wiggins, sistem informasi adalah “suatu sistem buatan manusia yang secara umum terdiri atas sekumpulan komponen berbasis komputer dan manual yang dibuat untuk menghimpun, menyimpan, dan mengelola data serta menyediakan informasi keluaran kepada pemakai”.
4. Menurut Turban, McLean dan Waterbe, sistem informasi adalah “sistem yang mengumpulkan, memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk tujuan spesifik”.

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu



organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan.

## 2.5. Definisi Data

Data merupakan bahan dasar yang diolah untuk dijadikan suatu informasi yang akan lebih berguna dan bermanfaat bagi pemakai informasi tersebut untuk mencapai tujuan yang telah ditetapkan. Menurut Eddy Prahasta dalam bukunya Sistem Informasi Geografis (2002 : 30) menyatakan : “Data merupakan bahasa, *mathematical*, dan simbol-simbol penggantian lain yang disepakati umum dalam menggambarkan objek, manusia, peristiwa, aktivitas, konsep, dan objek-objek penting lainnya.”.

Menurut Jogianto Hartono, MBA, Ph. Dalam bukunya yang berjudul “Pengenalan Komputer” (1999:3), bahwa: “Data adalah kumpulan kejadian yang diangkat dari suatu kenyataan”. Sedangkan menurut Gardon B. Davis dalam buku Teguh Wahyantono, (2004:2), menyebutkan: “Data sebagian dari bahan mentah dari informasi yang dirumuskan sebagai kelompok limbang-limbang tidak acak yang menunjukkan jumlah atau tindakan dan hal-hal lainnya”.

### 2.5.1. Sejarah *E-Learning*

Sejarah *E-Learning* atau pembelajaran elektronik pertama kali diperkenalkan oleh Universitas Ilionis di Urbana-Champaign dengan menggunakan sistem instruksi berbasis komputer (*Computer Assisted Instruction*) dan komputer bernama PLATO. Sejak saat itu, perkembangan *E-Learning* berkembang sejalan dengan perkembangan dan kemajuan teknologi.

Berikut perkembangan *E-Learning* dari masa ke masa :

1. Tahun 1990 : Pada masa *Computer-Based Training* (CBT) di mana mulai bermunculan aplikasi *E-Learning* yang berjalan dalam PC *standlone* ataupun berbentuk kemasan CD-ROM. Isi materi dalam bentuk tulisan maupun multimedia (Video dan Audio) dalam format mov, mpeg- 1, atau avi.
2. Tahun 1994 : Seiring dengan diterimanya CBT oleh masyarakat sejak

tahun 1994 CBT muncul dalam bentuk paket-paket yang lebih menarik dan diproduksi secara masal.

3. Tahun 1997 : *Learning Management System (LMS)*. Seiring dengan perkembangan teknologi internet, masyarakat di dunia mulai terkoneksi dengan internet. Kebutuhan akan informasi yang dapat diperoleh dengan cepat mulai dirasakan sebagai kebutuhan mutlak dan jarak serta lokasi bukanlah halangan lagi. Dari sinilah muncul LMS. Perkembangan LMS yang makin pesat membuat pemikiran baru untuk mengatasi masalah *interoperability* antar LMS yang satu dengan lainnya secara standar. Bentuk standar yang muncul misalnya standar yang dikeluarkan oleh *Airline Industry CBT Commettee (AICC)*, IMS, IEEE LOM, ARIADNE. Tahun 1999 sebagai tahun aplikasi E-learning berbasis Web. Perkembangan LMS menuju aplikasi *e-learning* berbasis Web berkembang secara total, baik untuk pembelajar (*learner*) maupun administrasi belajar mengajarnya. LMS mulai digabungkan dengan situs-situs informasi, majalah dan surat kabar. Isinya juga semakin kaya dengan perpaduan multimedia, video *streaming* serta penampilan interaktif dalam berbagai pilihan format data yang lebih standar dan berukuran kecil.

#### 2.5.2. Pengertian *E-Learning*

E-learning adalah suatu sistem atau konsep pendidikan yang memanfaatkan teknologi informasi dalam proses belajar mengajar.

Berikut beberapa pengertian E-learning dari berbagai sumber:

1. Pembelajaran yang disusun dengan tujuan menggunakan sistem elektronik atau komputer sehingga mampu mendukung proses pembelajaran (Michael, 2013:27).
2. Proses pembelajaran jarak jauh dengan menggabungkan prinsip-prinsip dalam proses pembelajaran dengan teknologi (Chandrawati, 2010).
3. Sistem pembelajaran yang digunakan sebagai sarana untuk proses belajar mengajar yang dilaksanakan tanpa harus bertatap muka secara langsung antara dosen dengan mahasiswa (Ardiansyah,

2013).

### 2.5.3. Manfaat Perkuliahan Online

Ada 3 (tiga) fungsi pembelajaran elektronik terhadap kegiatan pembelajaran di dalam kelas (*Classroom instruction*), yaitu sebagai suplemen yang sifatnya pilihan, pelengkap (komplemen), atau pengganti (substitusi) .

1. Suplemen Dikatakan berfungsi sebagai suplemen (tambahan), apabila mahasiswa mempunyai kebebasan memilih, apakah akan memanfaatkan materi pembelajaran elektronik atau tidak. Dalam hal ini, tidak ada kewajiban/keharusan bagi mahasiswa untuk mengakses materi pembelajaran elektronik. Sekalipun sifatnya opsional, mahasiswa yang memanfaatkannya tentu akan memiliki tambahan pengetahuan atau wawasan.
2. Komplemen (Tambahan) Dikatakan berfungsi sebagai komplemen (pelengkap) apabila materi pembelajaran elektronik diprogramkan untuk melengkapi materi pembelajaran yang diterima mahasiswa di dalam kelas . Sebagai Komplemen berarti materi pembelajaran elektronik diprogramkan untuk menjadi materi *reinforcement* (pengayaan) atau remedial bagi mahasiswa di dalam mengikuti kegiatan pembelajaran konvensional. Materi pembelajaran elektronik dikatakan sebagai *enrichment*, apabila kepada mahasiswa yang dapat dengan cepat menguasai/memahami materi pelajaran yang disampaikan dosen secara tatap muka (+9) diberikan kesempatan untuk mengakses materi pembelajaran elektronik yang memang secara khusus dikembangkan untuk mahasiswa. Tujuannya agar semakin memantapkan tingkat penguasaan mahasiswa terhadap materi pelajaran yang disajikan dosen di dalam kelas. Dikatakan sebagai program perbaikan, apabila kepada mahasiswa yang mengalami kesulitan memahami materi pelajaran yang disajikan dosen secara tatap muka di kelas (*Slow learners*) diberikan kesempatan untuk memanfaatkan materi pembelajaran elektronik yang memang secara khusus dirancang untuk mahasiswa.
3. Pengganti (Substitusi) Beberapa pendosenan tinggi di Negara-negara maju memberikan beberapa alternatif model kegiatan pembelajaran / perkuliahan kepada para mahasiswanya. Tujuannya agar para mahasiswa dapat secara

fleksibel mengelola kegiatan perkuliahannya sesuai dengan waktu dan aktivitas lain sehari-hari mahasiswa. (Sudarmaji, 2016).

## 2.6. Pengenalan *Database*, DBMS dan RDBMS

Basis data (atau *database*) adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut (<http://id.wikipedia.org/wiki/Database>). *Database* digunakan untuk menyimpan informasi atau data yang terintegrasi dengan baik di dalam komputer.

Untuk mengelola *database* diperlukan suatu perangkat lunak yang disebut DBMS (*Database Management System*). DBMS merupakan suatu sistem perangkat lunak yang memungkinkan *user* (pengguna) untuk membuat, memelihara, mengontrol, dan mengakses *database* secara praktis dan efisien. Dengan DBMS, *user* akan lebih mudah mengontrol dan memanipulasi data yang ada.

Sedangkan RDBMS atau *Relationship Database Management System* merupakan salah satu jenis DBMS yang mendukung adanya *relationship* atau hubungan antar tabel. Di samping RDBMS, terdapat jenis DBMS lain, misalnya *Hierarchy* DBMS, *Object Oriented* DBMS, dsb. Beberapa software atau perangkat lunak DBMS yang sering digunakan dalam aplikasi program antara lain :

1. DB2 - <http://www-306.ibm.com/software/data/db2/>
2. Microsoft SQL Server - <http://www.microsoft.com/sql/>
3. Oracle - <http://www.oracle.com>
4. Sybase - <http://www.sybase.com/>
5. Interbase - <http://www.borland.com/interbase>
6. Teradata - <http://www.teradata.com/>
7. Firebird - <http://www.firebirdsql.org/>
8. MySQL – <http://www.mysql.com>

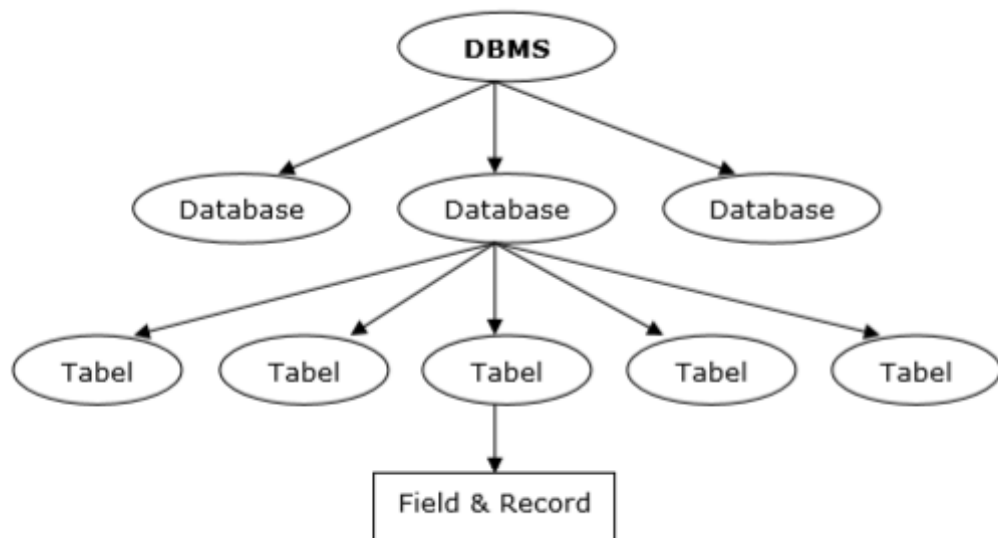
## 9. PostgreSQL - <http://www.postgresql.org/>

### Beberapa Istilah *Database*

1. *Table* merupakan kumpulan data (nilai) yang diorganisasikan ke dalam baris (*record*) dan kolom (*field*). Masing-masing kolom memiliki nama yang spesifik dan unik.
2. *Field* merupakan kolom dari sebuah *table*. *Field* memiliki ukuran *type* data tertentu yang menentukan bagaimana data nantinya tersimpan.
3. *Record Field* merupakan sebuah kumpulan nilai yang saling terkait.
4. *Key* merupakan suatu *field* yang dapat dijadikan kunci dalam operasi tabel. Dalam konsep *database*, *key* memiliki banyak jenis diantaranya *Primary Key*, *Foreign Key*, *Composite Key*, dll.
5. SQL atau *Structured Query Language* merupakan suatu bahasa (*language*) yang digunakan untuk mengakses *database*. SQL sering disebut juga sebagai *query*.

### 2.6.1. Hierarki *Database*

Dalam konsep *database*, urutan atau hierarki *database* sangatlah penting. Urutan atau hierarki *database* digambarkan dalam gambar sbb :



Gambar 2.1 Hirarki *Database*

### 2.6.2. Mysql

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi *GNU General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. Tidak seperti PHP atau *Apache* yang merupakan software yang dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh penulisnya masing-masing, MySQL dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia yaitu MySQL AB. MySQL AB memegang penuh hak cipta hampir atas semua kode sumbernya. Kedua orang Swedia dan satu orang Finlandia yang mendirikan MySQL AB adalah: David Axmark, Allan Larsson, dan Michael "Monty" Widenius. MySQL dapat *download* di situs resminya, <http://www.mysql.com>.

Fitur-fitur MySQL antara lain :

1. *Relational Database System*. Seperti halnya *software database* lain yang ada di pasaran, MySQL termasuk RDBMS.
2. *Arsitektur Client-Server*. MySQL memiliki arsitektur *client-server* dimana *server database* MySQL terinstal di server. *Client* MySQL dapat berada di komputer yang sama dengan server, dan dapat juga di komputer lain yang berkomunikasi dengan server melalui jaringan bahkan internet.
3. Mengenal perintah SQL standar. SQL (*Structured Query Language*) merupakan suatu bahasa standar yang berlaku di hampir semua *software database*. MySQL mendukung SQL versi SQL:2003.
4. Mendukung *Sub Select*. Mulai versi 4.1 MySQL telah mendukung *select* dalam *select (sub select)*.

5. Mendukung *Views*. MySQL mendukung *views* sejak versi 5.0
6. Mendukung *Stored Prosedured (SP)*. MySQL mendukung SP sejak versi 5.0
7. Mendukung *Triggers*. MySQL mendukung *trigger* pada versi 5.0 namun masih terbatas. Pengembang MySQL berjanji akan meningkatkan kemampuan *trigger* pada versi 5.1.
8. Mendukung *replication*.
9. Mendukung transaksi.
10. Mendukung *foreign key*.(Solichin & Luhur, 2015).

## 2.7. Mengenal SQL ( *Structured Query Language* )

SQL ( *Structured Query Language* ) adalah sebuah bahasa permintaan *database* yang terstruktur. Bahasa SQL ini dibuat sebagai bahasa yang dapat merelasikan beberapa tabel dalam *database* maupun merelasikan antar *database*. SQL dibagi menjadi tiga bentuk *Query*, yaitu :

1. DDL ( *Data Definition Language* ) DDL adalah sebuah metode *Query* SQL yang berguna untuk mendefinisikan data pada sebuah *Database*, *Query* yang dimiliki DDL adalah :
  1. CREATE : Digunakan untuk membuat *Database* dan Tabel
  2. Drop : Digunakan untuk menghapus Tabel dan *Database*
  3. Alter : Digunakan untuk melakukan perubahan struktur tabel yang telah dibuat, baik menambah *Field* ( *Add* ), mengganti nama *Field* ( *Change* ) ataupun menamakannya kembali ( *Rename* ), dan menghapus *Field* (Drop ).
2. DML ( *Data Manipulation Language* ) DML adalah sebuah metode *Query* yang dapat digunakan apabila DDL telah terjadi, sehingga fungsi dari *Query* DML ini untuk melakukan pemanipulasian *database* yang telah dibuat. *Query* yang dimiliki DML adalah :
  1. INSERT : Digunakan untuk memasukkan data pada Tabel *Database*

2. UPDATE : Digunakan untuk pengubahan terhadap data yang ada pada *Tabel Database*
3. DELETE : Digunakan untuk Penghapusan data pada tabel *Database*
3. DCL ( *Data Control Language* ) DCL adalah sebuah metode *Query SQL* yang digunakan untuk memberikan hak otorisasi mengakses *Database*, mengalokasikan *space*, pendefinisian *space*, dan pengauditan penggunaan *database*. *Query* yang dimiliki DCL adalah :
  1. GRANT : Untuk mengizinkan User mengakses Tabel dalam *Database*.
  2. REVOKE : Untuk membatalkan izin hak *user*, yang ditetapkan oleh perintah GRANT
  3. COMMIT : Menetapkan penyimpanan *Database*
  4. ROLLBACK : Membatalkan penyimpanan *Database*. (Saputro, 2012).

## 2.8. Optimasi *Query*

Optimasi *Query* adalah suatu proses yang digunakan untuk menganalisa *query* dalam menentukan sumber-sumber apa saja yang digunakan oleh *query* tersebut dan apakah penggunaan dari sumber tersebut dapat dikurangi tanpa merubah *output* atau bisa juga dikatakan bahwa optimasi *query* adalah sebuah prosedur untuk meningkatkan strategi evaluasi dari suatu *query* untuk membuat evaluasi tersebut menjadi lebih efektif. Optimasi *query* mencakup beberapa teknik seperti transformasi *query* ke dalam bentuk logika yang sama, memilih jalan akses yang tercepat dan mengoptimalkan penyimpanan data. (Dewi et al., 2017)

Optimalisasi *query* merupakan sebuah prosedur untuk meningkatkan strategi evaluasi proses akses data menjadi lebih efektif. Optimasi *query* mencakup beberapa teknik seperti transformasi *query* ke dalam bentuk logika yang sama, memilih akses data yang tercepat dan mengoptimalkan penyimpanan data. Adapun langkah-langkah yang dilakukan dalam penelitian ini meliputi :

1. Analisa objek, objek yang digunakan adalah sebuah perguruan tinggi dengan kriteria memiliki sejumlah mahasiswa , sejumlah mata kuliah, sejumlah kelas dan sejumlah dosen yang dijadikan sebagai data informasi terkait nilai mahasiswa. Objek memiliki aturan yang harus terpenuhi untuk



menampilkan informasi nilai mahasiswa.

2. Penggalian data, data-data asli dianalisis dan dilakukan analisa awal tentang kecenderungan adanya relasi antar data. Kecenderungan relasi ini harus bisa dimodelkan menggunakan alat pemodelan data berupa ER-Diagram. Pada saat implementasi menggunakan data simulasi, karena data-data dari sebuah perguruan tinggi bersifat privasi.
3. Penemuan akar masalah, berdasarkan analisa data dan objek terkait, akar permasalahan harus dapat diidentifikasi dengan baik. Misalnya masalah data telah terintegrasi dengan baik tetapi belum ada langkah optimal dalam prosesnya. Sehingga akan berdampak pada lamanya proses *query*.
4. Menentukan solusi sementara, solusi sementara ini akan memberikan deskripsi terhadap langkah awal penggunaan data yang telah dilakukan analisa optimalisasi dengan metode tertentu. Maka akan menghemat biaya proses dan memperlebar pita konektivitas akses data.
5. Perancangan basis data, dalam tahap ini merupakan tahap paling penting dalam perancangan *database relational*. Media yang digunakan adalah ER-Diagram dan dipetakan dalam bentuk skema relasional basis data.
6. Implementasi basis data, untuk menganalisis kecepatan proses rancangan basis data maka hasil rancangan diimplementasikan dalam sebuah DBMS. DBMS yang digunakan adalah MySQL karena bersifat bebas pakai non komersial dengan lisensi publik.
7. Analisa menggunakan bahasa SQL, untuk mendefinisikan data, memanipulasi dan mengontrol akses, diperlukan DBMS yang memiliki bahasa SQL yang kuat. Penggunaan bahasa SQL menggunakan analisa model eksperimen dengan *trial and error*.
8. Analisa optimalisasi query, setiap hasil dari pemrosesan query dianalisa lama prosesnya. Semakin lama prosesnya diasumsikan bahwa akan memakan banyak biaya sumber daya komputasi yang ada. Sumber daya komputasi yang dipakai adalah kapasitas memori penyimpanan sementara RAM, kecepatan prosesor serta lebar pita transmisi data dan medianya. (Fatkhurrochman1 et al., 2018).

## 2.9. Index

Seperti dibahas sebelumnya, *index* pada *database* sama halnya seperti daftar isi pada sebuah buku untuk memudahkan dalam pencarian data. Karena apabila tidak menggunakan *index* pada suatu *database*, maka *database* akan melakukan *scanning* atau pencarian dari setiap baris pada tabel yang diinginkan. *Scanning* atau pencarian tersebut akan memakan waktu yang lama dan akan memperlambat kinerja suatu *database*. Dalam SQL dapat membuat *index* dengan satu kolom atau beberapa kolom apabila *index* tersebut dipanggil. Jadi hasil dari *index* tersebut dapat berupa satu kolom (*column index*) dari sebuah tabel, atau

### 2.9.1. Column Indexes (Index Satu Kolom)

Semua tipe data *MySQL* dapat *diindex*-kan. Menggunakan *index* pada kolom yang sesuai merupakan cara terbaik untuk meningkatkan performa dari operator *SELECT*. Nomor maksimal *index* setiap tabel dan panjang *index* terdefinisi berdasarkan penyimpanan (*Storage Engine*).

### 2.9.2. Multiple-Column Indexes (Index Beberapa Kolom)

*MySQL* dapat membuat *index* gabungan (maksudnya membuat *index* dengan *query* memanggil dua kolom dari sebuah data atau tabel). *Multiple-Column index* yang dibuat maksimum 16 kolom. *Multiple-Column index* dapat dikatakan sebagai *array* yang mengandung nilai – nilai yang digabungkan dengan nilai pada kolom yang dibuat *index*.

*MySQL* menggunakan *index* untuk melakukan operasi-operasi sebagai berikut:

1. Untuk mencari baris-baris yang sesuai dengan klausa *WHERE* dengan cepat
2. Untuk mengeliminasi baris-baris dari pertimbangan. *MySQL* akan mencari *index* dengan nomor baris yang terkecil.
3. Untuk mengambil baris pada kolom yang telah digabungkan sebelumnya.

Dalam beberapa kasus, *query* dapat di optimasi untuk mengambil nilai-nilai tanpa membuka tabelnya terlebih dahulu. (Pamungkas, 2018).

## 2.10. Pemrograman Bahasa *Python*

*Python* merupakan salah satu bahasa pemrograman yang dirilis oleh Van Rossum (1991, 1995) dengan *sintaks* yang sederhana, (Rahmatullah et al., 2020).

*Python* telah menjadi salah satu bahasa pemrograman paling populer di dunia dalam beberapa tahun terakhir. Bahasa pemrograman ini digunakan dalam segala hal mulai dari *machine learning*, membangun situs web, dan pengujian software. *Python* dapat digunakan oleh seorang developer dan juga non-developer.

*Python*, salah satu bahasa pemrograman paling populer di dunia, telah menciptakan segalanya mulai dari algoritma rekomendasi *Netflix* hingga software yang mampu mengontrol mobil yang bisa mengemudi sendiri. *Python* adalah bahasa pemrograman yang dirancang untuk digunakan dalam berbagai aplikasi, termasuk ilmu data, pengembangan software dan *website*, otomatisasi, serta umumnya mampu menyelesaikan pekerjaan lainnya.

## **BAB 3. METODOLOGI PENELITIAN**

### **3.1. Tempat dan Waktu Penelitian**

#### 3.1.1. Tempat Penelitian

Tempat penelitian yang digunakan dalam proses penelitian ini berada di Politeknik Negeri Jember

#### 3.1.2. Waktu Penelitian

Waktu penelitian yang dilakukan dalam proses penelitian ini yaitu selama 8 bulan.

### **3.2. Alat dan Bahan**

#### 3.2.1. Alat

Alat-alat yang digunakan dalam penelitian ini yaitu perangkat keras dan perangkat lunak :

##### a. Perangkat keras :

Laptop : Asus VivoBook A442U  
RAM 12 GB  
Processor Intel Core i5 8<sup>th</sup> Gen  
NVIDIA GEFORCE

##### b. Perangkat lunak :

Windows 10  
MySQL  
Pyhton Jupiter Notebook  
Visual Studio Code

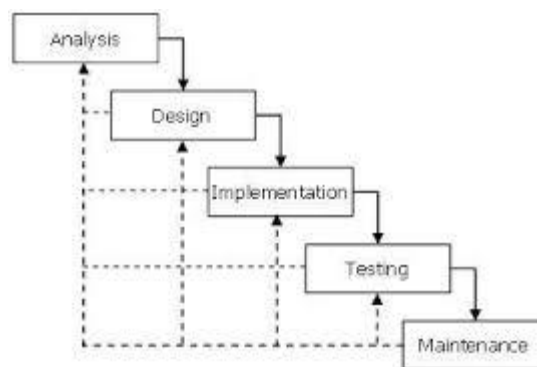
#### 3.2.2. Bahan

Bahan yang digunakan dalam penelitian ini adalah *database* pada *E-Learning* jurusan teknologi informasi Politeknik Negeri Jember yang didalamnya terdapat tiga prodi yakni Teknik informatika, manajemen informatika, dan teknik komputer. Bahan pada penelitian ini didapatkan melalui proses studi pustaka, wawancara terhadap narasumber bapak Ery

Setiyawan Jullev Atmadji, S.Kom, M.Cs.

### 3.3. Tahapan Penelitian

Pada tahapan penelitian ini menggunakan tahapan penelitian model *waterfall*. Model *waterfall* merupakan model klasik yang bersifat sistematis dan berurutan dalam membangun *sistem*. Disebut model *waterfall* karena tahap demi tahap menunggu selesainya tahap sebelumnya secara berurutan (Pressman, 2015).



Gambar 3.1 Model *Waterfall* (Bassil,2012)

#### 3.3.1. Tahapan tahapan metode *waterfall*

Berikut ini merupakan tahapan-tahapan metode *waterfall* :

##### 1. *Analysis*

Mencari informasi serta mengumpulkan data mengenai permasalahan yang sering dihadapi dalam *time respon* atau *query database* pada *E – Learning* JTI Politeknik Negeri Jember. Proses ini dapat dilakukan dengan cara studi literatur, wawancara, dan observasi. Optimalisasi *Query database* menggunakan *index* ini membantu untuk mempercepat *respon time* lebih optimal.

##### 2. *Design*

Pada tahapan ini merupakan suatu gambaran sistem yang akan diterapkan dalam pengembangan sistem optimalisasi *query database e – learning* JTI Politeknik Negeri Jember menggunakan *index*. Desain sistem yang akan diterapkan pada proses pengembangan ini yaitu *flowchart*, *ERD*.

### 3. *Implementation*

Tahapan ini merupakan tahap implementasi dari tahap sebelumnya yang akan diterapkan dalam pengembangan optimalisasi *query database e – learning* JTI Politeknik Negeri Jember menggunakan *index*. Pada tahapan ini dilakukan beberapa hal sebagai berikut :

- a) *Upload database* data E – Learning JTI Politeknik Negeri Jember pada *mysql*.
- b) Proses seleksi tabel - tabel mana yang diberikan *query index*.

### 4. *Testing*

Sebuah *sistem* yang telah dibuat akan dilakukan pengujian dengan *index* dan tidak menggunakan *index* pada tabel *tabel* yang telah dipilih yakni ada 64 tabel, sehingga hasil dari itu akan mengetahui perbandingan antara *query* dan tanpa *query*.

### 5. *Maintenance*

Tahap pemeliharaan software yang telah dibuat dengan melakukan pengecekan terhadap sistem untuk mencari tahu apabila terdapat fungsi-fungsi yang bermasalah atau *error*, sehingga dapat dilakukan perbaikan.

#### 3.3.2. Pengumpulan Data

Dalam penelitian yang dilakukan penulis menggunakan beberapa metode yang dilakukan untuk mendapatkan data yang tepat guna mencapai kesempurnaan sistem yang dibuat, adapun metode yang digunakan adalah sebagai berikut :

##### 1. Metode Studi Pustaka

Merupakan suatu metode yang pengumpulan data dan informasi pengetahuan yang didapatkan dari buku, jurnal, aplikasi ataupun internet yang berkaitan dengan topik penelitian. Pada penelitian ini

##### 2. Metode Wawancara

Merupakan suatu metode pengumpulan data yang dilakukan dengan cara berkomunikasi atau berdiskusi dengan narasumber mengenai permasalahan yang sering terjadi pada *respon time query database* pada *E – Learning* JTI Politeknik Negeri Jember. Proses wawancara pada penelitian ini dilakukan kepada Bapak Ery Setiyawan Jullev Atmadji,

S.Kom, M.Cs selaku Dosen Teknologi Informasi di Politeknik Negeri Jember.

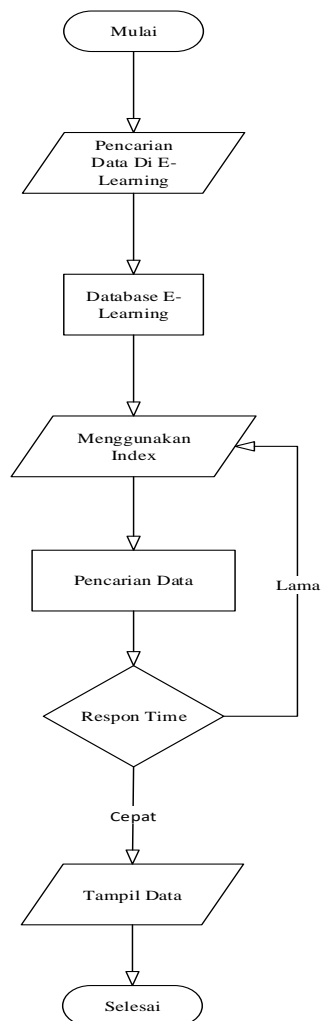
### **3.4. Rancangan Sistem**

Rancangan sistem merupakan kebutuhan penting yang menggambarkan bagaimana sebuah sistem dapat berjalan sesuai dengan kebutuhan yang diinginkan oleh pengguna (*user*). Sehingga dapat memudahkan *user* dalam memahami proses berjalannya sebuah sistem.

#### 3.4.1. Flowchart

*Flowchart* terdiri dari simbol-simbol tertentu yang menggambarkan jalannya proses secara berurutan yang menghubungkan proses satu dengan proses lainnya. Pada optimalisasi *query database E – Learning* JTI Politeknik Negeri Jember menggunakan *index* ini memiliki *flowchart* proses dan *flowchart* testing pada gambar berikut :

## 1. Flowchart Proses Index



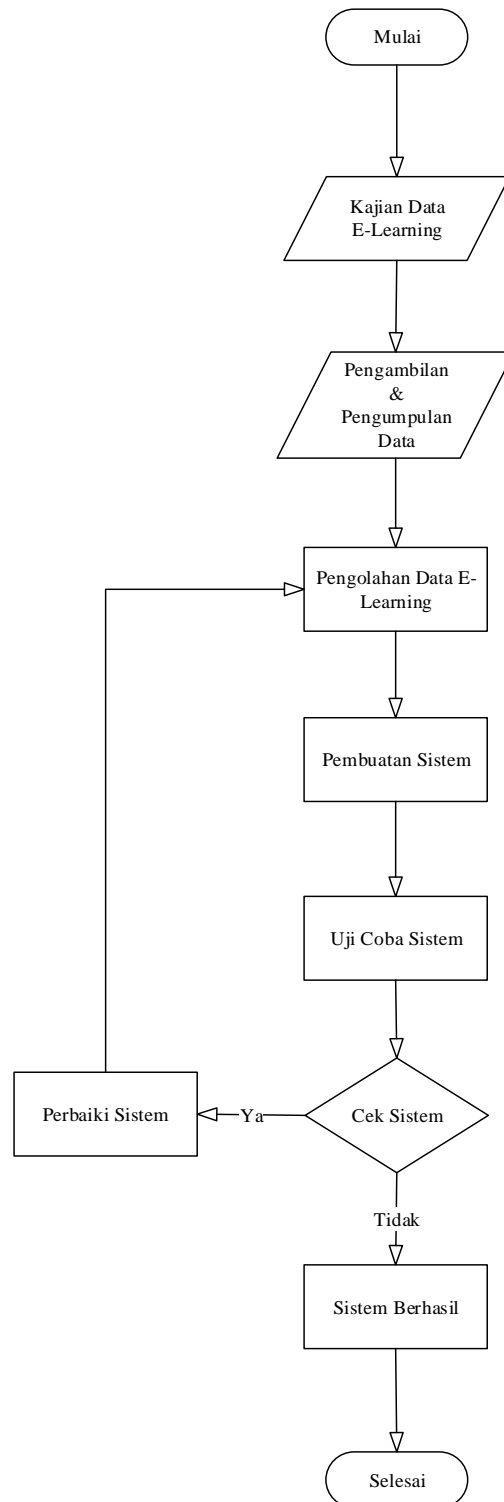
Gambar 3.2 Flowchart proses index

Gambar flowchart diatas merupakan proses berjalannya sistem ketika database E – Learning JTI Politeknik Negeri Jember menggunakan metode index dan tidak dari hasil tersebut dapat mengetahui hasil dari *respon time* mana yang lebih optimal.

Gambar flowchart diatas merupakan flowchart proses yang dimana sistem ketika database E – Learning JTI Politeknik Negeri Jember menggunakan metode index dan tidak dari hasil tersebut dapat menampilkan data hasil dari *respon time* yang lebih optimal.



## 2. Flowchart Testing Index



Gambar 3.3 Rancangan *flowchart testing system optimalisasi query database E – Learning* JTI Politeknik Negeri Jember menggunakan *index*

Gambar *flowchart testing system* diatas merupakan *flowchart testing system E-Learning* yang dimulai dari pengkajian data *E – Learning* JTI Politeknik Negeri Jember sampai berhasilnya sistem tersebut. Dengan menggunakan proses *index* pada *database E – Learning* JTI Politeknik Negeri Jember.

## BAB 4. HASIL DAN PEMBAHASAN

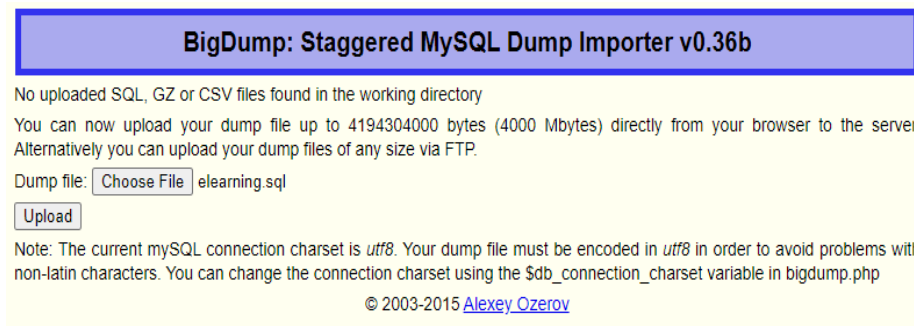
Pada Bab IV ini akan dikaji mengenai hasil penelitian yang meliputi analisis suatu proses *load database e – learning* JTI Politeknik Negeri Jember dengan menggunakan *query index*, serta membandingkan dengan *load database e – learning* JTI tanpa menggunakan *query index*.

### 4.1. Data Percobaan

Data yang diambil dalam percobaan ini yakni data pada *database e – learning* JTI Politeknik Negeri Jember. Tujuan dari pengambilan data ini yakni ingin mengetahui seberapa berpengaruhnya suatu sistem *e – learning* JTI jika *database* diberi *query index* dan membandingkan *load database* yang tanpa menggunakan *query index*. Nantinya akan diimplementasikan dalam sistem informasi *E – Learning* JTI. Adapun total jumlah *database* yang ada pada *e – learning* yakni 419 tabel, namun hanya ada 64 tabel yang akan diimplementasikan *query index*. Agar suatu sistem tersebut berjalan lebih optimal dari sebelumnya.

Faktor banyaknya user yang mengakses berkaitan dengan respon time dan tingkat *load* data pada sistem *E – Learning* lebih lama. Oleh karena itu dengan ini melakukan percobaan *query index* terhadap *database e – learning* yang nanti berguna untuk mengoptimalkan suatu *load* data pada sistem tersebut.

Proses *upload database e – learning* ini menggunakan *bigdump*. *Bigdump* sendiri adalah sebuah *script* yang digunakan untuk *import database* yang ukurannya cukup besar ke *mysql*. Hal ini mengatasi suatu tabel yang ada di *database* tidak ketinggalan *terupload*. Waktu proses *upload database* itu sendiri lebih cepat dari pada menggunakan *upload* secara manual di *mysql* langsung,



Gambar 4.1 Proses *upload database* menggunakan *bigdump*

#### 4.2. Proses Data

Berdasarkan data yang telah terupload di mysql, tahap selanjutnya yaitu melakukan proses menggunakan *moodle* untuk melihat alur suatu sistem E – learning JTI Politeknik Negeri Jember. Tujuan tahap ini untuk mengetahui seberapa lama *response time* jika belum menggunakan *index*. *Moodle* adalah paket perangkat lunak yang diproduksi untuk kegiatan belajar berbasis internet dan situs web yang menggunakan prinsip *social constructionist pedagogy*. *Moodle* merupakan salah satu aplikasi dari konsep dan mekanisme belajar mengajar yang memanfaatkan teknologi informasi, yang dikenal dengan konsep pembelajaran elektronik atau e-learning. *Moodle* dapat digunakan secara bebas sebagai produk sumber terbuka (*open source*) di bawah lisensi GNU. *Moodle* dapat *diinstal* di komputer dan sistem operasi apapun yang bisa menjalankan PHP dan mendukung *database SQL*.

*Moodle* juga menyediakan sebuah platform yang berhubungan langsung dengan *mysql local*. *Disini* sangat menguntungkan karena dari *online* kita dapat mengetahui hubungan relasi antar *table* yang ada di *moodle* tersebut. Dari hal tersebut dapat mempermudah untuk mengetahui tabel mana saja yang penting diberi *query index* di *database* tersebut.



Gambar 4.2 Moodle

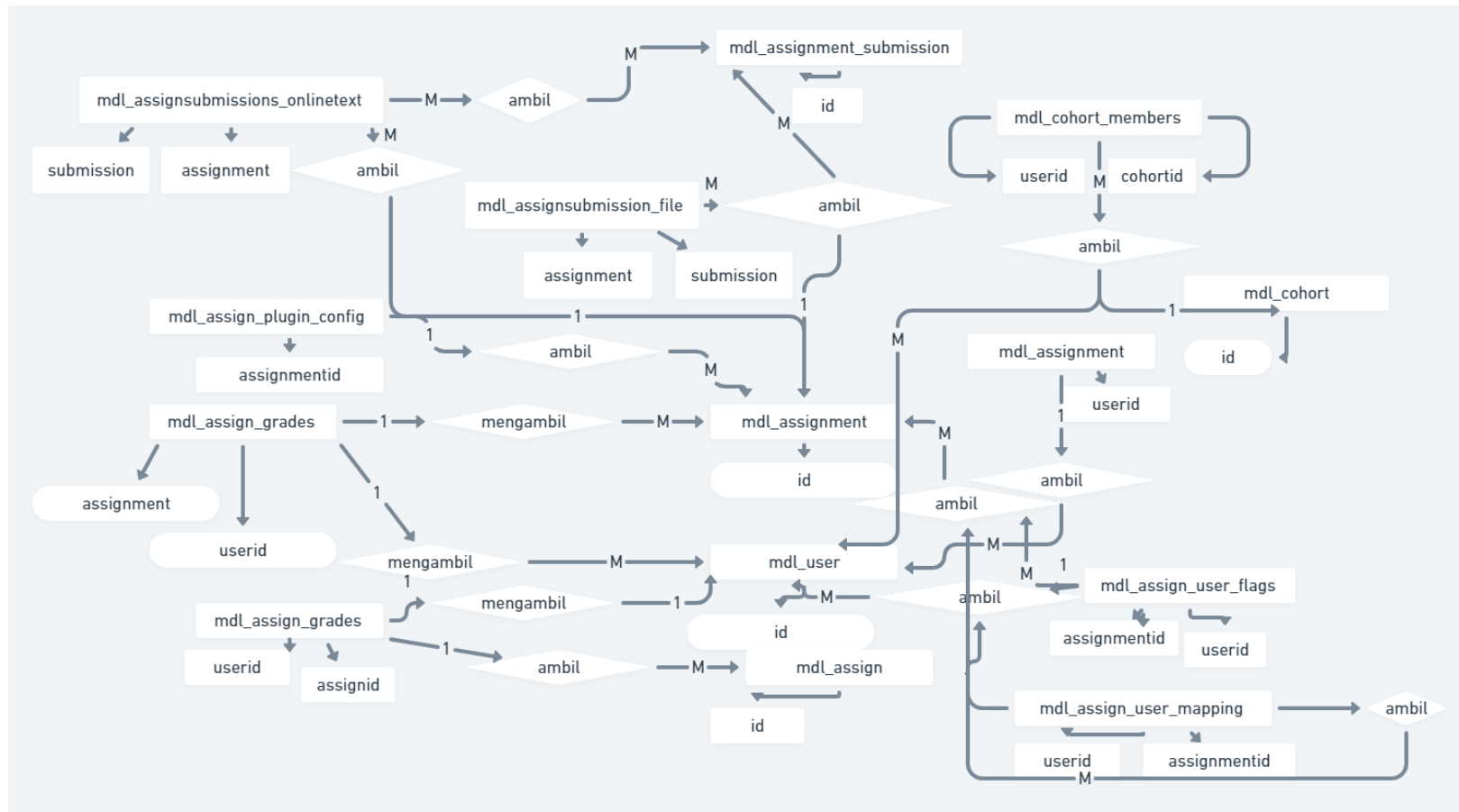
### 4.3. Tahapan Metode *Waterfall*

#### 4.3.1. *Analysis*

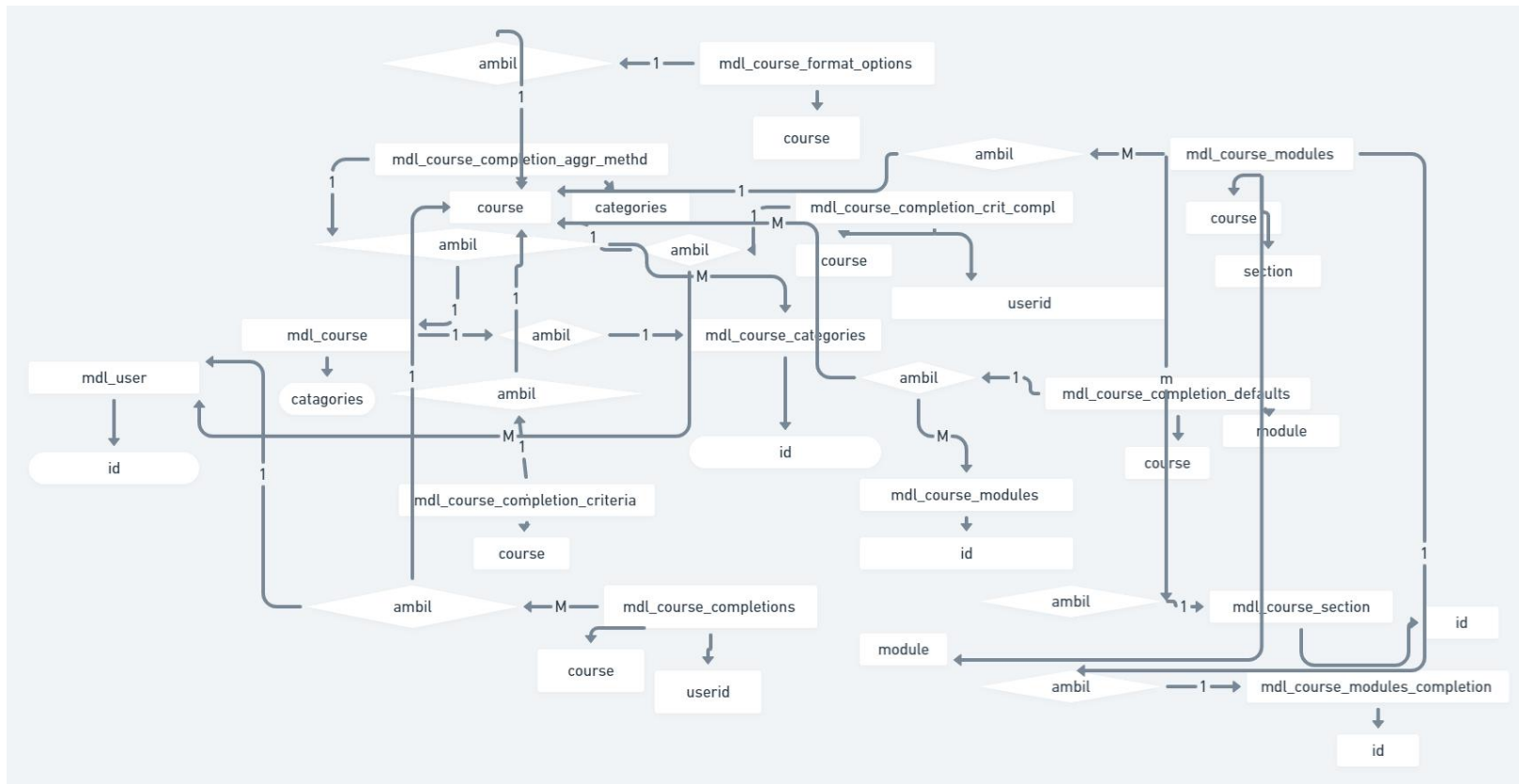
Mencari informasi serta mengumpulkan data mengenai permasalahan yang sering dihadapi dalam *time respon* atau *query database* pada *E – Learning* JTI Politeknik Negeri Jember. Proses ini dapat dilakukan dengan cara studi literatur, wawancara, dan observasi. Optimalisasi *Query database* menggunakan *index* ini membantu untuk mempercepat *respon time* lebih optimal.

#### 4.3.2. *Design*

Pada tahapan ini merupakan suatu gambaran sistem yang akan diterapkan dalam pengembangan sistem optimalisasi *query database e – learning* JTI Politeknik Negeri Jember menggunakan *index*. Desain sistem yang akan diterapkan pada proses pengembangan ini yaitu *flowchart*, *ERD*



Gambar 4.3 ERD database E – learning JTI

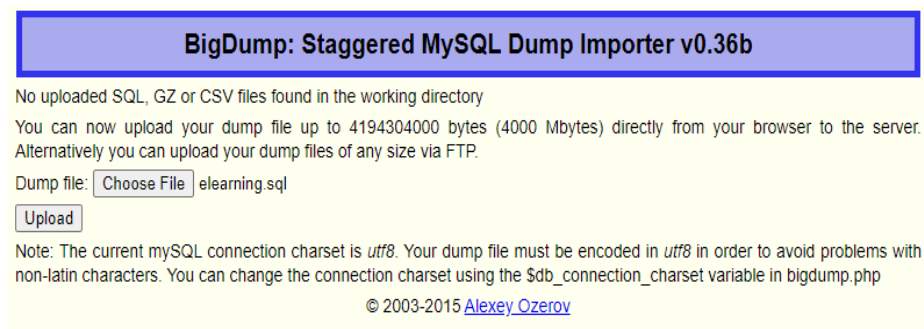


Gambar 4.4 Database E – learning JTI

### 4.3.3. Implementation

Tahapan ini merupakan tahap implementasi dari tahap sebelumnya yang akan diterapkan dalam pengembangan optimalisasi *query database e – learning* JTI Politeknik Negeri Jember menggunakan *index*. Pada tahapan ini dilakukan beberapa hal sebagai berikut :

- 1) *Upload database data E – Learning JTI Politeknik Negeri Jember pada mysql.*



Gambar 4.5 Upload database

- 2) *Proses seleksi tabel - tabel mana yang diberikan query index.*
  - a. Awal lakukan *upload* dua kali *database* yang diberikan oleh narasumber data administrator *e – learning* JTI Politeknik Negeri Jember. Untuk *database* awal dengan nama *e -learning* yang mewakili *database* berindex dan kedua dengan nama *e - learning\_kw* yang mewakili *database* tanpa *index*. Dari 419 tabel kedua database tersebut database pertama dilakukan pengecekan setiap tabelnya dimana saja kolom tabel yang sering menggunakan *klausula where* atau dalam kondisi *join* dengan tabel lainnya, kolom yang berisi nilai dengan jangkauan yang luas dan banyak data dalam tabel tersebut setelah itu, dicek lagi tabel mana yang berukuran besar memungkinkan menggunakan *query*. Setelah itu dari hasil itu ditemukan ada 7 tabel utama yakni :

1. mdl\_assign
2. mdl\_assignment
3. mdl\_course



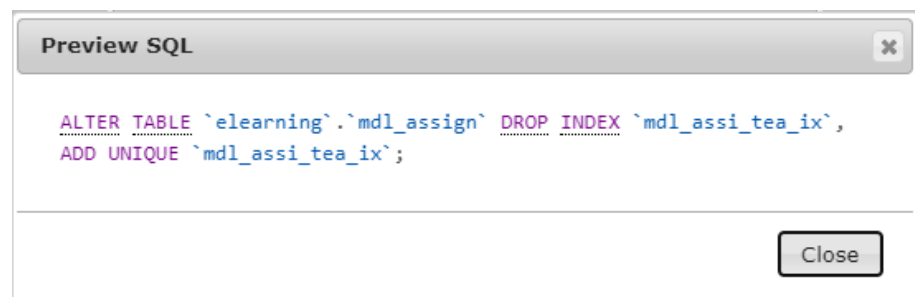
4. mdl\_cohort
5. mdl\_lesson
6. mdl\_user
7. mdl\_quiz

Dari data tabel 7 diatas dikembangkan lagi menjadi, mdl\_assign ada 17 tabel, mdl\_assignment ada dua tabel, mdl\_course ada 13 tabel, mdl\_cohort ada dua tabel, mdl\_lesson ada delapan tabel, mdl\_user ada 11 tabel, dan mdl\_quiz ada 11 tabel. Struktur *database E- learning* JTI Politeknik Negeri Jember terdiri dari 419 tabel yang di seleksi menjadi 64 yakni sebagai berikut :

1. mdl\_assign\_grades
2. mdl\_assign\_overrides
3. mdl\_assign\_plugin\_config
4. mdl\_assign\_submission
5. mdl\_assign\_user\_flags
6. mdl\_assign\_user\_mapping
7. mdl\_assignment\_upgrade
8. mdl\_assignsubmission\_file
9. mdl\_assignsubmissions\_onlinetext
10. mdl\_cohort
11. mdl\_cohort\_members
12. mdl\_course
13. mdl\_course\_categories
14. mdl\_course\_completion\_aggr\_methd
15. mdl\_course\_completion\_crit\_compl
16. mdl\_course\_completion\_criteria
17. mdl\_course\_completion\_defaults
18. mdl\_course\_completions
19. mdl\_course\_format\_options
20. mdl\_course\_modules

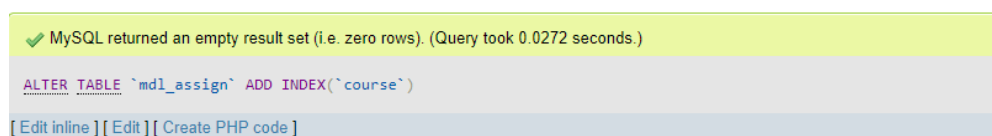
21. mdl\_course\_modules completion
22. mdl\_course\_published
23. mdl\_course\_request
24. mdl\_course\_sections
25. mdl\_lesson
26. mdl\_lesson\_answers
27. mdl\_lesson\_attempts
28. mdl\_lesson\_branch
29. mdl\_lesson\_grades
30. mdl\_lesson\_overrider
31. mdl\_lesson\_pages
32. mdl\_lesson\_timer
33. mdl\_quiz
34. mdl\_quiz reports
35. mdl\_quiz\_attempts
36. mdl\_quiz\_feedback
37. mdl\_quiz\_grades
38. mdl\_quiz\_overrides
39. mdl\_quiz\_overview\_regredes
40. mdl\_quiz\_sections
41. mdl\_quiz\_slots
42. mdl\_quiz\_slots\_tags
43. mdl\_quiz\_statistic
44. mdl\_user
45. mdl\_user\_devices
46. mdl\_user\_enrolments
47. mdl\_user\_info\_category
48. mdl\_user\_info\_field
49. mdl\_user\_lastaccess
50. mdl\_user\_password\_history
51. mdl\_user\_password\_resets

52. mdl\_user\_preferences
53. mdl\_user\_private\_key
54. mdl\_assign
55. mdl\_assignfeedback\_comments
56. mdl\_assignfeedback\_editpdf\_annot
57. mdl\_assignfeedback\_editpdf\_cmnt
58. mdl\_assignfeedback\_editpdf\_queue
59. mdl\_assignfeedback\_editpdf\_quick
60. mdl\_assignfeedback\_editpdf\_rot
61. mdl\_assignfeedback\_file
62. mdl\_assignment
63. mdl\_user\_info\_data
64. mdl\_assignment\_submissions



Gambar 4.6 Query index

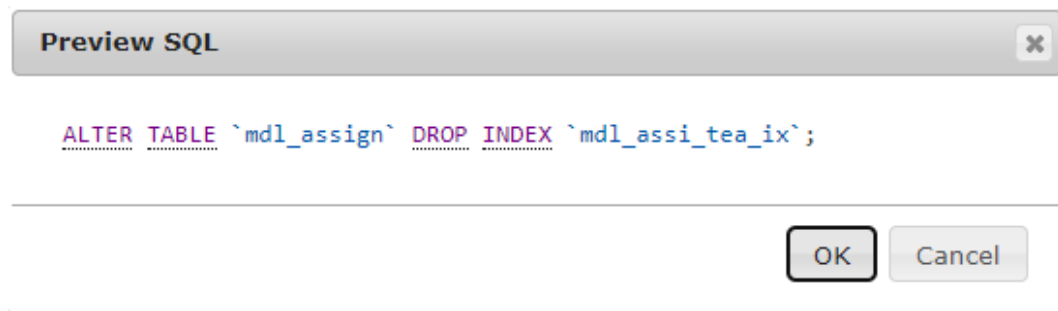
- b. Tahap kedua dari 64 tabel diatas dilakukan suatu pengecekan per-tabel untuk memberi *indexing* dan memahami suatu alur relasi antar tabel *e-learning* yang nanti akan dites melalui jupyter notebook.



Gambar 4.7 Query index

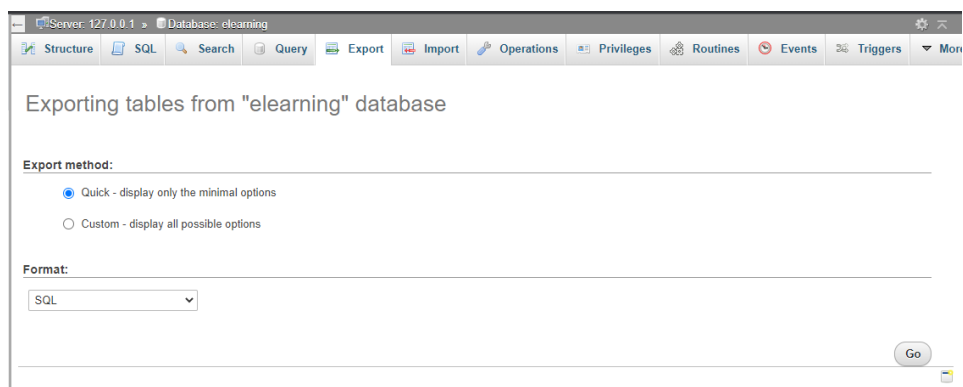
- c. Tahap ketiga masuk ke *database e-learning\_kw* yang mewakilkan *database* tanpa *index* dari tabel 419 menjadi 64 tabel yang *dimana* nantinya dilakukan cek ulang satu persatu lagi untuk adanya *index* dan

relasi antar tabelnya. Dari relasi itu kita dapat mengetahui adanya *indexing* antar tabel dari 64 tabel utama dari 419 tabel keseluruhan. Setelah tahapan ini dapat dibedakan dengan *database e-learning* yang mewakilkan *index* yang 64 tabel utamanya saling berelasi dan ada *index*. Sedangkan *e-learning\_kw* yang mewakilkan tanpa *index* untuk *index* dan relasinya dihapus dengan cara *drop index*. Hasil dari tahap itu dapat berguna untuk membandingkan tingkat *respon time* perdetiknya. Agar dapat mengetahui perbedaan antara *database* yang menggunakan *index* dan tanpa menggunakan *index*.

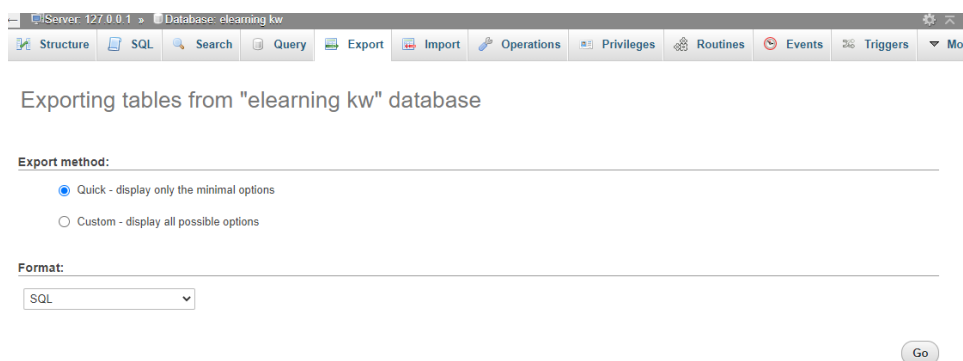


Gambar 4.8 *Drop index*

- d. Tahap keempat, kedua *database* di seleksi lagi dari 419 menjadi 64 tabel utama yang akan dilakukan testing di *python jupyter notebook*. Dari 64 tabel utama tersebut dilakukan *export* menjadi format *.txt* untuk dilakukan uji testing perbandingan antara *database e-learning* yang mewakilkan *database index* dan *e-learning\_kw* yang mewakilkan *database* tanpa *index* di *python jupyter notebook*. Masuk ke *database e-learning\_kw* yang mewakilkan *database* tanpa *index*.



Gambar 4.9 Export 64 tabel e -learning dengan index



Gambar 4.10 export 64 tabel e -learning\_kw dengan tanpa index

#### 4.3.4 Testing

Sebuah *sistem* yang telah dibuat akan dilakukan pengujian dengan *index* dan tidak menggunakan *index* pada tabel *tabel* yang telah dipilih yakni ada 64 tabel, sehingga hasil dari itu akan mengetahui perbandingan antara query dan tanpa *query*.

Hasil beberapa *query index* dan tanpa *index* dari beberapa tabel.

## 1. Proses pengolahan data dari *database* pada *python*

```
In [43]: 1 import psutil
         2 from datetime import datetime
         3 from time import time
         4 import matplotlib.pyplot as plt

In [44]: 1 import sqlite3 #memasukkan library sqlite3 untuk mysql
         2 import pandas as pd #memasukkan library pandas untuk pengolahan data

Import Data

In [45]: 1 fo = open("elearning_kw.txt", "r", encoding="utf-8")
         2 fol = open("elearning.txt", "r", encoding="utf-8")
         3 data = fo.read()
         4 #print(data)
         5 df = pd.DataFrame([], columns=["waktu", "cpu", "memory", "tersedia", "persen memory", "digunakan", "kosong",
         6                               "timestamp"])
         6
```

Gambar 4.11 *Import* data oleh *python*

Awal mula pengolahan data ini dengan mengimport *database* yang hendak diproses yaitu terdapat dua *file* antara lain *file* “elearning.txt” -yang mewakili *database* berindeks dan *file* “elearning\_kw.txt” -yang mewakili *database* tanpa *indeks*.

```
Untuk input ke database

In [6]: 1 #Fungsi input ke database
         2 def create_table(): #Membuat tabel
         3     c.execute('CREATE TABLE IF NOT EXISTS elearn (IsiString TEXT)')
         4
         5 def data_entry(query): #insert data ke tabel database
         6     c.execute("INSERT INTO elearn (IsiString) VALUES(?)", [query])
         7     conn.commit()
         8
         9
```

Gambar 4.12 Pembuatan fungsi *input* ke *database* *mysql*

Kemudian diberikan fungsi untuk menginput data ke *database* yang terdiri dari fungsi `create_table()` dan `data_entry()`.

### Fungsi Perbandingan

(ditambahkan input ke database di bagian `create_table()` kemudian dilanjut dengan `data_entry(variable)`)

```
In [46]: 1 def perbandingan(query, start, finish, dataframe): #tanpa melakukan looping ,
2         tic = time()
3         #ini untuk mengambil data dari baris start sampai finish
4         up = query[start:finish]
5
6         #variabel up dimasukkan ke database
7         create_table() #Membuat tabelnya
8         data_entry(up) #input datanya
9
10
11         toc = time()
12         now = datetime.now()
13         a = toc-tic
14         b = psutil.cpu_percent()
15         c = psutil.virtual_memory()[0]
16         d = psutil.virtual_memory()[1]
17         e = psutil.virtual_memory()[2]
18         f = psutil.virtual_memory()[3]
19         g = psutil.virtual_memory()[4]
20         h = now
21         df = pd.DataFrame({
22             "waktu": [a],
23             "cpu": [b],
24             "memory": [c],
25             "tersedia": [d],
26             "persen memory": [e],
27             "digunakan": [f],
28             "kosong": [g],
29             "timestamp": [h]
30         })
31
32         app = dataframe.append(df, ignore_index=True)
33         return app
```

Gambar 4.13 Pembuatan fungsi pembandingan

Selanjutnya ditambahkan fungsi pembandingan untuk melakukan perbandingan pada proses *query* yang dilakukan dengan parameter *outputnya* adalah data waktu, cpu, *memory*, tersedia, persen *memory*, digunakan, kosong, dan *timestamp*

### Proses untuk perbandingan disaat input data

```
In [47]: 1 conn = sqlite3.connect('elearn_fix.db')
2         c = conn.cursor()
3
4         ans1 = perbandingan(data, 0, 1000, df)
5         ans2 = perbandingan(data, 1001, 2000, ans1)
6         ans3 = perbandingan(data, 3001, 4000, ans2)
7
8         c.close()
9         conn.close()
```

Gambar 4.14 Proses membandingkan data

Setelah itu di proses untuk data perbandingannya. Disini terdapat tiga iterasi untuk memasukkan data ke *database*. *Database* yang dirujuk yaitu *database* 'elearn\_fix.db' yang kemudian di masukkan ke fungsi perbandingan dengan masing-masing iterasi yaitu untuk iterasi 1 yaitu data 0-1000, iterasi 2 yaitu data 1001-2000, dan iterasi 3 yaitu untuk data 3001-4000.

```
In [48]: 1 ans3
Out[48]:
```

	waktu	cpu	memory	tersedia	persen memory	digunakan	kosong	timestamp
0	0.130136	43.0	8451756032	1170620416	86.1	7281135616	1170620416	2021-12-20 09:36:52.822367
1	0.118210	67.6	8451756032	1169575936	86.2	7282180096	1169575936	2021-12-20 09:36:52.957806
2	0.135620	73.3	8451756032	1165492224	86.2	7286263808	1165492224	2021-12-20 09:36:53.120477

```
In [53]: 1 ans3.to_csv('tanpa_index.csv', index=False)
```

Gambar 4.15 Hasil dari perbandingan data

Selanjutnya didapatkan data kesimpulan untuk ketiga iterasi tersebut yang mana data tersebut berdasarkan parameter yang telah diberikan diawal.

Kemudian *diexport* data tersebut ke bentuk *file* 'csv'.

```
In [120]: 1 conn = sqlite3.connect('elearn_fix.db')
          2 sql1 = """
          3 SELECT
          4     IsiString
          5 FROM
          6     elearn;
          7 """
In [121]: 1 df1 = pd.read_sql(sql1, conn)
          2 df1
Out[121]:
```

IsiString

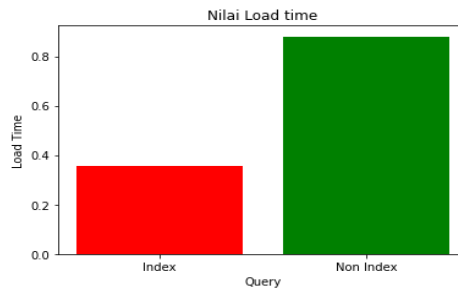
Gambar 4.16 Pengecekan data didalam *database mysql*

Untuk memastikan data sudah masuk pada *database* maka dilakukan pengecekan isi *database* dan didapatkan bahwa data sudah masuk pada *database mysql* yang dirujuk.

Selanjutnya masuk ke proses perbandingan tiap *loop* untuk digrafikkan.

## 2. Loop pertama

*Loop* ini yaitu tabel untuk mengirimkan data dari data 0 ke 1000 menuju *database mysql* dengan nilai *load time* dengan *query* Index 0.3548080921173095 detik dan non-Index 0.8798320293426514 detik

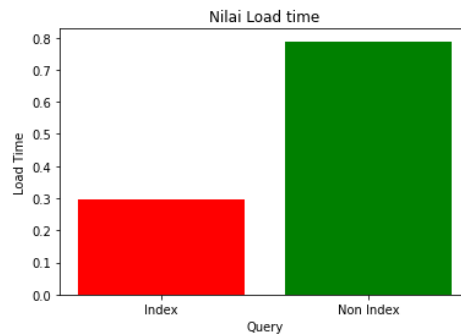


Gambar 4.17 Grafik hasil Loop 1



### 3. Loop kedua

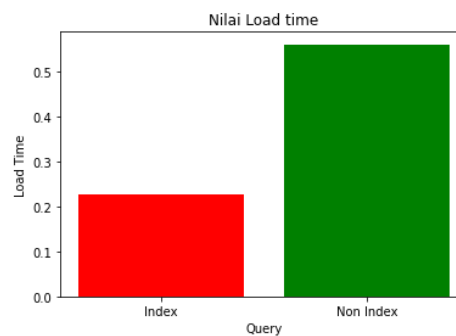
Loop ini yaitu tabel untuk mengirimkan data dari data 1001 ke 2000 menuju database mysql dengan nilai load time dengan query Index 0.2969295978546142 detik dan non-Index 0.7884509563446045 detik



Gambar 4.18 Grafik hasil Loop 2

### 4. Loop ketiga

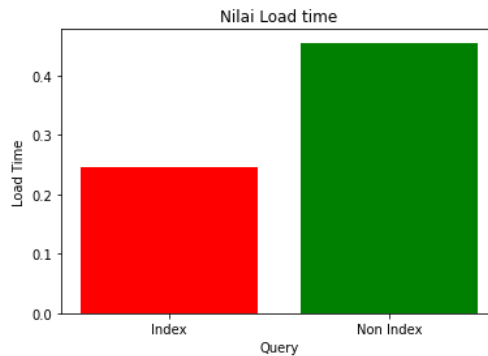
Loop ini yaitu tabel untuk mengirimkan data dari data 3001 ke 4000 menuju database mysql dengan nilai load time dengan query Index 0.2261500358581543 detik dan non-Index 0.5597920417785645 detik



Gambar 4.19 Grafik hasil Loop 3

### 5. Loop keempat

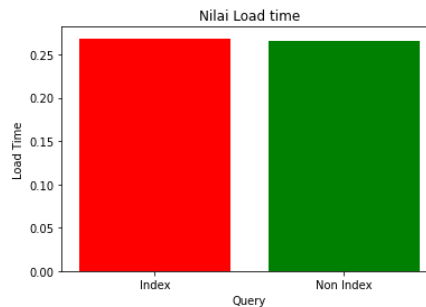
Loop ini yaitu tabel untuk mengirimkan data dari data 4001 ke 5000 menuju database mysql dengan nilai load time dengan query Index 0.2464649677276611 detik dan non-Index 0.4548995494842529 detik.



Gambar 4.20 Grafik hasil Loop 4

#### 6. Loop kelima

Loop ini yaitu tabel untuk mengirimkan data dari data 5001 ke 6000 menuju *database mysql* dengan nilai *load time* dengan *query* Index 0.2685210704803467 detik dan non-Index 0.2649304866790771 detik.



Gambar 4.21 Grafik hasil Loop 5

#### 4.3.5 Maintenance

Tahap pemeliharaan software yang telah dibuat dengan melakukan pengecekan terhadap sistem untuk mencari tahu apabila terdapat fungsi-fungsi yang bermasalah atau *error*, sehingga dapat dilakukan perbaikan.

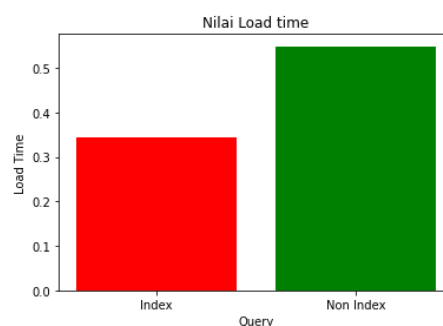
#### 4.4. Hasil pengujian Testing dengan Phyton

Hasil pengujian testing dengan pyhton ini membandingkan antara penggunaan query index dan tanpa query index. Data yang telah yang telah di *upload* ke *mysql* menggunakan *bigdump*. Proses selanjutnya yakni menggunakan *moodle* yang nantinya berguna untuk untuk *update* data yang belum terupload

beberapa ke mysql. Dari proses tersebut akan menghasilkan *output* suatu sistem *e - learning* JTI secara offline, namun databasenya juga dapat dilihat secara *online* dengan *platform* model itu. Setelah melakukan proses panjang itu langsung proses seleksi memilih tabel – tabel mana yang penting digunakan dalam sistem *e – learning*.

Proses *query index* ini sendiri melakukan beberapa tahap yakni pertama memberikan *query index* pada *foreign key* pada 64 tabel *database e – learning* JTI Politeknik Negeri Jember yang berguna untuk menghubungkan ke *primarykey* tabel relasi tersebut. Selanjutnya memberikan relasi antar tabel tersebut yang berguna untuk mengetahui perbandingan suatu *respon time* pada *database*. Data proses dari *query index* dan tanpa *index* tersebut di masukkan *kedalam excel* guna mengetahui tingkat *respon time load database e – learnig*.

Phyton adalah Bahasa pemrograman interpretatif multiguna. Tidak seperti bahasa yang lain yang susah dibaca dan dipahami, Proses selanjutnya yakni membuat sebuah program phyton yang berguna untuk menampilkan suatu hasil data dari *excel* yang *mengambarkan* sebuah grafik lebih cepat mana *respon timenya* dengan *query index* ataupun tidak. dengan nilai *load time* dengan *query Index* 0.3439012542366981 detik dan non-Index 0.5485406294465065 detik.



Gambar 4.22 Grafik rata-rata hasil keseluruhan

```
In [208]: 1 import matplotlib.pyplot as plt
2
3 # x-coordinates of left sides of bars
4 left = [1, 2]
5
6 # heights of bars
7 height = rata_rata
8 # print
9 # labels for bars
10 tick_label = langs
11
12 # plotting a bar chart
13 plt.bar(left, height, tick_label = tick_label,
14         width = 0.5, color = ['red', 'green'])
15
16 # naming the x-axis
17 plt.xlabel('Query')
18 # naming the y-axis
19 plt.ylabel('Load Time')
20 # plot title
21 plt.title('Rata-rata load time')
22
23 # function to show the plot
24 plt.show()
```

Gambar 4.23 *coding python* untuk menampilkan grafik perbandingan

## BAB 5. KESIMPULAN DAN SARAN

### 5.1. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang dilakukan tentang optimalisasi *query index* terhadap *database e – learning* JTI Politeknik Negeri Jember untuk mencegah *respon time* lama jika diakses oleh banyak *user* dapat diambil kesimpulan bahwa :

1. Optimalisasi *database* JTI Politeknik Negeri Jember yang cepat dan lama dapat dibedakan dengan menggunakan *index* dan tidak menggunakan *index* dengan menggunakan *index respon time* lebih cepat dengan rata rata yang cukup berbeda dibandingkan tanpa *index*.
2. Data yang utama dalam *query* ini terdapat 64 tabel yang saling berelasi. *Respon time* yang cepat berkisar 0.3439012542366981 detik. Sedangkan *respon time* yang lama berkisar 0.5485406294465065 detik.
3. Penggunaan *query index* dapat digunakan untuk *merubah respon time* pada *database* JTI yang awalnya *respon timenya* cukup lama dengan menggunakan *query index* maka *respon load database* pada sistem *e – learning* JTI membuat lebih optimal. *Query index* yang digunakan setiap tabelnya terdiri 1 – 4 setiap itu masing–masingnya terdiri 1 hingga 2 tabel yang berelasi. Dari pengujian yang telah dilakukan dengan data tersebut menghasilkan tingkat *respon time* lebih cepat dan berbeda dengan data yang tidak menggunakan *index*.

### 5.2. Saran

Berikut ini beberapa saran yang diberikan dan diharapkan dalam memberikan manfaat untuk pengembangan selanjutnya yaitu :

1. Perlu dilakukan penelitian lanjut dengan ruang lingkup berbeda dan pengambilan data di banyak jurusan dengan contoh topik *e – learning*

Politeknik Negeri Jember sehingga data *training*-nya lebih banyak dan tingkat *load time* akan lebih cepat.

2. Perlu pengembangan optimasi *query index* yang lebih baik. Sehingga data yang diproses akan lebih optimal dengan *respon time* yang maksimal.
3. Perlu penambahan data yang digunakan dalam penelitian sehingga hasil *load database* menggunakan *query index* lebih akurat. Semakin banyak data yang digunakan akan semakin baik untuk penelitian selanjutnya.

## DAFTAR PUSTAKA

- Bismo Raharjo, Y. A., Mantriwira, D., & Sumanto, F. S. (2018). Analisis Kinerja Optimasi Query Cross Join, Natural Join Dan Full Outer Join. *CSRID (Computer Science Research and Its Development Journal)*, 9(2), 106. <https://doi.org/10.22303/csrid.9.2.2017.106-115>
- Dewi, M. M., 1, & Rezeki2, N. (2017). Analisis Perbandingan Optimasi Query Nasted Join dan Hash Join pada MySQL Server. 31–41.
- Fatkhurrochman1, L2, A. W., & Juwar. (2018). Analisis Perbandingan Cartesian Product , Cross Join , Inner Join dan Outer Join dalam Si Akad. 19(1).
- Pamungkas, R. (2018). Optimalisasi Query Dalam Basis Data My Sql Menggunakan Index. *RESEARCH: Computer, Information System & Technology Management*, 1(1), 27. <https://doi.org/10.25273/research.v1i1.2453>
- Rahmatullah, A., Sriwijaya, U., Surbakti, A., & Sriwijaya, U. (2020). Review Paper : Implementasi Python Dalam Geologi Struktur Review Paper : Implementasi Python Dalam Geologi Struktur. October, 0–5.
- Saputro, H. (2012). Pembelajaran Basis Data (Mysql). *Modul Pembelajaran Praktek Basis Data (MySQL)*, 1–34.
- Solichin, A., & Luhur, U. B. (2015). *MySql 5: Dari Pemula Hingga Mahir. January 2010*.
- Sudarmaji. (2016). Migrasi dan Optimalisasi Database Sistem Informasi berbasis E-Learning Program Diploma III Manajemen Informatika Universitas Muhammadiyah Metro. *MIKROTIK: Jurnal Manajemen Informatika*, 6(2), 13. <https://ojs.ummetro.ac.id/index.php/mikrotik/article/view/440>

