

**PERANCANGAN DAN IMPLEMENTASI RESTful API PADA
SISTEM INFORMASI MONITORING DAN EVALUASI
TUGAS AKHIR MAHASISWA**

Proposal Skripsi



Disusun Oleh :

KUKOH SANTOSO

NPM. 1412170001

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PGRI RONGGOLawe (UNIROW) TUBAN
2021**

KATA PENGANTAR

Segala puji bagi Allah SWT yang telah melimpahkan rahmat, taufiq, dan hidayah-Nya sehingga penulis dapat menyelesaikan proposal skripsi ini.

Proposal skripsi ini berjudul “PERANCANGAN DAN IMPLEMENTASI RESTful API PADA SISTEM INFORMASI MONITORING DAN EVALUASI TUGAS AKHIR MAHASISWA” adalah sebagai salah satu syarat untuk melanjutkan mengerjakan skripsi pada program Strata-1 di Program Studi Teknik Informatika Universitas PGRI Ronggolawe Tuban tahun 2021.

Dalam penyusunan proposal skripsi ini banyak pihak yang telah membantu baik secara langsung maupun tidak langsung, oleh karena itu penulis mengucapkan banyak terimakasih kepada :

1. Kedua orang atas segala do’a, motivasi, dan semangat yang telah diberikan.
2. Bapak Andik Adi Suryanto, S.Kom., M.Kom. selaku ketua program studi Teknik Informatika Universitas PGRI Ronggolawe Tuban..
3. Bapak Andy Haryoko, ST., M.T. dan Asfan Muqtadir, S.Kom., M.Kom atas bimbingan, motivasi, dan saran yang diberikan selaku dosen pembimbing.
4. Segenap Dosen Prodi Teknik Informatika Universitas PGRI Ronggolawe Tuban.
5. Keluarga besar Universitas PGRI Ronggolawe Tuban khususnya mahasiswa Teknik Informatika 2017 yang telah berjuang bersama.
6. Seluruh civitas akademika Prodi Teknik Informatika Universitas PGRI Ronggolawe Tuban atas dukungan, serta kerjasamanya.

Penulis menyadari bahwa proposal ini masih jauh dari kata sempurna, oleh karena itu penulis sangat mengharapkan kritik dan saran yang bersifat membangun, sehingga proposal skripsi ini dapat dijadikan sebagai acuan tindak lanjut penelitian selanjutnya dan bermanfaat dalam perkembangan ilmu pengetahuan pada umumnya, khususnya bagi mahasiswa Teknik Informatika Universitas PGRI Ronggolawe Tuban.

Tuban, 01 Maret 2021

Penyusun

HALAMAN PERSETUJUAN

Proposal skripsi dengan judul :

**PERANCANGAN DAN IMPLEMENTASI RESTful API PADA
SISTEM INFORMASI MONITORING DAN EVALUASI
TUGAS AKHIR MAHASISWA**

Disusun Oleh :

KUKOH SANTOSO

NPM. 1412170001

Telah dilakukan pembimbingan proposal skripsi dan dinyatakan layak untuk mengikuti ujian proposal skripsi pada Program Studi Teknik Informatika Universitas PGRI Ronggolawe (UNIROW) Tuban.

Tuban, 01 Maret 2021

Menyetujui,

Pembimbing I

Pembimbing II

Andy Haryoko, S.T., M.T.

NIDN. 0726047704

Asfan Muqtadir, S.Kom., M.Kom

NIDN. 0724068905

HALAMAN PENGESAHAN

Proposal skripsi dengan judul :

**PERANCANGAN DAN IMPLEMENTASI RESTful API PADA
SISTEM INFORMASI MONITORING DAN EVALUASI
TUGAS AKHIR MAHASISWA**

Disusun Oleh :

KUKOH SANTOSO

NPM. 1412170001

Telah dilakukan ujian proposal skripsi dan dinyatakan layak untuk dilanjutkan dalam penelitian skripsi pada Program Studi Teknik Informatika Universitas PGRI Ronggolawe Tuban.

Tuban, 01 Maret 2021

Mengetahui,

Penguji I

Penguji II

XXXXXX

XXXXXX

NIDN.xxxxxxx

NIDN.xxxxxxx

Pembimbing I

Pembimbing II

Andy Haryoko, S.T., M.T.

NIDN. 0726047704

Asfan Muqtadir, S.Kom., M.Kom

NIDN. 0724068905

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
KATA PENGANTAR.....	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
DAFTAR ISI.....	v
DAFTAR GAMBAR.....	vii
DAFTAR TABEL	viii
BAB I PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	4
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI	
2.1 Tinjauan Pustaka	5
2.2 Dasar Teori.....	10
2.2.1 Sistem Informasi	10
2.2.2 PHP.....	11
2.2.3 MySQL.....	11
2.2.4 Database	12
2.2.5 Database Management System (DBMS).....	12
2.2.6 Unified Modeling Language (UML)	13
2.2.7 Data Flow Diagram (DFD)	16
2.2.8 Web Service	17
2.2.9 Representational State Transfer (REST)	17
2.2.10 Application Programming Interface (API).....	18
2.2.11 RESTful API	18
2.2.12 JavaScript Object Notation (JSON)	19
2.2.13 Framework	19

2.2.14	Laravel.....	20
2.2.15	Dart.....	20
2.2.16	SDK (Software Development Kit)	21
2.2.17	Flutter	21
2.2.18	AVD (Android Virtual Devices)	22
2.2.19	JDK (Java Development Kit)	22
2.2.20	Black Box Testing	22
BAB III METODE PENELITIAN		
3.1	Prosedur Penelitian.....	23
3.1.1	Studi Literatur	24
3.1.2	Observasi	24
3.1.3	Analisis Kebutuhan	24
3.1.4	Desain Sistem	26
3.1.5	Pembuatan Dokumentasi API	27
3.1.6	Pembuatan RESTful API	27
3.1.7	Implementasi	28
3.1.8	Pengujian	28
3.1.9	Penulisan Laporan	28
3.2	Bahan dan Alat Penelitian	28
3.2.1	Bahan.....	28
3.2.2	Alat	29
3.3	Arsitektur Sistem.....	30
BAB IV JADWAL PENELITIAN		
4.1	Jadwal Penelitian.....	31
DAFTAR PUSTAKA		33

DAFTAR GAMBAR

	Halaman
Gambar 3.1 Prosedur Penelitian.....	23
Gambar 3.2 Arsitektur Sistem.....	30

DAFTAR TABEL

	Halaman
Tabel 2.1 Penelitian terkait.....	5
Tabel 2.2 Simbol Use Case Diagram (Henderi, 2009).....	13
Tabel 2.3 Simbol Activity Diagram (Henderi, 2009).....	14
Tabel 2.4 Simbol Class Diagram (Henderi, 2009)	15
Tabel 2.5 Simbol Data Flow Diagram (Weli 2019)	16
Tabel 4.1 Jadwal Penelitian.....	31

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pendidikan tinggi berperan penting dalam menumbuhkan kemampuan, watak, dan budaya bangsa yang bermartabat untuk mencerdaskan kehidupan bangsa untuk kemajuan ilmu pengetahuan dan teknologi melalui penerapan nilai budaya Indonesia. Selain itu termasuk dalam tujuan pendidikan tinggi adalah meningkatkan kapabilitas dalam ilmu pengetahuan dan teknologi kepada alumni untuk meningkatkan kualitas bangsa dalam mengembangkan ilmu pengetahuan dan teknologi dengan tetap menerapkan nilai budaya bangsa melalui sebuah penelitian. Pendidikan tinggi dapat menyelenggarakan beberapa jenis program pendidikan, diantaranya program sarjana, magister, doktor, diploma, magister terapan, doktor terapan, program profesi, dan program spesialis (Pemerintah Republik Indonesia, 2012).

Mahasiswa yang ingin menyelesaikan pendidikannya harus memenuhi beberapa syarat, salah satunya pada jenjang pendidikan S1 mahasiswa harus menyelesaikan Tugas Akhir atau sering disebut dengan Skripsi untuk mendapatkan gelar sarjana. Tetapi masih terdapat beberapa poin yang perlu diperhatikan, diantaranya kurangnya informasi mengenai pengerjaan tugas akhir dan kurangnya monitoring dosen pembimbing dalam proses pengerjaan tugas akhir. Hal ini akan berdampak memperlambat mahasiswa dalam pengerjaan tugas akhir. Untuk membantu mahasiswa dan dosen pembimbing mendapatkan informasi mulai dari proses pengajuan judul tugas akhir, pengajuan dosen pembimbing, pengajuan sidang (seminar) proposal, bimbingan tugas akhir, hingga sidang tugas akhir dibutuhkan sistem yang dapat mengelola dan memberikan informasi tersebut (Fitrianto, 2017).

Sistem informasi monitoring dan evaluasi tugas akhir mahasiswa akhir-akhir ini telah menjadi poin pembahasan yang menarik dalam sebuah penelitian, tetapi terdapat beberapa poin yang perlu diperhatikan, salah satunya sebagian besar dari penelitian tersebut hanya menghasilkan sebuah website sistem informasi sehingga jika terdapat pihak lain yang ingin mengembangkan sistem

tersebut dengan bahasa pemrograman lain atau dengan *platform* yang berbeda harus merancang dan membangun sistem mulai dari awal, sehingga hal tersebut akan menghambat pengembang dalam mengembangkan sistem.

Terdapat sebuah teknologi yang dapat membantu pengembang dalam penyediaan data atau *resource* yaitu RESTful API. Dari uraian permasalahan diatas, penulis bermaksud melaksanakan penelitian dengan judul “PERANCANGAN DAN IMPLEMENTASI RESTful API PADA SISTEM INFORMASI MONITORING DAN EVALUASI TUGAS AKHIR MAHASISWA”, yang mana dalam penelitian ini penulis akan mengembangkan RESTful API dengan menggunakan *framework* Laravel dengan format pertukaran data JSON (*JavaScript Object Notation*) untuk komunikasi antara *front-end* dan *back-end* menggunakan protokol HTTP (*Hypertext Transfer Protocol*), kemudian RESTful API yang telah dibuat akan diimplementasikan kedalam dua jenis sistem yaitu sistem informasi berbasis website dan *mobile* untuk melakukan pengujian *request* dan *response*.

Selain itu dengan melakukan pendokumentasian API (*Application Programming Interface*) dalam sebuah dokumen khusus, sehingga akan membantu pengembang dalam hal ketersediaan data atau *resource* sistem informasi monitoring dan evaluasi tugas akhir mahasiswa khususnya di lingkungan Universitas PGRI Ronggolawe (Unirow) Tuban.

1.2 Rumusan Masalah

Dari penjelasan latar belakang tersebut didapatkan rumusan masalah berikut :

1. Bagaimana merancang RESTful API sistem monitoring dan evaluasi tugas akhir mahasiswa di Universitas PGRI Ronggolawe (Unirow) Tuban ?
2. Bagaimana membangun RESTful API sistem informasi monitoring dan evaluasi tugas akhir mahasiswa di Universitas PGRI Ronggolawe (Unirow) Tuban ?
3. Bagaimana mengimplementasikan RESTful API sistem informasi monitoring dan evaluasi tugas akhir mahasiswa Universitas PGRI Ronggolawe (Unirow) Tuban pada aplikasi berbasis website dan *mobile app* ?

1.3 Batasan Masalah

Dalam penelitian ini perlu adanya batasan masalah yang jelas, sehingga tidak melenceng dari rumusan masalah yang telah ditentukan, batasan tersebut meliputi :

1. Menggunakan format JSON untuk representasi data antara *client* dan *server*.
2. Dalam penelitian ini melibatkan sistem informasi berbasis website dan *mobile* untuk melakukan pengujian *request* dan *response* data dari *RESTful API*.
3. Sistem informasi mobile yang dibangun dalam penelitian ini hanya digunakan untuk melakukan pengujian menampilkan atau *read* data dari server.
4. *RESTful API* yang dibangun hanya untuk data mahasiswa Fakultas Teknik Universitas PGRI Ronggolawe Tuban yang sedang melaksanakan pengerjaan tugas akhir.
5. *RESTful API* dan website yang digunakan untuk pengujian dibangun dengan menggunakan *framework Laravel* dengan *database management system (DBMS) MySQL*.
6. Sistem informasi mobile android yang digunakan untuk pengujian dibangun dengan bahasa pemrograman dart menggunakan *Flutter mobile app SDK (Software Development Kit)*.

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah untuk :

1. Merancang dan membangun *RESTful API* sistem informasi monitoring dan evaluasi tugas akhir mahasiswa di Universitas PGRI Ronggolawe (Unirow) Tuban.
2. Mengimplementasikan konsep teknologi *RESTful API* pada sistem informasi monitoring dan evaluasi tugas akhir mahasiswa berbasis website dan *mobile* di Universitas PGRI Ronggolawe (Unirow) Tuban.

1.5 Manfaat Penelitian

Penelitian ini diharapkan dapat bermanfaat untuk :

1. Penulis
 - a. Merealisasikan teori-teori yang diperoleh selama dibangku perkuliahan tentang RESTful API dan *framework* Laravel kedalam pembuatan sistem.
 - b. Mengimplementasikan pemrograman mobile dengan RESTful API menggunakan Flutter SDK dalam pembuatan sistem.
2. Pengguna / Pengembang
 - a. Menyediakan data dan informasi dalam proses pengerjaan tugas akhir.
 - b. Menyediakan akses data atau *resource* bagi pengembang lain yang ingin mengembangkan sistem yang *multiplatform*.

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Dalam penelitian ini penulis mengacu pada beberapa jurnal yang berhubungan dengan perancangan dan implementasi RESTful API, adapun jurnal yang digunakan sebagai referensi utama dalam penelitian ini adalah :

Tabel 2.1 Penelitian terkait

No.	Penulis	Judul	Metode	Masalah	Hasil
1.	(Fitrianto, 2017)	Sistem Informasi Monitoring Dan Evaluasi Tugas Akhir Mahasiswa Studi kasus Universitas PGRI Ronggolawe (UNIROW) Tuban	Sistem informasi berbasis website	Masih kurangnya mahasiswa dalam mendapatkan informasi tentang syarat yang harus dipenuhi untuk mengerjakan skripsi, kurangnya informasi dosen pembimbing mengenai nama mahasiswa yang menjadi bimbingannya, kurangnya monitoring dalam pengerjaan (progres) mahasiswa hal ini akan berdampak memperlambat penyelesaian skripsi.	Sistem informasi berbasis website yang dapat mengkoordinasikan pengajuan skripsi, mempermudah mahasiswa dalam melakukan pengajuan skripsi, dan menyimpan data skripsi secara online.

2.	(Bagus et al., 2020)	Sistem Informasi Tugas Akhir Program Studi Teknik Informatika Universitas Mataram	Pengembangan sistem tugas akhir berbasis website menggunakan metode waterfall dan diagram UML (<i>Unified Modeling Language</i>)	Sistem manual akan menghambat proses bimbingan dan konsultasi antara mahasiswa dan dosen pembimbing ketika dosen pembimbing sedang diluar kota atau memiliki kesibukan lain, selain itu hal ini juga akan menyulitkan dosen pembimbing untuk mengontrol mahasiswa dalam proses pengerjaan skripsi.	Sistem Informasi Tugas Akhir berbasis website dengan menggunakan <i>framework</i> Laravel yang dapat mempercepat dalam penanganan administrasi tugas akhir.
3.	(Heryatno, 2020)	Pengembangan Sistem Informasi UIIPerkuliahan Dengan RESTful API	Teknologi <i>web service</i> untuk fleksibilitas pengembangan aplikasi dan ketangkasan aplikasi dalam mengatasi <i>request</i> dalam jumlah besar.	Aplikasi W-Simak saat ini belum menggunakan <i>web service</i> , menyebabkan setiap perubahan yang dilakukan mengharuskan <i>deploy</i> ulang keseluruhan aplikasi yaitu bagian <i>back end</i> dan <i>front end</i> yang berimplikasi pada waktu <i>deploy</i> aplikasi tersebut. Belum adanya <i>web service</i> juga menyebabkan aplikasi	RESTful API untuk mengelola nilai mahasiswa, presensi mahasiswa, dan pencetakan berkas perkuliahan, yang kemudian RESTful API tersebut dikonsumsi oleh aplikasi berbasis web.

				diluar W-Simak harus membuat ulang layanan yang sebenarnya sudah ada. Selain itu ketika aplikasi W-Simak diakses oleh banyak pengguna sekaligus, hal ini akan menyebabkan turunnya performa dan kemampuan aplikasi.	
4.	(Somya & Nathanael, 2019)	Pengembangan Sistem Informasi Pelatihan Berbasis Web Menggunakan Teknologi Web Service Dan Framework Laravel	Penerapan <i>web service</i> dan <i>framework</i> Laravel dalam proses integrasi data.	Untuk mengakses aplikasi Pincher ID pengguna harus menginstal aplikasi melalui <i>Playstore</i> atau <i>App Store</i> , hal tersebut menyebabkan aplikasi Pincher ID tidak dapat terindeks oleh <i>search engine</i> sehingga menghambat pengguna dalam menemukan aplikasi Pincher ID.	Sistem berbasis website yang menerapkan teknologi <i>web service</i> dapat membantu dalam integrasi data secara terpusat, serta dengan menggunakan <i>framework</i> laravel menghasilkan sebuah sistem yang ringan dan performa yang cepat.

5.	(Arsana & Adnyana, 2020)	Implementasi Web Service Pada Integrasi Data Kerja Praktik, Seminar Dan Tugas Akhir	Implementasi teknologi RESTful dengan menggunakan format data <i>JSON</i> untuk pertukaran data.	Dengan dikembangkannya dua sistem informasi yang berbeda pada STMIK STIKOM Indonesia yang dikembangkan dengan database yang berbeda, data master tidak dapat digunakan secara bersama-sama karena berbeda <i>platform</i> dan struktur tabel dalam <i>database</i> .	<i>RESTful web Service</i> yang dibangun dengan bahasa pemrograman PHP menggunakan <i>framework</i> Lumen yang digunakan dalam proses integrasi data.
6.	(Sutrisno et al., 2019)	Perancangan sistem pemasangan iklan online pada aplikasi <i>e-commerce</i> (e-gemanausa) menggunakan metode RESTful API dan <i>framework</i> laravel	RESTful API sebagai metode transfer data dan UML sebagai metode penggambaran sistem	Sistem <i>e-gemanausa</i> yang belum menerapkan metode <i>service oriented</i> menyulitkan dalam proses pengembangan.	Rancangan sistem pemasangan iklan <i>online</i> dengan menggunakan arsitektur <i>microservice</i> .

Dari jurnal pertama yang disusun oleh (Fitrianto , 2017) dengan judul Sistem Informasi Monitoring Dan Evaluasi Tugas Akhir Mahasiswa Studi kasus Universitas PGRI Ronggolawe (UNIROW) Tuban, menghasilkan sebuah website yang dibangun dengan *Yii framework* dan *MySQL DBMS* sehingga dapat memberikan panduan dan menyediakan informasi data mahasiswa dalam proses mengerjakan tugas akhir, serta membantu dosen dalam memonitoring progress pengerjaan tugas akhir mahasiswa.

Kemudian dari jurnal kedua yang ditulis oleh (Bagus et al., 2020) dengan judul Sistem Informasi Tugas Akhir Program Studi Teknik Informatika Universitas Mataram menghasilkan sebuah sistem berbasis website yang dapat mempercepat dalam proses pelayanan administrasi dalam mengerjakan tugas akhir.

Sedangkan pada jurnal ketiga dengan judul Pengembangan Sistem Informasi UIIPerkuliahan Dengan RESTful API yang disusun oleh (Heryatno, 2020), didapatkan sebuah RESTful API yang dapat digunakan sebagai sumber data atau *resource* saat terdapat aplikasi diluar Sistem Informasi UIIPerkuliahan ingin terhubung pada layanan yang sudah ada pada Sistem Informasi UIIPerkuliahan.

Selain itu (Somya & Nathanael, 2019) dalam jurnalnya yang berjudul Pengembangan Sistem Informasi Pelatihan Berbasis Web Menggunakan Teknologi Web Service Dan Framework Laravel menyatakan dengan penerapan *web service* pada Pincher ID dapat mempermudah proses integrasi data dapat dilakukan secara terpusat.

Kemudian dari jurnal kelima yang berjudul Implementasi Web Service Pada Integrasi Data Kerja Praktik, Seminar Dan Tugas Akhir yang ditulis oleh (Arsana & Adnyana, 2020), dengan mengimplementasi teknologi RESTful menggunakan format data *JSON* dapat dipergunakan untuk mengintegrasikan data menjadi lebih lebih ringan, mudah dibaca dan ditulis.

Dan dari jurnal terakhir dengan judul Perancangan sistem pemasangan iklan online pada aplikasi *e-commerce* (e-gemanusa) menggunakan metode RESTful API dan *framework* Laravel yang ditulis oleh (Sutrisno et al., 2019)

didapatkan hasil dengan penggunaan metode RESTful API akan mempermudah proses pengembangan sistem dan integrasi sistem dengan berbagai *platform*.

Dari beberapa jurnal tersebut dapat diambil kesimpulan bahwa penggunaan metode *RESTful API* kedepan akan pesat perkembangannya, terutama untuk komunikasi data antara *back-end* dan *front-end*. Sehingga dalam penelitian ini penulis mengajukan judul Perancangan Dan Implementasi RESTful API Pada Sistem Informasi Monitoring Dan Evaluasi Tugas Akhir Mahasiswa. Pada penelitian ini penulis mengembangkan RESTful API dengan format pertukaran data *JSON (JavaScript Object Nation)* untuk mengembalikan data dari *server* ke *client* melalui protokol *HTTP (Hypertext Transfer Protocol)*. Kemudian untuk pengujian, *RESTful API* yang telah dibuat akan diimplementasikan kedalam dua jenis sistem yang berbeda yaitu sistem berbasis *mobile* yang diangun dengan menggunakan *Flutter mobile app SDK (Software Development Kit)* dan sistem berbasis website yang dikembangkan dengan *framework* Laravel. Selain itu dengan membuat dokumentasi API dapat digunakan sebagai panduan pengembang yang akan menggunakan data pada *RESTful API* tersebut.

2.2 Dasar Teori

2.2.1 Sistem Informasi

(Sutabri, 2012) menjelaskan sistem informasi sebagai cara untuk mengelola kebutuhan transaksi yang mendukung fungsi manajerial sebuah organisasi untuk mendukung penyediaan data dan informasi kepada pihak lain melalui sebuah sistem. Komponen dari sistem informasi atau yang sering disebut dengan *building block* atau blok bangunan, yaitu :

- a. *Input block* : Merupakan representasi data yang diinput ke sistem.
- b. *Model block* : Merupakan gabungan metode matematik, logika, dan prosedur dalam proses manipulasi data input yang akan disimpan dalam *database* sehingga menghasilkan sebuah informasi atau data.
- c. *Output block* : Merupakan *output* data atau informasi yang telah diolah oleh sistem informasi dalam *model block* atau blok model.

- d. *Technologi block* : Merupakan teknologi yang diterapkan dalam *input block*, *model block*, dan *output block* untuk mengendalikan keseluruhan data dan informasi.
- e. *Database block* : Merupakan data yang tersimpan pada komputer yang mempunyai relasi satu sama lain.
- f. *Control block* : Merupakan rancangan yang dibangun untuk mencegah, mengatasi dan mengendalikan segala kemungkinan yang dapat merusak sistem.

2.2.2 PHP

(Jannah, et al., 2019) menerangkan PHP atau *Hypertext Preprocessor* sebagai bahasa pemrograman *open source* yang mana proses kompilasi dan penerjemahan kode berjalan pada sisi *server* kemudian hasil data dikembalikan ke *client* dalam bentuk *HTML (Hypertext Markup Language)*.

Sedangkan (Supaartagorn et al., 2010) mengartikan PHP sebagai bahasa pemrograman yang berjalan pada sisi *back-end* yang memiliki kelebihan dari segi performa, portabilitas, dan skalabilitas dalam proses manipulasi *database*, yang mana PHP merupakan bahasa pemrograman *open source* yang di desain khusus untuk pengembangan website.

2.2.3 MySQL

Menurut (Amin, 2018) *MySQL* diciptakan oleh *programmer* asal Swedia Michael “Monty” Wedius pada tahun 1979, yang merupakan pengembangan dari konsep utama *database* untuk memasukkan, memilih dan menyeleksi data secara otomatis dan mudah oleh banyak pengguna pada waktu yang bersamaan.

Sedangkan menurut (Turban, 2017) *MySQL* adalah *server* basis data *open source* yang mudah untuk digunakan dan dapat diandalkan karena memiliki kinerja yang cepat serta berfungsi sebagai *RDBMS (relational database manajemen system)* yang berjalan pada arsitektur *client server*.

2.2.4 Database

(Octavian, 2013) menjelaskan bahwa dalam buku yang ditulis oleh Simarmata & Paryudi (2006:1) bahwa :

- a. *Database* adalah sekumpulan data atau informasi yang cocok digunakan oleh sebuah perusahaan (Silberschatz, et al., 2002).
- b. Basisdata adalah sebuah cara yang digunakan untuk menyimpan data atau informasi (Stephens &Plew, 2000).
- c. *Database* adalah kumpulan dari sumber daya milik sebuah organisasi yang berbasis komputer (McLeod, et al., 2001).
- d. Dan (Ramakrishnan & Gehrke, 2003) menjelaskan bahwa *database* adalah sekumpulan data yang menjelaskan aktivitas sebuah organisasi atau lebih yang saling berhubungan.

Sedangkan menurut (Yudhanto & Adi, 2018) basis data adalah sekumpulan data yang dapat diolah dan dimanipulasi melalui perangkat lunak yang ada didalam komputer untuk menghasilkan sebuah data dan informasi yang disimpan secara sistematis.

2.2.5 Database Management System (DBMS)

Menurut (Yanto 2016) DBMS (*Database Management System*) adalah sebuah paket perangkat lunak seperti *MySQL, Microsoft SQL, Oracle, MS. Access* dan lain-lain yang digunakan untuk memasukkan, mengedit, menghapus, dan mengambil informasi dari *database* secara mudah dan efisien, selain itu *DBMS* memiliki beberapa kelebihan diantaranya :

1. Penggunaan memori dan penyimpanan yang lebih efisien.
2. Integritas data akan lebih terjamin.
3. Pembuatan antarmuka kedalam data akan lebih mudah.
4. Mempermudah pengelolaan basis data.
5. Flesibilitas dalam sistem keamanan.

2.2.6 Unified Modeling Language (UML)

Menurut (Mubarak, 2019) UML atau *Unified Modeling Language* merupakan standar dalam pembuatan *blue print* sebuah sistem berorientasi objek yang meliputi alur bisnis, kelas, skema basis data, dan komponen lain yang diperlukan dalam merancang sebuah *software* melalui gambar atau grafik.


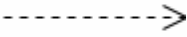
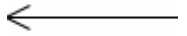
Menurut (Dharwiyanti & Wahono, 2003) UML menyediakan standar dalam perancangan sebuah model sistem melalui visual dan dokumentasi *software* yang dapat berjalan pada perangkat keras, sistem operasi, dan bahasa pemrograman yang beragam.







Menurut (Henderi, 2009) berikut beberapa jenis diagram yang dapat digunakan dalam perancangan sistem menggunakan *Unified Modeling Language* :

a. Use Case Diagram (UCD)

Merepresentasikan interaksi “apa” yang dapat dilakukan oleh pengguna atau sistem lain kepada sistem yang dirancang, hal ini sangat membantu dalam proses penyusunan *requerment* aplikasi, merancang skenario pengujian, dan menyiapkan hasil rancangan aplikasi kepada *client*.

Tabel 2.2 Simbol Use Case Diagram (Henderi, 2009)


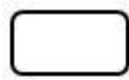

No	Simbol	Nama	Keterangan
1		<i>Actor</i>	Merupakan <i>user</i> atau sistem lain yang mempunyai hubungan dengan sistem yang dirancang untuk melakukan pekerjaan tertentu.
2		<i>Dependency</i>	Menunjukkan ketergantungan antara suatu elemen dengan elemen lain yang disebut dengan elemen <i>independent</i> atau tidak mandiri.
3		<i>Generalization</i>	Menunjukkan pewarisan spesifikasi sebuah elemen dari elemen lain.




4		<i>Include</i>	Menunjukkan penggunaan suatu fungsi yang telah ada pada fungsi <i>use case</i> lain.
5		<i>Extend</i>	Menunjukkan perluasan fungsionalitas yang telah ada pada <i>use case</i> lain.
6		<i>Association</i>	Menunjukkan hubungan elemen antara <i>use case</i> dengan aktor.
7		<i>System</i>	Menunjukkan cakupan paket yang terdapat pada sistem.
8		<i>Use Case</i>	Menggambarkan bagaimana aktor menggunakan sistem.
9		<i>Collaboration</i>	Menunjukkan elemen lain atau aturan yang mempunyai perilaku lebih luas dari jumlah elemennya.

b. Activity Diagram

Activity diagram merupakan model aliran atau kontrol aktivitas ke aktivitas yang lain pada sebuah sistem yang digambarkan secara global, dinamis dan alamiah.

Tabel 2.3 Simbol Activity Diagram (Henderi, 2009)

No.	Simbol	Nama	Keterangan
1		<i>Initial Node</i>	Menunjukkan awal dari sebuah aktivitas.
2		<i>Activity</i>	Menunjukkan aktivitas dari setiap kelas yang kemudian dapat diuraikan menjadi aktivitas yang lebih rinci.
3		<i>Decision</i>	Menunjukkan percabangan untuk pengambilan keputusan.

4		<i>Activity Final Node</i>	Menunjukkan akhir dari sebuah aktivitas yang sedang berjalan.
5		<i>Fork</i>	Menggambarkan suatu aliran yang dapat berubah menjadi beberapa bagian aliran baru.
6		<i>Join</i>	Penggabungan dari beberapa aliran menjadi satu aliran.

c. Class Diagram

Merupakan gambaran strukrur statis yang menunjukkan deskripsi serta logika dari suatu *class* didalam sebuah sistem.

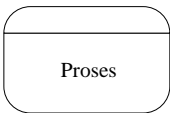
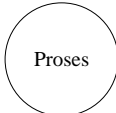
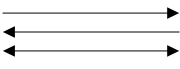
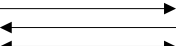
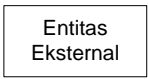
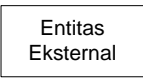
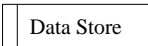
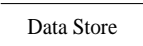
Tabel 2.4 Simbol Class Diagram (Henderi, 2009)

No	Simbol	Nama	Keterangan
1		<i>Generalization</i>	Pembagian struktur data dan perilaku dari objek induk (<i>ancestor</i>) kepada objek anak (<i>descendent</i>).
2		<i>N-Ary Association</i>	Cara untuk menghindar dari asosiasi memiliki objek lebih dari 2.
3		<i>Class</i>	Kumpulan objek yang memiliki operasi dan atribut yang sama.
4		<i>Realization</i>	Aktivitas nyata yang dilakukan oleh sebuah objek.
5		<i>Dependency</i>	Menunjukkan hubungan yang mempengaruhi elemen lain yang bergantung pada elemen <i>independent</i> saat terjadi perubahan.
6		<i>Association</i>	Menunjukkan hubungan yang terjadi antar objek.

2.2.7 Data Flow Diagram (DFD)

Menurut (Weli 2019) DFD atau *Data Flow Diagram* merupakan gambaran aliran data secara logis dalam suatu sistem. DFD juga digunakan untuk menggambarkan hasil analisis dan rancangan terhadap sistem yang baru. Terdapat dua jenis simbol yang digunakan dalam membuat *Data Flow Diagram* yaitu DeMacro & Yourdon (1979) dan Gane & Sarson (1979).

Tabel 2.5 Simbol Data Flow Diagram (Weli 2019)

No	Simbol		Nama	Keterangan
	Gane/Sarson	Yourdon/DeMacro		
1			<i>Process</i>	Menunjukkan bagian dari suatu sistem yang merubah <i>input</i> menjadi <i>output</i> , yang menjelaskan kegiatan yang sedang / akan berjalan.
2			<i>Data Flow</i>	Menggambarkan data yang mengalir dalam sistem yang sedang berjalan.
3			<i>Terminator</i>	Simbol yang menunjukkan entitas eksternal, yang memiliki kepentingan terhadap sistem .
4			<i>Data Store</i>	Berbagai media untuk melakukan penyimpanan data.

2.2.8 Web Service

(Sutanta & Mustofa, 2012) menjelaskan dari beberapa sumber bahwa *Web service* adalah sebuah cara sistem untuk berinteraksi dan mengakses data dari sistem lain menggunakan teknologi yang berbeda, sebagai sebuah *Remote Procedure Call* yang dapat mengeksekusi fungsi yang telah didefinisikan oleh aplikasi web melalui sebuah API (*Application Programming Interface*), *web service* memiliki beberapa kelebihan, diantaranya :

1. Memungkinkan melakukan pertukaran data melalui sistem operasi dan perangkat yang berbeda.
2. Dapat dikembangkan dan diakses menggunakan berbagai bahasa pemrograman.
3. Mengesampingkan jenis DBMS yang digunakan saat ingin terhubung ke *database*.
4. Memudahkan proses pertukaran data.
5. Komponen yang sama pada sebuah aplikasi dapat digunakan secara berulang.

2.2.9 Representational State Transfer (REST)

(Pautasso & Wilde, 2010) menerangkan bahwa REST (*Representational State Transfer*) adalah strukrur yang digunakan untuk mengembangkan *web services* dengan berfokus pada sumber daya dari sebuah sistem, termasuk bagaimana *resources* ditulis kedalam bahasa pemrograman yang berbeda menggunakan protokol HTTP secara tegas dengan cara yang konsisten dengan menetapkan pemetaan operasi CRUD (*create, read, update, delete*), sebagai berikut :

1. Untuk melakukan *create resource* atau mengirimkan data ke server, menggunakan POST.
2. Untuk melakukan *read resource* atau mengambil data dari server, menggunakan GET.
3. Untuk melakukan *update resource* atau mengubah data, menggunakan PUT.
4. Dan untuk melakukan penghapusan data atau *delete resource*, menggunakan DELETE.

Sedangkan menurut (Chen et al., 2017) REST adalah teknis komunikasi yang heterogen untuk aplikasi web, adopsi REST dapat menghasilkan arsitektur yang sederhana, dapat diskalakan, aman, efektif, dan handal. Banyak pengembang yang berhasil membuat API (*Application Programming Interface*) yang sederhana dan kuat pada *RESTful Web service*.

2.2.10 Application Programming Interface (API)

(Sutrisno et al., 2019) menerangkan bahwa API (*Application Programming Interface*) adalah penjelasan layanan-layanan yang tersedia didalam sebuah sistem atau aplikasi melalui sebuah dokumentasi pengembangan perangkat lunak yang berfungsi sebagai panduan kepada pengembang dalam mempelajari dan menggunakan fitur yang disediakan.

Sedangkan menurut (Cited & Data, 2017) API (*Application Programming Interface*) adalah uraian *interface* suatu sistem atau aplikasi untuk bertukar data dari sistem satu ke sistem lain.

2.2.11 RESTful API

(Chen et al., 2017) menjelaskan sumber daya atau *resource* merupakan jenis informasi yang dapat diakses dari sebuah aplikasi atau sistem, dapat berupa *object*, *database record*, *algorithm*, atau yang lainnya. Setiap *resource* diidentifikasi oleh sebuah URI (*Universal Resource Identifier*) yang unik dengan menggunakan metode HTTP GET, PUT, POST, DELETE, HEADER, dan OPTIONS, yang akan menghasilkan sebuah data untuk kemudian dikembalikan ke *client*.

Kemudian (Sutrisno et al., 2019) menjelaskan *RESTful API* merupakan generalisasi *interface* yang telah didokumentasikan menggunakan metode API untuk memudahkan pengembang dalam memahami sistem yang sedang berjalan, serta menjadi panduan dalam penggunaan fitur yang tersedia pada sistem melalui internet menggunakan sistem *web service* yang terdistribusi.

2.2.12 JavaScript Object Notation (JSON)

(Dawood, 2017) menerangkan bahwa JSON atau *JavaScript Object Notation* merupakan struktur pertukaran data yang dikembangkan dengan bahasa pemrograman *JavaScript* serta tidak memiliki ketergantungan dengan bahasa pemrograman lain karena mengaplikasikan bahasa yang sering digunakan dalam pemrograman, selain itu *JSON* memungkinkan proses pertukaran data yang cepat, mudah ditulis dan dibaca karena memiliki format yang sederhana. Berikut adalah kelebihan format data *JSON* dibandingkan dengan *XML (eXtensible Markup Language)* :

1. Penulisan format *JSON* lebih mudah dan terstruktur untuk data yang rumit dan kompleks.
2. Untuk data yang sama, ukuran karakter pada format *JSON* relatif lebih sedikit daripada *XML*, sehingga berpengaruh pada kecepatan transfer data.
3. *JSON* menggunakan *function eval()* *JavaScript* untuk menguraikan data, sedangkan *XML* menguraikan data menggunakan *XML HTTP Request*, sehingga penggunaan format *JSON* dirasa lebih sederhana.

2.2.13 Framework

(Yudhanto & Adi, 2018) menjelaskan bahwa dengan menggunakan *framework* atau kerangka kerja, akan memudahkan programmer dalam menuliskan kode program dengan cukup memanggil *library* atau fungsi yang sudah disediakan oleh *framework*, hal tersebut dapat mengakomodasi *developer* atau *programmer* dalam mengatasi persoalan dalam pemrograman, sehingga *programmer* cukup fokus pada proses membangun aplikasi.

Sedangkan menurut (Abdulloh, 2017) megartikan *framework* sebagai kumpulan kode yang sering digunakan dalam pembuatan aplikasi yang tersusun rapi pada sebuah folder sehingga *programmer* tidak perlu menuliskan kode program mulai dari awal, karena banyak hal yang telah disediakan oleh *framework* yang sudah siap untuk digunakan.

2.2.14 Laravel

Menurut (Chen et al., 2017) laravel merupakan kerangka kerja PHP *open-source* yang dibuat oleh Taylor Otwell untuk mengemangkan aplikasi web dengan mengikuti pola arsitektur MVC (*model-view-controller*), dengan menyediakan *authentication*, *routing*, *session manager*, *caching*, *IoC container*, *database migration*, serta *unit testing* yang terintegrasi untuk memberi pengembang kemampuan untuk membangun aplikasi yang kompleks dengan mudah.

Kemudian (Luthfi, 2017) menjelaskan proses modifikasi *database* dapat dilakukan dengan mudah menggunakan fitur migrasi yang telah disediakan oleh laravel, *migration* laravel juga mendukung beberapa basisdata seperti : *PostgreSQL*, *MySQL*, *SQLITE*, dan *MSSQL*, selain itu di *framework* laravel terdapat *Eloquent* yang dapat digunakan untuk mengimplementasikan *Record* aktif menggunakan standar *Object Oriented Programming*.

(Abdulloh, 2017) menerangkan bahwa laravel memiliki beberapa keunggulan dibandingkan dengan *framework* lain, diantaranya :

1. Banyak fitur pada Laravel yang tidak disediakan oleh *framework* lain.
2. Struktur penulisan pada laravel mudah dipahami oleh programmer pemula sekalipun.
3. Dokumentasi yang lengkap pada setiap versi.
4. Banyak *library* yang mendukung, karena laravel banyak digunakan oleh *programmer*.
5. *Library-library* laravel didukung oleh *composer*.
6. Memudahkan programmer dalam menampilkan data karena memiliki *template engine* sendiri yang disebut dengan *blade*.

2.2.15 Dart

Menurut (Tjandra & Chandra, 2020) dart adalah bahasa pemrograman yang memiliki kemiripan dengan bahasa pemrograman lain seperti java dan javascript, karena bahasa pemrograman dart diciptakan oleh google untuk menggantikan bahasa pemrograman javascript yang dirilis pertama kali pada

tahun 2011, dart menerapkan konsep *static typing* dimana programmer harus mendefinikan terlebih dahulu variabel yang akan digunakan.

Sedangkan menurut (Suryono & Hardiansah, 2020) dart adalah bahasa pemrograman *object oriented* yang menggunakan gaya penulisan bahasa C yang terkompilasi secara opsional kedalam bahasa pemrograman Javascript, sehingga memudahkan *programmer* yang pernah menggunakan bahasa pemrograman java.

2.2.16 SDK (Software Development Kit)

Menurut (RIZQI, 2014) *SDK* atau *Software Development Kit* merupakan *API (Application Programming Interface)* yang berfungsi sebagai *emulator* yang digunakan untuk menjalankan aplikasi tanpa harus melakukan *compile* ke dalam format apk dalam mengembangkan aplikasi berbasis android.

Sedangkan menurut (Yuntoto, 2015) *SDK* merupakan *Application Programming Interface (API)* yang digunakan dalam melakukan pengujian aplikasi android.

Kemudian (Mubarak, 2017) menerangkan bahwa *SDK* merupakan alat bantu yang dibutuhkan dalam mengembangkan aplikasi menggunakan bahasa pemrograman java pada *platform* Android.

2.2.17 Flutter

(Tjandra & Chandra, 2020) menerangkan bahwa flutter adalah *Software Development Kit* untuk membangun aplikasi mobile android dan iOS yang memiliki kinerja tinggi hanya dengan satu basis code *open source* yang dibuat oleh google, sehingga programmer dapat menghadirkan aplikasi dengan berkinerja tinggi pada berbagai jenis *platform*.

Sedangkan (Suryono & Hardiansah, 2020) menerangkan bahwa flutter merupakan sebuah *toolkit/framework* yang dibuat dan dikembangkan oleh google untuk membuat aplikasi *multi-platform* baik mobile, web, ataupun desktop dari sebuah basis code, selain itu flutter juga menawarkan keunggulan yaitu *fast development* (proses pengembangan cepat), *expressive and flexible UI* (menawarkan tampilan yang cantik), serta *native performance* dan fitur *hot reload* yang ditawarkan dapat membantu dalam proses pembuatan *user interface*.

2.2.18 AVD (*Android Virtual Devices*)

Menurut (Mubarak, 2017) AVD merupakan *emulator* yang dijadikan sebagai tempat pengujian aplikasi berbasis android yang berjalan pada *virtual Machine*.

Sedangkan (Yuntoto, 2015) menjelaskan bahwa AVD (*Android Virtual Devices*) merupakan bagian dari SDK Android yang berjalan pada *virtual machine* sebagai *emulator* untuk menjalankan dan melakukan pengujian aplikasi berbasis android.

2.2.19 JDK (*Java Development Kit*)

(Mubarak, 2017) menjelaskan bahwa JDK (*Java Development Kit*) merupakan *software* yang untuk melakukan proses kompilasi kedalam *bytecode* sehingga kode program dapat dipahami untuk dijalankan pada JRE (*Java Runtime Envirotment*), komputer yang akan mengembangkan aplikasi berbasis java harus menginstal JDK terlebih dulu sebelum memulai membuat aplikasi, tetapi untuk komputer yang akan menjalankan aplikasi tidak wajib menginstal JDK.

2.2.20 Black Box Testing

Black box testing menurut (Mubarak, 2017) merupakan teknik pengujian aplikasi yang hanya menguji dari segi fungsionalitas aplikasi untuk mengetahui struktur internal yang tidak sesuai dengan cara kerja sistem.

Sedangkan (Mustaqbal et al., 2015) menjelaskan bahwa *Black box testing* adalah cara menguji sistem yang hanya fokus pada spesifikasi fungsionalitas untuk menemukan kesalahan atau ketidaksesuaian fungsi, kesalahan dalam tampilan, kesalahan struktur dan akses database, kesalahan terminasi dan inisialisasi, ketidaksesuaian performa aplikasi, dengan tujuan memecahkan masalah-masalah berikut :

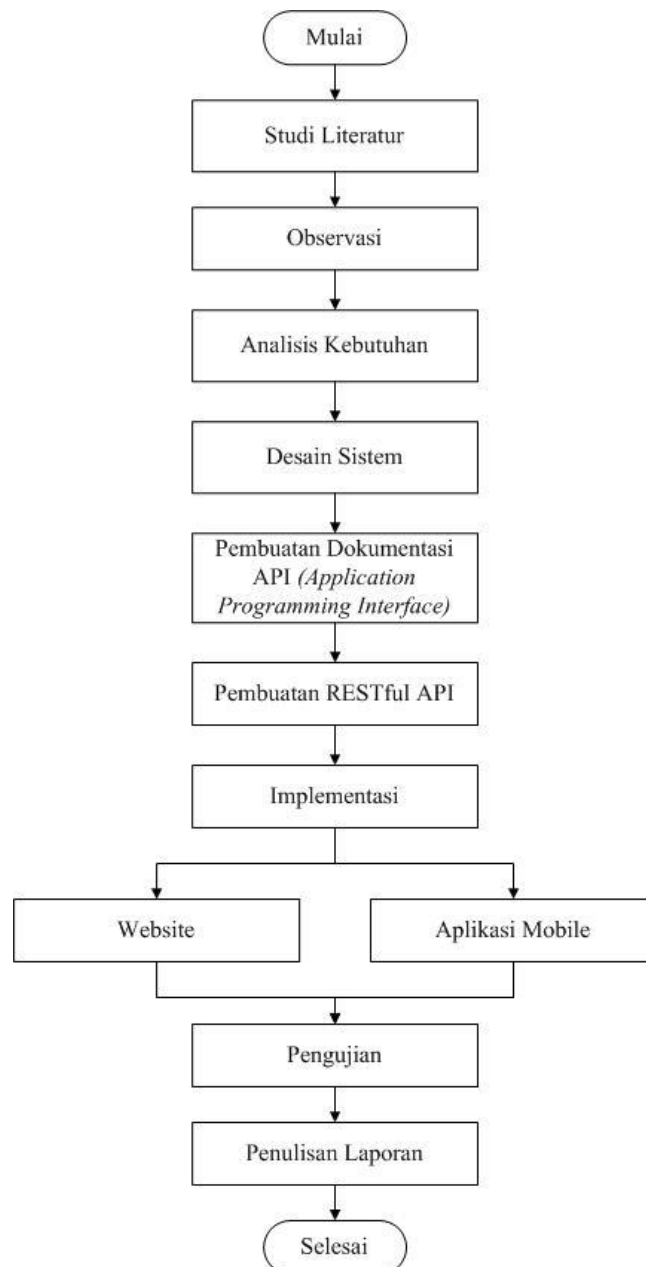
1. Kesesuaian fungsi-fungsi yang diuji.
2. Menentukan input yang baik untuk bahan pengujian aplikasi.
3. Tingkat sensitifitas aplikasi dalam menerima sebuah inputan.
4. Banyaknya data yang mampu ditangani oleh sistem.
5. Cara mengisolasi dan membuat kombinasi data.

BAB III

METODE PENELITIAN

3.1 Prosedur Penelitian

Penelitian ini menggunakan 9 tahapan utama sebagai acuan dalam pelaksanaan penelitian dari tahap awal sampai penelitian selesai seperti pada gambar 3.1 berikut.



Gambar 3.1 Prosedur Penelitian

3.1.1 Studi Literatur

Penelitian ini diawali dari proses pengumpulan literatur-literatur serta kajian yang berkaitan dengan perancangan dan implementasi RESTful API, yang didapatkan dari jurnal, buku, dan sumber lain. Terdapat 6 jurnal utama yang menjadi referensi penulis seperti yang telah dibahas pada tabel 2.1 pada tinjauan pustaka.

3.1.2 Observasi

Observasi dilakukan pada Sistem Informasi Monitoring Dan Evaluasi Tugas Akhir Mahasiswa dari penelitian (Fitrianto, 2017), yang dibangun dengan bahasa pemrograman PHP menggunakan *framework Yii* dan *MySQL DBMS (database management system)*. Yang mana pada sistem informasi tersebut belum diterapkan teknologi *RESTful API* sehingga belum ada ketersediaan akses data atau *resource* untuk pihak lain yang akan menggunakan *resource* Sistem Informasi Monitoring Dan Evaluasi Tugas Akhir Mahasiswa tersebut.

3.1.3 Analisis Kebutuhan

Setelah observasi selesai, tahap berikutnya dilakukan analisis lebih mendalam terhadap penelitian (Fitrianto, 2017) tentang Sistem Informasi Monitoring Dan Evaluasi Tugas Akhir Mahasiswa yang akan dijadikan dasar utama pada penelitian yang akan dilaksanakan oleh penulis, dengan tujuan untuk :

1. Mengetahui alur bisnis (*business flow*) Sistem Informasi Monitoring Dan Evaluasi Tugas Akhir Mahasiswa dari dokumen flow yang telah dirancang, yang mana terdapat 2 dokumen flow, yaitu :
 - a. Dokumen flow seminar proposal yang menjelaskan proses upload KRS, pengajuan judul, upload proposal, proses mendapatkan dosen pembimbing, proses bimbingan proposal, pengajuan seminar proposal, hingga seminar proposal selesai.
 - b. Dokumen flow skripsi yang menjelaskan proses dalam bimbingan dan pengajuan sidang skripsi hingga sidang skripsi selesai.
2. Mengetahui aliran data pada setiap proses kerja aplikasi melalui *Data Flow Diagram (DFD)* yang telah dirancang, dimana terdapat 5 proses utama yang

berjalan pada Sistem Informasi Monitoring Dan Evaluasi Tugas Akhir Mahasiswa, yaitu :

- a. Login
 - b. Pengolahan Data Dosen
 - c. Pengolahan Data Mahasiswa
 - d. Pengolahan Data Master
 - e. Dan Proses Pengolahan Data Skripsi
3. Mengetahui perancangan struktur *database* dari Sistem Informasi Monitoring dan Evaluasi Tugas Akhir Mahasiswa, terdapat 22 tabel pada database yang telah dirancang pada penelitian tersebut yang meliputi :
- a. Tabel user untuk menampung data pengguna.
 - b. Tabel prodi sebagai tempat untuk menyimpan data program studi yang ada di Universitas.
 - c. Tabel fakultas untuk menyimpan data fakultas yang ada di Universitas.
 - d. Tabel dosen untuk menyimpan data pribadi dosen.
 - e. Tabel dosen_login untuk menyimpan data riwayat dosen yang login ke sistem.
 - f. Tabel dosen_kontak untuk menyimpan data kontak dosen.
 - g. Tabel dosen_non_tuban untuk menyimpan data kelurahan dan kecamatan yang berada diluar wilayah kabupaten Tuban.
 - h. Tabel mahasiswa untuk menampung data identitas mahasiswa.
 - i. Tabel mahasiswa_kontak untuk menyimpan data kontak mahasiswa.
 - j. Tabel mahasiswa_login untuk menyimpan data riwayat mahasiswa yang login ke sistem.
 - k. Tabel non_tuban untuk menyimpan data kelurahan dan kecamatan yang berada diluar wilayah kabupaten Tuban.
 - l. Tabel kecamatan untuk menampung data kecamatan yang masuk di wilayah kabupaten Tuban.
 - m. Tabel kelurahan untuk menyimpan data kelurahan yang ada di wilayah kabupaten Tuban.
 - n. Tabel skr_agreement untuk menampung persetujuan yang telah diajukan.
 - o. Tabel skr_consult untuk menyimpan data bimbingan.

- p. Tabel *skr_defernce* untuk menyimpan data pengajuan siding.
 - q. Tabel *skr_level* untuk menyimpan data level yang sudah dilakukan mahasiswa dalam pengerjaan tugas akhir.
 - r. Tabel *skr_session* untuk menyimpan data ID skripsi mahasiswa.
 - s. Tabel *skr_title* untuk menyimpan data judul skripsi.
 - t. Table *skr_title_file* untuk menyimpan data nama file skripsi yang di upload ke sistem.
 - u. Tabel *skr_mentor* untuk menyimpan data dosen yang menyetujui menjadi dosen pembimbing.
 - v. Tabel *skr_final* untuk menyimpan data proposal skripsi yang telah diupload ke sistem.
4. Serta mengetahui *user interface (UI)* yang telah diimplementasikan pada sistem Informasi Monitoring dan Evaluasi Tugas Akhir Mahasiswa pada penelitian sebelumnya.

Selain itu pada tahap ini juga bertujuan untuk menentukan perangkat keras (*hardware*) dan perangkat lunak (*software*) yang dibutuhkan pada proses pelaksanaan penelitian.

3.1.4 Desain Sistem

Dari hasil analisis pada tahap sebelumnya, perlu adanya desain ulang sistem untuk menyesuaikan dengan tujuan dan mempermudah dalam implementasi, yang meliputi :

1. Membuat *blue print* sistem dengan menerapkan konsep UML (*Unified Modeling Language*) dalam bentuk :
 - a. *Use Case Diagram (UCD)* yang akan memberikan gambaran atau interaksi yang dilakukan oleh aktor terhadap sistem.
 - b. Skenario Use Case untuk menjabarkan *use case*, sekaligus sebagai acuan pada tahap proses pengujian sistem.
 - c. *Activity Diagram* untuk menggambarkan urutan aktifitas pada keseluruhan proses bisnis.
 - d. *Class Diagram* untuk menggambarkan hubungan antar objek, serta memberikan gambaran struktur dan deskripsi pada setiap *class*.

2. Dari *blue print* sistem yang telah dibuat, proses selanjutnya adalah merancang *Data Flow Diagram (DFD)* untuk memberikan gambaran aliran data yang terjadi pada Sistem Informasi Monitoring dan Evaluasi Tugas Akhir Mahasiswa.
3. Selajutnya melakukan desain *database* dengan menggunakan MySQL Workbench 8.0 CE, untuk menentukan struktur tabel, *field*, tipe data, serta relasi antar tabel pada *database*.
4. Tahap akhir dari desain sistem adalah membuat desain *user interface* Sistem Informasi Monitoring dan Evaluasi Tugas Akhir Mahasiswa dengan menggunakan aplikasi desain berbasis *cloud* yaitu Figma.

3.1.5 Pembuatan Dokumentasi API

Dari hasil desain sistem yang telah dibuat, tahap selanjutnya adalah proses pembuatan dokumentasi *API (Application Programming Interface)*, hal ini bertujuan untuk menentukan *API endpoint* atau *URL*, *method* yang digunakan, *request* data yang akan dikirimkan ke *server*, serta *response* data *JSON* yang akan dikembalikan ke *client* pada setiap *API endpoint*.

Selain itu dokumentasi ini nantinya akan digunakan sebagai panduan pengembang dalam menggunakan *RESTful API* Sistem Informasi Monitoring Dan Evaluasi Tugas Akhir Mahasiswa, untuk memudahkan dalam merancang dan membuat dokumentasi *API*, penulis menggunakan *Swagger API documentation tools* dengan mengadopsi *OpenAPI Specification* untuk mendeskripsikan *API* yang dibuat.

3.1.6 Pembuatan RESTful API

Setelah dokumentasi *API* selesai, dilanjutkan pada tahap berikutnya yaitu proses pembuatan *RESTful API* dengan menggunakan *framework* Laravel versi 7.0 dengan *DBMS (database management system)* *MySQL* sesuai dengan dokumentasi *API* yang telah dibuat. Pada penelitian ini penulis menggunakan XAMPP Control Panel v3.2.4 dengan PHP versi 7.4.2 sebagai *local web server* sebelum *RESTful API* dilakukan *hosting*, selain itu penulis juga menggunakan Postman v8.0 untuk melakukan pengujian dalam pembuatan *RESTful API* sebelum proses implementasi.

3.1.7 Implementasi

Setelah *RESTful API* selesai, *RESTful API* tersebut dilakukan proses *hosting* untuk kemudian dilanjutkan pada tahap implementasi. Pada tahap ini *RESTful API* Sistem Informasi Monitoring dan Evaluasi Tugas Akhir Mahasiswa yang telah dibuat dilakukan implementasi pada sistem informasi berbasis website yang dibangun menggunakan *framework* Laravel versi 7.0 dan bahasa pemrograman PHP, sedangkan untuk sistem informasi berbasis *mobile* dibangun menggunakan Flutter *SDK (Software Development Kit)* versi 3.12.2 dengan bahasa pemrograman Dart versi 3.11.0.

3.1.8 Pengujian

Selanjutnya yaitu dilakukan proses pengujian pada aplikasi yang telah dibuat untuk memastikan apakah *RESTful API* yang telah dibuat dapat berjalan sesuai dengan Skenario Use Case yang telah dirancang pada tahap desain sistem dengan menggunakan metode *Black box testing*.

3.1.9 Penulisan Laporan

Setelah dilakukan pengujian tahap terakhir adalah membuat laporan penelitian, sehingga dapat memberikan dokumentasi proses penelitian mulai dari awal hingga penelitian selesai, sehingga dapat menyajikan informasi atau penjelasan tentang penelitian yang telah dilaksanakan.

3.2 Bahan dan Alat Penelitian

3.2.1 Bahan

Bahan penelitian berupa data dan informasi dalam bentuk alur bisnis (*business flow*) dan tabel yang berisi atribut dan tipe data dari sistem informasi Monitoring dan Evaluasi Tugas Akhir Mahasiswa di Universitas PGRI Ronggolawe (Unirow) Tuban yang merupakan hasil penelitian yang telah dilaksanakan oleh Fitrianto pada tahun 2017.

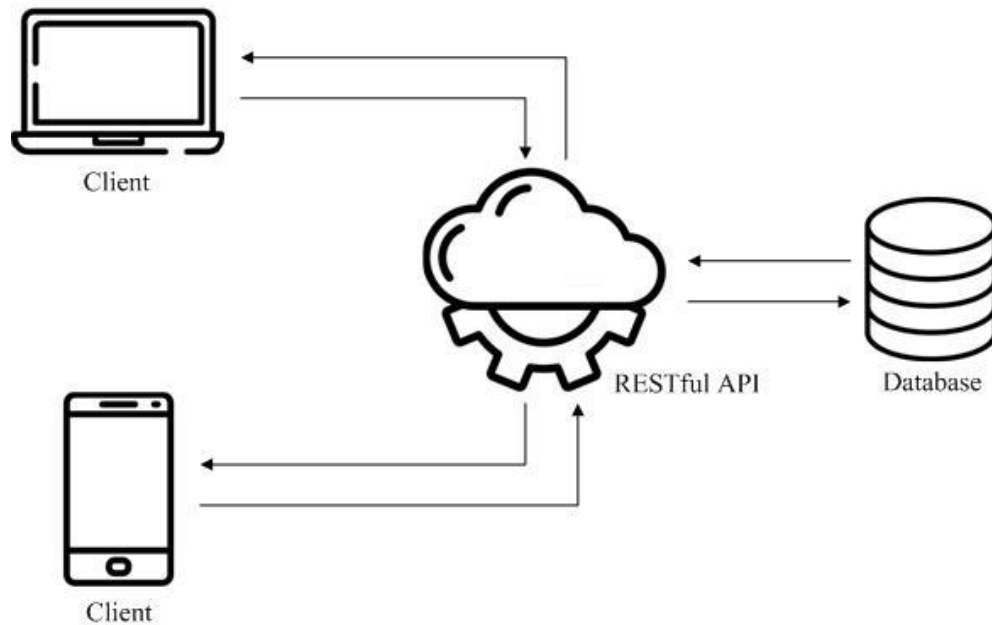
3.2.2 Alat

Peralatan yang digunakan penulis dalam mengerjakan penelitian ini meliputi :

1. Perangkat keras (*hardware*)
 - a. Processor AMD A8-7410 APU.
 - b. RAM DDR3 berkapasitas 12 GB
 - c. Harddisk berkapasitas 500 GB
 - d. Monitor beresolusi 1366x768px
2. Perangkat lunak (*software*)
 - a. Microsoft Office Visio, yang digunakan dalam membuat *blue print* pada proses desain sistem.
 - b. MySQL Workbench 8.0 CE, yang digunakan untuk mendesain *database*.
 - c. Figma, yang digunakan dalam proses desain *user interface* sistem.
 - d. *Swagger API documentation tools*, yang digunakan dalam proses pembuatan dokumentasi *RESTful API*.
 - e. *Text editor* Visual Studio Code.
 - f. Laravel versi 7.0, sebagai *framework* PHP yang digunakan dalam proses pembuatan *RESTful API* dan implementasi *RESTful API* pada sistem informasi berbasis website.
 - g. XAMPP v3.2.4, yang digunakan sebagai *local web server*.
 - h. Postman, yang digunakan dalam proses pengujian pada tahap pembuatan *RESTful API*.
 - i. *Flutter mobile app SDK (Software Development Kit)* v3.12.2 yang digunakan untuk mengimplementasikan *RESTful API* pada sistem informasi berbasis *mobile*.

3.3 Arsitektur Sistem

Arsitektur sistem dijadikan sebagai dasar dalam pengembangan sistem yang akan dibuat, seperti pada gambar 3.2 berikut.



Gambar 3.2 Arsitektur Sistem

Secara global, arsitektur sistem dibagi menjadi tiga bagian yaitu *RESTful API* yang menyediakan fungsi-fungsi yang akan diakses oleh *client* atau *front-end*, kemudian *client* melakukan *request* dengan mengirimkan data kepada *RESTful API* kemudian *response* data dikembalikan dalam format JSON sesuai yang dibutuhkan *client*. Kemudian untuk *database* merupakan tempat penyimpanan data baik dari *client* yang menggunakan sistem informasi berbasis website ataupun *mobile* disimpan dalam *database* yang sama.

BAB IV

JADWAL PENELITIAN

4.1 Jadwal Penelitian

Tabel 4.1 Jadwal Penelitian

No	Kegiatan	Bulan Ke										
		1	2	3	4	5	6	7	8	9	10	11
1	Persiapan											
	a. Identifikasi masalah											
	b. Pengajuan judul											
	c. Studi literatur											
	d. Observasi											
2	Analisis kebutuhan											
3	Desain sistem											
4	Pembuatan dokumentasi API											
5	Pembuatan RESTful API											
6	Implementasi RESTful API											
7	Pengujian											

8	Penulisan laporan											
	a. Bimbingan proposal											
	b. Sidang proposal											
	c. Penyelesaian laporan skripsi											
	d. Ujian skripsi											

DAFTAR PUSTAKA

- Abdullah, R. 2017. *Membuat Aplikasi Point of Sale dengan Laravel dan AJAX*. Jakarta: Elex Media Komputindo
- Amin, R. (2018). *PENGERTIAN DATABASE MySQL Pembahasan*.
- Arsana, N.A., & Adnyana, K.W. (2020). *IMPLEMENTASI WEB SERVICE PADA INTEGRASI DATA KERJA PRAKTIK , SEMINAR DAN TUGAS AKHIR*.
- Bagus, I., Widiartha, K., & Albar, M. A. (2020). *Sistem Informasi Tugas Akhir Program Studi Teknik Informatika (Information system of bachelor thesis at department of informatics engineering*. 2(1), 19–29.
- Chen, X., Ji, Z., Fan, Y., & Zhan, Y. (2017). Restful API Architecture Based on Laravel Framework. *Journal of Physics: Conference Series*, 910(1). <https://doi.org/10.1088/1742-6596/910/1/012016>
- Cited, R., & Data, P. P. (2017). (12) *United States Patent (2) invenor rairie Mahidini Haus desire*. 2(12).
- Dawood, R. (2017). Rancang Bangun Layanan Web (Web Service) Untuk Aplikasi Rekam Medis Praktik Pribadi Dokter. *Jurnal Karya Ilmiah Teknik Elektro*, 2(1), 1–8.
- Dharwiyanti, S., & Wahono, R. S. (2003). Pengantar Unified Modeling LAnguage (UML). *IlmuKomputer.Com*, 1–13. <http://www.unej.ac.id/pdf/yanti-uml.pdf>
- Fitrianto, K.A.(2017). Sistem Informasi Monitoring Dan Evaluasi Tugas Akhir Mahasiswa Studi Kasus Universitas Pgri Ronggolawe (Unirow) Tuban
- Henderi. (2009). Object Oriented Modelling With Unified Modeling Language (Uml). 5 November 2009, June, 77. <https://doi.org/10.13140/RG.2.1.3464.4088>
- Heryatno, M.W. (2020). *Pengembangan sistem informasi uiiperkuliahhan dengan restful api*.
- Jannah, M., et al. 2019. *Mahir Bahasa Pemrograman PHP*. Jakarta: Elex Media Komputindo
- Luthfi, F. (2017). Penggunaan Framework Laravel Dalam Rancang Bangun Modul Back-End Artikel Website Bisnisbisnis.ID. *JISKA (Jurnal Informatika Sunan Kalijaga)*, 2(1), 34.

- Mubarak, A. (2019). Rancang Bangun Aplikasi Web Sekolah Menggunakan Uml (Unified Modeling Language) Dan Bahasa Pemrograman Php (Php Hypertext Preprocessor) Berorientasi Objek. *JIKO (Jurnal Informatika Dan Komputer)*, 2(1), 19–25. <https://doi.org/10.33387/jiko.v2i1.1052>
- Mubarak, M.F. (2017). Aplikasi Pelaporan Pelayanan Publik Berbasis Android (Studi Kasus Ombudsman Makassar)
- Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015). *PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN)*. 1(3), 31–36.
- Octavian, D. T. (2013). Desain database sistem informasi penjualan barang. *Jurnal Teknologi Dan Informatika (Teknomatika)*, 1(2), 148–157.
- Pautasso, C., & Wilde, E. (2010). *RESTful web services*. November, 1359. <https://doi.org/10.1145/1772690.1772929>
- Pemerintah Republik Indonesia. (2012). UU No 12 Tahun 2012 tentang Pendidikan Tinggi. *Republik Indonesia*, May 2014, 32.
- RIZQI, R. F. (2014). *Pembuatan Aplikasi Lembar Kerja Siswa Elektronik (ELks) Berbasis Android*. <http://eprints.ums.ac.id/id/eprint/30664>
- Somya, R., & Nathanael, T. M. E. (2019). Pengembangan Sistem Informasi Pelatihan Berbasis Web Menggunakan Teknologi Web Service Dan Framework Laravel. *Jurnal Techno Nusa Mandiri*, 16(1), 51–58. <https://doi.org/10.33480/techno.v16i1.164>
- Sutrabri, T. 2012. *Konsep Sistem Informasi*. Yogyakarta: Andi Offset
- Studi, P., Dan, B., Indonesia, S., Bahasa, F., Seni, D. A. N., & Yogyakarta, U. N. (2015). Diajukan Kepada Fakultas Bahasa dan Seni Universitas Negeri Yogyakarta untuk Memenuhi Sebagian Persyaratan guna Memperoleh Gelar Sarjana Pendidikan. *Desy Fatmawati*, september.
- Supaartagorn, C., Goujon, M., McWilliam, H., Li, W., Valentin, F., Squizzato, S., Paern, J., & Lopez, R. (2010). A new bioinformatics analysis tools framework at EMBL--EBI. *Nucleic Acids Research*, 38(suppl 2), W695--W699.
- Suryono, S., Hardiansah. 2020. *Panduan Lengkap membuat aplikasi Android dan iOS dari Nol*. Surabaya: Surelabs Pub

- Sutanta, E., & Mustofa, K. (2012). Identifying The Needs of Web Service to Data Synchronization Between Information Systems as E-Government Ecosystem at Bantul-Yogyakarta. *Teknik Informatika - STMIK Bandung*, 2(3), 20–26. <https://repository.ugm.ac.id/id/eprint/33043>
- Sutrisno, S., Asyidiq, M., Santoso, S., Akhir, T., Aminuddin, A., Поконова Ю.В., Fotokita, T., Menggunakan, C. O. M., Hakim, A. A., Pratama, S., S, F. P., Novita, R., Subandri, M. A., Simaremare, Y. P. ., S, A. P., & Wibowo, R. P. (2019). Perancangan Sistem Pemasangan Iklan Online Pada Aplikasi E-Commerce (E-Gemanusa) Menggunakan Metode Restful Api Dan Framework Laravel. *Jurnal Sains, Teknologi Dan Industri*, 2(2), 32–38. <http://ejurnal.its.ac.id/index.php/teknik/article/view/5163/1552%0Ahttp://ejournal.lppm-unbaja.ac.id/index.php/saintek/article/view/99>
- Tjandra, S., & Chandra, G. S. (2020). Pemanfaatan Flutter dan Electron Framework pada Aplikasi Inventori dan Pengaturan Pengiriman Barang. *Journal of Information System, Graphics, Hospitality and Technology*, 2(02), 76–81. <https://doi.org/10.37823/insight.v2i02.109>
- Turban, E. et al. (2017). Electronic Commerce 2012: Managerial and Social Networks Perspectives, 7/e. New Jersey: Pearson Education. *Jurnal Informatika*, 8(1), 826–836.
- Weli. 2019. *Aplikasi Kasus Siklus Transaksi Bisnis: Suatu Pendekatan Pengembangan Sistem Informasi Akuntansi*. Jakarta: Universitas Katolik Indonesia Atma Jaya
- Yanto, R. 2016. *Manajemen Basis Data Menggunakan MySQL*. Yogyakarta: Deepublish
- Yudhanto, Y., Prasetyo, H.A. 2018. *Panduan Mudah Belajar Framework Laravel*. Jakarta: Elex Media Komputindo