

**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 hanya untuk Allah SWT yang karena rahmat, taufiq, dan hidayah-Nya proposal skripsi ini dapat terselesaikan.

Proposal skripsi ini berjudul “PERANCANGAN DAN IMPLEMENTASI RESTful API PADA SISTEM INFORMASI MONITORING DAN EVALUASI TUGAS AKHIR MAHASISWA” adalah sebagai salah satu syarat untuk 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 membantu secara langsung maupun tidak, maka penulis mengucapkan terimakasih kepada :

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

Penulis menyadari proposal skripsi ini tidak luput dari berbagai kekurangan. Penulis mengharapkan kritik dan saran demi kesempurnaan dan perbaikan sehingga akhirnya proposal skripsi ini dapat dijadikan acuan tindak lanjut penelitian selanjutnya dan bermanfaat untuk perkembangan ilmu pengetahuan pada umumnya, khususnya mahasiswa Teknik Informatika Universitas PGRI Ronggolawe Tuban.

Tuban, 19 Februari 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, 25 Februari 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, 25 Februari 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
<b>HALAMAN JUDUL .....</b>	<b>i</b>
<b>KATA PENGANTAR.....</b>	<b>ii</b>
<b>HALAMAN PERSETUJUAN .....</b>	<b>iii</b>
<b>HALAMAN PENGESAHAN .....</b>	<b>iv</b>
<b>DAFTAR ISI.....</b>	<b>v</b>
<b>DAFTAR GAMBAR.....</b>	<b>vii</b>
<b>DAFTAR TABEL .....</b>	<b>viii</b>
<b>BAB I PENDAHULUAN</b>	
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
<b>BAB II TINJAUAN PUSTAKA DAN DASAR TEORI</b>	
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.....	12
2.2.4 Database .....	12
2.2.5 Database Management System (DBMS).....	13
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).....	18
2.2.10 Application Programming Interface (API).....	19
2.2.11 RESTful API .....	19
2.2.12 JavaScript Object Notation (JSON) .....	20
2.2.13 Framework .....	20

2.2.14	Laravel.....	21
2.2.15	Dart.....	22
2.2.16	SDK (Software Development Kit) .....	22
2.2.17	Flutter .....	22
2.2.18	AVD (Android Virtual Devices) .....	23
2.2.19	JDK (Java Development Kit) .....	23
2.2.20	Black Box Testing .....	23
 <b>BAB III METODE PENELITIAN</b>		
3.1	Prosedur Penelitian.....	25
3.1.1	Studi Literatur .....	26
3.1.2	Observasi .....	26
3.1.3	Analisis Kebutuhan .....	26
3.1.4	Desain Sistem .....	28
3.1.5	Pembuatan Dokumentasi API .....	29
3.1.6	Pembuatan RESTful API .....	29
3.1.7	Implementasi .....	30
3.1.8	Pengujian .....	30
3.1.9	Penulisan Laporan .....	30
3.2	Bahan dan Alat Penelitian .....	30
3.2.1	Bahan.....	30
3.2.2	Alat .....	31
3.3	Arsitektur Sistem.....	32
 <b>BAB IV JADWAL PENELITIAN</b>		
4.1	Jadwal Penelitian.....	33
<b>DAFTAR PUSTAKA .....</b>		<b>35</b>

## DAFTAR GAMBAR

	Halaman
Tabel 2.1 Penelitian terkait.....	5
Tabel 2.2 Simbol Use Case Diagram (Henderi, 2009).....	14
Tabel 2.3 Simbol Activity Diagram (Henderi, 2009).....	15
Tabel 2.4 Simbol Class Diagram (Henderi, 2009) .....	16
Tabel 2.5 Simbol Data Flow Diagram (Weli 2019) .....	17
Tabel 4.1 Jadwal Penelitian.....	33

## DAFTAR TABEL

	Halaman
Gambar 3.1 Prosedur Penelitian.....	25
Gambar 3.2 Arsitektur Sistem.....	32



# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Pendidikan tinggi berfungsi untuk mengembangkan kemampuan dan membentuk watak serta peradaban bangsa yang bermartabat dalam rangka mencerdaskan kehidupan bangsa, serta mengembangkan ilmu pengetahuan dan teknologi dengan memperhatikan dan menerapkan nilai Humaniora. Selain itu termasuk dalam tujuan pendidikan tinggi adalah menghasilkan lulusan yang menguasai cabang ilmu pengetahuan dan/atau teknologi untuk memenuhi kepentingan nasional dan meningkatkan daya saing bangsa serta menghasilkan ilmu pengetahuan dan teknologi melalui penelitian dengan memperhatikan nilai Humaniora agar bermanfaat bagi kemajuan bangsa, serta kemajuan peradaban dan kesejahteraan umat manusia. Pendidikan tinggi bisa menyelenggarakan pendidikan akademik, vokasi, dan profesi, dengan program pendidikan diploma (D1, D2, D3, D4), sarjana (S1), magister (S2), doktor (S3), dan spesialis. (UU RI No. 12 Tahun 2012 Tentang Pendidikan Tinggi, 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 ada beberapa hal yang perlu diperhatikan, salah satunya yaitu 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).

Saat ini telah banyak penelitian yang membahas tentang sistem informasi monitoring dan evaluasi tugas akhir mahasiswa, tetapi ada beberapa hal yang perlu diperhatikan, karena dari penelitian sebelumnya hanya menghasilkan sebuah

website 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 akan menghambat pengembang dalam mengembangkan sistem tersebut.

Terdapat sebuah teknologi yang dapat membantu pengembang dalam penyediaan data atau *resource* yaitu RESTful API. Berdasarkan uraian permasalahan diatas, maka penulis akan melakukan penelitian dengan judul “PERANCANGAN DAN IMPLEMENTASI RESTful API PADA SISTEM INFORMASI MONITORING DAN EVALUASI TUGAS AKHIR MAHASISWA”, yang mana pada penelitian ini penulis mengembangkan RESTful API dengan menggunakan *framework* Laravel dengan menggunakan JSON (*JavaScript Object Notation*) sebagai format pertukaran data antara *client* dan *server* melalui metode HTTP (*Hypertext Transfer Protocol*), kemudian untuk menguji *request* dan *response* dari RESTful API yang telah dibuat akan diimplementasikan kedalam dua jenis *platform* yaitu *mobile app* dan website.

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**

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, maka diperoleh rumusan masalah sebagai berikut :

1. Bagaimana merancang RESTful API pada sistem informasi 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

Berdasarkan permasalahan yang timbul diatas maka perlu adanya batasan yang jelas dalam penelitian ini, yaitu :

1. Untuk representasi data yang akan dijadikan sebagai pesan antara *client-server* menggunakan format JSON.
2. Dalam penelitian ini melibatkan sistem informasi berbasis website dan mobile sebagai salah satu alat untuk menguji *request* dan *response* dari *server*.
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 yang sedang melakukan pengerjaan tugas akhir pada Fakultas Teknik Universitas PGRI Ronggolawe (Unirow) Tuban.
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

Penelitian ini bertujuan untuk :

1. Merancang dan membangun RESTful API pada 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

Manfaat dari penelitian ini adalah sebagai berikut :

1. Bagi 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. Bagi pengguna
  - a. Memudahkan ketersediaan 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

Penelitian ini mengacu pada beberapa sumber pustaka yang berhubungan dengan kasus dan metode yang akan diteliti, adapun beberapa jurnal yang digunakan sebagai dasar dan referensi dalam penelitian ini antara lain :

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 Monitoring Dan Evaluasi Tugas Akhir Mahasiswa Berbasis Website yang dapat mempermudah mahasiswa dalam melakukan pengajuan skripsi.

2.	(Bagus et al., 2020)	Sistem Informasi Tugas Akhir Program Studi Teknik Informatika Universitas Mataram	Pengembangan sistem informasi berbasis website menggunakan metode waterfall dan diagram UML ( <i>Unified Modeling Language</i> )	Proses administrasi skripsi masih berjalan secara manual, dimana pada saat pendaftaran judul, mahasiswa harus mengisi formulir judul skripsi, serta pendaftaran proposal seminar, tugas akhir dan sesi yudisial. Prosesnya membutuhkan waktu yang lama sehingga pelayanan administrasi akhir menjadi kurang efektif.	Sistem Informasi Tugas Akhir berbasis website dengan menggunakan <i>framework</i> Laravel
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	Pengembangan RESTful API untuk mengolah nilai mahasiswa, presensi mahasiswa, dan pencetakan berkas perkuliahan.

				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	Sistem informasi berbasis website dengan menggunakan teknologi <i>web service</i> dan <i>framework</i> Laravel	Pincher ID masih berbentuk aplikasi <i>mobile</i> menyebabkan Pincher ID hanya bisa diakses melalui perangkat <i>smartphone</i> atau <i>tablet</i> dengan bantuan pihak ketiga (Google Playstore/ App Store). Aplikasi <i>mobile</i> Pincher ID juga tidak dapat diindeks oleh mesin pencari sehingga menyulitkan masyarakat dalam menemukan aplikasi Pincher ID.	Pengembangan aplikasi Pincher ID yang semula berbasis <i>mobile</i> menjadi website dengan menggunakan <i>web service</i> dalam pengintegrasian data.

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 Sintesis ( <i>Synchronized Student's Final Project Management System</i> ) dan SISKAD (Sistem Informasi Akademik) di 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> .	Implementasi RESTful <i>Web service</i> dengan menggunakan bahasa pemrograman PHP di atas kerangka kerja ( <i>framework</i> ) Lumen Framework dengan format data <i>JSON</i> dapat dipergunakan untuk mengintegrasikan data.
6.	(Sutrisno et al., 2019)	Perancangan sistem pemasangan iklan online pada aplikasi <i>e-commerce</i> (e-gemansa) menggunakan metode RESTful API dan <i>framework</i> Laravel	RESTful API sebagai metode transfer data dan menggunakan UML sebagai metode penggambaran sistem	Belum adanya manajemen iklan yang terintegrasi dengan banyak web player. Dan sistem <i>e-gemansa</i> yang belum berorientasi servis membuat proses pengembangan sistem menjadi sulit.	Penggunaan metode RESTful API akan mempermudah proses pengembangan sistem dan integrasi sistem dengan berbagai <i>platform</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 sistem informasi monitoring dan evaluasi tugas akhir mahasiswa yang dikembangkan dengan menggunakan *framework Yii* dan *database management system (DBMS) MySQL*. Dari penelitian ini diharapkan dapat meningkatkan penyediaan informasi terkait dengan data mahasiswa yang sedang mengerjakan tugas akhir, memberikan panduan dan prosedur dalam pengerjaan 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 informasi tugas akhir yang diharapkan dapat mempersingkat proses pelayanan administrasi dalam pengerjaan 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-gemana) 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 metode-metode yang tertulis dalam penelitian diatas masih dapat dikembangkan lagi terlebih penggunaan metode RESTful API sebagai sarana komunikasi antara *server* dan *client* kedepannya akan berkembang sangat pesat. Sehingga penulis bermaksud melakukan penelitian yang berjudul Perancangan Dan Implementasi RESTful API Pada Sistem Informasi Monitoring Dan Evaluasi Tugas Akhir Mahasiswa. Pada penelitian ini penulis mengembangkan RESTful API dengan menggunakan *framework* Laravel dengan menggunakan JSON (*JavaScript Object Notation*) sebagai format pertukaran data antara *client* dan *server* melalui metode HTTP (*Hypertext Transfer Protocol*), kemudian untuk menguji *request* dan *response* dari RESTful API yang telah dibuat akan diimplementasikan kedalam dua jenis platform yaitu *mobile app* yang dibangun dengan menggunakan *Flutter mobile app SDK (Software Development Kit)* dan website yang dibangun dengan menggunakan *framework* Laravel. Selain itu dengan melakukan pendokumentasian API yang digunakan dalam sebuah dokumen khusus sehingga mudah untuk diimplementasikan oleh pengembang.

## **2.2 Dasar Teori**

### **2.2.1 Sistem Informasi**

Menurut (Sutabri, 2012) sistem informasi adalah suatu sistem dalam suatu organisasi untuk mengelola kebutuhan pengelolaan transaksi harian yang mendukung fungsi operasi manajerial organisasi dengan kegiatan strategi dari organisasi untuk menyediakan laporan atau informasi kepada pihak luar.

Sistem informasi terdiri dari beberapa komponen yang disebut dengan blok bangunan atau *building block*, yaitu :

- a. *Input block* atau blok masukan, merupakan input yang mewakili data yang masuk ke dalam sistem informasi.

- b. *Model block* atau blok model, yang terdiri dari kombinasi prosedur, logika dan metode matematik yang digunakan untuk memanipulasi data input untuk kemudian disimpan dalam basis data dengan cara yang sudah ditentukan yang selanjutnya menghasilkan keluaran yang diinginkan.
- c. *Output block* atau blok keluaran, yang merupakan informasi yang dihasilkan dari sebuah sistem informasi.
- d. *Technology block* atau blok teknologi, adalah teknologi yang digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran untuk membantu pengendalian informasi atau data secara keseluruhan.
- e. *Data base block* atau blok basis data, yang merupakan kumpulan dari data yang saling berhubungan antara satu dengan yang lainnya yang tersimpan dalam perangkat komputer.
- f. *Control block* atau blok kendali, merupakan pengendalian yang dirancang dan ditetapkan untuk mencegah atau mengatasi segala kemungkinan yang dapat merusak sistem.

### 2.2.2 PHP

Menurut (Jannah, et al., 2019) PHP atau *Hypertext Preprocessor* merupakan bahasa pemrograman *script server-side* atau berbentuk *script* yang disimpan dan dijalankan di komputer *server (Web Server)* sedangkan hasilnya dikirimkan ke komputer *client (Web Browser)* dalam bentuk *script HTML (Hypertext Markup Language)* yang didesain untuk pengembangan web dan bersifat *open source*.

Sedangkan menurut (Supaartagorn et al., 2010) PHP adalah *server-side scripting language* yang dirancang untuk aplikasi berbasis web. Banyak kelebihan dari bahasa pemrograman PHP, antara lain dari segi performa, skalabilitas, portabilitas, dan merupakan bahasa pemrograman yang *open source*, terutama untuk terkoneksi dan melakukan manipulasi terhadap sebuah basis data.

### 2.2.3 MySQL

Menurut (Amin, 2018) MySQL merupakan sebuah perangkat lunak /software sistem manajemen basis data SQL atau DBMS Multithread dan multi user yang memungkinkan pengoperasian data dikerjakan secara mudah dan otomatis.

Sedangkan menurut (Turban, 2017) MySQL adalah sebuah *database manajemen system* (DBMS) populer yang memiliki fungsi sebagai *relational database manajemen system* (RDBMS) yang bersifat *open source* dengan kinerja sangat cepat serta mudah untuk digunakan pada arsitektur *client server* atau *embedded systems*.

### 2.2.4 Database

(Octavian, 2013) menerangkan dalam buku Simarmata & Paryudi (2006:1) bahwa database atau basisdata adalah sebagai berikut:

- a. Menurut Stephens dan Plew (2000), *database* merupakan sebuah mekanisme yang digunakan untuk menyimpan data atau informasi.
- b. Silberschatz, dkk (2002) mengartikan basis data sebagai kumpulan data berisi informasi yang sesuai untuk sebuah perusahaan.
- c. Menurut Ramakrishnan dan Gehrke (2003) menyatakan basisdata sebagai kumpulan data, umumnya mendeskripsikan aktivitas suatu organisasi atau lebih yang berhubungan.
- d. Menurut McLeod, dkk (2001), mengartikan basis data adalah kumpulan seluruh sumber daya berbasis komputer milik organisasi.

Kemudian (Yudhanto & Adi, 2018) menjelaskan bahwa basis data (*database*) adalah kumpulan data yang disimpan secara sistematis didalam komputer yang dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan data atau informasi.

### 2.2.5 Database Management System (DBMS)

Menurut (Yanto 2016) Database Management System (DBMS) merupakan paket program (*Software*) yang dibuat untuk memudahkan dan mengefisienkan pemasukan, pengeditan, penghapusan serta pengambilan informasi terhadap database. Yang termasuk dalam DBMS antara lain, *Microsoft SQL, MySQL, Oracle, MS. Access*, dan lain-lain.

Adapun penyimpanan data dengan DBMS mempunyai manfaat dan kelebihan antara lain :

1. Efisiensi dalam hal media penyimpanan dan penggunaan memori.
2. Integritas data akan lebih terjamin.
3. Pembuatan antarmuka kedalam data akan lebih mudah.
4. Mempermudah pengelolaan basis data.
5. Memiliki sistem keamanan yang lebih fleksibel.

### 2.2.6 Unified Modeling Language (UML)


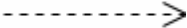






Menurut (Mubarak, 2019) UML (*Unified Modeling Language*) adalah sebuah bahasa yang memberikan standar penulisan sebuah *blue print* sistem berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis *Object-Oriented*.


Sedangkan menurut (Dharwiyanti & Wahono, 2003) UML adalah sebuah standar untuk merancang model sebuah sistem dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem perangkat lunak. Dalam perancangan sistem dengan menggunakan UML kita dapat menggunakan alat bantu sebagai berikut :

#### a. Use Case Diagram (UCD)

*Use case diagram* adalah rangkaian diagram yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor yang menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, yang menjelaskan tentang “apa” yang diperbuat sistem, dan bukan “bagaimana” sistem tersebut berjalan.

Tabel 2.2 Simbol Use Case Diagram (Henderi, 2009)







No	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang dimainkan pengguna ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan perubahan yang terjadi pada suatu elemen <i>independent</i> (mandiri) akan mempengaruhi elemen yang bergantung padanya elemen yang <i>independent</i> (tidak mandiri).
3		<i>Generalization</i>	Hubungan dimana objek <i>descendent</i> (anak) berbagi perilaku dan struktur data dari objek yang ada di atasnya <i>ancestor</i> (objek induk).
4		<i>Include</i>	Menspesifikasikan <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa hubungan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Penjelasan dari aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.

9		<i>Collaboration</i>	Interaksi antara aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemennya.
---	---	----------------------	---

b. Activity Diagram

*Activity diagram* merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan transisi di-trigger oleh selesainya *state* sebelumnya (*internal processing*) yang menggambarkan berbagai alir aktivitas dalam sistem yang dirancang, bagaimana awal masing-masing alir, percabangan yang mungkin terjadi, dan bagaimana *action* tersebut berakhir.



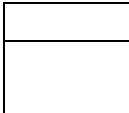



Tabel 2.3 Simbol Activity Diagram (Henderi, 2009)

No.	Simbol	Nama	Keterangan
1		<i>Initial Node</i>	Sebuah diagram aktivitas memiliki sebuah statusawal.
2		<i>Activity</i>	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3		<i>Decision</i>	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
4		<i>Activity Final Node</i>	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
5		<i>Fork</i>	Menggambarkan kegiatan yang dilakukan secara paralel.
6		<i>Join</i>	Digunakan untuk menunjukkan kegiatan yang digabungkan.

c. Class Diagram

*Class diagram* adalah gambaran struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain, yang menggambarkan atribut / property suatu sistem, sekaligus menawarkan layanan untuk memanipulasi metoda/fungsi.

Tabel 2.4 Simbol Class Diagram (Henderi, 2009)

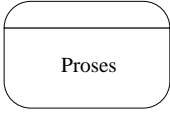
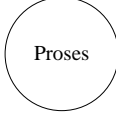
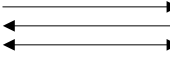
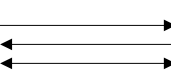
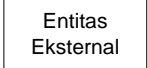
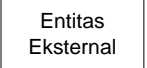
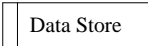

No	Simbol	Nama	Keterangan
1		<i>Generalization</i>	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)
2		<i>Nary Association</i>	Untuk menghindari asosiasi yang lebih dari 2 objek
3		<i>Class</i>	Kelas pada struktur sistem
4		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh sebuah objek.
5		<i>Dependency</i>	Relasi antar kelas dengan makna kebergantungan antar kelas.
6		<i>Association</i>	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity.

### 2.2.7 Data Flow Diagram (DFD)

Menurut (Weli 2019) DFD (*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 symbol yang digunakan dalam membuat DFD sebagai *Modeling Tool* 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 / proses</i>	Menggambarkan bagian dari sistem yang mentransformasikan <i>input</i> menjadi <i>output</i> , yang menjelaskan proses/kegiatan yang sedang/akan berjalan.
2			<i>Data Flow / aliran data</i>	Menggambarkan data yang mengalir dalam sistem yang sedang berjalan.
3			<i>Terminator</i>	Simbol yang mewakili entitas eksternal, baik dari dalam organisasi maupun dari luar organisasi yang memiliki kepentingan dengan sistem yang sedang berjalan.
4			<i>Data Store / tempat penyimpanan</i>	Berbagai media untuk menyimpan data.

### 2.2.8 Web Service

Menurut (Sutanta & Mustofa, 2012) *Web service* merupakan sebuah mekanisme interaksi antar sistem yang menunjang interoperabilitas dengan tujuan untuk suatu kepentingan integrasi data yang dapat diakses melalui internet oleh banyak pihak dan media akses dapat memanfaatkan teknologi berbeda. Selain itu *web service* dapat dipahami sebagai RPC (*Remote Procedure Call*) yang mampu

memproses fungsi-fungsi yang didefinisikan pada sebuah aplikasi *web* dan mengekspos sebuah API (*Application Programming Interface*) melalui *web*, *web service* memiliki beberapa kelebihan, yaitu :

1. Lintas *platform*, memungkinkan terjadinya pertukaran data, walaupun menggunakan perangkat dengan sistem operasi yang berbeda.
2. *Language independent*, dapat diakses dan dibangun oleh bahasa pemrograman apapun.
3. Jembatan penghubung dengan *database* tanpa perlu *driver database* dan tidak harus mengetahui jenis DBMS (*Database Management System*) yang digunakan.
4. Proses pertukaran data menjadi semakin lebih mudah dan cepat tanpa harus menyesuaikan aplikasi, *database*, dan *platform* yang digunakan.
5. Penggunaan kembali komponen aplikasi, aplikasi-aplikasi yang berbeda dapat menggunakan sebuah fungsi yang sama.

#### **2.2.9 Representational State Transfer (REST)**

(Pautasso & Wilde, 2010) menerangkan bahwa REST (*Representational State Transfer*) adalah seperangkat prinsip arsitektur yang digunakan untuk membangun *web services* yang fokus pada sumber daya sistem, termasuk bagaimana sumber daya *states* ditunjukkan dan ditransfer melalui HTTP oleh berbagai *client* yang ditulis kedalam bahasa pemrograman yang berbeda. REST menggunakan metode 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 sebuah dokumentasi pengembangan perangkat lunak yang mendeskripsikan fitur atau layanan yang disediakan oleh sebuah aplikasi atau sistem dengan tujuan untuk mempermudah pengembang untuk mengetahui sistem yang sedang berjalan sekaligus sebagai panduan untuk menggunakan fitur-fitur yang sudah disediakan.

Sedangkan menurut (Cited & Data, 2017) API (*Application Programming Interface*) adalah uraian *interface* suatu sistem atau aplikasi untuk berkomunikasi antara satu sistem dengan sistem yang lain.

#### **2.2.11 RESTful API**

(Dawood, 2017) menjelaskan salah satu kriteria desain *web services* yang paling sering digunakan adalah *RESTful web services* yang bekerja dengan cara *resource-oriented*. Pada *RESTful web service* permintaan dari *client* untuk mengakses *service* yang ditawarkan oleh *server* dilakukan dengan cara mengakses URL (*Uniform Resource Locator*) dari *resource* menggunakan method HTTP.

Kemudian (Sutrisno et al., 2019) mendefinisikan *RESTful API* sebagai sebuah sistem terdistribusi dengan menggunakan *web service* yang menyediakan sebuah kapabilitas di internet dengan mendokumentasikan *interface* dengan metode API dengan tujuan memudahkan pengembang dalam menggunakan fitur-fitur atau *resource* yang sudah disediakan oleh sistem yang sedang berjalan.

(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*.

### 2.2.12 JavaScript Object Notation (JSON)

(Dawood, 2017) menerangkan bahwa JSON (*JavaScript Object Notation*) adalah format pertukaran data yang mudah dibaca dan ditulis, serta ringan, sehingga mudah dibuat (*generate*) dan diterjemahkan. JSON merupakan format teks yang tidak tergantung pada bahasa pemrograman apapun karena menggunakan bahasa yang umum digunakan oleh *programmer*, sehingga ideal sebagai bahasa pertukaran data. Selain itu JSON memiliki beberapa kelebihan dibandingkan dengan XML (*Extensible Markup Language*), antara lain :

1. Untuk merepresentasikan sebuah struktur data yang rumit dan berbentuk hirarkis penulisan, JSON relatif lebih terstruktur dan mudah.
2. Ukuran karakter yang dibutuhkan JSON lebih kecil dibandingkan XML untuk data yang sama.
3. Melakukan browser parsing pada JSON lebih sederhana dibandingkan pada XML, JSON menggunakan function JavaScript `eval()` untuk melakukan parsing. Sementara dokumen XML diparsing oleh XML HTTP Request

### 2.2.13 Framework

(Yudhanto & Adi, 2018) mengartikan *framework* atau kerangka kerja sebagai kumpulan *script* (terutama *class* dan *function*) yang dapat membantu programmer dalam menangani berbagai masalah-masalah dalam pemrograman, seperti pemanggilan variabel, file, dan koneksi ke basis data, sehingga programmer dapat lebih fokus dan lebih cepat dalam membangun aplikasi, selain itu dengan komponen pemrograman yang siap digunakan ulang kapan saja programmer tidak perlu menulis kode program yang sama untuk fungsi yang sama.

Sedangkan menurut (Abdulloh, 2017) *framework* diartikan sebagai kumpulan kode-kode program yang dikumpulkan dan disusun secara rapi pada folder-folder agar mudah digunakan pada setiap pembuatan aplikasi, yang mana dengan keberadaan *framework* sangat membantu dalam pembuatan aplikasi karena kita tidak perlu membuat segala sesuatu dari awal.

#### 2.2.14 Laravel

Menurut (Chen et al., 2017) laravel merupakan *framework* PHP yang bersifat *open-source* yang diciptakan oleh Taylor Otwell untuk pengembangan aplikasi web yang 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.

(Luthfi, 2017) menjelaskan bahwa laravel memberikan keterbaruan alat untuk berinteraksi dengan *database* disebut dengan *migration*. Pengembang dapat dengan mudah untuk melakukan modifikasi sebuah *database* pada sebuah *platform* secara independen karena implementasi skema *database* direpresentasikan dalam sebuah class melalui *migration*. Selain itu *migration* juga dapat berjalan pada basis data yang telah didukung Laravel (*MySQL*, *PostgreSQL*, *MSSQL*, dan *SQLITE*) dan untuk implementasi *Active Record* pada Laravel disebut *Eloquent* yang menggunakan standard modern *OOP* (*Object Oriented Programming*).

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

1. Memiliki banyak fitur yang tidak dimiliki oleh *framework* lain.
2. Sintaks pada laravel menggunakan bahasa yang mudah dimengerti sehingga akan mudah dipahami oleh programmer pemula sekalipun.
3. Laravel memiliki dokumentasi yang cukup lengkap pada setiap versinya.
4. Dgunakan oleh banyak programmer, sehingga banyak *library* yang mendukung laravel.

5. Didukung oleh composer, sehingga *library- library* laravel dapat diperoleh dengan mudah dari internet.
6. Memiliki *template engine* tersendiri yang diberi nama *blade*, sehingga memudahkan programmer dalam menampilkan data pada template HTML.

### 2.2.15 Dart

Menurut (Tjandra & Chandra, 2020) dart merupakan bahasa pemrograman yang dibuat oleh google pada tahun 2011 untuk menggantikan Javascript, dart menggunakan *static typing* yang berarti sebelum memakai sebuah variabel, variabel tersebut perlu didefinisikan terlebih dahulu. *Syntax* pada dart sangat mudah dipelajari karena memiliki kemiripan dengan *syntax-syntax* pemrograman lain, seperti javascript dan java.

Sedangkan menurut (Suryono & Hardiansah, 2020) dart merupakan bahasa pemrograman yang berorientasi pada objek serta menggunakan gaya penulisan bahasa C yang terkompilasi secara opsional kedalam bahasa pemrograman Javascript.

### 2.2.16 SDK (*Software Development Kit*)

Menurut (RIZQI, 2014) *SDK (Software Development Kit)* adalah perangkat *API (Application Programming Interface)* yang digunakan untuk pengembangan aplikasi pada perangkat android.

Sedangkan menurut (Yuntoto, 2015) merupakan *API (Application Programming Interface)* yang digunakan sebagai unit-unit pengujian aplikasi.

Kemudian Nazruddin dalam (Mubarak, 2017) menerangkan bahwa SDK merupakan *tool API (Application Programming Interface)* yang diperlukan dalam pengembangan aplikasi pada *platform* Android menggunakan bahasa pemrograman java.

### 2.2.17 Flutter

(Tjandra & Chandra, 2020) menerangkan bahwa flutter adalah *SDK (Software Development Kit)* untuk mengembangkan aplikasi mobile android dan iOS dengan kinerja tinggi hanya dengan satu *codebase* (basis code) yang dibuat

oleh google dengan lisensi *open source*, yang memungkinkan programmer untuk menghadirkan aplikasi dengan berkinerja tinggi pada *platform* yang berbeda.

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*. Selain itu flutter juga menawarkan fitur *hot reload* yang dapat membantu dalam proses pembuatan *user interface*.

#### **2.2.18 AVD (*Android Virtual Devices*)**

(Yuntoto, 2015) menjelaskan bahwa *AVD (Android Virtual Devices)* merupakan bagian dari SDK Android yang berfungsi sebagai *emulator* untuk menjalankan aplikasi berbasis android.

Sedangkan menurut Nazruddin dalam (Mubarak, 2017) *AVD* merupakan *emulator* untuk menjalankan program aplikasi Android yang dijadikan sebagai tempat tes yang berjalan pada *virtual Machine*.

#### **2.2.19 JDK (*Java Development Kit*)**

Nazruddin dalam (Mubarak, 2017) menjelaskan bahwa *JDK (Java Development Kit)* merupakan perangkat lunak yang digunakan untuk melakukan proses kompilasi kedalam *bytecode* sehingga kode program dapat dibaca dan dijalankan oleh *JRE (Java Runtime Environment)*. Sehingga *JDK* harus terinstall pada komputer yang melakukan pembuatan aplikasi menggunakan java, namun tidak harus terinstall pada komputer yang akan menjalankan aplikasi yang dibangun dengan java.

#### **2.2.20 Black Box Testing**

Menurut (Mubarak, 2017) *Black box testing* adalah metode pengujian perangkat lunak yang menguji fungsionalitas aplikasi yang tidak sesuai dengan struktur internal atau cara kerja sistem.

Sedangkan menurut (Mustaqbal et al., 2015) *Black box testing* merupakan pengujian perangkat lunak yang hanya berfokus pada spesifikasi fungsionalitas dari perangkat lunak untuk menemukan kesalahan atau ketidaksesuaian fungsi, *interface errors* / kesalahan tampilan, kesalahan pada struktur data dan akses basis data, kesalahan terminasi dan inisialisasi, *performance errors* / kesalahan performansi. Sehingga dapat menjawab pertanyaan-pertanyaan berikut :

1. Bagaimana fungsi-fungsi diuji supaya dapat dinyatakan valid ?
2. Input seperti apa yang dapat menjadi bahan pengujian yang baik ?
3. Apakah sistem sensitif pada input-input tertentu ?
4. Berapa banyak data yang dapat ditangani sistem ?
5. Bagaimana sekumpulan data dapat diisolasi ?
6. Apa efek yang dapat membuat kombinasi data ditangani spesifik pada sistem operasi ?

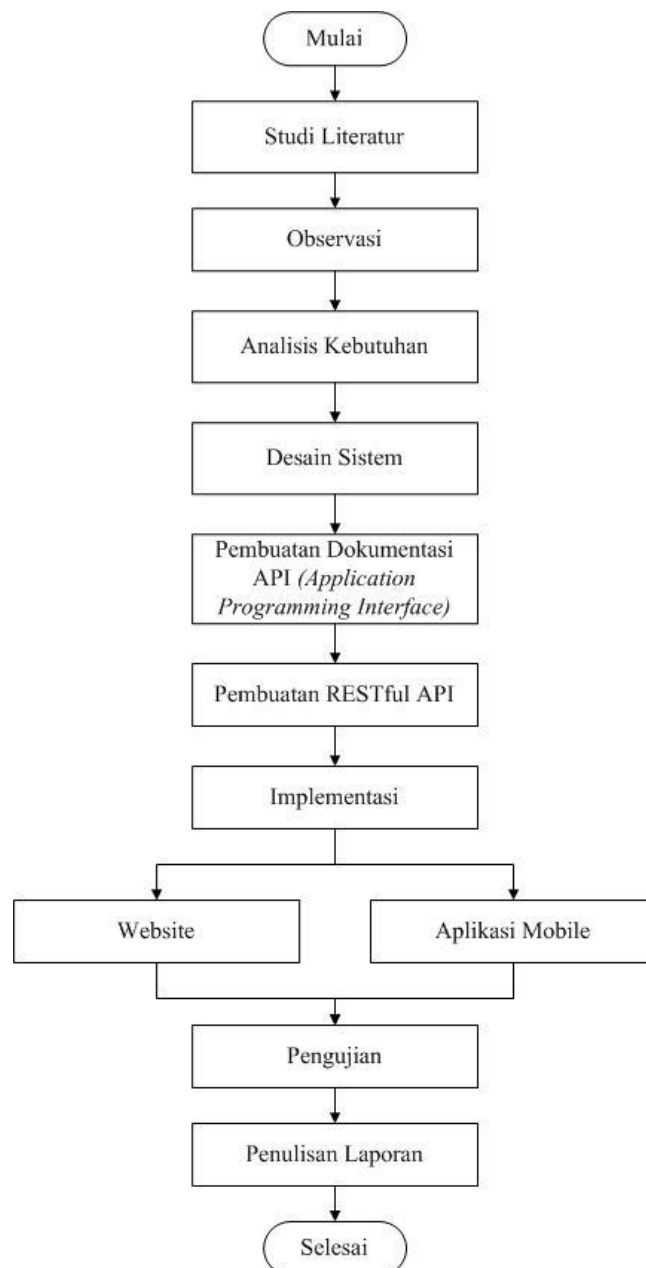


## BAB III

### METODE PENELITIAN

#### 3.1 Prosedur Penelitian

Dalam sebuah penelitian diperlukan sebuah tahapan yang sistematis sehingga penelitian dapat berjalan sesuai dengan yang diharapkan. Dalam penelitian ini penulis menggunakan prosedur penelitian seperti pada gambar berikut.



Gambar 3.1 Prosedur Penelitian

### 3.1.1 Studi Literatur

Penelitian ini diawali dengan proses pengumpulan literatur-literatur serta kajian yang berkaitan dengan perancangan dan implementasi RESTful API, kajian-kajian tersebut dapat dapat diambil dari buku, jurnal, dan sumber-sumber lain yang terkait. Terdapat 6 jurnal utama yang menjadi referensi penulis dalam penelitian ini, seperti yang dijelaskan 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 dikembangkan dengan bahasa pemrograman PHP dengan *framework Yii* dan *DBMS (database management system) MySQL*. 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

Pada tahap ini 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. Proses Login
  - b. Proses Pengolahan Data Dosen
  - c. Proses Pengolahan Data Mahasiswa
  - d. Proses 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 menyimpan data pengguna.
  - b. Tabel prodi 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 menyimpan 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 menyimpan data kecamatan yang ada di kabupaten Tuban.
  - m. Tabel kelurahan untuk menyimpan data keluaran yang ada di 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 yang telah dibuat.

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

#### 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 menggunakan UML (*Unified Modeling Language*), yaitu :
  - a. *Use Case Diagram (UCD)* yang akan memberikan gambaran atau interaksi yang dilakukan oleh aktor terhadap sistem.
  - b. Skenario Use Case untuk menjabarkan setiap use case yang ada, sekaligus sebagai acuan pada tahap proses pengujian sistem.
  - c. *Activity Diagram* untuk menggambarkan proses bisnis dan urutan aktifitas dari setiap proses secara keseluruhan.
  - d. *Class Diagram* untuk menggambarkan struktur dan deskripsi *class*, *package*, dan objek serta hubungannya satu sama lain.

2. Dari *blue print* sistem yang telah dibuat, proses selanjutnya adalah merancang *Data Flow Diagram (DFD)* untuk menggambarkan aliran data 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 dengan bahasa pemrograman PHP dan sistem informasi berbasis *mobile* yang dibangun menggunakan Flutter *SDK (Software Development Kit)* versi 3.12.2 dengan bahasa pemrograman Dart versi 3.11.0.

### **3.1.8 Pengujian**

Tahapan selanjutnya yaitu dilakukan pengujian Sistem Informasi Monitoring Dan Evaluasi Tugas Akhir Mahasiswa berbasis website maupun *mobile* yang telah dibuat dengan metode *Black box testing* untuk mengetahui apakah *RESTful API* yang telah dibuat dapat berjalan dengan baik sesuai dengan Skenario Use Case yang telah dirancang pada tahap desain sistem.

### **3.1.9 Penulisan Laporan**

Setelah dilakukan pengujian selanjutnya dilakukan penulisan laporan hasil penelitian, pada tahap ini dilakukan dokumentasi proses penelitian dari tahap awal sampai tahap akhir, sehingga dapat memberikan informasi atau penjelasan tentang penelitian yang telah dilaksanakan.

## **3.2 Bahan dan Alat Penelitian**

### **3.2.1 Bahan**

Bahan yang diperlukan dalam penelitian ini adalah informasi berupa 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 pada penelitian sebelumnya yang telah dilakukan oleh (Fitrianto, 2017).

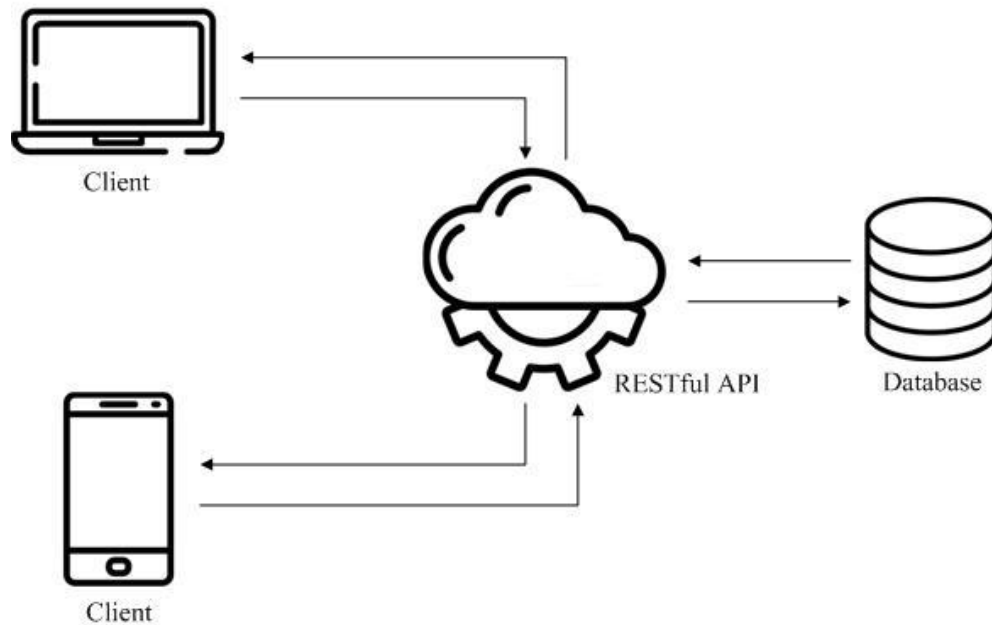
### 3.2.2 Alat

Peralatan yang digunakan penulis dalam mengerjakan penelitian ini meliputi perangkat keras (*hardware*) dan perangkat lunak (*software*), yaitu :

1. Perangkat keras (*hardware*)
  - a. Processor AMD A8-7410 APU with AMD Radeon R5 Graphics (4 CPUs), ~2.2GHz
  - b. RAM DDR3 berkapasitas 12 GB
  - c. Harddisk berkapasitas 500 GB
  - d. Monitor dengan resolusi 1366 x 768 px
  - e. Perangkat *mouse* dan *keyboard* standar
2. Perangkat lunak (*software*)
  - a. Microsoft Office Visio, yang digunakan untuk membuat *blue print* dalam proses desain sistem.
  - b. MySQL Workbench 8.0 CE, yang digunakan untuk membuat desain *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. Visual Studio Code, sebagai *text editor*.
  - f. Laravel versi 7.0, sebagai *framework* untuk PHP yang digunakan dalam proses pembuatan *RESTful API* sekaligus mengimplementasikan *RESTful API* pada sistem informasi berbasis website.
  - g. XAMPP v3.2.4, yang digunakan sebagai *local web server*.
  - h. Postman, yang digunakan untuk melakukan 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 yang diilustrasikan pada gambar 3.2 berikut.



Gambar 3.2 Arsitektur Sistem

Secara umum, arsitektur sistem terbagi menjadi tiga bagian utama yaitu : *client*, *RESTful API*, dan *database*. *RESTful API* menyediakan fungsi-fungsi yang dapat diakses oleh *client*, sedangkan *client* melakukan *request* kepada *RESTful API* untuk mendapatkan *response* data dalam format JSON sesuai yang dibutuhkan. 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 uiiperkuliahan 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