

PROPOSAL SKRIPSI

**RANCANG BANGUN MODUL WEB SERVICES SISTEM
INFORMASI MANAJEMEN TUGAS AKHIR MAHASISWA**



**Disusun Oleh:
EKO RUSMA UTOMO
(141080200056)**

**PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS MUHAMMADIYAH SIDOARJO
2020**

HALAMAN PERSETUJUAN

RANCANG BANGUN MODUL WEB SERVICES SISTEM INFORMASI MANAJEMEN TUGAS AKHIR MAHASISWA

Penelitian untuk S-1

Jurusan Informatika

Diajukan Oleh :

EKO RUSMA UTOMO

NIM : 141080200056

Telah Disetujui Oleh:

Dosen Pembimbing,

Acc untuk Sidang Proposal



Irwan A. Kautsar, S.Kom., M.Kom., Ph.D
NIK/NIP. 211452

HALAMAN PENGESAHAN

**RANCANG BANGUN MODUL WEB SERVICES SISTEM
INFORMASI MANAJEMEN TUGAS AKHIR MAHASISWA**

Penelitian untuk S-1
Jurusan Informatika

Diajukan Oleh:
Eko Rusma Utomo
141080200056

Disetujui Oleh:

Ketua Penguji

Dosen Penguji

(_____)

(_____)

Disahkan Oleh :

Ka. Prodi Informatika S-1
Universitas Muhammadiyah Sidoarjo

Ir. Sumarno, MM

KATA PENGANTAR

Alhamdulillah, puji dan syukur Penulis panjatkan kepada Allah Subhanahu Wa Ta'ala, yang telah melimpahkan Rahmat, Taufiq dan Hidayah-Nya, sehingga proposal Skripsi ini dapat diselesaikan.

Tujuan dari penyusunan Proposal Skripsi yang berjudul “ **Rancang Bangun Modul Web Services Sistem Informasi Manajemen Tugas Akhir Mahasiswa** “ ini, guna memenuhi salah satu syarat untuk bisa menempuh ujian sarjana pada Fakultas Sains Dan Teknologi Program Studi Informatika di Universitas Muhammadiyah Sidoarjo.

Didalam pengerjaan Proposal Skripsi ini. telah melibatkan banyak pihak yang telah membantu dalam banyak hal. Oleh sebab itu, penulis sampaikan terima kasih kepada :

1. Bapak Dr. Hidayatullah, M.Si selaku Rektor Universitas Muhammadiyah Sidoarjo yang telah menyediakan fasilitas dalam perkuliahan.
2. Bapak Dr. Hindarto, S.kom., MT selaku Dekan Fakultas Sains dan Teknologi Universitas Muhammadiyah Sidoarjo yang telah menyediakan fasilitas dalam perkuliahan.
3. Bapak Ir. Sumarno, MM selaku Kepala Program Studi S1 Informatika.
4. Bapak Irwan A. Kautsar, S.Kom., M.Kom., Ph.D selaku Dosen Pembimbing yang telah memberikan bimbingan dan bantuan dalam penyelesaian proposal skripsi ini.
5. Ibu Yulian Findawati, ST, M.MT selaku Dosen Seminar yang telah memberikan bimbingan dan bantuan dalam penyelesaian proposal skripsi ini.
6. Bapak dan Ibu Dosen Jurusan Informatika yang telah memberikan arahan selama studi.
7. Serta semua pihak yang telah membantu dalam pembuatan proposal ini baik secara langsung maupun tidak.

Penulis berharap semoga kebaikan dan kemurahan hati semua pihak yang telah membantu penyelesaian proposal skripsi ini mendapat balasan yang baik. Kritik dan saran sangat penulis harapkan demi sempurnanya penulisan ini.

Sidoarjo, 23 Desember 2020

(Eko Rusma Utomo)

DAFTAR ISI

Cover	i
Halaman Persetujuan	ii
Halaman Pengesahan	iii
Kata Pengantar	iv
Daftar Isi	vi
Daftar Gambar	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	3
1.6. Sistematika Penulisan	4
BAB II – Kajian Pustaka dan Dasar Teori	
2.1. Penelitian Terdahulu	5
2.2. Sistem Informasi	6
2.3. Tugas Akhir	6
2.4. Web Services	7
2.5. Python	9
2.6. Flask	9
2.7. WSGI	10
2.8. Werkzeug	10
2.9. Jinja2	10
2.10. Representational State Transfer (REST)	10
BAB III – Metode Penelitian	
3.1 Waktu dan Peralatan	13
3.2 Pengumpulan Data	13
3.3 Metode Penelitian	13
3.4 Arsitektur Sistem	14

3.5 Install Flask, Virtual Env dan Package	14
3.6 Flow Chart	15
3.7 DFD	22
3.8 Metode Pengembangan	24
DAFTAR PUSTAKA	28

DAFTAR GAMBAR

Gambar 2.3. Lapisan Dasar Web Services	7
Gambar 3.1 <i>Flow Chart</i> Mahasiswa	16
Gambar 3.2 <i>Flow Chart</i> Dosen Pembimbing	17
Gambar 3.3 <i>Flow Chart</i> Admin.....	18
Gambar 3.4 <i>Flow Chart</i> Kaprodi	19
Gambar 3.5 <i>Flow Chart</i> Rektor	20
Gambar 3.6 <i>Flow Chart</i> Public	21
Gambar 3.7 <i>DFD Level 0</i>	23
Gambar 3.8 <i>DFD Level 1</i>	24
Gambar 3.9 <i>Metode UCD</i>	25
Gambar 3.10 <i>User Centered Design Canvas</i>	27

BAB I

PENDAHULUAN

1.1 Latar Belakang

Tugas Akhir bagi seorang mahasiswa yang lebih populer dengan sebutan skripsi merupakan tugas final yang harus diselesaikan sebelum lulus jenjang pendidikan pada sebuah perguruan tinggi. Dalam proses pelaksanaannya, seorang mahasiswa akan dibimbing oleh dosen yang ditunjuk oleh jurusan yang memiliki latar belakang pengetahuan yang sebidang dengan judul tugas akhir yang diajukan. Dosen dalam bimbingannya akan memberikan arahan dan konsultasi secara regular maupun bertahap agar tugas akhir yang dibuat terarah dan lebih fokus supaya hasil yang diharapkan tercapai. Jadwal konsultasi atau bimbingan ditentukan oleh pembimbing sesuai dengan jadwal yang ada pada dosen pembimbing tersebut.

Rancang bangun modul *web services* sistem informasi tugas akhir mahasiswa ini berfokus untuk mendukung / support rancang bangun perangkat lunak sistem informasi tugas akhir mahasiswa yang telah ada sebelumnya. Perangkat lunak sistem informasi tugas akhir mahasiswa adalah aplikasi yang dirancang dan dibangun untuk memudahkan dosen ketika mengelola artifak-artifak/dokumen yang diajukan oleh mahasiswa sebagai bagian dari proses tugas akhir.

Dengan perkembangan teknologi sekarang ini sebuah aplikasi dapat dikembangkan dengan memberikan fitur-fitur tambahan yang menunjang dari fungsi pokok aplikasi itu sendiri. Seperti contohnya menambah fitur chatting, sistem login yang terintegrasi dengan e-mail atau facebook, dan lain-lain. Dengan latar belakang diatas dibutuhkan sebuah *web services* untuk pengembangan lebih lanjut dari perangkat lunak sistem informasi tugas akhir mahasiswa.

REST (Representational State Transfer) merupakan salah satu metode arsitektur dalam membangun teknologi *web services* yang mengimplentasikan konsep perpindahan antar state dengan digambarkan seperti suatu peramban yang sedang melakukan permintaan suatu halaman situs, sedangkan di sisi server akan melakukan pengiriman state halaman situs yang sekarang ke peramban [1]. Ide dasar dari REST adalah dengan fokus utama pada interaksi sumber daya dan pergantian antar state, bukan berfokus pada mengirim dan menerima pesan seperti layanan web berbasis SOAP. Prinsip dari metode REST, yaitu identifikasi resources melalui URI (Uniform Resource Identifier), adanya keseragaman resources (setiap resources pada REST dimanipulasi oleh empat operasi, yaitu GET, PUT, POST, DELETE untuk create, read, update, delete), self-descriptive messages (dapat mengakses bermacam-macam bentuk format, seperti PDF, HTML, JPEG, XML, JSON, plain text, dsb), hyperlink for stateful interaction [2].

Apabila teknologi *web services* ini diimplementasikan pada permasalahan yang telah diuraikan di atas, maka memungkinkan dilakukannya pengembangan aplikasi untuk menjadikan sistem yang memiliki performa yang baik, cepat dan mudah untuk dikembangkan terutama dalam pertukaran dan komunikasi data.

1.2 Rumusan Masalah

Berdasarkan latar belakang permasalahan yang telah dijelaskan diatas, maka penulis membuat perumusan masalah yang menjadi fokus dari penelitian yaitu sebagai berikut:

Bagaimana mengimplementasikan *web services* dengan metode RESTFull API untuk melakukan pengembangan fitur-fitur pada perangkat lunak sistem informasi manajemen tugas akhir mahasiswa sehingga memiliki performa yang baik, cepat dan mudah dalam pertukaran dan komunikasi data.

1.3 Batasan Masalah

Untuk mengarahkan pembahasan agar searah dan tidak keluar dari tema dan sesuai dengan permasalahan yang di angkat oleh penulis, maka penulis memiliki batasan masalah yaitu sebagai berikut:

1. Bahasa pemograman yang digunakan adalah bahasa pemograman Python menggunakan framework Flask.
2. Produk yang dikembangkan ialah RESTful API.
3. RESTful API diperuntukkan kepada aplikasi berbasis web.
4. Keluaran dari RESTful API ialah format JSON (JavaScript Object Notation).

1.4 Tujuan

Tujuan dari penelitian ini adalah menghasilkan sebuah RESTful API yang dapat digunakan oleh developer dalam pengembangan perangkat lunak manajemen sistem informasi tugas akhir mahasiswa.

1.5 Manfaat

Manfaat dari penelitian rancang bangun perangkat lunak untuk manajemen informasi tugas akhir adalah memberikan *output* dari hasil analisa permasalahan – permasalahan yang di alami oleh para *stakeholder* selama proses bimbingan tugas akhir.

1. Bagi Penulis
Untuk menjalankan kewajiban sebagai syarat kelulusan strata 1.
2. Bagi Universitas
Dapat menjadi tambahan referensi bagi mahasiswa dan dosen untuk melakukan penelitian atau bahan perkuliahan.
3. Bagi Masyarakat Umum
Dapat menjadi referensi bagi developer dalam mengembangkan aplikasi serupa

1.6 Sistematika Penulisan

Sistematika penulisan proposal “Rancang Bangun Modul *Web Services* Untuk Manajemen Informasi Tugas Akhir” sebagai berikut :

BAB I Pendahuluan

Berisi latar belakang dilakukan penelitian, rumusan masalah, batasan masalah, dan sistematika penulisan.

BAB II Kajian Pustaka dan Dasar Teori

Berisi tentang berbagai materi yang berhubungan dengan judul penulisan yaitu “RANCANG BANGUN MODUL *WEB SERVICES* UNTUK MANAJEMEN INFORMASI TUGAS AKHIR”.

BAB III Metode Penelitian

BAB II

KAJIAN PUSTAKAN DAN DASAR TEORI

2.1 Penelitian Terdahulu

Beberapa jurnal atau penelitian terdahulu yang dapat digunakan sebagai referensi, diantaranya sebagai berikut:

2.1.1 Penelitian Pertama

Penelitian pertama dengan judul “Implementasi Framework CodeIgniter dan Restful API pada Sistem Informasi Manajemen Tugas Akhir” tahun 2019 oleh Bayu Wijaya Putra, Ariansyah Saputra, Rudi Sanjaya. Penelitian tersebut membahas implementasi framework codeigniter dan restful Api pada sistem informasi manajemen tugas akhir. Menggunakan kerangka kerja CodeIgniter karena banyak fleksibilitas untuk pengembang, terutama menggabungkan dengan RESTful API. API ini telah diintegrasikan dengan Framework CodeIgniter. Hal ini dapat menjadi titik kuat untuk mengembangkan sistem terpadu. Dan juga framework CodeIgniter memiliki lusinan pustaka di GitHub untuk menangani semua kebutuhan pengembang web.

2.1.2 Penelitian Kedua

Penelitian kedua dengan judul “Analisis Perbandingan Metode Soap Dan Rest Yang Digunakan Pada Framework Flask Untuk Membangun Web Services” tahun 2019 oleh M Gilvy Langgawan Putra, M Ihsan Alfani Putera. Penelitian tersebut membahas teknologi *web services* yaitu SOAP (Simple Object Access Protocol) dan REST (REpresentational State Transfer) yang digunakan pada framework flask. Teknologi *web services* tersebut banyak digunakan dalam pengembangan aplikasi. Untuk *web services* yang berbasis Flask dengan menggunakan SOAP dan REST api, REST memiliki performance yang lebih bagus dibandingkan dengan SOAP untuk

pengatesan request dan respon untuk *web services* dengan menggunakan socket json dan xml lebih baik menggunakan socket XML.

Dengan demikian adanya persamaan penelitian terkait pengembangan *web servicess* manajemen informasi dalam skala universitas untuk menunjang berlangsung sistem informasi yang efisien dalam sebuah universitas, dan di harapkan output yang di hasilkan lebih baik dari penelitian terdahulu.

2.2 Sistem Informasi

Sistem informasi merupakan sistem buatan manusia yang terdiri dari banyak komponen yang terorganisasi dengan tujuan yang berfokus pada penyajian informasi (Suryantara, 2017).

Sementara menurut Pratama (2014), sistem informasi merupakan sistem yang disusun oleh empat elemen berikut :

1. Perangkat lunak (software).
2. Perangkat keras (hardware).
3. Infrastruktur.
4. Sumber daya manusia (SDM) yang terlatih.

Keempat elemen diatas bekerjasama dalam mengolah data menjadi informasi yang bermanfaat, termasuk bekerjasama dalam menentukan proses perencanaan, kontrol, koordinasi dan pengambilan keputusan, akibatnya sistem yang mengolah data menjadi informasi merupakan sistem yang kompleks.

2.3 Tugas Akhir

Tugas akhir merupakan bagian akhir dari perkuliahan yang lebih dikenal dengan istilah skripsi khususnya istilah yang dipakai di program studi sistem informasi yang merupakan satu keigatan akhir perkuliahan mahasiswa atau syarat kelulusan mahasiswa dalam mengambil sarjana.

Karena tugas akhir ini adalah penentu kelulusan dari mahasiswa dan nantinya akan ditanggungjawabkan di sidang ujian tugas akhir yang akan di review/uji oleh beberapa dosen yang memiliki kompetensi yang sesuai dengan tema tugas akhir yang akan diujikan.

2.4 Web Services

2.4.1 Definisi Web Services

Web service (Utama, 2010) adalah suatu sistem perangkat lunak yang dirancang untuk mendukung interoperabilitas dan interaksi antar sistem pada suatu jaringan. *Web services* digunakan sebagai suatu fasilitas yang disediakan oleh suatu *web site* untuk menyediakan layanan (dalam bentuk informasi) kepada sistem lain, sehingga sistem lain dapat berinteraksi dengan sistem tersebut melalui layanan-layanan (*service*) yang disediakan oleh suatu sistem yang menyediakan web service. *Web services* juga diartikan sebagai sebuah antar muka yang menggambarkan sekumpulan operasi-operasi yang dapat diakses melalui jaringan dalam bentuk XML (Kreger, 2001).

Web services memiliki empat lapisan dasar (Deviana, 2011), yaitu

Service Publication and Discovery (<u>UDDI</u>)
Service Description (<u>WSDL</u>)
XML Based Messaging (<u>SOAP</u>)
Common Internet Protocol (<u>HTTP</u> , <u>TCP/IP</u>)

Gambar 2.3. Lapisan Dasar Web Services

Lapisan dasar pada Gambar diatas merupakan blok bangunan *web services* yang mana menyediakan fasilitas komunikasi jarak jauh antara dua aplikasi yang merupakan layer arsitektur *web services*.

- a. Layer 1 : protokol internet standar yang digunakan sebagai sarana transportasi adalah HTTP dan TCP/IP.
- b. Layer 2 : Simple Object Access Protocol (SOAP) berbasiskan XML dan digunakan untuk pertukaran informasi antar sekelompok layanan.
- c. Layer 3 : Web service Definition Language (WSDL) digunakan untuk mendiskripsikan attribute layanan.
- d. Layer 4 : Universal Description Discovery and Integration, yang mana merupakan direktori pusat untuk deskripsi layanan.

2.4.2 Keuntungan *Web Services*

Menurut Scoot (2003), beberapa keuntungan menggunakan web service adalah:

1. Kerjasama yang baik di mana sebagai service jarak jauh sudah dapat digunakan oleh client dengan platform yang berbeda.
2. Model Hubungan antarmuka yang kuat dimaksudkan bahwa tipe data yang dikirim ke dan diterima dari Service jarak jauh tidak ambigu maksudnya model data yang didefinisikan oleh Service jarak jauh cocok dengan model data yang didefinisikan oleh mayoritas bahasa pemrograman yang telah ditetapkan.
3. Kemampuan standar Internet yang sudah ada dimaksudkan bahwa solusi yang sudah tersedia di dalam standar internet yang banyak dipakai dapat memakai tool set yang ada dan bisa menghasilkan produk dari teknologi tersebut.
4. Dukungan kepada semua infrastruktur komponen terdistribusi

2.5 Python

Python pertama kali dikembangkan oleh Guido van Rossum pada tahun 1990 di CWI, Belanda. Bahasa ini dikategorikan sebagai bahasa pemrograman tingkat tinggi (very-high-level language) dan juga merupakan object oriented dynamic language (bahasa berorientasi objek yang dinamis). Pengurangan source program secara besar-besaran juga merupakan tujuan dibuatnya bahasa ini.

Menurut Owo Sugiana (2003), Python bukan hanya “sekedar bahasa lain” untuk membuat aplikasi, tapi merupakan sebuah bahasa jenis baru. Secara umum Python menawarkan:

- Berorientasi objek
- Struktur pemrograman yang handal
- Arsitektur yang dapat dikembangkan (extensible) dan ditanam (embeddable) dalam bahasa lain
- Sintaks yang mudah dibaca

Python juga didukung oleh banyak package-package yang membantu dalam proses membuat sebuah aplikasi, sebagai contoh dalam pemrograman web terdapat framework Flask dan Django yang sangat membantu pengembangan pemrograman website.

Python juga merupakan multi platform yang dapat di gunakan di beberapa sistem operasi, yaitu : Windows, Mac OS X, dan Linux (Advernesia, 2019).

2.6 Flask

Flask merupakan salah satu framework python. Framework adalah kerangka aplikasi yang berisi tentang fungsi-fungsi, plugin dan konsep untuk membangun sebuah aplikasi. Dengan menggunakan framework maka aplikasi yang dibangun dapat tersusun dan dengan rapi (Ronacher, 2019).

Flask merupakan framework web yang dibangun dengan Bahasa pemograman python oleh Armin Ronacher. Flask didasarkan toolkit WSGI Werkzeug dan Template Jinja2.

2.7 WSGI

Web Server Gateway Interface (WSGI) merupakan standart baru dalam pengembangan aplikasi web dengan python. WSGI adalah spesifikasi untuk antarmuka universal antara server web dan aplikasi web.

2.8 Werkzeug

Merupakan toolkit WSGI, yang mengimplementasikan permintaan, objek respon, dan fungsi utilitas lainnya. Kerangka Flask menggunakan Werkzeug sebagai salah satu basisnya.

2.9 Jinja2

Adalah system templating yang populer di Python. Sebuah system templating web yang menggabungkan antara aplikasi web dengan sumber data untuk membangun aplikasi web yang dinamis.

2.10 Representational State Transfer (REST)

REST merupakan filosofi desain yang mendorong kita untuk menggunakan protokol dan fitur yang sudah ada pada web untuk memetakan permintaan terhadap sumber daya pada berbagai macam representasi dan manipulasi data di Internet (Scribner dan Seely, 2009).

REST adalah gaya arsitektural yang memiliki aturan seperti antarmuka yang seragam, sehingga jika aturan tersebut diterapkan pada web services akan dapat memaksimalkan kinerja web services terutama pada performa, skalabilitas, dan kemudahan untuk dimodifikasi. Pada arsitektur

REST, data dan fungsi dianggap sebagai sumber daya yang dapat diakses lewat Uniform Resource Identifier (URI), biasanya berupa tautan pada web.

REST menggunakan protokol HTTP yang bersifat stateless. Perintah HTTP yang bisa digunakan adalah fungsi GET, POST, PUT atau DELETE. Hasil yang dikirimkan dari server biasanya dalam bentuk format XML atau JSON sederhana tanpa ada protokol pemaketan data, sehingga informasi yang diterima lebih mudah dibaca dan diparsing pada sisi client. Dalam penerapannya, REST lebih banyak digunakan untuk web service yang berorientasi pada sumber daya. Maksud orientasi pada sumber daya adalah orientasi yang menyediakan sumber daya sebagai layanannya dan bukan kumpulan-kumpulan dari aktifitas yang mengolah sumber daya itu. Bentuk web service menggunakan REST style sangat cocok digunakan sebagai back-end dari aplikasi berbasis mobile karena cara aksesnya yang mudah dan hasil data yang dikirimkan berformat JSON sehingga ukuran file menjadi lebih kecil.

Menurut Fielding (2000), REST adalah arsitektur standar web yang menggunakan protokol HTTP dalam komunikasi data. Arsitektur tersebut didirikan berdasarkan sumber data dimana masing-masing komponen merupakan sumber data. Sumber data diakses oleh antarmuka yang sama dengan menggunakan metode standar HTTP. Dalam arsitektur REST, server yang mengikuti arsitektur REST menyediakan akses ke sumber data dan klien yang mengambil data. Setiap sumber data diidentifikasi menggunakan link URI. REST menggunakan berbagai format untuk menyajikan data, seperti teks, JSON dan XML. Berikut adalah metode HTTP yang umumnya digunakan dalam arsitektur REST :

1. GET untuk menyediakan akses untuk membaca sumber data.
2. PUT untuk memperbarui data yang tersedia.
3. DELETE untuk menghapus data.
4. POST untuk membuat data baru.

Fielding juga mengemukakan lima sifat dasar REST antara lain :

1. Uniform Interface yang mengharuskan pemetaan resource dilakukan secara terstruktur dan memungkinkan setiap bagian dari resource sistem (hasil pemetaan) dapat berkembang secara swadaya.
2. Stateless yang memastikan setiap satu request akan menerima satu response. Diluar request tidak terjadi interaksi apapun antara client dan server, dan setelah komunikasi antara client dan server tidak tersisa nilai apapun yang dipertahankan pada memory.
3. Cacheable yang memungkinkan server untuk memberikan response yang sama berdasarkan request dari client tanpa melalui back-end pada pola request tertentu.
4. Client-Server yang memisahkan antara proses pada back-end dengan interface sistem dimana antara client (interface) dan server (back-end system) berfokus pada pengembangannya masing-masing (logic mampu berkembang secara terpisah).
5. 5. Layered System yang melarang request dari client secara langsung dihubungkan kepada proses back-end. Hal ini ditujukan untuk meningkatkan skalabilitas dan keamanan sistem.

BAB III

METODE PENELITIAN

3.1 Waktu dan Peralatan

Waktu yang digunakan untuk melakukan penelitian adalah diluar jam kerja penulis dan waktu senggang saat di rumah atau dikampus. Bahan yang digunakan dalam melakukan penelitian ini adalah buku tentang pembuatan web dengan html, css, javascript. Sedangkan alat yang digunakan peneliti untuk membantu dalam melakukan penelitian adalah sebagai berikut:

- a. Hardware
 - 1. Laptop Asus
 - 2. Harddisk 500 GB
 - 3. Memory RAM 4 GB
- b. Software
 - 1. Sistem operasi linux ubuntu
 - 2. Editor Atom
 - 3. LibreOffice Writer
 - 4. Back end Python
 - 5. Framework Flask

3.2 Pengumpulan Data

Data didapatkan dari studi literatur dari berbagai buku dan jurnal skripsi terdahulu yang dapat dijadikan referensi berhubungan dengan penelitian

3.3 Metode Penelitian

Metode yang digunakan dalam penelitian ini adalah metode deskriptif, yakni metode yang menggambarkan secara sistematis, faktual dan akurat mengenai informasi yang sebenarnya sesuai fakta-fakta yang ditemui di lapangan.

3.4 Arsitektur Sistem

Arsitektur tersebut terdiri atas lapisan Business Logic dan data Access berfungsi untuk memproses setiap komunikasi (POST, GET, PUT dan DELETE). Pada penerapannya, menerjemahkan setiap request HTTP dari client. Sedangkan mengimplementasikan fungsionalitas inti dari sistem, dan merangkum logika bisnis yang relevan. Lapisan data access berperan untuk mengekspos data berdasarkan batasan batasan yang dimiliki oleh sistem. Data atau informasi yang diekspos menggunakan format data JSON. Jason adalah bahasa independen dan berdasarkan koleksi pasangan key/value dan mempunyai list value. Struktur ini memungkinkan untuk digunakan dalam setiap bahasa pemrograman modern yang membuatnya menjadi pilihan yang baik untuk berkomunikasi di dunia web. 18

Sistem yang dikembangkan memiliki beberapa bagian yaitu RESTful API, aplikasi berbasis web, dan basis data. Setiap bagiannya memiliki fungsi yang berbeda. Aplikasi berbasis web digunakan oleh developer untuk mengelola aplikasi pengguna layanan yaitu sistem informasi tugas akhir mahasiswa, dan melihat dokumentasi penggunaan RESTful API. RESTful API digunakan oleh aplikasi pengguna layanan untuk mengakses informasi data mahasiswa. Basis data digunakan sebagai tempat menyimpan data.

3.5 Install Flask, Virtual Env dan Package

Untuk system operasi Linux secara default sudah ada python. Untuk memastikannya bisa di lakukan dengan memasukkan perintah python – version, maka akan muncul python versi berapa yang menandakan system operasi tersebut sudah terinstall python.

Instalasi dilakukan lewat terminal, menggunakan perintah:

1. sudo apt-get update
2. sudo apt-get install python3.6

3. `sudo apt-get install python3-pip`

4. `sudo apt-get install python3-venv`

Pada langkah pertama, perintah tersebut digunakan untuk mengunduh daftar paket dari repositori dan "memperbaruinya" untuk mendapatkan informasi tentang versi terbaru paket.

Pada langkah ke dua, instalasi defaultnya adalah Python 2.7. Oleh karena itu kita gunakan python versi 3.

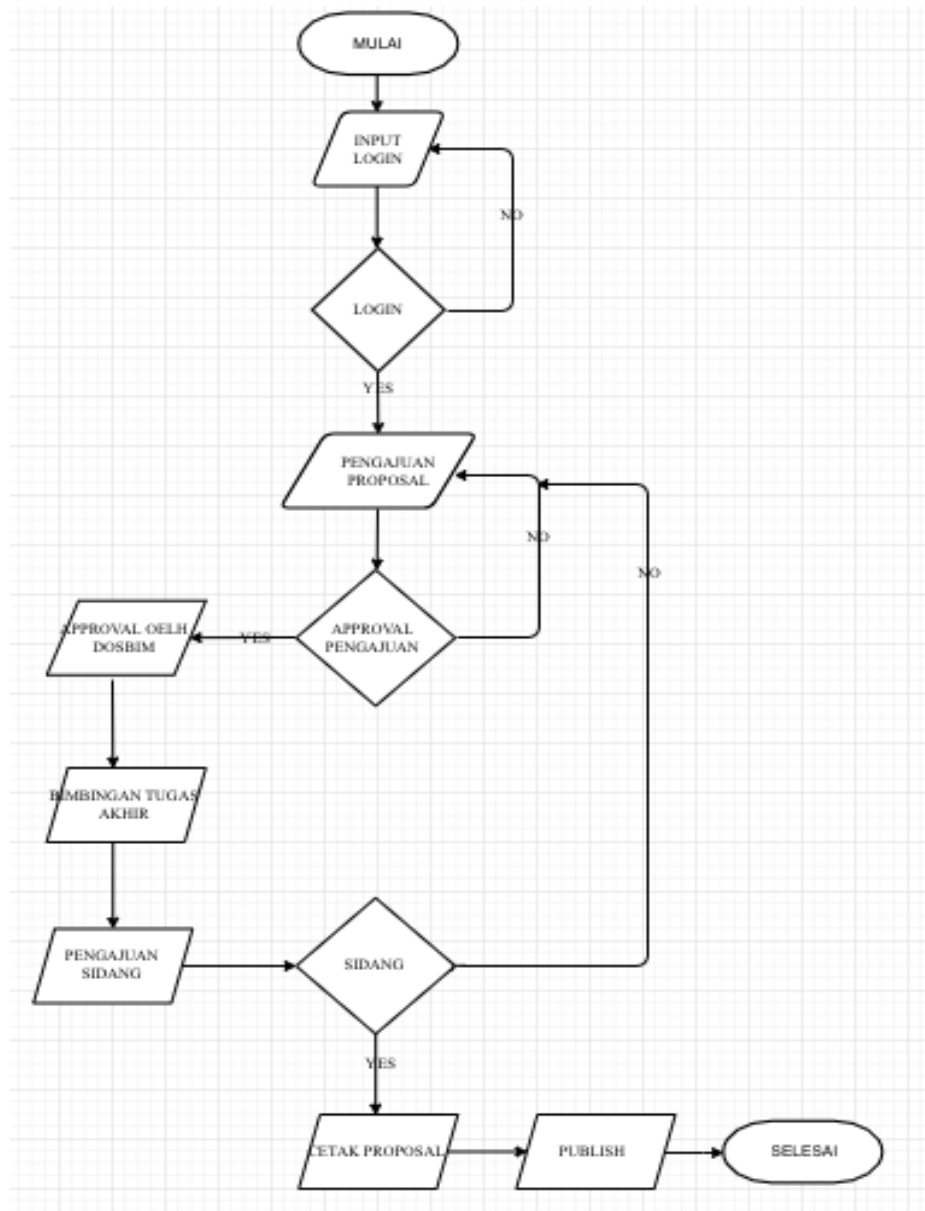
Pada langkah ke tiga, instalasi Library Python, semacam NPM di NodeJS.

Pada langkah ke empat, Virtualenv adalah perintah untuk membuat sebuah virtual envorenment. Venv adalah nama virtual envorenment yang kita buat

3.6 Flow Chart

Dengan menggunakan flow chart maka rancangan sistem dan urutan kegiatan akan lebih jelas dan ini merupakan langkah awal dalam pembuatan sistem.

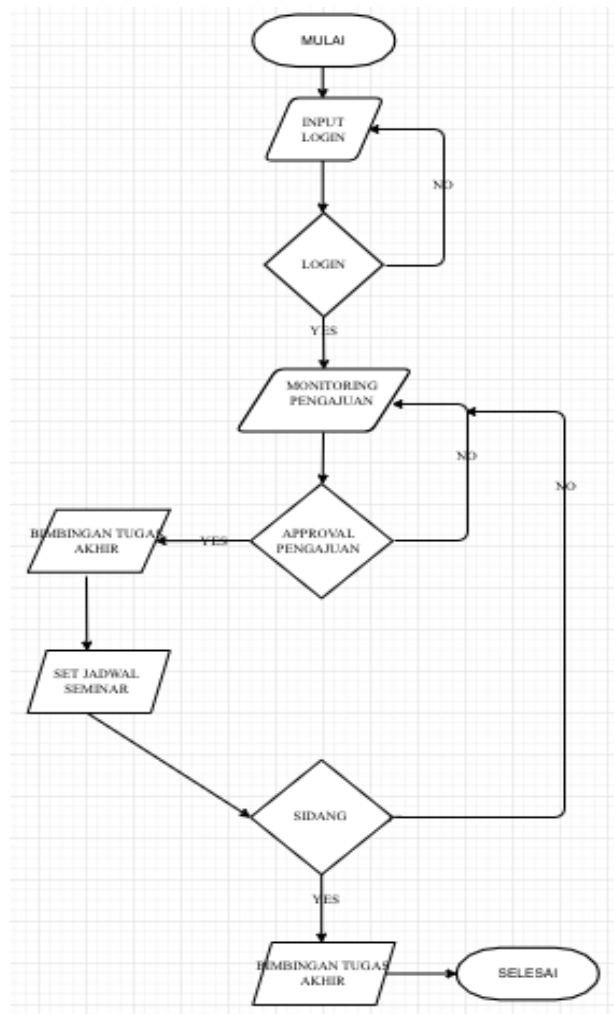
3.6.1 Flow Chart Mahasiswa



Gambar 3.1 Flow Chart Mahasiswa

Pada flow chart mahasiswa ini menggambarkan bagaimana proses seorang mahasiswa mulai dari mengakses sebuah perangkat lunak manajemen informasi tugas akhir mulai dari mahasiswa melakukan login, kemudian membuat pengajuan tugas akhir dan memilih dosen pembimbing sampai dengan proses terakhir.

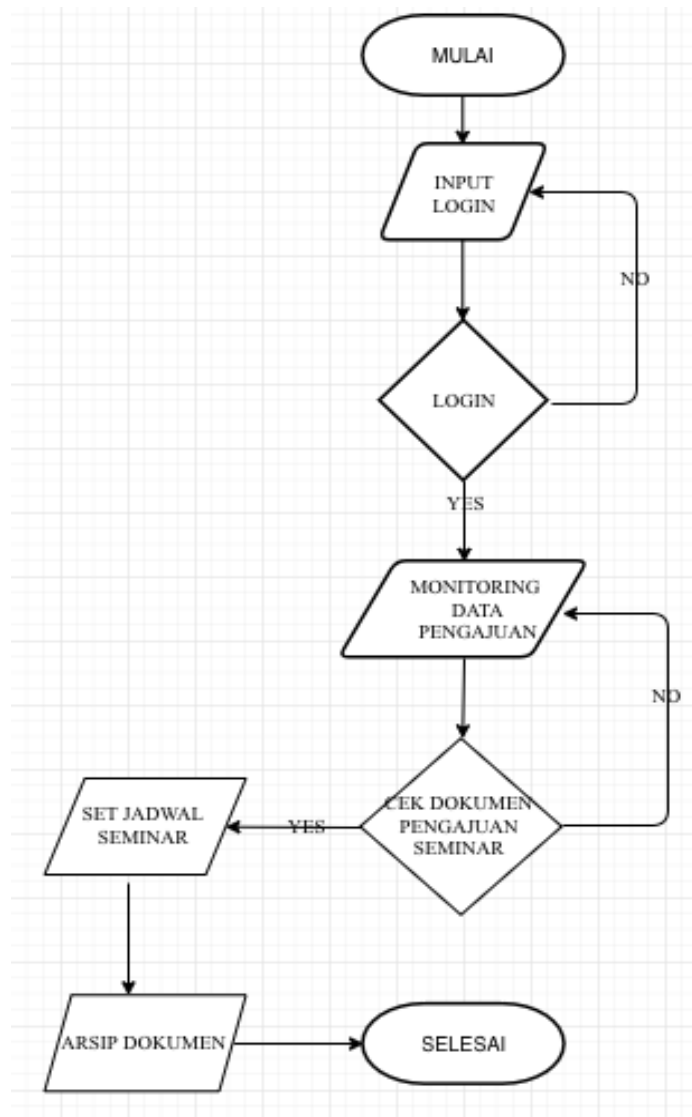
3.6.2 Flow Chart Dosen Pembimbing



Gambar 3.2 Flow Chart Dosen Pembimbing

Dalam flow chart ini, menggambarkan bagaimana seorang dosen pembimbing mulai melakukan login, melihat data yang pengajuan hingga memberikan pertanyaan kepada mahasiswa yang memilih beliau sebagai dosen pembimbing. Dan calon dosen pembimbing juga memiliki hak untuk menerima atau menolak pengajuan.

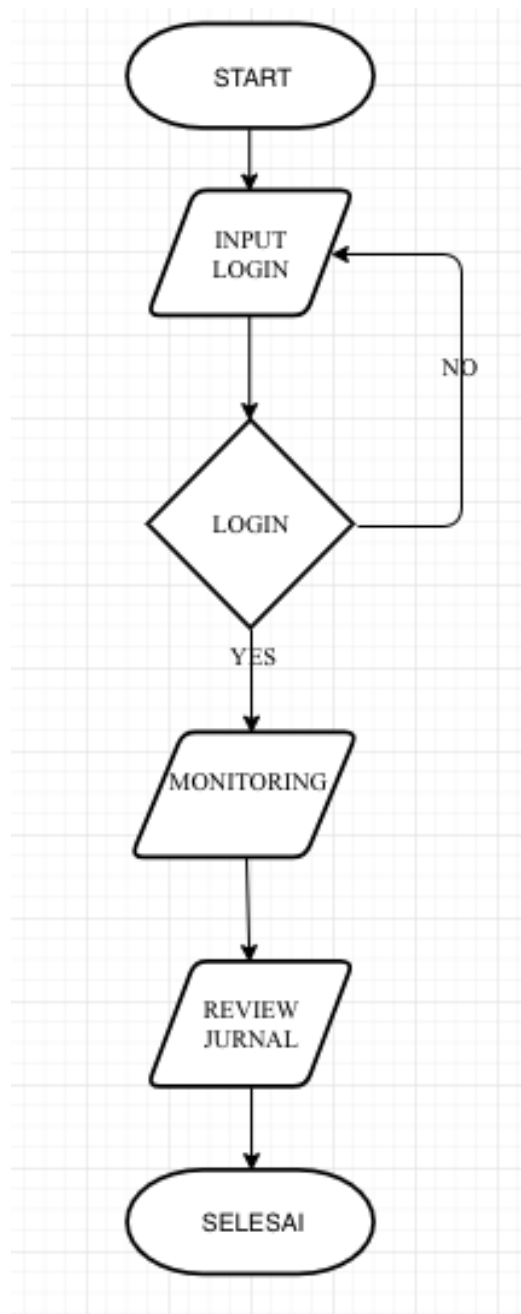
3.6.3 Flow Chart Admin



Gambar 3.3 Flow Chart Admin

Untuk flow chart admin, admin dapat melihat data pengajuan sidang tugas akhir yang di ajukan oleh mahasiswa melalui perangkat lunak manajemen informasi tugas akhir dan memeriksa berkas atau dokumen yang terkait pengajuan sidang tugas akhir.

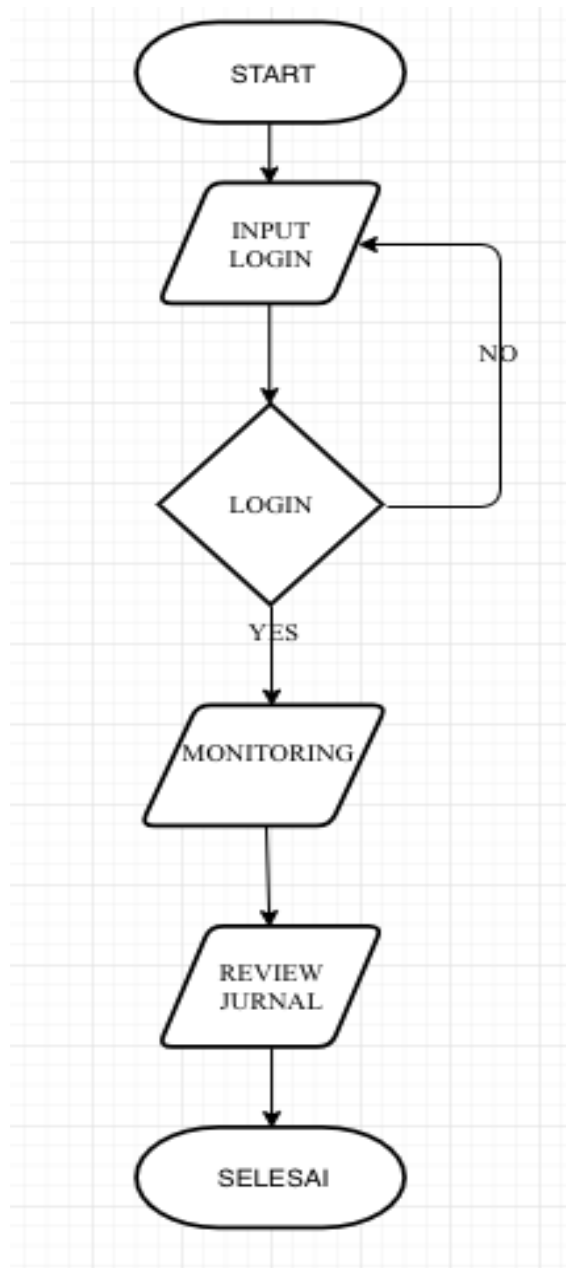
3.6.4 Flow Chart Kaprodi



Gambar 3.4 Flow Chart Kaprodi

Untuk flowchart kaprodi, dapat melihat prosentase proses bimbingan tugas akhir melalui data monitoring tugas akhir.

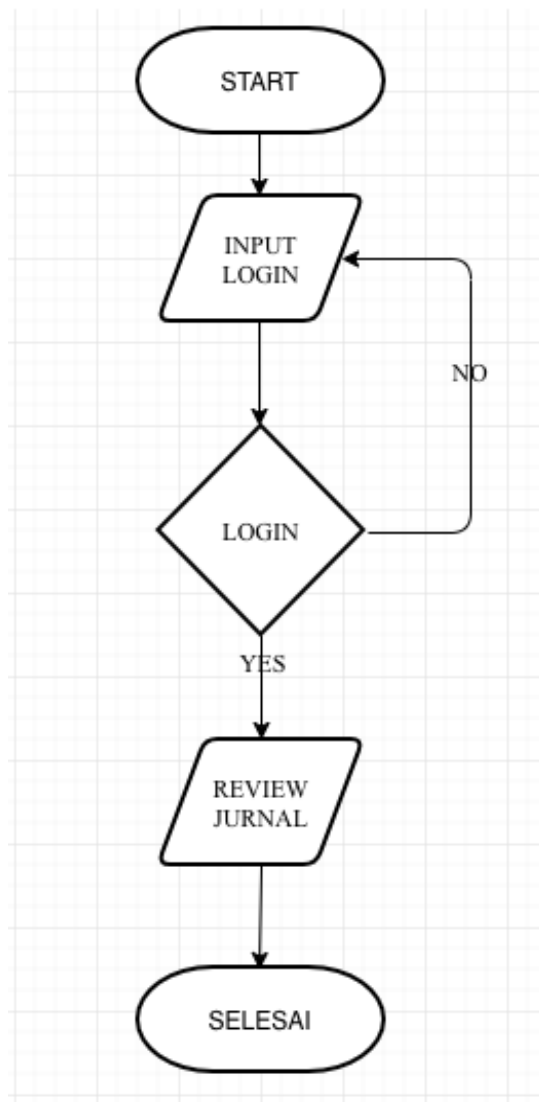
3.6.5 Flow Chart Dosen Rektor



Gambar 3.5 Flow Chart Rektor

Sama halnya dengan flowchart kaprodi, flow chart dapat melihat prosentase proses bimbingan tugas akhir melalui data monitoring tugas akhir.

3.6.6 Flow Chart Public



Gambar 3.6 Flow Chart Public

Dalam flow chart *public*, khalayak umum di berikan akses untuk melihat atau me-review hasil dari penelitian yang telah dilakukan.

Ketika sistem di akses maka akan menuju ke halaman awal aplikasi tersebut, sebelum user menggunakan aplikasi, maka user di wajibkan melakukan login terlebih dahulu dan ketika login inilah yang menentukan siapa yang

login. Artinya ada perbedaan class antar user atau ada sistem pembeda yang dapat mengetahui siapa melakukan login. Sistem akan mengetahui Mahasiswa atau Dosen Pembimbing atau Admin atau Kaprodi yang melakukan login.

3.7 DFD

Merupakan sebuah alat yang biasanya digunakan untuk memaparkan alur suatu program berupa gambar yang saling berhubungan dengan menggunakan alur data.

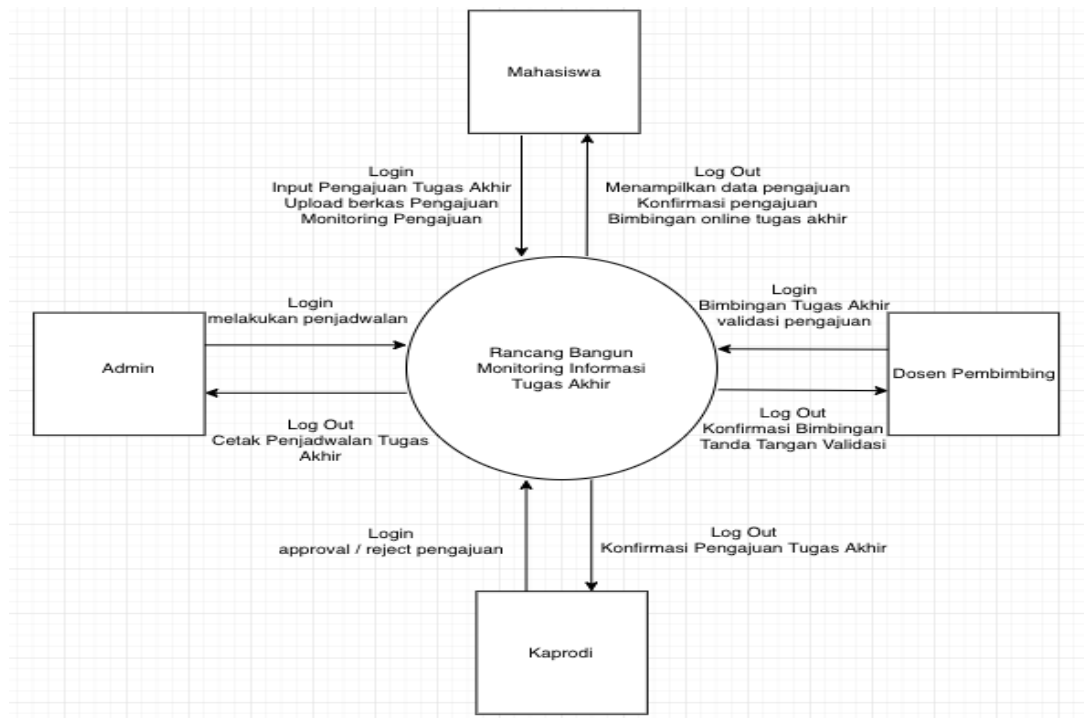
Ada beberapa komponen yang ada pada DFD (Data Flow Diagram) :

1. Komponen Terminator / Entitas Luar
2. Komponen Proses
3. Komponen Data Store
4. Komponen Alur Data

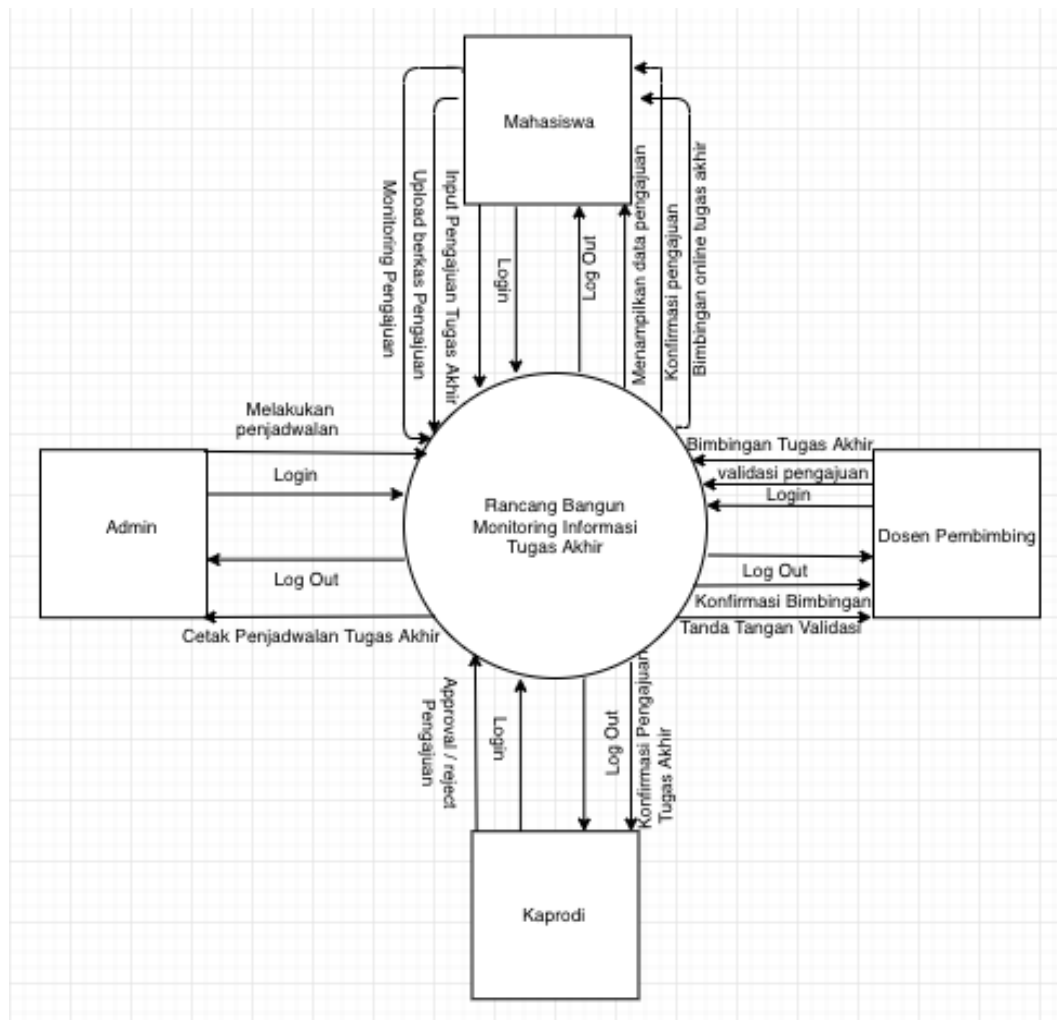
Ada beberapa syarat – syarat yang harus ada dalam membuat Data Flow Diagram, sebagai berikut :

1. Setiap komponen pada DFD harus di berikan nama.
2. Pada komponen proses harus di sertakan nomor.
3. Sesering mungkin untuk menggambarkan DFD agar mudah di pahami.
4. Tidak di perkenan menggambar DFD yang rumit / sulit di mengerti.
5. Bentuk DFD harus dapat di logikan dengan mudah.

Berikut adalah bentuk DFD Level 0 dari Rancang bangun monitoring informasi tugas akhir.



Gambar 3.7 DFD Level 0



Gambar 3.8 DFD Level 1

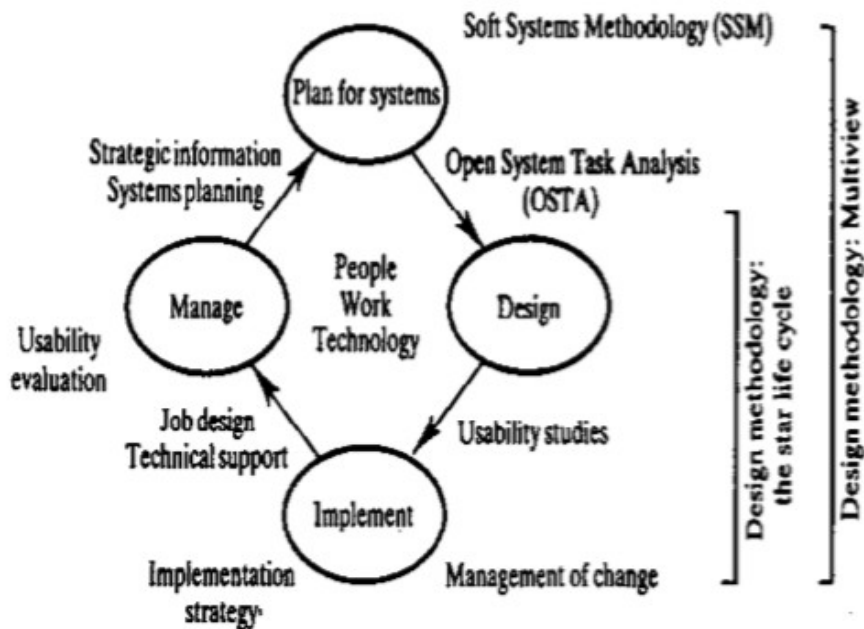
3.8 Metode Pengembangan

Metodelogi yang digunakan dalam pengembangan “ Rancang Bangun Modul Web Service Untuk Manajemen Informasi Tugas Akhir ” adalah *User Centered Design* (Widhiarso, 2019).

User Centered Design merupakan filosofi pengembangan sistem yang memusatkan pengguna sebagai dasar pengembangan. Artinya, pengembangan sistem sesuai dengan kebutuhan pengguna sebuah aplikasi. Metode tersebut di anggap sangat efektif dalam proses pengembangan sebuah sistem di bandingkan menggunakan metode *End User*.

Adapun prinsip – prinsip yang dalam UCD adalah :

1. Fokus pada pengguna
2. Perancangan Terintegrasi
3. Perancangan Interaktif



Gambar 3.9 Metode UCD

Untuk mendukung pengembangan menggunakan metode *User Centered Design*, dapat menggunakan tool yaitu *User Centered Design Canvas (UCDC)* (prelicz-zawadzka, user centered design canvas, 2019).

User Centered Design Canvas (UCDC) adalah sebuah tool yang digunakan untuk berbagai keperluan professional dalam pengembangan sistem yang menjadikan pengguna sebagai pusat dari sebuah proses pengembangan sistem.

Manfaat dari menggunakan *User Centered Design Canvas* adalah lebih memudahkan penjabaran atau pemetaan dari latar belakang dari sebuah

masalah yang di angkat untuk pengembangan sebuah aplikasi yang menggunakan metode UCD.

UCDC sendiri mengadopsi Business Model Canvas (BMC) yang dimana terdapat 9 bagian yang perlu ditanyakan oleh pengusaha pemula dalam memulai bisnisnya. Untuk itu, dalam UCD terdapat 9 bagian yang perlu di definisikan dalam rangka pengembangan fitur perangkat lunak yang akan dikembangkan. Diantaranya yaitu:

1. Business

Pada bagian ini, pengembang perlu mendefinisikan bentuk produk akhir/brand/nama aplikasi yang nantinya akan dikembangkan.

2. Users

Bagian ini diisi potensi pengguna dari aplikasi yang akan dikembangkan.

3. Problems

Pengembang perlu menuliskan permasalahan eksisting pada transaksi manual ataupun pada perangkat lunak yang telah digunakan. Dimana temuan dari masalah-masalah ini merupakan fitur yang akan dikembangkan menjadi perangkat lunak yang baru.

4. Motives

Bagian ini diisi dengan analisa pengembang mengenai urgensi penggunaan aplikasi yang akan dikembangkan.

Beberapa hambatan dari pemanfaatan aplikasi yang akan dikembangkan perlu dituliskan pada bagian ini.

1. Solutions

Bagian ini berisi solusi dari permasalahan yang dituliskan pada point ke 3.

2. Alternatives

Bagian ini berisi dari solusi alternative dari point 6.

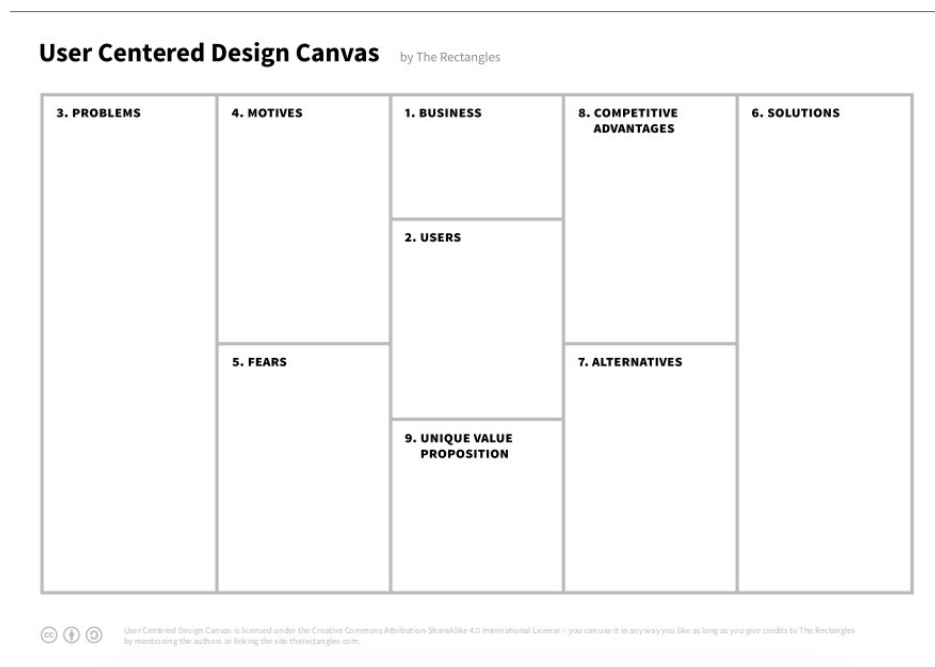
3. Competitive Advantages

Bagian ini berisi fitur-fitur yang ditawarkan dimana fitur tersebut tidak ada pada sistem eksisting.

4. Unique Value Proposition

Bagian diisi dengan penyampaian value (nilai) yang ditawarkan pada perangkat lunak yang baru atau yang akan dikembangkan.

Dalam implementasinya, pemanfaatan UCDC dapat menggunakan template seperti yang ditunjukkan pada gambar 3.11. Template ini diisi setelah melakukan proses analisa dan interview calon pengguna perangkat lunak yang akan dikembangkan.



Gambar 3.10 User Centered Design Canvas

DAFTAR PUSTAKA

- M. A. Rahman, I. Kuswardayan, and R. R. Hariadi, "Perancangan dan Implementasi *RESTful Web Services* untuk *Game* Sosial Food Merchant Saga pada perangkat Android," Tek. Inform. ITS, vol. 1, no. 2, pp. 2–5, 2013.
- R. Ramanathan and T. Korte, "Software service architecture to access weather data using RESTful web services," Fifth Int. Conf. Comput. Commun. Netw. Technol., pp. 1–8, 2014.
- Suryantara, I Gusti Ngurah. 2017. Merancang Aplikasi dengan Metodologi Extreme Programmings. Jakarta: Elex Media Komputindo.
- Pratama, I Putu Agus Eka. 2014. Handbook Jaringan Komputer. Bandung: Informatika Bandung.
- Deviana, H., 2011, Penerapan XML Web Service pada Sistem Distribusi Barang, Jurnal Generic, no. 2, vol. 6, hal. 55-62.
- Owo Sugiana. (2003). Membuat aplikasi bisnis menggunakan bahasa Python dan database berbasis SQL. Jakarta
- Short, Scott. 2003. Building XML Web Services For The Microsoft .Net Platform. Jakarta : PT Elex Media Komputindo.
- Ronacher, A. (2019, April 08). *Flask Web Development on drop at a time*. Retrieved from flask.pocoo: <http://flask.pocoo.org>
- Scribner, Kenn dan Seely Scott. 2009. Effective REST Services via .NET: For .NET Framework 3.5 (1st Edition). Addison-Wesley Professional.
- Fielding, Roy Thomas. 2000. Architectural Styles and the Design of Network based Software Architectures. University Of California, Irvine.