

APLIKASI MANAJEMEN PROYEK POLITIK POS INDOENSIA

APLIKASI MANAJEMEN PROYEK POLITEKNIK POS INDOENSIA

Luthfi M. Nabil, Hagan Rowlenstino A. S
D4 Teknik Informatika



Kreatif Industri Nusantara

Penulis:

Rolly Maulana Awangga

ISBN : 978-602-53897-0-2

Editor:

M. Yusril Helmi Setyawan

Penyunting:

Syafrial Fachrie Pane

Khaera Tunnisa

Diana Asri Wijayanti

Desain sampul dan Tata letak:

Deza Martha Akbar

Penerbit:

Kreatif Industri Nusantara

Redaksi:

Jl. Ligar Nyawang No. 2

Bandung 40191

Tel. 022 2045-8529

Email : awangga@kreatif.co.id

Distributor:

Informatics Research Center

Jl. Sariasih No. 54

Bandung 40151

Email : irc@poltekpos.ac.id

Cetakan Pertama, 2019

Hak cipta dilindungi undang-undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara
apapun tanpa ijin tertulis dari penerbit

*‘Jika Kamu tidak dapat
menahan lelahnya
belajar, Maka kamu harus
sanggup menahan
perihnya Kebodohan.’
Imam Syafi’i*

CONTRIBUTORS

ROLLY MAULANA AWANGGA, Informatics Research Center., Politeknik Pos Indonesia, Bandung, Indonesia

CONTENTS IN BRIEF

1 Bab 1 Pegantar	1
2 Analisis Dan Perancangan	13
3 Cara Instalasi Tools	51

DAFTAR ISI

Daftar Gambar	xiii
Daftar Tabel	xvii
Foreword	xxi
Kata Pengantar	xxiii
Acknowledgments	xxv
Acronyms	xxvii
Glossary	xxix
List of Symbols	xxxi
Introduction	xxxiii
<i>Luthfi M. Nabil, Hagan Rowlenstino A.S</i>	

1	Bab 1 Pegantar	1
1.1	Manajemen	1
1.2	Transparansi	2
1.3	API (Aplication Programming Interface)	3
1.4	Webservice	4

1.5	PHP (Hypertext Preprocessor)	7
1.6	Code Igniter	8
1.7	Node.js	10
1.8	Unified Modeling Language (UML)	11
1.9	Java Script	11
2	Analisis Dan Perancangan	13
2.1	Analisis Sistem	13
2.1.1	Analisis Yang Sedang Berjalan Pada Proses Kegiatan Proyek	14
2.1.2	Analisis Sistem yang Akan Dibangun	16
2.1.3	Analisis Kebutuhan Fungsional	18
2.1.4	Analisis Kebutuhan Non Fungsional	19
2.2	Perancangan Sistem	21
2.2.1	Use Case Diagram	21
2.3	User Interface	38
2.3.1	Login	38
2.3.2	Kelola Data Kegiatan	39
2.3.3	Kelola Data Dosen	40
2.3.4	Kelola Data Mahasiswa	41
2.3.5	Pengajuan Proposal	42
2.3.6	Pengajuan Bimbingan	43
2.3.7	List Bimbingan	44
2.3.8	Penilaian pembimbing	45
2.3.9	Penilaian Penguji	46
2.3.10	Penyetujuan Proposal	47
2.3.11	Pengaturan Proyek	48
2.3.12	Penjadwalan	49
3	Cara Instalasi Tools	51
3.1	Code Igniter	51
3.1.1	Cara Instalasi Code Ignitor	51
3.2	XAMPP	53
3.2.1	Kelebihan XAMPP	53
3.2.2	Kekurangan XAMPP	54
3.2.3	Cara Instalasi XAMPP di Windows	54
3.3	Node.js	58
3.3.1	Cara Instalasi Node.js di windows	58

3.4	VS Code	60
3.4.1	Keuntungan memakai VS Code	60
3.4.2	Kerugian menggunakan VS Code	63
3.4.3	Cara installasi VS Code di Windows	63
3.5	Teori	66
3.5.1	Fungsi	66
3.6	Package	66
3.7	Class, Object, Atribute, and Method	67
3.8	Pemanggilan Class	68
3.9	Pemakaian Package Fungsi Apabila File Didalam Folder	68
3.10	Pemakaian Package Kelas Apabila File didalam Folder	68
Daftar Pustaka		69

DAFTAR GAMBAR

2.1	Flowmap Pengajuan Proposal Sistem yang Sedang Berjalan	14
2.2	Flowmap Pengajuan Bimbingan Sistem yang Sedang Berjalan	15
2.3	Flowmap Pengajuan Proposal yang Akan Dibangun	16
2.4	Flowmap Kelola Kegiatan yang Akan Dibangun	17
2.5	Flowmap Bimbingan yang Akan Dibangun	18
2.6	Kebutuhan Perangkat Keras Server	20
2.7	Kebutuhan Perangkat Lunak Server	21
2.8	Kebutuhan Perangkat Lunak Client	21
2.9	Use Case	22
2.10	Definisi Aktor	23
2.11	Definisi Use Case	24
2.12	Definisi Use Case	25
2.13	Definisi Use Case	26

2.14	Skenario Use Case Login	27
2.15	Skenario Kelola Data Kegiatan	28
2.16	Skenario Kelola Data Mahasiswa	29
2.17	Skenario Kelola Data Dosen	30
2.18	Skenario Pengajuan Proposal	31
2.19	Skenario Pengajuan Bimbingan	32
2.20	Skenario Penyetujuan Bimbingan dan Sidang	33
2.21	Skenario Kelola Penilaian	34
2.22	Skenario Penyetujuan Nilai	35
2.23	Skenario Penjadwalan Sidang	36
2.24	Skenario Penyetujuan Proposal	37
2.25	Skenario Pengaturan Proyek	38
2.26	user Interface Login	39
2.27	user Interface Kelola Data Kegiatan	40
2.28	user Interface Kelola Data Dosen	41
2.29	user Interface Kelola Data Mahasiswa	42
2.30	user Interface Pengajuan Proposal	43
2.31	user Interface Pengajuan Bimbingan	44
2.32	user Interface List Bimbingan	45
2.33	user Interface Penilaian Pembimbing	46
2.34	user Interface Penilaian Penguji	47
2.35	user Interface Penyetujuan Proposal	48
2.36	user Interface Pengaturan Proyek	49
2.37	user Interface Penjadwalan	50
3.1	Web Code Ignitor	52
3.2	Hasil	53
3.3	XAMPP	54
3.4	XAMPP	55

3.5	XAMPP	55
3.6	XAMPP	56
3.7	XAMPP	57
3.8	XAMPP	57
3.9	XAMPP	58
3.10	Node.js	59
3.11	Node.js	59
3.12	Node.js	60
3.13	VS Code	61
3.14	VS Code	61
3.15	VS Code	62
3.16	VS Code	62
3.17	VS Code	63
3.18	VS Code	64
3.19	VS Code	64
3.20	VS Code	65
3.21	VS Code	66

DAFTAR TABEL

Listings

FOREWORD

Sepatah kata dari Kaprodi, Kabag Kemahasiswaan dan Mahasiswa

KATA PENGANTAR

Puji syukur kami senantiasa ucapkan atas kehadiran Allah SWT karena curahan rahmat serta karunia-nya lah kami akhirnya sampai pada tahap menyelesaikan laporan dengan judul Aplikasi Manajemen Proyek Politeknik Pos Indonesia. Ucapan terimakasih kami senantiasa ucapkan kepada orang tua kami yang telah memberikan motivasi serta nasihat kepada kami, karena berkat doa serta dukungannya kami dapat terus melanjutkan pendidikan di Politeknik Pos Indonesia. Kami juga ucapkan rasa terimakasih yang sebanyak-banyaknya untuk M. Harry K Saputra, S.T.,M.T.I. selaku dosen pembimbing yang telah membimbing kami untuk dapat menyelesaikan laporan project 3 dengan tepat waktu. Kami sungguh-sungguh berharap laporan ini bisa bermanfaat kepada masyarakat dengan tujuan untuk meningkatkan pengetahuan sekaligus wawasan terkait pengajuan proposal karya ilmiah, sekaligus mengetahui permasalahan dan cara untuk mengatasinya. Kami sadar bahwa laporan ini masih perlu diperbaiki kembali. Dengan demikian, kami benar-benar menantinya adanya kritik dan saran untuk perbaikan laporan yang hendak kami tulis dimasa selanjutnya, menyadari tidak ada suatu hal yang sempurna tanpa disertai saran yang konstruktif. Kami berharap laporan sederhana ini bisa dimengerti oleh setiap pihak terutama untuk para pembaca. Kami mohon maaf yang sebesar-besarnya jika ada perkataan yang kurang berkenan di hati.

L. M. NABIL, H. R. A. STEVANUS

Bandung, Jawa Barat

Februari, 2020

ACKNOWLEDGMENTS

Terima kasih atas semua masukan dari para mahasiswa agar bisa membuat buku ini lebih baik dan lebih mudah dimengerti.

Terima kasih ini juga ditujukan khusus untuk team IRC yang telah fokus untuk belajar dan memahami bagaimana buku ini mendampingi proses Intership.

R. M. A.

ACRONYMS

ACGIH	American Conference of Governmental Industrial Hygienists
AEC	Atomic Energy Commission
OSHA	Occupational Health and Safety Commission
SAMA	Scientific Apparatus Makers Association

GLOSSARY

git	Merupakan manajemen sumber kode yang dibuat oleh linus torvald.
bash	Merupakan bahasa sistem operasi berbasiskan *NIX.
linux	Sistem operasi berbasis sumber kode terbuka yang dibuat oleh Linus Torvald

SYMBOLS

- A Amplitude
- $\&$ Propositional logic symbol
- a Filter Coefficient

- \mathcal{B} Number of Beats

INTRODUCTION

LUTHFI M. NABIL, HAGAN ROWLENSTINO A.S

D4 Teknik Informatika

Bandung, Jawa Barat, Indonesia

Politeknik Pos Indonesia merupakan sebuah institusi pendidikan tinggi yang didirikan oleh Yayasan Pendidikan Bhakti Pos Indoensia (YPBPI) pada tahun 2001. Pada kurikulumnya Politeknik Pos Indonesia mengadakan kegiatan proyek pada beberapa tingkatan perkuliahannya sebagai latihan untuk menghadapi proyek pada dunia kerja nanti serta menjadi output karya mahasiswa.

Proyek adalah sebuah usaha sementara yang memiliki sasaran jelas dengan waktu pelaksanaan yang tegas. Pada umumnya, proyek sering dilakukan oleh pihak organisasi yang sudah berkembang dan dapat mengelola proyek dengan baik. Namun saat ini, pembuatan proyek sudah memiliki banyak metode dan digunakan untuk merancang beragam solusi yang dapat memenuhi kebutuhan masyarakat[1]. Tidak hanya organisasi, proyek juga banyak dilakukan oleh kalangan menengah seperti pengusaha kecil atau mahasiswa sekalipun[2].

Politeknik Pos Indonesia sudah menerapkan proyek sebagai kriteria penilaian. Hal ini diterapkan untuk memenuhi harapan institusi agar mahasiswa memiliki pengalaman dalam mengelola suatu proyek. Di sisi lain, dokumentasi dari setiap proyek yang dibuat oleh mahasiswa dapat menjadi penilaian dalam meningkatkan akreditasi. Namun, kegiatan proyek masih banyak membutuhkan manusia sebagai sarana penyampaian informasi kepada mahasiswa dan dosen yang mengikuti kegiatan.

Hal hal tersebut dapat menyebabkan adanya kemungkinan keterlambatan ataupun ketidaksampaian informasi kepada mahasiswa dan dosen yang mengikuti kegiatan. Tidak hanya itu, kesalahan penyampaian informasi juga dapat terjadi. Dan yang paling penting kita akan sulit untuk menyimpan, mengelompokkan, dan mencari kembali sebuah data proyek jika diperlukan. Walaupun dalam pengumpulan proposal sudah menggunakan google form, tetapi media tersebut tidak efektif karena kurangnya transparansi data dan mempersulit untuk pengelolaan data.

Untuk menghindari masalah tersebut, diperlukan sebuah sistem yang dapat melakukan pengelolaan data proyek yang dimasukkan oleh pengguna. Di saat yang bersamaan, penyampaian informasi akan dilakukan oleh system sesuai dengan data yang sudah tersedia.

Atas dasar hal tersebut, kami akan membangun sebuah sistem bernama Aplikasi Management Proyek Politeknik Pos Indonesia yang dapat memberikan solusi terhadap masalah yang dihadapi.

Rumusnya cuman sebagai contoh aja biar keren[1].

BAB 1

BAB 1 PEGANTAR

1.1 Manajemen

Manajemen merupakan sebuah kegiatan untuk mengatur dan mengawasi sebuah object yang terkait. Manajemen memiliki fungsi untuk mengurangi masalah yang akan timbul karena object yang tidak diawasi [1]. Selain itu manajemen dapat memberi manfaat seperti mengetahui kebutuhan yang harus dicapai sesuai dengan kondisi objek. Contoh manajemen yang sudah banyak diimplementasi seperti Supply Chain Management. Para ahli memandang ilmu manajemen dengan pengertian beragam. Mary Parker Follet, manajemen adalah seni dalam menyelesaikan tugas melalui perantara. Dalam hal ini, manajemen dapat diartikan sebagai suatu kegiatan yang dilakukan oleh seorang manager untuk mengarahkan bawahan atau orang lain dalam menyelesaikan pekerjaan demi tercapainya sebuah tujuan.

Beralih ke George Robert Terry, yang mengartikan manajemen sebagai proses khas dari beberapa tindakan, seperti perencanaan, pengorganisasian, penggerakan, dan pengawasan. Seluruh tindakan tersebut bertujuan mencapai target dengan memanfaatkan semua sumber daya yang tersedia.

Menurut Ricky W. Griffin, manajemen adalah proses perencanaan, organisasi, koordinasi, dan kontrol pada sumber daya agar tujuan tercapai secara efektif dan

efisien. Efektif di sini maksudnya tujuan tercapai sesuai rencana, dan efisien berarti bahwa manajemen dilakukan secara cermat, terorganisir, dan tepat waktu.

Berbeda, Lawrence A. Appley mengartikan manajemen sebagai keahlian dalam membangkitkan orang lain agar bersedia melakukan sesuatu. Tak harus seseorang, keahlian manajemen juga dapat dimiliki oleh organisasi maupun kelompok.

Terakhir, Hilman berpendapat bahwa manajemen merupakan fungsi untuk mencapai suatu target melalui perantara, serta melakukan pengawasan. Dengan begitu, tujuan dapat tercapai bersama.

Dari beberapa pendapat para ahli mengenai ilmu manajemen tersebut, pengertian manajemen tidak jauh dari usaha untuk mencapai sebuah tujuan dengan cara mengelola dan mengawasi. Dari penjelasan para ahli tersebut, Anda tentu sudah memahami pengertian dari ilmu manajemen.

1.2 Transparansi

Transparansi merupakan sebuah prinsip yang dibangun atas dasar arus informasi yang bebas [2]. Seluruh proses dan informasi harus dapat diakses oleh pihak-pihak yang berkepentingan, serta informasi yang tersedia harus memadai agar dapat dimengerti dan dipantau. Ada tiga model disclosure yang diterapkan oleh LSM:

1. model legalism yang mengacu pada model pengungkapan berbagai data dan informasi organisasi karena adanya tekanan regulasi. Misalnya, mendorong pengungkapan pelaporan keuangan lembaga nirlaba lewat aturan-aturan yang berlaku. Pemberlakuan UU Yayasan merupakan salah satu contoh upaya mendorong lembaga nirlaba untuk terbuka dan mengungkapkan informasi yang dimilikinya berkaitan dengan kepentingan masyarakat. Model ini diterapkan dengan mekanisme audit oleh akuntan publik dan mempublikasikan hasilnya di surat kabar.
2. model associatism di mana pengungkapan data dan informasi dilakukan berdasarkan kesepakatan asosiasi atau konsorsium yang membawahi organisasi tersebut. Misalnya, upaya pengaturan standar pelaporan keuangan yang dilakukan oleh Ikatan Akuntansi Indonesia. Begitu juga upaya beberapa koalisi atau asosiasi LSM, seperti LP3ES, Sawarung dan KPMM, yang membentuk dan menyepakati mekanisme pengungkapan secara khusus bagi para anggotanya. Beberapa upaya yang dilakukan berkaitan dengan model ini adalah laporan pertanggungjawaban anggota secara terbuka dalam sarasehan, atau pembuatan mekanisme/sistem keuangan yang mudah dipahami oleh konstituen.
3. Model kommunalism yang merujuk pada mekanisme pengungkapan data dan informasi yang dimiliki organisasi berdasarkan cara atau metode yang ditentukan oleh komunitas atau masyarakat konstituen. Masyarakatlah yang menentukan bagaimana proses transparansi dilakukan dan informasi apa saja yang harus diungkapkan. Lewat mekanisme semacam ini, pelibatan dan kontrol masyarakat terhadap kinerja organisasi bisa dilakukan secara optimal. Model ini diajukan

sebagai bentuk alternatif terhadap model pertama dan kedua yang dianggap belum mencerminkan transparansi secara utuh dan mudah dimanipulasi, meski sistem yang digunakan sudah cukup rapi dan menggunakan tenaga audit yang profesional.

1.3 API (Application Programming Interface)

API adalah seperangkat perintah, fungsi, serta protocol yang digunakan dalam membangun sebuah perangkat lunak. Dimana API memungkinkan penggunaan fungsi standar untuk berinteraksi dengan sistem yang lain seperti penerjemah dan jembatan agar menjadi lebih mudah [3]. API adalah singkatan dari Application Programming Interface, dan memungkinkan developer untuk mengintegrasikan dua bagian dari aplikasi atau dengan aplikasi yang berbeda secara bersamaan. API terdiri dari berbagai elemen seperti function, protocols, dan tools lainnya yang memungkinkan developers untuk membuat aplikasi.

Tujuan penggunaan API adalah untuk mempercepat proses development dengan menyediakan function secara terpisah sehingga developer tidak perlu membuat fitur yang serupa. Penerapan API akan sangat terasa jika fitur yang diinginkan sudah sangat kompleks, tentu membutuhkan waktu untuk membuat yang serupa dengannya. Misalnya: integrasi dengan payment gateway. Terdapat berbagai jenis sistem API yang dapat digunakan, termasuk sistem operasi, library, dan web.API yang bekerja pada tingkat sistem operasi membantu aplikasi berkomunikasi dengan layer dasar dan satu sama lain mengikuti serangkaian protokol dan spesifikasi. Contoh yang dapat menggambarkan spesifikasi tersebut adalah POSIX (Portable Operating System Interface). Dengan menggunakan standar POSIX, aplikasi yang di-compile untuk bekerja pada sistem operasi tertentu juga dapat bekerja pada sistem lain yang memiliki kriteria yang sama. Software library juga memiliki peran penting dalam menciptakan compatibility antar sistem yang berbeda.

Aplikasi yang berinteraksi dengan library harus mengikuti serangkaian aturan yang ditentukan oleh API. Pendekatan ini memudahkan software developer untuk membuat aplikasi yang berkomunikasi dengan berbagai library tanpa harus memikirkan kembali strategi yang digunakan selama semua library mengikut API yang sama. Kelebihan lain dari metode ini menunjukkan betapa mudahnya menggunakan library yang sama dengan bahasa pemrograman yang berbeda.

Seperti namanya, Web API dalam diakses melalui protokol HTTP, ini adalah konsep bukan teknologi. Kita bisa membuat Web API dengan menggunakan teknologi yang berbeda seperti PHP, Java, .NET, dll. Misalnya Rest API dari Twitter menyediakan akses read dan write data dengan mengintegrasikan twitter kedalam aplikasi kita sendiri.

Fitur Web API :

1. Mendukung fungsi CRUD yang bekerja melalui HTTP protocol dengan method GET, POST, PUT dan DELETE.
2. Memiliki response Accept Header dan HTTP status code

3. Response dengan format JSON, XML atau format apapun yang kamu inginkan. Akan tetapi kebanyakan digunakan kedalam format JSON.
4. Mendukung fitur MVC seperti routing, controllers, action results, filter, model, IOC container, dll.
5. Web API dapat berjalan di Apache atau web server lainnya yang didukung sesuai bahasa pemrograman yang digunakan.

Perbedaan Web API dan Web Service

1. Semua web service menggunakan API tapi tidak semua API digunakan sebagai web service
2. Web service memfasilitasi untuk melakukan interaksi antara dua perangkat atau aplikasi melalui jaringan. Sedangkan API bertindak sebagai penghubung antara dua aplikasi berbeda sehingga bisa berkomunikasi satu sama lain baik dengan ataupun tanpa jaringan.
3. Web service hanya menggunakan 3 style yaitu SOAP, REST, atau XML-RPC untuk berkomunikasi sedangkan API dapat menggunakan style apapun.
4. Web service selalu membutuhkan jaringan untuk pengoperasiannya sedangkan API tidak selalu memerlukan jaringan untuk operasinya.

1.4 Webservice

Web service merupakan suatu sistem yang dirancang guna mendukung interoperabilitas dan interaksi antar sistem pada sebuah jaringan [4]. Web service merupakan fasilitas yang digunakan oleh suatu website untuk menyediakan layanan kepada sistem lain, sehingga bertujuan untuk kolaborasi antar sistem, yang memungkinkan sebuah fungsi yang terdapat pada suatu website dapat digunakan oleh sistem aplikasi lainnya tanpa mengetahui detail program yang terdapat di dalamnya.

Komponen Web Service :

1. SOAP (Simple Object Access Protocol) adalah standar untuk bertukar pesan-pesan berbasis XML melalui jaringan komputer atau sebuah jalan untuk program yang berjalan pada suatu sistem operasi (OS) untuk berkomunikasi dengan program pada OS yang sama maupun berbeda dengan menggunakan HTTP dan XML sebagai mekanisme untuk pertukaran data, maka SOAP dapat berkomunikasi dengan berbagai aplikasi meskipun terdapat perbedaan sistem operasi, teknologi, dan bahasa pemrogramannya.
2. UDDI (Universal Description, Discovery and Integration) merupakan registry dalam format XML yang platform-independent dan dapat digunakan bagi bisnis-bisnis secara global untuk mendaftarkan diri mereka di internet, dan merupakan sebuah mekanisme untuk mencatat (register) dan mencari aplikasi layanan web.

UDDI adalah sebuah inisiatif industri terbuka yang didukung oleh Organization for The Advancement of Structured Information Standards (OASIS) agar bisnis-bisnis dapat menyebarluaskan daftar layanannya dan saling menemukan layanan lain, dan menentukan bagaimana layanan atau aplikasi software berinteraksi melalui internet.

3. WSDL (Web Services Description Language) merupakan kependekan dari Web Services Description Language. WSDL adalah menggunakan XML untuk mendeskripsikan Web services dan bagaimana untuk mengaksesnya. WSDL membantu pengguna web service dalam memakai/menggunakan layanan web service. Web service akan terkunci dan tidak berguna jika tanpa WSDL ini. WSDL menspesifikasikan lokasi service dan operasi (methods) yang disediakan oleh web service.

Manfaat Web Service :

1. Memperlihatkan fungsi yang ada pada jaringan

Web Services adalah unit dikelola kode yang dapat secara remote dipanggil menggunakan HTTP, ini dapat diaktifkan menggunakan permintaan HTTP. Web Services memungkinkan Anda untuk mengekspos fungsi kode yang ada melalui jaringan. Setelah terkena pada jaringan, aplikasi dapat menggunakan fungsionalitas program Anda.

2. Interoperabilitas

Web Services menggunakan standar industri dan protokol standar untuk komunikasi. Semua empat lapis (layanan transportasi, pesan XML, Deskripsi Layanan dan Service Discovery layer) menggunakan protokol yang didefinisikan dengan baik dalam stack protokol layanan web. Ini Standardisasi stack protokol memberikan banyak keuntungan seperti berbagai macam pilihan, pengurangan biaya karena kompetisi, dan peningkatan kualitas bisnis.

3. Protokol standar

Web Services menggunakan standar industri dan protokol standar untuk komunikasi. Semua empat lapis (layanan transportasi, pesan XML, Deskripsi Layanan dan Service Discovery layer) menggunakan protokol yang didefinisikan dengan baik dalam stack protokol layanan web. Ini Standardisasi stack protokol memberikan banyak keuntungan seperti berbagai macam pilihan, pengurangan biaya karena kompetisi, dan peningkatan kualitas bisnis.

4. Biaya Rendah Komunikasi

Web Services menggunakan SOAP terhadap protokol HTTP, sehingga Anda dapat menggunakan internet murah Anda yang ada untuk menerapkan Web Services. Solusi ini jauh lebih murah dibandingkan dengan solusi yang berpemilik seperti EDI B2B. Selain SOAP terhadap HTTP, Web Services juga dapat diterapkan pada mekanisme transport yang handal lainnya seperti FTP.

Karakteristik Web Service :

1. Berbasis XML

Web Services menggunakan XML data representasi dan lapisan transportasi data. Menggunakan XML menghilangkan jaringan, sistem operasi, atau mengikat platform. Aplikasi layanan berbasis web adalah aplikasi yang sangat dioperasikan di tingkat inti mereka.

2. Loosely Coupled

Sebagai konsumen Web Services tidak terikat pada layanan web tersebut secara langsung. Antarmuka web layanan dapat berubah sewaktu-waktu tanpa mengorbankan kemampuan untuk berinteraksi dengan layanan. Sistem erat menyiratkan bahwa logika client dan server yang erat dengan satu sama lain, menyiratkan bahwa jika perubahan satu antarmuka, yang lain harus diperbarui. Mengadopsi arsitektur digabungkan secara longgar yang cenderung untuk membuat sistem perangkat lunak lebih mudah dikelola dan memungkinkan sederhana integrasi antara sistem yang berbeda.

3. Coarse-Grained

Teknologi berorientasi objek seperti Java mengekspos layanan mereka melalui metode individu. Metode individu adalah terlalu halus operasi untuk menyediakan apapun kemampuan yang berguna pada tingkat perusahaan. Membangun program Javadari awal memerlukan penciptaan beberapa metode halus yang kemudian terdiri kelayanan coarse-grained yang dikonsumsi oleh baik klien atau layanan lain.

Bisnis dan antarmuka yang mengekspos mereka harus coarse-grained. Teknologi layanan web menyediakan cara alami untuk mendefinisikan coarse-grained layanan yang mengakses jumlah yang tepat dari logika bisnis.

4. Asynchronous

Sinkronisasi menunjuk kepada pengikatan klien untuk pelaksanaan layanan. Di sinkron invokasi, klien blok dan menunggu untuk layanan untuk menyelesaikan operasi sebelum melanjutkan. Operasi asynchronous memungkinkan klien untuk memohon layanan dan kemudian jalankan fungsi lainnya.

Asinkron klien mengambil mereka hasil pada titik kemudian dalam waktu, sementara sinkron klien menerima hasil mereka ketika layanan telah menyelesaikan. Kemampuan asinkron adalah faktor kunci dalam memungkinkan longgar ditambah sistem.

5. Mendukung Remote Procedure Calls(RPCs)

Web Services memungkinkan klien untuk memanggil metode pada objek terpencil yang menggunakan protokol berbasis XML, fungsi dan prosedur. Prosedur terpencil mengekspos parameter input dan output yang layanan web harus mendukung.

Pengembangan komponen melalui perusahaan JavaBeans (EJBs) dan komponen .NET telah semakin menjadi bagian dari arsitektur dan penyebaran perusahaan selama beberapa tahun. Teknologi kedua didistribusikan dan diakses melalui berbagai mekanisme RPC.

Web Services mendukung RPC menyediakan layanan sendiri, setara dengan komponen tradisional, atau menerjemahkan invokasi yang masuk ke dalam doa EJB atau komponen .NET.

6. Mendukung Dokumen Asing

Salah satu keunggulan utama dari XML adalah cara generik yang mewakili tidak hanya data, tetapi juga dokumen-dokumen kompleks. Dokumen-dokumen ini dapat sebagai sederhana sebagai mewakili alamat saat ini, atau mereka dapat sebagai kompleks yang mewakili seluruh buku atau permintaan penawaran (PPw). Web Services mendukung pertukaran transparan dokumen untuk memfasilitasi integrasi bisnis.

1.5 PHP (Hypertext Preprocessor)

Hypertext Preprocessor adalah kepanjangan dari PHP yaitu salah satu bahasa pemrograman web server-side yang bersifat open source . Bahasa interpreter yang mempunyai kemiripan dengan bahasa C dan Perl yang mempunyai kesederhanaan dalam perintah, yang digunakan untuk pembuatan aplikasi web [5]. Ada beberapa script PHP yang digunakan untuk membuat halaman website yang dinamis. Dinamis berarti halamannya dibuat dan ditampilkan ketika sang client memintanya. Mekanisme ini dapat menyebabkan informasi yang diterima oleh client selalu yang terbaru. Semua script PHP dieksekusi pada server dimana script tersebut dijalankan. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki banyak referensi yang banyak. Salah satu macam kerangka kerja pemrograman web berbasis bahasa pemrograman PHP dan menggunakan pendekatan Model-View-Controller (MVC) adalah CodeIgniter. Tujuan adanya CodeIgniter untuk mempermudah dan mempercepat saat dalam pengerjaan project dari pada menulis code dari awal.

Kelebihan :

1. Memiliki komunitas yang besar
2. Mudah dipelajari
3. Pengembangan cepat

Membuat aplikasi dengan menggunakan PHP akan lebih cepat daripada mengembangkan aplikasi web dengan bahasa pemrograman lainnya. PHP menyuguhkan banyak tools yang tersedia secara open source sehingga mempercepat proses dari start sampai finis.

Projek besar PHP antara lain adalah Prestashop, WordPress, Drupal, dan sebaianya.

Pengembangan aplikasi PHP mudah karena banyak dokumentasi, referensi dan developer yang membantu dalam pengembangannya.

4. Ringkas karena tidak perlu melakukan setting berlebihan, konfigurasi dengan database yang mudah serta proses pengembangan yang tidak memerlukan kompilasi pada saat install. Hal tersebut yang membuat PHP menjadi lebih ringkas dan praktis dibandingkan dengan bahasa pemrograman lainnya. Perlu diingat PHP dapat juga digunakan dalam dokumen HTML.
5. Maintenance mudah
6. Open source
7. PHP dapat membuat web menjadi dinamis
8. Multiplatform

Kekurangan :

1. Kurang pas jika digunakan untuk pengembangan dengan skala besar.
2. Tidak memiliki sistem pemrograman berorientasi objek yang sesungguhnya (sampai versi 4 ini).
3. Tidak dapat memisahkan antara tampilan dengan logika dengan baik.
4. PHP memiliki kelemahan keamanan tertentu jika programmer tidak teliti dalam melakukan pemrograman dan kurang memperhatikan isi dan konfigurasi PHP.
5. Kode PHP mudah dibaca semua orang dan dikompilasi hanya dapat dilakukan dengan tool yang mahal.
6. Untuk menggunakan PHP harus melakukan menginstall web server
7. Banyak kompetisi karena memiliki komunitas yang besar. Setiap saat akan bertambah terus menerus.
8. Terlihat kurang prestisius karena entry level yang berada ditingkat pemula.
9. Tidak ada tipe data pada PHP. Hal tersebut kadang muncul bug yang tidak diinginkan.

1.6 Code Igniter

Code Igniter merupakan sebuah framework PHP yang dapat membantu mempercepat developer dalam pengembangan aplikasi web berbasis PHP. Code Igniter menerapkan konsep MVC (Model, View, Control) yang saling keterkaitan antar satu sama lainnya [6]. MVC adalah konsep yang memisahkan komponen utama menjadi tiga bagian, yaitu :

▪ Model

Model merupakan bagian penanganan yang berhubungan dengan pengolahan atau manipulasi database. Seperti misalnya mengambil data dari database, menginput dan pengolahan database lainnya. Semua instruksi yang berhubungan dengan pengolahan database di letakkan dalam model.

▪ View

View merupakan bagian yang menangani halaman user interface atau halaman yang muncul pada user. tampilan dari user interface di kumpulkan pada view untuk memisahkannya dengan controller dan model sehingga memudahkan web designer dalam melakukan pengembangan tampilan halaman website.

▪ Controller

Controller merupakan kumpulan instruksi aksi yang menghubungkan model dan view, jadi user tidak akan berhubungan dengan model secara langsung, intinya dari view kemudian ke controller yang mengelola instruksi.

Kelebihan :

1. Berukuran sangat kecil. File download nya hanya sekitar 2MB, itupun sudah includedokumentasinya yang sangat lengkap.
2. Dokumentasi yang bagus. Saat anda mendownloadnya, telah disertakan dengan dokumentasi yang berisi pengantar, tutorial, bagaimana panduan penggunaan, serta referensi dokumentasi untuk komponen-komponennya.
3. Kompatibilitas dengan Hosting. CodeIgniter mampu berjalan dengan baik pada hampir semua platform hosting. CodeIgniter juga mendukung database-database paling umum, termasuk MySQL.
4. Tidak ada aturan coding yang ketat. Terserah anda jika anda hanya ingin menggunakan Controller, tanpa View, atau tidak menggunakan Model, atau tidak salah satu keduanya. Namun dengan menggunakan ketiga komponennya adalah pilihan lebih bijak.
5. Kinerja yang baik. Codeigniter sangat cepat bahkan mungkin bisa dibilang merupakan framework yang paling cepat yang ada saat ini.
6. Sangat mudah diintegrasikan. CodeIgniter sangat mengerti tentang pengembangan berbagai library saat ini. Karenanya CodeIgniter memberikan kemudahan untuk diintegrasikan dengan library-library yang tersedia saat ini.
7. Sedikit Konfigurasi. Konfigurasi CodeIgniter terletak di folder application/config. CodeIgniter tidak membutuhkan konfigurasi yang rumit, bahkan untuk mencoba menjalankannya, tanpa melakukan konfigurasi sedikitpun ia sudah bisa berjalan.

8. Mudah dipelajari. Disamping dokumentasi yang lengkap, ia juga memiliki berbagai forum diskusi.

Kekurangan :

1. CodeIgniter tidak ditujukan untuk pembuatan web dengan skala besar.
2. Library yang sangat terbatas. Hal ini dikarenakan sangat sulit mencari plugin tambahan yang terverifikasi secara resmi, karena pada situsnya CodeIgniter tidak menyediakan plugin-plugin tambahan untuk mendukung pengembangan aplikasi dengan CI.
3. Belum adanya editor khusus CodeIgniter, sehingga dalam melakukan create project dan modul-modulnya harus berpindah-pindah folder.

1.7 Node.js

Node.js atau yang biasa dipanggil Node, adalah sebuah server-side environment JavaScript. Ia didasarkan pada implementasi runtime Google - mesin yang dinamai V8. V8 dan Node sebagian besar diimplementasikan dalam C dan C ++, berfokus pada kinerja dan memori rendah konsumsi. Tapi, dimana V8 mendukung terutama JavaScript di browser (terutama Google Chrome), Node bertujuan untuk mendukung jangka panjang proses server [7]. Node.js pertama kali diciptakan dan diperkenalkan oleh Ryan Dahl, pada tahun 2009 sehingga JavaScript bisa digunakan sebagai bahasa pemrograman di sisi server, sekelas dengan PHP, ASP, C# Ruby dan lain sebagainya. Ryan Dahl adalah seorang pengembang dari Joyent yang merupakan sebuah perusahaan perangkat lunak dan infrastruktur Cloud. Ia memiliki ketertarikan dengan penerapan single-threaded pada bahasa pemrograman sisi server dan akhirnya memilih JavaScript sebagai bahasa untuk Node, setelah sebelumnya mencoba menggunakan Haskell, Lua dan C. Berbeda dengan bahasa pemrograman sisi server pada umumnya yang bersifat blocking, Node.js bersifat non-blocking, sebagaimana halnya JavaScript bekerja. Node.js berjalan dengan basis event (event-driven). Maksud dari Blocking secara sederhana adalah, bahwa suatu kode program akan dijalankan hingga selesai, baru kemudian beralih ke kode program selanjutnya.

Kelebihan :

1. Node.js menggunakan bahasa pemrograman JavaScript yang diklaim sebagai bahasa pemrograman yang paling populer dan banyak dikenal oleh masyarakat luas
2. Node.js mampu menangani ribuan koneksi bersamaan dengan penggunaan resource minimum untuk setiap prosesnya
3. Node.js sangat diandalkan terutama untuk membuat aplikasi real-time
4. Node.js adalah project open source, sehingga siapapun dapat melihat struktur kode dan juga dapat berkontribusi untuk pengembangannya

5. Penggunaan JavaScript di sisi server dan juga client meminimalisir ketidakcocokan antar dua sisi lingkungan pemrograman, seperti terkait komunikasi data yang mana menggunakan struktur JSON yang sama di kedua sisi, validasi form yang sama yang dapat dijalankan di sisi server dan client, dan sebagainya.
6. Database NoSQL seperti MongoDB dan CouchDB mendukung langsung Javascript sehingga interfacing dengan database ini akan jauh lebih mudah.
7. Node.js memakai V8 yang selalu mengikuti perkembangan standar ECMAScript (nama standar resmi dari JavaScript, Namun JavaScript yang lebih dikenal dalam implementasinya), sehingga tidak perlu ada kekhawatiran bahwa browser tidak mendukung fitur-fitur di Node.js.

1.8 Unified Modeling Language (UML)

Unified Modeling Language (UML) merupakan teknik untuk merancang sebuah sistem serta aplikasinya. UML merupakan bahasa yang menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi yang akan dirancang. UML juga disebut sebagai alat untuk analisis dan desain yang berorientasi objek. UML dapat digunakan dalam Bahasa pemrograman berorientasi objek seperti C++, Java, dan VB.NET [8].

1.9 Java Script

Javascript merupakan bahasa scripting yang digunakan untuk membuat aplikasi web, sifatnya client-side sehingga dapat diolah langsung di browser tanpa harus terhubung keserver terlebih dahulu. Walaupun namanya menggunakan kata Java, Javascript tidak berhubungan dengan bahasa pemrograman java, meskipun keduanya memiliki kemiripan dalam hal syntax yang meniru bahasa C. Manfaat penggunaan javascript dalam pembuatan website antara lain membuat tampilan website menjadi lebih interaktif [9]. JavaScript dibuat dan didesain selama sepuluh hari oleh Brandan Eich, seorang karyawan Netscape, pada bulan September 1995. Awalnya bahasa pemrograman ini disebut Mocha, kemudian diganti ke Mona, lalu LiveScript sebelum akhirnya resmi menyandang nama JavaScript. Versi pertama dari bahasa ini hanya terbatas di kalangan Netscape saja. Fungsionalitas yang ditawarkan pun terbatas. Namun, JavaScript terus dikembangkan oleh komunitas developer yang tak henti-hentinya mengerjakan bahasa pemrograman ini.

Pada tahun 1996, JavaScript secara resmi disebut sebagai ECMAScript, di mana ECMAScript 2 diluncurkan pada tahun 1998 dan ECMAScript 3 diperkenalkan pada tahun 1999. ECMAScript tersebut dikembangkan hingga akhirnya menjadi JavaScript sebagaimana yang kita kenal saat ini. Tak hanya lintas browser, JavaScript juga bisa digunakan di berbagai perangkat, termasuk perangkat mobile dan komputer.

Sejak saat itu, JavaScript terus bertumbuh dan berkembang. Pada akhirnya di tahun 2016, sebanyak 92 persen website diketahui menggunakan JavaScript. Hanya

dalam kurun waktu dua puluh tahun, JavaScript telah beralih dari bahasa pemrograman yang serba terbatas dan primitif menjadi salah satu tool terpenting bagi web developer. Jika sebagian besar waktu dihabiskan untuk berselancar di internet, maka Anda pastinya sudah sangat familiar dengan JavaScript.

Kelebihan :

1. Anda tidak membutuhkan compiler karena web browser mampu menginterpretasikannya dengan HTML
2. Lebih mudah dipelajari jika dibandingkan dengan bahasa pemrograman lainnya
3. Error atau kesalahan lebih mudah dicari dan ditangani
4. Dapat dialih tugaskan ke elemen halaman web atau even tertentu, misalnya klik atau mouseover
5. JS dapat digunakan di berbagai browser, platform, dan lain-lain
6. Anda dapat menggunakan JavaScript untuk memvalidasi input dan mengurangi keinginan untuk mengecek data secara manual
7. Dengan JavaScript, website Anda menjadi lebih interaktif dan juga mampu menarik perhatian lebih banyak pengunjung
8. Lebih cepat dan ringan jika dibandingkan dengan bahasa pemrograman lainnya

Kelemahan :

1. Berisiko tinggi terhadap eksploitasi
2. Dapat dimanfaatkan untuk mengaktifkan kode berbahaya di komputer pengguna
3. Tidak selalu didukung oleh berbagai browser dan perangkat
4. JS code snippet agak banyak
5. Dapat di-render secara berbeda pada masing-masing perangkat yang malah mengarah ke inkonsistensi

BAB 2

ANALISIS DAN PERANCANGAN

2.1 Analisis Sistem

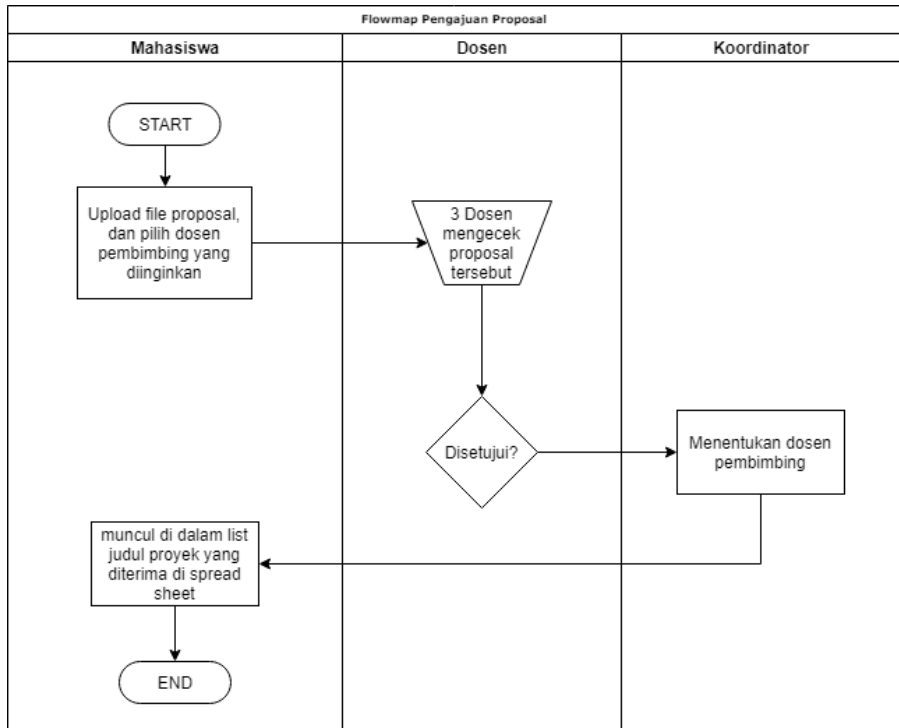
Analisis merupakan tahap awal suatu proses untuk menentukan kebutuhan apa saja yang diperlukan dari suatu aplikasi, dimana terdapat keperluan pada saat membangun aplikasi maupun pada saat implementasi dari aplikasi tersebut. Analisis juga merupakan cara untuk melakukan pemahaman dari suatu sistem informasi yang telah dibuat [10]. Analisis sistem berfungsi sebagai penjelasan dari suatu perancangan sistem ke dalam bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan yang terjadi sehingga dapat mencapai sebuah tujuan perancangan sistem yang dibuat [10]. Analisis juga merupakan cara untuk melakukan pemahaman dari suatu sistem informasi yang telah dibuat. Tahap analisis bertujuan untuk mengetahui sistem informasi, proses atau alur, serta hubungan antar proses yang terdapat dalam aplikasi tersebut [10].

2.1.1 Analisis Yang Sedang Berjalan Pada Proses Kegiatan Proyek

Analisis ini adalah tahap awal untuk perancangan sistem. Analisis ini meliputi analisis prosedur. Dengan demikian, aplikasi yang dibuat akan sesuai dengan prosedur yang ada.

2.1.1.1 Analisis Prosedur (Flowmap) Analisis prosedur pada kegiatan proyek yang sedang berjalan pada Politeknik Pos Indonesia dapat digambarkan dalam bentuk flowmap

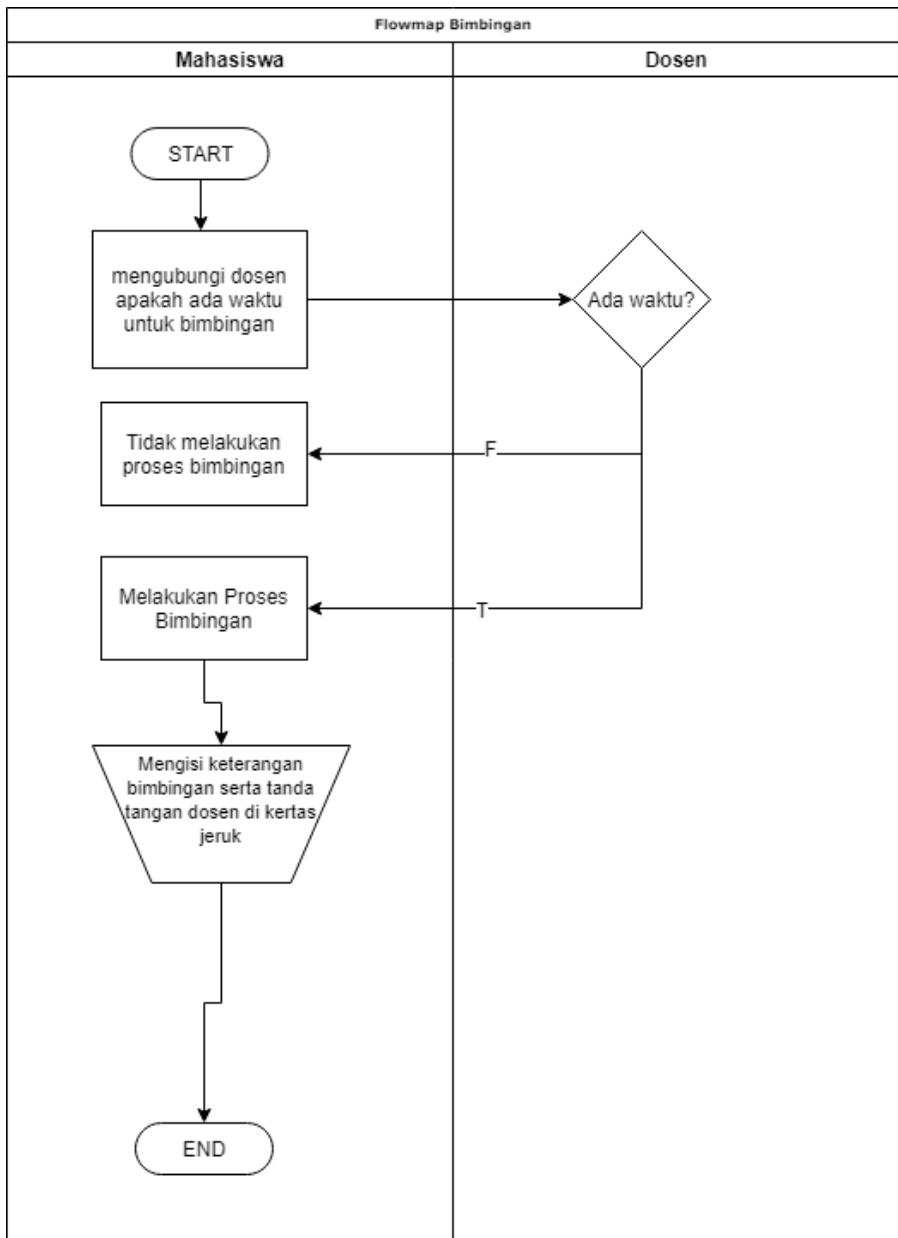
- Analisis Sistem Yang Berjalan Pada Proses Pengajuan Proposal



Gambar 2.1 Flowmap Pengajuan Proposal Sistem yang Sedang Berjalan

Pada gambar flowmap proses pengajuan proposal, terlihat proses ini terjadi antara Mahasiswa dan Dosen. Mahasiswa mulai dengan mengupload file proposal dan memilih dosen pembimbing yang diinginkan, setelah itu akan ada 3 dosen yang mengecek proposal tersebut apakah disetujui atau tidak, dan menentukan dosen pembimbingnya apakah akan sesuai dengan yang diinginkan mahasiswa atau berbeda.

- Analisis Sistem Yang Berjalan Pada Proses Bimbingan



Gambar 2.2 Flowmap Pengajuan Bimbingan Sistem yang Sedang Berjalan

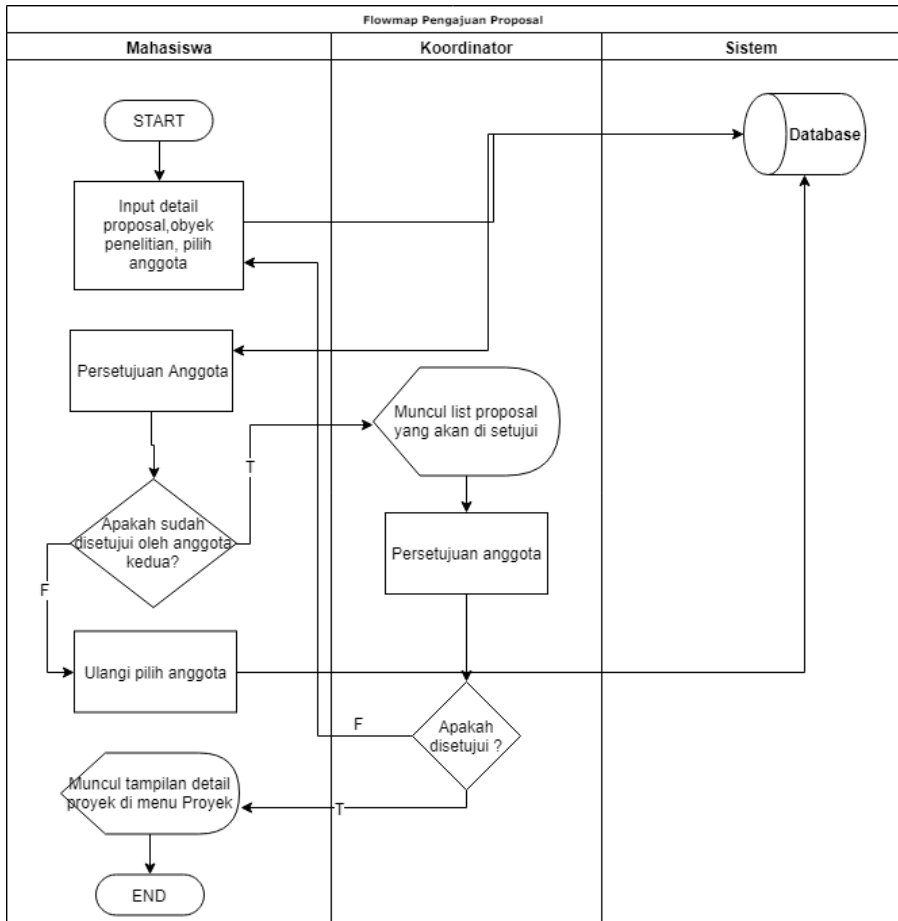
Pada gambar flowmap proses bimbingan, terlihat proses ini terjadi antara Mahasiswa dan Dosen. Mahasiswa mulai dengan mengontak dosen apakah dapat melakukan bimbingan. Jika bias maka proses bimbingan dilakukan, lalu

mengisi keterangan bimbingan yang telah dijalani dan meminta tanda tangan dosen sebagai persetujuan/validasi bahwa proses bimbingan tersebut benar. Hal ini dilakukan diatas kertas jeruk.

2.1.2 Analisis Sistem yang Akan Dibangun

prosedur pada kegiatan proyek akan dibangun pada Politeknik Pos Indonesia dapat digambarkan dalam bentuk flowmap

- Analisis Sistem Yang Akan Dibangun Pada Proses Pengajuan Proposal

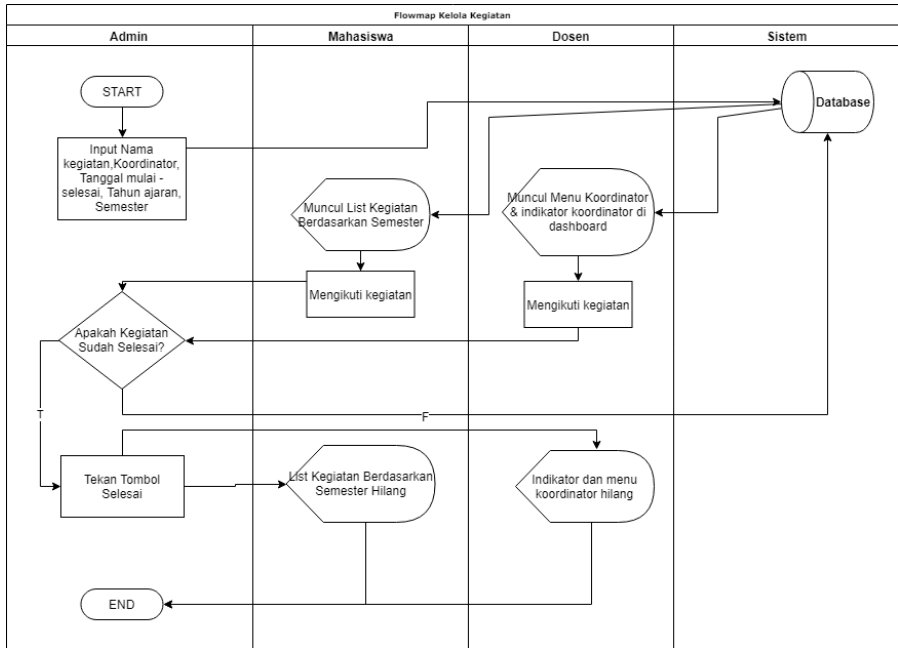


Gambar 2.3 Flowmap Pengajuan Proposal yang Akan Dibangun

Proses pengajuan proposal dimulai dari mahasiswa input detail proposal, obyek penelitian, dan pilih anggota, setelah itu akan menunggu sampai anggota menyetu-

jui bahwa ia akan ikut sebagai anggota, lalu pada menu koordinator akan muncul proposal proposal yang di kumpulkan yang akan di cek apakah memenuhi kriteria atau tidak, jika disetujui maka akan muncul status di tampilan detail proyek di mahasiswa

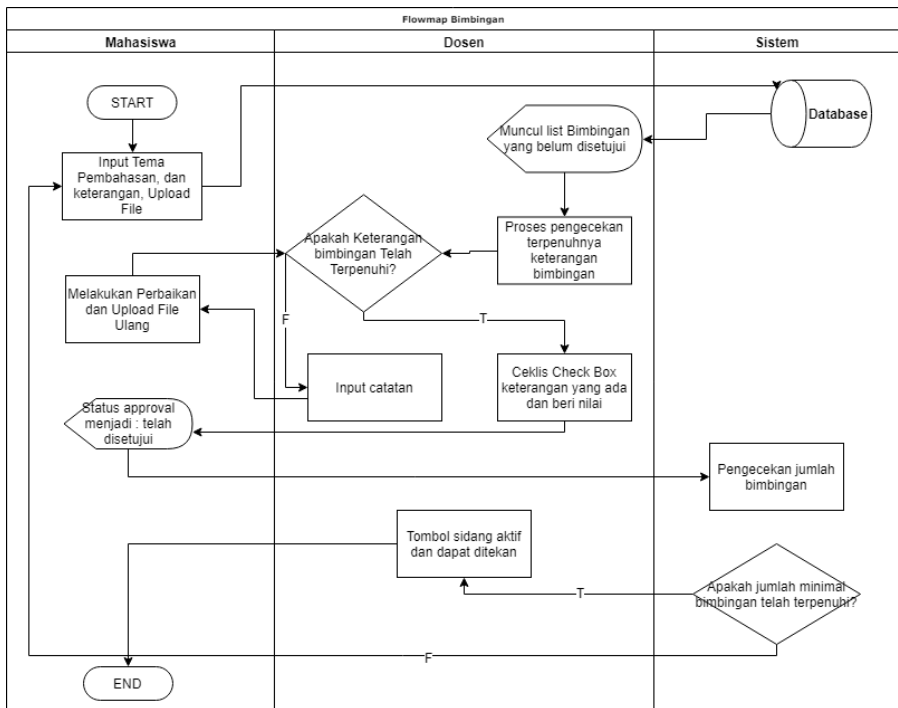
▪ Analisis Sistem Yang Akan Dibangun Pada Proses Kelola Kegiatan



Gambar 2.4 Flowmap Kelola Kegiatan yang Akan Dibangun

Proses kelola kegiatan dimulai dari admin input nama kegiatan, koordinator yang sudah ditentukan, tanggal mulai-selesai, tahun ajaran, dan semester, setelah itu akan muncul list kegiatan pada menu proyek di mahasiswa yang tersortir dengan semester, lalu jika dosen merupakan koordinator maka menu koordinator akan muncul, setelah itu kegiatan proyek berjalan. Apabila sudah selesai, maka admin akan menekan tombol selesai, dan list kegiatan serta menu koordinator akan hilang.

▪ Analisis Sistem Yang Akan Dibangun Pada Proses Bimbingan



Gambar 2.5 Flowmap Bimbingan yang Akan Dibangun

Proses bimbingan dimulai dari mahasiswa input pembahasan, keterangan, dan upload file, setelah itu akan muncul di menu bimbingan dosen, lalu dosen akan mengecek file yang telah di upload apakah keterangan sudah terpehuni(dilaksanakan) , jika sudah, dosen akan memilih checkbox keterangan yang telah di selesaikan, dan memeberikan nilai, maka status bimbingan pada mahasiswa akan berubah menjadi sudah disetujui. Apabila jumlah bimbingan telah memenuhi jumlah minimum, maka tombol sidang akan aktif dan mahasiswa dapat disidangkan

2.1.3 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional merupakan suatu kebutuhan secara lengkap yang berhubungan dengan kebutuhan sistem yang dibuat. Adapun kebutuhan fungsional yang akan dibuat adalah sebagai berikut :

1. Login
2. Kelola Data Kegiatan
3. Kelola Data Dosen
4. Kelola Data Mahasiswa

5. Pengajuan Proposal
6. Pengajuan Bimbingan
7. Penyetujuan Bimbingan dan Sidang
8. Kelola Penilaian
9. Persetujuan Nilai
10. Penjadwalan Sidang
11. Penyetujuan Proposal
12. Pengaturan Proyek

Setiap proses memiliki fungsi masing-masing pada sebuah table atau data yang terdapat pada database yang telah dirancang sebelumnya. Dan setiap proses berbubungan langsung dengan entitas atau user.

2.1.4 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional dilakukan untuk mengetahui spesifikasi kebutuhan untuk perancangan sistem meliputi operasional sistem, dan keamanan sistem. Spesifikasi kebutuhan melibatkan analisis perangkat keras (Hardware) dan perangkat lunak (Software) .

1. Kebutuhan Perangkat Keras (Hardware)

Analisis yang akan dibuat ini digunakan untuk membantu proses pengolahan data pengajuan. Antara lain melibatkan Admin, Dosen, Koordinator, dan Mahasiswa.

No.	Nama Perangkat	Spesifikasi	Keterangan
1.	Harddisk	HDD atau SSD	Media untuk menyimpan data aplikasi yang dibuat.
2.	Memory	1 GB	Memory sistem yang digunakan.
3.	Processor	Intel atau AMD	Untuk kecepatan transfer data dari sistem yang sangat bergantung pada kecepatan processor perangkat.
4.	Infrastruktur Jaringan		Bisa dianalogikan sebagai alur proses dari titik awal proses sampai pada akhir proses.
5.	Monitor		Untuk menampilkan isi perancangan sistem.

Gambar 2.6 Kebutuhan Perangkat Keras Server

2. Kebutuhan Perangkat Lunak

Analisis yang akan dibuat ini digunakan untuk membantu proses pengolahan data pengajuan. Dimana perangkat lunak yang dibutuhkan untuk membuat serta mengakses sistem yang akan dibangun ini.

No.	Tools/Software	Fungsi
1.	Windows	Sistem operasi
2.	Html, PHP, Framework Code Igniter, JavaScript.	Bahasa pemrograman yang digunakan
3.	PDF (<i>Portable Document Format</i>)	<i>Document</i> laporan
4.	Google Chrome	<i>Web browser</i>

Gambar 2.7 Kebutuhan Perangkat Lunak Server

No.	Tools/Software	Fungsi
1.	<i>Web Browser</i>	Untuk membuka website

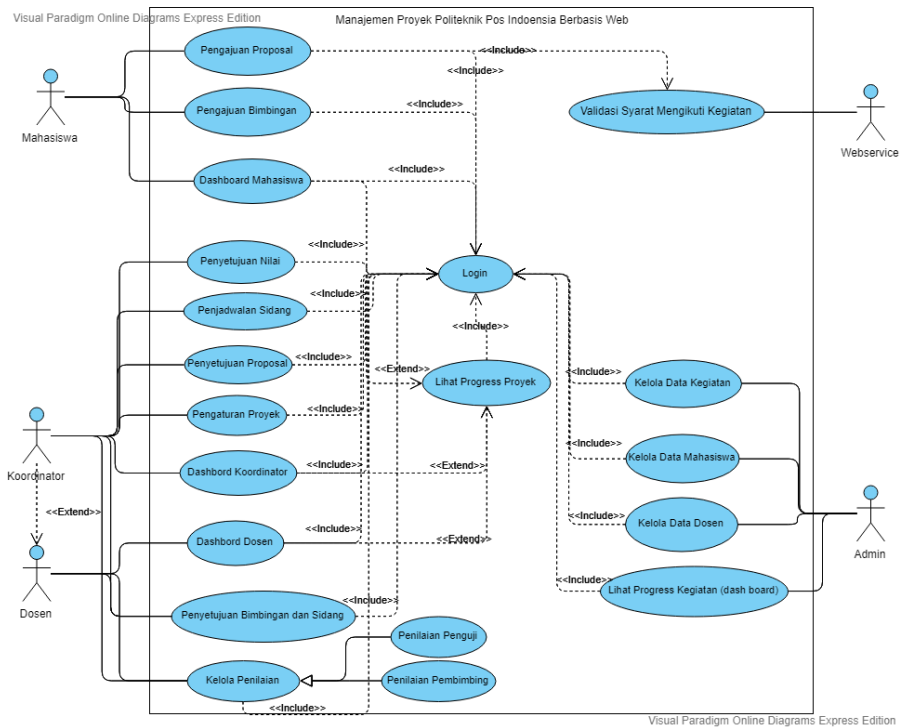
Gambar 2.8 Kebutuhan Perangkat Lunak Client

2.2 Perancangan Sistem

Berikut ini adalah suatu gambar analisa data secara kompleks dapat diimplementasikan pada Aplikasi Manajemen Proyek Politeknik Pos Indoensia, menggunakan notasi UML (Unified Modeling Language).

2.2.1 Use Case Diagram

Use case diagram adalah sebuah diagram yang menunjukkan suatu kelompok use case dan actors beserta dengan relationships-nya atau hubungannya secara kompleks [11].



Gambar 2.9 Use Case

2.2.1.1 Definisi Aktor Aktor adalah entitas - entitas yang terlibat dalam sebuah sistem aktor tidak hanya dapat berupa orang atau manusia tapi juga hal yang lainnya. Pada bagian ini akan dijelaskan aktor - aktor yang ada dalam sistem ini.

No.	Aktor	Deskripsi
1.	Admin	a. <u>Kelola Data Kegiatan</u> b. <u>Kelola Data Mahasiswa</u> c. <u>Kelola Data Dosen</u> d. <u>Lihat Proses Kegiatan(Dashboard)</u>
2.	Mahasiswa	a. <u>Pengajuan Proposal</u> b. <u>Pengajuan Bimbingan</u> c. <u>Dashboard Mahasiswa</u>
3.	Dosen	a. <u>Penyetujuan Bimbingan dan Sidang</u> b. <u>Kelola Penilaian</u> c. <u>Dashboard Dosen</u>
4.	Koodinator	a. <u>Penyetujuan Bimbingan dan Sidang</u> b. <u>Kelola Penilaian</u> c. <u>Dashboard Koordinator</u> d. <u>Persetujuan Nilai</u> e. <u>Penjadwalan Sidang</u> f. <u>Persetujuan Proposal</u> g. <u>Pengaturan Proyek</u>
5.	Webservis	a. <u>Validasi Mengikuti Kegiatan</u>

Gambar 2.10 Definisi Aktor

2.2.1.2 Definisi Use Case Use case merupakan teknik untuk merekam persyaratan fungsional sebuah sistem. Pada bagian ini akan dijelaskan proses proses yang ada pada usecase.

No	Deskripsi Use Case	Penjelasan
1.	<u>Login</u>	Merupakan aktivitas validasi Admin, Dosen, Mahasiswa, Koordinator agar dapat melakukan akses ke dalam sistem.
2.	<u>Kelola Data Kegiatan</u>	Aktor Admin melakukan kelola data kegiatan proyek.
3.	<u>Kelola Data Mahasiswa</u>	Aktor Admin melakukan kelola data mahasiswa dengan mengupload file excel
4.	<u>Kelola Data Dosen</u>	Aktor Admin melakukan kelola data dosen dengan menginputkan detail data dosen
5.	<u>Pengajuan Proposal</u>	Aktir mahasiswa melakukan pengajuan proposal dengan cara menginputkan detail proposal dan mengupload file berformat pdf
6.	<u>Pengajuan Bimbingan</u>	Aktor Mahasiswa melakukan bimbingan dengan meninputkan parameter progress bimbingan yang telah disepakati dan mengupload file progress

Gambar 2.11 Definisi Use Case

7.	<u>Penyetujuan Bimbingan dan Sidang</u>	Aktor Dosen menyetujui sesi bimbingan dengan mencentang checkbox apabila file yang telah di upload mahasiswa sesuai dengan progress yang diinginkan. Apabila jumlah total bimbingan telah memenuhi jumlah minimal bimbingan, maka dosen dapat menyetujui siding mahasiswa dengan menekan tombol siding.
8.	<u>Kelola Penilaian</u>	Aktor Dosen melakukan penginputan nilai setelah melakukan proses siding terhadap mahasiswa, baik sebagai penguji maupun pembimbing.
9.	<u>Persetujuan Nilai</u>	Aktor Koordinator akan mempertimbangkan nilai yang telah diberikan oleh para dosen yang telah melakukan proses siding terhadap mahasiswa dan menetapkan nilai akhirnya
10.	<u>Penjadwalan Sidang</u>	Aktor Koodinator akan memntukan tanggal sidang, waktu siding, dan ruangan siding kepada mahasiswa yang telah di setuju dapat melakukan siding oleh dosen pembimbingnya
11.	<u>Persetujuan Proposal</u>	Aktor Koordinator akan mempertimbangkan proposal yang telah diinputkan oleh mahasiswa apakah sesuai dengan ketentuan atau tidak

Gambar 2.12 Definisi Use Case

12.	<u>Pengaturan Proyek</u>	Aktor <u>Koodinator</u> akan <u>menentukan</u> parameter-parameter yang akan <u>menjadi</u> <u>penilaian</u> pada <u>proyek</u> serta <u>persentase</u> nya
13.	<u>Validasi Kegiatan</u>	Aktor <u>Webservis</u> akan <u>menyediakan</u> data <u>nilai</u> yang <u>menjadi</u> <u>syarat</u> <u>mengikuti</u> <u>kegiatan</u> dan status <u>krs</u> sebagai <u>validasi</u> untuk <u>mengirimkan</u> <u>proposal</u>

Gambar 2.13 Definisi Use Case

a. Identifikasi	
Nomor Uji	UC001
Nama	Login
Tujuan	Memberikan hak akses pengguna terhadap sistem dengan melakukan validasi terhadap <i>nik/npm</i> , <i>password</i> , yang dimasukan oleh pengguna.
b. Deskripsi	
Aktor	Admin, Dosen, Mahasiswa, Koordinator
c. Skenario Utama	
Kondisi Awal	Form Login sudah tersedia
Aksi Aktor	Reaksi Sistem
Memasukkan <i>nik/npm</i> dan <i>password</i>	Form Login akan menampilkan <i>textbox</i> <i>nik/npm</i> dan <i>password</i> .
Admin, Dosen, Mahasiswa dan Koordinator melakukan konfirmasi persetujuan terhadap <i>nik/npm</i> , <i>password</i> yang telah dimasukan dengan menekan tombol Login.	Aplikasi melakukan validasi terhadap <i>nik/npm</i> , <i>password</i> yang telah dimasukan oleh pengguna dengan melakukan pengecekan pada basis data.
Kondisi Akhir	Pada akhir interaksi <i>nik/npm</i> , <i>password</i> yang dimasukan pengguna valid, maka pengguna atau user akan langsung masuk kehalaman utama dan dapat menggunakan sistem sesuai hak aksesnya.

Gambar 2.14 Skenario Use Case Login

2.2.1.3 Skenario Use Case Login

a. Identifikasi		
Nomor Uji	UC002	
Nama	Kelola Data Kegiatan	
Tujuan	Mengelola data kegiatan	
b. Deskripsi		
Aktor	Admin	
c. Skenario Utama		
Kondisi Awal	Admin <i>login</i> terlebih dahulu jika <i>valid</i> maka masuk ke <i>dashboard</i> admin jika tidak <i>valid</i> maka akan <u>muncul pesan error</u> bahwa <i>nik</i> dan <i>password</i> tidak sesuai.	
Aksi Aktor	Reaksi Sistem	
Admin menambahkan data	Sistem menampilkan form detail kegiatan dalam modal	
Admin melakukan kelola data	Sistem akan menampilkan data yang akan dikelola.	
Kondisi Akhir	List Data.	

Gambar 2.15 Skenario Kelola Data Kegiatan

2.2.1.4 Skenario Kelola Data Kegiatan

a. Identifikasi	
Nomor Uji	UC003
Nama	Kelola Data Mahasiswa
Tujuan	Mengelola data mahasiswa
b. Deskripsi	
Aktor	Admin
c. Skenario Utama	
Kondisi Awal	Admin <i>login</i> terlebih dahulu jika <i>valid</i> maka masuk ke <i>dashboard</i> admin jika tidak <i>valid</i> maka akan muncul pesan <i>error</i> bahwa <i>nik</i> dan <i>password</i> tidak sesuai.
Aksi Aktor	Reaksi Sistem
Admin menambahkan data	Sistem menampilkan form upload
Admin melakukan kelola data	Sistem akan menampilkan data yang akan dikelola.
Kondisi Akhir	List data.

Gambar 2.16 Skenario Kelola Data Mahasiswa

2.2.1.5 Skenario Kelola Data Mahasiswa

a. Identifikasi	
Nomor Uji	UC004
Nama	Kelola Data Dosen
Tujuan	Mengelola data dosen
b. Deskripsi	
Aktor	Admin
c. Skenario Utama	
Kondisi Awal	Admin <i>login</i> terlebih dahulu jika <i>valid</i> maka masuk ke <i>dashboard</i> admin jika tidak <i>valid</i> maka akan <u>muncul pesan error</u> bahwa <i>nik</i> dan <i>password</i> tidak sesuai.
Aksi Aktor	Reaksi Sistem
Admin menambahkan data	Sistem menampilkan form detail dosen dalam modal
Admin melakukan kelola data	Sistem akan menampilkan data yang akan dikelola.
Kondisi Akhir	List Data.

Gambar 2.17 Skenario Kelola Data Dosen

2.2.1.6 Skenario Kelola Data Dosen

a. <u>Identifikasi</u>	
<u>Nomor Uji</u>	UC005
<u>Nama</u>	Pengajuan proposal
<u>Tujuan</u>	Mengajukan proposal
b. <u>Deskripsi</u>	
<u>Aktor</u>	<u>Mahasiswa</u>
c. <u>Skenario Utama</u>	
<u>Kondisi Awal</u>	Mahasiswa <i>login</i> terlebih dahulu jika <i>valid</i> maka masuk ke <i>dashboard</i> mahasiswa jika <i>tidak valid</i> maka akan muncul pesan <i>error</i> bahwa <i>npm</i> dan <i>password</i> tidak sesuai.
<u>Aksi Aktor</u>	<u>Reaksi Sistem</u>
Mahasiswa menginput detail proposal dan mengirim file proposal tersebut dalam bentuk pdf	Sistem menampilkan form detail proposal dalam modal
<u>Kondisi Akhir</u>	Data Detail Proposal.

Gambar 2.18 Skenario Pengajuan Proposal

2.2.1.7 Skenario Pengajuan Proposal

a. Identifikasi	
Nomor Uji	UC006
Nama	Pengajuan bimbingan
Tujuan	Mengajukan bimbingan
b. Deskripsi	
Aktor	Mahasiswa
c. Skenario Utama	
Kondisi Awal	Mahasiswa <i>login</i> terlebih dahulu jika <i>valid</i> maka masuk ke <i>dashboard</i> mahasiswa jika tidak <i>valid</i> maka akan muncul pesan <i>error</i> bahwa <i>npm</i> dan <i>password</i> tidak sesuai.
Aksi Aktor	Reaksi Sistem
Mahasiswa menginput parameter bimbingan dan mengirim file progress.	Sistem menampilkan data bimbingan
Kondisi Akhir	List data bimbingan.

Gambar 2.19 Skenario Pengajuan Bimbingan

2.2.1.8 Skenario Pengajuan Bimbingan

a. <u>Identifikasi</u>	
<u>Nomor Uji</u>	UC007
<u>Nama</u>	<u>Penyetujuan Bimbingan dan Sidang</u>
<u>Tujuan</u>	<u>Menyetujui sesi bimbingan dan menyetujui kegiatan sidang</u>
b. <u>Deskripsi</u>	
<u>Aktor</u>	<u>Dosen, Koordinator</u>
c. <u>Skenario Utama</u>	
<u>Kondisi Awal</u>	<u>Dosen, Koordinator login terlebih dahulu jika valid maka masuk ke form admin jika tidak valid maka akan muncul pesan error bahwa nik dan password tidak sesuai.</u>
<u>Aksi Aktor</u>	<u>Reaksi Sistem</u>
<u>Dosen, Koordinator melihat progress bimbingan dari file yang telah di upload, jika sudah sesuai maka men-check pada check box parameter</u>	<u>Sistem mengupdate status bimbingan pada mahasiswa menjadi "sudah disetujui"</u>
<u>Dosen, Koordinator menekan tombol sidang apabila jumlah bimbingan sudah memenuhi jumlah minmun bimbingan</u>	<u>Sistem akan mengubah status proyek sehingga akan tampil pada menu penjadwalan pada koordinator</u>
<u>Kondisi Akhir</u>	<u>List Data Bimbingan</u>

Gambar 2.20 Skenario Penyetujuan Bimbingan dan Sidang

2.2.1.9 Skenario Penyetujuan Bimbingan dan Sidang

a. Identifikasi	
Nomor Uji	UC008
Nama	Kelola Penilaian
Tujuan	Mengelola Penilaian
b. Deskripsi	
Aktor	Dosen, Koordinator
c. Skenario Utama	
Kondisi Awal	Dosen, Koordinator <i>login</i> terlebih dahulu jika <i>valid</i> maka masuk ke <i>form</i> admin jika tidak <i>valid</i> maka akan <u>muncul pesan <i>error</i></u> bahwa <i>nik</i> dan <i>password</i> tidak sesuai.
Aksi Aktor	Reaksi Sistem
Dosen, Koordinator menginputkan nilai sebagai pembimbing	Sistem memasukkan inputkan kedalam database sehingga akan muncul di menu penyetujuan nilai di koordinator
Dosen, Koordinator menginputkan nilai sebagai penguji	Sistem memasukkan inputkan kedalam database sehingga akan muncul di menu penyetujuan nilai di koordinator
Kondisi Akhir	List Data Mahasiswa yang diuji atau di bimbing

Gambar 2.21 Skenario Kelola Penilaian

2.2.1.10 Skenario Kelola Penilaian

a. Identifikasi	
Nomor Uji	UC009
Nama	Penyetujuan Nilai
Tujuan	Menyetujui/ Menetapkan Nilai
b. Deskripsi	
Aktor	Koordinator
c. Skenario Utama	
Kondisi Awal	Koordinator <i>login</i> terlebih dahulu jika <i>valid</i> maka masuk ke <i>form</i> admin jika tidak <i>valid</i> maka akan <u>muncul pesan error</u> bahwa <i>nik</i> dan <i>password</i> tidak sesuai.
Aksi Aktor	Reaksi Sistem
Menetapkan keputusan nilai akhir dari mahasiswa	Sistem mengubah status nilai tersebut sehingga di menu proyek mahasiswa status nilai tersebut berubah
Kondisi Akhir	List Data Nilai Mahasiswa

Gambar 2.22 Skenario Penyetujuan Nilai

2.2.1.11 Skenario Penyetujuan Nilai

a. <u>Identifikasi</u>	
<u>Nomor Uji</u>	UC010
<u>Nama</u>	<u>Penjadwalan Sidang</u>
<u>Tujuan</u>	<u>Menetapkan Jadwal Sidang</u>
b. <u>Deskripsi</u>	
<u>Aktor</u>	<u>Koordinator</u>
c. <u>Skenario Utama</u>	
<u>Kondisi Awal</u>	<u>Koordinator login terlebih dahulu jika valid maka masuk ke form admin jika tidak valid maka akan muncul pesan error bahwa nik dan password tidak sesuai.</u>
<u>Aksi Aktor</u>	<u>Reaksi Sistem</u>
<u>Menginputkan tanggal sidang, waktu sidang, ruangan sidang</u>	<u>Sistem mengupdate tabel sehingga tampilan di section Status Sidang di menu proyek mahasiswa berubah</u>
<u>Kondisi Akhir</u>	<u>List Data Mahasiswa siap sidang</u>

Gambar 2.23 Skenario Penjadwalan Sidang

2.2.1.12 Skenario Penjadwalan Sidang

a. Identifikasi	
Nomor Uji	UC011
Nama	Penyetujuan Proposal
Tujuan	Menyetujui Proposal
b. Deskripsi	
Aktor	Koordinator
c. Skenario Utama	
Kondisi Awal	Koordinator login terlebih dahulu jika valid maka masuk ke form admin jika tidak valid maka akan muncul pesan error bahwa nik dan password tidak sesuai.
Aksi Aktor	Reaksi Sistem
Menolak atau menyetujui proposal, jika setuju berikan alasan approval dan pilih dosen pembimbing.	Sistem mengupdate status dari proposal yang telah diinputkan mahasiswa sehingga tampilan menu proyek mahasiswa berubah
Kondisi Akhir	List Data Proposal

Gambar 2.24 Skenario Penyetujuan Proposal

2.2.1.13 Skenario Penyetujuan Proposal

a. Identifikasi	
Nomor Uji	UC012
Nama	Pengaturan Proyek
Tujuan	Mengatur parameter proyek
b. Deskripsi	
Aktor	Koordinator
c. Skenario Utama	
Kondisi Awal	Koordinator <i>login</i> terlebih dahulu jika <i>valid</i> maka masuk ke <i>form</i> admin jika tidak <i>valid</i> maka akan <u>muncul pesan error</u> bahwa <i>nik</i> dan <i>password</i> tidak sesuai.
Aksi Aktor	Reaksi Sistem
Mengatur parameter untuk penilaian serta persentase nilai	Sistem mengupdate detail kegiatan proyek tersebut
Kondisi Akhir	List Data Parameter

Gambar 2.25 Skenario Pengaturan Proyek

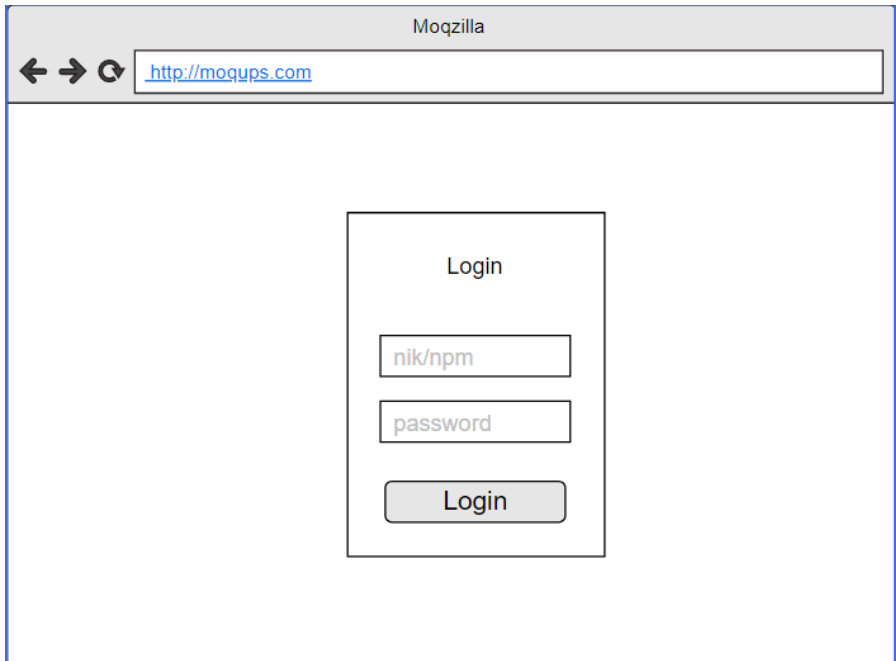
2.2.1.14 Skenario Pengaturan Proyek

2.3 User Interface

User Interface dirancang untuk merancang antarmuka untuk membangun website dari perancangan sistem yang dibuat.

2.3.1 Login

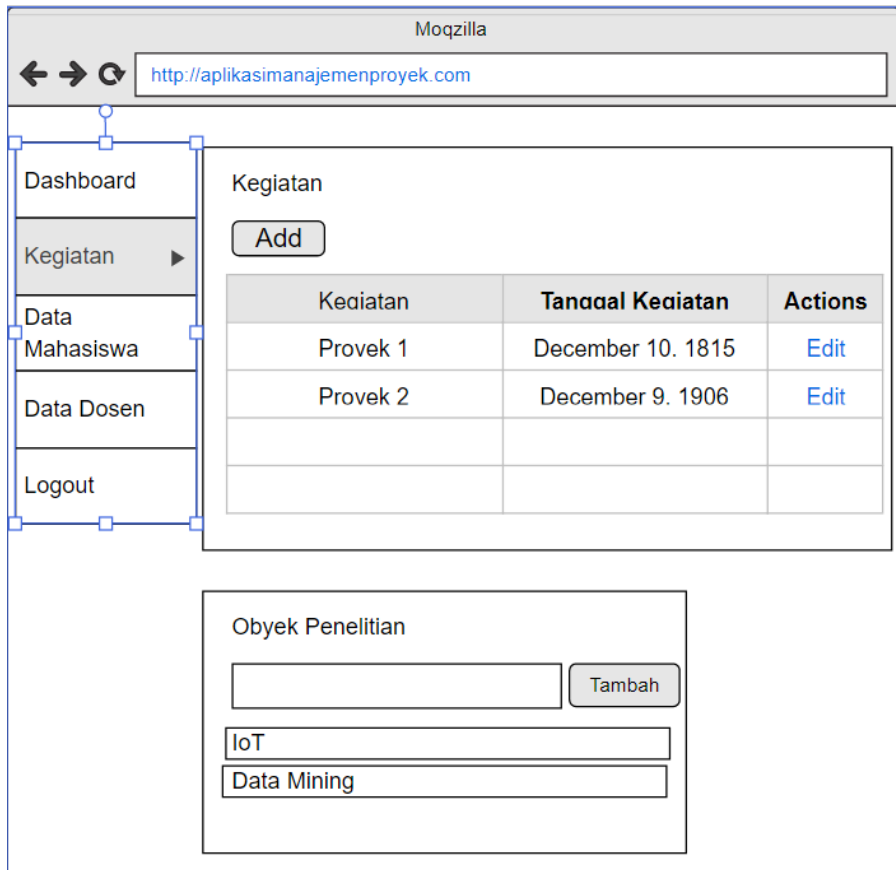
Tampilan rancangan user interface dari login



Gambar 2.26 user Interface Login

2.3.2 Kelola Data Kegiatan

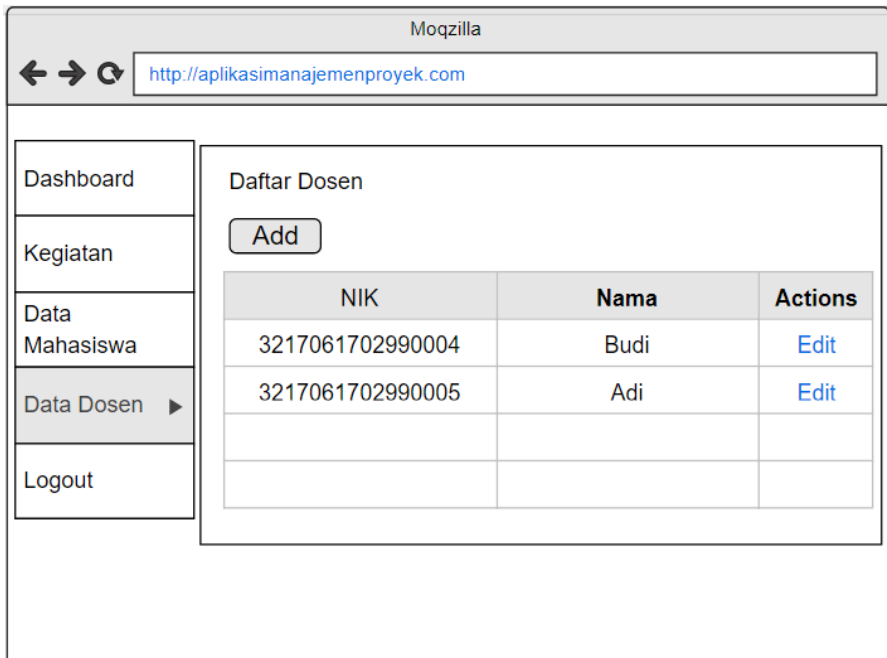
Tampilan rancangan user interface dari Kelola Data



Gambar 2.27 user Interface Kelola Data Kegiatan

2.3.3 Kelola Data Dosen

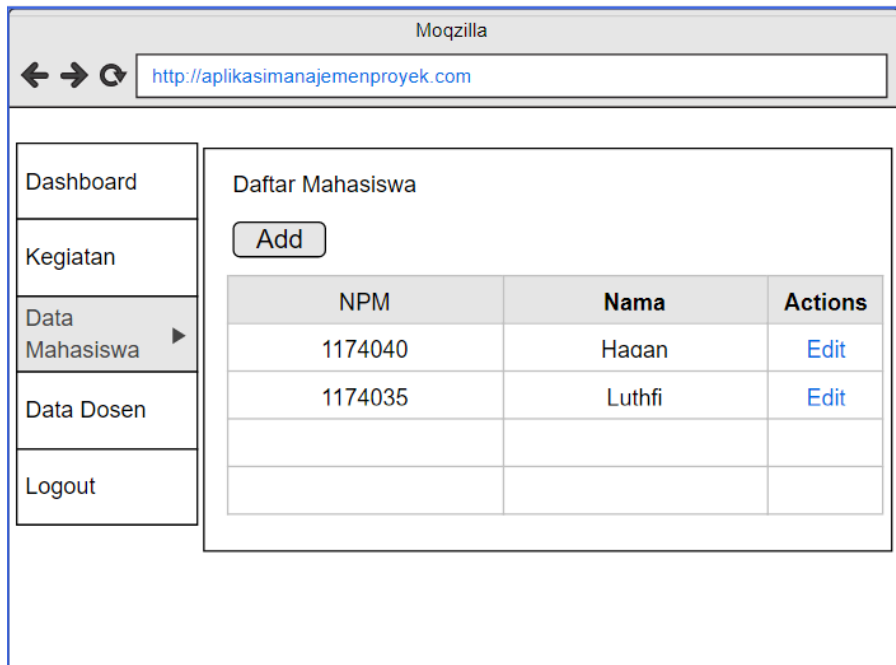
Tampilan rancangan user interface dari Kelola Data Dosen



Gambar 2.28 user Interface Kelola Data Dosen

2.3.4 Kelola Data Mahasiswa

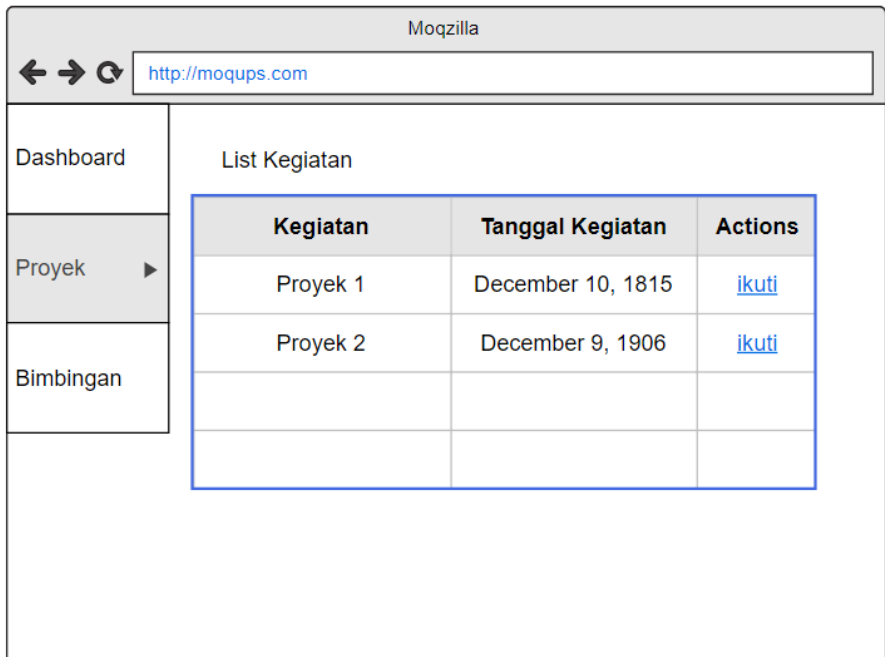
Tampilan rancangan user interface dari Kelola Data Mahasiswa



Gambar 2.29 user Interface Kelola Data Mahasiswa

2.3.5 Pengajuan Proposal

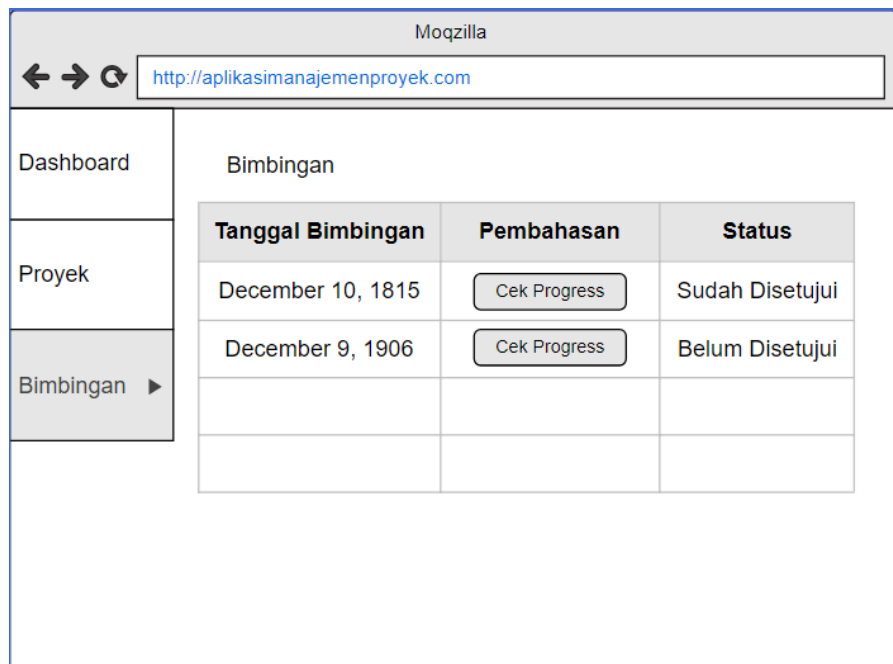
Tampilan rancangan user interface dari Pengajuan Proposal



Gambar 2.30 user Interface Pengajuan Proposal

2.3.6 Pengajuan Bimbingan

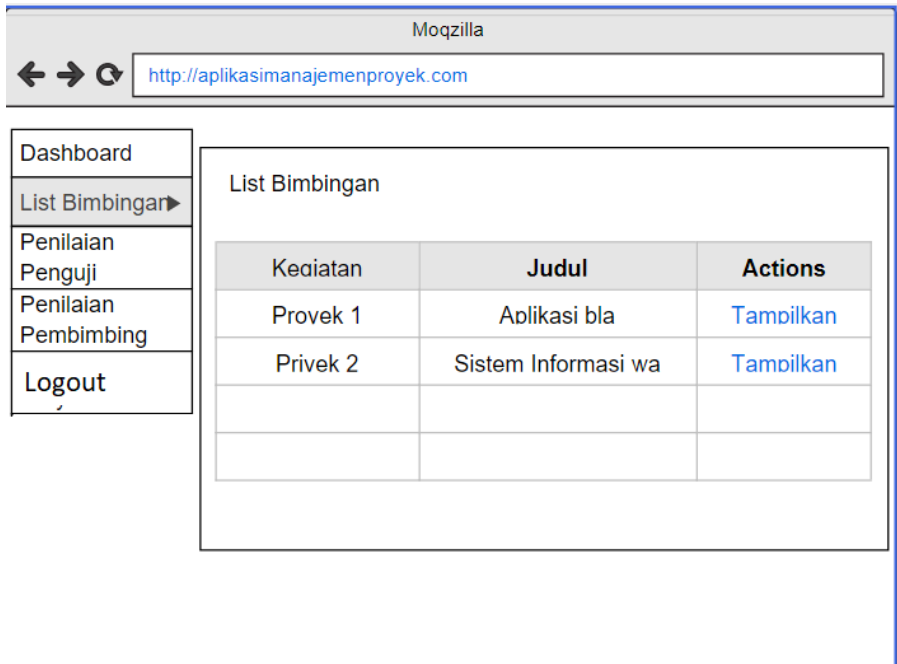
Tampilan rancangan user interface dari Pengajuan Bimbingan



Gambar 2.31 user Interface Pengajuan Bimbingan

2.3.7 List Bimbingan

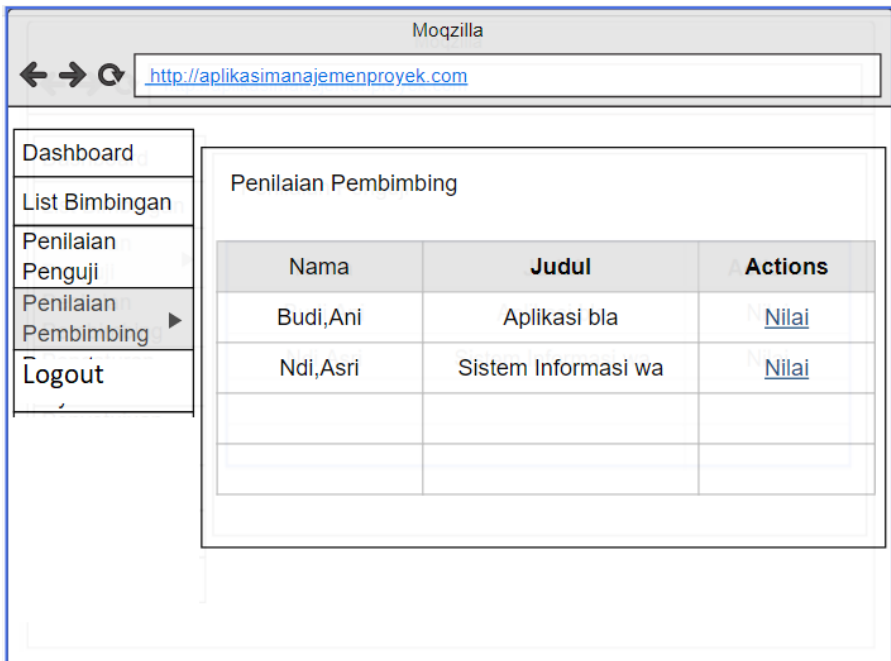
Tampilan rancangan user interface dari List Bimbingan



Gambar 2.32 user Interface List Bimbingan

2.3.8 Penilaian pembimbing

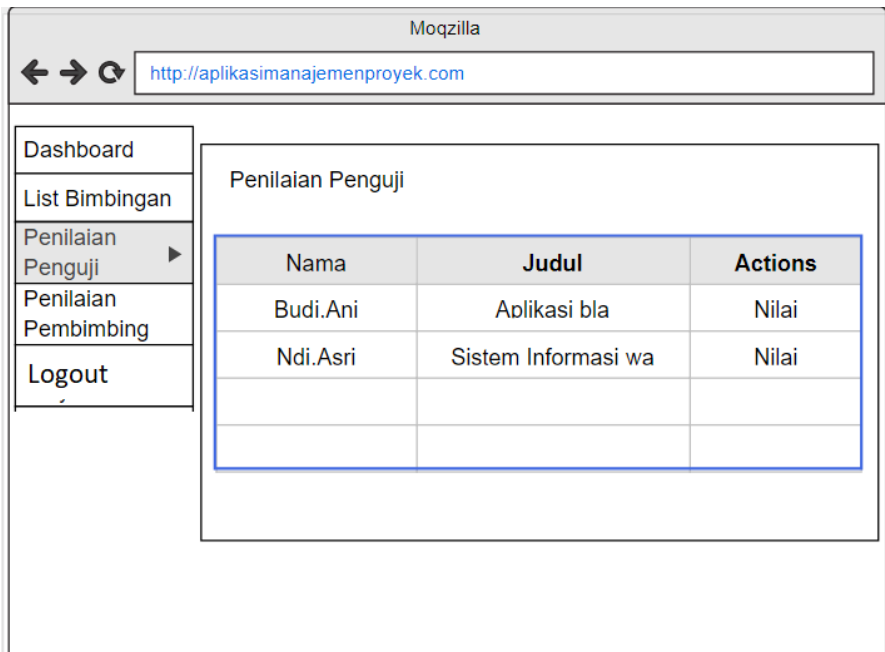
Tampilan rancangan user interface dari Penilaian Pembimbing



Gambar 2.33 user Interface Penilaian Pembimbing

2.3.9 Penilaian Penguji

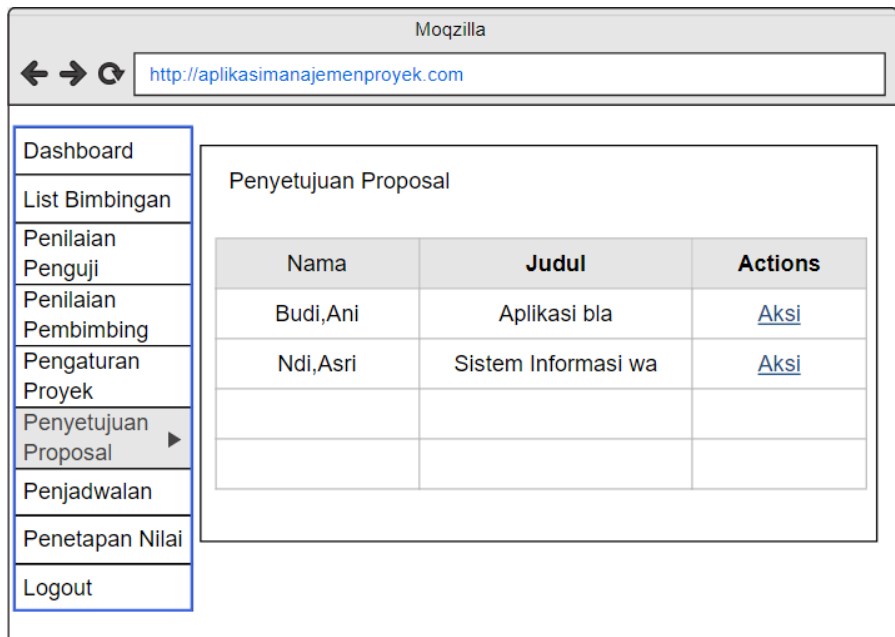
Tampilan rancangan user interface dari Penilaian Penguji



Gambar 2.34 user Interface Penilaian Penguji

2.3.10 Penyetujuan Proposal

Tampilan rancangan user interface dari Penyetujuan Proposal



Gambar 2.35 user Interface Penyetujuan Proposal

2.3.11 Pengaturan Proyek

Tampilan rancangan user interface dari Pengaturan Proyek

Moqzilla

← → ↻ <http://aplikasimanajemenproyek.com>

Dashboard

List Bimbingan

Penilaian Penguji

Penilaian Pembimbing

Pengaturan Proyek ▶

Penyetujuan Proposal

Penjadwalan

Penetapan Nilai

Logout

Detail Kegiatan

Nama Proyek

Batas Minimal Bimbingan

Persentase Nilai Sidang

Persentase Nilai Bimbingan

Persentase Nilai Progress

Update

Setting Parameter Progress

Tambah

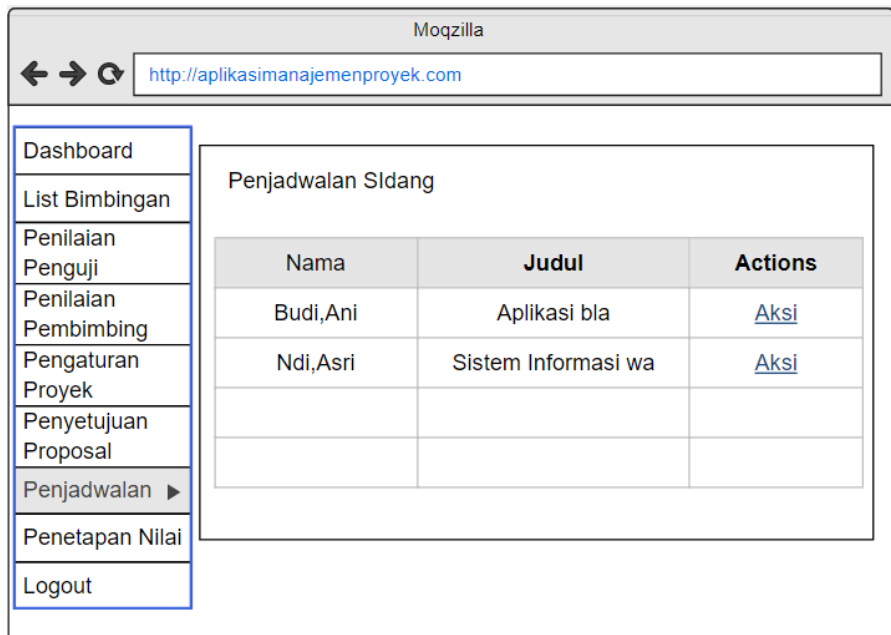
Bab 1

Bab 2

Gambar 2.36 user Interface Pengaturan Proyek

2.3.12 Penjadwalan

Tampilan rancangan user interface dari Penjadwalan



Gambar 2.37 user Interface Penjadwalan

BAB 3

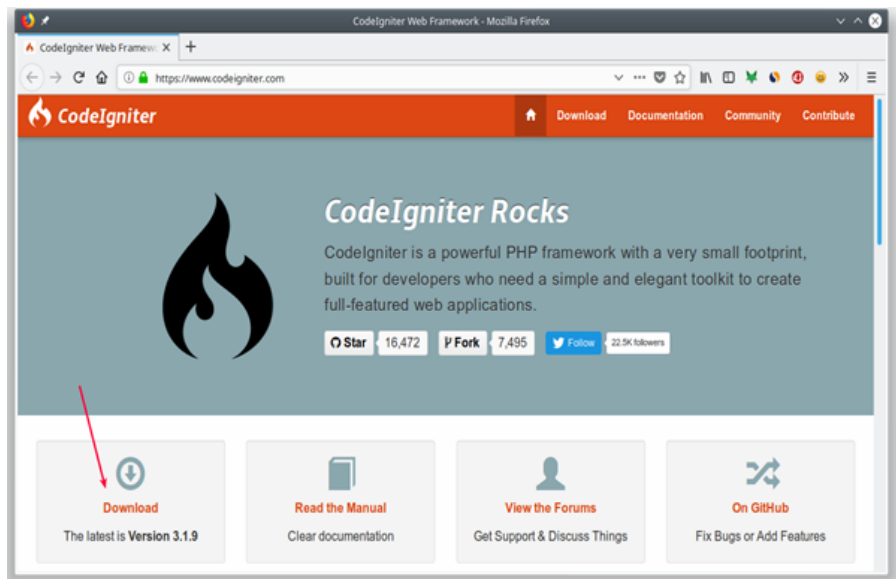
CARA INSTALASI TOOLS

3.1 Code Igniter

CodeIgniter merupakan sebuah framework PHP yang sederhana serta mudah untuk dipelajari. Selain itu codeigniter merupakan framework php yang menggunakan konsep desain MVC yaitu Model, View, Controller.

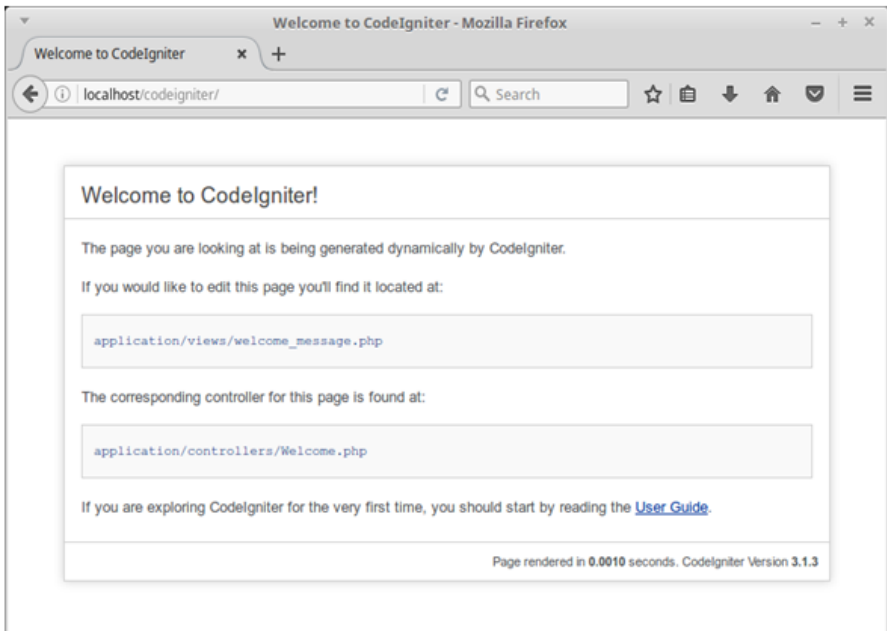
3.1.1 Cara Instalasi Code Ignitor

- Buka web Codeigniter



Gambar 3.1 Web Code Ignitor

- Lalu Download Codeigniter.
- Kemudian simpan Codeigniter pada file htdocs di xampp
- Selanjutnya lakukan ekstrak pada file tersebut.
- Aktifkan Xamppnya, lalu buka browser dan ketikan `http://localhost/codeigniter`
- Maka akan muncul tampilan seperti dibawah. CodeIgniter telah berhasil melakukan instalasi.



Gambar 3.2 Hasil

3.2 XAMPP

Xampp merupakan sebuah perangkat lunak bebas, yang mendukung banyak sistem operasi serta kompilasi dari beberapa program. Selain itu xampp terdiri dari satu paket aplikasi lengkap yang dibutuhkan untuk membuah sebuah web server, serta di dalamnya terdapat Apache, Mysql,dan Php.

3.2.1 Kelebihan XAMPP

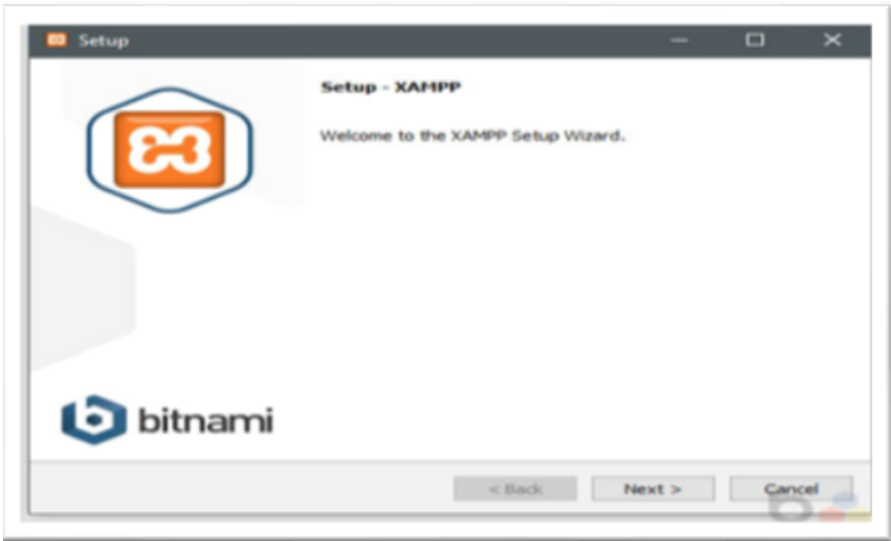
1. Database storage engine banyak digunakan oleh programmer web developer dikarenakan sifatnya yang bebas.
2. Xampp memiliki kapasitas yan cukup sekitar 60.000 tabel dengan record mencapai sekitar 5.000.000.000 bahkan xampp yang terbaru sudah lebih jumlah recordnya.
3. Pada xampp terdapat keamanan datanya cukup aman meskipun tidak sebanding dengan postgre dan oracle.
4. Engine pada xampp ini multiplatform sehingga mampu diimplementasikan di berbagai system operasi
5. Xampp mempunyai kelebihan utama yaitu kecepatannya.

3.2.2 Kekurangan XAMPP

1. Xampp tidak cocok dalam menangani data dengan jumlah yang sangat besar, baik untuk menyimpan data ataupun untuk memproses data.
2. Serta memiliki keterbatasan pada kemampuan kinerja pada server ketika suatu data disimpan telah melebihi batas maksimal, maka kemampuan daya tamping server tidak dapat menerapkan konsep Technology Cluste.

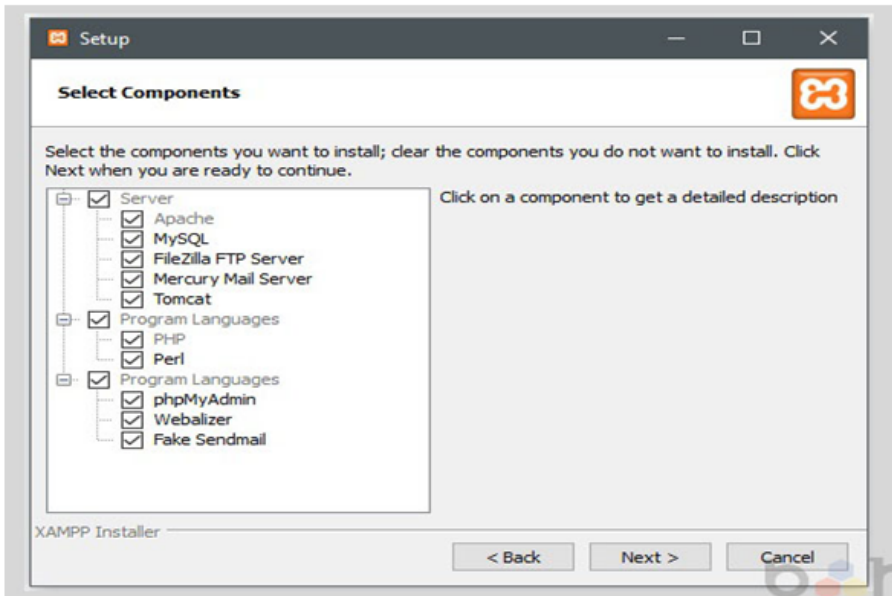
3.2.3 Cara Instalasi XAMPP di Windows

1. Download terlebih dahulu xampp nya, sesuai dengan windows yang dimiliki.
2. Setelah itu simpan data xampp di data C, lalu double klik pada mentahan xampp, seperti gambar dibawah.
3. Kemudian akan muncul splash screen bitnami, setelah itu akan ada pesan eror, itu terjadi karena masalah permission atau hak akses, lalu klik ok karena akan dikonfigurasi ke hard drive lain.
4. Setelah itu akan muncul bagian intaslasi dari Setup Wizard Xampp, seperti gambar dibawah ini.



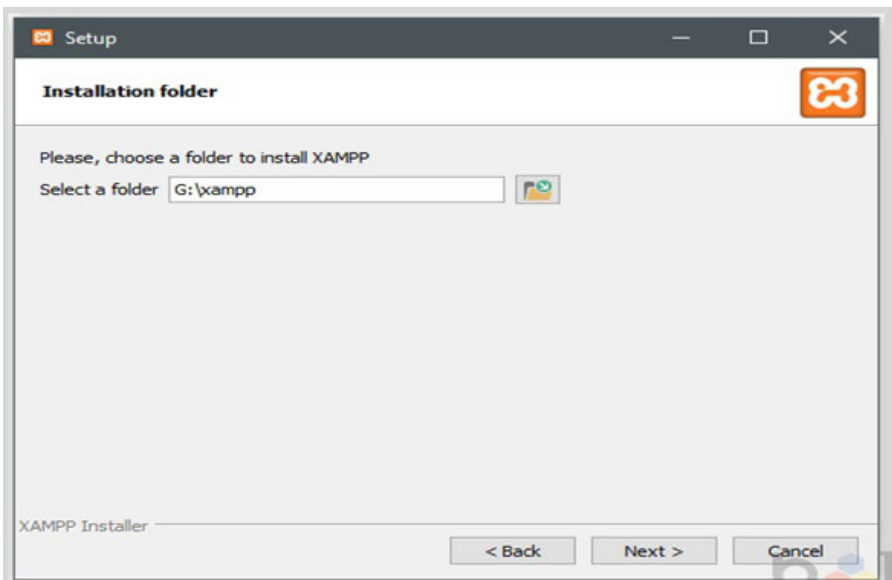
Gambar 3.3 XAMPP

5. Kemudian akan muncul halaman pemilihan komponen apa saja yang ingin diinstal pada web server xampp ini, lalu masuk dan sesuaikan modul-modul instalasi sesuai kebutuhan. Jika sudah sesuai, klik Next saja.



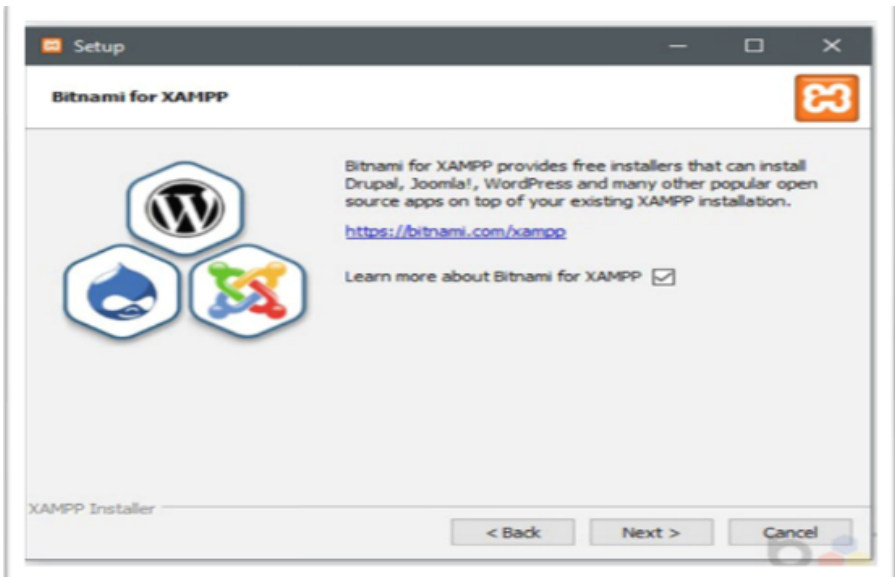
Gambar 3.4 XAMPP

6. selanjutnya akan diarahkan untuk penyimpanan folder instalasi xampp, jika sudah bawaan maka akan langsung diarahkan ke drive C, lalu klik Next saja



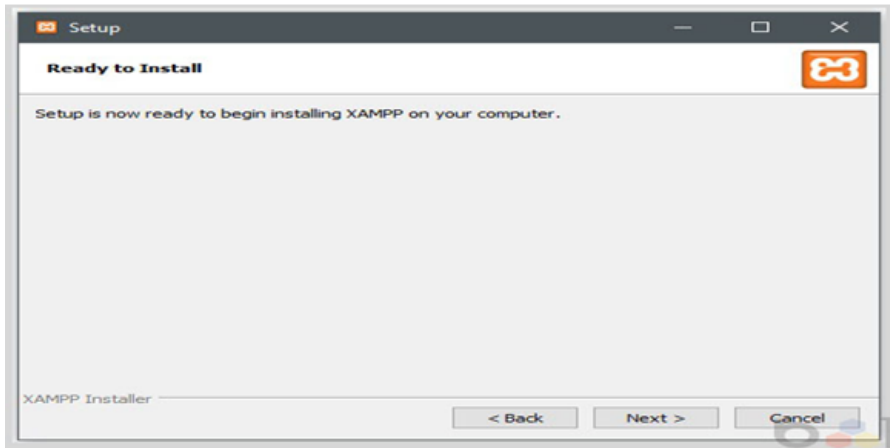
Gambar 3.5 XAMPP

7. Langkah berikutnya merupakan tampilan dari promosi bitnami. Pada langkah ini terdapat sebuah checkbox pada bagian Learn More about Bitnami For Xampp, hilangkan ceklis yang sudah ada sebelumnya, jika tidak ingin mengikuti petunjuk instalasi CMS. Jika sudah klik Next saja.



Gambar 3.6 XAMPP

8. Pada langkah selanjutnya Ready to Install ini untuk memastikan, apakah tahap yang sudah dilakukan sudah benar atau tidak, jika merasa masih ada yang kurang dapat melakukan back. Jika sudah sesuai klik Next saja.



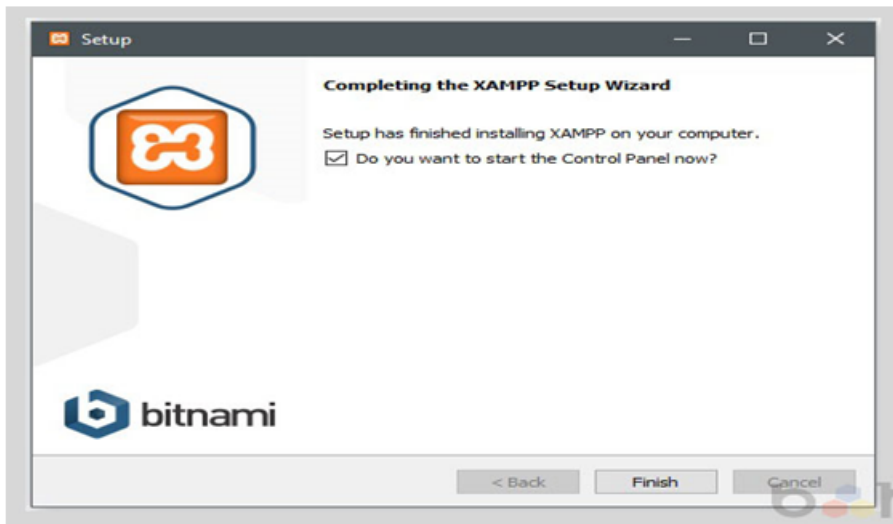
Gambar 3.7 XAMPP

9. Selanjutnya melakukan proses file-file xampp ke dalam hard drive folder yang telah ditetapkan sebelumnya. Jika sudah instalasinya klik Next saja.



Gambar 3.8 XAMPP

10. Langkah berikutnya ke bagian Completing the Xampp Setup Wizard dimana ini langkah terakhir pada instalasi xampp. Jika sudah langsung klik finish.



Gambar 3.9 XAMPP

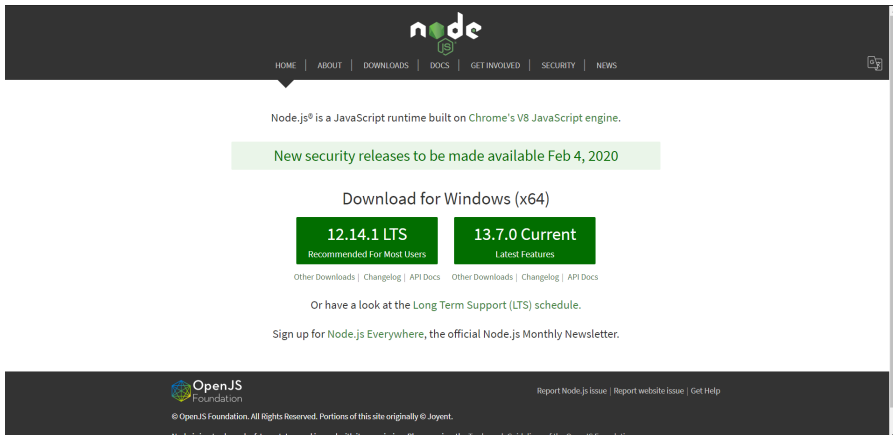
11. Jika sudah silahkan klik start pada Apache dan Mysql hingga tulisan Apache dan Mysql di block warna hijau.

3.3 Node.js

Node.js adalah environment berbasis JavaScript yang dapat Anda gunakan untuk membuat server web dan aplikasi jaringan. Anda juga dapat menggunakannya untuk melakukan tugas-tugas bermanfaat di komputer Anda seperti menyatukan dan memperkecil file JavaScript dan mengkompilasi file Sass ke dalam CSS.

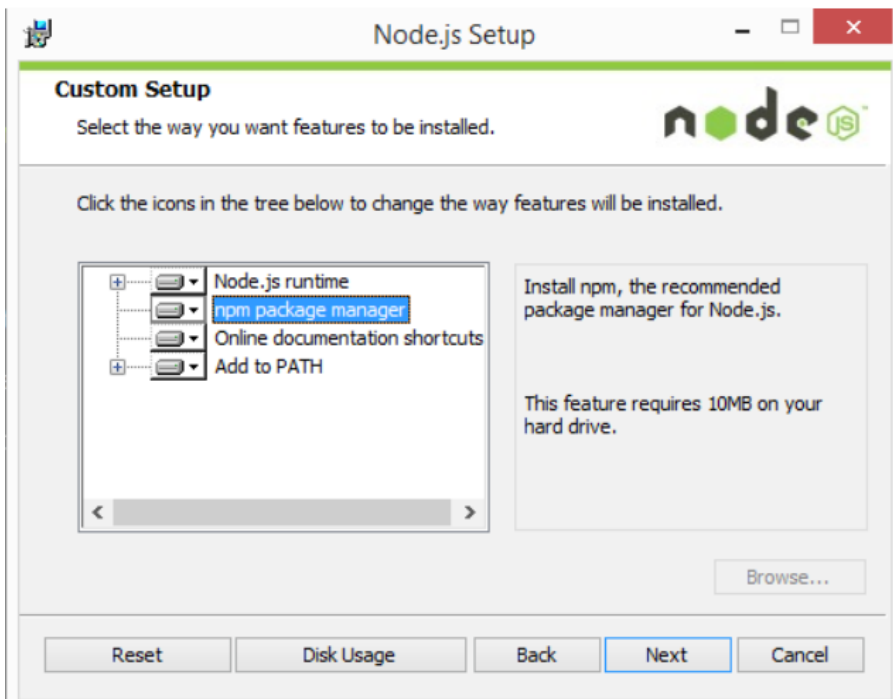
3.3.1 Cara Instalasi Node.js di windows

1. Download installer untuk windows di nodejs.org



Gambar 3.10 Node.js

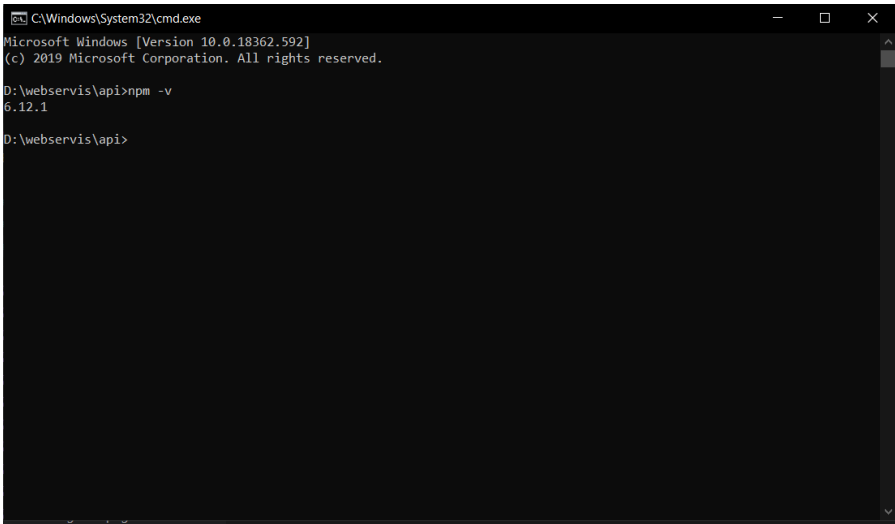
2. Jalankan installer tersebut
3. Ikuti langkah - langkah installer tersebut



Gambar 3.11 Node.js

4. Lalu restart PC anda

5. lalu test dengan menggunakan CMD dengan mengetikkan `npm -v`

A screenshot of a Windows Command Prompt window. The title bar shows 'C:\Windows\System32\cmd.exe'. The window content displays the following text:

```
Microsoft Windows [Version 10.0.18362.592]  
(c) 2019 Microsoft Corporation. All rights reserved.  
  
D:\webservis\api>npm -v  
6.12.1  
  
D:\webservis\api>
```

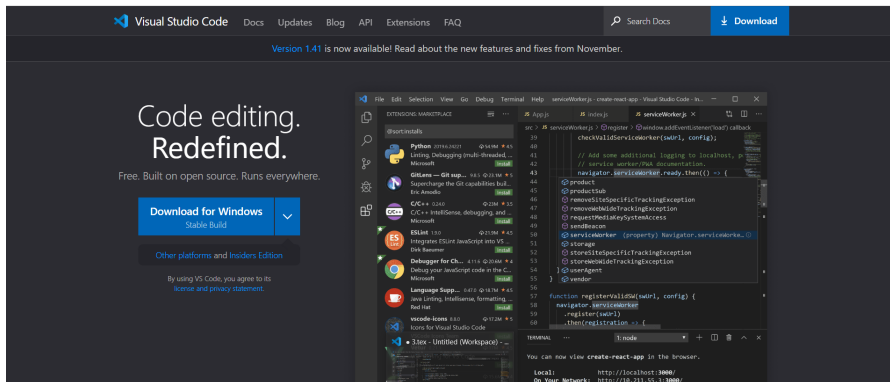
Gambar 3.12 Node js

3.4 VS Code

Visual Studio Code atau VS Code adalah teks editor open-source berbasis Electron yang diracik oleh Microsoft.

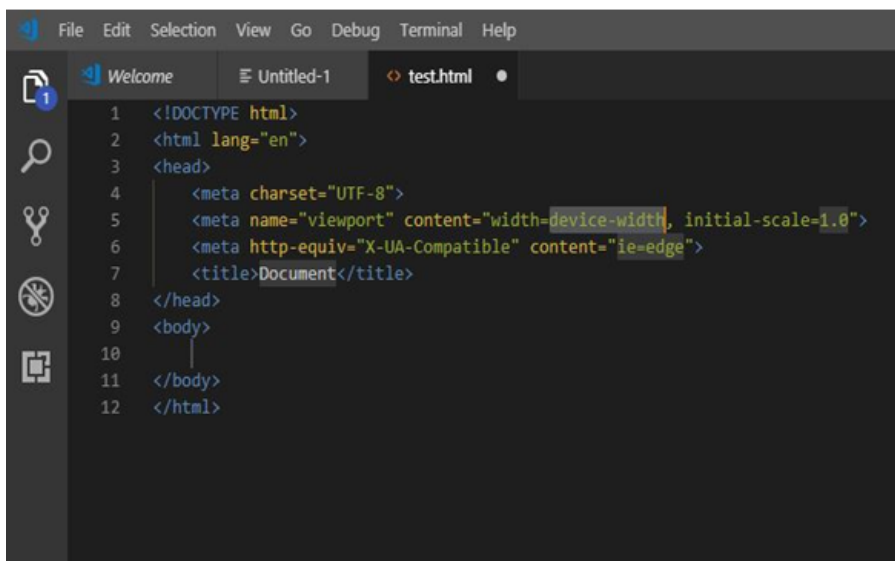
3.4.1 Keuntungan memakai VS Code

- VS Code gratis



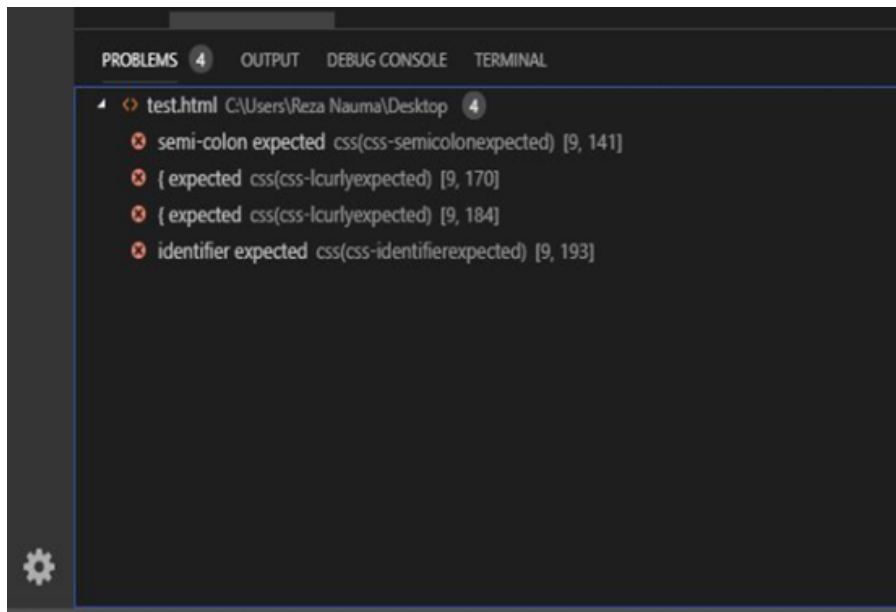
Gambar 3.13 VS Code

- Terinstall otomatis fungsi EMMET



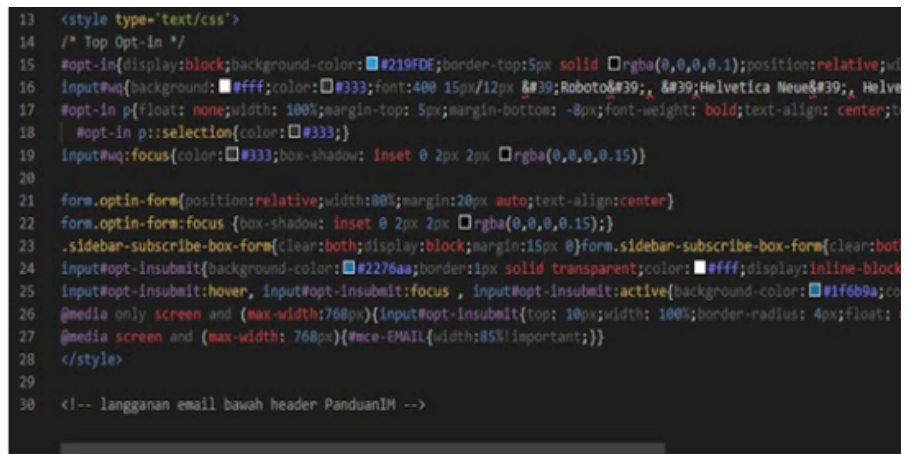
Gambar 3.14 VS Code

- Terinstall otomatis fungsi LINTER dan INTELLISENSE



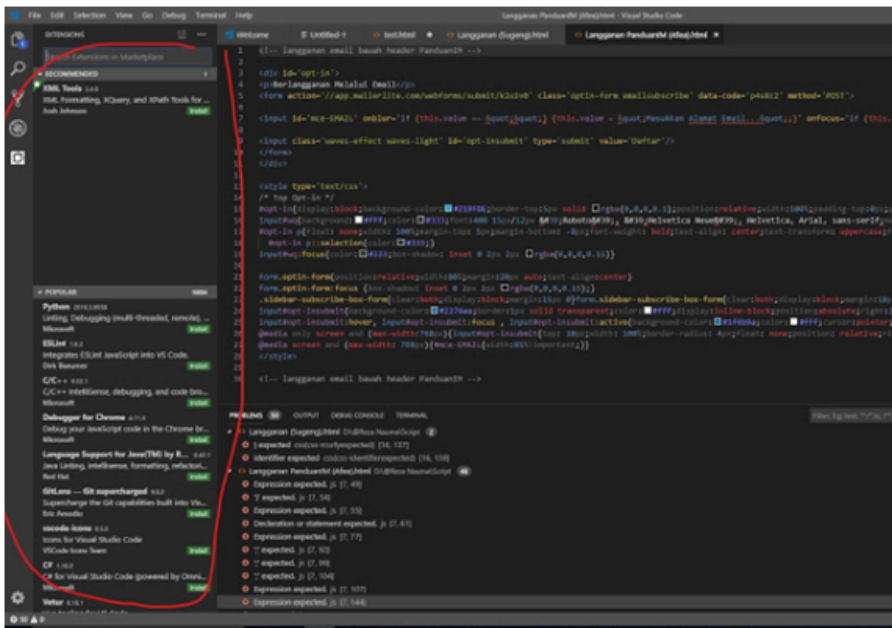
Gambar 3.15 VS Code

- Kemudahan Bekerja dengan Warna



Gambar 3.16 VS Code

- Memiliki sidebar kaya akan fitur



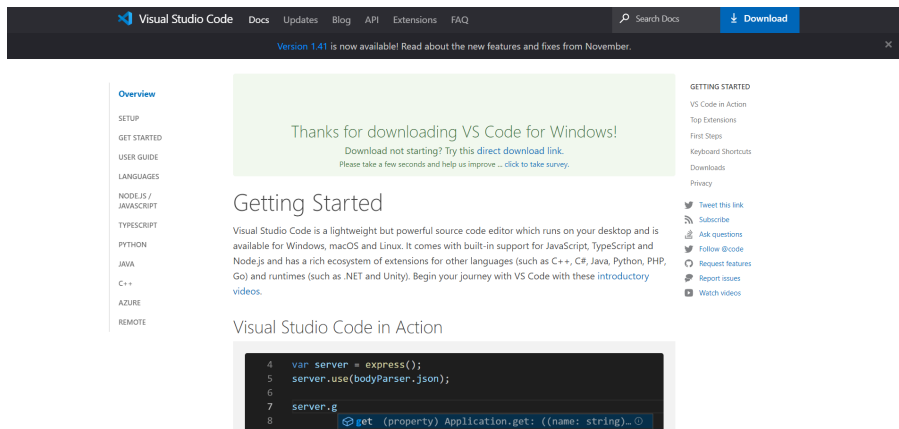
Gambar 3.17 VS Code

3.4.2 Kerugian menggunakan VS Code

Satu-satunya kelemahan dari Visual Studio Code itu di performa. Visual Studio Code ini lebih berat. Apalagi bagi kamu pengguna OS Windows dengan RAM 4GB ke-bawah. Software Visual Studio Code ini sepertinya akan optimal jika dijalankan dengan spesifikasi hardware minimal CPU 2 core / 4 thread, 4GB+ RAM.

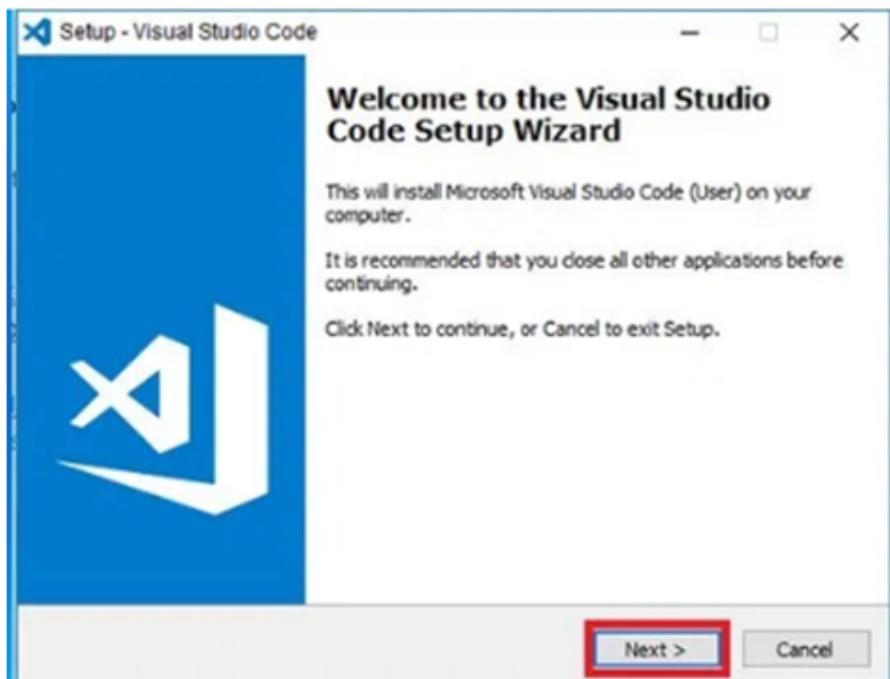
3.4.3 Cara instalasi VS Code di Windows

1. Download file instalasi VS Code



Gambar 3.18 VS Code

2. Apabila telah di download, buka file instalasinya dan tekan next.



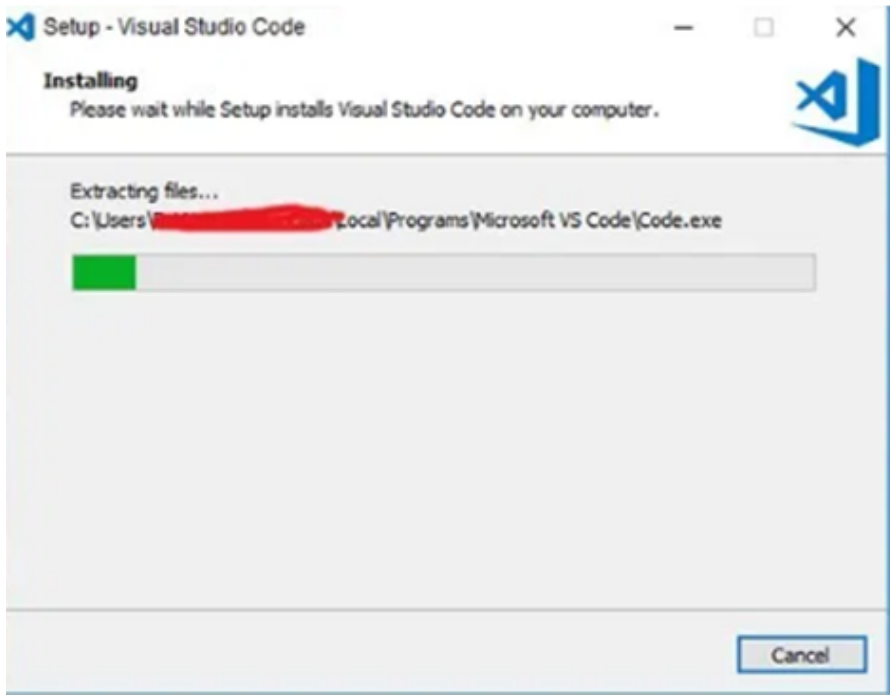
Gambar 3.19 VS Code

3. Pada form persetujuan Lisensi, tekan I accept the agreement kemudian tekan next.



Gambar 3.20 VS Code

4. Selanjutnya tekan next hingga instalasi selesai, lalu tekan finish.



Gambar 3.21 VS Code

3.5 Teori

3.5.1 Fungsi

Fungsi adalah sebuah blok kode yang memiliki nama fungsi dan kode program di dalamnya jika dijalankan maka fungsi itu akan mengembalikan nilai. Fungsi dapat dipanggil berkali-kali sesuai dengan nama fungsi yang telah didefinisikan. Fungsi memiliki nilai kembalian (return). Contoh fungsi

```
1 def nambahinAngka(angka1, angka2):  
2     hasil = angka1 + angka2  
3     return hasil
```

Apabila kita dapat memberikan nilai ke `angka1` dan `angka2`, dan apa bila sudah diberi nilai dan program sudah dijalankan, maka program pun akan mengembalikan nilai berupa hasil dari penjumlahan angka 1 dan angka 2.

3.6 Package

Package merupakan sekumpulan modul yang dikemas oleh programmer dengan tujuan agar mempermudah dalam pembuatan kode program. Kita dapat membuat se-

buah kode program atau fungsi didalamnya dan dapat secara mudah menggunakan kode program itu dengan cara memanggilnya pada kode program lainnya atau import package. Contoh nya adalah sebagai berikut

```
1 def my_biodata(nama, umur):
2     bio = "nama saya " + nama + " umur saya " + umur
3     return bio
4
5 def my_study(kampus, prodi):
6     study = "saya kuliah di " + kampus + " program studi " + prodi
7     return study
```

Kode diatas merupakan isi dari le fungsi.py, sedangkan saya ingin menjalankan program fungsi.py pada main.py sehingga kode program pada le main.py akan dituliskan seperti berikut:

```
1 import fungsi
2 nama = "Dinda Majesty"
3 umur = "19 Tahun"
4 biodata = my_biodata(nama, umur)
5 print(biodata)
6
7 kampus = "Politeknik Pos Indonesia"
8 prodi = "D4-Teknik Informatika"
9 kuliah = my_study(kampus, prodi)
10 print(kuliah)
```

Kode program pada le main.py akan mengimport kode program yang ada pada le fungsi.py, sehingga dengan adanya fungsi dan package kita dapat dengan mudah melakukan pemanggilan fungsi yang telah kita deskripsikan sebelumnya, walaupun berada pada le python yang berbeda.

3.7 Class, Object, Attribute, and Method

Class atau Kelas merupakan sebuah blueprint/kerangka dari objek yang berisi fungsi dan dibuat untuk mendefenisikan objek dengan atribut yang sesuai dengan kelas yang telah dibuat yang nantinya akan diinisiasikan. Objek adalah sebuah wujud yang dapat kita lakukan perintah sesuai dengan methodnya, Sebuah kelas harus memiliki objek yang nantinya akan di kodekan sesuai dengan fungsi yang telah dibuat pada kelas, tanpa adanya objek sebuah kelas tidak akan bisa menjalankan fungsi-fungsi didalamnya. Atribut berisi variabel yang memiliki tipe data dan dapat kita berikan pada objek, atribut ada 2 yaitu kelas atribut dan instansi atribut, perbedaannya hanya di letak, kalau kelas atribut ada di bawah kelas, dan instansi atribut ada didalam fungsi, atribut itu sebuah variabel yang dimiliki oleh parentnya seperti fungsi atau class. .Method merupakan kode program yang berisi tindakan atau perintah untuk menjalankan objek.

```
1 class Fungsi(object):
2
3     def Nama(self, namakamu):
4         self.kamu = namakamu
```

3.8 Pemanggilan Class

Pemanggilan library kelas dapat dilakukan dengan cara import dan membuat objek dari kelas tersebut. Contohnya, kita memiliki le python yang diberi nama ngitung dan didalamnya terdapat class Ngitung yang memiliki banyak fungsi didalamnya. Untuk melakukan pemanggilan class maka kita bisa mengetikkan kode seperti berikut.

```
1 import Fungsi
```

3.9 Pemakaian Package Fungsi Apabila File Didalam Folder

Pemakaian Package fungsi apabila le terdapat didalam sebuah folder maka kita bisa menggunakan from folder import le dan from le import fungsi. Contohnya, kita memiliki folder src yang didalamnya terdapat le fungsi.py dan didalam fungsi.py terdapat fungsi Berhitung, untuk mengimportkan fungsi maka kita dapat mengetikkan kode seperti berikut.

```
1 from src import fungsi
2 from fungsi import Berhitung
```

3.10 Pemakaian Package Kelas Apabila File didalam Folder

Pemakaian package kelas apabila le terdapat didalam sebuah folder maka kita bisa menggunakan from folder import le dan from le import kelas. Contohnya, kita memiliki folder src yang didalamnya terdapat le fungsi.py dan didalam fungsi.py terdapat kelas Ngitung, maka untuk melakukan import kelas kita dapat mengetikkan kode sebagai berikut.

```
1 from src import fungsi
2 Kelas = fungsi.Nama(namakamu)
```

DAFTAR PUSTAKA

1. R. Awangga, "Sampeu: Servicing web map tile service over web map service to increase computation performance," in *IOP Conference Series: Earth and Environmental Science*, vol. 145, no. 1. IOP Publishing, 2018, p. 012057.

