



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pembangkit Listrik Tenaga Mikrohidro (PLTMH)

Pembangkit listrik tenaga mikro hidro mengacu pada pembangkit berskala kecil dengan *ouput* di bawah 100 KW yang memanfaatkan potensi aliran air yang terdapat di pendesaan sebagai sumber tenaga, misalnya saluran irigasi, sungai atau air terjun. PLTMH memiliki konstruksi yang sederhana, mudah dioperasikan, mudah dalam perawatan, serta dengan biaya investasi yang terjangkau sehingga cocok diterapkan untuk menerangi wilayah pendesaan yang tidak terjangkau aliran listrik PLN. Secara teknis PLTMH memiliki 3 (tiga) komponen utama yaitu air (sebagai sumber energi) turbin dan generator. PLTMH mendapatkan energi dari aliran air yang memiliki perbedaan ketinggian tertentu. Pada dasarnya PLTMH memanfaatkan energi potensial jatuhan air (*head*). Semakin tinggi jatuhan air maka semakin besar energi potensial air dapat diubah menjadi energi listrik. (www.kajianpustaka.com/2016/10/pembangkit-listrik-tenaga-mikrdro.html, akses 12 juli 2017).

2.1.2 Teknologi

Perkembangan teknologi masa kini berkembang sangat pesat. Teknologi sudah mejadi bagian tidak terpisah dalam kehidupan manusia sehari-hari, sehingga teknologi menjadi suatu penunjang perkembangan manusia di belahan masyarakat, teknologi telah membantu memperbaiki ekonomi, pangan dan masih banyak lainnya. Hal ini dapat dibuktikan dengan banyaknya inovasi-inovasi yang telah dibuat di dunia ini, dari yang sederhana hingga yang menghebohkan dunia.

Perkembangan teknologi dalam teknologi sistem informasi memberikan kemudahan dalam menyelesaikan suatu pekerjaan bagi suatu organisasi atau



perusahaan, sehingga pekerjaan-pekerjaan dapat diselesaikan dengan mudah dan cepat.

Jogiyanto (2009:7), Teknologi yang digunakan di sistem teknologi informasi adalah teknologi komputer, teknologi telekomunikasi dan teknologi apapun yang dapat memberikan nilai tambah untuk organisasi atau perusahaan.

Selain itu juga Jogiyanto (2009:51-52), menjelaskan bahwa teknologi merupakan komponen yang di sistem informasi. Tanpa adanya teknologi yang mendukung, maka sistem informasi tidak akan dapat menghasilkan informasi yang tepat waktunya. Komponen teknologi mempercepat sistem informasi dalam pengolahan datanya. Komponen teknologi dapat dikelompokkan kedalam dua macam kategori, yaitu teknologi sistem komputer (perangkat keras dan perangkat lunak) dan teknologi sistem telekomunikasi.

2.1.3 Pengertian *Internet*

Febrian (2006:2) menjelaskan *internet* merupakan terhubungnya berbagai mesin komputer yang mengolah informasi di dunia ini, baik berupa server, komputer pribadi, handphone, komputer genggam, PDA, dan lain sebagainya.

Sujatmiko (2012:138), menjelaskan *Internet* adalah jaringan global yang menghubungkan berjuta-juta komputer di seluruh dunia melalui jalur telepon kabel maupun satelit. Ada juga yang menjelaskan dari Yugianto dan Rachman (2012:36), Internet adalah suatu sistem jaringan komunikasi beberapa komputer yang terhubung tanpa batas waktu maupun tempat, sehingga dapat dikatakan sebagai komunitas jaringan global.

Dengan demikian *internet* merupakan jaringan komunikasi yang digunakan melalui kabel telepon atau satelit secara global yang dapat terhubung di berbagai komputer tanpa ada batas waktu maupun tempat



2.1.4 Pelayanan Operasional

Pelayanan adalah kemudahan yang diberikan sehubungan dengan jual beli barang/jasa atau usaha melayani kebutuhan orang lain dengan memperoleh imbalan uang/jasa (KBBI.Web.id.layan, diakses pada tanggal 11 juni 2017).

Kamus Besar Bahasa Indonesia (2015:394) menjelaskan “Operasional adalah sesuatu yang berhubungan dengan proses kerja”.

Dengan demikian dapat disimpulkan pelayanan operasional merupakan suatu proses dalam melayani kebutuhan orang lain berupa jual beli barang atau jasa yang berhubungan dengan proses kerja sehingga dapat memperoleh imbalan uang atau jasa.

2.1.5 Smartphone

Menurut Williams & Sawyer (2011), *smartphone* merupakan ponsel multimedia yang menggabungkan fungsionalitas PC dan handset, sehingga menghasilkan gadget yang mewah, dimana terdapat teks, kamera, pemutar musik, video, game, akses email, tv digital, *search engine*, pengelola informasi pribadi, fitur GPS, jasa telpon internet dan bahkan terdapat telepon yang juga berfungsi sebagai kartu kredit.

Smartphone adalah sebuah telpon genggam yang menyajikan fitur canggih seperti surat elektronik, *internet* dan kemampuan membaca buku elektronik (*e-book*). Dengan kata lain *smartphone* merupakan komputer kecil yang mempunyai kemampuan sebuah telepon yang mempunyai daya guna bagi manusia (<http://www.duniabombom.com/2014/08/pengertian-smartphonen.html>, diakses 15 juni 2017).

2.1.6 Perangkat Lunak

A.S. Rosa & M. Shalahuddin (2014:2), menjelaskan perangkat lunak (*software*) merupakan program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara



penggunaan. Sebuah program komputer tanpa terasosiasi dengan dokumentasinya maka belum dapat disebut dengan perangkat lunak.

Sukanto & M. Shalahuddin (2013:2) *Software* atau perangkat lunak adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain dan cara penggunaan (user manual). Adapula pula yang menjelaskan perangkat lunak adalah kumpulan program-program komputer yang memungkinkan *hardware* memproses data (Sutarman, 2012:14).

Karakter perangkat lunak menurut A.S. Rosa & M. Shalahuddin (2014:2) adalah sebagai berikut :

1. Perangkat lunak dibangun dengan rekayasa (*software engineering*) bukan di produksi secara *manufaktur* atau pabrikan
2. Perangkat lunak tidak pernah usang (*wear out*) karena kecacatan dalam perangkat lunak dapat diperbaiki
3. Barang produksi biasanya komponen barunya akan terus diproduksi, sedangkan perangkat lunak biasanya terus diperbaiki seiring bertambahnya kebutuhan.

Menurut A.S. Rosa & M. Shalahuddin (2014:4) produk perangkat lunak yang dibuat oleh *developer* perangkat lunak terdiri dari dua jenis:

1. Produk Generik
Produk perangkat lunak yang dibuat oleh pengembang perangkat lunak untuk dijual atau dipopulerkan tanpa yang memesan terlebih dahulu.
2. Produk pemesanan
Produk perangkat lunak yang dibuat karena ada pelanggan yang melakukan pemesanan.

2.1.7 Pengertian Aplikasi

Aplikasi salah satu bagian dalam teknologi informasi yang digunakan untuk meningkatkan kualitas pekerjaan. Dalam suatu perusahaan atau organisasi sangat



membutuhkan sebuah aplikasi untuk mempermudah dan mempercepat pekerjaannya. Aplikasi yang dibangun harus sesuai dengan manajemen dalam perusahaan atau organisasi tersebut.

Astropoddin (2013:7) menjelaskan aplikasi dibuat untuk mengerjakan atau menyelesaikan masalah-masalah khusus. Ada juga menurut Sutabri (2012:147), Aplikasi adalah alat terapan yang fungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya. Kemudian menurut Sujatmiko (2012:23), Aplikasi adalah program komputer yang dibuat oleh perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu, misalnya *Ms-word*, *Ms-Excel*

2.1.8 Aplikasi *Mobile*

Aplikasi *mobile* dapat dengan mudah melakukan berbagai macam aktifitas mulai dari hiburan, belajar, mengerjakan pekerjaan kantor, *browsing* dan lain sebagainya.

Suryana (2008), menjelaskan “Aplikasi *mobile* merupakan aplikasi yang cara aksesnya menggunakan perangkat gerak (*mobile device*), seperti *handphone*, *smartphone* dan *Pdphone*”.

Aplikasi *mobile* adalah sebuah *software* yang di buat dan di teruntukan perangkat *portable smartphone* yang mengharuskan proses *men-download software mobile* aplikasi di toko aplikasi agar dapat di gunakan. sedangkan untuk jenis toko aplikasinya pun bervariasi seperti *Apple app store*, *Play store*, atau *blacbary app*. namun saat ini pasar toko aplikasi yang mampu menguasai pasar aplikasi adalah *google playstore* atau *playstore* sehingga yang ingin menggunakan aplikasi atau mempublikasikan sebuah aplikasi maka dengan mudah melihat dan *men-download* di *playstore*, yang di dalamnya ada banyak aplikasi yang di tawarkan dari mulai aplikasi toko online seperti *lazada*, *tokopedia*, *shope* dan aplikasi aplikasi lainnya. (<https://idprogrammer.com/pengertian-mobile-web-dan-mobile-aplikasi/>, diakses 27 juni 2017).



2.2. Teori Khusus

2.2.1. Pemrograman Berorientasi Objek (PBO)

Sukanto dan Shalahuddin (2016:99), menjelaskan Pemrograman berorientasi obyek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan obyek yang berisi data dan operasi yang diberlakukan terhadapnya.

Menurut Nugroho (2004:12) , ada beberapa karakter yang menjadi ciri-ciri pendekatan berorientasi objek adalah sebagai berikut:

1. Pendekatan lebih pada data dan bukannya pada prosedur atau fungsi
2. Program besar dibagi pada apa yang dinamakan objek-objek
3. Struktur data dirancang dan menjadi karakteristik dari objek-objek.
4. Fungsi- fungsi yang mengoperasikan data tergabung dalam suatu objek yang sama
5. Data tersembunyi dan terlindung dari fungsi atau prosedur yang ada diluar.
6. Objek-objek dapat saling berkomunikasi dengan saling mengirim *message* satu sama lain
7. Pendekatan adalah dari bawah ke atas.

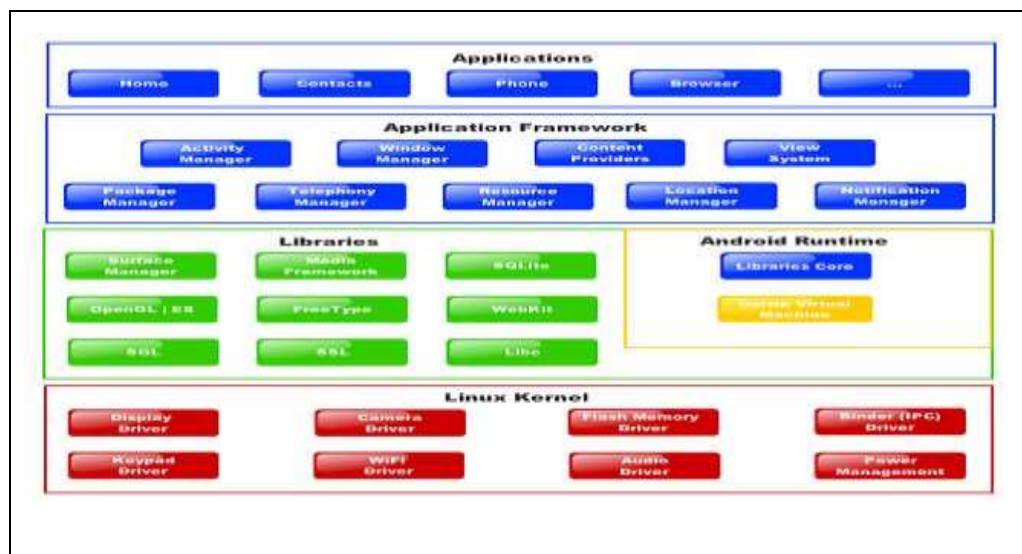
2.2.2 Pengertian *Android*

Perkembangan teknologi semakin canggih, salah satu perkembangan teknologi yang lagi populer pada masa kini adalah *android*. *Android* bukan lagi barang tabu bagi manusia, tetapi *android* sudah menjadi bagian dalam kehidupan manusia pada masa kini. *Android* memberikan kecanggihan dengan *fitur-fitur* yang canggih sehingga dapat mempermudah segala aktivitas manusia. Seperti memberikan *fitur game*, *music player*, sampai *vedio player* sehingga menjadi lebih mudah menikmati hiburan kapan saja dan dimanapun.

Nazrudin Sfaat (2011) menjelaskan “ *Android* adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware*, dan aplikasi”. Ada juga pendapat lain dari Jubilee Enterprise



(2015:1), mengatakan “ *Android* merupakan sistem operasi *open source*, dan *google* merilis kodenya dibawah lisensi *apache*. Kode *open source* dan lisensi perizinan pada *android* memungkinkan perangkat lunak untuk dimodifikasi secara bebas dan didistribusikan oleh para pembuat perangkat, operator nirkabel dan pengembang aplikasi. *Android* memiliki arsitektur, dapat ditampilkan pada gambar 2.1



(Sumber :Nasaruddin, 2011)

Gambar 2.1 Arsitektur Android

- Applications:** *Android* akan menggabungkan dengan serangkaian aplikasi inti termasuk *client email*, program SMS, kalender, peta, *browser*, kontak, dan lain-lain.
- Application Framework:** Dengan menyediakan sebuah *platform* pengembangan yang terbuka, pengembang *android* menawarkan kemampuan untuk membangun aplikasi yang sangat kaya dan inovatif.
- Libraries:** *Android* termasuk satu set pustaka C/C++ yang digunakan oleh berbagai komponen sistem *android*.
- Android Runtime:** *Android* termasuk satu set perpustakaan inti yang menyediakan sebagian besar fungsi yang tersedia di perpustakaan inti dari bahasa pemrograman Java.



- e. *Linux Kernel: Android* bergantung pada *Linux* versi 2.6 untuk layanan sistem inti seperti keamanan, manajemen memori, manajemen proses, *network stack*, dan model pengemudi. Kernel juga bertindak sebagai lapisan abstraksi antara *hardware* dan seluruh *software stack*.

2.2.3. Fitur-fitur Android

Android memiliki sejumlah besar komunitas pengembang aplikasi yang memperluas fungsionalitas perangkat, umumnya ditulis dalam versi kustomisasi bahasa pemrograman java. Berikut ini *fitur-fitur android* :

Tabel 2.1 *Fitur-fitur Android*

Antarmuka	Layar sistem operasi <i>android</i> menampilkan antarmuka yang bagus dan intuitif.
Konektifitas	GSM/EDGE, IDEN,CDMA,EV-DO,UMTS,Bluetooth, WiFi, LTE, NFC, dan WiMAX.
Penyimpanan	<i>SQLite</i> , <i>database</i> relasional ringan yang digunakan untuk penyimpanan data
<i>Messaging</i>	SMS dan MMS
Media	H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC,AAC 5.1, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, dan BMP
<i>Web browser</i>	Berdasarkan <i>layout engine WebKit</i> , dengan <i>Chrome's V8 JavaScript engine</i> yang mendukung HTML5 dan CSS3
<i>Multi-touch</i>	<i>Android</i> memiliki dukungan <i>multi-touch</i>
<i>Multi-tasking</i>	Pengguna dapat berpindah dari satu <i>task</i> ke <i>task</i> lainnya dan beberapa aplikasi dapat berjalan pada waktu yang bersamaan
Ukuran-widgets	<i>Widgets</i> dapat diatur ukurannya sesuai dengan keinginan pengguna, baik memperbesar ukuran untuk menampilkan lebih banyak konten, atau memperkecil ukuran untuk menyediakan lebih banyak ruang.
<i>Multi-language</i>	Mendukung teks dalam berbagai bahasa
GCM	<i>Google Cloud Messaging</i> (GCM) adalah layanan yang memungkinkan <i>developer</i> untuk mengirim data pesan pendek kepada pengguna tanpa menggunakan solusi <i>sync proprietary</i>
<i>Wi-fi Direct</i>	Teknologi yang memungkinkan aplikasi untuk menemukan dan memasang secara langsung, melalui koneksi <i>peer-to-peer-high-bandwidth</i>
<i>Android beam</i>	Teknologi berbasis NFC populer yang memungkinkan pengguna untuk berbagi konten secara langsung hanya dengan menyentuh dua perangkat

(Sumber : *Jubilee Enterprise* (2015:1))



2.2.4 Web Server

Nugroho (2004:5), Secara global server dapat diartikan sebagai pusat dan berfungsi sebagai “pelayan” yang berguna untuk pengiriman data dan/atau penerimaan data serta mengatur pengiriman dan penerimaan data di antara komputer-komputer yang tersambung atau dengan kata lain server berfungsi menyediakan pelayanan terhadap klien.

Nugroho (2004:6), Menjelaskan *Web server* adalah sebuah bentuk server yang khusus digunakan untuk menyimpan halaman *web site* atau *home page*.

Secara umum *Web server* terbagi menjadi dua tipe yaitu *web server* secara *offline* dan *web server* secara *online*. *Web server* ini dapat juga diartikan sebagai rumahnya *website* atau situs. Setiap situs yang ada di internet pastilah berjalan diatas *web server*. Jadi peranan *web server* ini sangatlah penting dalam pembuatan situs.

2.2.5. Perancangan Perangkat Lunak

Rosa A.S dan M. Salahudin (2014:4), menjelaskan Perancangan perangkat lunak merupakan pembangunan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomi yang dipercaya dan bekerja secara efisien menggunakan mesin.

Adapun pendapat lain dari Jogianto (2005:211), tahap desain adalah perancangan komponen-komponen sistem informasi dengan tujuan untuk dikomunikasikan dengan *user* bukan pemograman.

2.2.6. Metode *Disciplined Agile Delivery* (DAD)

Metode yang digunakan dalam perancangan perangkat lunak ini yaitu menggunakan metode *Disciplined Agile Delivery* (DAD).

Ambles and Lines (2012), Kerangka kerja DAD yang menitik beratkan pada beberapa karakteristik seperti *People First, learning-oriented hybrid agile*



approach to IT solution delivery. DAD memiliki daur hidup berbasis *risk-value*, *goal-driven*, *scalable* dan *enterprise-aware*.

IMB Corporation (2011), Proses kerja DAD adalah metedeologi pengembangan *Hybird*, dalam pengertian merupakan penggabungan dari beberapa model proses perangkat lunak seperti *Scrum*, *XP*, *Lean*, *Agile Modeling*, *Harmony Proces* dan *Unified Proces*. Singkatnya DAD menampung setiap *best practices* dari semua model proses berkarakteristik *agile* yang telah ada sebelumnya, sehingga DAD memiliki tiga tahap dalam pengembangan sistem yaitu:

1. Tahap *Inception*

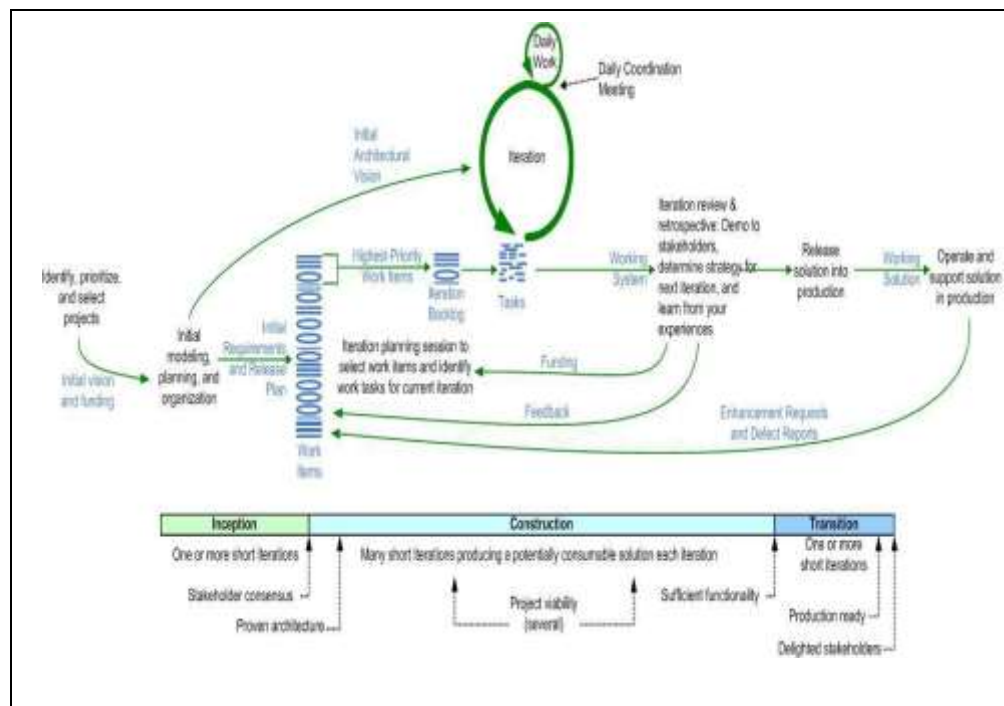
Tahap ini membentuk sebuah lingkup proyek (*project scope*) atau tim, mengidentifikasi rencana awal dalam proyek yang akan dijalankan, mempersiapkan lingkungan pengerjaan proyek serta mengestimasi biaya dan penjadwalan. Proses iterasi dilakukan satu kali, dan artifak yang dihasilkan adalah dokumen *vision*, *software project plan*.

2. Tahap *Construction*

Tahap aktivitas memodelkan dan membangun sistem aplikasi. Proses iterasi dilakukan dua hingga delapan kali. Artifak yang dihasilkan adalah arsitektur sistem (*snapshot*), *use case (snapshot)*, *source code*, *code documention*.

3. Tahap *Transition*.

Tahap ini adalah tahap aktivitas sistem (*intergration* sistem dan *user tasting*), me- *review* kembali sistem aplikasi dan menginstalasikan sistem aplikasi. Proses iterasi dapat dilakukan satu hingga dua kali. Artifak yang akan dihasilkan adalah dokumen spesifikasi sistem dan testing aplikasi



Sumber : IBM Corporation (2012)

Gambar 2.2 Metode *Disciplined Agile Delivery* (DAD)

2.2.7 Black Box Testing

Menurut Pressman (2012:587,597), *black box testing* berkaitan dengan pengujian-pengujian yang dilakukan pada antarmuka perangkat lunak. Pengujian kontak hitam mengkaji beberapa aspek fundamental dari suatu sistem dengan sedikit memperhatikan struktur logis mental dari perangkat lunak. Pengujian kontak hitam disebut pengujian perilaku, berfokus pada persyaratan fungsional perangkat lunak. Artinya teknik pengujian kotak hitam membuat beberapa kumpulan kondisi masukan yang sepenuhnya akan melakukan semua kebutuhan fungsional untuk program.

Menurut Rosa A.S dan M. Shalahuddin (2014:275), *black box testing* adalah menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksud untuk mengetahui apakah fungsi-fungsi,



masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Menurut Pressman (2012:597), Pengujian kotak hitam berupaya untuk menemukan kesalahan dalam kategori berikut:

1. Fungsi yang salah atau hilang
2. Kesalahan antara muka
3. Kesalahan dalam struktur data atau akses basis data eksternal
4. Kesalahan perilaku atau kinerja
5. Kesalahan inisialisasi dan penghentian.

2.2.8. Unified Modeling Language (UML)

Rosa A.S & M. Shalahuddin (2014:133), UML adalah salah satu standar bahasa yang digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisa dan desain serta menggambarkan arsitektur dalam pemograman berorientasi objek.

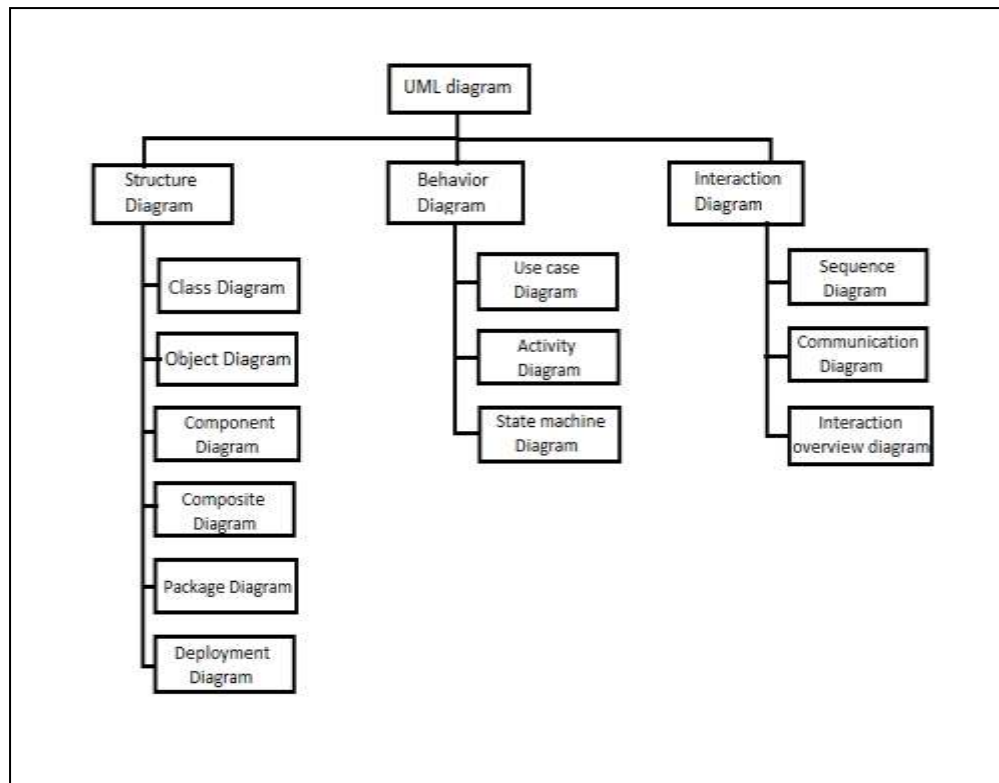
Menurut Sukamto & M. Shalahuddin (2013:133). UML adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi obyek.

UML menyediakan bahasa pemodelan visual yaitu proses penggambaran informasi-informasi secara grafis dengan notasi-notasi baku yang telah disepakati sebelumnya. Dengan menggunakan pomedelan UML ini, pengembang dapat melakukan :

1. Tinjauan umum bagaimana arsitektur sistem secara keseluruhan
2. Penelaahan bagaimana objek-objek dalam sistem saling mengirimkan pesan dan saling bekerja sama satu sama lain
3. Menguji apakah sistem sudah berfungsi seperti yang seharusnya
4. Dokumentasi sistem untuk keperluan-keperluan tertentu di masa yang akan datang.



Pembagian kategori dan macam-macam diagram dapat dilihat pada gambar 2.3.



(Sumber : Rosa A.S & M. Shalahudin , 2013)

Gambar 2.3 Diagram UML

Berikut ini penjelasan singkat dari pembagian kategori tersebut :

1. *Structure* diagram yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan
2. *Behavior* diagram yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem
3. *Interaction* diagram yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar sub sistem pada suatu sistem.



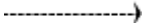
2.2.9. Jenis-jenis Diagram

2.2.9.1. Use Case Diagram

Menurut Indrajani (2015:45), *Usecase diagram* merupakan suatu diagram yang berisi *usecase*, *actor* serta *relationship*, yang merupakan titik awal dalam menganalisis kebutuhan sistem pada saat perancangan.

Nugroho (2005:19), *Usecase diagram* bersifat statis yang memperlihatkan himpunan *usecase* dan *actor* untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan dan diharapkan pengguna serta antara analis dan *client*, sedangkan *usecase* merupakan konstruksi untuk mendeskripsikan bagaimana sistem akan terlihat di mata *user* dan *actor* mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. Sebuah *actor* mungkin hanya memberikan informasi inputan pada sistem, hanya menerima informasi dari sistem atau keduanya menerima, dan memberi informasi pada sistem. Simbol-simbol *Usecase Diagram* dapat dilihat pada tabel 2.2.

Tabel 2.2 Simbol-Simbol *Usecase Diagram*

NO	GAMBAR	Nama	KETERANGAN
1.		<i>Actor</i>	<i>Actor</i> menggambarkan pengguna dari sistem, dapat berupa manusia atau sistem terotomatisasi lain yang berinteraksi dengan sistem lain untuk berbagi, mengirim dan menerima informasi.
2.		<i>Use case</i>	Interaksi antara <i>actor</i> dengan <i>software</i>
3.		<i>Association</i>	Hubungan antara <i>actor</i> dan <i>usecase</i>
4.		<i>Generalization</i>	Hubungan generaliasasi antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
5.		Relasi <i>include</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini

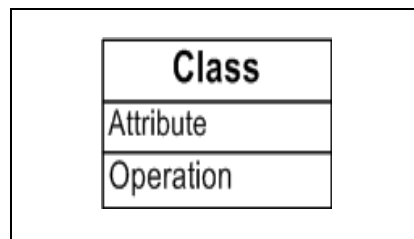
(Sumber: Indrajani, 2015:46)



2.2.9.2. Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut dengan atribut dan metode atau operasi (Rosa A.S & M. Shalahudin, 2014:141).

Class diagram bersifat statis, digaram ini memperlihatkan himpunan kelas-kelas, antarmuka- antarmuka, kolaborasi-kolaborasi , serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi obyek. Meskipun bersifat statis, sering pula *class diagram* membuat kelas-kelas aktif (Nugroho,2005:19).



Sumber : Nugroho (2005;111)

Gambar 2.4 Struktur *class*

- a. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas
- b. Operasi adalah fungsi-fungsi yang dimiliki oleh suatu kelas

2.2.9.3. Sequence Diagram

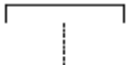
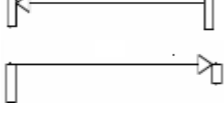
Menurut Indrajani (2015:50), *Sequence Diagram* merupakan suatu diagram yang menggambarkan bagaimana objek-objek berpartisipasi dalam bagian interaksi dan pesan yang ditukar dalam uruta waktu.

Menurut Nugroho (2005:91) *Sequence Diagram* adalah *interaction diagram* yang memperlihatkan event-event yang berurutan sepanjang berjalanya waktu. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antara objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem waktu atau urutan gunakan *sequence diagram*, Berikut



merupakan simbol yang digunakan dalam *sequence diagram* yang dapat dilihat pada tabel 2.3

Tabel 2.3 Simbol-simbol *sequence diagram*

No	Simbol	Deskripsi
1.	<i>LifeLine</i> 	Objek <i>entity</i> , yang saling berinteraksi
2.	<i>Message</i> 	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3.	<i>Activation</i> 	<i>Activation</i> dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah <i>lifeline</i> . Mengindikasikan sebuah obyek yang akan melakukan sebuah aksi.
4.	<i>Actor</i> 	Actor menggambarkan actor yang terlibat dalam system

(Sumber: Indrajani 2015:51)

2.2.9.4 Activity Diagram

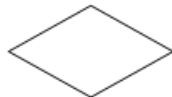
Menurut Nugroho (2005) *Activity diagram* adalah salah satu cara untuk memodelkan *event-event* yang terjadi dalam suatu *use case*. *Activity diagram* digunakan untuk memodelkan aspek dinamis dari sistem. *Activity diagram* secara esensial mirip diagram alir (*flowchart*), memperlihatkan aliran kendali dari suatu aktivitas ke aktivitas lainnya. Ada juga yang mendefinisikan Rosa A.S & M.Shalahudin, (2014:161) *Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada



pada perangkat lunak. *Activity diagram* menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem

Berikut adalah simbol-simbol yang ada pada *Activity diagram*:

Tabel 2.4 Simbol-simbol *Activity diagram*

No	Simbol	Deskripsi
1	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
2	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3	Percabangan / decision 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
4	Status akhir 	Status akhir yang dilakuka sistem, sebuah diagram aktivitas memiliki sebuah status akhir
5	Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Sumber : Rosa A.S & M. Shalahuddin,2014:162)

2.3 Teori Program

2.3.1. Java Development Kit (JDK)

Kadir(2013:4), *Java development kit* (JDK) adalah perangkat pengembang aplikasi *java* bersifat *open source*. Perangkat ini mutlak diperlukan untuk membuat aplikasi android, mengingat aplikasi *android* bersifat *java*.

Safaat (2013:2), JDK merupakan *software* yang bisa dilakukan instalasi melalui *repository* sistem operasi yang digunakan.



Wahana komputer (2013:7), *Android* SDK berjalan dengan menggunakan *resource dai java SE development kit* (JDK). Oleh karena itu jika komputer belum terinstal JDK, maka harus dimulai untu men-*dowload* JDK.

2.3.2. *Software Development Kit* (SDK)

Kadir (2013:5) adalah kumpulan *software* yang berisi mengenai pustaka, *debugger* (alat pencari kesalahan program), *emulator* (peniru perangkat bergerak), dokumentasi, kode contoh dan panduan.

Menurut Rickyanto(2005:9) Secara garis besar Java 2 SDK terdiri dari :

1. *Java compiler*
2. *Java virtual machine* (sering disebut *java runtime enviroment/ JRE*)
3. *Java class libraries*(koleksi *class* yang dapat digunakan untuk menghasilkan program *java*
4. *Java appletviewer* (program untuk menjalankan *applet* tanpa menggunakan *browser*
5. *Java debugger* dan *tool* lain
6. Dokumentasi(dalam paket *download* terpisah)

2.3.3 *Android Development Tools* (ADT)

ADT merupakan plugin untuk *eclipse* yang didesain untuk pengembangan aplikasiandroid.(<http://haidibarasa.wordpress.com./2013/07/15/pengertianandroid-development-tools-adt-html>,diakses 23/05/2017).

Wahana komputer (2013:10), menjelaskan “ *Plugin android development tools* untuk *eclipse* adalah sebuah pengembangan lebih lanjut dari *eclipse* yang dapat digunakan untuk mendukung pembuatan dan proses *debugging* dari aplikasi *android* yang sedang dibuat”. Dijelaskan pula dengan ADT dapat melakukan hal-hal berikut ini dengan menggunakan *eclipse*:

1. Membuat proyek *android* baru
2. Mengakses peralatan *android emulator* beserta tipe perangkatnya



3. Kompile dan *debug* aplikasi *android*
4. Ekspor aplikasi *android* menjadi file paket APK
5. Membuat sertifikasi digital untuk code-signing atas aplikasi APK yang telah dibuat.

2.3.4 Java

Menurut Kadir (2013:4) menjelaskan “*Java* merupakan bahasa pemrograman berorientasi objek dan bebas platform, dikembangkan oleh *SUN Micro System* dengan sejumlah keunggulan yang memungkinkan *java* dijadikan sebagai bahasa pengembangan *enterprise*”.

Menurut Sukanto dan Shalahuddin (2013:103), “*Java* adalah bahasa pemrograman objek murni karena semua kode programnya dibungkus dalam kelas.”

2.3.5. Eclipse

Wahana komputer (2013:10), *Eclipse* adalah *software development environment* multibahasa yang berfitur *extensible plugin*. *Extensible plugin* berarti pengembang dapat mengembangkan beberapa jenis aplikasi, misalnya *java*, *C*, *C++* atau *Python* dalam satu aplikasi IDE dengan cara menambahkan *plugin*.

Kadir (2013:5), *eclipse* merupakan pengembang aplikasi yang tergolong sebagai *Integrated Development Environment* (IDE), karena menyediakan berbagai fasilitas untuk pembuatan aplikasi.

2.3.6. Basis data

Menurut Sujatmiko (2012:40), mengemukakan bahwa, “basis data merupakan kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan program komputer untuk memperoleh informasi dari basis data”. Adapula pendapat lain dari Pratama



(2014:17) “elemen basis data pada sistem informasi berfungsi sebagai media untuk penyimpanan data dan informasi yang dimiliki oleh sistem informasi bersangkutan”.

Dengan demikian dapat disimpulkan bahwa basis data adalah kumpulan data yang saling berkaitan yang disimpan dalam suatu komputer yang berfungsi sebagai penyimpanan data dan informasi yang dimiliki oleh aplikasi tertentu.

2.3.7. My Structured Query Language (Mysql)

Pada saat ini Mysql salah satu *database server* yang sangat populer di dunia, semua itu karena bahasa dasar yang digunakan untuk mengakses basis data adalah *Structured Query Language* (SQL). MySQL sudah sangat familiar bagi *developer* dengan sintak yang mudah dipahami dan tidak rumit, pengaksesan database dapat dilakukan dengan mudah, dan didukung dengan pemrograman umum seperti PHP, Java, C++ dan lain-lainnya serta Mysql bersifat *open source*. Database dapat dibuat dengan untuk keperluan sistem basis data yang cepat, handal, dan mudah digunakan.

Menurut Nugroho (2013:94) menjelaskan bahwa “Mysql adalah *software* atau program *database server*”.

Menurut Madcoms (2012:282) menjelaskan bahwa “Mysql adalah salah satu *database* untuk *server* yang menggunakan SQL yang bersifat *free* (gratis atau tidak perlu membayar untuk menggunakannya), selain itu dapat berjalan di berbagai platform antara lain linux dan windows”.

2.3.8 XAMPP

Riyanto (2011: 01), XAMPP merupakan paket PHP dan MySQL berbasis *open source*, yang dapat digunakan sebagai tool pembantu pengembangan aplikasi berbasis PHP. XAMPP mengombinasikan beberapa paket perangkat lunak berbeda ke dalam satu paket.



Riyanto (2011:16) Fitur XAMPP berupa PHP My Admin untuk melakukan administrasi *database*, mulai dari membuat, menghapus dan memodifikasi *database* dengan mudah dan cepat termasuk administrasi atribut *database*, seperti *table*, *view*, *stored*, *procedure*, *store function* dan *trigger*.

2.3.9 PHP

Ardhana (2012:88), “PHP (*Hypertext Prerocessor*) adalah bahasa pemrograman berbasis *server-side* yang dapat melakukan parsing *script* web sehingga dari sisi *client* menghasilkan suatu tampilan yang menarik.”

Sujatmiko (2012:213), “PHP adalah bahasa pemrograman yang bisa kita gunakan untuk membuat aplikasi web.”

Risnandar, dkk (2013:57), “PHP (*Hypertext Preprocessing*) merupakan bahasa pemrograman yang biasa digunakan untuk membuat halaman HTML.

Dengan demikian dapat didefinisikan PHP merupakan bahasa pemrograman bersifat open source yang digunakan untuk membangun aplikasi web.

2.4 Referensi Jurnal

Dalam Tugas Akhir ini penulis menggunakan beberapa perbandingan jurnal. Dibawah ini akan diuraikan perbandingan dari referensi jurnal tersebut.

Tabel 2.5 Referensi Jurnal

No	Judul Jurnal	Metode	Pembahasan
1.	Rancang Bangun Aplikasi Fasilitas Umum Berbasis Web Services. E-journal Tek. Informatika, Vol. 8, No 1 (2016) ISSN : 2301-8364	Metode <i>Desciplined Agile Delivery</i> (DAD)	Permasalahanya adalah masyarakat sulit mendapatkan infromasi mengenai fasilitas umum di kota manado secara cepat. Hasilnya adalah dengan rancang bangun aplikasi berbasis <i>android</i> ini maka masyarakat lebih mudah mendapatkan infromasi mengenai fasilitas umum secara cepat, mudah dan akurat.



Lanjutan Tabel 2.5 Referensi Jurnal

No	Judul Jurnal	Metode	Pembahasan
2.	Pembuatan Aplikasi Cek Tagihan Listrik Berbasis Android. Jurnal Dimensi Tek. Elektro Vol. 1 No.1 (2013) 24-28	Metode parsing HTML	Permasalahannya adalah proses dalam mengecek tagihan listrik bulanan, pelanggan harus mengunjungi kantor PLN, Hasilnya dengan membangun aplikasi berbasis android ini pelanggan dapat mengecek tagihan listrik secara realtime, tanpa harus ke pln
3.	Perancangan Sistem Informasi Pemetaan dan Pemantauan Daerah Aliran Sungai (DAS) Tondano di Kota Manado Berbasis SMS-Gateway E-Jurnal Tek.Informatika, Vol. 5, No.1 (2015) ISSN : 2301-8364	Metode Desciplined Agile Delivery (DAD).	Permasalahannya adalah kurangnya infromasi kepada masyarakat kota manado mengenai DAS dan infromasi mengenai ketinggian air tidak diperoleh dalam waktu singkat. Hasilnya dengan menrancang bangun sistem infromasi pemetaan dan pemantauan DAS berbasis web sehingga dapat mempermudah masyarakat memperoleh informasi.
4.	Analisa dan perancangan aplikasi penilaian pegawai dalam jabatan struktural di pemerintahan kota manado. E-Journal Tek. Elektro dan Komputer (2014), ISSN : 2301-8402	Metode Sistem Pendukung Keputusan <i>Analytical Hierarchy Proses</i> (AHP) Metode Pengembangan Perangkat Lunak <i>Desciplined Agile Delivery</i> (DAD)	Permasalahannya adalah proses penilai pegawai masih dilakukan secara subyektif, sehingga biasanya terjadi ketidakadilan dalam kenaikan jabatan. Hasilnya adalah dengan adanya aplikasi penilaian pegawai dengan menggunakan metode AHP , maka penilain dilakukan berdasarkan kriteria dan alternatif yang telah ditentukan, sehingga lebih objektif dan pegawai yang terpilih untuk menduduki jabatan jatuh pada orang yang tepat.

