

BAB II

LANDASAN TEORI

1.1. Tinjauan Studi

Penelitian ini merujuk pada beberapa penelitian yang terkait dengan permasalahan pada penelitian yang berjudul “Penerapan Sistem Informasi Pembayaran Tagihan Air Pamsimas Tirta Langgeng Berbasis Web di Desa Batealit Kabupaten Jepara”. Penelitian terkait digunakan penulis sebagai referensi sekaligus bahan pertimbangan dengan rancangan yang akan penulis bangun.

Penelitian yang dilakukan oleh Trias Bratakusuma, Edi Kuswanto dan Zanuvar Rifai pada tahun 2019 yang berjudul “Integrasi Pembayaran Pamsimas Desa Dermaji dengan PPOB Menggunakan Web Service”. Penelitian ini menjelaskan tentang pelayanan pembayaran tagihan rekening air masih dilakukan secara manual sehingga membutuhkan waktu yang lama ketika pelanggan banyak yang berdatangan untuk melakukan pembayaran. Selain itu pembuatan laporan tahunan masih tercatat secara manual pula, yaitu dengan pencatatan data pada sebuah buku besar. Penelitian ini bertujuan untuk menghasilkan sebuah sistem informasi pembayaran pamsimas secara online dan realtime untuk menghindari terjadinya *double payment*. Penelitian ini dilakukan dengan menggunakan metode pengembangan *Rapid Application Development (RAD)*. Pada tahap implementasi peneliti menggunakan bahasa pemrograman *PHP* dan *Framework Codeigniter*. Pada tahap akhir dihasilkan sebuah aplikasi pembayaran pamsimas dengan PPOB *web service* untuk desa Dermaji yang diharapkan dapat mempermudah warga desa Dermaji untuk melakukan proses pembayaran pamsimas di berbagai loket PPOB yang telah disediakan [4].

Penelitian yang dilakukan oleh Zulkarnain pada tahun 2020 yang berjudul “Sistem Informasi Pengolahan Data Pelanggan Pada CV. Indoprint Banda Aceh”. Penelitian ini membahas tentang proses pencatatan data pelanggan dan pengolahan tagihan pelanggan masih dilakukan secara sederhana yaitu dengan pencatatan data pelanggan ke dalam sebuah buku tagihan yang telah disediakan oleh perusahaan, kemudian pada setiap akhir bulan bagian keuangan akan merekap kembali semua data yang ada baik dari transaksi keuangan maupun tagihan pelanggan. Sistem yang berjalan memiliki banyak kekurangan diantaranya membutuhkan proses yang lama dalam proses pengolahan data pelanggan maupun dalam proses pembuatan laporan tiap bulannya. Metode pengumpulan data dilakukan dengan cara studi kepustakaan, observasi dan wawancara. Hasil dari penelitian ini adalah memberikan kemudahan bagi staff bidang administrasi CV.

Indoprint Banda Aceh dalam proses pengolahan data pelanggan dan mempercepat proses pembuatan laporan [5].

Penelitian dilakukan oleh Aang Alim Murtopo dan Devi Cici Angesti pada Tahun 2017 dengan judul “Sistem Informasi Pelayanan Tagihan Rekening dan Pengaduan Pelanggan Berbasis *SMS Gateway* di PDAM Kota Tegal”. Penelitian ini membahas tentang bagaimana membangun aplikasi untuk memberikan suatu informasi pelayanan kepada pelanggan jika ingin mengetahui besarnya tagihan rekening pemakaian air. Pelanggan tidak perlu jauh - jauh mendatangi kantor PDAM kota Tegal ataupun menghabiskan biaya telepon apabila ada komplain maupun pengaduan pelanggan. Penelitian ini diharapkan dapat meningkatkan kinerja petugas untuk memberikan pelayanan yang efisien serta penyampaian segala informasi yang dibutuhkan pada setiap bagian dihasilkan secara tepat, cepat dan akurat. Dalam membuat aplikasi, peneliti merancang *software* yang dapat menangani masalah pembayaran tagihan pemakaian air serta pengaduan keluhan pelanggan berbasis *SMS Gateway*. Metode pengembangan dalam penelitian ini menggunakan *System Development Life Cycle (SDLC) waterfall model*. Hasil dari penelitian ini memberikan kemudahan bagi pihak PDAM untuk menyampaikan segala informasi kepada pelanggan mengenai tagihan rekening pemakaian air sebelum tanggal jatuh tempo sehingga pelanggan mendapat mengetahui informasi tepat pada waktunya serta tidak terlambat dalam membayar tagihan [6].

Penelitian yang dilakukan oleh Lydia Salvina Helling pada tahun 2018 yang berjudul “Perancangan Sistem Informasi Pelayanan Pelanggan Pada Citra Laundry Bogor”. Penelitian ini menjelaskan tentang prosedur pelayanan konsumen pada Citra Laundry masih menggunakan sistem yang manual, mereka tidak memiliki data pelanggan maupun data kepemilikan barang secara spesifik. Pelanggan datang ke tempat Citra Laundry kemudian memberikan pakaian ataupun sejenisnya, dan petugas akan menimbang barang yang akan di laundry setelah itu petugas akan memberikan nota sederhana untuk detail pembayaran dan waktu pengambilan pakaian yang di laundry. Kemudian pelanggan akan datang kembali untuk melakukan pembayaran sesuai dengan waktu pengambilan. Sistem yang berjalan memiliki banyak kekurangan diantaranya dapat terjadi kesalahan petugas atau *human error* dalam proses pelayanan pelanggan. Penelitian ini bertujuan untuk menghasilkan sebuah sistem pelayanan konsumen yang terkomputerisasi. Pengembangan sistem menggunakan *Waterfall method* dalam perangkat lunaknya serta menggunakan modelling sistem DFD (*Data Flow Diagram*). Tahap akhir penelitian ini menghasilkan suatu sistem informasi pelayanan pelanggan yang lebih efisien dalam penyimpanan

maupun mengolah pendataan yang besar serta sangat efektif dalam menangani kebutuhan pelayanan konsumen yang cepat dan tersistematis [7].

Penelitian yang dilakukan oleh Siti Zulaiha pada tahun 2018 dalam jurnalnya yang berjudul “Perancangan dan Implementasi Aplikasi Sistem Pengolahan Data Administrasi Pembayaran Rekening Air (Studi Kasus Kantor BLUD SPAM IKK Pota Kecamatan Sambi Rampas NTT)”. Penelitian ini menyebutkan bahwa penggunaan teknologi komputer pada BLUD SPAM IKK Pota masih belum terlaksana dengan baik, khususnya dalam pelayanan administrasi pembayaran rekening air masih dilakukan secara manual dan membutuhkan waktu yang lama dalam proses rekap data. Dalam membangun sistem informasi berbasis web, peneliti menggunakan *Algoritma Bubble Sort*. Pada tahap implementasi peneliti menggunakan bahasa pemrograman *PHP* dan *MySQL* sebagai penyimpanan basis data. Penelitian ini bertujuan untuk membantu pelanggan BLUD SPAM IKK Pota Kecamatan Sambi Rampas Provinsi Nusa Tenggara Timur (NTT) dalam memperoleh informasi tagihan dan pembayaran rekening air pelanggan serta membantu pihak kantor dan petugas dalam mengelola administrasi pelanggan [8].

Tabel 1.1 Perbandingan Penelitian Sebelumnya.

No.	Pembuat	Kelebihan	Kekurangan	Studi Kasus
1.	Trias Bratakusuma, Edi Kuswanto, Zanuar Rifai	Aplikasi ini terintegrasi dengan PPOB berbasis <i>web service</i>	Aplikasi ini hanya dapat diakses oleh petugas dan belum ada akses untuk pelanggan	Purwokerto
2.	Zulkarnain	Aplikasi dibangun dengan menggunakan <i>Microsoft Visual Basic</i>	Aplikasi hanya terbatas desktop platform	Aceh
3.	Aang Alim Murtopo dan Devi Cici Angesti	Aplikasi ini dapat memberikan pelayanan dan informasi kepada pelanggan melalui <i>SMS</i>	Aplikasi ini berbasis <i>software</i> yang tidak bisa melayani secara <i>realtime</i> ketika sistem <i>offline</i>	Tegal

4.	Lydia Salvina Helling	Aplikasi ini dapat menampilkan data pelanggan serta melakukan <i>printout</i> struk pembayaran	Aplikasi belum tersistem secara <i>online</i>	Bogor
5.	Siti Zulaiha	Aplikasi ini menggunakan algoritma <i>Bubble Sort</i>	Aplikasi ini tidak ada akses untuk pelanggan	Sambi Rampas, NTT

1.2. Tinjauan Pustaka

1.2.1. Pengertian Pamsimas

Program Penyediaan Air Minum dan Sanitasi yang Berbasis Masyarakat (PAMSIMAS) adalah salah satu program dan aksi nyata yang dilakukan oleh Pemerintah Pusat maupun Pemerintahan Daerah dengan mendapatkan dukungan dari Bank Dunia, untuk memaksimalkan penyediaan air minum, fasilitas sanitasi dan meningkatkan kesehatan masyarakat terutama untuk mengurangi angka diare dan penyakit lainnya yang ditularkan melalui media air serta lingkungan yang berbasis masyarakat.

Tujuan Program Pamsimas adalah untuk menciptakan kebiasaan masyarakat berperilaku hidup sehat dan bersih melalui peningkatan akses pelayanan air minum dan fasilitas sanitasi modern bagi masyarakat miskin pedesaan terutama masyarakat di desa yang tertinggal maupun masyarakat di daerah pinggiran kota. Adapun tujuan program PAMSIMAS secara rinci adalah sebagai berikut [9]:

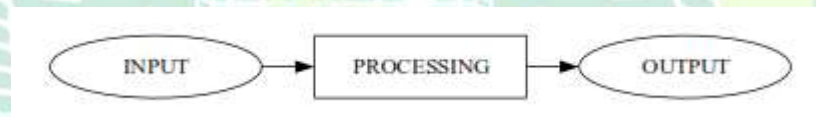
1. Meningkatkan praktik hidup bersih dan sehat dalam lingkungan masyarakat.
2. Meningkatkan jumlah masyarakat yang mendapatkan akses program layanan air minum dan sanitasi yang berkelanjutan serta dapat dikelola secara efektif.
3. Meningkatkan kapasitas masyarakat serta lembaga lokal dalam menyelenggarakan pelayanan air minum dan fasilitas sanitasi yang berbasis masyarakat.
4. Meningkatkan efektivitas dan keberlanjutan jangka panjang dalam membangun sarana maupun prasarana air minum dan sanitasi modern berbasis masyarakat.

Sasaran program ini adalah masyarakat, terutama kelompok masyarakat miskin di daerah pedesaan dan pinggiran kota yang memiliki prevalensi penyakit terhadap air yang tinggi serta belum mendapatkan akses pelayanan air minum dan sanitasi [10].

1.2.2. Pengertian Sistem Informasi

Sistem Informasi merupakan rangkaian dari orang-orang, data, proses, interface, teknologi, prosedur-prosedur yang mengeluarkan informasi untuk mencapai tujuan dalam memecahkan masalah atau mengambil keputusan. Dapat disimpulkan bahwa sistem informasi terdiri dari beberapa komponen seperti manusia, komputer, teknologi informasi, dan prosedur kerja yang saling bekerjasama sehingga menghasilkan proses perubahan data menjadi informasi serta dirancang untuk mencapai suatu tujuan atau sasaran organisasi [11].

Menurut Abdilah (2016), sistem informasi terdiri dari tiga konsep dasar, diantaranya: masukan (input), proses (processing) dan keluaran (output). Ketiga elemen dasar ini menghasilkan suatu informasi yang dibutuhkan untuk melakukan pengambilan keputusan, pengendalian operasi, analisis permasalahan dan menciptakan produk atau jasa baru.



Gambar 1.1 Konsep Dasar Sistem Informasi.

(Sumber : Jurnal Teknologi Terpadu, 2016)

1.2.3. Website

Website merupakan sekumpulan halaman yang berisi suatu informasi data digital seperti text, gambar, suara, animasi maupun video ataupun gabungan diantara semuanya yang tersedia melalui koneksi internet sehingga dapat dilihat dan diakses oleh orang-orang yang ada seluruh penjuru dunia. *Website* dirancang dengan menggunakan bahasa standar HTML. *Script* HTML ini kemudian diterjemahkan oleh *browser web* sehingga dapat menampilkan segala informasi yang mampu diakses oleh semua orang. Secara garis besar, website dapat dibagi menjadi 3 jenis [12], diantaranya:

1. Website Statis, merupakan jenis website yang tidak dapat melakukan pembaharuan secara berkala, sehingga akan selalu tetap dari waktu ke waktu.

Website jenis ini biasanya digunakan untuk menampilkan informasi profil dari pemilik website seperti halnya profil suatu perusahaan ataupun organisasi.

2. Website dinamis, adalah jenis website dengan konten dan fiturnya secara berkala dapat selalu diperbaharui oleh pengelola web maupun pemilik website. Website ini banyak dimiliki oleh perusahaan maupun perseorangan yang aktivitas bisnisnya berkaitan dengan dunia maya. Contoh paling sederhana dalam adalah *blog website* dan website berita.
3. Website interaktif, adalah jenis website yang masuk kedalam kategori website dinamis, dimana selalu melakukan pembaharuan informasi dari waktu ke waktu. Namun, segala informasi tidak hanya diubah oleh administrator website akan tetapi sebagian besar dilakukan oleh pengguna situs itu sendiri.

1.2.4. Pemrograman Web

Web Programming atau Pemrograman Web terdiri dari kata pemrograman dan web. Pemrograman dapat diartikan sebagai suatu proses ataupun cara pembuatan program menggunakan bahasa pemrograman. Adapun bahasa pemrograman adalah bahasa yang dapat digunakan untuk memberikan instruksi pada sebuah komputer sehingga dapat memproses suatu data serta informasi dapat ditampilkan sesuai dengan yang dikehendaki oleh pemrogram. Dengan demikian pemrograman web dapat diartikan sebagai kegiatan membuat suatu program atau aplikasi yang berbasis web dengan menggunakan bahasa pemrograman tertentu, sehingga dapat melakukan *processing* pada suatu data serta menghasilkan informasi sesuai yang dikehendaki oleh pemilik situs web.

Bahasa pemrograman web terdiri dari beberapa unsur Bahasa pemrograman. Setidaknya ada 5 bahasa pemrograman yang biasanya digunakan untuk membangun sebuah *website* yang dinamis, yang mana setiap bahasa memiliki perannya masing-masing sesuai dengan kebutuhan, diantaranya adalah:

1. HTML, berperan sebagai pembentuk struktur halaman pada website dengan menempatkan setiap elemen-element website sesuai dengan *layout* yang dikehendaki.
2. CSS, berperan sebagai perancang desain pada website dengan mengatur elemen html agar tampil lebih menarik pada browser.

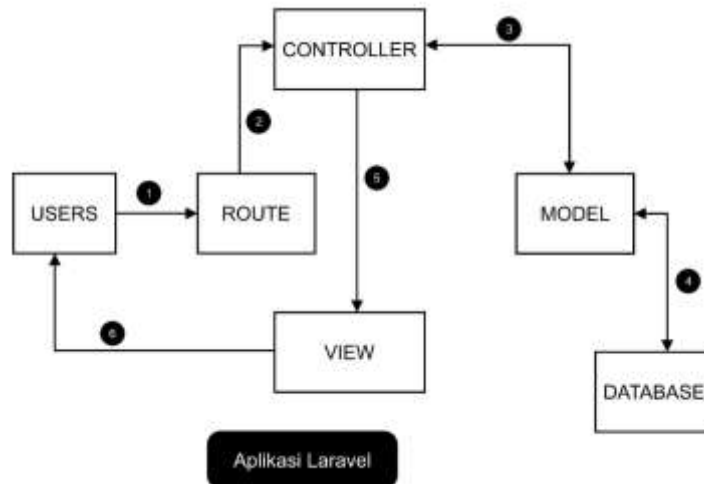
3. PHP, berperan sebagai pemroses data pada bagian server sesuai yang dibutuhkan oleh client menjadi suatu informasi yang siap untuk ditampilkan, juga sebagai penghubung antara aplikasi web dengan databasenya. ASP dan Java dapat menjadi alternatif bahasa pemrograman yang dapat digunakan selain PHP.
4. SQL, berperan sebagai bahasa yang mengatur proses transaksi data antara aplikasi dan database sebagai media penyimpanan data. Terdapat beberapa *database* yang umum digunakan seperti Mysql, SQL Server, Oracle dan lain sebagainya.
5. JavaScript, berperan sebagai bahasa yang melakukan pemrosesan data pada sisi klien serta secara dinamis dapat memanipulasi HTML maupun CSS.

Dari 5 bahasa program diatas, hingga saat ini telah berkembang secara signifikan menjadi bentuk yang berbeda, ada yang berupa *library* seperti halnya jQuery yang merupakan *library function* JavaScript serta ada juga yang berupa *framework* seperti Bootstrap yang memberikan kemudahan *programmer* untuk mendesain website yang lebih cepat dan dinamis [12].

1.2.5. Laravel Framework

Laravel merupakan sebuah *framework web* berbasis PHP yang bersifat *open-source* dan tidak berbayar, diciptakan oleh Taylor Otwell dan diperuntukkan bagi pengembang aplikasi web yang menggunakan pola MVC. Struktur pola MVC yang terdapat pada laravel sedikit berbeda dengan struktur pola MVC pada umumnya. Pada laravel terdapat *routing* sebagai jembatan antara *request* dari *user* dengan *controller*. Jadi *controller* tidak dapat langsung menerima *request* dari *user* tersebut [13].

Menurut Muhammad Azamuddin (2018), *Laravel* merupakan *framework* yang mengutamakan *developer experience*. Hal ini sangat cocok untuk memudahkan dalam proses pembuatan suatu aplikasi yang aman dan modern.



Gambar 1.2

Visualisasi MVC Pada Laravel.

Dari gambar diatas tentang sistem kerja Laravel, dapat dijelaskan secara singkat sebagai berikut :

1. User mengakses melalui *router* tertentu.
2. *Router* tersebut oleh aplikasi telah dipetakan ke *controller action*.
3. *Controller action* akan menggunakan *model* untuk mengakses data atau langsung mengembalikan *view* tanpa data (langsung menuju tahap ke-5).
4. *Model* berinteraksi ke *database* untuk mendapatkan data atau menyimpan data.
5. Setelah data berhasil didapatkan melalui *model*, *controller* akan mengembalikan sebuah *view* sekaligus data (jika ada).
6. *View* tersebut yang akhirnya dapat dilihat atau ditampilkan ke sisi *user*.

1.2.6. Database

Database diartikan sebagai kumpulan item data yang berhubungan antara satu dengan yang lainnya, yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, tersimpan ke dalam hardware komputer dan dengan menggunakan suatu software tertentu dalam melakukan manipulasi untuk memperoleh kegunaan tertentu [14].

Keamanan database merupakan pemberian perlindungan pada database terhadap segala bentuk ancaman dan gangguan, baik bersifat teknis maupun administratif. Hal ini penting karena seringkali terdapat gangguan yang sangat bervariasi yang terjadi terhadap database, seperti software, hardware, data, dan manusia. Secara umum,

terjadinya gangguan pada database, baik secara fisik maupun nonfisik, seperti pencurian data, hilangnya kerahasiaan data, kehilangan kemampuan data, serta kehilangan integritas data [14].

Untuk memberi perlindungan keamanan pada database dapat dilakukan beberapa cara, antara lain pemberian otoritas pada pengguna untuk mengakses objek-objek dalam database.

Aspek-aspek layanan keamanan data yang ada dalam suatu database adalah:

1. Kerahasiaan (confidentiality). Layanan agar pesan tetap aman tidak dapat dibaca oleh oknum-oknum yang tidak berhak.
2. Integritas data (data integrity). Layanan untuk memberi jaminan bahwa pesan masih tetap asli/utuh, atau tidak pernah dimanipulasi selama proses pengiriman.
3. Otentikasi (authentication). Layanan yang berkaitan dengan identifikasi, baik mengidentifikasi suatu kebenaran pihak-pihak yang melakukan berkomunikasi (user authentication) maupun mengidentifikasi kebenaran dari suatu sumber pesan (data origin authentication).
4. Penyangkalan (non-repudation). Layanan untuk melakukan pencegahan terhadap entitas yang berkomunikasi untuk melakukan penyangkalan, yaitu pengirim pesan menyangkal bahwa telah melakukan proses pengiriman pesan, ataupun penerima pesan yang menyangkal telah menerima pesan yang telah dikirimkan.

1.2.7. XAMPP

Xampp merupakan singkatan dari X (empat sistem operasi apapun), Xampp merupakan perangkat lunak bebas yang mendukung banyak sistem operasi, dan merupakan kompilasi dari beberapa program. Seperti Apache, MYSQL, PHPP, dan Perl. Xampp adalah tools yang menyajikan satu buah paket yang berisi paket perangkat lunak. Dalam paket Xampp sudah mencakup Apache (Web Server), Mysql (Database), PHP (server side scripting), Perl, FTP server, PhpMyAdmin, serta berbagai pustaka bantu lainnya [15].

1.2.8. MySQL

MySQL merupakan sistem manajemen basis data relational. Artinya, data yang dikelola dalam database akan ditempatkan pada tabel yang berbeda sehingga

pengolahan data akan jauh lebih cepat. MySQL dapat digunakan untuk mengelola database dari yang kecil sampai yang sangat besar [15].

MySQL disebut juga dengan *Structured Query Language* (SQL), suatu bahasa terstruktur yang digunakan secara khusus untuk melakukan pengolahan database. SQL juga dapat diartikan sebagai antarmuka untuk manajemen relasional, termasuk didalamnya sistem yang berjalan pada komputer pribadi.

Selain itu, SQL merupakan bahasa pemrograman yang dirancang khusus untuk mengirim suatu perintah query kedalam database dengan mengakses data berdasarkan pengalamatan tertentu. Sebagian besar perangkat lunak database mengimplementasikan SQL dengan sedikit berbeda, akan tetapi seluruh database SQL mendukung subset standar yang sudah ada.

Jadi, SQL adalah suatu permintaan yang disematkan kedalam suatu database ataupun SMBD tertentu. Dengan kata lain, SQL merupakan bahasa atau perintah yang melekat pada SMBD. Sebagai suatu bahasa permintaan, SQL didukung oleh SMBD, seperti MySQL Server, MySQL, Interbase, PostgreSQL, maupun Oracle. Selain itu SQL juga didukung database non-server, seperti Microsoft Access atau Paradox.

1.2.9. UML (Unified Modeling Language)

UML merupakan kependekan dari *Unified Modeling Language* yang artinya adalah sekumpulan alat yang digunakan untuk memvisualkan serta mendokumentasikan suatu hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual. Selain itu, dapat diartikan juga sebagai satu kumpulan konveksi pemodelan yang digunakan untuk menentukan dan menggambarkan sebuah sistem *software* yang berkaitan dengan objek.






Menurut Munir (2016), UML adalah sebuah alat bantu yang memiliki kemampuan handal didalam pengembangan sistem yang berorientasi objek. Kualitas UML ini dibuktikan dengan tersedianya bahasa pemodelan visual yang memudahkan pengembang sistem untuk membuat cetak biru atas keinginan mereka dalam bentuk yang baku dan memberikan kemudahan mekanisme dalam berbagi teknis serta mendiskusikan rancangan mereka kepada pihak lain. Dengan memanfaatkan UML kita dapat merancang model untuk setiap jenis aplikasi perangkat lunak, yang mana aplikasi tersebut dapat dioperasikan pada perangkat keras, sistem operasi dan jaringan



manapun, serta dapat diterjemahkan dalam bahasa pemrograman apapun. Namun dikarenakan UML menggunakan class dan operation dalam konsep dasarnya, maka UML lebih cocok sebagai penulisan perangkat lunak dalam bahasa-bahasa yang berorientasi objek [11].

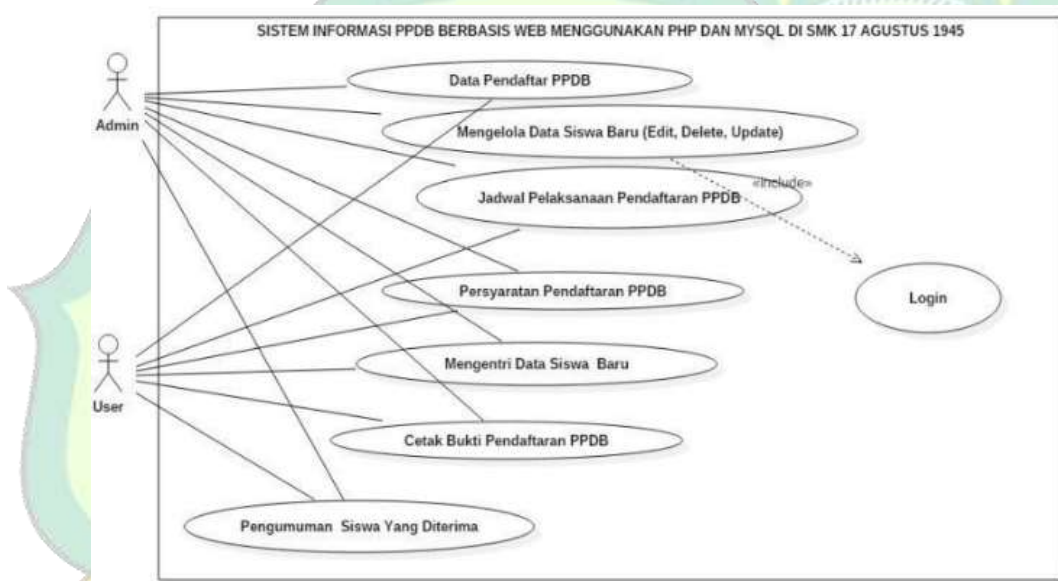
1.2.10. Use Case Diagram

Use case adalah pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. Diagram ini menampilkan fungsionalitas sistem ataupun kelas serta bagaimana sistem dapat terhubung dengan dunia luar, seperti menyusun daftar layanan pembayaran misalnya. Use case diagram dapat juga digunakan untuk memperoleh kebutuhan sistem dan memahami bagaimana harusnya sistem bekerja [16].

Tabel 1.2 Simbol *Use Case Diagram*

No	Gambar	Nama	Keterangan
1.		<i>Actor</i>	<i>Actor</i> adalah proses, orang, atau sistem lain yang dapat berinteraksi dengan sistem yang akan dibuat.
2.		<i>Include</i>	Relasi <i>usecase</i> tambahan yang mana <i>usecase</i> yang ditambah memerlukan <i>usecase</i> ini dalam menjalankan fungsinya
3.		<i>Generalization</i>	Hubungan pada objek anak berbagi perilaku dan struktur data dari objek induk.
4.		<i>Extend</i>	Relasi <i>use-case</i> tambahan yang mana <i>use-case</i> yang ditambah mampu berdiri sendiri meskipun tanpa <i>use case</i> tambahan
5.		<i>Use Case</i>	Keterangan dari urutan aksi yang ditampilkan dalam sistem.

6.		<i>Collaboration</i>	Interaksi elemen lain yang berkerja sama untuk menyediakan perilaku yang besar dari jumlah yang ada.
7.		<i>Association</i>	Menghubungkan antara objek satu ke objek yang lainnya



Gambar 1.3 Contoh *Use Case Diagram*



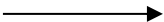


(Sumber : Jurnal Sistem Informasi Penerimaan Peserta Didik Baru Berbasis Android Menggunakan Metode Agile, 2018)

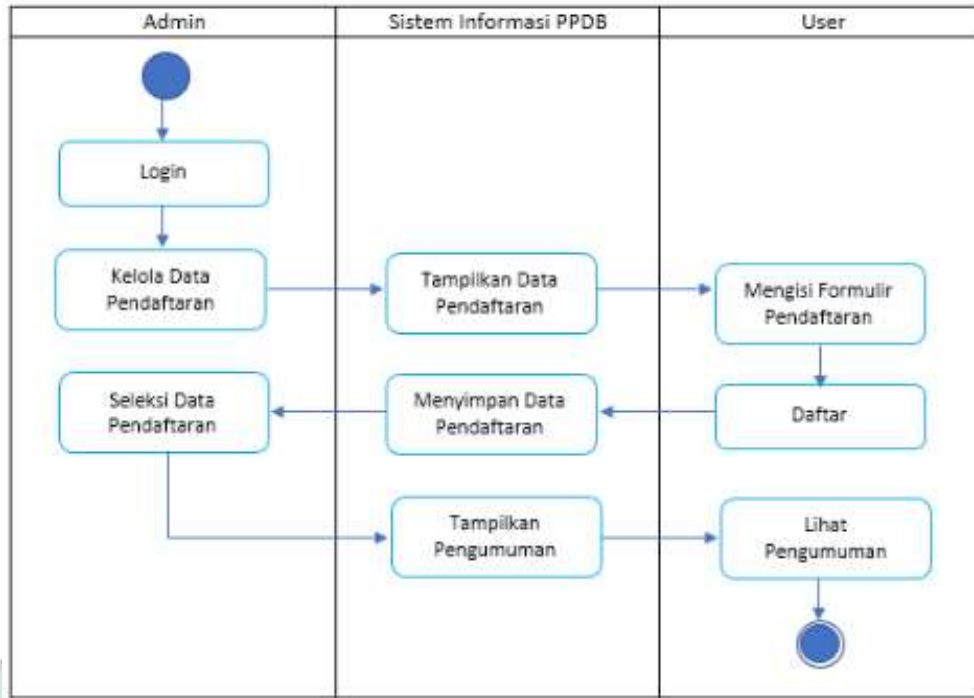
1.2.11. Activity Diagram

Activity diagrams menggambarkan berbagai alur aktivitas pada sistem yang sedang dibuat, bagaimana setiap alir dimulai, keputusan apa yang dapat terjadi, serta bagaimana suatu aktivitas akan berakhir. *Activity diagram* juga dapat menjelaskan bagaimana proses paralel yang mungkin saja terjadi pada beberapa aktivitas. Sebuah kegiatan dapat direalisasikan oleh satu atau beberapa use case. Aktivitas menggambarkan sebuah proses yang sedang berjalan, sedangkan *use case*

menggambarkan bagaimana penggunaan sistem yang dilakukan *actor* untuk melakukan aktivitas [16].

Tabel 1.3 Simbol *Activity Diagram*

No	Simbol	Nama	Deskripsi
1.		<i>Action State</i>	Menggambarkan keadaan elemen dalam suatu aliran aktifitas.
2.		<i>State</i>	Menggunkan kondisi suatu elemen.
3.		<i>Flow Control</i>	Menggambarkan aliran aktifitas dari suatu elemen ke elemen lain.
4.		<i>Initial State</i>	Menggambarkan titik awal siklus hidup suatu elemen.
5.		<i>Final State</i>	Menggambarkan titik akhir yang menjadi kondisi akhir suatu elemen.





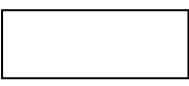
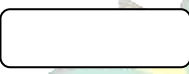
Gambar 1.4 Contoh Activity Diagram

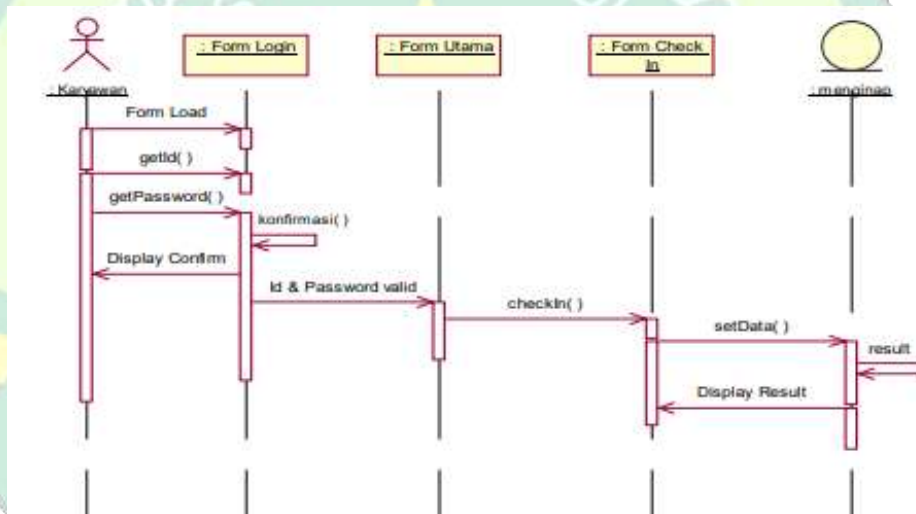
1.2.12. Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek berupa pesan (*message*) yang digambarkan terhadap waktu. *Sequence diagram* terdiri antar dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). Message digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, *message* akan dipetakan menjadi operasi atau metoda dari *class*. *Activation bar* menunjukkan lamanya eksekusi sebuah proses [16].

Tabel 1.4 Simbol *Sequence Diagram*

No	Simbol	Nama	Deskripsi
1.		Object	Menggambarkan pos-pos objek yang pengirim dan penerima message.
2.		<i>Message</i>	Menggunakan aliran pesan yang dikirm oleh pos-pos objek.

3.		<i>Actor</i>	Menggambarkan identitas / subjek yang dapat melakukan suatu proses.
4.		<i>Lifeline</i>	Menggambarkan tempat mulai dan berakhirnya suatu objek
5.		<i>State</i>	Menggambar kondisi suatu objek
6.		<i>Action State</i>	Menggambar keadaan objek dalam melakukan aktivitas


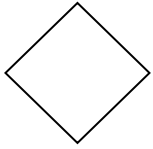
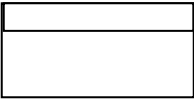






Gambar 1.5 Contoh *Sequence Diagram*

1.2.13. Class Diagram

Class diagram adalah visualisasi kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Diagram ini memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas didalam model desain (dalam *logical view*) dari suatu sistem. Kelas memiliki 3 area utama diantaranya: nama, atribut, dan operasi. Nama berfungsi untuk member identitas pada sebuah kelas, atribut fungsinya adalah untuk menunjukkan karakteristik pada data yang dimiliki suatu objek di dalam kelas, sedangkan operasi fungsinya adalah memberikan sebuah fungsi ke sebuah objek [16].

Tabel 1.5 Simbol *Class Diagram*

No	Simbol	Nama	Deskripsi
1.		<i>Generalitation</i>	Hubungan antara objek anak dari berbagai perilaku dengan struktur dari dari objek yang ada diatas objek induk.
2.		<i>Nary Assosiation</i>	Menggambarkan asosiasi percabangan jika pilihan aktivitas lebih dari dua objek.
3.		<i>Flow Control</i>	Gabungan dari beberapa objek dengan berbagi atribut serta operasi yang sama.
4.		<i>Collaboration</i>	Menggabarkan deskripsi berurutan yang dapat ditampilkan sistem dalam menghasilkan suatu hsil terstruktur bagi sebuah objek.
5.		<i>Dependency</i>	Menggambarkan suatu hubungan di mana terjadi perubahan pada suatu objek sehingga dapat mempengaruhi objek yang bergantung/tidak dapat berdirisendiri.

6.		<i>Realization</i>	Operasi yang benar-benar dilakukan pada sebuah objek.
7.		<i>Associattion</i>	Menggambarkan antara objek yang satu dengan objek yang lain.



Gambar 1.6 Contoh *Class Diagram*

1.2.14. Black Box Testing

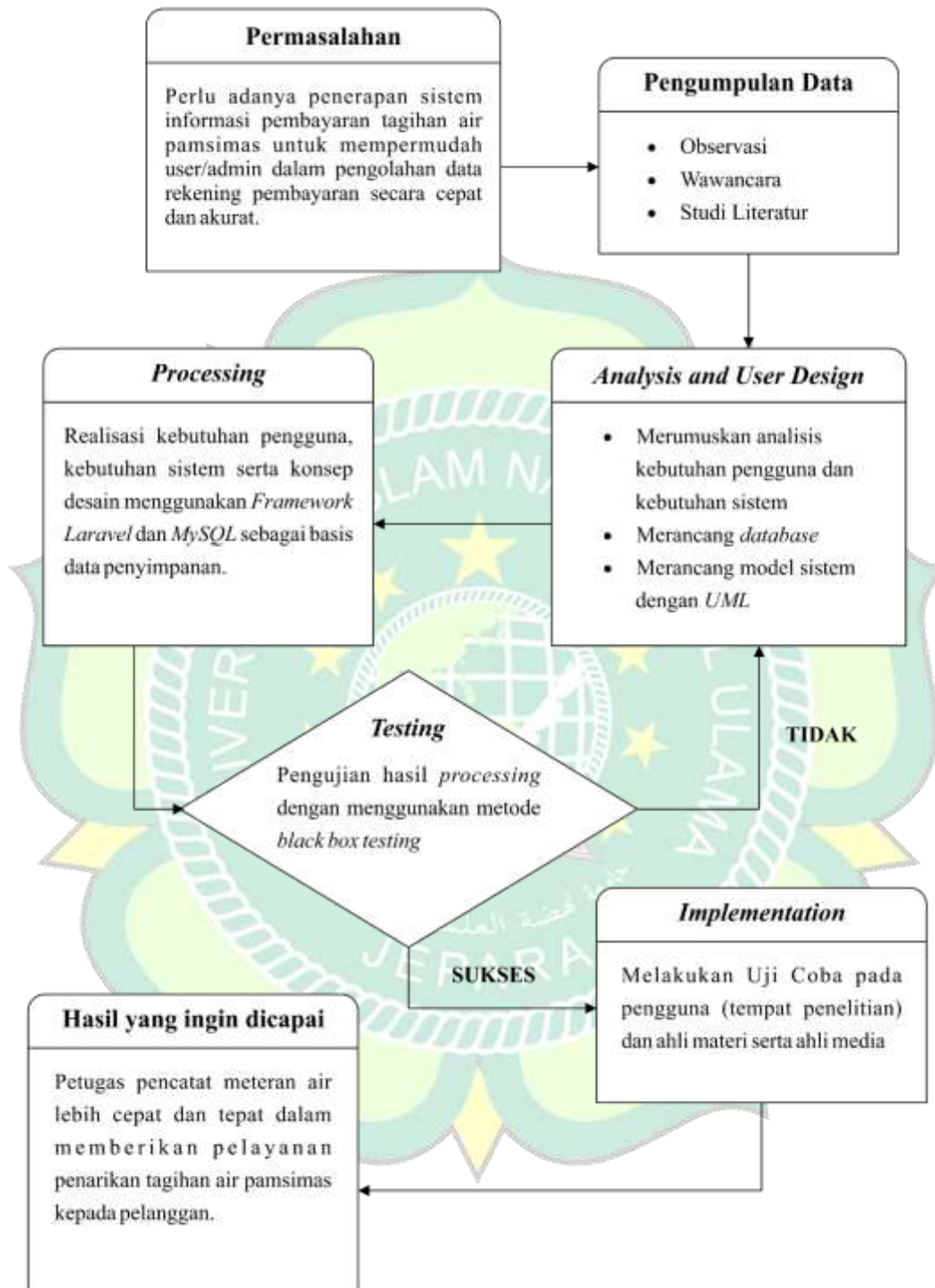
Black box testing adalah teknik pengujian tanpa perlu mengetahui struktur internal dari suatu *software* yang akan diuji karena pengujian ini hanya berfokus kepada masukan dan keluaran terhadap spesifikasi suatu *software*.

Menurut Mustaqbal, dkk (2015), *blackbox testing* berfokus pada spesifikasi fungsional dari perangkat lunak, kumpulan kondisi input dan melakukan pengetesan pada fungsional program. Tujuan dari *blackbox testing* itu sendiri adalah untuk menemukan keadaan dimana perangkat lunak tidak berjalan sesuai dengan spesifikasinya tanpa melihat struktur internal dari perangkat lunak tersebut.



Gambar 1.7 Black Box Testing

1.3. Kerangka Pemikiran



Gambar 1.8 Kerangka Pemikiran