

BAB II

LANDASAN TEORI

1.1 Tinjauan Studi

Tinjauan studi berguna bagi peneliti untuk di jadikan sebagai referensi dan pegangan penelitian yang selanjutnya akan di buat penelitiannya. Dengan adanya penelitian sebelumnya , memudahkan peneliti untuk membahas sesuai topik yang di bahas. Penelitian sebelumnya juga dapat di jadikan perbandingan penelitian sehingga menghasilkan penelitian baru yang lebih bermanfaat. Penelitian ini merujuk pada beberapa penelitian yang telah di lakukan sebelumnya untuk di jadikan referensi sekaligus menjadi media bertukar informasi, diantaranya.

Menurut Yonathan Dri Handoko dan Aloysius Bagas Pradipta Irianto (2016) Yang berjudul Pengimplementasian Sistem Informasi Manajemen Bank Sampah untuk Meningkatkan Kinerja Usaha Kecil Menengah Di Yogyakarta. Dalam jurnalnya membahas tentang pengelolaan bank sampah yang secara perlahan mulai membangun sebuah infrastruktur pelayanan berbasis teknologi informasi dengan menerapkan sistem informasi untuk melayani transaksi perbankan dan saat ini masih dalam proses penyelesaian, hanya saja fokus dalam penerapan teknologi informasi ini masih mengarah pada sistem transaksi yang melibatkan pihak external yaitu Nasabah dan Bank Sampah, sedangkan untuk aktifitas transaksi lainnya seperti pembukuan, penjualan, laporan, pengajian serta transaksi non perbankan masih menggunakan metode manual. Untuk bisa menentukan kebutuhan yang tepat pada bank sampah badegan , perlu di terapkan metode analisis pada kebutuhan pengguna. Dalam pengembangan sistem yang di lakukan oleh penliti adalah menggunakan metode UCD (*User Center Design*) . metode ini adalah metode pengembangan sistem yang menggunakan tahapan pengguna dalam proses pengembangan sistem. Tujuan dari metode ini adalah agar sistem dapat di kembangkan dan dapat berinteraksi sesuai kebutuhan dari pengguna yang sebenarnya.[1]

Menurut Taufiq Andri, Abdillah Gunawan, Renaldi Faiza dalam jurnalnya yang berjudul Sistem Informasi Terintegrasi Pada Proses Pendaftaran dan Menabung di Bank Sampah Induk Cimahi Berbasis Mobile tahun 2016. Dala

jurnalnya menjelaskan tahun 2014 sampai tahun 2016 tercatat bahwa nasabah bank sampah induk cimahi berjumlah 825 sedangkan jumlah penduduk cimahi pada tahun 2014 adalah 579.015 jiwa. Perbandingan antara jumlah nasabah bank sampah dengan jumlah penduduk yang ada di cimahi hanya 0.23% saja yang sudah menjadi nasabah bank sampah. dalam hal ini masyarakat cimahi kurang tertarik untuk menjadi nasabah bank sampah induk cimahi. Dengan adanya perkembangan teknologi mobile ini, dalam mendapatkan informasi dan berkomunikasi lebih mudah dan lebih cepat. Dengan adanya teknologi mobile transaksi jual beli dapat di lakukan dengan lebih mudah dan cepat. Pembeli dan penjual tidak perlu bertemu langsung cukup melakukan transaksi lewat mobile aplikasi dan tidak perlu datang untuk mendaftar menjadi nasabah bank sampah[3].

Menurut Asti Afianti dalam jurnalnya yang berjudul perancangan Aplikasi Bank sampah berbasis Framework codeigniter di desa suwawal timur kecamatan pakis aji kabupate jepara tahun 2016. Dalam jurnalnya menjelaskan tentang bank sampah cermat adalah suatu organisasi tingkat kelurahan yang berada di desa suwawal timur kecamatan pakis aji kabupaten jepara. Pengelolaan sampah ini dapat membantu warga memperoleh penghasilan tambahan yang berupa tabungan simpanan, pencatatan bukti transaksi yang masih manual di sangatlah kurang efektif dan efisien karena sangat merepotkan dalam pencatatan data transaksi. Pada penelitian ini peneliti menggunakan metode RAD (*Rapid Application Development*) sebagai metode pengembangan sistem , metode ini di pilih karena memepersingkat waktu pengembangan selain itu perancangan dan implementasi pada model ini di sesuaikan oleh kebutuhan pengguna. Perancangan aplikasi ini menggunakan *framework* CodeIgniter karena *framework* ini memiliki dokumentasi yang sangat bagus serta ukuran filr yang sangat kecil, selain itu *framework* codeigniter ini juga sangat mudah di intergrasikan karena *framework* ini memberika kemudahan untuk diintegrasikan dengan *library*. Hasil dari aplikasi ini adalah untuk memudahkan admin bank sampah dalam melakukan pengolahan data, seperti: data nasabah, data anggota, maupun data sampah serta transaksi penjualan dan pembelian sampah agar data yang dihasilkan lebih akurat dan cepat[4].

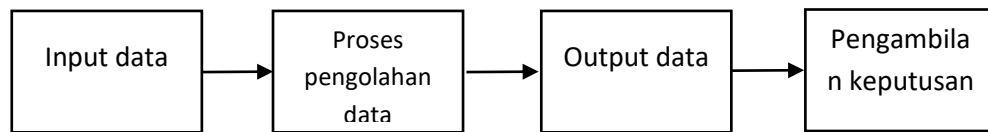
Untuk mengatasi permasalahan yang di hadapi oleh Bank Sampah Prima, peneliti membuat suatu aplikasi bank sampah untuk membantu mempermudah pengelolaan administrasi dan manajemen bank sampah, mulai dari pengelolaan data nasabah sampai pada pembuatan laporan akhir. Aplikasi ini di rancang menggunakan Metode perancangan *Waterfall* dan menggunakan *framework Codeigniter* atau biasa di sebut dengan framework CI. Framework ini memiliki dokumentasi yang bagus dan ukurannya sangat kecil selain itu Framework CI ini mudah untuk diintegrasikan karena CI memberikan kemudahan untuk diintegrasikan dengan *library*.

1.2 Tinjauan Pustaka

1.2.1 Sistem Informasi

Sistem informasi manajemen merupakan proses komunikasi dimana informasi di masukan, di rekam, di simpan dan di proses dengan keputusan mengenai perencanaan, pengoperasian, dan pengawasan[5]. Rangkaian sistem informasi membentuk satu kesatuan sistem dengan tujuan yang sama dengan melalui proses mengumpulkan, pengorganisasian dan pendistribusian data, sehingga data tersebut di olah menjadi informasi yang berguna bagi pengambilan keputusan. Untuk melengkapi suatu sistem informasi manajemen di butuhkan sistem pengoperasian yang terdiri atas *Software* (perangkat lunak), *Hardware* (perangkat keras), *Database* (basis data), prosedur dan juga personil.

Sistem informasi harus tetap terjaga untuk menjamin sistem agar dapat menyiapkan informasi yang di butuhkan oleh client atau user. Sistem yang baik merupakan sistem yang menghasilkan data informasi menjadikan sebagai data yang berfungsi bagi penggunanya. Proses kerja sistem informasi adalah *input* (masukan) data yang di butuhkan suatu organisasi, dan di masukan dalam sebuah sistem informasi yang tersimpan di dalam *database*, selanjutnya data akan di proses yang akan menghasilkan sebuah informasi guna mendukung suatu keputusan dalam *management* berdasarkan kondisi informasi yang di tampilkan.



Gambar 2.1 Proses Kerja Sistem Informasi[5]

1.2.2 Website

Website merupakan salah satu sumber daya internet yang berkembang pesat. Pendistribusian informasi web dilakukan melalui pendekatan *Hyperlink* yang memungkinkan suatu *text, graphic, voice* ataupun object lain yang menjadi acuan untuk membuka halaman-halaman lain. Melalui pendekatan ini, seseorang dapat memperoleh informasi dari halaman ke halaman lain[6]

1.2.3 Framework Codeigniter

1.2.3.1 Pengertian Framework CodeIgniter

Framework adalah kumpulan potongan program yang tersusun dengan rapi, sehingga dapat digunakan untuk membuat aplikasi tanpa harus membuat semua kode program dari awal. Saat ini ada banyak framework PHP, contohnya Yii, CodeIgniter, Zend, Cake PHP, dan lain-lain[7]. Keuntungan menggunakan *framework* adalah waktu untuk pembuatan aplikasi lebih cepat dengan kode program yang lebih mudah dipahami serta tidak perlu fokus dalam satu kode komponen website, serta tidak perlu lagi membuat kode penunjang aplikasi *website* seperti koneksi *database*, validasi *form*, GUI dan keamanan.

Pada penelitian ini, peneliti menggunakan *CodeIgniter framework* dalam mendukung pembuatan dan perancangan sistem. Sebab *CodeIgniter* adalah salah satu *Framework* yang menggunakan bahasa PHP yang membantu mempercepat *developer* untuk mengembangkan aplikasi web dalam pembuatannya dengan banyak fitur yang memungkinkan pengembangan aplikasi web lebih dinamis, tertata, cepat serta mudah dan lebih fleksibel[7]. Metode yang digunakan *Framework CodeIgniter* adalah menggunakan metode MVC atau biasa disebut *model, view, controller*.

Selain mempermudah dalam pengembangan *Codeigniter* mempunyai beberapa keuntungan di antaranya:

1. Di tulis menggunakan PHP 5
2. Berukuran kecil

3. URL lebih sederhana
4. Memiliki *Library* yang lengkap
5. Dokumentasi lengkap dan jelas

1.2.4 MVC (*Model-View-Controller*)

Model-View-Controller (MVC) merupakan sebuah konsep yang di perkenalkan oleh penemu *Smalltalk* (*Trygve Reenskaug*) untuk meng-*enkapsulasi* data bersama dengan pemrosesan (*model*), secara sederhana MVC dapat di artikan sebagai konsep populer dalam pembangunan sebuah aplikasi.[8]

Pola MVC (*Model-View-Controller*) memecahkan sebuah aplikasi menjadi tiga modul asosiasi: *model*, *view*, dan *controller* Yaitu:

1. *Model*

Model di gunakan untuk mengelola informasi dan memberitahu pengamat ketika ada perubahan informasi. *Model* mengandung data dan fungsi yang berhubungan dengan pemrosesan data. *Model* berfungsi sebagai meringkas data dan fungsi yang beroperasi di dalamnya.

2. *View*

View berfungsi sebagai penampil grafis atau tampilan dari sebuah kode yang di buat. *View* memiliki keterkaitan erat dengan *model* karena *view* berfungsi sebagai peampil isi dari *model* dan menampilkan data ke layar. Selain itu ketika *model* berubah maka secara otomatis *view* akan menampilkan ulang dari kode yang di rubah tersebut.

3. *Controller*

Merupakan penghubung antara *model* dan *view*. Dalam *controller* terdapat class dan fungsi-fungsi yang memproses permintaan dari *view* ke dalam struktur data ke dalam *model*. *Controller* juga tidak boleh berisi kode program yang mengandung data. Tugas dari *controller* adalah menyediakan variable yang akan di tampilkan ke dalam *view*, memanggil *model* untuk melakukan akses ke database, dan menyediakan validasi untuk mengecek inputan.

1.2.5 Bank sampah

Bank sampah adalah tempat menabung dengan sampah, dimana sampah yang di tabung pada bank sampah adalah sampah jenis anorganik yang mempunyai nilai jual. Sampah yang memiliki nilai jual adalah sampah yang dapat di daur ulang menjadi produk baru[9]. Sistem yang di gunakan oleh bank sampah adalah sama seperti bank pada umumnya, hanya saja object yang di tabung bukanlah uang melainkan sampah yang layak jual seperti, botol bekas, kardus, plastik, botol kaca dll. Sampah ini nanti akan di setorkan ke pada petugas bank sampah dan petugas akan menimbang serta mencatat berat sampah yang di hasilkan dan nantinya akan di catat di buku rekening oleh petugas bank sampah. dari hasil setor sampah tersebut nasabah akan menerima uang dari hasil menjual sampah kepada bank sampah. Hasil yang di peroleh dari penjualan sampah tersebut dapat di simpan jangka panjang atau di ambil langsung.

1.2.6 PHP (HyperText Preprocessor)

PHP atau biasa di sebut *HyperText Preprocessor* adalah bahasa pemrograman web server yang bersifat *open source*. PHP merupakan script yang terintegrasi oleh HTML dan berada pada server (*server site HTML embedded scripting*). PHP juga di gunakan untuk membuat website dinamis yang berarti tampilan di buat saat halaman di minta oleh client. Mekanisme ini menyebabkan informasi yang di terima oleh clien selalu up to date[10]

1.2.7 Web Server

Web server merupakan salah satu software yang beroperasi untuk mendistribusikan web page ke user/client dengan menggunakan protocol komunikasi tertentu[11]. *Web server* juga di gunakan untuk menjalankan program yang berbasis PHP yang sudah terintegrasi aplikasi luar dan menjadi bagian dari web server. Secara garis besar web server di gunakan untuk menangani pemrosesan dan penanganan permintaan *website*. HTML di proses oleh web server dengan pengaksesan sebuah website dengan mengetikan URL atau halaman web browser, setelah di terima dan dapat di akses di web browser selanjutnya meminta kases kepada web browser, lalu mengirimkan data tersebut untuk di terima atau di tolak jika yang di minta tidak di temukan.

1.2.8 Bank Sampah Prima

Bank sampah prima adalah bank sampah yang berdiri pada tahun 2016 yang terletak di Jl. Raya Kudus-Pati, Ngembal Rejo, Ngembal rejo, Kecamatan Jekulo Kabupaten Kudus, Jawa tengah. Awal berdiri bank sampah yang di kelola swadaya oleh masyarakat setempat adalah melihat dengan keprihatinan warga terhadap limbah sampah dan barang bekas yang setiap hari semakin banyak di hasilkan. Bank sampah prima memiliki kepengurusan serta anggota.

Tabel 2.1 : Kepengurusan Bank Sampah Prima

No	Jabatan	Nama
1	Ketua	Ali Sufyan
2	Sekretaris	Elis Supartika
3	Bendahara	Sulamah
4	Teller	Ghonimah
5	Kreasi	Ela
6	Sie. Umum	1. Dina 2. Salsa 3. Sania 4. Mira 5. Gita 6. Rahmi 7. Oka 8. Amara
7	Anggota	1 Alina Fitriani 2 Ela Fidayana 3 Elis Supartika 4 Ema Rochana 5 Hanik Hidayati 6 Hidayati Noor 7 Nikmah 8 Lestari Pujiastuti



		9	Mariyah
		10	Khamidah
		11	Sulamah
		12	Tutut Handayani
		13	Gonimah
		14	Niswah
		15	Khotijah
		16	Rusmiyati
		17	Ririn
		18	Iis suntari
		19	Wahyuni
		20	Tumarti
		21	Siti Nurazizah
		22	RG
		23	PKK RT 7
		24	PKK RW V
		25	Pondom Darunnajah
		26	Lilis Ibrahim
		27	Hidayatul M
		28	Mega
		29	Ali Sofyan
		30	Umi Hanifah
		31	Dian Lestari
		32	Maulina Anisatin
		33	Ajeng Huni Firdausa
		34	Ekawati
		35	Romlah
		36	Murwati
		37	Sofia
		38	Nikmatun Hadanah
		39	Nurhasanah

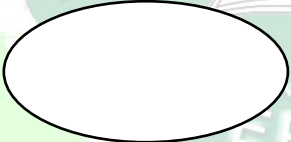
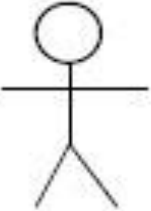
1.2.9 Permodelan UML

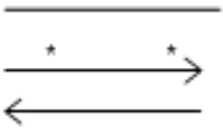
Unified Modelling Language (UML) merupakan sebuah bahasa dimana untuk menampilkan grafis, atau gambar untuk memvisualkan atau menampilkan, membangun dan pendokumentasian dari sebuah sistem yang di bangun dan di kembangkan oleh perangkat lunak berbasis objek[3]. Ketika kita membuat menggunakan konsep ini ada aturan yang harus di ikuti bagaimana elemen yang kita buat saling berhubungan dengan yang lain dengan mengikuti standar yang ada. Konsep UML ini tidak hanya sekedar menampilkan diagram tetapi juga menceritakan konteks yang di buat.

1.2.9.1 Use Case Diagram

Use Case diagram merupakan permodelan yang di gunakan untuk *behavior* (kelakuan) sistem informasi yang di buat yang mendiskripsikan interaksi satu atau lebih actor dengan sistem dalam menggambarkan sebuah fungsi yang di harapkan dalam perancangan yang meliputi actor, use case dan relasi sebagai tindakan pada actor dengan penilaian terukur[7], [12]. Diagram use case juga di gunakan untuk permodelan sebuah perilaku suatu sistem yang di perlukan oleh penggunanya.

Tabel 2.2 Simbol Use Case

NO	SIMBOL	NAMA	DESKRIPSI
1.		<i>Case</i>	Menggambarkan fungsional interaksi antara sistem dengan aktor
2.		<i>Actor</i>	Menunjukkan entitas yang di atau subjek yang melakukan suatu proses

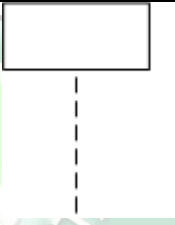
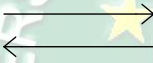
3.		<i>Relasi</i>	Menggambarkan hubungan antar case dengan aktor, atau case dengan case.
----	---	---------------	--



1.2.9.2 *Sequence Diagram*

Sequence Diagram merupakan diagram yang menggambarkan interaksi antar objek di dalam maupun di sekitar sistem dengan berupa pesan yang di gambarkan terhadap waktu. Sequence diagram terdiri antara dimensi waktu dengan objek. Sequence diagram dapat di gunakan untuk menunjukan rangkaian yang di lakukan untuk menghasilkan keluaran tertentu[3]. Berikut symbol-simbolnya:

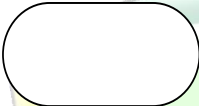




Tabel 2.3 Simbol Squence Diagram

SIMBOL	NAMA	KETERANGAN
	<i>Object</i>	Menggambarkan pos-pos objek yang mengirim dan menerima pesan.
	<i>Message</i>	Berfungsi untuk menggunakan aliran pesan yang dikirim oleh pos-pos objek.

1.2.9.3 Activity Diagram

Activity Diagram atau Diagram Aktivitas di gunakan untuk menggambarkan *workflow* (aliran kerja) atau aktivitas sistem dan menu yang ada pada perangkat lunak. Diagram aktivitas menggambarkan aktivitas sistem bagaimana sistem itu di rakit. Berikut adalah symbol-simbol *activity diagram*:

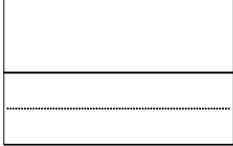



Tabel 2.4 Simbol-Simbol *Activity Diagram*

SIMBOL	NAMA	KETERANGAN
	<i>Action State</i>	Menggambarkan keadaan elemen dalam suatu aktivitas.
	<i>State</i>	Berfungsi untuk menggunakan kondisi suatu elemen.
	<i>Flow Control</i> dan	Penghubung aliran aktivitas dari elemen satu ke elemen lain
	<i>Initial State</i>	Titik awal dari suatu elemen.
	<i>Final State</i>	Titik akhir dari suatu elemen.

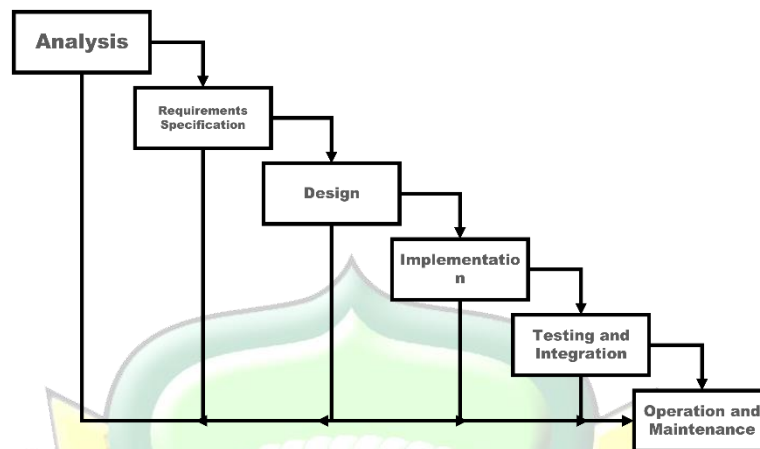
1.2.9.4 Class Diagram

Class Diagram merupakan diagram yang menggambarkan struktur untuk mendefinisikan kelas-kelas dalam perancangan sistem. Di dalam *class diagram* terdapat atribut atau property dan juga metode. Class diagram memiliki symbol sebagai berikut:

Tabel 2.5 Simbol-simbol *class diagram*

SIMBOL	NAMA	KETERANGAN
	<i>Class</i>	Blok-blok pembangun yang ada pada pemrograman berorientasi objek.
	<i>Association</i>	Garis penghubung (relasi) antar class
	<i>Dependency</i>	Menunjukkan operasi pada suatu kelas yang menggunakan kelas lain.
	<i>Aggregation</i>	Penghubung antar class dengan makna untuk semua bagian. Digunakan jika kelas yang satu adalah bagian semua dari kelas lain.

1.3 Metode Waterfall



Gambar 2.2 Tahapan Metode Waterfall[13]

Metode *Waterfall* adalah metode yang di cetuskan pada tahun 1970 sebagai contoh metodologi pengembangan perangkat lunak atau sistem yang tidak bekerja secara baik[14]. Metode *waterfall* merupakan model yang sering di gunakan sebagai pengembangan sistem karena di anggap sebagai pendekatan klasik dalam pengembangan sistem

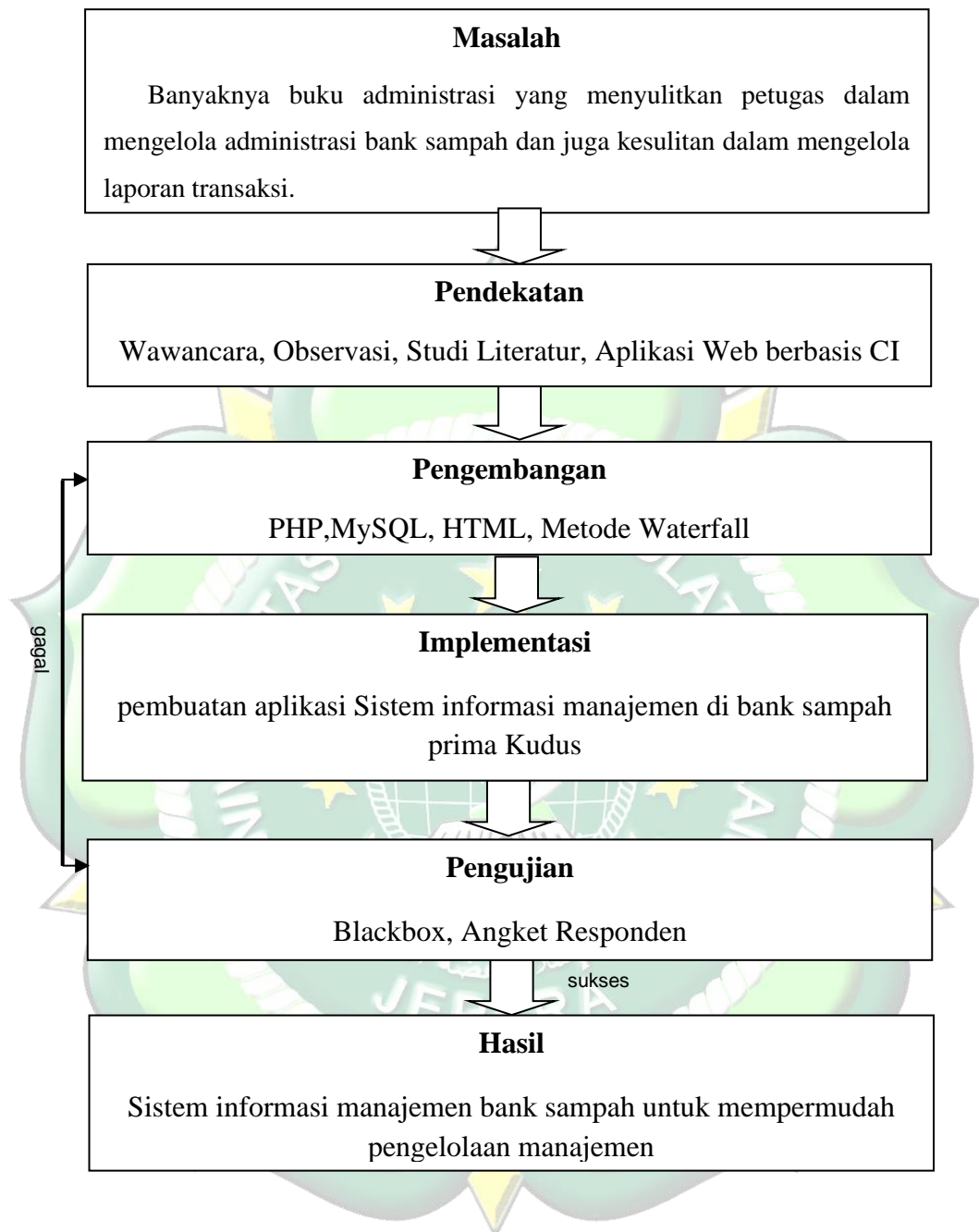
1.4 Black Box Testing



Gambar 2.3 Alur Black Box

Black Box Testing adalah pengujian sistem yang berfokus pada persyaratan fungsional perangkat lunak. Dalam pengujian *Blackbox Testing* ini dapat di temukan kesalahan seperti fungsi-fungsi yang salah, kesalahan *interface*, kesalahan struktur data atau database external, kesalahan kerja, instalasi dan kesalahan terminasi[15].

1.5 Kerangka Pemikiran



Gambar 2.4 Kerangka Pemikiran