

BAB II TINJAUAN PUSTAKA

2.1 Sistem Informasi

Sistem Informasi adalah sistem yang terdiri dari orang-orang dan komputer yang memproses atau menafsirkan informasi. Sistem informasi ini menyangkut *hardware, software, insfrastruktur* dan orang-orang. Sistem merupakan Sekumpulan Unsur/ elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan. Sistem juga terdiri dari beberapa komponen yang saling berhubungan dan bekerjasama untuk mencapai sasaran atau tujuan, sebagai contoh yaitu sistem pembelajaran terdiri dari guru, siswa, tujuan, metode, materi hingga penilaian (Wijoyo *et al.*, 2021).

2.2 Pelayanan Administrasi Terpadu

Pelayanan administrasi terpadu merupakan suatu penyelenggaraan pelayanan publik di kecamatan dimana dalam proses pengelolaannya mulai dari permohonan sampai ketahap terbitnya dokumen dilakukan dalam satu loket atau meja pelayanan(Ardiansyah, 2019) .

2.3 Kecamatan Kemiling

Kemiling adalah sebuah kecamatan di Kota Bandar Lampung, Lampung, Indonesia. Letaknya yang berada di daerah perbatasan Kota Bandar Lampung-Kabupaten Pesawaran membuatnya sangat strategis. Fasilitas pendidikan yang ada di sini cukup banyak. Ada 2 Universitas besar ada di sini, yaitu Universitas Sang Bumi Ruwai Jurai (USBRJ) dan Universitas Malahayati. Berdasarkan kecamatan tersebut terdapat data kelurahan terdaftar seperti berikut:

1. Beringin Raya, dengan kode pos 35158
2. Kedaung, dengan kode pos 35158
3. Kemiling Permai, dengan kode pos 35158
4. Pinang Jaya, dengan kode pos 35158
5. Sumber Agung, dengan kode pos 35158
6. Sumber Rejo, dengan kode pos 35153

7. Beringin Jaya, dengan kode pos 35158
8. Kemiling Raya, dengan kode pos 35158
9. Sumber Rejo Sejahtera, dengan kode pos 35153

2.4 CodeIgniter

CodeIgniter adalah sebuah *framework* php yang bersifat *open source* dan menggunakan metode MVC (*Model, View, Controller*) untuk memudahkan *developer* atau programmer. Dibandingkan *web framework* yang lain *CodeIgniter* memiliki desain yang lebih sederhana dan bersifat tidak kaku. Sebuah *framework* php yang bersifat gratis dan berukuran kecil merupakan keunggulan tersendiri jika dibandingkan framework lain (Supardi dan Hermawan, 2019).

2.5 Website

Website merupakan halaman yang menampilkan informasi data teks, gambar, suara, video atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis. Halaman pertama sebuah *website* disebut *homepage*. *Website/Situs* merupakan kumpulan informasi atau kumpulan halaman/*page* yang bisa diakses lewat jalur *internet*. Setiap orang di berbagai tempat dan segala waktu bisa menggunakannya selama terhubung secara *online* (Harianto, *et al.*, 2019).

2.6 PHP

PHP (*Personal Home Page*) adalah pemrograman (interpreter) yang melakukan proses penerjemahan baris sumber menjadi kode mesin yang dimengerti oleh komputer secara dinamis. Pengertian PHP juga merupakan singkatan dari *Hypertext Preprocessor* dengan Bahasa yang berbentuk skrip yang bersifat *server side* yang dimana proses pengerjaan kode program dilakukan di *server*, dan hasilnya akan ditampilkan di *browser* (Sihombing, 2022).

2.7 MySql

MySQL adalah sebuah *database management system* (manajemen basis data) menggunakan perintah dasar SQL (*Structured Query Language*) yang cukup terkenal. *Database management system* (DBMS) MySQL multi pengguna dan bersifat gratis. Mysql digunakan sebagai wadah dalam mengelola data yang

dapat disimpan didigunakan kembali dengan cara yang lebih efisien (Setyawan and Pratiwi, 2019).

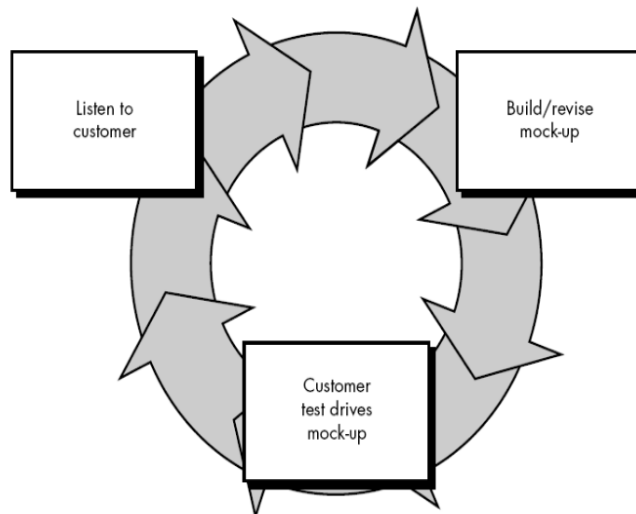
2.8 Metode Pengembang Sistem

Metode pengembang sistem merupakan metode yang digunakan sebagai alur proses dalam pengembangan, sehingga penelitian dapat di kembangkan sesuai tahapan dari metode pengembang sistem (Rosa dan Shalahuddin, 2019).

2.8.1 *Prototype*

Prototype merupakan metode yang digunakan untuk mendapatkan kebutuhan pengguna lebih sepesifik dalam segi teknis. Model prototype dapat digunakan untuk menyambungkan ketidakpahaman pengguna mengenai teknis dan memperjelas spesifikasi kebutuhan yang diinginkan pengguna kepada pengembang perangkat lunak (Rosa dan Shalahuddin, 2019).

Prototype dapat dimulai dengan mengumpulkan kebutuhan pengguna terhadap perangkat lunak yang akan dibuat, kemudian dirancang menggunakan model prototype dalam bentuk mockup untuk mempermudah pengguna mengevaluasi prototype tersebut. Sehingga hasil dari sistem yang dikembangkan telah sesuai dengan permintaan dan kebutuhan pengguna.



Gambar 2.1 Metode Prototype
Sumber : (Rosa dan Shalahuddin, 2019)

1. Kelebihan Prototype
 - a. Menghemat waktu dan Biaya pengembangan
 - b. Adanya keterlibatan pemilik sistem sehingga kesalahan sistem bisa diminimalisir dari awal proses
 - c. Membantu anggota tim untuk berkomunikasi secara efektif
 - d. Klien memiliki kepuasan tersendiri karena sudah memiliki gambaran dari sistem yang akan dibuat.
 - e. Implementasi atau penggunaan sistem lebih mudah karena klien sudah tahu gambaran sistem sebelumnya
 - f. Kemudahan dalam memperkirakan pengembangan sistem selanjutnya
 - g. Memungkinkan klien untuk mempersiapkan perangkat lunak yang cocok dengan sistem yang akan dibuat.

2. Kelemahan Prototype
 - a. Prototype adalah metode yang menghabiskan banyak waktu jika klien kurang puas ditahapan awal.
 - b. Klien terus menerus menambah *requirement* dari sistem, pegen dibuatkan yang seperti inilah seperti itulah, sehingga menambah kompleksitas pembuatan sistem.
 - c. Sistem akan terhambat jika komunikasi kedua belah pihak tidak berjalan secara efektif.

2.8.2 Tahapan Penelitian

Tahapan dalam penelitian sebagai langkah-langkah penelitian yang harus dikerjakan, berikut adalah tahapan penelitian menggunakan prototype.

1. Mendengarkan pelanggan
Pelanggan dan pengembang bersama sama mendefinisikan format seluruh perangkat lunak, semua kebutuhan dan garis besar sistem yang akan dibuat.
2. Membangun atau memperbaiki *mockup*
Dengan membuat perancangan sementara seperti *mockup* yang berfokus pada penggambaran terhadap sistem yang dibangun kepada pengguna (Misal membuat rancangan *input* dan *output*).

3. Pelanggan melihat atau menguji *mockup*

Penggambaran sistem yang digunakan seperti *mockup* perlu di uji kepada pengguna untuk mendapatkan kebutuhan yang sesuai dengan keinginannya, selanjutnya pada sistem yang telah dibangun perlu diuji untuk mengetahui kelayakan terhadap fungsi sistem.


2.9 Alat Pengembang Sistem

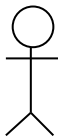

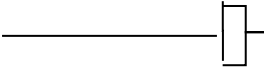
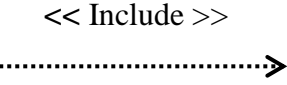
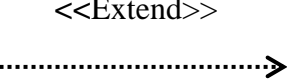
Alat pengembang sistem merupakan konsep desain yang digunakan untuk menggambarkan sistem dengan menggunakan diagram (Agarina and Karim, 2019). Penyesuaian alat yang digunakan harus sesuai dengan metode pengembangan yang dilakukan salah satunya adalah penerapan *Unified Modelling Language*. Menurut Rosa dan Salahuddin (2019), *Unified Modelling Language* adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. Berikut ini merupakan penjelasan tentang masing-masing diagram yang ada pada *Unified Modelling Language*.

2.9.1 Use Case Diagram

Menurut Rosa dan Salahuddin (2019), *use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Use Case Diagram* dapat dilihat pada tabel 2.1 :

Tabel 2.1 Simbol *Use Case Diagram*


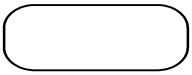
No	Simbol	Deskripsi
1.		<i>Use case</i> : Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i> .

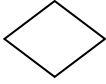


No	Simbol	Deskripsi
2.		Aktor: seseorang/sesuatu yang berinteraksi dengan yang akan dibuat. diluar sistem informasi. Biasanya dinyatakan menggunakan kata benda
3.		Asosiasi (<i>association</i>): merupakan komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.		Generalisasi (<i>generalization</i>): merupakan hubungan (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum
5.		Include berarti <i>use case</i> yang ditambahkan akan dipanggil saat <i>use case</i> tambahan dijalankan.
6.		Ekstensi (<i>extend</i>) merupakan <i>use case</i> tambahan ke sebuah <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.

2.9.2 Activity Diagram

Menurut Rosa dan Salahuddin (2019), *activity* diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *activity diagram* dapat dilihat pada tabel 2.2:

Tabel 2.2 Simbol *Activity Diagram*



No.	Simbol	Keterangan
1.		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.

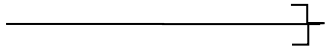
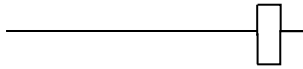
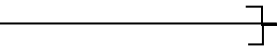
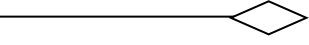
No.	Simbol	Keterangan		
3.		Percabangan (<i>Decision</i>) merupakan asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.		
4.		Penggabungan (<i>Join</i>) merupakan asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.		
5.	<table border="1" data-bbox="392 663 659 779"> <tr> <td>Nama swimlane</td> </tr> <tr> <td> </td> </tr> </table>	Nama swimlane		Swimlane Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas.
Nama swimlane				
6.		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.		

2.9.3 Class Diagram

Menurut Rosa dan Salahuddin (2019), *Class diagram* mengembangkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Class Diagram* dapat dilihat pada tabel 2.3:

Tabel 2.3 Simbol *Class Diagram*

No.	Simbol	Deskripsi			
1.	<table border="1" data-bbox="392 1368 639 1541"> <tr> <td>Nama_kelas</td> </tr> <tr> <td>+Attribute</td> </tr> <tr> <td>+Operasi</td> </tr> </table>	Nama_kelas	+Attribute	+Operasi	Kelas pada struktur sistem.
Nama_kelas					
+Attribute					
+Operasi					
2.	Antar Muka/Interface  Nama_Interface	Sama dengan konsep interface dalam pemrograman berorientasi objek.			
3.	Asosiasi / Association 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan symbol			

No.	Simbol	Deskripsi
4.	Asosiasi Berarah / Directed Association 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan symbol.
5.	Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
6.	Ketergantungan / dependency 	Relasi antar kelas dengan makna ketergantungan antar kelas.
7.	Agregasi / aggregation 	Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>)

2.10 Pengujian *Black Box Testing*

Black box testing menurut Rosa dan Salahuddin (2019) yaitu pengujian perangkat lunak dari segi pendefinisian fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Pengujian yang dilakukan dengan membuat kasus yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji dilakukan harus dibuat dengan benar dan salah, seperti proses *login* “Jika user memasukkan *username* dan *password* yang benar maka dapat *login* ?”.

2.11 Tinjauan Pustaka

Tinjauan pustaka merupakan sumber referensi yang diambil dari jurnal penelitian yang berkaitan dengan penelitian, berikut penjelasan tinjauan pustaka:

Tabel 2.4 Tinjauan Pustaka

No.	Penulis dan Tahun	Judul	Metode	Hasil
1	(Ardiansyah, 2019)	Pengembangan Sistem Informasi Registrasi Pelayanan Administrasi Terpadu Kecamatan (Paten) Di Kecamatan Tarogong Kidul Kabupaten Garut	Model perancangannya waterfall	Hasil penelitian memudahkan pengguna karena adanya aplikasi Registrasi PATEN, proses pengecekan dan penyimpanan bisa berjalan optimal karena didalamnya meliputi proses pengecekan, penyimpanan data, input, edit, hapus, dan memudahkan dalam pembuatan laporan
2	(Ananda Gayatri <i>et al.</i> , 2021)	Sistem Informasi Pendataan Dan Pengarsipan Berkas Pelayanan Administrasi Terpadu Kecamatan (Paten) Di Kecamatan Sumbawa Dengan Metode Waterfall	Metode Waterfall	Hasil akhir dari penelitian adalah Sistem Informasi Pendataan dan Pengarsipan Berkas Pelayanan Administrasi Terpadu Kecamatan (PATEN) di Kecamatan Sumbawa Berbasis Web yang dapat mengelola data secara efektif dan efisien sehingga mempermudah dalam melakukan pelaporan
3	(Handrina, 2021)	Pelayanan Administrasi Terpadu Kecamatan (PATEN) (Studi di Kantor Kecamatan Aur Birugo Tigo Baleh Kota Bukittinggi)	Deskriptif dengan menggunakan pendekatan kuantitatif	Berdasarkan hasil penelitian ini dapat dijadikan sebagai acuan untuk kantor Kecamatan Aur Birugo Tigo Baleh untuk menunjang terciptanya visi dan misi yang telah dibuat oleh Kantor Kecamatan ABTB

No.	Penulis dan Tahun	Judul	Metode	Hasil
4	(Nurjoko, Damayanti and Kurniawan, 2018)	Sistem Informasi Layanan Administrasi Kependudukan dengan Konsep Smart Village 1	Waterfall,	Sistem informasi ini dapat memproses pengolahan data penduduk untuk menampilkan informasi data penduduk, tingkat kesejahteraan penduduk dan pelayanan administrasi surat-menyurat yang dilakukan secara online one service oleh warga masyarakat Desa Puralaksana Way Tenong.