

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Sistem Informasi**

Sistem informasi adalah cara-cara yang diorganisasi untuk mengumpulkan, memasukkan, mengolah, menyimpan data serta untuk mengendalikan serta melaporkan informasi sedemikian rupa sehingga sebuah organisasi dapat mencapai tujuan yang telah ditetapkan (Fadlullah et al., 2022).

Sistem informasi merupakan suatu asosiasi terdiri dari beberapa modul yang saling terintegrasi dimana menyajikan informasi dan pengelolaan data untuk disajikan sesuai dengan kebutuhan user, database, source code dan model design mewakili dari untaian suatu rancangan system informasi berguna untuk mempermudah dalam pengembangan dan maintenance (Maciej Serda et al., 2015)

#### **2.2 Arsip Digital**

Arsip digital adalah arsip yang diciptakan, digunakan, dan dipelihara sebagai bukti transaksi, aktivitas, dan fungsi Lembaga atau individu yang ditransfer dan diolah dengan system computer (Simangunsong & Informatika, 2018), penyimpanan arsip dalam bentuk digital merupakan salah satu alternative solusi dalam praktik pengelolaan arsip manual. Sistem penyimpanan arsip adalah rangkaian prosedur yang digunakan untuk menyimpan seluruh rekaman sebuah lembaga, perorangan, dan organisasi. Setiap pelaksanaan kegiatan sebuah lembaga ditulis secara lengkap dan detail. Sehingga, arsip ini bisa dijadikan sebagai sumber informasi akurat dan terpercaya. Tujuan penyimpanan arsip secara umum adalah agar mempercepat dan

- Arsip Dinamis adalah arsip yang digunakan secara langsung dalam kegiatan pencipta arsip dan disimpan selama jangka waktu tertentu. Arsip dinamis dibagi menjadi 3 jenis yaitu arsip aktif, arsip inaktif, dan arsip vital.
- Arsip Statis adalah arsip yang tidak dipergunakan secara langsung untuk perencanaan penyelenggaraan kehidupan berbangsas pada umumnya, maupun untuk penyelenggaraan administrasi negara.

Didalam pengorganisasian arsip, hal yang harus diperhatikan adalah siapa yang menjalankan arsip dalam suatu organisasi agar dapat diketahui dengan jelas siapa penganggung jawabnya. Dengan demikian pembagian tugas dan wewenang

pengelolaan arsip dapat dilaksanakan sebaik-baiknya, adanya kejelasan siapa yang mengelola dan yang bertanggung jawab dapat dipastikan

Sistem pengarsipan adalah cara pengaturan penyimpanan arsip secara logis dan sistematis dengan memakai abjad, numerik, huruf atau kombinasi huruf dan nomor sebagai indentitas arsip yang terkait.

Adapun macam-macam pengarsipan sebagai berikut:

- a. Sistem abjad, system abjad adalah system penyimpanan arsip dengan menggunakan metode penyusunan menurut abjad. Umumnya dipakai untuk penyusunan terhadap nama orang, nama Perusahaan, nama tempat, nama benda dan subjek.
- b. sistem perihal (subjek), system perihal adalah system penyimpanan dan penemuan Kembali surat berpedoman pada perihal surat atau pokok isi surat, isi dokumen sering disebut perihal, pokok masalah, permasalahan, masalah surat dan subjek.
- c. Sistem nomor, system nomor adalah system penyimpanan dokumen berdasarkan kode nomor sebagai pengganti dari nama orang atau nama badan. Hampir sama dengan system abjad yang penyimpanan dokumen berdasarkan kepada nama, system nomor pun penyimpanan dokumen berdasarkan nama, hanya saja disini diganti dengan kode nomor.

- d. Sistem geografis, system geografis adalah suatu penyimpanan arsip berdasarkan pembagian wilayah atau daerah yang menjadi Alamat suatu surat. Surat disimpan dan dikemukakan kembali menurut kelompok atau tempat penyimpanan berdasarkan geografi, wilayah atau kota dari surat berasal dan tujuan surat dikirim.

Dalam pengelolaan arsip dikenal tiga azas sentralisasi, azas desentralisasi, dan azas kombinasi antara sentral dan disentral. Sentralisasi berarti penyimpanan arsip yang dipusatkan disatu unit kerja khusus yang lazim disebut sentral arsip atau pusat arsip. Dengan sentralisasi arsip maka semua surat-surat kantor yang selesai diproses akan disimpan di sentral arsip. Desentralisasi dalam pengelolaan arsip berarti semua unit kerja mengelola arsipnya masing-masing. Azas kombinasi berarti menggabungkan kedua azas sebelumnya. Azas ini diterapkan dalam rangka mengatasi kelemahan yang ada pada azas sentra dan desentra yang sering dijumpai dalam pengelolaan arsip diperkantoran. Dalam penerapan azas kombinasi, pengelolaan arsip aktif dilakukan secara sentralisasi sedangkan inaktif dilakukan secara desentralisasi.

### **2.3 Pemrograman**

Pemrograman komputer adalah proses pembuatan program atau aplikasi yang digunakan oleh komputer atau perangkat lunak lainnya untuk melakukan tugas-tugas tertentu. Dalam penelitian ini pemrograman yang digunakan adalah XAMPP dan PHP. (Santori:2007)

## 2.4 Database

### 1. MYSQL

MySQL merupakan program database yang dapat mengirim dan menerima data dengan sangat cepat dan dalam multi-pengguna. MySQL memiliki dua lisensi, perangkat lunak *free* dan *shareware*. MySQL adalah sistem manajemen *server* database yang dapat mengelola banyak pengguna dan dapat memproses beberapa pernyataan dari beberapa pengguna secara bersamaan (Talango, 2020).

### 2. Framework Codeigniter

Codeigniter adalah sebuah framework untuk web yang dibuat dalam format PHP. Format yang dibuat ini selanjutnya dapat digunakan untuk membuat sistem aplikasi web yang kompleks. Codeigniter dapat mempercepat proses pembuatan web yang kompleks, karena semua class dan modul yang dibutuhkan sudah ada dan programmer hanya tinggal menggunakannya kembali pada aplikasi web yang akan dibuat (Nilawati et al., 2019).


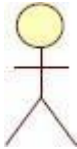

## 2.5 Tools Perancangan Sistem

### 1. Unified Modeling Language (UML)

Menurut (Hendini, 2016), Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang digunakan untuk dokumentasi, spesifikasi, dan pengembangan perangkat lunak. Selain sebagai alat untuk memfasilitasi pengembangan sistem, UML menyediakan teknik untuk membuat sistem berbasis objek. Dalam UML terdapat tipe-tipe diagram UML yaitu :

- *Use Case Diagram*

*Use case* diagram merupakan pemodelan yang digunakan untuk membuat perilaku sistem informasi dalam menentukan fungsionalitas mana yang tersedia dalam sistem informasi dan siapa yang berwenang untuk menggunakannya. Simbol-simbol yang digunakan dalam *Use Case Diagram* dapat dilihat pada tabel 2.1 di bawah ini.



Gambar	Keterangan
<p><b>Use Case</b></p> 	<p><i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, yang dinyatakan dengan menggunakan kata kerja.</p>
<p><b>Actor</b></p> 	<p><i>Actor</i> atau Aktor adalah <i>Abstraction</i> dari orang atau sistem lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi actor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem.</p>
<p><b>Asosiasi/association</b></p> 	<p>Asosiasi antara aktor dan <i>use case</i>, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan data.</p>
<p><b>Generalization</b></p>	<p>Asosiasi antara actor dan <i>use case</i>, digambarkan dengan tanda panah terbuka</p>
<p>Ekstensi (extend)</p> <p>&lt;-----</p> <p>&lt;&lt;Extend&gt;&gt;</p>	<p><i>Extend</i>, merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi</p>

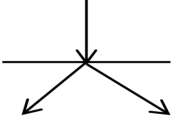
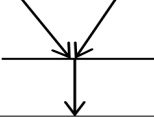
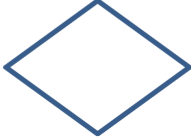

<p><i>Include</i></p> <p>-----&gt;</p> <p>&lt;&lt;Include&gt;&gt;</p>	<p><i>Include</i>, merupakan di dalam <i>use case</i> lain atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program</p>
---	---

**Tabel 2.1** Simbol *Use Case* Diagram

- *Activity Diagram* ( Diagram Aktivitas )

*Activity Diagram* menggambarkan *workflow* atau aktivitas dari sebuah proses, Simbol-simbol yang digunakan yaitu:

Gambar	Keterangan
<p><b>Status Awal</b></p> <p><i>Start Point</i></p> 	<p>Status awal atau <i>start point</i>, diletakkan pada pojok kiri atas dan merupakan awal aktivitas.</p>
<p><b>Status Akhir End</b></p> <p><i>Point</i></p> 	<p>Status akhir atau <i>End Point</i>, merupakan akhir aktivitas.</p>
	<p><i>Activities</i>, menggambar kan suatu proses atau kegiatan bisnis.</p>


	<i>Fork</i> /percabangan, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
<p><b>Percabangan Join</b></p> 	<i>Join</i> (penggabungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi.
<p><b>Decision Points</b></p> 	<i>Decision points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i> .
<p><b>Swimlane</b></p> 	<i>Swimlane</i> , pembagian activity diagram untuk menunjukkan siapa melakukan apa.




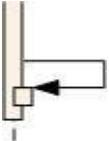


Sumber: (Setiawan & Pasha, 2020)

**Tabel 2.2 Simbol Activity Diagram**

- Diagram Urutan (*Sequence Diagram*)

*Sequence Diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *Sequence Diagram* yaitu:

Gambar	Keterangan
<p><b>Entity Class</b></p> 	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.

<p><b>Boundary Class</b></p> 	<p><i>Boundary Class</i>, berisi kumpulan kelas yang menjadi interfaces atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan form <i>entry</i> dan form cetak.</p>
<p><b>Control class</b></p> 	<p><i>Control class</i>, suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.</p>
<p><b>Message</b></p> 	<p><i>Message</i>, simbol mengirim pesan antar class.</p>
<p><b>Recursive</b></p> 	<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
<p><b>Activation</b></p> 	<p><i>Activation</i>, mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi.</p>
<p><b>Lifeline</b></p> 	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i></p>

**Tabel 2.3 Simbol Sequence Diagram**








- Diagram Kelas (*Class Diagram*)

Diagram kelas merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class diagram* juga menunjukkan atribut- atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan.



*Class diagram* secara khas meliputi : Kelas (*Class*), Relasi *Assosiations*, *Generalitation* dan *Aggregation*, atribut (*Attributes*), operasi (*operation/method*) dan *visibility*, tingkat akses objek eksternal kepada suatu operasi atau attribute.

**SIMBOL CLASS DIAGRAM**

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	<u>Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor</u>
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	<u>Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri</u>
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

**Tabel 2.4 Simbol Class Diagram**

## 2.6 Metode Penelitian

Metodologi yang dikembangkan untuk penelitian ini adalah metode Extreme Programming, metode ini merupakan sebuah proses rekayasa perangkat lunak yang menggunakan pendekatan berorientasi objek dan memiliki beberapa tahapan yang terdiri dari perencanaan, perancangan, pengkodean, dan pengujian. Berikut tahapan-tahapan XP:

- Planning (Perencanaan)

Planning dibagi menjadi 2:

1. Release planning yaitu tim mengidentifikasi dan merencanakan serangkaian rilis produk.

2. Iteration Planning yaitu menentukan apa yang akan diimplementasikan dalam iterasi berikutnya

- Design (Perancangan)

Desain sederhana dan fleksibel, sering kali dengan focus pada pengujian unit.

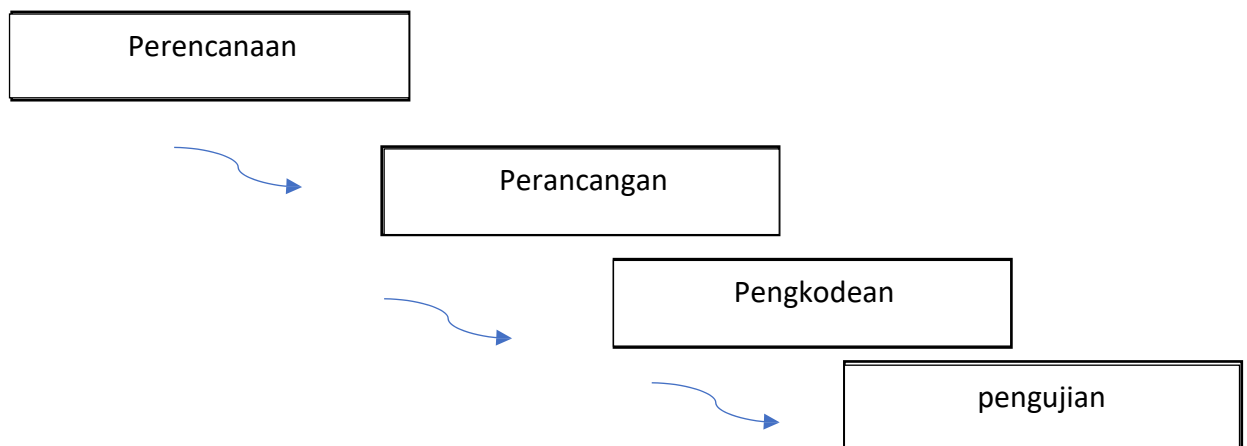
- Coding (Pengkodean)

Pengembang menulis kode dengan standar kualitas tinggi, dengan berfokus pada tes otomatis.

- Testing (pengujian)

Testing dibagi menjadi 2 tahapan:

1. Penguji integrasi yaitu memastikan berbagai bagian system bekerja bersama.
2. Pengujian penerimaan yaitu melibatkan pelanggan atau pemangku kepentingan untuk menguji kesesuaian produk.



Gambar 2.6 Tahapan Extreme Programming

## 2.7 Literatur Review

Dalam penelitian ini menggunakan beberapa penelitian sebelumnya untuk mendukung penelitian yang sedang dilakukan. Berikut merupakan penelitian sebelumnya yang dapat dijadikan tinjauan Pustaka pada penelitian ini dapat dilihat pada table 2.7.

**Tabel 2.7** Literature review

No	Penulis	Judul	Metode	Hasil
1	Ahmad Bahrudin, Jupriyadi, Permata, 2020	Optimasi Arsip Penyimpanan Dokumen Foto Menggunakan Algoritma Kompresi Deflate (Studi Kasus : Studio Muezzart)	Metode <i>Deflate Prototyp e</i>	Sisttm yang dibangun dapat menjadikan aplikasi penyimpanan foto yang mudah digunakan dan dalam proses pencarian data siaran bisa lebih cepat tanpa membuka satu persatu
2	Seputra et al., 2022	Sistem Pengarsipan Dokumen Guru Dan Pegawai Menggunakan Metode Mixture Modelling Berbasis Web	Metode <i>Mixture modelling</i>	Sistem informasi repository dokumen Al-daring dikembangkan dengan memanfaatkan google drive sebagai media penyimpanan dokumen
3	Andi Nurkholis,	Digitalisasi	Metode	Untuk

	Jupriyadi, Arief Budiman, Donaya Pasha, Syaiful Ahdan, Rio Andika, Zahrina Amalia, 2022	Pelayanan Administrasi Surat Pada Desa Bandarsari	<i>Prototype</i>	mempermudah pelayanan administrasi masyarakat secara digital.
4	Puja Irawan, Dimas Aulia Pudjie Prasetya dan Petrus Sokibi, 2020	Rancang Bangun Sistem Pengarsipan Surat Kedinasan berbasis Web Menggunakan Framework Codeigniter	Metode <i>Waterfall</i>	Pemanfaatan teknologi yang dapat memudahkan bagian Tata Usaha untuk melakukan pengolahan dan penyebaran arsip
5	Ajai Rahadi, Ahmadi, Dwi Retnosari, 2022	Sistem Informasi Digitalisasi Dokumen Berbasis Multiuser Di Kecamatan Kelompok Hulu Kabupaten Kotabaru	Metode <i>Waterfall</i>	Fungsi-fungsi yang terdapat pada sistem berjalan dengan baik dan sesuai dengan kebutuhan dan rancangan
6	Dimas Rifqi Ekaryanto, Siti Ramadhani, 2021	Rancang Bangun Sistem Informasi Digitalisasi Manajemen Dokumen Pada Rumah	Metode <i>Prototype</i>	Membantu mencegah proses penyimpanan dan pengambilan dokumen yang diperlukan yang sangat lama

		Sakit Awal Bros Pekanbaru		dalam waktu yang cepat, akurat, dan detail
7	Kodrat Imam Satoso, Adian Fatctur Rochim, Yuli Christyono, Tri Handayani,Ami n Taufiq, Putut 2020	Studi Perbaikan Pengelolaan Perpustakaan Dan Sistim Pengelolaan Arsip & Dokumen Di PT Badak Ngl	Metode <i>Waterfall</i>	Untuk mempermudah pengaksesan informasi berbasis arsip dimanapun kita berada bias mengakses
8	Yudin Wahyudin, Dhian Nur Rahayu, 2020	Analisis Metode Pengembangan Sistem Informasi Berbasis Website: A Literatur Review	Metode <i>System Literature Review</i> (SLR)	Semua metode yang digunakan menghasilkan hasil yang berbeda sesuai metode pengembangan system yang digunakan

