

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Teori Dasar**

##### **2.1.1 Sistem**

Sistem mencakup atas bagian-bagian yang saling berhubungan yang bekerja bersama. Manajemen sistem fisik dan sosial adalah apa yang membuat bisnis dan sistem informasi. (Tyoso, J. S. P. 2016).

Suatu sistem adalah pengklasifikasian yang koheren dari bagian-bagian yang berperilaku dengan cara yang bisa diprediksi dan mempunyai hubungan yang dibutuhkan satu sama lain. (Hamdani, 2019).

##### **2.2.1 Sistem Informasi**

Untuk menghimpun, menyimpan, memproses, dan mengatur data dengan lebih baik, serta mengatur dan menyebarluaskan informasi dengan lebih baik, lembaga bekerja pada sistem komunikasi. Sistem komunikasi ialah sistem yang dipakai di dalam organisasi untuk mengkoordinasikan kebutuhan pelaporan kegiatan strategis dan kebutuhan pemrosesan transaksi sehari-hari untuk mendukung fungsi proses administrasi. (Mardiyati & Cholifah, 2022).

Komponen sistem seperti perangkat lunak, perangkat keras, dan perangkat otak membentuk sistem informasi, yang memproses data menjadi keluaran yang bisa dipakai untuk organisasi. Sebagaimana dikutip oleh (Mulyanto dalam Kuswara dan Kusmana, 2017:18).

##### **2.1.3 Aplikasi**

Perangkat lunak, atau aplikasi seperti yang lebih sering dikenal, adalah program komputer pra-bangun yang menyediakan dokumentasi dan bentuk panduan lain untuk penggunaanya (Syani dan Wersstantia, 2018).

Setiap aspek manajemen memerlukan aplikasi, namun kami memakai TI setiap hari tanpa memikirkannya.

Aplikasi, seperti yang didefinisikan oleh Hasan Abdurahman dan Asep Ririh Riswaya (2014), adalah perangkat lunak pra-paket yang memfasilitasi eksekusi instruksi pengguna untuk memberikan hasil yang lebih akurat selaras terhadap tujuan perangkat lunak yang mendasarinya. Memecahkan masalah dalam program ini memerlukan penggunaan salah satu teknik pemrosesan data bawaan program, yang seringkali didasarkan pada perhitungan dan pemrosesan data yang diinginkan atau diantisipasi.

#### **2.1.4 Android**



**Gambar 2.1 Android**

Android ialah sistem operasi perangkat mobile. OS bertindak sebagai "jembatan" diantara perangkat dan pengguna, memfasilitasi komunikasi diantara kedua pihak dan memfasilitasi penggunaan aplikasi perangkat. (Kuswanto dan Radiansah, 2018).

#### **2.1.5 Android Studio**



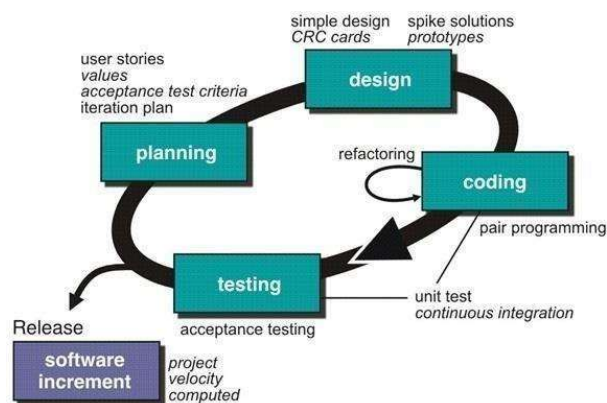
**Gambar 2.2 Android Studio**

Android Studio ialah IDE berbasis IntelliJ IDEA untuk membuat aplikasi Android. Ini berfungsi sebagai editor kode andal dan alat pengembangan IntelliJ, dan menawarkan sejumlah fitur lain untuk meningkatkan produktivitas saat mengembangkan aplikasi Android (Sianturi, Usman, Firmansyah, Telaumbanua, & Widyaningsih, 2019).

### 2.1.6 Basis Data

Basis data adalah kumpulan informasi yang telah disusun secara sistematis, seperti yang dikemukakan oleh (Suharyanto, Chandra, dan Gunawan 2017). Penggunaan, pemutakhiran, dan pemrosesan data yang disimpan dalam basis data komputer membutuhkan sistem manajemen informasi minimal. Dengan perangkat lunak berbasis file, setiap komunitas pengguna bertanggung jawab untuk mengelola kumpulan file datanya sendiri. Redundansi, sering dikenal sebagai duplikasi data, dapat terjadi akibat hal ini. (Suharyanto, Chandra, & Gunawan, 2017)

### 2.1.7 Metode Extreme Programming (XP)



**Gambar 2.3 Metode Extreme Programming (XP)**

Extreme Programming adalah pendekatan rekayasa perangkat lunak populer yang dipakai oleh pemrogram untuk membuat perangkat lunak. Bila proyek pengembangan Anda harus cukup fleksibel untuk mengakomodasi perubahan tak terelakkan yang muncul selama pembuatan aplikasi, XP adalah jalan yang harus ditempuh. (Suryantara, I. G. N., Kom, S., & Kom, M. 2017)

## Langkah Metodologi dalam Pengembangan Perangkat Lunak

Komponen *Extreme Programming* (XP) adalah:

1. *Planning*/Perencanaan

Pada titik ini, kita belajar tentang lingkungan bisnis aplikasi, menentukan keluarannya, fungsinya, fungsionalitas yang akan kita buat, memperkirakan berapa lama dan berapa biayanya, dan memetakan proses pengembangan.

2. *Design*/Perancangan

Pada fase ini, Anda akan membuat antarmuka perangkat lunak dasar. Pada tahap ini, CRC (Class Responsibility Collaborator) mungkin berguna sebagai alat desain. Gunakan diagram kasus, diagram kelas, dan diagram objek semuanya memakai CRC untuk menggambar (membangun) kelas.

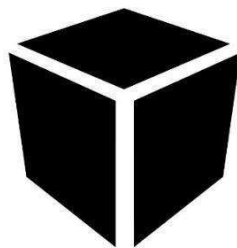
3. *Coding*/Pengkodean

Pemrograman pasangan (di mana dua atau lebih pengembang bekerja sama dalam satu kode) adalah prinsip utama Pemrograman Ekstrim.

4. *Testing*/Pengujian

Di sini, akan membuat aplikasi berfungsi sebagaimana mestinya dan tidak mempunyai masalah apa pun sembari memastikannya selaras terhadap cara klien menjalankan bisnis.

### 2.1.8 *Black Box Testing*



**Gambar 2.4 *Black Box Testing***

Black Box Testing didefinisikan sebagai proses penilaian terhadap keluaran suatu program secara mandiri terhadap pelaksanaannya (STMIK Nusa Mandiri Jakarta, - AMIK BSI Purwokerto, & - AMIK BSI Purwokerto, 2018).

Oleh karena itu, disebut sebagai kotak hitam yang tidak terlihat isinya. Dalam pengujian ini, hanya dilaksanakan untuk kesalahan khusus, fungsi tertentu, fungsi yang ada dalam aplikasi, kesalahan antarmuka tampilan, kesalahan struktur data atau database, kesalahan kinerja, serta inisialisasi dan terminasi program.

## **2.2 Teori Khusus**

### **2.2.1 Laundry**

Pada umumnya usaha jasa laundry merupakan usaha jasa laundry, dalam usaha jasa laundry perlu dilaksanakan pencatatan laporan keuangan. Pencatatan laporan ini menjadi dasar bagi pemilik untuk mengambil keputusan terhadap kelangsungan usahanya.

*Laundry*, mengacu pada definisi Muhammad Syawal Ainul Yaqin (2016), adalah divisi housekeeping yang menangani semua laundry tamu dan staf. Ungkapan "laundry" sering dipakai untuk mengilustrasikan proses pencucian pakaian atau lokasi di mana tugas ini dilaksanakan. Per 2018 (Simargolang & Nasution). Selain merujuk pada ruang fisik tempat seseorang mencuci pakaian, istilah tersebut juga bisa merujuk pada tindakan mencuci pakaian atau tekstil lainnya. (Yaqin, 2016).

### **2.2.2 Java**

Karena kesamaan diantara bahasa Java dan bahasa Inggris, bahasa ini dianggap sebagai bahasa tingkat tinggi (bahasa yang mudah dipahami oleh manusia). Namun, standar yang tepat (sintaksis) dibutuhkan saat menuangkan pemikiran ke atas kertas. (Rusli, Rinarta, Atmojo, & Bali, 2016)

Java adalah bahasa pemrograman berorientasi objek yang bisa dipakai pada komputer desktop dan perangkat seluler (Simanjuntak & Alfisyahri, 2020). Tata bahasa Java diturunkan dari C dan C++ dan kemudian disederhanakan lebih lanjut.

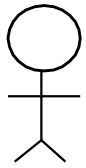
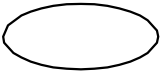
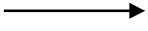


### **2.2.3 UML (Unified Modeling Language)**


Perangkat lunak berorientasi objek bisa dirancang dan dikembangkan dengan bantuan Unified Modeling Language, sebuah pendekatan pemodelan visual. Prosedur dan kelas bisnis dijelaskan secara rinci memakai Unified Modelling Language (UML), suatu bentuk standar penulisan atau cetak biru. Sejak Prihandoyo (2018). Diagram UML yang umum dipakai untuk pengembangan sistem meliputi:

#### A. *Use Case Diagram*

Diagram use case adalah representasi grafis dari kemungkinan interaksi pengguna dengan sistem. Use case diagram menampilkan use case yang berbeda dan tipe pengguna yang berbeda dalam suatu sistem dan seringkali disertai dengan tipe diagram lainnya. (Putra & Putri, 2023). Kasus penggunaan diwakili oleh lingkaran atau *elips*.

**Tabel 2.1** *Use Case Diagram*



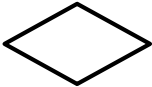


Simbol	Nama	Keterangan
	<i>Actor</i>	Mengilustrasikan peran individu, sistem lain, atau alat ketika berinteraksi dengan <i>use case</i> .
	<i>Use Case</i>	Abstraksi dan keterhubungan diantarasistem dan pelaku.
	<i>Association</i>	Abstraksi dari koneksi diantara pelakudengan <i>use case</i> .
	<i>Generalisasi</i>	Menampilkan keahlian aktor untuk bisa berpartisipasi dengan <i>use case</i> .
	<i>Include</i>	Menampilkan bahwasanya <i>use case</i> sepenuhnya mewakili fungsionalitas dari <i>use case</i> lain.

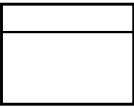
	<i>Extend</i>	Menampilkan bahwasanya use case adalah pelengkap fungsional untuk use case lain bila kondisinya terpenuhi.
---	---------------	--

### ***B. Activity Diagram***

Diagram aktivitas mengilustrasikan pengklasifikasian utas dalam tampilan sistem, dan setiap proses mempunyai struktur. Diagram aktivitas dipakai dalam Unified Modeling Language untuk mewakili proses komputasi dan organisasi, bersama dengan aliran data yang terjadi diantaranya. Diagram aktivitas sistem membantu untuk memahami keseluruhan proses. (Gultom & Simanjuntak, 2021).

**Tabel 2.2** *Activity Diagram*

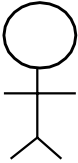

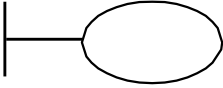
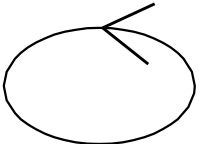
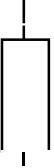
<b>Simbol</b>	<b>Nama</b>	<b>Keterangan</b>
	Status Awal	Selalu ada titik awal dalam diagram aktivitas.
	Aktivitas	Tindakan sistematis, yang dilaksanakan oleh sistem, seringkali dimulai dengan kata kerja.
	Percabangan/ <i>Decision</i>	Ketika banyak pilihan tersedia, cerita bercabang.
	Penggabungan/ <i>Join</i>	Mengkonsolidasikan proses terpisah ke dalam satu operasi.
	Status Akhir	Dalam diagram aktivitas, langkah terakhir mewakili perilaku aktual sistem.

	<i>Swimlane</i>	Perusahaan yang bertanggung jawab berada di swimlane sendiri. sehubungan dengan apa yang terjadi.
---	-----------------	---


### C. *Sequence Diagram*

Dalam rekayasa perangkat lunak, Sequence Diagram dipakai untuk menampilkan waktu dan urutan interaksi antar proses. Untuk mengilustrasikan rangkaian peristiwa yang mengarah pada hasil yang diinginkan, diagram urutan sering dipakai. Ada pedoman untuk memakai setiap diagram urutan yang disediakan. (Gultom & Simanjuntak, 2021).

**Tabel 2.3** *Sequence Diagram*

Simbol	Nama	Keterangan
	<i>Actor</i>	Mengilustrasikan peran individu, sistem lain, atau perangkat saat berinteraksi dengan <i>use case</i> .
	<i>Entity Class</i>	Mengilustrasikan keterkaitan kegiatan yang akan dilaksanakan.
	<i>Boundary Class</i>	Mengilustrasikan sebuah penggambaran didasarkan pada <i>form</i> .
	<i>Control Class</i>	Mengilustrasikan penghubung diantara <i>boundary</i> dengan tabel.
	<i>A Focus Of Control And A Life Line</i>	Mengilustrasikan lokasi awal dan akhir pesan.





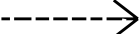


	<i>A Message</i>	Mengilustrasikan pengiriman pesan.
---	------------------	------------------------------------

#### ***D. Class Diagram***

*Class Diagram* ialah semacam diagram UML yang menampilkan kelas sistem, atribut, metode, dan koneksi antar objek untuk mengilustrasikan struktur sistem. Karena diagram kelas menentukan elemen apa dari sistem yang dimodelkan yang dibutuhkan, mereka termasuk dalam kategori diagram struktur. (Pulungan & Purwanto, 2023).

**Tabel 2.4** *Class Diagram*

<b>Simbol</b>	<b>Nama</b>	<b>Keterangan</b>
	Generalisasi	Asosiasi di mana turunan objek mengambil bentuk dan perilaku atasannya.
	<i>Nary Association</i>	Tindakan pencegahan diambil untuk mencegah mengaitkan lebih dari dua hal secara bersamaan.
	<i>Collaboration</i>	Menjelaskan langkah-langkah dalam alur kerja sistem yang menghasilkan hasil yang bisa diamati bagi pengguna.
	<i>Realization</i>	Penyelesaian tugas oleh objek
	<i>Dependency</i>	Sambungan di mana peristiwa yang memengaruhi satu kumpulan komponen akan berpengaruh pada kumpulan elemen lain yang bergantung pada kumpulan pertama itu.

	<i>Association</i>	Item yang menghubungkan dua hal lainnya bersama-sama.
--	--------------------	---

### 2.3 Penelitian Terdahulu

**Tabel 2.5** Penelitian Terdahulu

No	Nama Peneliti	Judul Penelitian	Hasil
1	Pedri Afandi, Rahmat Fauzi	“Implementasi Metode Extreme Programming Dalam Perancangan Aplikasi Penjualan Toko Bangunan Memakai Pemrograman Java”  (ISSN (Online) 2715-6265 Jurnal Comasie)	Bahasa pemrograman Java dan NetBeans IDE dipakai untuk membangun aplikasi, dan desainnya didukung oleh database MySQL. Temuan dari studi yang diimplementasikan menampilkan bahwasanya lebih mudah untuk melaksanakan transaksi penjualan, memasukkan item, dan melacak inventaris di toko Batam Advanced Building.
2	Rezania Agramanisti Azdy, Arsia Rini	“Penerapan Extreme Programming Dalam Membangun Aplikasi Pengaduan Layanan Pelanggan (Palapa) Pada Perguruan Tinggi”  (p-ISSN: 2355-7699 e-ISSN: 2528-6579 Jurnal Teknologi Informasi dan	Studi ini menampilkan bahwasanya memakai Pemrograman Ekstrim untuk pembuatan aplikasi memungkinkan pengembangan cepat dengan ukuran tim yang kecil. Membangun bagian yang paling mendasar, bagian verifikasi akun, memungkinkan pengembangan berlangsung secara cepat, dan kemudian bagian selanjutnya bisa dikonstruksikan. Temuan pengujian kotak hitam memvalidasi kemampuan aplikasi untuk berfungsi seperti yang diharapkan dan untuk memenuhi persyaratan penerimaan cerita pengguna.

		Ilmu Komputer (JTIK))	
3	Anggy Trisnadoli	<p>“Implementasi Extreme Programming (XP) Agile Software Development pada Pengembangan Sistem Informasi KELUARGAKU ”</p> <p>(ISSN: 2541-1004 e-ISSN: 2622-4615 Jurnal Informatika Universitas Pamulang)</p>	<p>Riset ini menampilkan kelayakan metodologi Pengembangan Perangkat Lunak Agile didasarkan pada proses pengembangan aktual yang diterapkan. Selain itu, Sistem Informasi “KELUARGAKU” dapat lebih membantu PUSYANRA Lancang Kuning dalam menjalankan program kerjanya bila penggunaannya mendapatkan nilai maksimal darinya.</p>
4	Ida Bagus Gede Sarasvananda <sup>1</sup> , I Komang Arya Ganda Wiguna <sup>2</sup> , Styawati <sup>3</sup>	<p>“Pendekatan Metode Extreme Programming untuk Pengembangan Sistem Informasi Manajemen Surat Menyurat pada LPIK STIKI”</p> <p>(ISSN: 2541-1004 e-ISSN: 2622-4615 Jurnal Informatika Universitas Pamulang)</p>	<p>Alasan mengapa metode XP dipilih adalah karena sistem bisa dikonstruksikan dalam waktu singkat, dan persyaratan serta proses bisnis peserta sistem sering berubah. Hasil penelitian membuktikan bahwasanya sistem yang dikembangkan bisa berfungsi selaras terhadap harapan pengguna. Hasil dari prosedur pengujian kotak hitam mengkonfirmasi hal ini. Metodologi Extreme Programming telah berhasil dipakai untuk mempercepat pengembangan sistem di dunia nyata, dan sekarang banyak dipakai oleh para akademisi.</p>
5	Noer Azni Septiani, Fauzan Yusuf Habibie	<p>“Penggunaan Metode Extreme Programming Pada Perancangan</p>	<p>Extreme Programming (XP) ialah salah satu dari banyak metodologi yang dipakai dalam pengembangan perangkat lunak; dalam riset ini dipakai untuk membuat sistem informasi pelayanan publik;</p>

		<p>Sistem Informasi Pelayanan Publik”</p> <p>(e-ISSN 2685-998X</p> <p>Jurnal Sistem Komputer dan Informatika (JSON))</p>	<p>PHP adalah bahasa pilihan untuk sistem backend, dan MySQL adalah database pilihan; penelitian menyimpulkan bahwasanya perancangan sistem informasi ini dengan memakai metodologi XP memberikan cara yang paling efektif untuk mengatasi tantangan yang dihadapi oleh pemerintah Desa Sodong di daerah ini.</p>
6	<p>Sri Mardiyati, Wahyu Nur Cholifah</p>	<p>“!Perancangan Sistem Informasi Jasa Pelanggan Laundry Berbasis Android Dengan Metode Personal Extreme Programming!!!”1</p> <p>(P-ISSN: 2746-5411</p> <p>E-ISSN: 2807-5528</p> <p>Jurnal Widya Volume 3, Nomor 2, Oktober 2022: halaman 126-135)</p>	<p>Data dari pembelian yang dilaksanakan oleh pelanggan bisa direkam dan dianalisis memakai sistem ini. Bila Anda melaksanakan ini, akan lebih mudah untuk memperbaiki masalah. Proses penelitian mencakup atas tiga langkah terpisah: mengidentifikasi topik penelitian, menghimpun data yang relevan, dan melanjutkan melalui analisis sistem, desain, pengembangan, dan pengujian. LMIS berbasis Android dikembangkan memakai metodologi PXP (Personal Extreme Programming). Perencanaan, perancangan, pengkodean, dan pengujian ialah tahapan pengembangan perangkat lunak yang membentuk metodologi XP.</p>
7	<p>Evan Susanto, Tri Hartati Utami, Dedy Hermanto</p>	<p>Sistem Informasi Pemesanan Laundry Berbasis Android Di Kota Palembang</p>	<p>Teknik Rational Unified Process dipakai untuk membuat sistem data ini. (RUP). Pada tahap analisis, alat seperti PIECES dan use case diagram dipakai. Diagram kelas dipakai selama fase desain. Java dipakai sebagai bahasa pemrograman utama, dan MySQL adalah basis data pilihan, saat membuat aplikasi. Pemilik perusahaan binatu akan membisa manfaat dari lebih banyak paparan dan layanan yang disederhanakan berkat metode yang baru dikembangkan ini. temukan fasilitas binatu dengan nilai tertinggi di area tersebut.</p>

## 2.4 Kerangka Pemikiran

Pola pikir adalah kerangka mental untuk memecahkan masalah yang dihadapi. Keberhasilan setiap studi kuantitatif bergantung pada kemampuan periset untuk berpikir kritis (Arif, Sukuryadi, & Fatimaturrahmi, 2019). Berikut ini yang menjadi dasar dilaksanakannya penelitian ini:



**Gambar 2.5 Kerangka Pemikiran**