

## **BAB 2**

### **TINJAUAN PUSTAKA**

#### **2.1 Profil Kecamatan**

Kecamatan Regol merupakan satu bagian Wilayah yang berada di Kota Bandung dan dipimpin oleh Camat. Kelurahan yang berada didalamnya yakni Ancol, Balonggede, Ciateul, Ciseureuh, Cigereleng, Pasirluyu dan Pungkur. Kelurahan adalah perangkat Kecamatan yang dibentuk untuk membantu atau melaksanakan sebagian tugas camat. Berdasarkan surat edaran Walikota nomor 213 Tahun 2018, bahwa tugas camat salah satunya yakni melaksanakan dan mengkoordinasikan penyelenggaraan kegiatan pemerintahan yang dilakukan oleh perangkat daerah di tingkat Kecamatan dan Kelurahan.

##### **2.1.1 Sejarah Kecamatan**

Sebelum menjadi Kecamatan, Regol adalah Kelurahan yang terletak di jalan Moch Ramdhan. Karena adanya pemekaran, Regol menjadi sebuah Kecamatan dengan terbagi dalam 3 Kelurahan antara lain Kelurahan Ciateul, Kelurahan Pungkur, dan Balonggede dan sekarang menjadi 7 Kelurahan ditambah dengan Kelurahan Pasirluyu, Kelurahan Cigereleng, Kelurahan Ciseureuh, dan Kelurahan Ancol.

Kecamatan Regol merupakan salah satu bagian wilayah yang berada di Kota Bandung dengan memiliki luas lahan sebesar 430 Ha. Secara administratif Kecamatan Regol dibatasi oleh :

1. Bagian Selatan : Kecamatan Bandung Kidul
2. Bagian Utara : Kecamatan Sumur Bandung
3. Bagian Timur : Kecamatan Lengkong
4. Bagian Barat : Kecamatan Astana Anyar

Secara geografis Kecamatan Regol memiliki bentuk wilayah datar / berombak sebesar 100% dari total keseluruhan luas wilayah. Ditinjau dari sudut

ketinggian tanah, Kecamatan Regol berada pada ketinggian 675 m diatas permukaan air laut. Suhu maksimum dan minimum di Kecamatan Regol berkisar 28 C0 sedangkan dilihat dari segi hujan berkisar 2400 mm/th dan jumlah hari dengan curah hujan yang terbanyak sebesar 45 hari.

Kecamatan Regol merupakan salah satu kecamatan sebelah selatan Kota Bandung dengan luas wilayah 430 Ha, dengan jumlah penduduk 79.646 jiwa terdiri dari 39.747 jiwa laki-laki dan 39.899 jiwa perempuan dengan jumlah Kepala Keluarga di Kecamatan Regol saat ini mencapai sekitar 24.585 KK yang meliputi 371 RT dan 60 RW. Sebagian wilayah Kecamatan Regol terdiri dari pemukiman / pekarangan sedangkan kegiatan ekonominya didominasi oleh jasa perdagangan dan perkantoran.

Berdasarkan data kependudukan dari kecamatan pada tahun 2021 yang dilihat dari segi kepadatan penduduk sebesar 204 jiwa per hektar dan dilihat dari pertumbuhan penduduk, intensitas populasinya akan terus bertambah dari waktu ke waktu.

### **2.1.2 Logo Kecamatan**

Kecamatan Regol memiliki logo kewilayahan yang mana logo tersebut menjadi simbol untuk mencerminkan sebuah Kewilayahan. Berikut adalah logo Kecamatan Regol.



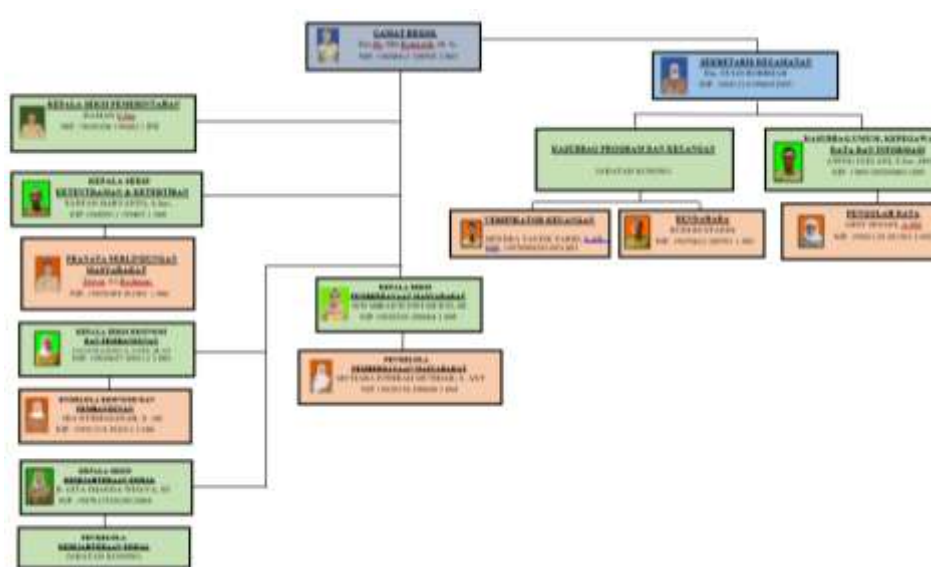
Gambar 2.1-1 Logo Kecamatan Regol

**2.1.3 Visi dan Misi Kecamatan**

Visi dari Kecamatan Regol adalah terwujudnya pelayanan yang prima menuju masyarakat yang maju dan sejahtera di Kecamatan Regol.

Misi dari Kecamatan regol adalah mewujudkan pelayanan publik yang professional, berbasis teknologi informasi, dan meningkatkan akuntabilitas kinerja kecamatan dan kelurahan

**2.1.4 Struktur Organisasi**



Gambar 2.1-2 Struktur Organisasi Kecamatan Regol

**2.1.5 Deskripsi Tugas dan Tanggung Jawab**

Berdasarkan struktur organisasi pada gambar 2.2 maka berikut adalah penjabaran dari deskripsi tugas dan tanggung jawab untuk setiap jabatan yang ada.

Table 2.1-1 Deskripsi Tugas dan Tanggung Jawab

No	Jabatan	Deskripsi Tugas dan Tanggung Jawab
----	---------	------------------------------------

1	Sekretaris Kecamatan	<p>a. merencanakan program, kegiatan dan kinerja lingkup Sekretariat dan mengoordinasikan penyusunan rencana kerja, program kerja dan anggaran Kecamatan berdasarkan kebijakan operasional Kecamatan sebagai pedoman pelaksanaan tugas;</p>
		<p>b. membina dan mendistribusikan tugas kepada bawahan berdasarkan arah kebijakan umum Kecamatan agar tujuan dan sasaran tercapai secara efektif dan efisien;</p>
		<p>c. mengoordinasikan pelaksanaan program dan kegiatan lingkup kesekretariatan yang meliputi pengelolaan umum, kepegawaian, Data dan Informasi, data dan informasi, pengelolaan keuangan, pengoordinasian penyusunan program, serta pengoordinasian tugas-tugas seksi;</p>
		<p>d. mengoordinasikan pelaksanaan fasilitasi, pembinaan dan pengendalian tata naskah dinas lingkup Sekretariat dan Kecamatan;</p>
		<p>e. mengoordinasikan pengelolaan dokumentasi peraturan perundang-undangan, pengelolaan kearsipan, protokol dan hubungan masyarakat di lingkungan Sekretariat dan Kecamatan;</p>
		<p>f. mengoordinasikan penyusunan rencana kebutuhan formasi dan mutasi pegawai;</p>
		<p>g. mengoordinasikan penatausahaan BMD;</p>

		<p>h. mengoordinasikan penatausahaan keuangan Kecamatan;</p>
		<p>i. mengoordinasikan penyusunan bahan penetapan rencana kerja daerah yang meliputi RPJPD, RPJMD, RKPD, Renstra dan Renja, serta rencana kerja lainnya sesuai dengan ketentuan peraturan perundang-undangan;</p>
		<p>j. mengoordinasikan penyusunan bahan penetapan laporan kinerja daerah yang meliputi LKPJ, LPPD, IPPD, LKIP dan laporan lainnya sesuai dengan ketentuan peraturan perundangundangan;</p>
		<p>k. mengoordinasikan pengolahan, penataan dan penyimpanan data dan/atau informasi publik di lingkungan Kecamatan;</p>
		<p>l. mengoordinasikan pelayanan informasi publik;</p>
		<p>m. melaksanakan monitoring, evaluasi dan pelaporan lingkup sekretariat dan Kecamatan; dan</p>
		<p>n. melaksanakan tugas kedinasan lain yang diberikan oleh atasan sesuai dengan tugas dan fungsinya.</p>
2	Sub Bagian Umum, Kepegawaian, Data dan Informasi	<p>a. menyusun rencana program kerja dan bahan penetapan kinerja lingkup Sub Bagian Umum, Kepegawaian, Data dan Informasi;</p>
		<p>b. membina, membagi dan mengarahkan tugas kepada bawahan berdasarkan arah kebijakan umum</p>

		Kecamatan agar tujuan dan sasaran tercapai secara efektif dan efisien;
		c. melaksanakan dan mengoordinasikan administrasi persuratan yang meliputi penerimaan, pencatatan, pendistribusian dan pengiriman naskah dinas;
		e. melaksanakan pengelolaan kearsipan naskah dinas dan dokumentasi kedinasan serta pengelolaan perpustakaan Kecamatan;
		g. melaksanakan dan mengoordinasikan administrasi pengumpulan, pengolahan, penyimpanan dan pemeliharaan data serta dokumentasi kepegawaian;
3	Unit Kerja Lainnya	a. Melaksanakan tugas berdasarkan arahan pimpinan

## 2.2 Landasan Teori

### 2.2.1 Sistem

Sistem adalah hubungan antara unit yang satu dengan unit lainnya yang saling berhubungan satu dengan yang lainnya dan tidak dapat dipisahkan serta menuju suatu kesatuan dalam rangka mencapai tujuan yang telah ditetapkan sebelumnya. Misalnya, apabila satu unit dalam suatu perusahaan mengalami gangguan, unit yang lainnya pun akan terganggu dalam mencapai tujuan yang telah ditetapkan [3]. Sedangkan menurut Ibu Utami Dewi, Sistem merupakan kumpulan dari elemen elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. [4]

### 2.2.2 Informasi

Informasi adalah data yang diolah menjadi bentuk lebih berguna dan lebih berarti bagi yang menerimanya. Informasi juga disebut data yang diproses atau data yang memiliki arti. Informasi merupakan data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakan. Para pembuat

keputusan memahami bahwa informasi menjadi faktor kritis dalam menentukan kesuksesan atau kegagalan dalam suatu bidang usaha. Sistem apapun tanpa ada informasi tidak akan berguna, karena sistem tersebut akan mengalami kemacetan dan akhirnya berhenti. Informasi dapat berupa data mentah, data tersusun, kapasitas sebuah saluran informasi, dan sebagainya [5]

### **2.2.3 Sistem Informasi**

Sistem informasi adalah sekumpulan hardware, software, brainware, prosedur, dan/atau aturan yang diorganisasikan secara integral untuk mengolah data menjadi informasi yang bermanfaat guna memecahkan masalah dan pengambilan keputusan. Sistem informasi adalah satu kesatuan data olahan yang terintegrasi dan saling melengkapi yang menghasilkan data olahan, baik dalam bentuk gambar, suara maupun tulisan [4].

### **2.2.4 Surat**

Surat adalah sarana komunikasi untuk menyampaikan informasi tertulis oleh suatu pihak kepada pihak lain dengan tujuan memberitahukan maksud pesan dari si pengirim. Informasi yang diberikan di dalam surat berupa pengantar, pemberitahuan, tugas, permintaan, perjanjian, pesanan, perintah, laporan dan putusan. Selain itu, surat juga dapat berisi peringatan, teguran, dan penghargaan [1]

### **2.2.5 PHP**

PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan *web* dan dapat ditanamkan pada sebuah skripsi HTML. Bahasa PHP dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti *C*, *Java*, dan *Perl* serta mudah untuk dipelajari.

PHP merupakan bahasa scripting *server-side*, dimana pemrosesan datanya dilakukan pada sisi *server*. Sederhananya, serverlah yang akan menerjemahkan skrip program, baru kemudian hasilnya akan dikirim kepada *client* yang melakukan permintaan.

Adapun pengertian lain PHP adalah akronim dari *Hypertext Preprocessor*, yaitu suatu bahasa pemrograman berbasis kode – kode (*script*) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke web *browser* menjadi kode HTML”.

Menurut *Kustiyaningsih* (2011:114), “PHP (atau resminya PHP: *Hypertext Preprocessor*) adalah skrip bersifat *server – side* yang ditambahkan ke dalam HTML”.

Pada prinsipnya *server* akan bekerja apabila ada permintaan dari *client*. Dalam hal ini *client* menggunakan kode-kode PHP untuk mengirimkan permintaan ke *server*.

Sistem kerja dari PHP diawali dengan permintaan yang berasal dari halaman *website* oleh *browser*. Berdasarkan URL atau alamat *website* dalam jaringan internet, *browser* akan menemukan sebuah alamat dari *webserver*, mengidentifikasi halaman yang dikehendaki, dan menyampaikan segala informasi yang dibutuhkan oleh *webserver*.

Selanjutnya *webserver* akan mencarikan berkas yang diminta dan menampilkan isinya di *browser*. *Browser* yang mendapatkan isinya segera menerjemahkan kode HTML dan menampilkannya. Lalu bagaimana apabila yang dipanggil oleh user adalah halaman yang mengandung *script* PHP? Pada prinsipnya sama dengan memanggil kode HTML, namun pada saat permintaan dikirim ke *web-server*, *web-server* akan memeriksa tipe *file* yang diminta *user*. Jika tipe *file* yang diminta adalah PHP, maka akan memeriksa isi *script* dari halaman PHP tersebut.

Apabila dalam *file* tersebut tidak mengandung *script* PHP, permintaan *user* akan langsung ditampilkan ke *browser*, namun jika dalam *file* tersebut mengandung *script* PHP, maka proses akan dilanjutkan ke modul PHP sebagai mesin yang menerjemahkan *script-script* PHP dan mengolah *script* tersebut, sehingga dapat dikonversikan ke kode-kode HTML lalu ditampilkan ke *browser user* [5].



### 2.2.6 HTML

HTML (*Hyper Text Markup Language*) adalah sebuah bahasa *formatting* yang digunakan untuk membuat sebuah halaman *website*. Di dalam dunia pemrograman berbasis *website* (*Web Programming*), HTML menjadi pondasi dasar pada halaman *website*. sebuah file HTML di simpan dengan ekstensi `.html` (`dot html`) dan dapat di eksekusi atau diakses menggunakan web *browser* (Google Chrome, Mozilla Firefox, Opera, Safari dan lain-lain). seperti yang sudah di jelaskan, HTML adalah dasar dari sebuah *website* untuk membuat sebuah *website* tidak cukup hanya menggunakan HTML, kita memerlukan bantuan CSS, JavaScript dan PHP untuk membuat sebuah *website* yang dinamis jika halaman *website* dibuat hanya menggunakan HTML saja maka halaman *website* tersebut disebut halaman statis karena tidak memiliki aksi atau fungsi-fungsi yang dapat mengelola *website* tentu *developer* akan sangat disibukkan dengan harus mengubah lagi file HTML setiap ingin mengupdate artikel [6].

### 2.2.7 LARAVEL

Laravel adalah framework berbasis bahasa pemrograman PHP yang bisa digunakan untuk membantu proses pengembangan sebuah *website* agar lebih maksimal. Dengan menggunakan Laravel, *website* yang dihasilkan akan lebih dinamis.

Kehadiran framework Laravel menjadikan bahasa pemrograman PHP menjadi lebih powerful. Perlu kita ketahui bahwa kehadiran framework Laravel selalu menghadirkan fitur-fitur terbaru dibandingkan framework lainnya.

Framework Laravel menggunakan struktur MVC (Model View Controller). MVC merupakan model aplikasi yang memisahkan antara data dan tampilan berdasarkan komponen aplikasi. Dengan adanya model MVC, pengguna Laravel menjadi lebih mudah dalam mempelajari Laravel. Serta menjadikan proses pembuatan aplikasi berbasis *website* menjadi lebih cepat [7].

### 2.2.8 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*database management system*) atau DBMS yang *multithread, multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL[8].

### 2.2.9 Database

Basis data (*database*) adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut (<http://id.wikipedia.org/wiki/Database>). *Database* digunakan untuk menyimpan informasi atau data yang terintegrasi dengan baik di dalam komputer[8].

### 2.2.10 Unified Modelling Language (UML)

Menurut Rosa dan Shalahuddin (2014:133), berpendapat bahwa UML (Unified Modeling Language) adalah “Salah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisa & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek”. Sedangkan Mulyani (2016:48) mengatakan UML (Unified Modeling Language) adalah “Sebuah teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada sistem”.

Dari beberapa penjelasan teori tersebut dapat disimpulkan bahwa UML (Unified Modeling Language) adalah bahasa yang sering digunakan untuk membangun sebuah sistem perangkat lunak dengan melakukan penganalisaan desain dan spesifikasi dalam pemrograman berorientasi objek.


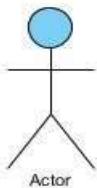

UML (Unified Modeling Language) memiliki diagram-diagram yang digunakan dalam pembuatan aplikasi berorientasi objek, diantaranya (Rosa dan Shalahuddin, 2014:155):

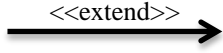

#### 1. Use Case Diagram

*Use Case Diagram* merupakan pemodelan untuk melakukan (behavior) sistem informai 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 itu.

Berikut adalah simbol-simbol yang ada pada diagram *use case* (Rosa dan Shalahuddin, 2014:156):

symbol 2.2-1 Simbol-simbol Diagram *Use Case*

Simbol	Deskripsi
<p><i>Use case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal <i>frase</i> nama <i>Use Case</i></p>
<p>Aktor / actor</p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informaasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang: biasanya dinyatakan menggunakan kata benda diawal <i>frase</i> nama aktor</p>
<p>Asosiasi / assosiation</p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor</p>

<p>Ekstensi / <i>extend</i></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dinamakan <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.</p>
<p>Generalisasi / <i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>


Sumber : Rosa dan Shalahuddin (2014:156)





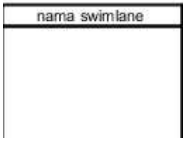

## 2. Activity Diagram

*Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Perlu diperhatikan bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Berikut adalah simbol-simbol yang ada pada diagram aktivitas (Rosa dan Shalahuddin, 2014:162):

symbol 2.2-2 Simbol-simbol *Activity Diagram*

Simbol	Deskripsi
<p>Status awal</p> 	<p>Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal</p>
<p>Aktivitas</p>	<p>Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja</p>

	
<p>Percabangan / <i>decision</i></p> 	<p>Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu</p>
<p>Penggabungan / <i>join</i></p> 	<p>Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu</p>
<p>Status akhir</p> 	<p>Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir</p>
<p><i>Swimlane</i></p>  <p>Atau</p> 	<p>Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi</p>

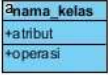






Sumber : Rosa dan Shalahuddin (2014:162)

### 3. *Class Diagram*

*Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

Berikut adalah simbol-simbol yang ada pada diagram Kelas (Rosa dan Shalahuddin, 2013:146) :

symbol 2.2-3 Simbol-simbol *Class Diagram*



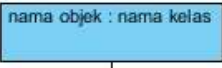
Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka / <i>Interface</i>  Nama_interface	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi / <i>Association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Asosiasi berarah / <i>Directed Association</i> 	Relasi antarkelas dengan makna kelas satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus)
Kebergantungan / <i>Dependency</i> 	Relasi antar kelas dengan makna kebergantungan antarkelas
Agregasi / <i>Aggregation</i> 	Relas antarkelas dengan makna semuabagian ( <i>whole-part</i> )

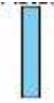



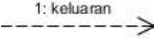
#### 4. *Sequence Diagram*

Diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Untuk menggambarkan diagram sekuen maka harus diketahui objekobjek yang terlibat dalam sebuah use case beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

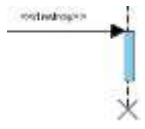
Berikut adalah simbol-simbol yang ada pada diagram sekuen (Rosa dan Shalahuddin,2014:165):

symbol 2.2-4 Simbol-simbol *Sequence Diagram*

Simbol	Deskripsi
<p>Aktor</p> <p>Atau</p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi itu sendiri, jadi walaupun simbol dari aktor adalah orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal <i>frase</i> nama aktor</p>
<p>Garis hidup / <i>Lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p> 	<p>Menytakan objek yang berinteraksi pesan</p>

<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.</p>
<p>Pesan tipe <i>create</i></p> 	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p>Pesan tipe <i>call</i></p> 	<p>Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri.</p> <p>Arah panah mengarah pada objek yang memiliki operasi / metode, karena ini memanggil operasi / metode maka operasi / metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi</p>
<p>Pesan tipe <i>send</i></p> 	<p>Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim</p>
<p>Pesan tipe <i>return</i></p> 	<p>Menyatakan suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>



<p>Pesan tipe <i>destroy</i></p> 	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada <i>destroy</i></p>
--	---

### 2.2.11 State Of Art

Penyusunan skripsi ini mengambil beberapa referensi dari penelitian sebelumnya termasuk jurnal-jurnal yang berhubungan dengan penelitian ini. Penyusunan yang digunakan dalam penyusunan skripsi ini dapat dilihat pada Tabel dibawah ini.

Table 2.2-1 *State Of Art*

No	Penulis	Judul	Perbedaan	Persamaan
1	Sanjaya, Kadek Oki, Subawa, I Gede Bendesa and Asmarajaya, I Kadek Andy. 2020	Perancangan Sistem Informasi Surat Menyurat Terintegrasi (SUMATRI) Berbasis Website dan Android [9]	Sistem yang dibangun hanya sampai dengan Rancangan Antarmuka	Tema yang diambil adalah Pengelolaan surat Menyurat
2	Sasongko, Jati and Diartono, Dwi Agus. 2009	Rancang Bangun Sistem Informasi Manajemen Surat [2]	Sistem yang dibuat hanya sebagai arsip surat aja	Tema yang diambil adalah Pengelolaan surat Menyurat
3	Mentari, Karisma Ayu, Andria and Muntahana, Hani Atun. 2021	RANCANG BANGUN SISTEM INFORMASI PENGELOLAAN	Sistem informasi yang dibangun hanya sebagai pengarsip surat	Tema yang diambil adalah Pengelolaan surat Menyurat

		SURAT MASUK DAN SURAT KELUAR BERBASIS WEB STUDI KASUS SDN KRAJAN 02 [10]	masuk dan surat keluar, dan sebagai sarana informasi saja	
4	Sitohang, Hengki Tamando	SISTEM INFORMASI PENGAGENDAAN SURAT BERBASIS WEB PADA PENGADILAN TINGGI MEDAN [11]	Sistem informasi yang dibangun hanya sebagai pengarsip surat masuk dan surat keluar, dan sebagai sarana informasi saja	Tema yang diambil adalah Pengelolaan surat Menyurat
5	Prawono, Joko Agus and Pamungkas, Anton Respati	SISTEM INFORMASI PENGELOLAAN SURAT MASUK DAN SURAT KELUAR DI STMIK AUB SURAKARTA [12]	Sistem informasi yang dibangun hanya sebagai pengarsip surat masuk dan surat keluar, dan sebagai sarana informasi saja	Tema yang diambil adalah Pengelolaan surat Menyurat