BAB II
TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1. Tinjauan Pustaka

Dalam penelitian ini akan digunakan lima tinjauan studi yang nantinya dapat mendukung penelitian, berikut ini merupakan tinjauan studi yang diambil yaitu :

Tabel 2. 1 Tinjauan Pustaka Penelitian Terdahulu

No	Karakteristik	Keterangan
1	Rohim (2015)	
	Judul	Semarang Charity Map, Penyajian Peta Donasi Sosial Kota Semarang Berbasis Blogger Javascript
	Penulis	Wahyu Nur Rohim, Moehammad Awaluddin, Andri Suprayogi
	Penerbit	Jurnal Geodesi Undip Volume ,Nomor , (ISSN : 2337-845X)
	Tanggal/Tahun	April 2015
	Permasalahan	Keterbatasan informasi mengenai kegiatan sosial, panti sosial, komunitas sosial dan data penyandang masalah kesejahteraan sosial yang selanjutnya kami sebut donasi sosial di kota semarang sedikit banyak menghambat mereka dalam melakukan manajemen donasi.
	Tujuan Penelitian	Peta utama mampu menyajikan data kegiatan sosial, panti sosial dan komunitas sosialdengan fitur petunjuk jalan, estimasi jarak tempuh perjalanan dan estimasi waktu.
	Subjek penelitian	Pemetaan donasi sosial di Kota Semarang
	Metode Penelitian	Kualitatif dan deskriptif
	Hasil Penelitian	Aplikasi SIG berbasis web
2	Indraswari (2018)	
	Judul	Analisis Aplikasi Arcgis 10.3 Untuk Pembuatan Daerah Aliran Sungai Dan Penggunaan Lahan Di Das Samajid Kabupaten Sampang, Madura

	Penulis	Devita Indraswari, Nida Hanifah, Mutia Januar Ramadani, Yuli Priyana
	Penerbit	Prosiding Seminar Nasional Geografi UMS IX (ISBN: 978-602-361-137-9)
	Tanggal/Tahun	Juni 2018
	Permasalahan	Pembuatan peta daerah aliran sungai Samajid disertai penggunaan lahannya
	Tujuan Penelitian	Tujuan dari penelitian ini adalah untuk mengetahui perkembangan penggunaan lahan yang terjadi di Daerah Aliran Sungai Samajid dari hulu, tengah, hingga hilir
	Subjek penelitian	Pemetaan Daerah Aliran Sungai Di DAS Samajid, Kabupaten Sampang, Madura
	Metode Penelitian	Deskriptif dan kualitatif
	Hasil Penelitian	Aplikasi Penginderaan Jauh dan SIG untuk Restorasi Sungai
3	Lestari & Arsyad (2018)	
	Judul	Studi Penggunaan Lahan Berbasis Data Citra Satelit Dengan Metode Sistem Informasi Geografis (SIG)
	Penulis	Senifa Citra Lestari, Muhammad Arsyad
	Penerbit	Jurnal Sains dan Pendidikan Fisika (JSPF) ISSN Cetak: 1858-330X dan ISSN Online: 2548-6373
	Tanggal/Tahun	April 2018
	Permasalahan	Pengidentifikasian penggunaan lahan dan menganalisis sebaran luas penggunaan lahan
	Tujuan Penelitian	Pendigitasian kelompok kelas penggunaan lahan yang sama dan menampilkan tabel luasan penggunaan lahan
	Subjek penelitian	Pengolahan data aktivitas penggunaan lahan yang dilakukan oleh masyarakat di Kecamatan Sopai
	Metode Penelitian	Penelitian terapan
	Hasil Penelitian	Sistem Informasi Geografis (SIG) Penggunaan Lahan Berbasis Data Citra Satelit
4	Siswanto & Jazman (2016)	
	Judul	Rancang Bangun WebGIS Pemetaan Lokasi Panti Sosial Menggunakan Pmapper
	Penulis	Joko Siswanto, Muhammad Jazman
	Penerbit	Jurnal Rekayasa Dan Manajemen Sistem Informasi, Vol.2, No 2 (e-ISSN 2502-8995 ISSN 2460-8181)
	Tanggal/Tahun	Agustus 2016

	Permasalahan	Belum ada sistem pemetaan panti asuhan yang memberikan informasi lengkap tentang panti asuhan termasuk peta lokasi panti asuhan
	Tujuan Penelitian	Membangun sistem informasi mengenai peta persebaran panti asuhan di Kota Pekanbaru berbasis web.
	Subjek penelitian	Data persebaran panti asuhan Kota Pekanbaru
	Metode Penelitian	System Development Life Cycle Model Waterfall
	Hasil Penelitian	Webgis Pemetaan Lokasi Panti Sosial
5	Suryanto (2017)	
	Judul	Pemetaan Potensi Lahan Pertanian Di Kabupaten Tuban Berbasis Sistem Informasi Geografis
	Penulis	Andik Adi Suryanto, Amaludin Arifia, Asfan Muqtadir, Dwi Kurnia Basuki
	Penerbit	SAINTEKBU: Jurnal Sains dan Teknologi Volume 9 No.2 (ISSN Print : 1979-7141, ISSN Online : 2541-1942)
	Tanggal/Tahun	Mei 2017
	Permasalahan	Kurangnya informasi yang berhubungan dengan produktifitas dan kelayakan penggunaan lahan
	Tujuan Penelitian	Memberikan informasi pemetaan lahan pertanian di Kabupaten Tuban yang digunakan sebagai pengambilan informasi untuk analisa kebijakan bidang pertanian.
	Subjek penelitian	Pengolahan data lahan pertanian di Kabupaten Tuban
	Metode Penelitian	*SDLC (System Development Life Cycle)*
	Hasil Penelitian	Sistem Informasi Geografis Pemetaan Potensi Lahan Pertanian Di Kabupaten Tuban

2.2. Sistem Informasi Geografi

Sistem informasi geografis merupakan sistem yang dibuat untuk mengintegrasikan data spasial (peta, *vector*, dan citra digital), atribut (tabel sistem basisdata), dan *properties* (Prahasta, 2007)

2.2.1. Konsep Dasar Sistem Informasi Geografis

SIG (Sistem Informasi Geografis) atau dikenal pula dengan GIS *(Geographical Information System)* merupakan suatu istilah dalam bidang pemetaan yang memiliki ruang lingkup mengenai bagaimana suatu sistem dapatmenghubungkan objek geografis dengan informasinya (Citra, 2008). Definisi SIG berdasarkan pemahaman tiap-tiap kata yaitu:

1. **Geografis**

   SIG dibangun berdasarkan geografi atau spasial, didalam terdapat objek-objek yang berupa fisik, budaya, atau ekonomi alamiah. Keterangan spasial dari objek-objek ini ditampilkan pada suatu peta untuk memberikan gambaran yang representatif dari keadaan sebenarannya di muka bumi.

2. **Informasi**

   Pada representasi permukaan geigrafis SIG, terdapat beberapa objek-objek dimana tiap-tiap objek memiliki informasi masing-masing yang unik. Hubungan langsung antara objek dan informasi yang bersifat interaktif membuat peta menjadi *intelligent.*

3. **Sistem**

   Sistem merupakan kumpulan elemen-elemen yang saling berintegrasi dan berketergantungan dalam lingkungan yang dinamis untuk mencapai tujuan tertentu. Pada SIG sistem merupakan kumpulan dari informasi, data geospital, dan juga sistem komputer atau perangkat elektronik lainnya.

Dari pengertian tiap kata diatas, maka SIG dapat didefinisikan sebagai sistem informasi berbasis komputer yang menggabungkan unsur peta (geografis)

dan informasinya tentang peta terebut (data atribut) yang dirancang untuk mendapatkan, mengolah, memaniplasi, anaisa, memperagakan dan menampilkan data spasial untuk menyelesaikan perencanaan, mengolah dan meneliti permasalahan.

2.2.2. Komponen Sistem Informasi Geografis

Sistem Informasi Geografis merupakan sistem kompleks yang umunya terintegrasi dengan sistem komputer lainnya di tingkat fungsional dan jaringan. Jika diuraikan SIG terdiri dari komponen dengan berbagai kerakteristiknya (Prahasta, 2014) :

1. Perangkat Keras

Sistem Informasi Geografis tersedia diberbagai *platform* perangkat keras, mulai dari kelas PC *desktop*, *workstation*, hinga *multi-user host*.

2. Perangkat Lunak

Sistem Informasi Geografis merupakan sistem perangkat lunak dimana basis datanya memegang peranan kunci. Pada Sistem Informasi Geografis lama, subsistem diimplementasikan oleh modul-modul perangkat lunak hingga tidak mengherankan jika ada perangkat Sistem Informasi Geografis yang terdiri dari ratusan modul program (*.exe) yang dapat dieksekusi sendiri.

3. Data dan Informasi Geografis

Sistem Informasi Geografis dapat mengumpulkan dan menyimpan data/inforasi yang diperlukan baik tidak langsung (dengan meng-*import*-nya) maupun langsung dengan mendijitasi data spasialnya (*on-screean/head-ups* pada layar monitor atau cara manual dengan *digitizer*) dari peta analog dan memasukan data atributnya dari tabel/laporan dengan menggunakan *keyboard*.

4. Manajemen

Proyeksi sistem Informasi Geografis akan berhasil jika dikelola dengan baik dan dkerjakan orang yang memiliki keahlian yang tepat pada semua tingkatan.

2.3. Jenis Peta

Peta adalah gambaran permukaan bumi yang digambar pada bidang datar dan diperkecil dengan skala tertentu dilengkapi simbol sebagai penjelas (Menguasai ArcGIS10 untuk Pemula, 2015:9). Peta membantu memberikan informasi mengenai segala hal yang berada di permukaan bumi, seperti letak suatu wilayah, jarak antar kota, lkasi pegunungan, sungai, danau, lahan persawahan, jallan raya, bandara, dan sebagainya. Berdasarkan data dan informasi ada dua macam jenis peta yang berlaku di indonesia (Menguasai ArcGIS10 untuk Pemula, 2015:9), yaitu sebagai berikut:

1. Peta Rupa Bumi (RBI)

Peta rupa bumi indonesia atau peta dasar adalah peta yang menunjukan objek-objek di permukaan bumi pada posisi yang sebenarnya. Peta ini digunakan sebagai dasar kegiatan yang berhubungan dengan data inormasi yang menggunakan refrensi geografis.

2. Peta Tematik

Peta tematik adalah peta yang menyajikan informasi tentang suatu tea atau maksud tertentu dalamkaitannya dengan unsur topografi yang spesifik sesuai tema peta. Detail topografi pada peta tematik diambil dari peta dasar dan tema peta dapat diketahui dari judul peta.

2.4. Data Spasial

Data spasial adalah data mengenai objek atau unsur geografis yang dapat diindenfikasikan dan mempunyai acuan lokasi berdasarkan koordinat tertentu. Data spasial meliputi data grafis dan data atribut. Salah satu syarat terwujudnya SIG adalah data spasial (Mukhlis, 2013). Data ini memiliki sistem koordinat tertentu sebagai dasar referensinya dan mempunyai dua bagian penting yang berbeda dari data lain yaitu :

1. Informasi lokasi (spasial), berkaitan dengan suatu koordinat baik koordinat goegrafis (lintang dan bujur) dan koordinat XY, termasuk diantaranya proyeksi.
2. Informasi deskriptif (atribut) atau informasi nonspasial, suatu lokasi yang memiliki beberapa keterangan yang berkaitan dengannya, Contoh jenis vegetasi, populasi, luasan, dan kode pos.

2.5. Model Data Spasial Sistem Infornasi Geografis

Data spasial adalah data dalam bentuk grafis yang menunjukkan ruang lokasi atau tempat tempat di permukaan bumi. Data spasial dilambangkan dengan titik, garis, serta poligon. Terdapat dua model dalam data spasial, yaitu model data raster dan model data vektor. Keduanya memiliki karakteristik yang berbeda, selain itu dalam pemanfaatannya tergantung dari masukan data dan hasil akhir yang akan dihasilkan. Model data tersebut merupakan representasi dari obyek-obyek geografi yang terekam sehingga dapat dikenali dan diproses oleh komputer.

2.5.1. Model Data Raster

Model data raster bertugas untuk menampilkan dan menyimpan *content* data spasial dengan menggunakan struktur matriks/susnan piksel yang membentuk *grid*.

Setiap piksel/sel memiliki atribut tunggal. Akurasi horizontal model data ini bergantung pada resolusi spasial/ukuran pikselnya (Prahasta, 2014).

2.5.2. Model Data Vektor

Meodel data vektor menampilkan, menempatkan, dan menyimpan data spasial dengan titik, garis/kurva, poligin beserta atributnya.bentuk sajian ini didefinisikan oleh sistem koordinat kartesian 2D. Pada model vektor, garis/kurva merupakan kumpulan titik yang terhubung. Sedangkan area/poligon disimpan sebagai *list* titik dengan titik awal dan titik ahi yang merupakan koordinat yang sama (Prahasta, 2014).

2.6. Digitasi Peta

Digitasi secara umum dapat didefinisikan sebagai proses konfersi data analog kedalam format digital seperti jalan, rumah, sawah dan lain-lain yang sebelumnya dalam format raster pada sebuah citra satelit resolusi tertinggi dapat diubah kedalam format digital salah satunya melalui digitasi onscreen atau digitasi di layar monitor.

2.7. *Framework Codeigniter*

Framework Codeigniter merupakan kumpulan intruksi-intruksi yang di kumpulkan dalam *class* dan *function* dengan fungsi masing-masing untuk memudahkan *developer* dalam pemanggilannya tanpa harus menuliskan *syntax* program yang sama secara berulang-ulang. Hal ini memiliki kegunaan untuk menghemat waktu dan mencegah perulangan penulisan *syntax* agar tercipta nya *source code* yang bersih dan terstruktur. *MVC* adalah teknik atau konsep yang memisahkan komponen utama menjadi tiga komponen yaitu *model*, *view* dan *controller*.

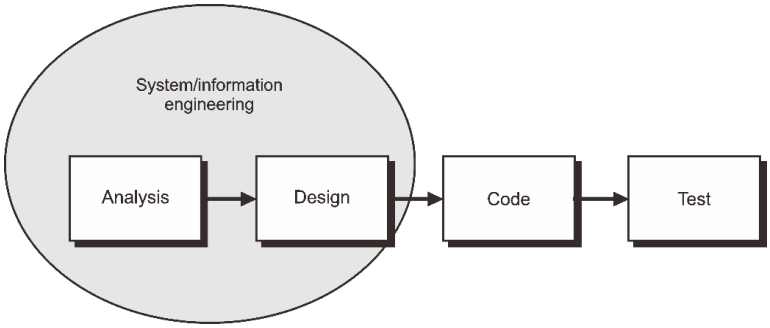
2.8. *ArcGIS*

ArcGIS adalah salah satu software yang dikembangkan oleh ESRI (*Environment Science & Research Institue*) yang merupakan kompilasi fungsi-fungsi dari berbagai macam *software* GIS yang berbeda seperti GIS *desktop*, *server*, dan GIS berbasis web. Software ini mulai dirilis oleh ESRI Pada tahun 2000. Produk Utama Dari *ArcGIS* adalah *ArcGIS* desktop, dimana arcgis desktop merupakan *software* GIS professional yang komprehensif dan dikelompokkan atas tiga komponen yaitu : *ArcView* (komponen yang focus ke penggunaan data yang komprehensif, pemetaan dan analisis), *ArcEditor* (lebih fokus ke arah editing data spasial) dan *ArcInfo* (lebih lengkap dalam menyajikan fungsi-fungsi GIS termasuk untuk keperluan analisi *geoprosesing*).

2.9. Metode Pengembangan Sistem Model *Waterfall*

Menurut (Pressman, 2010) model *waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software*. *Waterfall* adalah suatu metodologi pengembangan perangkat lunak yang mengusulkan pendekatan kepada perangkat lunak sistematik dan sekuensial yang mulai pada tingkat kemajuan sistem pada seluruh analisis, *design*, kode, pengujian dan pemeliharaan. Adapun tahapan-tahapan dalam metode *waterfall* adalah sebagai berikut:

Ilustrasi model waterfall diatas dapat dilihat pada gambar 1 berikut ini :



Gambar 2. 1 Ilustrasi *model waterfall* (Pressman, 2010)

1. **Analisis Kebutuhan Perangkat Lunak (*Analysis*)**

   Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh pengguna. Mendefinisikan format perangkat lunak, identifikasi spesifikasi sistem yang dibutuhkan, modul *input* data, modul *output* data.

2. **Desain (*Design*)**

   Desain perangkat lunak adalah proses multilangkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasi menjadi program pada tahap selanjutnya.

3. **Pembuatan kode program (*Coding*)**

   Desain harus ditranslasikan ke dalam program perangkat lunak.Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. **Pengujian (*Testing*)**

Pengujian fokus pada perangkat lunak dari segi lojik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

2.10. Bagan Alir Dokumen (*Flowchart*)

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urut-urutan prosedur dari suatu program. *Flowchart* menolong analis dan programmer untuk memecakan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoprasian. Bagan alir dokumen di gambarkan menggunakan dengan simbol-simbol *standart* tertentu. Bagan, nama dan fungsinya seperti yang disajikan pada tabel berikut :

Tabel 2. 2 Simbol Bagan Alir Dokumen (*Flowchart*)

Bagan	Nama	Fungsi
	terminator	Awal atau akhir program
	flow	Arah aliran program
	preparation	inisialisasi/pemberian nilai awal
	proces	Proses/pengolahan data
	input/output data	input/output data

	sub program	sub program
	decision	Seleksi atau kondisi
	on page connector	Penghubung bagian-bagian flowchart pada halaman yang sama
	off page connector	Penghubung bagian-bagian flowchart pada halaman yang berbeda
	comment	Tempat komentar tentang suatu proses

2.11. *Unified Modelling Language* (UML)

Unified Modelling Language (UML) adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk visualisasi, merancang model sebuah sistem UML dapat dibuat untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan class dan operation dalam konsep dasarnya, maka lebih cocok untuk penulisan piranti lunak dalam bahasa berorientasi objek seperti C++, Java, atau WEB.

(Rosa, 2011) mengatakan *Unified Modeling Language* (UML) adalah sebuah bahasa untuk menentukan, visualisasi, kontruksi, dan mendokumentasikan *artifact* (bagian dari informasi yang digunakan atau dihasilkan dalam suatu proses pembuatan perangkat lunak). UML merupakan notasi yang lengkap untuk membuat visualisasi model suatu sistem. Sistem berisi informasi dan fungsi, namun secara normal digunakan untuk pemodelan sistem komputer. UML tidak hanya digunakan

dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan".

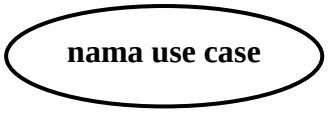
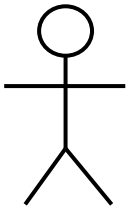
2.12. *Diagram* dalam UML

Diagram berbentuk grafik yang menunjukkan simbol elemen model yang disusun untuk mengilustrasikan bagian atau aspek tertentu dari sistem. Sebuah *diagram* merupakan bagian dari suatu *view* tertentu dan ketika digambarkan biasanya dialokasikan untuk *view* tertentu. Adapun jenis *diagram* antara lain:

1. *Use Case Diagram*

(Rosa, 2011) "*Use case* atau *diagram use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuar. *Use case* merupakan konstruksi untuk mendeskripsikan bagaimana sistem akan terlihat di mata *user*, sedangkan *use case diagram* memfasilitasi komunikasi diantara analis dan pengguna serta antara analis dan *client*".

Tabel 2. 3 Simbol *Use Case Diagram*

Simbol	Deskripsi
Use Case 	Menjelaskan tentang fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama *use case* pada sistem
Aktor/*actor* 	Menjelaskan tentang orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya

	dinyatakan menggunakan kata benda di awal frase nama aktor pada sistem.
Asosiasi/*asociation*	Menjelaskan tentang komunikasi antara aktor dan *use case* yang berpatisipasi pada *use case* atau *use case* memiliki interaksi dengan aktor pada sistem.
Ekstensi/*exten* <<*extend*>>	Menjelaskan tentang relasi *use case* tambahan ke sebuah *use case* di mana *use case*yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tmbahan itu; mirip dengan prinsip *inheritance* pada pemrograman berorientasi objek; biasanya *use case* tambahan memiliki nama depan yang sama dengan *use case* yang ditambahkan pada sistem.
Generalisasi/*generalization*	Menjelaskan tentang hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah *use case* di mana fungsi yang satu adalah fungsi yang lebih umum dari lainnya pada sistem.
Menggunakan/*Include/uses* <<*extend*>>	Menjelaskan tentang relasi *use case* tambahan ke sebuah *use case* di mana *use case* yang ditambahkan memerlukan *use case* ini untuk menjalankan fungsinya atau sebagai syarat dijalankan *use case* ini pada sistem.

2. Class Diagram

(Rosa, 2011) "*Class* adalah deskripsi kelompok objek-objek dengan *property*. *Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Sebuah sistem biasanya

mempunyai beberapa *class diagram*. *Class diagram* sangat membantu dalam visualisasi struktur kelas dari suatu sistem". Simbol *class diagram*.

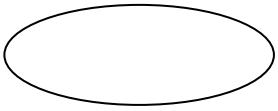
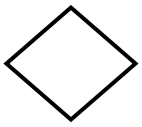
Tabel 2. 4 Simbol *Class Diagram*

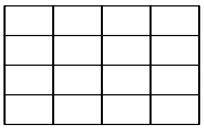
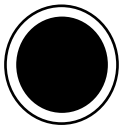
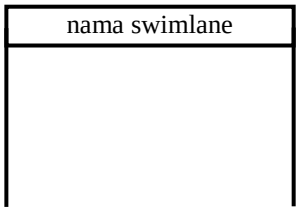
Simbol	Deskripsi
Kelas Site Config +sqlDNS:string +Adminemail:String	Menjelaskan tentang kelas pada struktur sistem yang akan dibuat.
Asosiasi/*asociation*	Menjelaskan tentang relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity* pada sistem.
Asosiasi berarah/ *directed association*	Menjelaskan tentang relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan *multiplicity* pada sistem.
Generalisasi	Menjelaskan tentang relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus) pada sistem.
Kebergantungan/ *dependecy*	Menjelaskan tentang relasi antar kelas dengan makna kebergantungan antar kelas pada sistem.
Agregasi/*agregation*	Menjelaskan tentang relasi antar kelas dengan makna semua bagian (*whole-part*) pada sistem.

3. *Activity Diagram*

(Rosa, 2011) "*Activity diagram* ini menggambarkan rangkaian aliran dari aktivitas dari sebuah sistem atau proses bisnis. *Activity diagram* adalah teknik untuk mendeskripsikan logika *procedural*, proses bisnis dan aliran kerja dalam banyak kasus. *Activity diagram* mempunyai peran penting seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *activity diagram* bisa mendukung paralel sedangkan *flowchart* tidak bisa".

Tabel 2. 5 Simbol *Activity Diagram*

Simbol	Deskripsi
Status awal	Menjelaskan tentang status awal aktivitas sistem,sebuah diagram aktivitas memiliki sebuah status awal pada sistem.
Aktivitas 	Menjelaskan tentang aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja pada sistem.
Percabangan/*decision* 	Menjelaskan tentang asosiasi percabangan di mana jika ada pilihan aktivitas lebih dari satu pada sistem.
Penggabungan/*join*	Menjelaskan tentang asosiasi penggabungan di mana lebih dari satu aktivitas digabungkan menjadi satu pada sistem.

Tabel 	Menjelaskan tentang suatu file komputer dari mana data bisa dibaca atau direkam selama kejadian bisnis pada sistem.
Dokumen	Menjelaskan tentang menunjukan dokumen sumber atau laporan pada sistem.
Status akhir 	Menjelaskan tentang status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir pada sistem.
Swimlane 	Menjelaskan tentang memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi pada sistem.

2.13. *Black Box Testing*

Black box testing merupakan teknik pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak. Pengujian bekerja dengan mengabaikan struktur pada *control* sehingga berfokus pada informasi utama. Pengujian menggunakan *black box testing* memungkinkan pengembang untuk membuat daftar kondisi *input* yang akan melatih seluruh batasa-batasan fungsional pada suatu sistem (Ma'ruf, et al., 2020).

1. Keuntungan dalam menggunakan metode *black box testing* adalah pengujian tidak perlu memiliki pengetahuan tentang bahasa pemrograman tertentu. Pengujian dilakukan dari sudut pandang pengguna sehingga programmer dan tester keduanya saling bergantung satu sama lain
2. Kekurangan dari metode *black box testing* adalah pengujian kasus sulit didesain tanpa spesifikasi yang jelas. Memungkinkan memiliki pengulangan pengujian yang sudah dilakukan oleh *developer*. Beberapa bagian *back end* tidak diuji sama sekali.

Black box testing berfokus pada spesifikasi fungsional dari perangkat lunak. *Tester* dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengetesan pada spesifikasi fungsional program. *Black box testing* bukanlah solusi alternatif dari *white box testing* tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh *white box testing* (Mustaqbal, et al., 2015).

Black box testing cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka *(interface errors).*
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi *(performance errors).*
5. Kesalahan inisialisasi dan terminasi.

Saat ini terdapat banyak metoda atau teknik untuk melaksanakan *black box testing*, antara lain:

1. *Equivalence Partitioning*
2. *Boundary Value Analysis/Limit Testing*
3. *Comparison Testing*

4. *Sample Testing*
5. *Robustness Testing*
6. *Behavior Testing*
7. *Requirement Testing*
8. *Performance Testing*
9. Uji Ketahanan (*Endurance Testing)*
10. Uji Sebab-Akibat *(Cause-Effect Relationship Testing)*

Pada pengujian aplikasi monitoring hanya berfokus pada metode/teknik *Boundary Value Analysis/Limit Testing* untuk melaksanakan pengujian *black box testing*.

2.14. *Boundary Value Analysis (BVA)*

Boundary value analysis merupakan salah satu teknik dalam metode pengujian black box yang fokus pada proses input dengan menguji nilai batas atas dan nilai batas bawah, prinsip kerja BVA yaitu, BVA bekerja pada proses input. Pengalaman menunjukkan bahwa teknik ini memiliki peluang lebih tinggi untuk mendeteksi kesalahan (Ardana, 2019).

Boundary value analysis adalah salah satu teknik black box testing yang melakukan pengujian pada batas atas dan batas bawah nilai yang diisikan pada aplikasi (Mustaqbal, et al., 2015). Beberapa prinsip yang mendasari pada boundary value analysis (BVA) yaitu :

1) Banyak kesalahan terjadi pada kesalahan masukan.
2) BVA mengijinkan untuk menyeleksi kasus uji yang menguji batasan nilai input.

3) BVA merupakan komplemen dari *equivalence partitioning*. Lebih pada memilih elemen-elemen di dalam kelas ekivalen pada bagian sisi batas dari kelas.