

**RANCANG BANGUN SISTEM INFORMASI DAERAH
RAWAN KRIMINALITAS
(STUDI KASUS: KOTA MAKASSAR)**

SKRIPSI



EKA FITRI RAMADANI

H071191043

**PROGRAM STUDI SISTEM INFORMASI
DEPARTEMEN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS HASANUDDIN
MAKASSAR**

2023

**RANCANG BANGUN SISTEM INFORMASI DAERAH
RAWAN KRIMINALITAS
(STUDI KASUS: KOTA MAKASSAR)**

SKRIPSI



**Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana
Komputer pada Program Studi Sistem Informasi Departemen Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin**

EKA FITRI RAMADANI

H071191043

**PROGRAM STUDI SISTEM INFORMASI
DEPARTEMEN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS HASANUDDIN
MAKASSAR**

2023

LEMBAR PERNYATAAN KEOTENTIKAN

Yang bertandatangan dibawah ini:

Nama : Eka Fitri Ramadani

Nim : H071191043

Program Studi : Sistem Informasi

Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

**RANCANG BANGUN SISTEM INFORMASI DAERAH RAWAN
KRIMINALITAS**

(STUDI KASUS: KOTA MAKASSAR)

Adalah karya tulisan saya sendiri dan bukan merupakan pengambil alih tulisan orang lain, dan belum pernah dipublikasikan dalam bentuk apapun.

Makassar, 10 Agustus 2023



Eka Fitri Ramadani

NIM. H071191043

**RANCANG BANGUN SISTEM INFORMASI DAERAH RAWAN
KRIMINALITAS
(STUDI KASUS: KOTA MAKASSAR)**

Disusun dan diajukan oleh

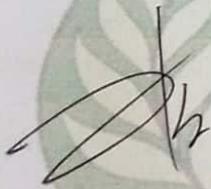
**EKA FITRI RAMADANI
H071191043**

UNIVERSITAS HASANUDDIN

Menyetujui.

Pembimbing Utama

Pembimbing Pertama



Dr. Hendra, S.Si., M.Kom
NIP. 197601022002121001

Dr. Muhammad Hasbi, M.Sc
NIP. 196307201989031003

Kepala Program Studi



Dr. Hendra, S.Si., M.Kom
NIP. 197601022002121001

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :
Nama : Eka Fitri Ramadani
NIM : H071191043
Program Studi : Sistem Informasi
Judul Skripsi : Rancang Bangun Sistem Informasi Daerah Rawan
Kriminalitas (Studi Kasus: Kota Makassar)

Telah berhasil dipertahankan dihadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Komputer pada Program Studi Sistem Informasi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin.

DEWAN PENGUJI

		Tanda Tangan
Ketua	: Dr. Hendra, S.Si., M.Kom	(.....)
Sekretaris	: Dr. Muhammad Hasbi, M.Sc	(.....)
Anggota	: A. Muh Amil Siddik, S.Si., M.Si	(.....)
Anggota	: Muhammad Sadno, S.Si., M.Si	(.....)

Ditetapkan di : Makassar
Tanggal : 10 Agustus 2023



KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Puji Syukur penulis panjatkan kepada Allah SWT, karena dengan rahmat, hidayah dan karunia-Nya, penulis dapat menyelesaikan penyusunan skripsi ini yang berjudul “Rancang Bangun Sistem Informasi Daerah Rawan Kriminalitas (Studi Kasus: Kota Makassar)”. Shalawat serta salam penulis curahkan kepada Rasulullah SAW yang telah menjadi teladan bagi umat manusia.

Skripsi ini dibuat dan diajukan untuk memenuhi syarat dalam menyelesaikan Pendidikan strata 1 (S1) Sarjana Komputer. Penulis menyadari bahwa penulisan skripsi ini tidak lepas dari bantuan dan dukungan berbagai pihak yang telah memberikan kontribusi besar dalam penyusunan skripsi ini. Oleh karena itu, penulis ingin mengucapkan terima kasih yang setulus-tulusnya kepada Ibu **Nurdaliah**, selaku ibu penulis yang selalu memberikan kasih sayang, doa, dukungan dan nasehatnya kepada penulis sekaligus sebagai penerang setiap langkah perjalanan penulis.

Dengan segala kerendahan hati penulis menyampaikan terima kasih yang setulus-tulusnya kepada:

1. Bapak **Prof. Dr. Ir. Jamaluddin Jompa, M.Sc.**, selaku rektor Universitas Hasanuddin Makassar.
2. Bapak **Dr. Eng Amiruddin, M.Si.**, selaku dekan Fakultas Matematika dan Ilmu Pengetahuan Alam beserta staf yang telah membantu dan mengarahkan penulis dalam berbagai hal, baik dalam urusan akademik maupun administrasi.
3. Bapak **Prof. Dr. Nurdin, S.Si., M.Si.**, selaku ketua Departemen Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam.
4. Bapak **Dr. Hendra, S.Si., M.Si.**, selaku Ketua Program Studi Sistem Informasi sekaligus Pembimbing Utama penulis yang senantiasa memberikan bimbingan dan arahan selama masa studi penulis hingga penyusunan skripsi ini.
5. Bapak **Dr. Muhammad Hasbi, M.Sc.**, selaku Pembimbing Pertama penulis yang senantiasa memberikan bimbingan dan arahan selama masa studi penulis hingga penyusunan skripsi ini.
6. Dosen Penguji, Bapak **A. Muh. Amil Siddik, S.Si., M.Si.** dan Bapak **Muhammad Sadno, S.Si., M.Si.**, yang telah meluangkan waktunya sejak

seminar proposal hingga sidang skripsi untuk memberikan saran atau masukan untuk penyusunan skripsi penulis.

7. Penasehat Akademik penulis Ibu **Rozalina Amran, S.T., M.Eng.**, yang telah membantu dan mengarahkan penulis dalam berbagai hal selama masa studi penulis.
8. **Bapak/Ibu Dosen Program Studi Sistem Informasi**, yang telah mendidik dan memberikan ilmunya kepada penulis selama masa perkuliahan serta kepada **Staf dan Pegawai** Departemen Matematika yang telah membantu dalam proses administrasi.
9. **Keluarga Tercinta**, yang selalu memberikan doa, semangat dan dukungan dalam setiap langkah penulis.
10. Sahabat **“Girls”** yaitu Alike, Anugrah, Caca, Rahmi dan Sakinah yang saling memberikan informasi, dukungan dan semangat selama masa perkuliahan hingga selesainya penulisan skripsi ini.
11. Teman-teman Program Studi **Sistem Informasi Angkatan 2019**, yang senantiasa memberikan dukungan selama masa perkuliahan hingga selesainya penulisan skripsi ini.
12. Teman-teman **“KKNT Smart Village Barru Posko 10”** yaitu Alya, Cecil, Ila, Mita, Yuni, Ulla, kak Rahmat, yang telah menjadi bagian dari perjalanan dan momen-momen berharga penulis.
13. **Responden Penelitian**, yang telah bersedia meluangkan waktu dan memberikan informasi berharga untuk penyusunan skripsi ini.

Akhir kata, semua dukungan dan masukan yang telah diberikan oleh berbagai pihak sangat berarti bagi penulis. Penulis berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan pihak yang telah membantu penyusunan skripsi ini. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu pengetahuan. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan penulis untuk menambah kesempurnaan penelitian skripsi ini.

Makassar, 10 Agustus 2023



Eka Fitri Ramadani

**PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR
UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Hasanuddin, saya yang bertanda tangan dibawah ini:

Nama : Eka Fitri Ramadani
NIM : H071191043
Program Studi : Sistem Informasi
Departemen : Matematika
Fakultas : Matematika dan Ilmu Pengetahuan Alam
Jenis Karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Hasanuddin **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty- Free Right*)** atas karya ilmiah saya yang berjudul:

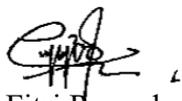
Rancang Bangun Sistem Informasi Daerah Rawan Kriminalitas (Studi Kasus: Kota Makassar)

beserta perangkat yang ada (jika diperlukan). Terkait dengan hal diatas, maka pihak universitas berhak menyimpan, mengalih-media/format-kan, mengelola dalam bentuk pangkalan data (database), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di Makassar Pada tanggal 10 Agustus 2023

Yang menyatakan



(Eka Fitri Ramadani)

ABSTRAK

Tindak kriminalitas di berbagai kota sudah menjadi permasalahan sosial yang membuat warga menjadi resah, terutama di kota – kota besar seperti kota Makassar. Jumlah kejahatan yang dilaporkan menurut kesatuan resort wilayah di Kota Makassar pada tahun 2013 sebanyak 4.891 kasus dari 19.042 kasus, pada tahun 2014 sebanyak 4.985 kasus dari 18.769 kasus dan pada tahun 2015 sebanyak 5.546 kasus dari 19.826 kasus. Sehingga dapat disimpulkan di Kota Makassar terjadi peningkatan jumlah kejahatan dari tahun 2013 sampai dengan 2015 dengan rata rata 26,74 % dari jumlah kejahatan yang dilaporkan di Provinsi Sulawesi Selatan. Maka dari itu, akan dibangun sebuah sistem informasi daerah rawan kriminalitas di Kota Makassar yang dapat memudahkan pihak kepolisian untuk mengetahui tingkat kerawanan terjadinya kriminalitas di Kota Makassar sehingga bisa dijadikan dasar untuk menentukan strategi yang tepat dan arah penanggulangan kejahatan bagi wilayah yang rawan serta bagi masyarakat diharapkan untuk lebih meningkatkan kewaspadaan di daerah–daerah yang rawan. Sistem informasi geografis berbasis *web* ini dirancang dan dibangun dengan menggunakan metode *waterfall*, bahasa pemrograman PHP dengan *framework Laravel*, *Leaflet JavaScript Library* dan MySQL sebagai *database server*. Hasil pengujian menggunakan *blackbox testing* didapatkan bahwa seluruh fitur berhasil dijalankan dan berfungsi dengan baik serta pengujian dengan menggunakan *User Acceptance Testing* diperoleh persentase skor 89,75%.

Kata Kunci: Kriminalitas, Sistem Informasi Geografis, Metode *Waterfall*, *Blackbox Testing*, *User Acceptance Testing*

ABSTRACT

Criminal acts in various cities have become a social problem that makes residents restless, especially in big cities like Makassar. The number of crimes reported according to the regional resort unit in Makassar City in 2013 was 4,891 cases out of 19,042 cases, in 2014 there were 4,985 cases out of 18,769 cases and in 2015 there were 5,546 cases out of 19,826 cases. So it can be concluded that there was an increase in the number of crimes in Makassar City from 2013 to 2015 with an average of 26.74% of the number of crimes reported in South Sulawesi Province. Therefore, a crime-prone area information system will be built in Makassar City which can make it easier for the police to know the level of vulnerability of crime occurrence in Makassar City so that it can be used as a basis for determining the right strategy and direction for crime prevention for vulnerable areas and for the community is expected to increase vigilance in areas that are prone. This web-based geographics information system is designed and built using the waterfall method, PHP programming language with Laravel framework, Leaflet JavaScript Library and MySQL as the database server. The test results using blackbox testing found that all features were successfully run and functioned well and testing using User Acceptance Testing obtained a percentage score of 89.75%.”

Keywords: Criminal, Geographic Information Systems, Waterfall Method, Blackbox Testing, User Acceptance Testing

DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PERNYATAAN KEOTENTIKAN.....	ii
HALAMAN PERSETUJUAN PEMBIMBING	iii
HALAMAN PENGESAHAN.....	iv
KATA PENGANTAR.....	v
PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR.....	vii
ABSTRAK	viii
<i>ABSTRACT</i>	ix
DAFTAR ISI	x
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xvi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	3
BAB II TINJAUAN PUSTAKA	4
2.1 Kota Makassar	4
2.2 Kriminalitas	4
2.3 Sistem dan Informasi.....	5
2.3.1 Sistem.....	5
2.3.2 Informasi.....	7
2.4 Sistem Informasi dan Sistem Informasi Geografis.....	8
2.4.1 Sistem Informasi	8
2.4.2 Sistem Informasi Geografis (SIG).....	8
2.5 HTML (<i>Hypertext Markup Language</i>).....	10
2.6 PHP (<i>Hypertext Preprocessor</i>)	11
2.7 MySQL (<i>My Structured Query Language</i>).....	11
2.8 <i>Framework Laravel</i>	12
2.9 <i>Leaflet JavaScript (Leaflet JS)</i>	13

2.10	<i>Geo JavaScript Object Notation (GeoJSON)</i>	13
2.11	Metode Desain dan Pengembangan Sistem.....	14
2.11.1	<i>Unified Modeling Language (UML)</i>	14
2.11.2	<i>Entity Relationship Diagram (ERD)</i>	18
2.11.3	Model <i>Waterfall</i>	19
2.12	Efektivitas Sistem Informasi	21
2.13	<i>Blackbox Testing</i>	21
2.14	<i>User Acceptance Testing (UAT)</i>	23
2.15	Penelitian Terdahulu	23
BAB III METODE PENELITIAN.....		28
3.1	Waktu dan Lokasi Penelitian	28
3.2	Metode Pengumpulan Data	28
3.3	Instrumen Penelitian.....	29
3.4	Teknik Pengolahan dan Analisis Data	29
3.5	Tahap Penelitian	30
3.6	Teknik Pengujian Sistem	31
3.7	Metode Perancangan Aplikasi	31
3.8	Prosedur Sistem Yang Diusulkan	32
3.9	Rancangan Sistem	32
3.10	Rancangan UI.....	33
3.10.1	<i>User</i>	33
3.10.2	Admin	37
BAB IV HASIL DAN PEMBAHASAN.....		44
4.1	Implementasi Sistem	44
4.2	Implementasi <i>Entity Relationship Diagram (ERD)</i>	44
4.3	Relasi Antar Tabel	45
4.4	Implementasi <i>Activity Diagram</i>	45
4.4.1	<i>User</i>	45
4.4.2	Admin	47
4.5	<i>User Interface</i>	55
4.5.1	<i>User</i>	55
4.5.2	Admin	64

4.6	Skenario Pengujian Sistem	82
4.6.1	<i>Blackbox Testing</i>	82
4.6.2	<i>User Acceptance Testing</i>	88
BAB V KESIMPULAN DAN SARAN.....		92
5.1	Kesimpulan.....	92
5.2	Saran	92
DAFTAR PUSTAKA		93
LAMPIRAN.....		96

DAFTAR GAMBAR

Gambar 2.1 Karakteristik Sistem	6
Gambar 2.2 Siklus Pengolahan Data (<i>Data Processing Life Cycle</i>).....	7
Gambar 2.3 Komponen Sistem Informasi Geografis (SIG).....	9
Gambar 2.4 Tahapan Model <i>Waterfall</i>	20
Gambar 3.1 Alur Penelitian.....	30
Gambar 3.2 Prosedur Sistem Yang Diusulkan.....	32
Gambar 3.3 Rancangan <i>Use Case Diagram</i>	33
Gambar 3.4 Halaman Beranda <i>User</i>	34
Gambar 3.5 Halaman Profil <i>User</i>	35
Gambar 3.6 Halaman Peta <i>User</i>	36
Gambar 3.7 Halaman Kritik dan Saran <i>User</i>	37
Gambar 3.8 Halaman Login Admin.....	38
Gambar 3.9 Halaman Beranda Admin	39
Gambar 3.10 Halaman Profil Admin	40
Gambar 3.11 Halaman Data Kriminalitas Admin.....	41
Gambar 3.12 Halaman Input Data Kriminalitas Admin	41
Gambar 3.13 Halaman Data Kantor Polisi Admin.....	42
Gambar 3.14 Halaman Input Data Kantor Polisi	43
Gambar 3.15 Halaman Kritik dan Saran Admin.....	43
Gambar 4.1 <i>Entity Relationship Diagram</i>	44
Gambar 4.2 Relasi Antar Tabel.....	45
Gambar 4.3 <i>Activity Diagram</i> Halaman Peta Kriminalitas <i>User</i>	46
Gambar 4.4 <i>Activity Diagram</i> Halaman Kritik dan Saran <i>User</i>	47
Gambar 4.5 <i>Activity Diagram</i> Input Data Kecamatan	48
Gambar 4.6 <i>Activity Diagram</i> Kelola Data Kelurahan	49
Gambar 4.7 <i>Activity Diagram</i> Kelola Data Jenis Kriminalitas	50
Gambar 4.8 <i>Activity Diagram</i> Kelola Data Kriminalitas	51
Gambar 4.9 <i>Activity Diagram</i> Kelola Data Kantor Polisi.....	52
Gambar 4.10 <i>Activity Diagram</i> Kelola Data Artikel.....	53
Gambar 4.11 <i>Activity Diagram</i> Kelola Admin	54
Gambar 4.12 <i>Activity Diagram</i> Hapus Data Kritik dan Saran.....	55

Gambar 4.13 Tampilan Halaman Beranda.....	56
Gambar 4.14 Tampilan Halaman Detail Artikel	57
Gambar 4.15 Tampilan Halaman Profil	57
Gambar 4.16 Tampilan Halaman Peta Kriminalitas	59
Gambar 4.17 Tampilan Halaman Pilih <i>Date Range</i>	59
Gambar 4.18 Tampilan Halaman Pemetaan Kriminalitas Berdasarkan <i>Date Range</i>	60
Gambar 4.19 Tampilan Halaman Peta Kriminalitas Berdasarkan Kecamatan	61
Gambar 4.20 Tampilan Halaman Pilih <i>Date Range</i> Peta Kriminalitas Berdasarkan Kecamatan.....	61
Gambar 4.21 Tampilan Halaman Peta Kriminalitas Berdasarkan Kecamatan Sesuai dengan <i>Date Range</i>	62
Gambar 4.22 Tampilan Halaman Kritik dan Saran <i>User</i>	63
Gambar 4.23 Tampilan Halaman Tutorial	63
Gambar 4.24 Tampilan Halaman Login Admin	64
Gambar 4.25 Tampilan Halaman Dashboard Admin.....	65
Gambar 4.26 Tampilan Halaman Kelola Data Artikel.....	65
Gambar 4.27 Tampilan Halaman Tambah Data Artikel	66
Gambar 4.28 Tampilan Halaman Edit Data Artikel.....	66
Gambar 4.29 Tampilan Halaman Hapus Data Artikel	67
Gambar 4.30 Tampilan Halaman Kelola Data Kecamatan	68
Gambar 4.31 Tampilan Halaman Tambah Data Kecamatan	68
Gambar 4.32 Tampilan Halaman Edit Data Kecamatan	69
Gambar 4.33 Tampilan Halaman Hapus Data Kecamatan	69
Gambar 4.34 Tampilan Halaman Kelola Data Kelurahan	70
Gambar 4.35 Tampilan Halaman Tambah Data Kelurahan.....	70
Gambar 4.36 Tampilan Halaman Edit Data Kelurahan	71
Gambar 4.37 Tampilan Halaman Hapus Data Kelurahan.....	71
Gambar 4.38 Tampilan Halaman Kelola Data Jenis Kriminalitas.....	72
Gambar 4.39 Tampilan Halaman Tambah Data Jenis Kriminalitas	72
Gambar 4.40 Tampilan Halaman Edit Data Jenis Kriminalitas.....	73
Gambar 4.41 Tampilan Halaman Hapus Data Jenis Kriminalitas	73

Gambar 4.42 Tampilan Halaman Kelola Data Kantor Polisi.....	74
Gambar 4.43 Tampilan Halaman Tambah Data Kantor Polisi	74
Gambar 4.44 Tampilan Halaman Edit Data Kantor Polisi.....	75
Gambar 4.45 Tampilan Halaman Hapus Data Kantor Polisi	75
Gambar 4.46 Tampilan Halaman Kelola Data Kriminalitas	76
Gambar 4.47 Tampilan Halaman Tambah Data Kriminalitas	77
Gambar 4.48 Tampilan Halaman Edit Data Kriminalitas	78
Gambar 4.49 Tampilan Halaman Hapus Data Kriminalitas	79
Gambar 4.50 Tampilan Halaman Data Kritik dan Saran dari <i>User</i>	80
Gambar 4.51 Tampilan Halaman Hapus Data Kritik dan Saran	80
Gambar 4.52 Tampilan Halaman Kelola Admin	81
Gambar 4.53 Tampilan Halaman Tambah Admin	81
Gambar 4.54 Tampilan Halaman Hapus Data Admin	81

DAFTAR TABEL

Tabel 2.1 Simbol <i>Use Case Diagram</i>	15
Tabel 2.2 Elemen <i>Activity Diagram</i>	16
Tabel 2.3 Komponen-Komponen ERD.....	18
Tabel 3.1 Waktu Penelitian	28
Tabel 4.1 Parameter Tingkat Kriminalitas	58
Tabel 4.2 Pengujian Halaman Login.....	82
Tabel 4.3 Pengujian Kelola Data Kecamatan	83
Tabel 4.4 Pengujian Kelola Data Kelurahan.....	84
Tabel 4.5 Pengujian Kelola Data Jenis Kriminalitas	84
Tabel 4.6 Pengujian Kelola Data Kantor Polisi	85
Tabel 4.7 Pengujian Kelola Data Kriminalitas	86
Tabel 4.8 Pengujian Kelola Data Admin	86
Tabel 4.9 Pengujian Hapus Data Kritik dan Saran	87
Tabel 4.10 Pengujian Halaman Beranda.....	87
Tabel 4.11 Pengujian Halaman Peta Kriminalitas	88
Tabel 4.12 Pengujian Halaman Kritik dan Saran.....	88
Tabel 4.13 Bobot Nilai Jawaban	89
Tabel 4.14 Hasil Persentase	89

BAB I

PENDAHULUAN

1.1 Latar Belakang

Tindak kriminalitas di berbagai kota sudah menjadi permasalahan sosial yang membuat warga menjadi resah, terutama di kota – kota besar seperti kota Makassar. Kriminalitas atau tindak kriminal merupakan suatu perbuatan manusia yang melanggar hukum atau sebuah tindak kejahatan (Unayah & Sabarisman, 2015). Pelaku kriminalitas disebut juga dengan seorang kriminal. Kriminalitas bisa dilakukan oleh siapa saja, baik perempuan maupun laki-laki. Tindak kriminal juga disebabkan oleh banyak faktor.

Kepadatan penduduk yang tinggi dapat menimbulkan berbagai masalah yang terkait dengan kependudukan misalnya tingkat kemiskinan, terbatasnya lahan permukiman, kurangnya lapangan kerja dan masalah lainnya (Christiani, dkk, 2014). Kepadatan penduduk juga identik dengan tingginya tingkat kriminalitas. Hal ini disebabkan oleh terus meningkatnya angka pengangguran di daerah-daerah dengan kepadatan penduduk yang tinggi. Semakin padatnya penduduk di suatu wilayah, maka akan semakin luas pula ruang gerak para pelaku tindak kriminal karena semakin kecil kemungkinan mereka akan tertangkap (Fajri & Rizki, 2019; Sabiq & Nurwati, 2021).

Kota Makassar adalah salah satu kota dengan jumlah penduduk pada tahun pada tahun 2021 sebanyak 1.427.619 jiwa. Pertumbuhan penduduk dari tahun 2020 ke 2021 sebesar 0,26 persen. Luas wilayah Kota Makassar yaitu 175,77 km² dengan kepadatan penduduk 8.122 jiwa per kilometer persegi. Sementara Tingkat Pengangguran Terbuka (TPT) pada tahun 2021 sebesar 13,18 persen. Sehingga seiring dengan meningkatnya laju pertumbuhan penduduk dari tahun ke tahun juga menimbulkan peningkatan kriminalitas.

Kota Makassar juga merupakan salah satu kota yang pernah berpartisipasi dalam penilaian *Most Liveable City Indonesia* (MCLI) pada tahun 2017 oleh IAP. Hasil penilaian tersebut mengidentifikasi Kota Makassar sebagai kota paling tidak layak huni dengan mendapat nilai terendah pada aspek fasilitas keamanan dan terendah kedua dari aspek keamanan (Edison dkk, 2019). Jumlah kejahatan yang dilaporkan menurut kesatuan resort wilayah di Kota Makassar pada tahun 2013

sebanyak 4.891 kasus dari 19.042 kasus, pada tahun 2014 sebanyak 4.985 kasus dari 18.769 kasus dan pada tahun 2015 sebanyak 5.546 kasus dari 19.826 kasus. Sehingga dapat disimpulkan di Kota Makassar terjadi peningkatan jumlah kejahatan dari tahun 2013 sampai dengan 2015 dengan rata rata 26,74 % dari jumlah kejahatan yang dilaporkan di Provinsi Sulawesi Selatan (Badan Pusat Statistik, 2016).

Dari permasalahan diatas maka akan dibangun sebuah sistem informasi daerah rawan kriminalitas di Kota Makassar. Sistem informasi ini akan dibangun dengan menggunakan *framework Laravel* dan *Leaflet JavaScript Library*. Dengan adanya Sistem Informasi Geografis ini, diharapkan dapat membantu pihak kepolisian untuk mengetahui tingkat kerawanan terjadinya kriminalitas di suatu wilayah sehingga bisa dijadikan dasar untuk menentukan strategi yang tepat dan arah penanggulangan kejahatan bagi wilayah tersebut serta bagi masyarakat diharapkan untuk lebih waspada di daerah–daerah yang rawan.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diuraikan diatas, maka dapat diambil rumusan masalah sebagai berikut:

1. Bagaimana kebutuhan fungsional sistem informasi daerah rawan kriminalitas di Kota Makassar?
2. Bagaimana merancang dan membangun sistem informasi daerah rawan kriminalitas di Kota Makassar?
3. Bagaimana efektivitas sistem informasi daerah rawan kriminalitas di Kota Makassar?

1.3 Batasan Masalah

Berdasarkan rumusan masalah diatas, maka batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Aplikasi berbasis web ini dibangun dengan menggunakan *framework Laravel version 8.83.27* dan *Leaflet JavaScript Library version 1.7.1*.
2. Aplikasi ini menggunakan MySQL sebagai *database server*.
3. Data berasal dari POLRESTABES Makassar.

1.4 Tujuan Penelitian

Adapun tujuan yang ingin dicapai dalam penelitian ini adalah sebagai berikut:

1. Mengetahui kebutuhan fungsional sistem informasi daerah rawan kriminalitas di Kota Makassar.
2. Merancang dan membangun sistem informasi daerah rawan kriminalitas di Kota Makassar.
3. Mengetahui efektivitas sistem informasi daerah rawan kriminalitas di Kota Makassar.

BAB II

TINJAUAN PUSTAKA

2.1 Kota Makassar

Kota Makassar merupakan salah satu kota metropolitan di Indonesia sekaligus sebagai ibu kota Provinsi Sulawesi Selatan. Kota Makassar secara astronomis terletak antara $119^{\circ}24'17'38''$ bujur timur dan $5^{\circ}08'06'19''$. Berdasarkan posisi geografis, Kota Makassar memiliki batas-batas: Kabupaten Maros di sebelah utara, Kabupaten Gowa di sebelah selatan, Selat Makassar di sebelah barat dan Kabupaten Maros di sebelah timur. Luas wilayah Kota Makassar tercatat $175,77 \text{ km}^2$ yang meliputi 15 kecamatan dan 153 kelurahan. Kecamatan yang ada di Kota Makassar diantaranya, Kecamatan Mariso, Kecamatan Mamajang, Kecamatan Tamalate, Kecamatan Rappocini, Kecamatan Makassar, Kecamatan Ujung Pandang, Kecamatan Wajo, Kecamatan Bontoala, Kecamatan Ujung Tanah, Kecamatan Kepulauan Sangkarrang, Kecamatan Tallo, Kecamatan Panakkukang, Kecamatan Manggala, Kecamatan Biringkanaya dan Kecamatan Tamalanrea. Kecamatan terluas di Kota Makassar adalah Kecamatan Biringkanaya dengan luas $48,22 \text{ km}$ persegi atau mencakup 27,43 persen dari luas Kota Makassar secara keseluruhan. Sedangkan, kecamatan dengan luas terkecil adalah Kecamatan Kepulauan Sangkarrang dengan luas $1,54 \text{ km}$ persegi atau hanya sebesar 0,88 persen dari luas Kota Makassar.

Kota Makassar juga merupakan salah satu kota dengan tingkat kriminalitas yang tinggi. Berdasarkan statistik kriminal, jumlah kejahatan yang dilaporkan menurut Kesatuan Resor Wilayah pada tahun 2013 sebanyak 4.891 kasus, tahun 2014 sebanyak 4.985 kasus dan tahun 2015 sebanyak 5.546 kasus. Dapat dilihat dari tahun 2013 sampai dengan 2015, terjadi peningkatan jumlah kejahatan yang dilaporkan di Kota Makassar. Adapun tindak kriminalitas yang sering terjadi di Kota Makassar pada tahun 2014 yaitu pencurian, penganiayaan/kekerasan, penyalahgunaan/pengedaran narkoba, penipuan/penggelapan, perampokan dan perjudian.

2.2 Kriminalitas

Menurut Unayah & Sabarisman (2015), Kriminalitas atau tindak kriminal merupakan suatu perbuatan manusia yang melanggar hukum atau sebuah tindak

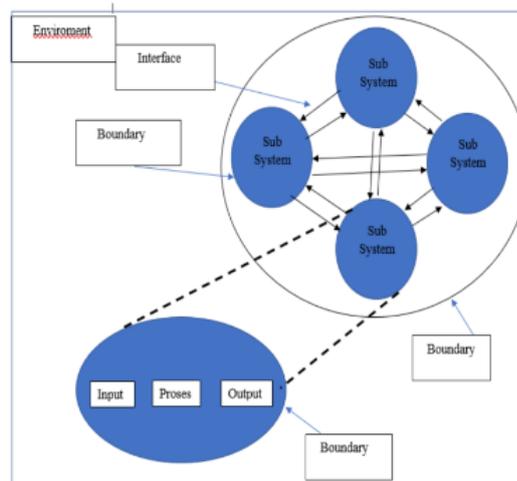
kejahatan. Kriminalitas merupakan perilaku yang dapat merugikan banyak pihak. Pelaku kriminalitas disebut juga dengan seorang kriminal. Kriminalitas bisa dilakukan oleh siapa saja, baik perempuan maupun laki-laki. Kriminalitas ini biasanya dilakukan oleh individu, kelompok maupun organisasi. Bentuk-bentuk tindak kriminal berupa kejahatan terhadap nyawa, kejahatan terhadap fisik/badan, kejahatan terhadap kesusilaan, kejahatan terhadap hak/barang dengan penggunaan kekerasan, kejahatan terhadap narkoba dan lain-lain.

Penyebab terjadinya kriminalitas disebabkan oleh banyak faktor. Biasanya dipengaruhi oleh kondisi sosial, meningkatnya kepadatan penduduk, kesenjangan sosial, tingkat pengangguran terbuka, dan faktor ekonomi. Terjadinya tindak kriminalitas juga dapat menimbulkan beberapa dampak negatif bagi korban. Dampak yang ditimbulkan oleh pelaku kriminalitas dapat berupa kerugian secara psikis pada korban, menimbulkan traumatis, rasa tidak aman dan cemas pada korban. Salah satu upaya untuk menekan tindak kriminalitas adalah dengan membuat para pelaku jera. Oleh karena itu, diperlukan tindakan tegas oleh aparat hukum.

2.3 Sistem dan Informasi

2.3.1 Sistem

Menurut Prehanto (2020), Sistem merupakan bagian-bagian komponen yang memiliki hubungan satu sama lain baik fisik maupun non fisik yang bekerja sama untuk mencapai tujuan yang telah ditetapkan secara harmonis. Sistem juga dapat diartikan sebagai suatu kesatuan dari berbagai komponen yang saling berkaitan dan bekerja sama untuk mewujudkan suatu tujuan tertentu. Secara umum, unsur-unsur yang mewakili suatu sistem adalah masukan (*input*), pengolahan (*proses*) dan keluaran (*output*). Sistem juga tidak terlepas dari lingkungan luarnya (*environment*). Untuk lebih jelasnya, karakteristik dari sistem bisa dilihat pada Gambar 2.1.



Gambar 2.1 Karakteristik Sistem

Sumber: (Prehanto, 2020)

1. Komponen (*Component*)

Sistem terdiri dari sejumlah komponen yang saling berinteraksi yang bekerja sama untuk membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem, dimana setiap subsistem memiliki sifat-sifat sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

2. Batasan Sistem (*Boundary*)

Ruang lingkup suatu sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau antara sistem dengan lingkungan luarnya.

3. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar sistem merupakan segala bentuk apapun yang ada di luar lingkup atau batasan sistem yang mempengaruhi operasi sistem. Dimana environment yang bersifat menguntungkan harus dijaga sedangkan yang bersifat merugikan harus tetap dijaga namun dikendalikan.

4. Penghubung Sistem (*Interface*)

Penghubung sistem merupakan media atau alat bantu yang menghubungkan sistem dengan subsistem yang lain.

5. Masukan Sistem (*Input*)

Masukan sistem adalah energi atau sumber daya yang dimasukkan ke dalam sistem, dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

6. Keluaran Sistem (*Output*)

Keluaran sistem merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang bermanfaat.

7. Pengolah Sistem (*Process*)

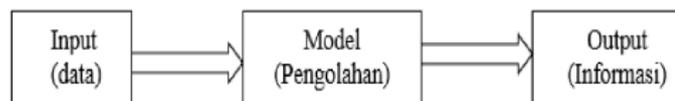
Pengolah sistem merupakan sebuah proses yang akan mengubah masukan (*input*) menjadi keluaran (*output*).

8. Sasaran Sistem (*Objective*)

Suatu sistem memiliki sasaran dan tujuan yang pasti. Sistem dapat dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.3.2 Informasi

Menurut Mulyani (2017), Informasi merupakan data yang telah diolah dan tersedia untuk seseorang, organisasi atau siapa saja yang membutuhkannya. Jadi, Informasi adalah hasil pengolahan data menjadi bentuk yang lebih bermanfaat untuk si penerima informasi. Sumber dari Informasi merupakan data yang menggambarkan peristiwa yang telah terjadi. Sumber ini harus diolah melalui sebuah siklus yang dinamakan sebagai siklus pengolahan data (*data processing life cycle*) (Prehanto, 2020). Siklus pengolahan data (*Data Processing Life Cycle*) dapat dilihat pada gambar 2.2.



Gambar 2.2 Siklus Pengolahan Data (*Data Processing Life Cycle*)

Sumber: (Prehanto, 2020)

Kualitas dari informasi tergantung dari tiga hal, diantaranya:

1. Akurat (*accurate*)

Informasi harus tepat dan bebas dari kesalahan dan tidak menyesatkan karena dari sumber informasi sampai ke penerima informasi mungkin akan banyak mengalami gangguan (*noise*) yang dapat mengubah atau merusak informasi tersebut.

2. Tepat waktu (*timelines*)

Informasi harus sampai kepada penerima dalam waktu yang tepat, tidak boleh terlambat. Informasi yang tersedia pada saat yang dibutuhkan serta informasi yang disajikan terbaru sangat berpengaruh, karena informasi

merupakan landasan dalam pengambilan keputusan. Apabila pengambilan keputusan terlambat, maka akan berakibat fatal bagi organisasi.

3. Relevan (*relevance*)

Informasi yang disajikan berpengaruh dan berguna dalam proses pengambilan keputusan sehingga dapat mempengaruhi keputusan penerimanya. Sebuah informasi yang baik bermanfaat bagi penerimanya.

2.4 Sistem Informasi dan Sistem Informasi Geografis

2.4.1 Sistem Informasi

Menurut Anggraeni (2017), Sistem Informasi merupakan kombinasi umum dari manusia, teknologi, perangkat lunak, jaringan komunikasi dan sumber daya data yang mengumpulkan, mengelola, dan menyebarkan informasi dalam suatu organisasi. Sistem informasi juga merujuk pada interaksi antara manusia, data, informasi, teknologi dan algoritma. Tujuan dari sistem informasi adalah untuk menghasilkan informasi. Salah satu contoh sistem informasi adalah sistem informasi akademik yang memungkinkan mahasiswa untuk memperoleh data-data akademiknya dan bahkan dapat mendaftarkan matakuliah yang diambil pada semester baru.

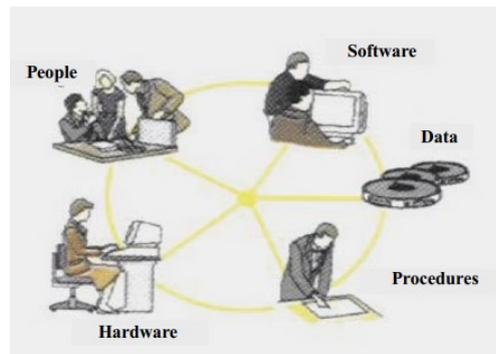
2.4.2 Sistem Informasi Geografis (SIG)

Sistem Informasi Geografis (SIG) atau *Geographic Information System* (GIS) adalah sebuah sistem yang didesain untuk menangkap, menyimpan, memanipulasi, menganalisa, mengatur, dan menampilkan seluruh jenis data geografis. Sistem informasi geografis dapat disimpulkan sebagai gabungan kartografi, analisis statistik dan teknologi sistem basis data (*database*) (Irwansyah, 2013). Sistem informasi geografis merupakan alat untuk mentransformasikan data spasial dari dunia nyata (*real world*). Data spasial merupakan data yang memiliki referensi posisi geografis objek yang digambarkan dengan menggunakan sistem koordinat garis lintang (*latitude*) dan garis bujur (*longitude*). Data spasial terbagi atas dua model data yaitu data *raster* dan data *vektor*. Model data *vektor* berbasiskan pada titik (*points*) dengan nilai koordinat (x,y) untuk membangun objek spasialnya. Objek yang dibangun terbagi menjadi tiga bagian, yaitu:

- a. Titik (*point*) merupakan representasi grafis yang paling sederhana pada suatu objek. Contohnya Lokasi Fasilitas Kesehatan dan lain-lain.

- b. Garis (*line*) merupakan bentuk *linear* yang menghubungkan dua atau lebih titik yang merepresentasikan objek dalam satu dimensi. Contohnya jalan, sungai dan lain-lain.
- c. Area (*polygon*) merupakan representasi objek dalam dua dimensi. Contohnya Danau, Lahan dan lain-lain.

Sedangkan data *raster* atau sel *grid* adalah data yang dihasilkan dari sistem penginderaan jauh. Pada data *raster*, objek geografis direpresentasikan sebagai struktur sel *grid* yang disebut dengan *pixel* (*picture element*). Data *raster* sangat berguna untuk merepresentasikan batas-batas yang berubah secara gradual, seperti jenis tanah, kelembaban tanah, vegetasi, suhu tanah, dan lain-lain (Irwansyah, 2013). Adapun komponen-komponen yang membangun sebuah Sistem Informasi Geografis (SIG) dapat dilihat pada Gambar 2.3.



Gambar 2.3 Komponen Sistem Informasi Geografis (SIG)

Sumber: (Fazal, 2008)

1. Perangkat Keras (*Hardware*)

Hardware terdiri dari sistem komputer di mana perangkat lunak Sistem Informasi Geografis (SIG) akan dijalankan. Pilihan sistem perangkat keras berkisar dari *personal computers* hingga *multi user supercomputers*. Komputer ini pada dasarnya harus memiliki *processor* yang efisien untuk menjalankan perangkat lunak dan memori yang cukup untuk menyimpan informasi (data). Contoh perangkat keras lainnya yang dibutuhkan untuk menjalankan sistem, diantaranya *monitor*, *CPU*, *Keyboard*, *Mouse* dan sebagainya.

2. Perangkat Lunak (*Software*)

Perangkat lunak SIG menyediakan fungsi dan alat yang diperlukan untuk menyimpan, menganalisis dan menampilkan informasi geografis.

3. Data

Data merupakan informasi yang dibutuhkan dan yang akan diolah dalam aplikasi. Data yang digunakan dapat berupa data grafis maupun atribut. Data grafis/spasial merupakan data yang bereferensi keruangan. Sedangkan data atribut merupakan data yang merepresentasikan aspek-aspek deskriptif dari fenomena yang dimodelkannya.

4. Method

SIG yang sukses beroperasi sesuai dengan rencana yang dirancang dengan baik, yang merupakan model dan praktik pengoperasian yang unik untuk setiap tugas. Ada berbagai teknik yang digunakan untuk pembuatan peta dan penggunaan lebih lanjut untuk proyek apa pun. Pembuatan peta dapat berupa raster otomatis ke pembuat vektor atau dapat dibuat vektor secara manual menggunakan gambar yang dipindai. Sumber peta digital ini dapat berupa peta yang disiapkan oleh lembaga survei atau citra satelit.

5. Orang (*People*)

Pengguna Sistem Informasi Geografis berkisar dari spesialis teknis yang merancang dan memelihara sistem hingga mereka yang menggunakannya untuk membantu dalam melakukan pekerjaan sehari-hari.

6. Jaringan (*Network*)

Dengan perkembangan IT yang pesat, saat ini yang paling mendasar adalah jaringan. Karena tanpa jaringan akan sulit untuk berkomunikasi dan berbagi informasi digital. SIG saat ini sangat bergantung pada internet untuk memperoleh dan berbagi kumpulan data geografis.

2.5 HTML (*Hypertext Markup Language*)

HTML (*Hypertext Markup Language*) dikembangkan pada tahun 1989 oleh Tim Berners-Lee. Tujuan dari pengembangan HTML ini adalah untuk menghubungkan satu halaman web dengan halaman web lainnya. HTML merupakan pondasi web karena HTML memberitahukan web browser bagaimana menyusun dan menyajikan konten di halaman web. HTML berfungsi untuk menampilkan berbagai komponen web. HTML dapat menampilkan tabel, tautan, teks, gambar dan audio (Solichin, 2016).

HTML mengalami perkembangan dari waktu ke waktu. Untuk menetapkan standar perintah HTML didirikan sebuah konsorsium W3C (*World Wide Web Consortium*) oleh Tim Berners-Lee. Adapun standar versi HTML yang pernah dikeluarkan oleh W3C, yaitu HTML 2.0 pada tahun 1995, HTML 3.2 pada Januari 1997, HTML 4.0 pada Desember 1997 yang kemudian mengalami revisi menjadi HTML 4.01 pada Desember 1999, dan XHTML pada Mei 2000 serta HTML5 yang terakhir dirilis. HTML5 atau versi terbaru dari HTML ini menambahkan lebih banyak fitur seperti penanganan video, audio dan penyimpanan lokal.

2.6 PHP (*Hypertext Preprocessor*)

PHP (*Hypertext Preprocessor*) dikembangkan oleh Rasmus Lerdorf untuk bahasa pemrograman berbasis web. PHP adalah bahasa *script*, artinya kode ini langsung dimasukkan ke dalam kode HTML. Tag HTML digunakan untuk membungkus bahasa pemrograman PHP yang ada di dalam file HTML. PHP juga merupakan bahasa pemrograman *server-side scripting*, dimana *script* dari PHP nantinya akan diproses di *server*. Adapun jenis *server* yang mendukung PHP, yaitu Apache, Xitami, dan *LiteSpeed*. PHP ini banyak digunakan untuk memprogram situs web dinamis. PHP bersifat *open source* dan dapat digunakan di *Windows*, *Linux*, *Unix*, dan *Mac OS* serta dapat dijalankan secara *runtime* melalui *console* dan dapat menjalankan perintah-perintah sistem (EMS, 2016).

Selain gratis dan cross platform, kelebihan lain yang dimiliki oleh PHP adalah mendukung hampir semua perangkat basis data. Mulai dari MySQL, Oracle, PostgreSQL, MariaDB, SQLite dan masih banyak lagi. PHP juga menyediakan banyak *library* bawaan yang bisa langsung digunakan. PHP memiliki *library* untuk menampilkan PDF *on-the-fly* (langsung di *browser*), menampilkan grafik *on-the-fly*, mengelola file, kompresi, enkripsi dan sebagainya.

2.7 MySQL (*My Structured Query Language*)

MySQL merupakan sebuah perangkat lunak manajemen basis data atau DBMS (*Database Management System*) yang menggunakan perintah dasar SQL (*Structured Query Language*). Dengan kata lain, MySQL merupakan DBMS (*Database Management System*) yang menggunakan SQL (*Structured Query Language*) sebagai bahasa penghubung antara perangkat lunak aplikasi dengan *database server*. MySQL bersifat *open source* dengan dua bentuk lisensi, yaitu *free*

software (perangkat lunak bebas) dan *shareware* (perangkat lunak berpemilik yang penggunaannya terbatas). MySQL adalah *server* basis data dengan lisensi GNU *General Public License* (GPL) sehingga dapat digunakan untuk keperluan individu ataupun komersil tanpa perlu membayar lisensi.

MySQL dikembangkan oleh sebuah perusahaan Swedia bernama MySQL AB sekitar tahun 1994-1995 yang pada saat itu bernama TcX Data Konsult AB, tetapi cikal bakal kodenya telah ada semenjak tahun 1979. Awal mulanya TcX membuat MySQL dengan tujuan mengembangkan aplikasi web untuk klien. TcX merupakan perusahaan pengembang *software* serta konsultan *database*. Saat ini MySQL sudah diakuisisi oleh Oracle Corp. *Software* mySQL bisa diunduh di <http://mysql.com>.

2.8 Framework Laravel

Laravel merupakan *framework* (kerangka kerja) aplikasi *web* berbasis PHP yang *open source*. *Framework Laravel* ini dapat membantu dalam pembuatan *website* dan memperindah tampilan *website*. *Laravel* fokus di bagian *end-user*, yang berarti fokus pada kejelasan serta kesederhanaan, baik penyusunan ataupun tampilan dan menciptakan fungsionalitas aplikasi *website* yang bekerja sebagaimana mestinya. *Framework* ini cukup terkenal dengan dokumentasinya yang lengkap dan selalu diperbarui secara berkala. *Laravel* juga dilengkapi dengan *command line tool* yang bernama “*Artisan*” yang dapat digunakan untuk *packaging bundle* dan *instalasi bundle* melalui *command prompt* (Aminudin, 2015).

Laravel dirilis dibawah lisensi MIT dengan kode sumber yang sudah disediakan oleh *Github*, sama seperti *framework-framework* yang lain. *Laravel* dibangun dengan konsep MVC (*Model-View-Controller*). *Model-View-Controller* atau MVC adalah sebuah metode untuk membuat sebuah aplikasi dengan memisahkan kode program dengan tiga komponen utama, yakni data (*Model*), tampilan (*View*) dan bagaimana cara memprosesnya (*Controller*).

1. *Model* mewakili struktur data. Model memiliki fungsi untuk membantu dalam pengelolaan *database*, misalnya mengatur, memanipulasi dan mengorganisasikan data yang ada di *database*.
2. *View* merupakan bagian yang mengatur tampilan atau menampilkan informasi dalam bentuk *Graphical User Interface* (GUI).

3. *Controller* merupakan bagian yang menghubungkan antara *model* dan *view*. *Controller* ini berisi perintah-perintah yang berfungsi untuk memproses suatu data dan mengirimkannya ke halaman *website* (Yudhanto & Prasetyo, 2019).

2.9 Leaflet JavaScript (Leaflet JS)

Leaflet JavaScript merupakan *library JavaScript* yang bersifat *open source*, digunakan untuk membangun aplikasi pemetaan berbasis web. *Leaflet* pertama kali dirilis pada tahun 2011 oleh Vladimir Agafonkin. *Library* ini mendukung sebagian *platform mobile* dan *platform desktop*, HTML5 dan CSS3 serta *Google Maps API*. *Library* ini juga dapat diintegrasikan dengan banyak *plugin*, mudah digunakan dan memiliki desain yang bagus. Peta digital ditampilkan menggunakan *Leaflet JavaScript* yang mendukung *file* berformat GeoJSON yang merupakan format data yang dapat menampung unsur-unsur geografis.

Dengan memanfaatkan *Leaflet*, seseorang tanpa latar belakang sistem informasi geografis (SIG) juga dapat dengan mudah menampilkan peta interaktif berbasis web pada *server*. *Leaflet* juga menyediakan fungsionalitas untuk menambahkan penanda, *popup*, garis *overlay* dan bentuk (Tanjaya dkk, 2016). *Leaflet JS* selalu mengalami perkembangan dari waktu ke waktu. Versi terbaru *leaflet* adalah *leaflet 1.9.3* yang rilis pada 18 November 2022. *Leaflet* bisa diunduh di <https://leafletjs.com/>. Beratnya hanya sekitar 42 KB.

2.10 Geo JavaScript Object Notation (GeoJSON)

GeoJSON merupakan format pertukaran data *geospasial* yang berbasis *JavaScript Object Notation*. GeoJSON menggunakan sistem referensi koordinat geografis *World Geodetic System 1984* dengan menggunakan satuan derajat desimal. GeoJSON memiliki beberapa fitur-fitur geometri, diantaranya:

1. *Point* yang berfungsi untuk merepresentasikan lokasi dan alamat,
2. *Line String* yang berfungsi untuk merepresentasikan rute, batas dan jalan, dan
3. *Polygon* yang berfungsi untuk merepresentasikan bidang tanah (Yusuf dkk, 2020).

Ketiga fitur-fitur dari GeoJSON ini dapat dikombinasikan menjadi kumpulan multi bagian dari fitur yang seragam, yaitu *MultiPoint*, *MultiLineString*,

MultiPolygon dan *GeometryCollection* (kumpulan multi bagian dari fitur yang beragam). Adapun karakteristik dari GeoJSON sendiri adalah sebagai data vektor, GeoJSON mempunyai ketepatan posisi yang dapat dipercaya. GeoJSON menggunakan format data text (JSON) sehingga lebih mudah diimplementasikan ke web dan lebih mudah di partisi menjadi bagian-bagian yang lebih kecil. Terdapat banyak layanan yang menggunakan GeoJSON seperti aplikasi navigasi dan perutean seluler. GeoJSON juga didukung oleh banyak *engine* pemetaan dan *software* GIS, termasuk *ArcGIS*, *Leaflet*, *MapServer*, *GeoServer*, *GeoDjango*, dan lain sebagainya.

2.11 Metode Desain dan Pengembangan Sistem

2.11.1 *Unified Modeling Language* (UML)

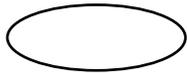
Menurut Jazman (2018), *Unified Modeling Language* (UML) adalah bahasa yang sudah menjadi standar untuk memvisualisasikan, mendeskripsikan, dan mendokumentasikan sistem piranti perangkat lunak. UML merupakan sebuah bahasa yang berbasis grafik atau representasi visual lainnya untuk memvisualisasi, menspesifikasikan, membangun dan mendokumentasikan proses pengembangan perangkat lunak. Dengan kata lain, UML merupakan teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada sistem. UML juga merupakan bahasa pemodelan yang berbasis OO (*Object-Oriented*). UML memiliki banyak diagram yang digunakan untuk melakukan pemodelan data maupun sistem, diantaranya *use case diagram* dan *activity diagram*.

a. *Use Case Diagram*

Menurut Mulyani (2017), *Use Case Diagram* yaitu diagram yang digunakan untuk menggambarkan hubungan antara sistem dengan aktor. *Use case* dapat menggambarkan jenis interaksi antara pengguna sistem dengan sistem itu sendiri. Untuk melakukan pemodelan, langkah pertama memerlukan suatu diagram yang dapat menunjukkan bagaimana aksi aktor dengan aksi sistem itu sendiri, seperti yang terlihat pada *use case diagram*. *Use case diagram* digunakan untuk mendeskripsikan sebuah sistem dan berguna untuk memperlihatkan proses aktivitas secara urut dalam sistem.

Use case diagram memiliki beberapa simbol seperti yang terlihat pada Tabel 2.1.

Tabel 2.1 Simbol *Use Case Diagram*

Simbol	Keterangan	Fungsi
	<i>Package</i>	Menambahkan paket baru dalam diagram.
	<i>Use Case</i>	Menambahkan <i>use case</i> dalam diagram.
	<i>Actors</i>	Menambahkan <i>actor</i> dalam diagram.
	<i>Unidirectional Association</i>	Menggambarkan relasi antara <i>actor</i> dengan <i>use case</i> .
	<i>Dependencies or Instantiates</i>	Menggambarkan kebergantungan (<i>dependencies</i>) antar item dalam diagram.
	<i>Generalization</i>	Menggambarkan relasi lanjut antar <i>use case</i> atau menggambarkan struktur pewarisan antar <i>actor</i> .

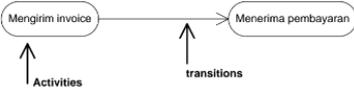
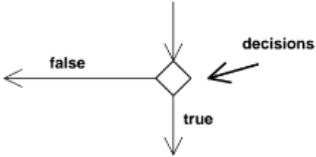
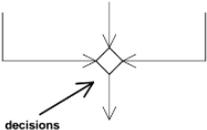
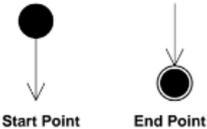
Sumber: (Samsuddin, 2018)

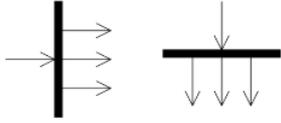
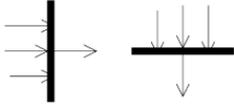
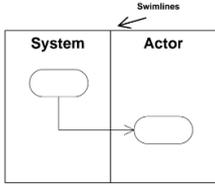
b. *Activity Diagram*

Menurut Mulyani (2017), *Activity Diagram* merupakan program yang digunakan untuk menggambarkan alur kerja (aktivitas) pada *use case* (proses), logika, proses bisnis serta hubungan antara aktor dengan aktivitas yang terlibat dalam *use case*. Dengan kata lain, *Activity diagram* menggambarkan aktivitas pengguna sistem dengan keseluruhan menu yang ada pada sistem. *Activity diagram* berfungsi untuk memperlihatkan urutan

aktivitas proses pada sistem dan membantu memahami proses secara keseluruhan serta menggambarkan aliran parallel, bercabang dan bersamaan dari sistem. Adapun elemen-elemen pada *activity diagram* dapat dilihat pada Tabel 2.2.

Tabel 2.2 Elemen Activity Diagram

Elemen	Fungsi
	<p>Activities merupakan elemen yang digunakan untuk menggambarkan aktivitas.</p> <p>Transitions merupakan elemen yang digunakan untuk menggambarkan transisi dari elemen yang satu ke elemen yang lainnya.</p>
	<p>Decisions merupakan elemen yang digunakan untuk percabangan logika.</p>
	<p>Merge point merupakan elemen yang digunakan untuk menggabungkan percabangan proses.</p>
	<p>Start point merupakan elemen yang digunakan untuk memulai <i>activity diagram</i>.</p> <p>End point merupakan elemen yang digunakan untuk mengakhiri <i>activity diagram</i>.</p>

Elemen	Fungsi
	<p>Concurrency merupakan elemen yang digunakan sebagai percabangan proses (bukan percabangan logika).</p>
	<p>Synchronization merupakan elemen yang digunakan untuk menggabungkan proses yang dipisahkan oleh <i>concurrency</i>.</p>
	<p>Swimlines merupakan elemen yang digunakan untuk memisahkan antara actor dan sistem ataupun antara aktor yang satu dengan aktor yang lain atau antara sistem yang satu dengan sistem yang lain.</p>
	<p>Sinyal merupakan acuan waktu yang bisa dijadikan <i>trigger</i> (pemicu) untuk aktivitas tertentu, misalnya setiap akhir jam kerja seluruh staff wajib memberikan laporan kepada manajer.</p>

Sumber: (Mulyani, 2017)

Adapun beberapa relasi yang terdapat pada *use case diagram* meliputi:

1. *Association*, menghubungkan link antar elemen,
2. *Generalization*, disebut juga *inheritance* (pewarisan), sebuah elemen dapat merupakan spesialisasi dari elemen lainnya,
3. *Dependency*, sebuah elemen bergantung dalam beberapa cara ke elemen lainnya, dan
4. *Aggregation*, bentuk *assosiation* dimana sebuah elemen berisi elemen lainnya.

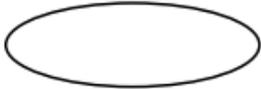
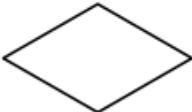
Tipe relasi (*stereotype*) yang mungkin terjadi pada *use case diagram* meliputi:

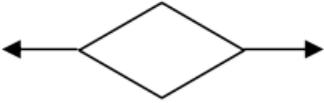
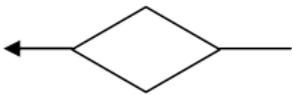
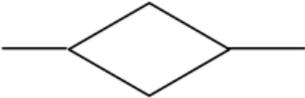
1. <<*include*>> merupakan kelakuan yang harus terpenuhi agar sebuah *event* dapat terjadi, dimana pada kondisi ini sebuah *use case* adalah bagian dari *use case* lainnya,
2. <<*extends*>> merupakan kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm, dan
3. <<*communicates*>>, mungkin ditambahkan untuk asosiasi yang menunjukkan asosiasinya adalah *communicates association*. Ini merupakan pilihan selama asosiasi hanya tipe *relationship* yang dibolehkan antara aktor dan *use case* (Samsuddin, 2018).

2.11.2 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan suatu diagram yang menjelaskan hubungan antar data dalam *database* berdasarkan objek-objek data yang mempunyai hubungan antar relasi. Atau dengan kata lain, *Entity Relationship Diagram* (ERD) merupakan representasi grafis yang mewakili logika *database*. Untuk membuat sebuah sistem *database* yang rapi dibutuhkan *Entity Relationship Diagram* (ERD). Adapun komponen-komponen ERD dapat dilihat pada Tabel 2.4.

Tabel 2.3 Komponen-Komponen ERD

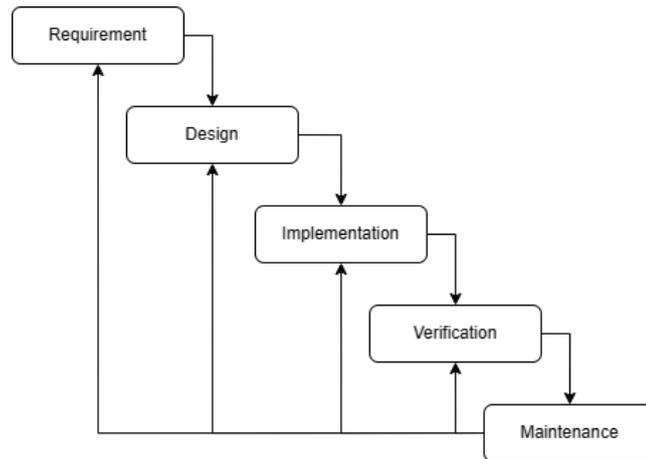
Notasi	Komponen	Keterangan
	Entitas	Individu yang mewakili suatu objek dan dapat dibedakan dengan objek yang lain.
	Atribut	Properti yang dimiliki oleh suatu entitas, dimana dapat mendeskripsikan karakteristik dari entitas tersebut.
	Relasi	Menunjukkan hubungan antara sejumlah entitas yang berbeda.

Notasi	Komponen	Keterangan
	Relasi 1:1	Relasi yang menunjukkan bahwa setiap entitas pada himpunan entitas pertama berhubungan dengan paling banyak satu entitas pada himpunan entitas kedua.
	Relasi 1:N	Relasi yang menunjukkan bahwa hubungan antara entitas pertama dengan entitas kedua adalah satu banding banyak atau sebaliknya. Setiap entitas dapat berelasi dengan banyak entitas pada himpunan entitas yang lain.
	Relasi N:N	Hubungan ini menunjukkan bahwa setiap entitas pada himpunan entitas yang pertama dapat berhubungan dengan banyak entitas pada himpunan entitas yang kedua, demikian juga sebaliknya.

Sumber: (Edi & Betshani, 2009)

2.11.3 Model *Waterfall*

Terdapat banyak model pengembangan perangkat lunak yang disebut dengan SDLC (*Software Development Life Cycle*), salah satunya adalah model *waterfall*. Model *waterfall* merupakan sebuah metode pengembangan sistem yang menggunakan pendekatan sistematis dan berurutan. Dimana setiap fase akan diselesaikan terlebih dahulu sebelum melangkah ke fase selanjutnya. Adapun tahapan pada metode *waterfall* dapat dilihat pada Gambar 2.4.

Gambar 2.4 Tahapan Model *Waterfall*

Sumber: (Wau, 2022)

1. *Requirement*

Requirement atau analisis kebutuhan yaitu mengumpulkan kebutuhan secara lengkap untuk dianalisis dan mendefinisikan kebutuhan apa saja yang harus dicapai oleh program. Informasi dapat diperoleh melalui wawancara, diskusi, atau survei.

2. *Design*

Design ini dilakukan dengan merancang desain perangkat lunak sebagai perkiraan sebelum dibuatnya kode. Desain sistem dapat dibuat menggunakan *Flowchart*, *Mind Map*, *Use Case Diagram*, *Entity Relationship Diagram (ERD)* dan sebagainya.

3. *Implementation*

Merupakan tahap dimana seluruh desain yang sebelumnya sudah dibuat diubah menjadi kode-kode program. Kode yang dihasilkan masih berbentuk modul-modul yang harus digabungkan di tahap selanjutnya.

4. *Verification*

Pada tahap ini, pengguna atau klien yang langsung melakukan pengujian pada sistem, apakah sistem telah sesuai dengan yang disetujui atau belum sesuai.

5. *Maintenance*

Tahap ini merupakan tahap terakhir dari model waterfall. Pemeliharaan berupa memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.

2.12 Efektivitas Sistem Informasi

Efektivitas merupakan kemampuan untuk mencapai keberhasilan kerja yang maksimal yang mencakup pencapaian target dalam hal kuantitas, kualitas dan waktu. Model kesuksesan sistem informasi DeLone dan Mclean (1992) merefleksikan 6 pengukuran sistem informasi, yaitu kualitas sistem (*system quality*), kualitas informasi (*information quality*), penggunaan (*use*), kepuasan pengguna (*user satisfaction*), dampak individual (*individual impact*), dan dampak organisasional (*organization impact*). Model ini didasarkan pada proses dan hubungan kausal. Model ini tidak mengukur keenam dimensi pengukuran kesuksesan sistem informasi secara independen tetapi mengukurnya secara keseluruhan satu mempengaruhi yang lainnya. Dari model ini, dapat dijelaskan bahwa kualitas sistem (*system quality*) dan kualitas informasi (*information quality*) secara mandiri dan bersama-sama mempengaruhi baik penggunaan (*use*) dan kepuasan pengguna (*user satisfaction*). Besarnya penggunaan (*use*) dapat mempengaruhi kepuasan pengguna (*user satisfaction*) secara positif atau negatif. Penggunaan (*use*) dan kepuasan pengguna (*user satisfaction*) mempengaruhi dampak individual (*individual impact*) dan selanjutnya mempengaruhi dampak organisasional (*organization impact*) (Yuliana, 2016).

Jadi semakin tinggi kualitas sistem yang dirasakan pengguna, maka semakin puas pula mereka pada kualitas sistem tersebut dan akan berpengaruh positif terhadap penggunaan sistem. Demikian pula, semakin tinggi kualitas informasi yang dirasakan pengguna, maka semakin puas pula mereka pada sistem informasi tersebut dan akan berpengaruh positif terhadap penggunaan sistem. Sehingga kepuasan pengguna merupakan matriks kunci indikator kesuksesan sistem informasi (DeLone & Mclean, 1992).

2.13 *Blackbox Testing*

Blackbox Testing merupakan teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak. *Blackbox testing* bekerja dengan mengabaikan struktur kontrol sehingga perhatiannya difokuskan pada informasi domain. Keuntungan penggunaan *blackbox testing* meliputi:

1. Penguji tidak perlu memiliki pengetahuan tentang bahasa pemrograman tertentu,

2. Pengujian dilakukan dari sudut pandang pengguna, ini membantu untuk mengungkapkan ambiguitas atau inkonsistensi dalam spesifikasi persyaratan, dan
3. *Programmer* dan *tester* keduanya bergantung satu sama lain (Jaya, 2018).

Terdapat banyak metode untuk melaksanakan *blackbox testing*, diantaranya: *Equivalence Class Partitioning*, *Boundary Value Analysis*, dan *Error Guessy*, *Decision Table Testing*, dan *State Transition Testing*.

1. *Equivalence Class Partitioning* merupakan teknik pengujian berdasarkan masukan setiap menu yang ada pada sistem. Teknik ini memeriksa *input* dan *output* dengan membagi *input* ke dalam kelas-kelas yang setara. Tujuan dari teknik ini yaitu untuk mengidentifikasi apakah ada fungsi yang salah atau hilang dan mengidentifikasi kesalahan pada *interface* serta pada struktur data atau akses *database*.
2. *Boundary Value Analysis* merupakan teknik pengujian yang fokus pada proses masukan dengan menguji nilai batas atas dan nilai batas bawah. Teknik ini berfungsi untuk mengidentifikasi kesalahan yang muncul akibat keterbatasan *input*.
3. *Error Guessy* merupakan teknik pengujian dengan cara menebak masukan dan keluaran yang bertujuan untuk memperbaiki kesalahan yang ada pada sistem. Dalam pengujian ini, penguji mencoba mengidentifikasi jenis-jenis kesalahan yang mungkin terjadi dalam sistem dan menciptakan skenario pengujian yang menggambarkan situasi tersebut.
4. *Decision Table Testing* merupakan teknik yang menggunakan tabel untuk menggambarkan kombinasi kondisi *input* dan *output* yang diharapkan. Setiap kombinasi kondisi menghasilkan tindakan tertentu atau hasil, dan tabel ini membantu mengidentifikasi skenario pengujian yang mencakup berbagai kemungkinan kombinasi.
5. *State Transition Testing* merupakan teknik pengujian yang cocok digunakan untuk perangkat lunak yang memiliki berbagai keadaan atau status yang berubah seiring waktu. Teknik ini berfokus pada transisi antara status dan mengidentifikasi kasus uji yang mencakup perubahan keadaan.

2.14 *User Acceptance Testing (UAT)*

User Acceptance Testing (UAT) merupakan tahapan pengujian dalam siklus pengembangan perangkat lunak dimana pengguna akhir melakukan pengujian pada perangkat lunak tersebut untuk memastikan bahwa perangkat lunak tersebut telah memenuhi persyaratan atau tujuan yang telah ditentukan sebelumnya. UAT dilakukan setelah tahap pengembangan perangkat lunak selesai dilakukan dan sebelum peluncuran aplikasi. Hasil dari UAT ini akan digunakan untuk menentukan apakah perangkat lunak siap untuk digunakan atau perlu adanya perbaikan lebih lanjut sebelum diluncurkan untuk meningkatkan kualitas atau kepuasan pengguna akhir. Dengan melakukan UAT, tim pengembang dapat mengidentifikasi kekurangan atau masalah dalam perangkat lunak, mengambil Langkah-langkah perbaikan yang diperlukan, serta memastikan bahwa perangkat lunak telah diuji secara memadai untuk kualitas dan kelayakan. Perhitungan dalam UAT menggunakan *skala likert*. *Skala likert* merupakan metode pengukuran yang menggunakan skala untuk mengatur tingkat persetujuan terhadap sesuatu.

Jenis-jenis *User Acceptance Testing*:

1. *Alpha Testing* merupakan pengujian yang dilakukan oleh tim pengembang. Pengujian ini dilakukan oleh tim internal sebelum dirilis ke pengguna akhir. Tujuan pengujian ini untuk mengidentifikasi masalah atau kekurangan pada perangkat lunak sebelum menuju ke *beta testing*.
2. *Beta Testing* merupakan pengujian perangkat lunak sebelum dirilis. Pengujian ini melibatkan langsung calon pengguna akhir. Tujuan dari pengujian ini, untuk mengetahui apakah perangkat lunak sudah siap diluncurkan secara resmi atau perlu dilakukan perbaikan lebih lanjut. Dengan adanya *beta testing*, dapat mengetahui tingkat penerimaan penggunaan terhadap aplikasi dari sisi pengguna akhir (Maindra & Hadikusuma, 2023).

2.15 Penelitian Terdahulu

Sistem Informasi Geografis Pemetaan Titik Lokasi Daerah Rawan Kriminalitas Kota Solok (Studi Kasus: Polres Solok Kota) oleh Andrianto & Muhammad Jazman (2018): sistem yang dikembangkan dengan menggunakan bahasa pemrograman PHP 5, database MySQL (PHPMyadmin) sebagai wadah

sentralisasi data, JQuery (*Javascript Query*), HTML (*Hypertext Markup Language*) sebagai *platform Front-End* dan *Back-End* serta *leaflet* untuk penampilan peta. Sistem ini dapat menampilkan data lokasi kriminalitas yang ada di Kota Solok sehingga bisa membantu pihak Polres Solok Kota untuk mengambil keputusan terhadap tindakan dan kewaspadaan di daerah yang rawan serta membantu masyarakat untuk memantau perkembangan tingkat kerawanan kriminalitas di Kota Solok. Kelemahan dari sistem ini, dibagian klasterisasi proses hanya dapat ditampilkan dalam pilihan kecamatan dan tahun.

Sistem Informasi Geografis Daerah Rawan Kriminalitas di Kabupaten Garut oleh Setiawan & Salam (2021): sistem ini dikembangkan dengan menggunakan bahasa pemrograman PHP, *framework laravel*, MySQL dan *framework bootstrap* untuk mengatur keindahan tampilan. Metodologi yang digunakan adalah *Rational Unified Process* (RUP) dan pengujian dengan menggunakan *blackbox testing*. Sistem ini mampu menampilkan titik lokasi kriminalitas dan masyarakat bisa melakukan pelaporan secara online tentang kritik, saran maupun pengaduan lainnya.

Rancang Bangun Sistem Informasi Geografis Pemetaan Daerah Rawan Kriminalitas di Kota Mataram Berbasis Web (Syahputra dkk, 2019): sistem ini dikembangkan dengan menggunakan *framework Codeigniter*, bahasa pemrograman PHP dan HTML serta *Google Map API* untuk menampilkan rute fasilitas Kesehatan. Sistem ini bertujuan untuk memberikan informasi tentang lokasi, pemetaan dan membantu orang untuk mengetahui titik distribusi kriminalitas di Kota Mataram. Pada sistem ini data kriminalitas ditampilkan dalam bentuk *polygon* dan per kecamatan.

Sistem Informasi Geografis Pemetaan Persebaran Kriminalitas di Kota Denpasar oleh Damayanti dkk, (2016): sistem ini dikembangkan menggunakan Google Maps dan Metode *Single Exponential Smoothing*. Metode *Single Exponential Smoothing* digunakan untuk melakukan perhitungan prediksi jumlah kriminalitas pada bulan selanjutnya. Sistem ini dapat menampilkan peta informasi pola persebaran kriminalitas di Kota Denpasar. Pada sistem ini data kriminalitas ditampilkan dalam bentuk *polygon* per kelurahan di Kota Denpasar.

Sistem Informasi Geografis (GIS) Wilayah Kriminalitas Berbasis Web di Kabupaten Pali oleh Cahyo & Sahfitri, (2019): sistem ini dikembangkan dengan menggunakan bahasa pemrograman PHP. Metodologi pengembangan sistem yang digunakan adalah Agile Development Software. Sistem ini dapat menampilkan lokasi wilayah rawan kriminalitas dan tingkat kriminalitas di setiap daerah di Kabupaten Pali. Sistem ini dapat memudahkan pihak kepolisian untuk mengetahui tingkat kriminalitas tiap daerah yang bisa digunakan sebagai bahan acuan mengatasi lokasi-lokasi yang rawan.

No	Peneliti	Judul Penelitian	Hasil Penelitian	Persamaan	Perbedaan
1	Andrianto & Muhammad Jazman (2018)	Sistem Informasi Geografis Pemetaan Titik Lokasi Daerah Rawan Kriminalitas Kota Solok (Studi Kasus: Polres Solok Kota)	Sistem ini dapat menampilkan data lokasi kriminalitas yang ada di Kota Solok.	Menggunakan bahasa pemrograman PHP dan HTML, <i>database</i> MySQL serta <i>blackbox testing</i> untuk pengujian sistem.	Menggunakan <i>framework</i> Laravel versi 8 serta pada halaman tambah data kriminalitas bisa langsung <i>drag and drop marker</i> untuk mengisi <i>latitude</i> dan <i>longitude</i> .
2	Setiawan & Salam (2021)	Sistem Informasi Geografis Daerah Rawan Kriminalitas di Kabupaten Garut	Sistem ini mampu menampilkan titik lokasi kriminalitas dan masyarakat bisa melakukan pelaporan secara online baik kritik, saran maupun pengaduan lainnya.	Menggunakan bahasa pemrograman PHP, <i>framework laravel</i> , <i>database</i> MySQL serta <i>blackbox testing</i> untuk pengujian sistem.	Menggunakan model <i>waterfall</i> dan dapat menampilkan tingkat kriminalitas tiap kecamatan yang ditandai dengan warna tiap <i>polygon</i> .
3	Syahputra dkk, (2019)	Rancang Bangun Sistem Informasi Geografis Pemetaan	Sistem ini bertujuan untuk memberikan informasi tentang lokasi	Menggunakan bahasa pemrograman PHP dan HTML, <i>Google Map API</i>	Menggunakan <i>framework</i> Laravel versi 8 serta <i>leaflet javascript library</i> untuk

No	Peneliti	Judul Penelitian	Hasil Penelitian	Persamaan	Perbedaan
		Daerah Rawan Kriminalitas di Kota Mataram Berbasis Web.	dan pemetaan kriminalitas di Kota Mataram.	untuk menampilkan rute fasilitas Kesehatan serta <i>blackbox testing</i> untuk pengujian sistem.	menampilkan titik lokasi kriminalitas.
4	Damayanti dkk, (2016)	Sistem Informasi Geografis Pemetaan Persebaran Kriminalitas di Kota Denpasar	Sistem ini dapat menampilkan peta informasi pola persebaran kriminalitas di Kota Denpasar.	Menggunakan bahasa pemrograman PHP, <i>database MySQL</i> .	Menggunakan <i>leaflet javascript library</i> untuk menampilkan titik lokasi kriminalitas dan dapat menampilkan titik kriminalitas.
5	Cahyo & Sahfitri, (2019)	Sistem Informasi Geografis (GIS) Wilayah Kriminalitas Berbasis Web di Kabupaten Pali	Sistem ini dapat menampilkan lokasi wilayah rawan kriminalitas dan tingkat kriminalitas di setiap daerah di Kabupaten Pali.	Menggunakan bahasa pemrograman PHP.	Menggunakan model <i>waterfall</i> dan dapat menampilkan tingkat kriminalitas tiap kecamatan yang ditandai dengan warna tiap <i>polygon</i> .