



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1. Pengertian Sistem Informasi

Menurut Laudon (2012), sistem informasi adalah komponen-komponen yang saling berkaitan yang bekerja sama untuk mengumpulkan, mengolah, menyimpan dan menampilkan informasi untuk mendukung pengambilan keputusan, koordinasi, pengaturan, analisa dan visualisasi pada sebuah organisasi.

Menurut Yakub (2012:10), menyatakan bahwa Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk tujuan tertentu.

2.1.2 Pengertian Bencana

Sebagai negara rawan bencana, sangat penting bagi Indonesia memiliki kesiapsiagaan dalam mengantisipasi bencana untuk dapat mengurangi dampak yang ditimbulkan oleh bencana tersebut. Upaya pencegahan dan mitigasi bencana menjadi sangat penting untuk mengurangi risiko bencana yang mungkin timbul. Sumber : (Majelis Guru Besar ITB, 2009 Dalam Dhuha, Arif dan Abdi 2016:326). Sedangkan menurut UU No.24 tahun 2007 bencana adalah peristiwa atau rangkaian peristiwa yang mengancam dan mengganggu kehidupan dan penghidupan masyarakat yang disebabkan baik oleh faktor alam dan faktor non alam maupun manusia, sehingga mengakibatkan timbulnya korban jiwa manusia, kerusakan lingkungan, kerugian harta benda dan dampak Psikologis.

2.1.3 Pengertian Bencana Alam

Menurut *Asian Disaster Reduction Center* “bencana alam adalah gangguan serius pada masyarakat yang bisa menyebabkan kerugian secara meluas dan dirasakan oleh semua lapisan masyarakat dan lingkungan, dan untuk mengatasinya dibutuhkan kemampuan yang melebihi kemampuan manusia”.



Bencana alam menjadi salah satu peristiwa yang tidak dapat diprediksi kapan terjadinya dan tentunya sangat tidak diharapkan oleh semua makhluk hidup.

2.1.4 Pengertian Kapasitas Bencana

Kapasitas merupakan seperangkat kemampuan yang memungkinkan masyarakat untuk meningkatkan daya tahan terhadap efek bahaya yang mengancam/merusak, dan meningkatkan ketahanan serta kemampuan masyarakat untuk mengatasi dampak dari kejadian yang membahayakan. Kekuatan/potensi yang ada pada diri setiap individu dan kelompok sosial. Kapasitas ini dapat berkaitan dengan sumberdaya, keterampilan, pengetahuan, kemampuan organisasi dan sikap untuk bertindak dan merespon suatu krisis. (Anderson & Woodrow, 1989 dalam jurnal Dhuha, Arif dan Abdi 2016:328).

2.1.5 Lampiran Peraturan Menteri No 33 Tahun 2006 Tentang Mitigasi Bencana.

Menurut jurnal Linda Tondobalo (2011:62) Ada empat hal penting dalam mitigasi bencana, yaitu :

1. Tersedia informasi dan peta kawasan rawan bencana untuk tiap jenis bencana.
2. Sosialisasi untuk meningkatkan pemahaman dan kesadaran masyarakat dalam menghadapi bencana, karena bermukim di daerah rawan bencana.
3. Mengetahui apa yang perlu dilakukan dan dihindari,serta mengetahui cara penyelamatan diri jika bencana timbul.
4. Pengaturan dan penataan kawasan rawan bencana untuk mengurangi ancaman bencana.

Berbagai potensi bencana diuraikan dalam pedoman ini :

1. Bencana Banjir
2. Bencana Longsor
3. Bencana letusan Gunung Api
4. Bencana Gempa Bumi
5. Bencana Tsunami
6. Bencana Kebakaran



7. Bencana Kekeringan
8. Bencana Angin Siklon Tropis
9. Bencana Wabah Penyakit
10. Bencana Kegagalan Teknologi
11. Bencana Konflik

2.2. Teori Khusus

2.2.1. Pemrograman Berorientasi Objek (PBO)

Sukanto dan Shalahuddin (2016:99), menjelaskan Pemrograman berorientasi obyek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan obyek yang berisi data dan operasi yang diberlakukan terhadapnya.

Menurut Nugroho (2016:12) , ada beberapa karakter yang menjadi ciri-ciri pendekatan berorientasi objek adalah sebagai berikut:

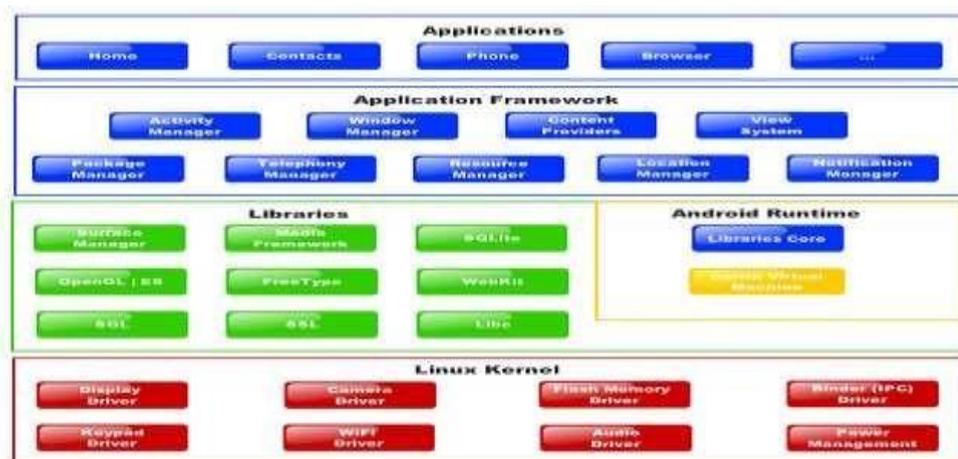
1. Pendekatan lebih pada data dan bukannya pada prosedur atau fungsi
2. Program besar dibagi pada apa yang dinamakan objek-objek
3. Struktur data dirancang dan menjadi karakteristik dari objek-objek.
4. Fungsi- fungsi yang mengoperasikan data tergabung dalam suatu objek yang sama
5. Data tersembunyi dan terlindung dari fungsi atau prosedur yang ada diluar.
6. Objek-objek dapat saling berkomunikasi dengan saling mengirim *message* satu sama lain.
7. Pendekatan adalah dari bawah ke atas.

2.2.2 Pengertian Android

Perkembangan teknologi semakin canggih, salah satu perkembangan teknologi yang lagi populer pada masa kini adalah *android*. *Android* bukan lagi barang tabu bagi manusia, tetapi *android* sudah menjadi bagian dalam kehidupan manusia pada masa kini. *Android* memberikan kecanggihan dengan *fitur-fitur* yang canggih sehingga dapat mempermudah segala aktivitas manusia. Seperti

memberikan *fitur game, music player, sampai vedio player* sehingga menjadi lebih mudah menikmati hiburan kapan saja dan dimanapun.

Nazrudin Sfaat (2011) menjelaskan “*Android* adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware*, dan aplikasi”. Ada juga pendapat lain dari Jubilee Enterpri.(2015:1), mengatakan “*Android* merupakan sistem operasi *open source*, dan *google* merilis kodenya dibawah lisensi *apache*. Kode *open source* dan lisensi perizinan pada *android* memungkinkan perangkat lunak untuk dimodifikasi secara bebas dan didistribusikan oleh para pembuat perangkat, operator nirkabel dan pengembang aplikasi. *Android* memiliki arsitektur, dapat ditampilkan pada gambar 2.1



(Sumber :Nasaruddin, 2011)

Gambar 2.1 Arsitektur Android

- Applications*: *Android* akan menggabungkan dengan serangkaian aplikasi inti termasuk *clien email*, program SMS, kalender, peta, *browser*, kontak, dan lain-lain.
- Application Framework*: Dengan menyediakan sebuah *platform* pengembangan yang terbuka, pengembang *android* menawarkan kemampuan untuk membangun aplikasi yang sangat kaya dan inovatif.
- Libraries*: *Android* termasuk satu set pustaka C/C++ yang digunakan oleh berbagai komponen sistem *android*.



- d. *Android Runtime*: *Android* termasuk satu set perpustakaan inti yang menyediakan sebagian besar fungsi yang tersedia di perpustakaan inti dari bahasa pemrograman Java.
- e. *Linux Kernel*: *Android* bergantung pada *Linux versi 2.6* untuk layanan sistem inti seperti keamanan, manajemen memori, manajemen proses, *network stack*, dan model pengemudi. Kernel juga bertindak sebagai lapisan abstraksi antara *hardware* dan seluruh *software stack*.

2.2.3. Fitur-fitur *Android*

Android memiliki sejumlah besar komunitas pengembang aplikasi yang memperluas fungsionalitas perangkat, umumnya ditulis dalam versi kustomisasi bahasa pemrograman java. Berikut ini *fitur-fitur android* :

Tabel 2.1 *Fitur-fitur Android*

Antarmuka	Layar sistem operasi <i>android</i> menampilkan antarmuka yang bagus dan intuitif.
Konektifitas	GSM/EDGE, IDEN,CDMA,EV-DO,UMTS,Bluetooth, WiFi,LTE, NFC, dan WiMAX.
Penyimpanan	<i>SQLite</i> , database relasional ringan yang digunakan untuk penyimpanan data
<i>Messaging</i>	SMS dan MMS
Media	H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC,AAC 5.1, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, dan BMP
<i>Web browser</i>	Berdasarkan <i>layout engine WebKit</i> , dengan <i>Chrome's V8 JavaScript engine</i> yang mendukung HTML5 dan CSS3
<i>Multi-touch</i>	<i>Android</i> memiliki dukungan <i>multi-touch</i>
<i>Multi-tasking</i>	Pengguna dapat berpindah dari satu <i>task</i> ke <i>task</i> lainnya dan beberapa aplikasi dapat berjalan pada waktu yang bersamaan

Lanjutan **Tabel 2.1** *Fitur-fitur Android*

Ukuran- <i>widgets</i>	<i>Widgets</i> dapat diatur ukurannya sesuai dengan keinginan pengguna, baik memperbesar ukuran untuk menampilkan lebih banyak konten, atau memperkecil ukuran untuk menyediakan lebih banyak ruang.
<i>Multi-language</i>	Mendukung teks dalam berbagai bahasa
GCM	<i>Google Cloud Messaging</i> (GCM) adalah layanan yang memungkinkan <i>developer</i> untuk mengirim data pesan pendek kepada pengguna tanpa menggunakan solusi <i>sync proprietary</i>
<i>Wi-fi Direct</i>	Teknologi yang memungkinkan aplikasi untuk menemukan dan memasang secara langsung, melalui koneksi <i>peer-to-peer-high-bandwidth</i>
<i>Android beam</i>	Teknologi berbasis NFC populer yang memungkinkan pengguna untuk berbagi konten secara langsung hanya dengan menyentuh dua perangkat

(Sumber : *Jubilee Enterprise* (2015))

2.2.4 Web Server

Menurut Sadeli (2011:2), menjelaskan bahwa ”*Web server* adalah sebuah perangkat lunak *server* yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan *web browser* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman *web* yang umumnya berbentuk dokumen HTML”.

Nugroho (2004:6), Menjelaskan *Web server* adalah sebuah bentuk server yang khusus digunakan untuk menyimpan halaman *web site* atau *home page*.



Secara umum *Web server* terbagi menjadi dua tipe yaitu *web server* secara *offline* dan *web server* secara *online*. *Web server* ini dapat juga diartikan sebagai rumahnya *website* atau situs. Setiap situs yang ada di internet pastilah berjalan diatas *web server*. Jadi peranan *web server* ini sangatlah penting dalam pembuatan situs.

2.2.5. Perancangan Perangkat Lunak

Rosa A.S dan M. Salahudin (2014:4), menjelaskan Perancangan perangkat lunak merupakan pembangunan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomi yang dipercaya dan bekerja secara efisien menggunakan mesin.

2.2.6. Metode Pengembangan Sistem

Metode pengembangan yang penulis pakai adalah metode *Rational Unified Process* (RUP). RUP merupakan suatu metode rekayasa perangkat lunak yang dikembangkan dengan mengumpulkan berbagai *best practises* yang terdapat dalam industri pengembangan perangkat lunak. Ciri utama metode ini adalah menggunakan *use-case driven* dan pendekatan iteratif untuk siklus pengembangan perangkat lunak. RUP menggunakan konsep *object oriented*, dengan aktifitas yang berfokus pada pengembangan model dengan menggunakan *Unified Model Language* (UML). Melalui gambar dibawah dapat dilihat bahwa RUP memiliki yaitu:

a. Dimensi pertama

Dimensi ini mewakili aspek-aspek dinamis dari pengembangan perangkat lunak, Digambarkan secara horizontal. Aspek ini dijabarkan dalam tahapan pengembangan atau fase. Setiap fase akan memiliki suatu *major milestone* yang menandakan akhir dari awal dari phase selanjutnya. Setiap phase dapat berdiri dari satu atau beberapa iteraksi. Dimensi ini terdiri atas :



1. *Inception*

Pada tahap ini dilakukan pemodelan proses bisnis yang dibutuhkan dan mendefinisikan kebutuhan user yang akan dibuat.

2. *Elaboration*

Tahap ini lebih difokuskan pada perencanaan arsitektur sistem, dan mendeteksi apakah arsitektur yang dibuat diinginkan, serta mendeteksi resiko yang mungkin terjadi dari arsitektur yang buat. Tahap ini lebih pada analisis dan desain sistem serta implementasi sistem yang fokus pada purwarupa sistem (prototype).

3. *Construction*

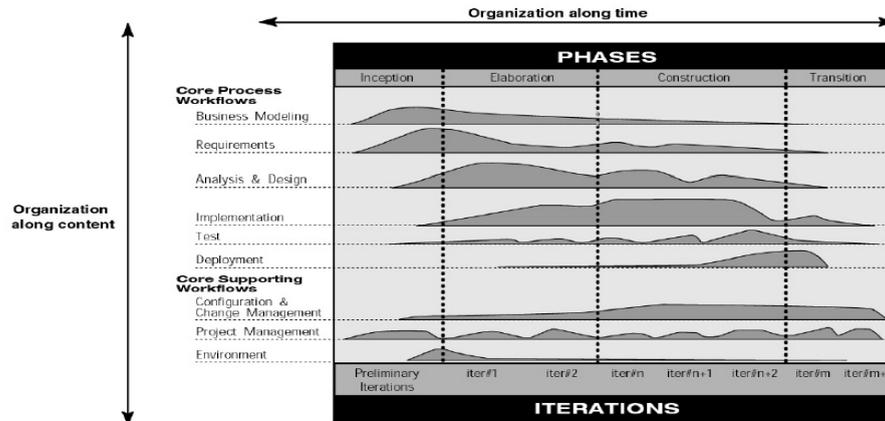
Pada tahap ini dilakukan pengembangan komponen dan fitur-fitur sistem. Tahap ini lebih pada implementasi dan pengujian sistem yang fokus pada imple-mentasi perangkat lunak pada kode program.

4. *Transition*

Pada tahap ini dilakukan deployment atau instalasi sistem agar dapat dimengerti oleh user. Aktifitas pada tahap ini termasuk pada pelatihan user, pemeliha-raan, dan pengujian sistem.

b. Dimensi kedua

Dimensi ini mewakili aspek- aspek statis dari proses pengembangan perangkat lunak yang dikelompok kan ke dalam beberapa disiplin, Di gambarkan secara vertikal. Proses pengembangan perangkat lunak yang dijelaskan kedalam beberapa disiplin terdiri dari empat elemen penting, yakni *who is doing, what, how* dan *when*. Dimensi ini terdiri atas *Business Modeling, Requirement, Analysis and Design, Implementation, Test, Deployment, Configuration* dan *Change Manegement, Project Management, Environtment*.



Gambar 2.2 Arsitektur *Rational Unified Process* (RUP)

(sumber : jurnal sutedi,melda agarina,2017)

2.2.7. *Unified Modeling Language* (UML)

Rosa A.S & M. Shalahuddin (2014:133), UML adalah salah satu standar bahasa yang digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisa dan desain serta menggambarkan arsitektur dalam pemrograman berorientasi objek.

Menurut Sukamto & M. Shalahuddin (2013:133). UML adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi obyek.

UML menyediakan bahasa pemodelan visual yaitu proses penggambaran informasi-informasi secara grafis dengan notasi-notasi baku yang telah disepakati sebelumnya.

Dengan menggunakan pomedelan UML ini,pengembang dapat melakukan :

1. Tinjauan umum bagaimana arsitektur sistem secara keseluruhan
2. Penelaahan bagaimana objek-objek dalam sistem saling mengirimkan pesan dan saling bekerja sama satu sama lain
3. Menguji apakah sistem sudah berfungsi seperti yang seharusnya
4. Dokumentasi sistem untuk keperluan-keperluan tertentu di masa yang akan datang.



2.2.8. Jenis-jenis Diagram

2.2.8.1. Use Case Diagram

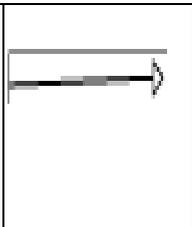
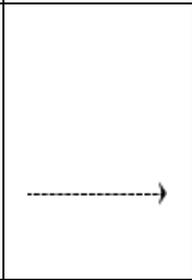
Menurut Indrajani (2015:45), *Usecase diagram* merupakan suatu diagram yang berisi *usecase*, *actor* serta *relationship*, yang merupakan titik awal dalam menganalisis kebutuhan sistem pada saat perancangan.

Widodo (2011:10), *Usecase diagram* bersifat statis yang memperlihatkan himpunan *usecase* dan *actor* untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan dan diharapkan pengguna serta antara analis dan *client*, sedangkan *usecase* merupakan konstruksi untuk mendeskripsikan bagaimana sistem akan terlihat di mata *user* dan *actor* mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. Sebuah *actor* mungkin hanya memberikan informasi inputan pada sistem, hanya menerima informasi dari sistem atau keduanya menerima, dan memberi informasi pada sistem. Simbol-simbol *Usecase Diagram* dapat dilihat pada tabel 2.2.

Tabel 2.2 Simbol-Simbol *Usecase Diagram*

NO	GAMBAR	Nama	KETERANGAN
1.		<i>Actor</i>	<i>Actor</i> menggambarkan pengguna dari sistem, dapat berupa manusia atau sistem terotomatisasi lain yang berinteraksi dengan sistem lain untuk berbagi, mengirim dan menerima informasi.
2.		<i>Use case</i>	Interaksi antara <i>actor</i> dengan <i>software</i>
3.		<i>Association</i>	Hubungan antara <i>actor</i> dan <i>usecase</i>

Lanjutan **Tabel 2.2** Simbol-Simbol *Usecase Diagram*

4		<i>Generalization</i>	Hubungan generaliasasi antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
5		Relasi <i>include</i>	Relasi <i>use case</i> tambahan ke sebuah <i>usecase</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini

(Sumber: Indrajani, 2015:46)

2.2.8.2. Class Diagram

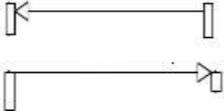
Class diagram menggambarkan struktur sistem dari segi pendefisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut dengan atribut dan metode atau operasi (Rosa A.S & M. Shalahudin,2014:141).

Diagram ini digunakan untuk menggambarkan perbedaan yang mendasar antara *class-class*, hubungan antar-*class*, di mana sub-sistem *class* tersebut. Pada *class diagram* terdapat nama *class*, *attributes*, *operations*, serta *association* (hubungan antar-*class*) (Indrajani, 2015: 49).

2.2.8.3. Sequence Diagram

Merupakan suatu diagram interaksi yang menggambarkan bagaimana objek-objek berpartisipasi dalam bagian interaksi (*particular interaction*) dan pesan yang ditukar dalam urutan waktu. *Sequence* diagram dapat digambarkan dalam beberapa level secara detail dan untuk tujuan yang berbeda pada beberapa langkah yang dikembangkan secara *lifecycle* (Indrajani, 2015: 50).,Berikut merupakan simbol yang digunakan dalam *squence diagram* yang dapat dilihat pada tabel 2.3

Tabel 2.3 Simbol-simbol *sequence diagram*

No	Simbol	Deskripsi
1.		Objek <i>entity</i> , yang saling berinteraksi
2.		Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3.		<i>Activation</i> dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah <i>lifeline</i> . Mengindikasikan sebuah obyek yang akan melakukan sebuah aksi.
4.		Actor menggambarkan actor yang terlibat dalam system

(Sumber: Indrajani 2015:51)

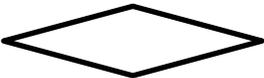
2.2.8.4 Activity Diagram

Digunakan untuk menganalisis *behavior* dengan *use case* yang lebih kompleks dan menunjukkan interaksi-interaksi di antara mereka satu sama lain. *Activity diagram* sebenarnya memiliki kesamaan dengan *statechart diagram* dalam hal menggambarkan aliran data pada model bisnis, tetapi *activity diagram* biasanya digunakan untuk menggambarkan aktivitas bisnis yang lebih kompleks, di mana digambarkan hubungan antar satu *use case* dengan *use case* lainnya (Indrajani, 2015: 46).

. Ada juga yang mendefinisikan Rosa A.S & M.Shalahudin, (2014:161) *Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. *Activity diagram* menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Berikut adalah simbol-simbol yang ada pada *Activity diagram*:

Tabel 2.4 Simbol-simbol *Activity diagram*

No	Simbol	Deskripsi
1		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
2		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3		Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
4		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
5		Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Sumber : Rosa A.S & M. Shalahuddin, 2014:162)

2.3 Teori Program

2.3.1 Android

Android adalah sebuah sistem operasi seperangkat mobile berbasis linux yang mencakup sistem operasi, *middleware*, dan aplikasi. Sumber :

(Nazaruddin, 2011 dalam Aphrizal 2017:51).

2.3.2 WEB

Web adalah suatu sistem yang berkaitan dengan dokumen yang digunakan sebagai media untuk menampilkan teks, gambar, multimedia dan lainnya pada jaringan internet Menurut jurnal Ahmidi, dan Hermawan (2013:69),. sedangkan menurut (Putra, 2009 dalam Aphrizal 2017:51) Perkembangan internet tidak terlepas dari Web atau *word wide web* disingkat Web.



2.3.3 PHP

Hypertext Preprocessor (PHP) merupakan perangkat lunak open source yang diatur dalam aturan *General Purpose Licences* (GPL). Pemrograman PHP sangat cocok dikembangkan dalam lingkungan web, karena PHP bisa dilekatkan pada *script* HTML atau sebaliknya. PHP dikhususkan untuk pengembangan web dinamis. Maksudnya, PHP mampu menghasilkan website yang secara terus menerus hasilnya bisa berubah-ubah sesuai dengan pola yang diberikan, dan bergantung pada permintaan client browser.

2.3.4 JAVA

Menurut Heriyanto (2007:3) Java adalah bahasa yang dapat dijalankan disembarang platform, di beragam lingkungan internet, consumer electronic products, dan computer applications.

Menurut Abdul Kadir (2011:3) Java merupakan pemrograman yang bersifat lintas- platform. Artinya, bahasa ini dapat dipakasi untuk menyusun program pada berbagai sistem operasi (misalnya Linux, Windows, dan UNIX).

Jadi berdasarkan pengertian diatas Java merupakan pemrograman yang dapat dijalankan multi platform (bisa berjalan di berbagai macam sistem operasi) yang mendukung Java dan menjalankan perangkat lunak ataupun jaringan.

2.3.5 Pengertian *Adobe Dreamweaver*

Wahana komputer (2012:1), “*Dreamweaver* adalah salah satu aplikasi editor *web* yang digunakan untuk Merancang secara visual yang dapat memudahkan *user* dalam membantu suatu *website*”.

Elcom (2013:1), “*Dreamweaver* adalah sebuah editor profesional yang menggunakan HTML untuk mendesain *web* secara visual dan mengelola situs halaman *web*”.

Jadi, berdasarkan pendapat dari beberapa ahli di atas, dapat disimpulkan bahwa *Dreamweaver* merupakan *software* editor profesional untuk mengembangkan suatu situs *web*.



2.4 Referensi Jurnal

Dalam Tugas Akhir ini penulis menggunakan beberapa perbandingan jurnal. Dibawah ini akan diuraikan perbandingan dari referensi jurnal tersebut.

Tabel 2.5 Referensi Jurnal

No	Judul /Penulis/Tahun	Masalah	Teori	Metode Penelitian	Hasil
1	Analisis Penentuan Zonasi Risiko Bencana Tanah Longsor Berbasis sistem Informasi Geografis. Penulis : Dhuha Ginanjar, Arief Liala dan abdi Sukmono. Tahun : 2016	Bagaimana analisis perbandingan risiko bencana tanah longsor di kabupaten banjar negara menggunakan metode AHP dan SNI	Kerentanan bencana Awoton (1997). Kapasitas bencana oleh Anderson dan woodrow 1989 dalam saputra (2015)	Metode SNI (standar Nasional Indonesia) AHP (Analytical Hierarchy Proses)	Dari hasil pemetaan risiko bencana tanah longsor metode SNI diperoleh daerah risiko tinggi sebesar 69,961%, sedang 25,868%, dan rendah 4,171%. Sedangkan hasil metode AHP diperoleh daerah risiko tinggi sebesar 73,244%, sedang 23,592%, dan rendah 3,165% yang tersebar di Kabupaten Banjarnegara.



2	Pemetaan Aplikasi Sistem Presensi Siswa Berbasis Android pada SMP N 2 Tinambung. Penulis Aphrizal Tahun : 2017	Bagaimana orang tua wali atau wali siswa bisa mengontrol kehadiran atau keberadaan anaknya pada saat jam belajar.	Android Ambon,Nazarudin (2011). WEB Beta,sidik (2014)	Rational Unified Proseses (RUP)	Sistem presensi siswa berbasis android dilakukan untuk meningkatkan kedisiplinan terhadap siswa.
3	Sistem Informasi Geografis Bencana Alam Banjir Jakarta Selatan. Penulis : Sri Melati Sagita Tahun : 2016	Bagaimana cara mengetahui titik-titik banjir yang berada dijakarta selatan, sehingga masyarakat dapat menghadiri lokasi tersebut sebelum melintasinya.	Pengertian Sistem Veza (2016)	Menggunakan perancangan Unified Modeling Lnuage (UML)	Mendapatkan informasi bencana banjir dijakarta selatan yang di sajukan dalam bentuk peta
4	PEMAHAMAN TENTANG KAWASAN RAWAN BENCANA DAN TINJAUAN	Kejadian alam yang mendatangkan bencana ada yang bisa diduga ada yg tidak, Kondisi-	Peraturan menteri pekerjaan Umum No.21,22,23 dan 24 tahun 2007.		Penata ruang dan Migitasi Bencana yang dapat meminimalisasi dampak bencana karena premis penata



	<p>TERHADAP KEBIJAKAN DAN PERATURAN TERKAIT Penulis : Linda Tondobala Tahun : 2011</p>	<p>kondisi yang menyebabkan diperlukannya sumber data yang baru dan akurat terkait kondisi kebencanaan di Pulau Sulawesi, guna penyelenggaraan penataan ruang wilayah nasional khususnya wilayah pulau sulawesi yang dilakukan secara komprehensif, holistik, terkoordinasi, terpadu, efektif, dan efisien dengan berbasiskan mitigasi bencana.</p>			<p>ruang adalah keseimbangan lingkungan hidup.</p>
--	--	---	--	--	--



5	<p>Pengembangan Sistem Informasi Geografis pada platform google untuk penanggulangan kebakaran di Jakarta Selatan.</p> <p>Penulis: Edy Irwansyah, Sena Adhi Nugraha, Tri Datar Wijaya.</p> <p>Tahun : 2016</p>	<p>Dengan adanya kemungkinan hambatan khususnya bencana kebakaran maka suku dinas pemadam kebakaran Jakarta Selatan perlu adanya sistem yang dapat memantau penanganan bencana kebakaran yang berbasis ruang.</p>	<p>Sistem Informasi Geografi (Arronof, 1889)</p>	<p>Metode yang digunakan adalah Waterfall</p>	<p>Pemanfaatan aplikasi dari sistem SIG khususnya yang berbasis web berguna untuk pemadam kebakaran sebagai alat bantu untuk menanggulangi bahaya kebakaran khususnya Jakarta Selatan.</p>
---	--	---	--	---	--