

BAB II

TINJAUAN PUSTAKA

2.1 Sistem

Sistem adalah rangkaian dari dua atau lebih komponen-komponen yang saling berhubungan, yang berinteraksi untuk mencapai suatu tujuan. Sebagian besar sistem terdiri dari subsistem yang lebih kecil yang mendukung sistem yang lebih besar (Romney, Marshall B dan Steinbart, 2015).

Sistem adalah suatu jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan pokok perusahaan (Mulyadi, 2016).

Berdasarkan pengertian diatas dapat disimpulkan bahwa sistem adalah kumpulan dari komponen-komponen yang saling berkaitan satu dengan yang lain untuk mencapai tujuan dalam melaksanakan suatu kegiatan pokok perusahaan.

2.2 Pengaduan Masyarakat

Pengaduan masyarakat adalah informasi atau pemberitahuan yang disampaikan oleh masyarakat, baik perseorangan dan atau keluarga yang berasal dari pegawai di lingkungan pemerintahan dan atau masyarakat umum yang berisi keluhan dan atau ketidakpuasan terkait dengan perilaku dan atau pelaksanaan tugas dan fungsi anggota pemerintah, yang dilakukan pegawai pemerintahan, dan atau informasi tentang dugaan pelanggaran kode etik atau disiplin pegawai yang dilakukan oleh pegawai pemerintahan (Hapasari, 2019).

2.3 Orang Hilang

Definisi orang hilang menurut Kamus Besar Bahasa Indonesia adalah sebagai berikut :

1. Tidak ada lagi, lenyap, tidak kelihatan. Seseorang itu tidak ada lagi tanpa sebab jelas atau tidak terlihat lagi dari pandangan.
2. Tidak dikenang lagi, tidak ^{diingat} diingat lagi, lenyap. Seseorang yang mempunyai ketenaran tetapi tidak diketahui ⁵ nya begitu saja karena suatu kejadian.

3. Tidak ada, tidak kedengaran lagi. Seseorang sering memperlihatkan dirinya di lingkungan sekitar, suatu saat tidak ada lagi terdengar kabarnya dikarenakan sesuatu hal terjadi padanya, sehingga orang disekitarnya yang sering mendengar pembicaraanya tidak lagi mendengarnya.

Kitab Undang-undang Hukum Perdata (Staatsblad 1847 Nomor 23, Burgerlijk Wetboek Voor Indonesie (BW)) di pasal 467-471 telah mencantumkan ketentuan mengenai mafqud/orang hilang. KUH perdata tidak menggunakan istilah mafqud, akan tetapi menggunakan istilah “Orang yang duperkirakan telah meninggal dunia” (Joel C, Hana S, dan Anindya H, 2015).

2.4 Android

Android adalah sebuah sistem operasi untuk Android adalah sistem operasi yang berbasis linux untuk telepon seluler seperti telepon pintar dan komputer tablet. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti lunak (Hadi Muhammad D dan Rangga Gelar Guntara, 2018)

2.5 MapBox

Mapbox adalah salah satu penyedia peta kustom terbesar di situs-situsternama seperti Foursquare, Pinterest, Evernote sejak pada tahun 2010, Mapboxmemperbanyak pilihan peta kustomnya untuk mengisi keterbatasan yang dimilikipenyedia peta seperti google maps. Mapbox merupakan pencipta atau contributorsejumlah pustaka dan aplikasi peta bebas terkenal, misalnya spesifikasi MBTiles,Kartografi TileMill IDE, pustakka Java Script Leaflet, Bahasa gaya dan parser peta CartoCSS, dan pustaka Java Script Mapbox.js. Data Mapbox diambil dari sumber sumber data terbuka seperti Open Street Map dan NAS, dan sumber-sumber databerbayar seperti Digital Globe. Teknologinya dibangun menggunakan Node.Js,CouchDB, Mapnik, GDAL dan Leafletjs. MapBox, yaitu dukungan terhadapGeoJSON – encoding obyek geografis dalam bentuk JSON. Mapbox jugamerupakan turunan dari Open Street Map (Suryo Atmojo, 2018).

2.6 Firebase

Firebase adalah penyedia layanan cloud dengan back-end sebagai servisnya yang berbasis di San Fransisco, California. Firebase membuat sejumlah produk untuk pengembangan aplikasi mobile ataupun web. Firebase didirikan oleh Andrew Lee dan James (Anisya S dan Rizki Fitrah

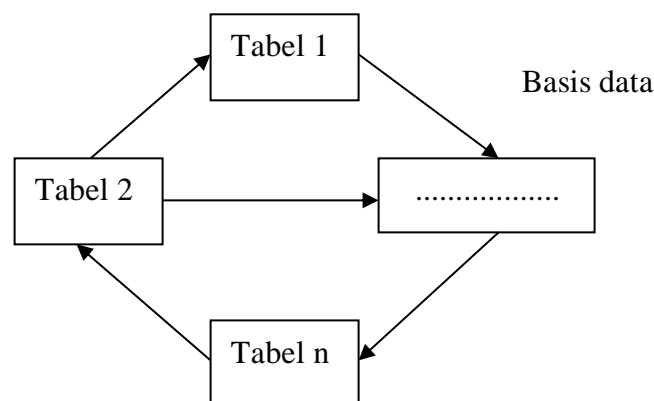
F, 2018).

Firestore Realtime Database memungkinkan anda untuk membuat aplikasi kolaboratif dan kaya fitur dengan menyediakan akses yang aman ke database, langsung dari kode sisi klien. Realtime Database adalah database NoSQL yang di hosting di cloud. Data disimpan di drive lokal, bahkan saat offline sekalipun, peristiwa realtime terus berlangsung, sehingga pengguna akhir akan merasakan pengalaman yang responsif. Ketika koneksi perangkat pulih kembali, realtime database akan menyingkronkan perubahan data lokal dengan update jarak jauh yang terjadi selama klien offline, sehingga perbedaan akan otomatis digabungkan.

Cloud Firestore adalah database NoSQL yang dihosting di cloud dan dapat diakses langsung melalui SDK asli oleh iOS, Android dan aplikasi web. Selain REST dan RPC API, Cloud Firestore juga tersedia Node.js, Python, dan Go SDK yang asli (Firestore, 2020).

2.7 Basis Data

Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat. Pada buku ini menggunakan basis data relasional yang diimplementasikan dengan tabel-tabel yang saling memiliki relasi seperti pada gambar berikut



Gambar 2.1 Ilustrasi Basis Data

Sistem informasi tidak dapat dipisahkan dengan kebutuhan basis data apapun bentuknya, entah berupa file teks ataupun DBMS.

Kebutuhan basis data dalam sistem informasi meliputi :

- Memasukkan, menyimpan dan mengambil data.
- Membuat laporan berdasarkan data yang telah disimpan.

Tujuan dari dibuatnya tabel-tabel disini adalah untuk menyimpan data ke dalam tabel-tabel agar mudah diakses. Oleh karena itu, untuk merancang tabel-tabel yang akan dibuat maka dibutuhkan pola pikir penyimpanan data nantinya jika dalam bentuk baris-baris data (record) dimana setiap baris terdiri dari beberapa kolom (Rosa dan Shalahuddin, 2019).


2.8 Unified Modelling Language (UML)



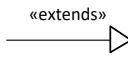



UML ada karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan, jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya uml paling banyak digunakan pada metodologi berorientasi objek (Rosa dan Shalahuddin, 2019).

2.8.1 Use Case Diagram

Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Syarat penamaan pada use case adalah nama didefinisikan sesimpel mungkin dan dapat dipahami (Rosa dan Shalahuddin, 2019). Adapun simbol-simbol sebagai berikut :

Tabel 2.1 Simbol *Use Case*

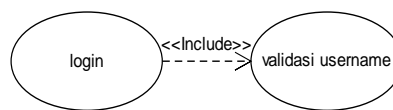
Keterangan	Simbol	Deskripsi
<i>Use Case</i>		Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal-awal frase nama <i>use case</i>

<p>Aktor</p> 	<p>Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar itu sendiri. Jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama aktor.</p>
<p>Asosiasi</p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.</p>
<p>Ekstensi</p> 	<p>Relasi use case tambahan ke sebuah <i>use case</i>, dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan use case yang ditambahkan, misal</p>  <p>Arah panah mengarah pada <i>use case</i> yang ditambahkan.</p>
<p>Generalisasi</p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya :</p> 

Arah panah mengarah pada *use case* yang menjadi generalisasinya (umum).

Ada dua sudut pandang yang cukup besar mengenai *include* di *use case* :

a. *Include* berarti *use case* yang ditambahkan akan selalu dipanggil saat *use case* tambahan dijalankan, misal pada kasus berikut :

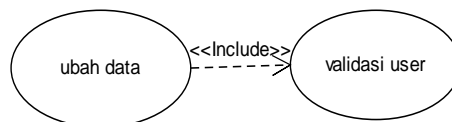


Menggunakan
an/include/u
ses

<<Include>>
----->

«uses»
—————>

b. *Include* berarti *use case* yang tambahan akan selalu melakukan pengecekan apakah *use case* yang ditambahkan telah dijalankan sebelum *use case* tambahan dijalankan, misal pada kasus berikut :



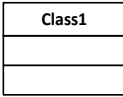
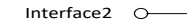



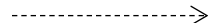
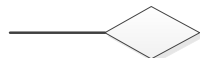
Ke dua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi,

2.8.2 Class Diagram

Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki suatu kelas,

sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas (Rosa dan Shalahuddin, 2019).

Tabel 2.2. Simbol *Class Diagram*

Simbol	Deskripsi
<p>Kelas</p> 	Kelas pada struktur sistem.
<p>Natarmuka/<i>interface</i></p> 	Sama dengan konsep <i>interface</i> dalam pemograman berorientasi objek.
<p>Asosiasi</p> 	Relasi antar kelas dalam makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
<p>Asosiasi berarah</p> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus).
<p>Kebergantungan</p> 	Relasi antar kelas dengan makna kebergantungan antar kelas.
<p>Agregasi</p> 	Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>).

2.8.3 Activity Diagram

Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Shalahuddin, 2019). Simbol-simbol yang terdapat pada activity diagram adalah seperti pada Tabel 2.2.

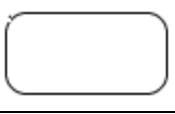


Tabel 2.3 Simbol *ActivityDiagram*

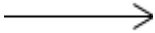


Keterangan	Simbol	Deskripsi
Status awal		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas	aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan		Asosiasi percabangan dimanajika ada pilihan aktivitas lebih darisatu.
Penggabungan		Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Swimlane		Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
Status akhir		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

2.8.4 Sequence Diagram

Sequence diagram secara grafis menggambarkan bagaimana objek berinteraksi antara satu sama lain melalui pesan pada sebuah use case atau operasi. (Rosa dan Shalahuddin, 2019)

Tabel 2.4 Simbol *Sequence Diagram*




GAMBAR	NAMA	KETERANGAN
	<i>State</i>	Nilai atribut dan nilai link pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
	<i>InitialPseudo State</i>	Bagaimana objek dibentuk atau diawali
	<i>FinalState</i>	Bagaimana objek dibentuk dan dihancurkan




	<i>Transition</i>	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	<i>Node</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

2.8.5 Diagram Alir (Flowmap Diagram)

Bagan alir adalah bagan yang menunjukkan aliran di dalam program atau prosedur sistem secara logika. Flowmap ini berfungsi untuk memodelkan masukan, keluaran, proses dan transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan flowmap ini harus dapat memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi (Yeldi Latif, 2016).

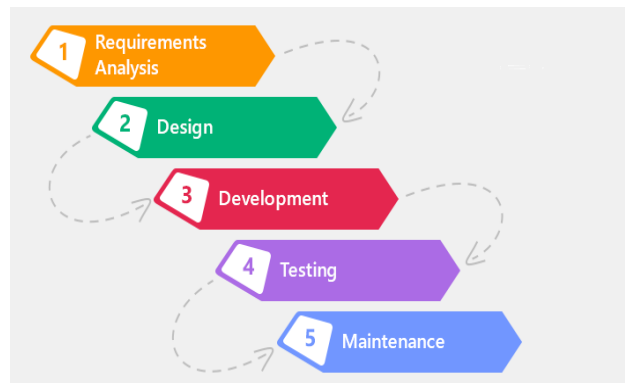
Tabel 2.5 Simbol *Flowmap Diagram*

Simbol	Nama	Keterangan
	Terminator awal/akhir	Simbol untuk memulai atau mengakhiri suatu program
	Dokumen	Menunjukkan dokumen berupa dokumen input dan output pada proses manual dan proses berbasis komputer
	Proses manual	Menunjukkan kegiatan proses yang dilakukan secara manual

	Proses Komputer	Menunjukkan kegiatan proses yang dilakukan secara komputerisasi
	Arah aliran data	Menunjukkan arah aliran dokumen antar bagian yang terkait pada suatu sistem
	Penyimpanan manual	Menunjukkan media penyimpanan data / informasi secara manual

2.9 Metode Pengembangan Sistem

Pada metode penelitian ini dilakukan rekayasa perangkat lunak yang digunakan adalah model *Waterfall* seperti pada gambar berikut ini:



Gambar 2.2 Metode Pengembangan Model *Waterfall*

Keterangan:

1. Requirement Analisis

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi ini biasanya dapat diperoleh melalui wawancara, diskusi atau survei langsung.

2. System Design

Spesifikasi kebutuhan dari tahap sebelumnya akan dipelajari dalam fase ini dan desain sistem disiapkan. Desain Sistem membantu dalam menentukan perangkat keras(hardware) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

3. Development

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut unit, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai unit testing.

4. Integration & Testing

Seluruh unit yang dikembangkan dalam tahap implementasi diintegrasikan ke dalam sistem setelah pengujian yang dilakukan masing-masing unit. Setelah integrasi seluruh sistem diuji untuk mengecek setiap kegagalan maupun kesalahan.

5. Operation & Maintenance

Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Perbaikan implementasi unit sistem dan peningkatan jasa sistem sebagai kebutuhan baru.