

BAB II

LANDASAN TEORI

2.1 Penelitian Sebelumnya

Penelitian yang berkaitan dengan aplikasi pencarian tempat wisata pernah dilakukan sebelumnya, diantaranya Penelitian dengan judul “Sistem Informasi Obyek Wisata (*Tour Guide*) Secara *Real Time* Menggunakan GPS di Bogor Via Mobile Berbasis *Android*” oleh Bondan Dwi Hatmoko (Hatmoko, 2014). Aplikasi ini dirancang untuk membantu para wisatawan agar lebih mudah untuk menemukan beberapa lokasi wisata di Bogor, rute ke lokasi wisata dari lokasi pengguna dan juga galeri foto tempat wisata beserta informasi obyek wisata tersebut. Dengan rancangan Sistem Aplikasi Informasi Obyek Wisata (*Tour Guide*) Secara *Real Time* menggunakan GPS di Bogor pada platform Android, dapat membantu para wisatawan untuk mendapatkan informasi wisata secara efektif dan efisien. Kelebihan sistem yang dirancang adalah memberikan informasi yang dibutuhkan secara tepat dan akurat dengan didukung oleh data visual yang ada seperti foto tempat wisata dan informasinya. Penelitian ini menggunakan metode research and development (R&D), karena penelitian ini merupakan pengembangan dan membuat prototype baru.

Penelitian lain yang pernah dilakukan, yaitu Penelitian oleh Wisnu Prasetya Utama (Utama, 2015) dengan judul “Aplikasi Sebaran Objek Wisata di Kota Bengkulu Berbasis *Android*” Penelitian ini bertujuan untuk merancang dan membangun suatu aplikasi sebaran objek wisata di kota Bengkulu berbasis android dengan metode analisis tetangga terdekat (*nearest neighbour analysis*). Aplikasi ini juga memanfaatkan peta dari *google maps* dalam menunjukkan lokasi pengguna ke lokasi objek wisata di Kota Bengkulu yang dicari. Aplikasi ini dibangun dengan menggunakan bahasa pemrograman *Java for Android* dengan IDE Eclipse ADT (*Android Developer Tools*) dan *JDK7*. Metode pengembangan sistem yang digunakan

untuk membangun aplikasi ini adalah model *sekuensial linier* dan *Unified Modeling Language* (UML) sebagai perancangan sistem.

Penelitian berikutnya, yaitu Penelitian yang dilakukan oleh Erik Kurniadi (Erik Kurniadi, 2018) Yang Berjudul “Rancang Bangun Aplikasi Wisata Kabupaten Kuningan Berbasis Android Menggunakan Metode *Location Based Service* (LBS)” Penelitian Rancang Bangun Aplikasi Wisata ini dilakukan bertujuan untuk membuat suatu aplikasi wisata yang beroperasi pada platform android dan berbasis lokasi sehingga mampu memberikan informasi dan rute menuju tempat-tempat wisata yang ada di wilayah Kabupaten Kuningan. Penelitian ini menawarkan sebuah aplikasi android yang dapat memberikan kemudahan kepada wisatawan yang ingin mengunjungi obyek wisata. Aplikasi ini menyediakan informasi mengenai obyek wisata yang ada dari mulai informasi lokasi, fasilitas, jam buka dan tutup serta panduan untuk mencapai lokasi dengan menggunakan metode *Location Based Service*. Setiap wisatawan dapat mendownload aplikasi dan menjalankannya di smartphone. Pengguna aplikasi dapat memilih obyek wisata berdasarkan kategori dan sistem akan menyediakan fasilitas panduan menuju lokasi wisata yang ingin dikunjungi. Aplikasi ini diharapkan mampu menjadi sebuah solusi permasalahan diatas dan menjadi media promosi wisata di kuningan.

Penelitian terkait selanjutnya oleh Dhiya Ulhaq (Ulhaq & Alamsyah, 2017) yang mengangkat judul “Implementasi Web Service Dalam Pencarian Objek Wisata Berbasis Android”. Dengan hasil penelitian yang diharapkan Aplikasi ini dibuat untuk membantu wisatawan untuk mencari informasi wisata, mencari objek wisata, dan melihat data wisata. User juga dapat memberi review terhadap suatu objek wisata berupa rating, menambahkan lokasi dan komentar sehingga dapat membantu mempromosikan tempat wisata daerah tersebut. Tujuan penelitian ini adalah untuk mengimplementasikan web service dalam aplikasi mobile, dan juga arsitektur Rest, sehingga dapat digunakan dalam banyak client baik versi web maupun dekstop, sehingga mempermudah pengolahan data dan akses kebeberapa

aplikasi dan user dalam mengakses tempat wisata baik berupa informasi wisata maupun jarak wisata tersebut. tahapan penelitian yang dilakukan ini Pembuatan dan implementasi web service dalam pencarian objek wisata berbasis android. perancangan sistem yang digunakan adalah UML (Use Case, Activity dan Class Diagram),

Perbandingan penelitian dapat dilihat pada tabel berikut.

Tabel 2.1 Perbandingan Penelitian

Nama Peneliti / Tahun	Judul	Metode	Hasil
(Hatmoko, 2014)	Sistem Informasi Obyek Wisata (<i>Tour Guide</i>) Secara <i>Real Time</i> Menggunakan <i>Gps</i> Di Bogor Via Mobile Berbasis <i>Android</i>	<i>research and development</i>	Dengan rancangan Aplikasi Informasi Obyek Wisata (<i>Tour Guide</i>) Secara <i>Real Time</i> Menggunakan <i>GPS</i> di Bogor pada platform android,Membantu para wisatawan untuk mendapatkan informasi wisata secara efektif dan efisien.
(Utama, 2015)	Aplikasi Sebaran Objek Wisata Di Kota Bengkulu Berbasis Android	<i>sekuensial linier</i>	menghasilkan aplikasi yang memberikan informasi mengenai objek wisata Bengkulu berbasis

			android yang dapat digunakan oleh pengguna untuk pencarian lokasi objek wisata di Kota Bengkulu dengan memanfaatkan GPS dan Google Maps.
Erik Kurniadi,(2018)	Rancang bangun aplikasi wisata kabupaten kuningan Berbasis android menggunakan metode <i>location based service</i>	<i>LBS</i>	Dengan adanya aplikasi wisata ini diharapkan pengguna dapat terbantu untuk menemukan tujuan wisata dan informasi lainnya yang diperlukan pengguna dengan pemanfaatan teknologi <i>smartphone</i> , internet,dan LBS.
(Ulhaq & Alamsyah, 2017)	Implementasi Web Service Dalam Pencarian Objek Wisata Berbasis Android	<i>UML</i>	Aplikasi ini Dapat membantu wisatawan untuk mencari informasi wisata, mencari objek wisata, dan melihat data wisata.

			User juga dapat memberi review terhadap suatu objek wisata berupa rating, menambahkan lokasi dan komentar sehingga dapat membantu mempromosikan tempat wisata daerah tersebut.
--	--	--	--

Dari beberapa perbandingan penelitian dapat disimpulkan bahwa Penelitian yang salah satunya berkaitan dengan Pencarian tempat wisata yang paling mendekati adalah judul “Aplikasi Sebaran Objek Wisata di Kota Bengkulu Berbasis *Android*” yang ditulis oleh Wisnu Prasetya Utama (Utama, 2015) yang menggunakan metode *sekuensial linier* sebagai analisis sistem yang memiliki alur pengerjaan yang hampir sama walau dengan metode yang berbeda. Banyak rancangan aplikasi yang hampir menyerupai sehubungan tentang pencarian wisata walaupun tidak sedikit beberapa komponen yang tentunya sedikit berbeda baik dalam segi batasan masalah maupun manfaat penelitian. Sama halnya dengan penelitian terkait seperti penelitian yang dilakukan oleh Dhiya Ulhaq (Ulhaq & Alamsyah, 2017) yang melakukan “Implementasi *Web Service* Dalam Pencarian Objek Wisata Berbasis *Android*”. Metode perancangan system yang digunakan menggunakan tahap-tahap SDLC (System Development Life Cycle) meliputi analisis, disain dan implementasi. Dari proses pembuatan sistem ini, Tujuan penelitian ini adalah untuk mengimplementasikan web service dalam aplikasi mobile dan juga arsitektur Rest, sehingga dapat digunakan dalam banyak client baik versi web maupun dekstop, sehingga mempermudah pengolahan

data dan akses ke beberapa aplikasi dan user dalam mengakses tempat wisata baik berupa informasi wisata maupun jarak wisata tersebut. Metode yang digunakan untuk membangun aplikasi ini adalah *Unified Modeling Language* (UML) sebagai perancangan sistem

Pada penelitian dengan judul Aplikasi Pencarian Tempat Wisata di Kabupaten Flores Timur Berbasis Android (Studi Kasus : Kabupaten, Flores Timur), ini menghasilkan sebuah aplikasi yang mampu menyediakan informasi tentang tempat wisata alam, kuliner, religi, penginapan dan transportasi di Kabupaten Flores Timur. Aplikasi ini memberikan kemudahan bagi wisatawan lokal maupun mancanegara khususnya dalam mengakses informasi tempat wisata alam, kuliner, religi, penginapan dan transportasi. Aplikasi ini dibuat dengan menggunakan bahasa pemrograman *java*. Metode pengembangan aplikasi yang digunakan dalam penelitian ini adalah *Unified Modeling Language*.

2.2 Gambaran Umum Kabupaten Flores Timur

Kabupaten Flores Timur terbentuk pada tanggal 20 desember 1958 bersamaan dengan tepatnya UU.NO 69 / 1958.tentang pembentukan daerah tingkat II dalam wilayah daerah tingkat I Bali, NTB, DAN NTT

Kabupaten Flores Timur sebuah Kabupaten yang berada di pulau flores. Di dalam wilayah propinsi NTT (Nusa Tenggara Timur). Kota Larantuka adalah ibukota Kabupaten Flores Timur. Berdasarkan namanya sudah dapat di bayangkan kalau Kabupaten Flores Timur ini terletak di bagian timur dari pulau flores.

Kabupaten Flores Timur ini terdiri atas beberapa kepulauan. Dengan pulau terbesarnya adalah pulau Solor dan pulau Adonara. Batasan wilayah Kabupaten Flores Timur adalah batas sebelah utara flores adalah dengan laut flores, batas sebelah timur adalah dengan kabupaten Lembata, batas sebelah selatan adalah dengan laut sawu, batas sebelah barat adalah dengan kabupaten sikka.

Kabupaten Flores Timur terkait dengan adat-istiadat serta tata cara yang telah menjadi tradisi dari generasi ke generasi yang menjadikan sebagai masyarakat yang sangat menghargai etnis, budaya, agama, dan ras.

2.3 Pengertian Aplikasi

Aplikasi Perangkat lunak aplikasi adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Contoh utama perangkat lunak aplikasi adalah pengolah kata, lembar kerja, dan pemutar media. Beberapa aplikasi yang digabung bersama menjadi suatu paket kadang disebut sebagai suatu paket atau suite aplikasi (*application suite*). Contohnya adalah *Microsoft Office* dan *Open Office.org*, yang menggabungkan suatu aplikasi pengolah kata, lembar kerja, serta beberapa aplikasi lainnya. Aplikasi-aplikasi dalam suatu paket biasanya memiliki antarmuka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan setiap aplikasi. Sering kali, aplikasi ini memiliki kemampuan untuk saling berinteraksi satu sama lain sehingga menguntungkan pengguna. Contohnya, suatu lembar kerja dapat dibenamkan dalam suatu dokumen pengolah kata walaupun dibuat pada aplikasi lembar kerja yang terpisah. (Muttaqin, Permana, & Salisah, 2017)

2.4 Penngertian Tempat Wisata

Tempat Wisata adalah suatu perjalanan yang dilakukan untuk rekreasi atau liburan dan juga persiapan yang dilakukan untuk kegiatan ini. Seorang wisatawan atau turis adalah seseorang yang melakukan perjalanan paling tidak sejauh 80 km (50 mil) dari rumahnya dengan tujuan rekreasi, adalah pengertian definisi dari Organisasi Pariwisata Dunia.

Menurut Undang Undang No. 10/2009 tentang Kepariwisata, yang dimaksud dengan pariwisata adalah berbagai macam kegiatan wisata yang didukung oleh berbagai fasilitas serta layanan yang disediakan masyarakat, pengusaha, Pemerintah dan Pemerintah Daerah. (Wahab, 1985).)

2.5 Pengertian Android

Android adalah sebuah sistem operasi untuk perangkat *mobile* yang berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang buat menciptakan aplikasi mereka. Awalnya, Google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat peranti lunak untuk *smartphone*. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia

Android merupakan generasi baru *platform mobile* dan merupakan *platform* yang memberikan pengembang untuk melakukan sesuai dengan yang diharapkannya (Safaat,2012)

2.6 Google Map API

Google Maps API adalah API yang paling populer di internet. Pencatatan yang dilakukan pada bulan Mei tahun 2010 ini menyatakan bahwa 43% *mashup* (aplikasi dan situs web yang menggabungkan dua atau lebih sumber data) menggunakan Google Maps API. Beberapa tujuan dari penggunaan Google Maps API adalah untuk melihat lokasi, mencari alamat, mendapatkan petunjuk mengemudi dan lain sebagainya.(Maldzhanski, 2015)

2.7 Pengertian PHP (*PHP Hypertext Preprocessor*)

Menurut (Hari Sugiarto, 2016) “PHP adalah pemrograman interpreter yaitu proses penerjemahan baris kode sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan”. PHP merupakan singkatan dari *Hypertext Preprocessor*, dan merupakan bahasa yang disertakan dalam dokumen HTML sekaligus bekerja di sisi *server* (*server-side HTML embedded scripting*)”.

Beberapa kelebihan dari PHP menurut Saputra dan Agustin (2012:4) antara lain:

- a. Mudah dipelajari.
- b. Mampu lintas platform.
- c. *Free* (gratis).bersifat *open source*
- d. PHP memiliki tingkat akses yang cepat
- e. Didukung oleh beberapa macam *web server*.
- f. Mendukung database.

2.8 Pengertian JQuery

Merupakan suatu *library Javascript* yang memungkinkan dalam membuat program *web* pada suatu halaman *web* tanpa harus menambahkan *event* ataupun properti pada halaman *web* tersebut. Dengan JQuery, suatu halaman *web* yang menjadi aplikasi *web*, jika dilihat *source* - nya akan terlihat seperti dokumen html biasa dalam artian tidak ada kode *Javascript* yang terlihat langsung yang biasa disebut dengan *unobstrusive JavaScript programming* (Hari Sugiarto, 2016)

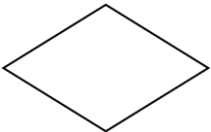
2.9 Sistem Database

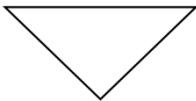
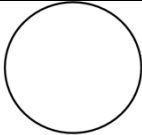
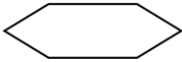
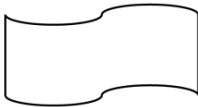
Secara umum, *database* atau basis data berarti koleksi data yang saling terkait. Secara praktis, basis data dapat dianggap sebagai suatu penyusun data yang terstruktur yang disimpan dalam media pengingat (*harddisk*) yang tujuannya adalah agar data tersebut dapat diakses dengan mudah dan cepat. Sesungguhnya ada beberapa jenis *database*, antara lain: *database* hierarkis, *database* jaringan, dan *database* relasional. *Database* relasional merupakan database yang populer saat ini dan telah diterapkan pada berbagai *platform*, dari *PC* hingga mini komputer. Sebuah *database* relasional tersusun atas sejumlah tabel (Kadir, 2008).

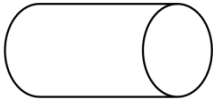
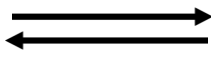
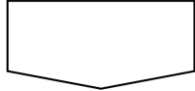
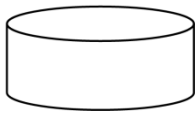
2.10 Diagram Alir (*Flowchart*)

Flowchart sistem dibuat untuk menunjukkan alur kerja dari suatu sistem secara umum. Dengan adanya *flowchart* sistem, dapat mengidentifikasi permasalahan yang ada pada sistem yang lama dan menganalisa kebutuhan dari sistem tersebut sebagai langkah awal dalam rancangan sistem baru yang akan dibentuk. Simbol - simbol dalam *flowchart* dapat dilihat pada tabel 2.2.

Tabel 2.2 Simbol *Flowchart*.

No	Simbol	Keterangan
1		Menyatakan permulaan atau akhir suatu program
2		Menyatakan suatu tindakan (proses) yang dilakukan oleh computer
3		Menyatakan input berasal dari disk atau output disimpan ke disk
4		Memasukkan data secara manual
5		Menujukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban : ya / tidak

6		Menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu
7		Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
8		Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
9		Menyatakan proses input atau output tanpa tergantung jenis peralatannya
10		Mencetak keluaran dalam bentuk dokumen (melalui printer)
11		Menyatakan input berasal dari kartu atau output ditulis ke kartu
12		Menyatakan suatu tindakan (proses) yang tidak dilakukan oleh computer
13		Mencetak keluaran dalam layar monitor
14		I / O yang menggunakan pita kertas berlubang

15		I / O yang menggunakan drum maknetik
16		Menyatakan jalannya arus suatu proses
17		Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda
18		Database

2.11 Unified Modelling Language (UML)

Unified Modelling Language (UML) adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. *Unified Modelling Language* menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan *Unified Modelling Language* kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena *Unified Modelling Language* juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa-bahasa berorientasi objek seperti *C++*, *Java*, *C#* atau *VB.NET*. Walaupun demikian, *Unified Modelling Language* tetap dapat digunakan untuk modeling aplikasi prosedural dalam *VB* atau *C*.

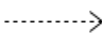
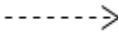
Seperti bahasa-bahasa lainnya, *Unified Modelling Language* mendefinisikan notasi dan *syntax*/semantik. Notasi *Unified Modelling Language* merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu,

dan *Unified Modelling Language syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi *Unified Modelling Language* terutama diturunkan dari tiga notasi yang telah ada sebelumnya: Grady Booch *OOD (Object-Oriented Design)*, Jim Rumbaugh *OMT (Object Modeling Technique)*, dan Ivar Jacobson *OOSE (Object-Oriented Software Engineering)*.

1. *Use Case Diagram*

Use case diagram merupakan gambaran skenario dari interaksi antara pengguna dengan sistem. *Use case diagram* menggambarkan hubungan antara aktor dan kegiatan yang dapat dilakukannya terhadap aplikasi.

Tabel 2.3 Simbol-simbol *Use Case Diagram*

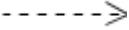
No.	Gambar	Nama	Keterangan
1.		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2.		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3.		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4.		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .

5.		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7.		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8.		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
9.		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10.		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

2. Class Diagram

Class Diagram merupakan diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

Tabel 2.4 Simbol-simbol *Class Diagram*

No.	Gambar	Nama	Keterangan
1.		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2.		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3.		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4.		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
5.		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6.		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

3. *Activity Diagram*

Activity Diagram adalah representasi grafis dari seluruh tahapan alur kerja yang mengandung aktivitas, pilihan tindakan, perulangan dan hasil dari aktivitas tersebut. Diagram ini dapat digunakan untuk menjelaskan proses bisnis dan alur kerja operasional secara langkah demi langkah dari komponen suatu sistem.

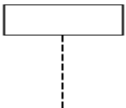
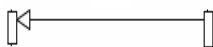
Tabel 2.5 Simbol-simbol *Activity Diagram*

No.	Gambar	Nama	Keterangan
1.		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2.		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
3.		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4.		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan.
5.		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.

4. *Sequence Diagram*

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem berupa *message* terhadap waktu.

Tabel 2.6 Simbol-simbol *Sequence Diagram*

No.	Gambar	Nama	Keterangan
1.		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
3.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-tentang aktifitas yang terjadi.