

TUGAS AKHIR

**PEMBANGUNAN SISTEM APLIKASI PENYEWAAN MOBIL
BERBASIS ANDROID STUDI KASUS di CV ABU SULAIMAN RENT
CAR**

Disusun oleh :

Nama : Arnold Solian Siagian

NIM : 2011-81-086



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS ESA UNGGUL
JAKARTA
2015**

TUGAS AKHIR

**PEMBANGUNAN SISTEM APLIKASI PENYEWAAN MOBIL
BERBASIS ANDROID STUDI KASUS di CV ABU SULAIMAN RENT
CAR**

Diajukan Untuk Memenuhi Persyaratan Kurikulum Sarjana Strata-1 pada
Program Studi Teknik Informatika Fakultas Ilmu Komputer

Disusun oleh :

Nama : Arnold Solian Siagian

NIM : 2011-81-086



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS ESA UNGGUL
JAKARTA
2015**

LEMBAR PENGESAHAN TUGAS AKHIR

Nama : Arnold Solian Siagian
NIM : 2011-81-086
Program Studi : Teknik informatika
Fakultas : Ilmu Komputer
Judul Tugas Akhir : Pembangunan Sistem Aplikasi Penyewaan Mobil Berbasis
Android Studi Kasus di CV Abu Sulaiman Rent Car

Tugas akhir di atas telah disetujui dan diterima sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer, jenjang pendidikan Strata-1 Fakultas Ilmu Komputer, Universitas Esa Unggul.

Jakarta, September 2015

Disetujui Oleh,

Kartini S.Kom, MMSI

Pembimbing 1

Bambang Irawan, S.Kom, M.Kom

Pembimbing 2

Mengetahui,

Bambang Irawan, S.Kom, M.Kom

Ka. Prodi Teknik Informatika

Dr.Ir.Husni S Sastramihardja, M.T

Dekan Fakultas Ilmu Komputer

LEMBAR PENGESAHAN PENGUJI SIDANG

Nama : Arnold Solian Siagian
NIM : 2011-81-086
Program Studi : Teknik Informatika
Fakultas : Ilmu Komputer
Judul Tugas Akhir : Pembangunan Sistem Aplikasi Penyewaan Mobil Berbasis
Android Studi Kasus di CV Abu Sulaiman Rent Car

Tugas akhir ini telah disetujui dan diterima sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer, jenjang pendidikan Strata-1 Fakultas Ilmu Komputer, Universitas Esa Unggul Jakarta, September 2015

Penguji I : Bambang Irawan, S.Kom, M.Kom _____

Penguji II : Drs. Holder Simorangkir, M.Kom _____

Penguji III : Kundang Karsono, S.Kom, M.Kom _____

Mengetahui,

Bambang Irawan, S.Kom, M.Kom

Ka. Prodi Teknik Informatika

LEMBAR PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini :

Nama : Arnold Solian Siagian
NIM : 2011-81-086
Program Studi : Teknik informatika
Fakultas : Ilmu Komputer

Menyatakan bahwa Tugas Akhir ini adalah murni hasil karya saya, apabila saya melihat hasil karya orang lain, maka saya mencantumkan sumbernya dengan ketentuan yang berlaku. Saya bersedia dikenakan sanksi pembatalan Laporan Tugas Akhir ini, apabila terbukti melakukan tindakan plagiat.

Demikian pernyataan ini saya buat dengan sebenarnya.

Arnold Solian Siagian

ABSTRAK

Perkembangan *mobile device* saat ini sangat berkembang pesat, perkembangannya seakan semua kegiatan dapat dilakukan dengan aplikasi *mobile*, begitu juga halnya dalam bidang transportasi dalam hal ini bidang rental mobil, pada proses penyewaan mobil yang lama dilakukan dengan manual terdapat beberapa kendala utama dalam prosesnya seperti pelanggan yang sudah terdaftar atau sudah pernah merental mobil harus datang kembali ke tempat rental mobil atau menelepon apabila akan memesan mobil di mana kurang efektifnya, namun dengan adanya aplikasi rental mobil ini para pelanggan dapat memesan mobil dengan mudah dengan adanya aplikasi *mobile* setiap pelanggan dapat melihat informasi mobil yang akan di sewa apakah tersedia karena informasi mobil dapat dilihat hanya dalam sebuah aplikasi *mobile* dan untuk pengelola juga akan mengefektifkan pekerjaan mereka dengan adanya aplikasi *mobile* android pengelola dapat melakukan pendataan seperti data pelanggan, data mobil, data sopir dan data pelanggan yang sedang melakukan penyewaan hanya dalam sebuah aplikasi *mobile* android dan datanya disimpan dengan komputerisasi dan akan mudah diakses dan proses penyewaan mobil dapat dilakukan dengan *online* sehingga *human error* dalam rental Abu Sulaiman rent car dapat dikurangi, dan berdasarkan analisa PIECES oleh karena itu sistem lama yang berjalan pada Abu Sulaiman Rental Car akan dikembangkan dengan sistem yang baru yaitu secara *online* menggunakan android, pembuatan aplikasi *mobile* ini menggunakan *software* Eclipse dengan pengembangan metode *Extreme Programing*.

Kata kunci : *Mobile, Android, online, penyewaan, rental mobil, Eclipse, analisis PIECES, metode Extreme Programing.*

KATA PENGANTAR

Puji syukur kepada Tuhan Yesus Kristus atas kasih karunia dan penyertaan yang tak pernah habis-habisnya sepanjang waktu dalam hidup saya, sehingga pembuatan tugas akhir yang berjudul “*Pembangunan Sistem Aplikasi Penyewaan Mobil Berbasis Android Studi Kasus di CV Abu Sulaiman Rent Car.*” dapat diselesaikan dengan baik.

Tugas Akhir ini dibuat dalam rangka memenuhi persyaratan dalam menyelesaikan Program Studi Stara-1 jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Esa Unggul.

Dalam proses pembuatan tugas akhir ini tidak lepas dari bimbingan, dukungan ,motivasi serta doa berbagai pihak. Untuk itu, ucapan terima kasih sebesar-besarnya ditujukan kepada :

1. **Bapak Dr. Ir. Arief Kusuma A.P, MBA.** Selaku Rektor Universitas Esa Unggul.
2. **Bapak Dr. Ir. Husni S.Sastramiharja, M.T.** Selaku Dekan Fakultas Ilmu Komputer.
3. **Bapak Bambang Irawan, S.Kom, M.Kom.** Selaku Kepala Program Studi Teknik Informatika dan Dosen Pembimbing II.
4. **Ibu Kartini S.Kom, MMSI.** Selaku pembimbing Dosen Pembimbing I.
5. Bapak Hafizt Selaku Pengelola Abu Sulaiman Rent Car yang telah mengizinkan saya untuk melakukan penelitian.
6. Kedua Orang Tua saya yang senantiasa memberikan dukungan melalui doa dan nasehat-nasehat hingga saya semangat untuk menyelesaikan tugas akhir ini.
7. Bang Jonatan, Kak Melita, Bang boy, Bang Alimson yang selalu mendukung dalam berbagai hal.
8. Teman-teman Fasilkom angkatan 2011.

9. Semua pihak yang telah membantu untuk kelancaran penyusunan tugas akhir ini.

Penyusunan tugas akhir ini masih terdapat banyak kekurangan. Untuk itu, di harapkan masukan berupa kritik dan saran yang bersifat membangun.

Akhirnya, semoga tugas akhir ini dapat memberikan manfaat bagi penulis, pembaca dan semua pihak yang berkepentingan dengan tugas akhir ini

Jakarta, September 2015

Arnold Solian Siagian

DAFTAR ISI

Halaman Judul.....	i
Lembar Pengesahan Tugas Akhir.....	ii
Lembar Pengesahan Penguji Sidang	iii
Lembar Pernyataan Keaslian.....	iv
Abstrak.....	v
Kata Pengantar	vi
Daftar Isi	viii
Daftar Gambar.....	xi
Daftar Tabel	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah.....	2
1.3 Tujuan Penelitian	2
1.4 Batasan Masalah	3
1.5 Metodologi.....	3
1.6 Sistematika Penulisan	5
BAB II LANDASAN TEORI.....	6
2.1 Aplikasi <i>mobile</i>	6
2.2 Pengetian Sistem	5
2.3 Pengertian Informasi	8
2.4 Pengertian Sistem Informasi	9
2.5 Pengertian Penyewaan	9
2.6 Pengetian Android	10
2.7 Sistem Arsitektur Android	10

2.8	<i>User Interface (UI)</i>	11
2.9	Java	12
2.10	XML.....	13
2.11	JSON.....	13
2.11.1	Pengertian JSON.....	13
2.11.2	Struktur penulisa JSON.....	14
2.11.3	JSON dengan PHP dan MySQL	15
2.11.4	Parsing Data JSON ke Android.....	16
2.12	Eclipse	18
2.13	Metodologi pengembangan perangkat lunak.....	19
2.13.1	XP (Extreme Programming).....	21
2.13.2	Prinsip Dasar XP.....	21
2.13.3	Kunci Utama XP	21
2.13.4	Kelebihan XP	23
2.14	PIECES	24
2.15	UML.....	25
BAB III GAMBARAN UMUM DAN ANALISIS MASALAH		33
3.1	Tempat dan Waktu Penelitian	33
3.2	Struktur Organisasi.....	33
3.3	Visi dan Misi.....	33
3.4	Proses Bisnis Penyewaan	35
3.5	Proses Pengembalian	37
3.6	Analisis PIECES.....	38
3.7	Masalah Proses bisnis lama.....	41

BAB IV HASIL DAN PEMBAHASAN.....	41
4.1	Usulan Pemecahan Masalah..... 41
4.2	Pemodelan Aplikasi (Design App)..... 42
4.2.1	<i>Usecase Diagram</i> 42
4.2.1.1	Definisi Aktor 44
4.2.2	<i>Class Diagram</i> 47
4.2.3	<i>Activity Diagram</i> 47
4.2.4	<i>Sequence Diagram</i> 49
4.2.5	<i>Package Diagram</i> 52
4.2.7	<i>Component Diagram</i> 52
4.2.8	<i>Deployment Diagram</i> 54
4.3	Aturan penyewaan rental mobil 55
4.4	Implementasi dan <i>testing</i> aplikasi. 56
4.4.1	Implementasi Aplikasi 57
4.4.1.1	Implementasi Aplikasi admin..... 57
4.4.1.2	Implementasi Aplikasi Pelanggan 60
4.4.1.3	Implementasi Aplikasi Sopir..... 64
4.5	Analisis Proses Penyewaan baru 67
4.6	<i>Testing</i> Aplikasi 68
BAB V KESIMPULAN DAN SARAN.....	77
5.1	Kesimpulan 77
5.2	Saran 77

Daftar Gambar

Gambar 2.1	<i>Sistem Arcithecture Android</i>	11
Gambar 2.2	Logo Eclipse	19
Gambar 2.3	<i>Extreme Programming</i>	21
Gambar 2.4	<i>Structure UML</i>	26
Gambar 2.5	<i>Class Diagram</i>	27
Gambar 2.6	<i>Component Diagram</i>	28
Gambar 2.7	<i>Deployment Diagram</i>	29
Gambar 2.8	<i>Use case Diagram</i>	29
Gambar 2.9	<i>Sequence Diagram</i>	30
Gambar 2.10	<i>Sequence Diagram</i>	31
Gambar 2.11	<i>Activity diagram</i>	32
Gambar 3.1	<i>Structure Organisasi</i>	33
Gambar 3.2	Proses bisnis penyewaan mobil	35
Gambar 3.3	Proses bisnis pengembalian	37
Gambar 4.1	<i>Usecase diagram pelanggan</i>	43
Gambar 4.2	<i>Usecase Diagram Admin</i>	44
Gambar 4.3	<i>Usecase Diagram Sopir</i>	45
Gambar 4.4	<i>Class Diagram Rental mobil</i>	46
Gambar 4.5	<i>Activity Diagram user Admin</i>	47
Gambar 4.6	<i>Activity Diagram user Pelanggan</i>	48
Gambar 4.7	<i>Activity Diagram user Sopir</i>	49
Gambar 4.8	<i>Sequence Diagram user admin</i>	50
Gambar 4.9	<i>Sequence Diagram user Pelanggan</i>	51
Gambar 4.10	<i>Sequence Diagram user sopir</i>	52
Gambar 4.11	<i>Package Diagram Rental mobil</i>	53
Gambar 4.12	<i>Component Diagram Rental mobil</i>	54

Gambar 4.13	<i>Deployment Diagram</i> Rental mobil	54
Gambar 4.14	Menu Utama Admin.....	56
Gambar 4.15	Daftar mobil.....	57
Gambar 4.16	Daftar Pelanggan.....	57
Gambar 4.17	Daftar Sopir.....	58
Gambar 4.18	Daftar Penyewaan Baru.....	58
Gambar 4.19	Set sopir bertugas	59
Gambar 4.20	Rental List	59
Gambar 4.21	Status Penyewaan.....	60
Gambar 4.22	Login Pelanggan	61
Gambar 4.23	Daftar Mobil.....	61
Gambar 4.24	Form Penyewaan.....	62
Gambar 4.25	Detail My Order	63
Gambar 4.26	Login Sopir	64
Gambar 4.27	Daftar Penyewa	64
Gambar 4.28	Proses Penyewaan	65
Gambar 4.29	<i>Close Order</i>	65

DAFTAR TABEL

Tabel 4.1 Usulan Sistem baru.....	41
Tabel 4.2 Definisi Aktor.....	45
Tabel 4.3. <i>Testing unit</i> aplikasi pelanggan.....	67
Tabel 4.4. <i>Testing unit</i> aplikasi Sopir	68
Tabel 4.5. <i>Testing unit</i> aplikasi admin	69
Tabel 4.6. <i>Intergration Testing</i> aplikasi pelanggan	70
Tabel 4.7. <i>Intergration Testing</i> aplikasi Sopir	71
Tabel 4.8. <i>Intergration Testing</i> admin	72
Tabel 4.9. <i>Acceptance Testing</i> aplikasi pelanggan.....	73
Tabel 4.10. <i>Acceptance Testing</i> aplikasi Sopir	74
Tabel 4.11. <i>Acceptance Testing</i> Aplikasi admin.....	75

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan Sistem informasi pada era globalisasi ini merupakan sebuah solusi untuk mengatasi beberapa masalah dalam dunia bisnis. Perkembangan sistem informasi ini sangat berpengaruh terhadap pelayanan, sebuah perusahaan dituntut meningkatkan pelayanan terhadap konsumen yang semakin hari semakin bertambah dan kebutuhan konsumen akan produk barang dan jasa yang mengedepankan kualitas pelayanan (Suwardi, 2011).

Perkembangan teknologi saat ini terlebih teknologi perangkat *mobile* menuntut dunia bisnis harus bisa memanfaatkan teknologi sebagai sarana pembantu proses bisnis, seperti pemanfaatan kemajuan teknologi untuk membantu pelanggan di CV Abu Sulaiman Rent Car untuk memesan mobil kembali, karna perangkat *mobile* kini banyak digunakan dan sudah menjadi bagian yang tak terpisahkan dari kehidupan sehari – hari.

CV Abu Sulaiman Rent Car adalah sebuah perusahaan jasa yang bergerak di bidang transportasi yaitu peminjaman mobil atau rental mobil saat ini sistem penyewaan mobil dilakukan dengan cara yang lama yaitu via telepon atau langsung datang ke rental mobil tersebut, dari proses bisnis yang lama ada beberapa kekurangan seperti lamanya proses penyewaan mobil untuk di sewa, lamanya proses penyewaan karna pelanggan setia pada rental CV Abu Sulaiman Rent car tidak bisa mengetahui informasi mobil yang tersedia secara *real time*, melakukan penyewaan dengan cepat dan juga pengelola juga masih menggunakan alat kerja yang lama seperti pendataan pelanggan, mobil, sopir, dengan buku tulis saja, beberapa masalah tersebut menjadi ide utama pembuatan aplikasi *mobile* ini dan akan dikembangkan ke sistem penyewaan mobil yang baru.

Hasil pemikiran dari uraian diatas dituangkan kedalam sebuah penelitian tugas akhir dengan mengangkat judul ***“Pembangunan Sistem Aplikasi Penyewaan Mobil Berbasis Android Studi Kasus di CV Abu Sulaiman Rent Car”***.

1.2 Identifikasi Masalah

Identifikasi Masalah dalam penelitian ini meliputi :

1. Bagaimana membangun sistem penyewaan mobil yang berbasis android agar pelanggan dirental mobil CV Abu Sulaiman Rent Car tidak harus datang kembali ke tempat rental mobil untuk memesan mobil ?
2. Bagaimana mengimplementasikan aplikasi android untuk membantu agar pengelola rental mobil Sulaiman Rent car dapat memberi informasi mobil kepada pelanggan, mengelolah data rental dengan baik agar memberi pelayanan terbaik kepada pelanggan?

1.3 Tujuan Penelitian

Adapun Tujuan yang ingin dicapai meliputi :

1. Pengelola dan Pelanggan dapat kembali melakukan proses penyewaan dengan cepat dan rental mobil dapat memberi informasi harga mobil, spesifikasi mobil, sehingga pelanggan mendapat informasi yang lengkap mengenai mobil-mobil yang akan disewakan dan tidak perlu datang ke tempat rental mobil kembali.
2. Rental mobil dapat efisien mengelola data mobil, pelanggan supir.

Manfaat penelitian ini adalah :

1. Manfaat penelitian ini bagi pengelola bisa dijadikan sebagai sarana alternatif dalam meningkatkan jasa penyewaan mobil di tempat penyewaan mobil.
2. Pengelola mendapat suatu sistem penyewaan mobil menggunakan perangkat mobile yang berbasis Android dengan efisien.

1.4 Batasan Masalah

1. Aplikasi dibuat diatas *platform* Android sehingga dimungkinkan hanya dapat dijalankan pada *smartphone* berbasis Android
2. Aplikasi ini hanya bisa di gunakan oleh pelanggan di CV Abu Sulaiman Rent Car dan Pengelola
3. Pengembangan Aplikasi ini hanya membahas proses penyewaan mobil di CV Abu Sulaiman rent car dengan berbasis Android

1.5 Metodologi

Metode penelitian yang digunakan untuk mencapai tujuan yang dirumuskan dalam penelitian adalah sebagai berikut

1. Metode pengumpulan data

Teknik yang dilakukan untuk mengumpulkan data:

A. Survey

a. Observasi

Pada metode observasi ini penulis melakukan pengamatan langsung di CV Abu Sulaiman Rent Car. Penelitian ini dilakukan untuk mengetahui proses bisnis yang sedang berjalan.

b. Wawancara

Pada metode wawancara ini, penulisan melakukan wawancara kepada pengelola CV Abu Sulaiman Rent Car.

c. *Internet Browsing*

Melakukan pencarian jurnal atau artikel di internet mengenai definisi-definisi dari istilah yang berkaitan dengan penelitian.

B. Studi Pustaka

Studi pustaka dilakukan dengan menggunakan sumber-sumber seperti artikel, buku, dan jurnal. Studi literatur ini berguna untuk mengetahui landasan teori serta pengeahuan dan informasi lingkungan internal dan eksternal sebagai acuan dalam penulisan skripsi.

2. Metode Analisis

Metode Analisis dengan Metode PIECES yang menggunakan enam untuk mengidentifikasi suatu masalah di CV Abu Sulaiman Rent Car :

1. kinerja (*performance*)
2. informasi (*information*)
3. ekonomi (*economic*)
4. pengendalian (*control*)
5. efisiensi (*efficiency*)
6. pelayanan (*service*)

1.6 Sistematika Penulisan Laporan

Sistematika yang digunakan dalam penulisan tugas akhir ini meliputi :

BAB I PENDAHULUAN

Pada Bab ini dijelaskan mengenai latar belakang, identifikasi masalah, tujuan dan manfaat penelitian.

BAB II LANDASAN TEORI

Pada Bab ini dijelaskan mengenai teori dasar sebagai penunjang dalam pembuatan tugas akhir ini.

BAB III GAMBARAN UMUM DAN ANALISIS MASALAH

Pada bab ini dijelaskan bagaimana menganalisis permasalahan terhadap aktivitas yang ada, menawarkan usulan sistem yang baru serta merancang pengembangan aplikasi.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini akan dibahas tentang pembuatan model, perancangan aplikasi, dan ujicoba aplikasi.

BAB V KESIMPULAN DAN SARAN

Pada bab ini berisi kesimpulan dan saran terhadap aplikasi yang dibuat agar dapat terus dikembangkan menjadi lebih baik.

BAB II

LANDASAN TEORI

2.1 Aplikasi *Mobile*

Kata *mobile* mempunyai arti bergerak atau berpindah, sehingga aplikasi *mobile* (Rangsang Purnama, 2010) adalah sebutan untuk aplikasi yang berjalan di *mobile device*. Dengan menggunakan aplikasi *mobile*, dapat dengan mudah melakukan berbagai macam aktifitas mulai dari hiburan, berjualan, belajar, mengerjakan pekerjaan kantor, *browsing* dan lain sebagainya.

2.2 Pengertian Sistem

Pengertian Sistem sangat luas dan beranekaragam, sehingga timbul berbagai definisi dan istilah tentang sistem menurut Gordon.B.Davis mengatakan: “Sistem merupakan kumpulan dari elemen-elemen (sub sistem) yang secara bersama-sama membentuk satu kesatuan dan saling berinteraksi dalam mencapai tujuan”. (Tata Sutabri, 2012)

Terdapat dua pendekatan dalam mendefinisikan sistem yaitu yang menekankan pada prosedur yang saling berhubungan untuk melakukan suatu kegiatan dan yang menekankan pada elemen-elemen yang berinteraksi, berhubungan untuk mencapai tujuan tertentu(Tata Sutabri, 2012).

Suatu sistem memiliki karekteristik atau sifat-sifat tertentu, yaitu:

1. Komponen Sistem (*System Components*)

Komponen sistem atau elemen-elemen sistem dapat berupa suatu sub sistem atau bagian-bagian dari sistem. Setiap sistem baik besar maupun kecil, selalu mengandung komponen-komponen atau subsistem- subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu

fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai suatu yang lebih besar yang disebut *supra system*.

2. Batas Sistem (*System Boundary*)

Batas sistem merupakan daerah-daerah yang membatasi antara satu sistem dengan sistem lainnya dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.).

3. Lingkungan Luar Sistem (*System Environment*)

Lingkungan luar sistem dari suatu sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem yang dapat bersifat menguntungkan dan dapat pula merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem yang harus dijaga dan dipelihara. Sedangkan yang merugikan harus ditahan dan dikendalikan, karena akan mengganggu kelangsungan hidup sistem.

4. Penghubung Sistem (*System Interprest*)

Penghubung sistem merupakan media penghubung antara satu sub sistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Dengan penghubung akan terjadi interaksi antar subsistem, sehingga membentuk satu kesatuan.

5. Masukan Sistem (*System Input*)

Masukan adalah suatu energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Contoh *maintenance*

input di dalam sistem komputer adalah program, yang digunakan untuk mengoperasikan komputer. Sedangkan *signal input* adalah energi yang diproses untuk mendapatkan keluaran. Contoh *signal input* di dalam sistem komputer adalah data, yang dapat diolah menjadi Informasi .

6. Keluaran Sistem (*System Output*)

Keluaran (*Output*) merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna .

7. Pengolah Sistem (*System Process*)

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya, yang bertugas untuk merubah masukan menjadi keluaran .

8. Sasaran Sistem (*System Objective*)

Suatu sistem pasti memiliki tujuan (*goal*) atau sasaran (*objective*). Suatu operasi sistem akan berguna dan berhasil apabila mencapai sasaran atau tujuannya. Sasaran sistem sangat menentukan masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem .

2.3 Pengertian Informasi

Dalam Informasi kehidupan sehari - hari informasi sering diartikan sebagai data. Dalam ruang sistem informasi hal ini berbeda walau sangat erat, dimana informasi merupakan data yang diproses, sehingga data ditransformasikan menjadi informasi dapat diproses dari data yang bentuknya kurang bermakna menjadi data yang berarti. Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan (Tata Sutarbi, 2012)

2.4 Pengertian Sistem Informasi

Sistem Informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan yang diperlukan (Tata Sutabri, 2012).

Manfaat adanya sistem informasi dalam suatu instansi yaitu:

1. Menyajikan informasi guna mendukung pengambilan suatu keputusan.
2. Menyajikan informasi guna mendukung operasi harian.
3. Menyajikan informasi yang berkenaan dengan kepengurusan.

Beberapa komponen sistem informasi dapat diklasifikasikan sebagai :

1. Perangkat keras (*hardware*) dan perangkat lunak (*software*) yang berfungsi sebagai mesin.
2. Manusia (*people*) dan prosedur (*procedures*) yang merupakan manusia dan tata cara menggunakan mesin.
3. Data merupakan jembatan penghubung antara manusia dan mesin agar terjadi suatu proses pengolahan data.

2.5 Pengertian Penyewaan

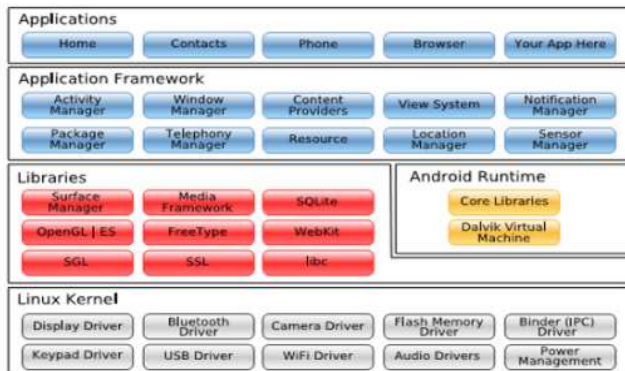
Penyewaan berasal dari kata dasar Sewa yang mendapat tambahan kataimbuhan Pe dan akhiran an. Sewa sendiri mempunyai arti yaitu pakaian sesuatu dengan membayar uang (www.kamusbesar.com, diakses pada tanggal 28 Juni 2015), sedangkan Penyewaan adalah suatu kegiatan yang melayani jasa peminjaman dengan tidak mengabaikan suatu ketentuan atau kesepakatan dan syarat – syarat yang berlaku didalam organisasi tersebut guna mencapai satu tujuan bersama.

2.6 Pengertian Android

Android merupakan perangkat bergerak pada sistem operasi untuk telepon seluler yang berbasis linux, Android menyediakan platform yang terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Android merupakan generasi baru platform mobile, platform yang memberikan pengembang untuk melakukan pengembangan sesuai dengan yang diharapkannya (Teguh Arifianto, 2011).

2.7 Sistem Arsitektur Android

Secara sederhana arsitektur Android merupakan sebuah kernel Linux dan sekumpulan pustaka Java dalam suatu *framework* yang mengatur dan menyediakan alur proses aplikasi.



Gambar 2.1 Sistem *architecture* of Android

(Sumber : Reto Meier, 2010)

2.8 *User interface* (UI)

Antarmuka pemakai (*User Interface*) merupakan mekanisme komunikasi antara pengguna (*user*) dengan sistem. Antarmuka pemakai (*User Interface*) dapat menerima informasi dari pengguna (*user*) dan memberikan informasi kepada pengguna (*user*) untuk membantu mengarahkan alur penelusuran masalah sampai ditemukan suatu solusi. *user interface*, berfungsi untuk menginputkan pengetahuan baru ke dalam basis pengetahuan sistem pakar, menampilkan penjelasan *sistem* dan memberikan panduan pemakaian sistem secara menyeluruh *step by step* sehingga *user* mengerti apa yang akan dilakukan terhadap suatu sistem (Teguh Arifianto, 2011)..

2.9 Java

Java API (*Application Programming Interface*) adalah sebuah layer pemrograman Java yang berisi *class-class* yang sudah didefinisikan dan antar muka pemrograman yang akan membantu pada pengembangan aplikasi dalam perancangan aplikasi (Arie Prabawati, 2010).

Berbeda dengan bahasa pemrograman yang lain yang hanya perlu di-*compile* atau hanya di-interpret sudah dapat menjalankan program di komputer, Java membutuhkan kedua hal tersebut (*compile dan interpret*) baru dapat menjalankan program yang telah dibuat.

Beberapa aturan dasar yang perlu diperhatikan sebelum mulai menulis program Java:

1. Setiap memulai untuk menulis program Java harus menggunakan *keyword class*.

2. Penggunaan tanda kurung (seperti { }, (), []) harus selalu berpasangan. Penggunaan tanda – tanda kurung tersebut memiliki peran masing – masing yang akan saudara pelajari nantinya.
3. Java bersifat *case sensitif*, artinya setiap huruf besar dan huruf kecil akan dibedakan oleh Java.
4. Penggunaan tanda ; (titik koma) untuk mengakhiri setiap perintah, pendeklarasian serta hal – hal lain.
5. Setiap program yang ditulis harus disimpan dengan nama **java** (diisi sesuai dengan nama **class**). Dan setelah di-*compile* akan menghasilkan **class**. Contoh: file **dkp.java** setelah di-*compile* akan menghasilkan **dkp.class**. Setiap program **.java** minimal menghasilkan satu file **.class**.

2.10 XML

XML adalah singkatan dari *eXtensible Markup language*. Bahasa *markup* adalah sekumpulan aturan-aturan yang mendefinisikan suatu sintaks yang digunakan untuk menjelaskan dan mendeskripsikan teks atau data dalam sebuah dokumen melalui penggunaan tag. XML terletak pada inti *Web Service* yang digunakan untuk mendeskripsikan data. Fungsi utama dari XML adalah komunikasi antar aplikasi, integrasi data, dan aplikasi-aplikasi yang berbeda dengan partner luaran. Dengan standarisasi XML, aplikasi-aplikasi yang berbeda dapat dengan mudah berkomunikasi antar satu dengan yang lain (Akhmad Dharma Kasman, 2013).

2.11 JSON

2.11.1 Pengertian JSON

JSON (*JavaScript Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah

diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 – Desember 1999 (Saat ini sudah terbit edisi ke-5 – Juni 2011). JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran-data. (www.json.org, di akses pada 28 juni 2015).

2.11.2 Struktur Penulisan JSON

JSON terbuat dari dua struktur :

1. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel hash (*hash table*), daftar berkunci (*keyed list*), atau *associative array*.
2. Daftar nilai terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (*array*), vektor (*vector*), daftar (*list*), atau urutan (*sequence*).

JSON menggunakan bentuk sebagai berikut:

Objek adalah sepasang nama/nilai yang tidak terurutkan. Objek dimulai dengan { (kurung kurawal buka) dan diakhiri dengan } (kurung kurawal tutup). Setiap nama diikuti dengan : (titik dua) dan setiap pasangan nama/nilai dipisahkan oleh , (koma).

Larik adalah kumpulan nilai yang terurutkan. Larik dimulai dengan [(kurung kotak buka) dan diakhiri dengan] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma).

Nilai (*value*) dapat berupa sebuah **string** dalam tanda kutip ganda, atau *angka*, atau true atau false atau null, atau sebuah *objek* atau sebuah *larik*. Struktur-struktur tersebut dapat disusun bertingkat.

String adalah kumpulan dari nol atau lebih karakter Unicode, yang dibungkus dengan tanda kutip ganda. Di dalam string dapat digunakan *backslash escapes* "\" untuk membentuk karakter khusus. Sebuah karakter mewakili karakter tunggal pada string. String sangat mirip dengan string C atau Java.

Angka adalah sangat mirip dengan angka di C atau Java, kecuali format oktal dan heksadesimal tidak digunakan.

Spasi kosong (*whitespace*) dapat disisipkan di antara pasangan tanda-tanda tersebut, kecuali beberapa detil *encoding* yang secara lengkap dipaparkan oleh bahasa pemrograman yang bersangkutan.

2.11.3 JSON dengan PHP dan MySQL

Sama seperti XML, JSON tidak menggantikan peran database, tetapi menjadi sarana pertukaran data yang diambil dari database. Pada versi PHP 5.2 telah terdapat fungsi untuk membuat JSON, yaitu fungsi **json_encode()**. Adapun penulisannya sebagai berikut :

string json_encode (mixed \$value [, int \$option = 0])

Contoh JSON dalam skrip php :

```
<?php
    //koneksi ke database
    $dbhost = "localhost";
    $dbuser = "root";
    $dbpass = "";
    $dbname = "anggota";
    Mysql_connect ($dbhost,$dbuser,$dbpass);
    Mysql_select_db(db_name);
    //query ambil data member
    $sql = "SELECT * FROM data member";
    $q = mysql_query($sql) or die(mysql_error());

    //buat variable $rows sebagai array
    $rows = array();
    While ($r = mysql_fetch_assoc($q) {
        $rows[] = $r;
    }
    ` $data = "{member:".json_encode($rows)}";
    echo $data;
?>
```

2.11.4 Parsing Data JSON ke Android

Untuk membuat parsing Data menggunakan JSON dilakukan dengan cara membuat sebuah class dalam java dengan nama JSONParser.java. Disini dimanfaatkan modul-modul yang akan dibutuhkan dengan mengimport *library* json, Apache dan IO milik

java untuk berkomunikasi dengan web server Apache.
(www.json.org, di akses pada 28 juni 2015).

```
import *;  
import org.apache.http.HttpResponse;  
import org.apache.http.client.methods.HttpGet;  
import org.apache.http.client.methods.HttpPost;  
import org.apache.http.client.utils.URLEncodedUtils;  
import org.apache.http.impl.client.DefaultHttpClient;  
import org.json.JSONException;  
import org.json.JSONObject;  
public class JSONParser {  
    static InputStream is = null;  
    static JSONObject jObj = null;  
    static String json = "";  
    // constructor  
    public JSONParser() {}  
    public JSONObject makeHttpRequest(String url, String  
method,List<NameValuePair> params) {  
        // Making HTTP request  
        try {  
            // check for request method  
            if (method == "POST") {  
                // request method is POST  
                // defaultHttpClient  
                DefaultHttpClient httpClient = new DefaultHttpClient();  
                HttpPost httpPost = new HttpPost(url);  
                httpPost.setEntity(new UrlEncodedFormEntity(params));
```

```

HttpResponse httpResponse = httpClient.execute(httpPost);
HttpEntity httpEntity = httpResponse.getEntity();
is = httpEntity.getContent();
} else if (method == "GET") {
// request method is GET
DefaultHttpClient httpClient = new DefaultHttpClient();
String paramString = URLEncoder.format(params, "utf-8");
url += "?" + paramString;
HttpGet httpGet = new HttpGet(url);
HttpResponse httpResponse = httpClient.execute(httpGet);
HttpEntity httpEntity = httpResponse.getEntity();
...
}
try {jObj = new JSONObject(json);
} catch (JSONException e) {
Log.e("JSON Parser", "Error parsing data " + e.toString());
return jObj;
}
}
}

```

2.12 Eclipse

Eclipse adalah sebuah *IDE (Integrated Development Environment)* untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (*platform independent*) (Akhmad Dharma Kasman, 2013).

Berikut ini adalah sifat dari Eclipse:

1. *Multi platform*: Target sistem operasi /Eclipse adalah *Microsoft Windows, Linux, Solaris, AIX HP-UX dan Mac OS X*.
2. *Multi-language* : Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis

bahasa pemrograman lainnya, seperti *C/C++*, *Cobol*, *Python*, *Perl*, *PHP* dan sebagainya.

3. *Multi-role* : Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari Eclipse yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plugin*.



Gambar 2.2 Logo Eclipse

(sumber : www.eclipse.org, diakses pada tanggal 19 juli 2015)

2.13 Metodologi Pengembangan Rekayasa Perangkat Lunak

2.13.1 XP (*Extreme Programming*)

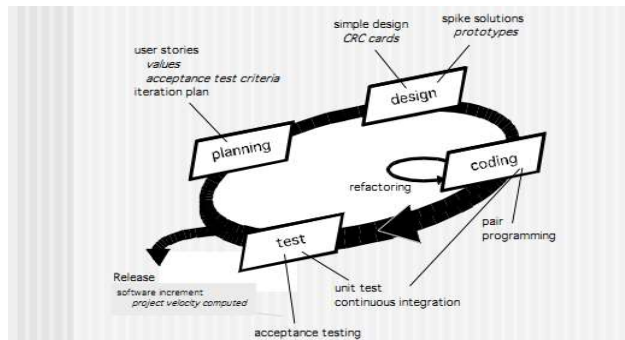
Extreme Programming merupakan salah satu metodologi dalam pengembangan rekayasa perangkat lunak dan juga merupakan salah satu dari beberapa *agile software development methodologies*

yang berfokus pada coding sebagai aktivitas utama disemua tahap pada siklus pengembangan yang lebih *responsive* terhadap kebutuhan *customer* (“*agile*”) dibandingkan dengan metode-metode tradisional sambil membangun suatu *software* dengan kualitas yang lebih baik, selain itu *extreme programming* meliputi seluruh area pengembangan perangkat lunak. Model agile proses ini dikembangkan oleh Kent Beck dan Ward Cunningham pada bulan maret 1996. Model *extreme programming* ini merupakan yang terpopuler dari beberapa metodologi pengembangan software yang dipakai untuk mengimplementasi proyek pengembangan perangkat lunak.

Extreme programming menggunakan pendekatan berorientasi objek sebagai paradigma pembangunan yang lebih disukai dan meliputi satu set *rules* dan *practices* yang berlaku dalam konteks dari 4 aktivitas *framework* (Pressman, 2010):

- a. **Planning**, aktivitas yang juga disebut sebagai *the planning game* dimulai dengan “mendengarkan” – sebuah aktifitas mengumpulkan kebutuhan yang memungkinkan para anggota teknikal dari tim XP untuk memahami konteks bisnis untuk perangkat lunak dan untuk mendapatkan *broad feel* untuk *output* yang dibutuhkan dan fitur utama serta fungsionalitas. Pada tahap ini, *customer* dan *developer* bekerja sama untuk menentukan bagaimana mengelompokkan cerita ke dalam rilis berikutnya/peningkatan perangkat lunak selanjutnya, yang akan dibangun oleh tim XP.
- b. **Design**, menyediakan panduan implementasi untuk proses seperti yang sudah dituliskan. Gagasan pusat di dalam XP adalah desain terjadi sebelum dan setelah *coding* dimulai.
- c. **Coding**, menterjemahkan penjabaran yang sudah dilakukan pada tahap pengkodean.

- d. **Testing**, pada tahap ini, *unit test* yang dikreasikan harus diimplementasikan menggunakan *framework* yang memungkinkan mereka menjadi otomatis (karenanya, dapat dieksekusi dengan mudah dan berulang-ulang). Hal ini mendorong strategi regresi *testing* ketika kode dimodifikasi.



Gambar 2.3. *Extreme Programming*

(Sumber : Roger S Pressman, ” Software Engineering A Practitioner's Approach 7th Edition”, 2010)

2.13.2 Prinsip dasar *Extreme Programming*

Terdapat lima prinsip dasar yang sangat fundamental dalam *Extreme Programming*, dimana prinsip -prinsip ini digunakan untuk menentukan apakah semua tindakan/pekerjaan yang telah dilakukan akan sukses atau sebaliknya (dalam konteks *Extreme Programming*) (Pressman, 2010) Kelima prinsip tersebut adalah:

1. Aliran umpan balik (*Rapid Feedback*)
2. Asumsi kesederhanaan (*Asume Simplicity*)
3. Penambahan perubahan (*Incremental Change*)

4. Pemelukan pekerjaan (*Embrace Work*)
5. Kualitas kerja (*Quality Work*)

2.13.3 Kunci Utama XP

XP sebagai sebuah metode yang dinamis diperlihatkan dalam empat *values* yang dimilikinya dan keempatnya merupakan dasar-dasar yang diperlukan dalam XP (Pressman, 2010), Kent Beck menyatakan bahwa tujuan jangka pendek individu sering berbenturan dengan tujuan sosial jangka panjang. Karena itu dibuatlah *values* yang menjadi aturan, hukuman, dan juga penghargaan. Keempat *values* tersebut adalah:

1. *Communication* (Komunikasi)

Komunikasi menekankan pada pendekatan yang lebih menekankan pada orang per orang secara langsung, dimana hal tersebut akan lebih baik daripada hanya berdasarkan pada dokumen yang menjelaskan tentang *software* yang dibangun. Dan juga, kemungkinan penggunaan dari beberapa praktis XP sehingga dibutuhkan alokasi waktu yang banyak untuk berkomunikasi dengan *customer*.

2. *Simplicity* (Kesederhanaan)

Simplicity adalah sebuah nilai (*value*) dari XP yang digunakan untuk memberikan solusi dari *problem* atau permasalahan yang dihadapi oleh *customer* sehingga *problem* atau masalah tersebut bisa disederhanakan. Kedua pihak yaitu

developer dan *customer* bisa mengerti solusi *software* jika itu tidak nyata, dimana solusi tersebut merupakan kandidat untuk praktis yang lain yang disebut dengan *Refactoring* (salah satu inti dari praktis XP).

3. *Feedback* (Umpan Balik)

Feedback berarti bahwa segala sesuatu yang telah dilakukan atau dicapai dievaluasi dengan *respect*/ reaksi untuk mengetahui bagaimana agar pekerjaan tersebut berjalan dengan baik dan menghasilkan *software* yang sesuai dengan kebutuhan *customer*. Pernyataan Bagaimana agar pekerjaan tersebut berjalan dengan baik mengindikasikan bahwa *feedback* diperoleh dari evaluasi setiap bagian pekerjaan dari solusi. *Feedback*, yang merupakan hasil kerja dari Nyquist di Laboratorium Bell pada awal tahun 1930, digunakan untuk menjamin bahwa solusi tersebut adalah benar. Pada saat *engineers* (insinyur) mulai mempelajari *feedback*, hal tersebut telah menjadi salah satu jalan dalam mencari solusi dari suatu masalah.

4. *Courage* (Keberanian)

Courage berarti bahwa pihak pengembang mempersiapkan segala sesuatunya untuk membuat keputusan yang sangat penting yang mendukung praktis XP pada saat membangun dan merilis (meluncurkan) *software* kepada *customer* untuk masing-masing iterasi

2.13.4 Kelebihan XP

Adapun kelebihan dari pemakaian *Extreme Programming* adalah sebagai berikut:

1. *Extreme Programming* adalah metode yang semi-formal yang artinya metode ini adalah metode yang fleksibel dan perubahan akan selalu diterima oleh developer tidak seperti pada metode *Waterfall* yang tidak fleksibel sehingga akan sulit mengadaptasinya ketika terjadi perubahan.

Resiko dari penggunaan metode *Extreme Programming* ini tidak lebih besar jika Kita memakai metode *Rapid Application Development* (RAD).

2.14 PIECES

Metode PIECES adalah metode yang menggunakan enam untuk mengidentifikasi suatu masalah terhadap kinerja (*performance*), informasi (*information*), ekonomi (*economic*), pengendalian (*control*), efisiensi (*efficiency*), dan pelayanan (*service*) variabel. Hasil analisis dari metode ini biasanya didapatkan rumusan masalah utama dari suatu masalah.

1. Analisis Kinerja System (*Performance*)

Masalah kinerja terjadi ketika tugas-tugas bisnis dijalankan dan tidak mencapai sasaran. Kinerja diukur dari jumlah produksi dan waktu tanggap. Jumlah produksi adalah sejumlah pekerjaan yang bisa diselesaikan selama jangka waktu tertentu. Waktu tanggap adalah keterlambatan rata-rata antara suatu transaksi dengan tanggapan yang diberikan kepada transaksi tersebut.

2. Analisis Informasi (*Information*)

Informasi merupakan hal penting karena dengan informasi tersebut pihak manajemen dan user dapat melakukan langkah selanjutnya. Apabila kemampuan sistem informasi baik, maka *user* akan mendapatkan informasi yang akurat, tepat waktu dan relevan sesuai dengan yang diharapkan. Informasi juga dapat merupakan fokus dari suatu batasan atau kebijakan.

3. Analisis Ekonomi (*Economy*)

Pemanfaatan biaya yang digunakan dari pemanfaatan informasi. Peningkatan terhadap kebutuhan ekonomis mempengaruhi pengendalian biaya dan peningkatan manfaat. Saat ini banyak perusahaan dan manajemen mulai menerapkan *paperless system* (meminimalkan penggunaan kertas) dalam rangka penghematan. Oleh karena itu dilihat dari penggunaan bahan kertas yang berlebihan dan biaya iklan di media cetak untuk media publikasi, sistem ini dinilai kurang ekonomis.

4. Analisis Kontrol (*Control*)

Analisis ini digunakan untuk membandingkan sistem yang dianalisa berdasarkan pada segi ketepatan waktu, kemudahan akses, dan ketelitian data yang diproses.

5. Analisis Pelayanan (*Service*)

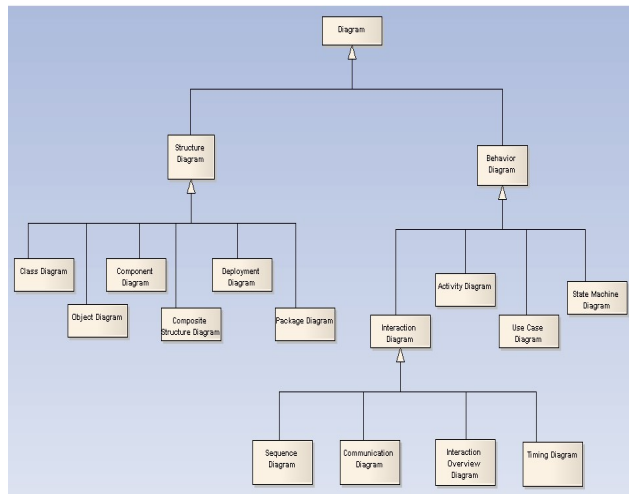
Peningkatan pelayanan memperlihatkan kategori yang beragam. Proyek yang dipilih merupakan peningkatan pelayanan yang lebih baik bagi manajemen (*marketing*), user dan bagian lain yang merupakan simbol kualitas dari suatu sistem informasi.

2.15 UML

UML adalah sebuah standarisasi pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman

berorientasi objek. UML muncul 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.(A.S&Shalahuddin, 2013)

Model *UML* terdiri atas banyak elemen-elemen grafis yang digabungkan membentuk diagram. Tujuan representasi elemen-elemen grafis ke dalam diagram adalah untuk menyajikan beragam sudut pandang



dari sebuah sistem berdasarkan fungsi masing-masing diagram tersebut. Kumpulan dari beragam sudut pandang inilah yang disebut sebuah model.

UML mendefinisikan diagram diagram sebagai berikut:

Gambar 2.4 *Structure UML*

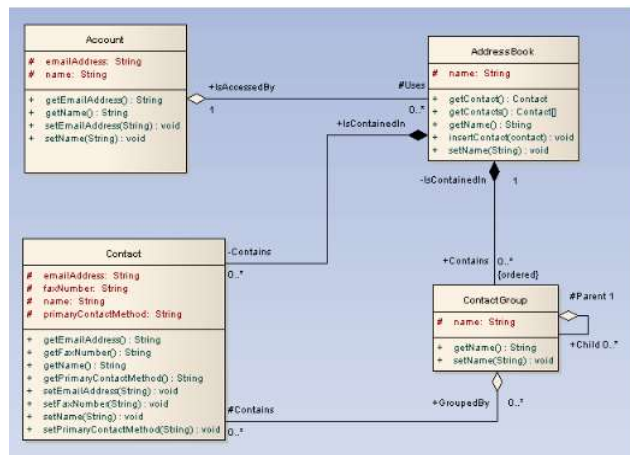
(sumber : Enterprise Architect 7.0)

1. *Class Diagram*

Class diagram menggambarkan struktur statis dari kelas dalam sistem dan menggambarkan atribut, operasi dan hubungan antara kelas. Class diagram membantu dalam memvisualisasikan struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Selama tahap desain, *class diagram* berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur *system* yang dibuat.

Class memiliki tiga area pokok :

1. Nama (dan *stereotype*)
2. Atribut
3. Metoda

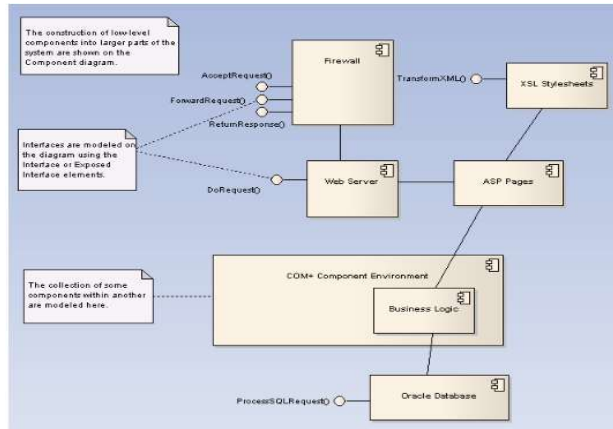


Gambar 2.5 *Class Diagram*

(sumber : Enterprise Architect 7.0)

2. *Component Diagram*

Component diagram menggambarkan struktur fisik dari kode, pemetaan pandangan logis dari kelas proyek untuk kode aktual dimana logikanya dilaksanakan.

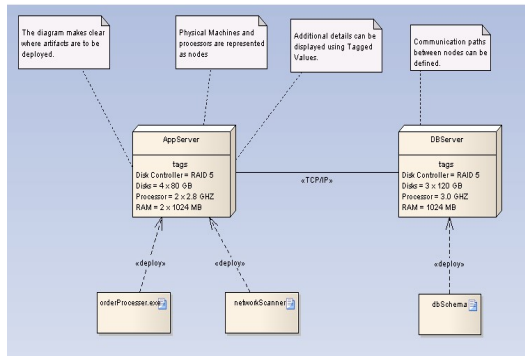


Gambar 2.6 *component diagram*

(sumber : Enterprise Architect 7.0)

3. *Deployment diagram*

Deployment diagram memberikan gambaran dari arsitektur fisik perangkat lunak, perangkat keras, dan artefak dari sistem. *Deployment diagram* dapat dianggap sebagai ujung spektrum dari kasus penggunaan, menggambarkan bentuk fisik dari sistem yang bertentangan dengan gambar konseptual dari pengguna dan perangkat berinteraksi dengan sistem.

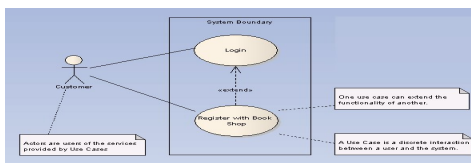


Gambar 2.7 Deployment Diagram
(sumber : Enterprise Architect 7.0)

4. Use Case Diagram

Diagram yang menggambarkan *actor*, *use case* dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor. Sebuah use case digambarkan sebagai elips horizontal dalam suatu diagram UML *use case*. *Use Case* memiliki dua istilah:

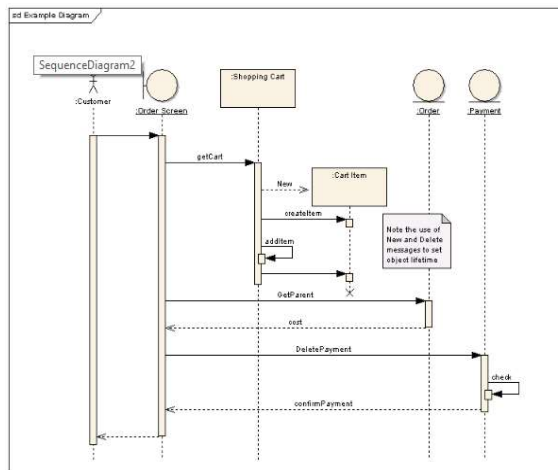
- System *use case*; interaksi dengan sistem.
- Business use case*; interaksi bisnis dengan konsumen atau kejadian nyata.



Gambar 2.8 use case diagram
(Sumber : Enterprise Architect 7.0)

5. Sequence diagram

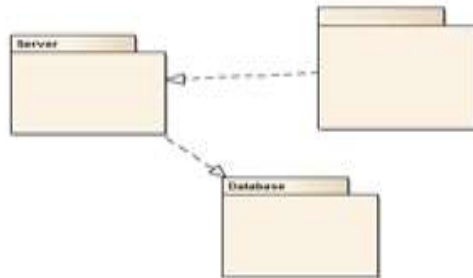
Sequence diagram menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Secara mudahnya *sequence diagram* adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan *use case diagram*.



Gambar 2.9 *sequence diagram*
(Sumber : Enterprise Architect 7.0)

6. *Package Diagram*

Di dalam diagram ini menjelaskan tentang elemen-elemen yang terkait dan berhubungan. Elemen-elemen tersebut berupa *database*, *server*, dan aplikasi .

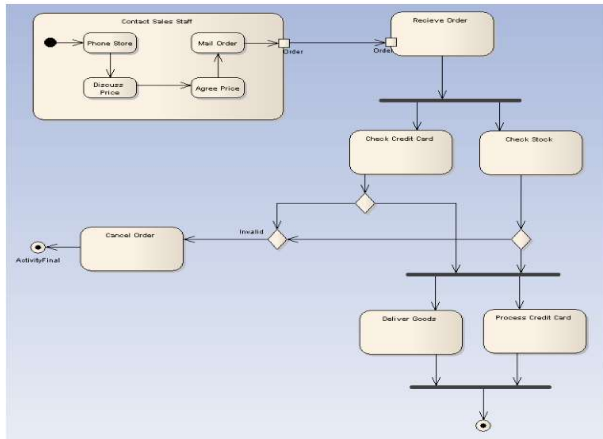


Gambar 2.10 *Package Diagram*

(Sumber : Enterprise Architect 7.0)

7. *Activity Diagram*

Menggambarkan aktifitas-aktifitas, objek, *state*, transisi *state* dan *event*. Dengan kata lain kegiatan diagram alur kerja menggambarkan perilaku sistem untuk aktivitas.



Gambar 2.11 Activity diagram
 (Sumber : Enterprise Architect 7.0)

BAB III
GAMBARAN UMUM DAN ANALISI MASALAH

3.1 Tempat dan waktu penelitian

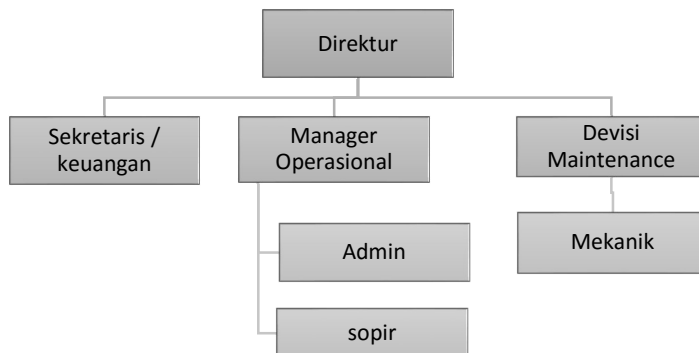
Tempat dan waktu penelitian dilaksanakan pada:

Tempat : JL. Raya Kebayoran Lama No. 18, Jakarta Selatan

Waktu : 10 Februari 2015

Telp : 021 (53652345)

3.2 Struktur Organisasi CV Abu Sulaiaman Rent Car



Gambar 3.1 Sturktur Organisasi

3.3 Visi dan Misi

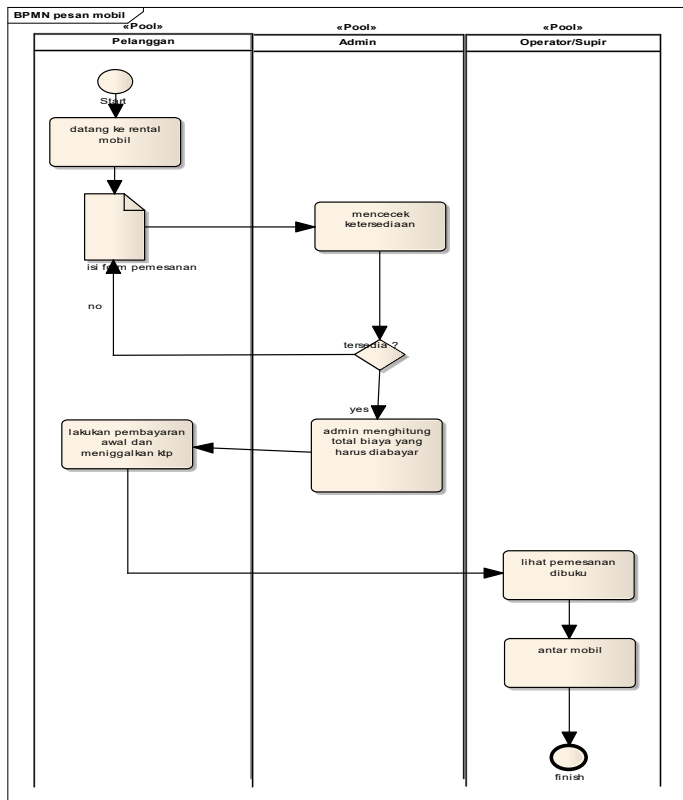
Visi :

Menjadi perusahaan dengan pelayanan terbaik yang siap bersaing di bisnis car rental dan perusahaan yang sehat, Profesional dan Inovatif serta membentuk SDM yang berkualitas dalam mencapai kesejahteraan bersama.

Misi :

1. Meningkatkan kepuasan pelanggan dengan kualitas layanan yang terbaik.
2. Mengantisipasi kecenderungan pasar dan kebutuhan pelanggan.
3. Mengembangkan Sumber Daya Manusia dengan memberikan penghargaan atas prestasi kerja.

3.4 Proses bisnis penyewaan

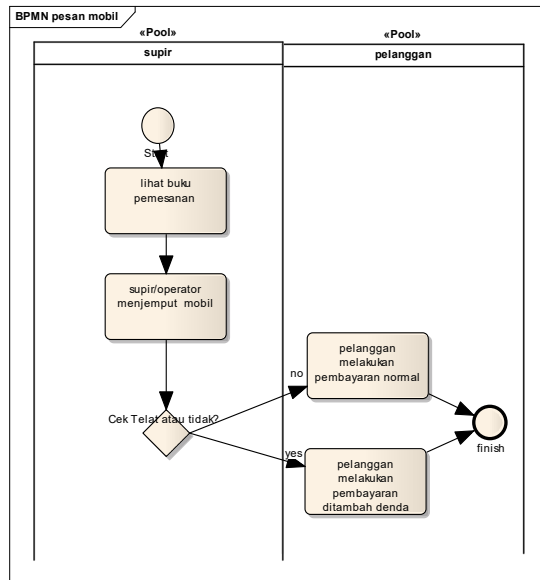


Gambar 3.2 proses bisnis (penyewaan)

Keterangan gambar:

1. Pelanggan datang ke rental mobil dan mengisi form penyewaan seperti identitas, tanggal penyewaan, lama waktu pemsanan, alamat pengantaran, apakah pakai supir atau tidak.
2. Admin mengecek data dan ketersediaan mobil dibuku.
3. Jika tersedia admin melakukan perhitungan total biaya peminjaman mobil .
4. Pelanggan melakukan pembayaran DP dan meninggalkan KTP
5. Admin memberi data penyewaan mobil berdasarkan form penyewaan pelanggan kepada supir.
6. Supir mengantar mobil ke pelanggan .
7. Proses selesai.

3.5 Proses Pengembalian



Gambar 3.3 proses bisnis (pengembalian)

Keterangan gambar :

1. Supir melihat buku penyewaan dan menjemput mobil sesuai alamat pengebalian .
2. Pelanggan mengembalikan mobil tepat waktu sesuai dengan ketentuan peminjaman mobil, apabila melewati waktu pengembalian maka pelanggan akan dikenakan biaya tambahan atau denda.
3. Pelanggan membayar sisa total biaya penyewaan.
4. Proses selesai.

3.6 Analisis PIECES

Berdasarkan proses bisnis diatas dapat dilakukan identifikasi masalah dengan cara melakukan observasi untuk mengamati permasalahan yang terjadi dan melakukan wawancara kepada pengelola CV Abu Sulaiman Rent Car.

Penyewaan dengan cara manual sudah berjalan dengan baik. Namun untuk membuat proses penywaan lebih efektif maka diperlukan suatu aplikasi *mobile* bagi pelanggan yang sudah terdaftar agar bisa memesan mobil dengan cepat tanpa harus datang kembali ke rental dan admin dapat mengelola segala data dengan cepat dan efisien.

Berdasarkan proses penyewaan diatas maka dilakukan analisis PIECES sebagai berikut :

1. Kinerja (*Performance*)

Kinerja adalah kemampuan sistem dalam menyelesaikan tugas dengan cepat sehingga sasaran dapat tercapai. Kinerja diukur dengan throughput (jumlah dari pekerjaan yang dapat diselesaikan selama jangka waktu tertentu) dan response time (waktu tanggap) adalah keterlambatan rata-rata antara suatu transaksi dengan tanggapan yang diberikan kepada transaksi tersebut dan Memproses data memerlukan waktu yang relatif lama karena banyaknya kegiatan pencatatan yang dilakukan seperti menulis, menggandakan, menghitung, mencocokkan data sehingga waktu tanggap (*response time*) semakin panjang.

2. Informasi (*Information*)

Pelanggan tidak bisa melihat informasi harga mobil *real time* ke rental mobil untuk mendapatkan informasi karna pelanggan harus datang ke rental mobil.

3. Ekonomi (*Economy*)

Kurang ekonomis karena penggunaan kertas, tinta dan alat tulis yang berlipat ganda dalam pencatatan data dan pembuatan laporan peminjaman mengakibatkan terjadinya pemborosan biaya.

4. Kontrol (Control)

Data – data seperti data form penyewaan, data transaksi semuanya menggunakan kertas yang siapa saja bisa melihat dan tidak adanya keamanan data.

5. Efisiensi (Efficiency)

Sistem lama sudah berjalan dengan baik namun belum efisien dilihat dari segi proses penyewaannya dimana pelanggan harus datang kembali ke rental mobil untuk sekedar mendapatkan informasi atau melakukan penyewaan mobil dan sopir tidak bisa melihat daftar penyewaan dengan cepat.

6. Pelayanan (Service)

Pelayanan terhadap pelanggan sudah berjalan dengan baik namun proses pemesanannya masih manual dimana pelanggan harus datang ke rental mobil .

3.7 Masalah di proses bisnis yang lama

1. Proses penyewaan mobil Di CV Abu Sulaiman Rent Car masih manual dimana pelanggan harus datang ke rental mobil untuk menyewa mobil.
2. Proses pendataan di CV abu Sulaiman Rent Car masih konvensional, dimana pendataannya masih menggunakan kertas,

BAB IV HASIL DAN PEMBAHASAN

4.1 Usulan Pemecahan Masalah

Berdasarkan hasil analisis PIECES dan Analisis Kebutuhan Sistem, maka diusulkanlah sebuah konsep pembuatan *mobile app* berdasarkan kebutuhannya, *User Stories* merupakan penjelasan tentang apa yang akan dilakukan oleh *user*. *User* dalam *mobile app* ini dibedakan menjadi tiga jenis, yaitu *user* sebagai Admin dan *user* sebagai pelanggan, *user* sebagai sopir.

Tabel 4.1 Usulan Sistem Baru

1	Pelanggan	Pelanggan CV Abu Sulaiman Rent car bisa melakukan penyewaan mobil, dengan <i>online</i> lewat aplikasi android
2	Admin	Admin bisa mengelolah data yang ada dalam sistem seperti mengelola data mobil , data operator. melihat daftar penyewa, menentukan supir yang akan bertugas lewat aplikasi Android.

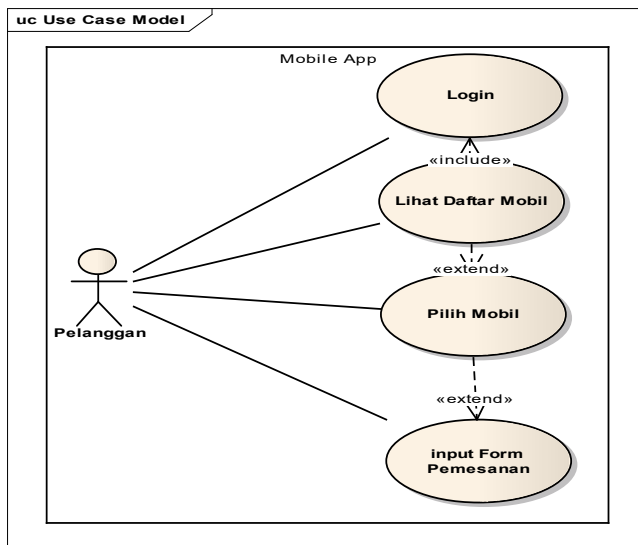
3	Sopir	Sopir bisa melihat data penyewa, mengupdate data pembayaran DP awal saat pengantaran mobil dan sisa pembayaran di saat penjemputan mobil dalam aplikasi android
---	-------	---

4.2 Pemodelan Aplikasi (*Design App*)

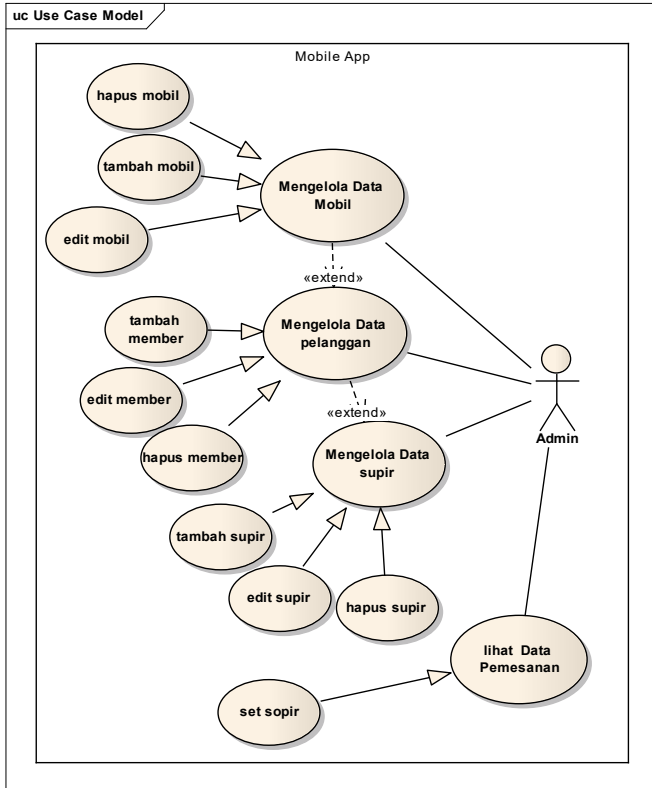
Pentingnya pemodelan aplikasi adalah untuk menggambarkan jalannya skenario program dalam model *object oriented* yang dirancang menggunakan Diagram UML.

4.2.1 Use Case

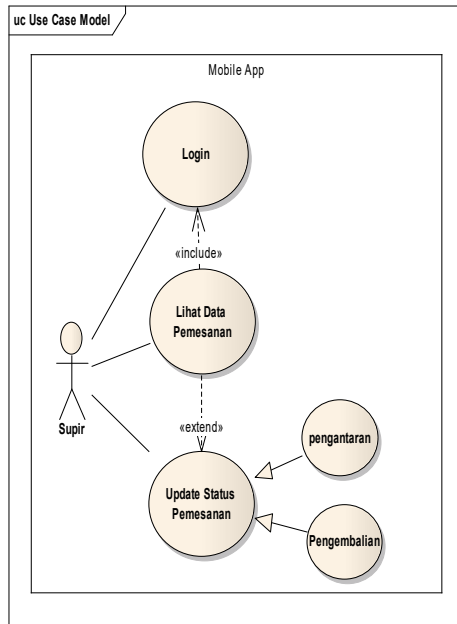
Use case digambarkan dalam 3 jenis berdasarkan jenis aktor yang terlibat dalam interaksi menggunakan mobile aplikasi ini. Aktor tersebut adalah user admin, user pelanggan dan user sopir.



Gambar 4.1 Usecase Diagram Rental Mobil Pelanggan



Gambar 4.2 Usecase Diagram Rental mobil Admin



Gambar 4.3 Usecase Diagram Rental mobil Admin

4.2.1.1 Definisi Aktor

Definisi aktor merupakan penjelasan dari apa yang di lakukan aktor – aktor yang terlibat dalam perangkat lunak yang dibangun, adapun deskripsi dari aktor – aktor yang terlibat dalam sistem rental mobil adalah.

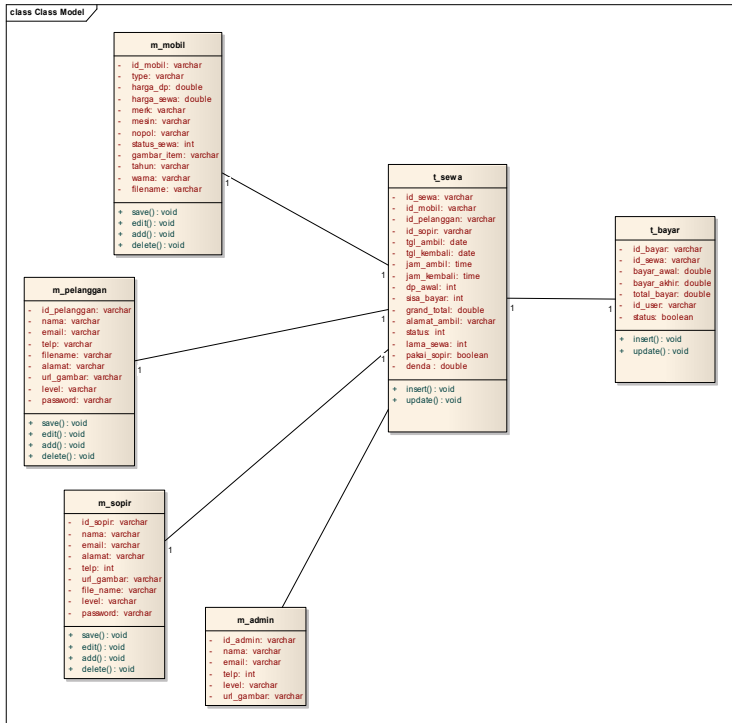
Tabel 4.2. Definisi Aktor

No	Aktor	Deskripsi
1	Pelanggan	Pelanggan merupakan aktor yang sudah terdaftar di rental mobil yang sudah melalui proses seleksi dari survey dari rental mobil, pelanggan bisa login, melihat datar mobil yang tersedia, memilih mobil dan mengisi form penyewaan di aplikasi android
2	Admin	Admin merupakan aktor yang mempunyai hak untuk mengelolah data yang ada dalam sistem seperti mengelola data mobil , data supir, data pelanggan dan melihat daftar pemesana mobil dan menset sopir yang bertugas .
3	Sopir	Sopir merupakan Aktor yang terdapat dalam Sistem yang memiliki hak untuk melihat daftar pelanggan yang menyewa mobil, mengupdate status data penyewaan pertama saat mobil diantar dan mengupdate status penyewaan saat mobil kembali.

4.2.2 Class Diagram

Class diagram adalah diagram yang digunakan untuk menampilkan bebeerap kelas yang ada di sistem aplikasi rental mobil yang akan dikembangkan, *class diagram* menunjukkan bagaimana class dalam sistem yang akan dibangun saling berkolaborasi untuk mencapai suatu

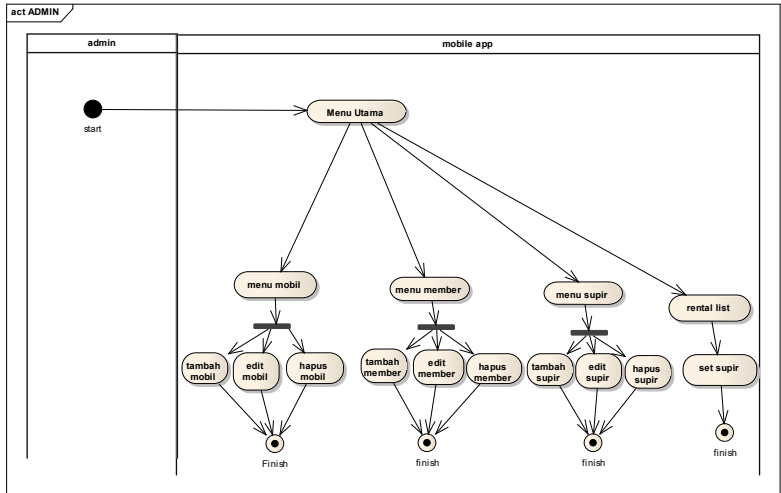
tujuan, berikut adalah gambar *class diagram* yang akan dibangun di sistem aplikasi penyewaan mobil :



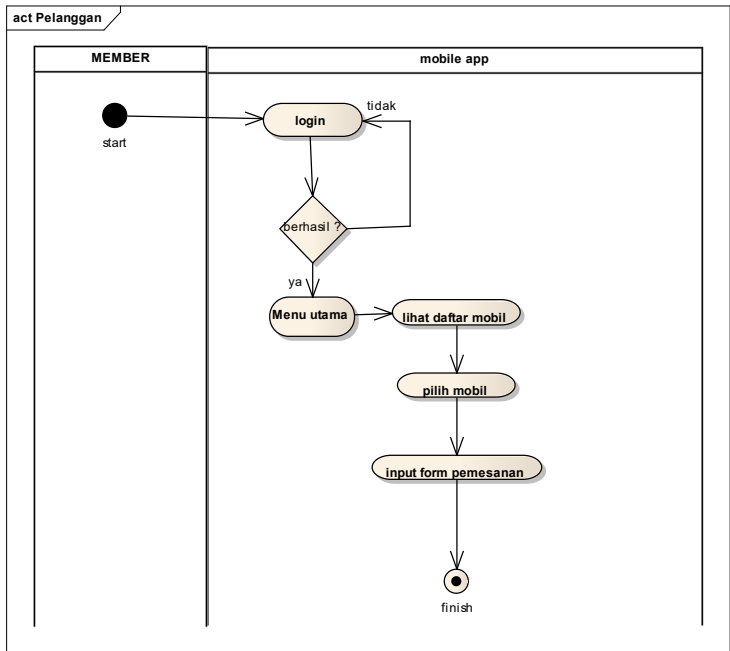
Gambar 4.4 Class Diagram

4.2.3 Activity Diagram

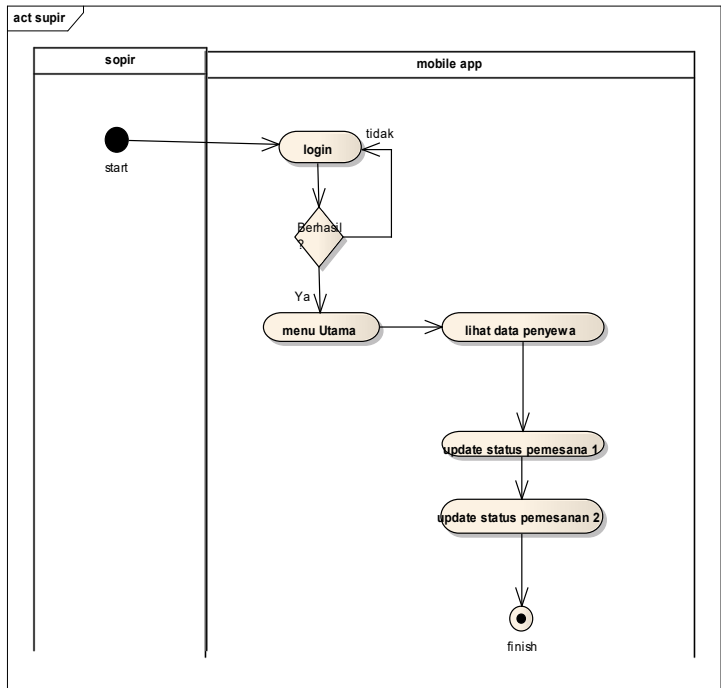
Activity Diagram sebagai gambaran tentang apa yang dikerjakan oleh user pada mobile app ini.



Gambar 4.5 Activity Diagram User Admin

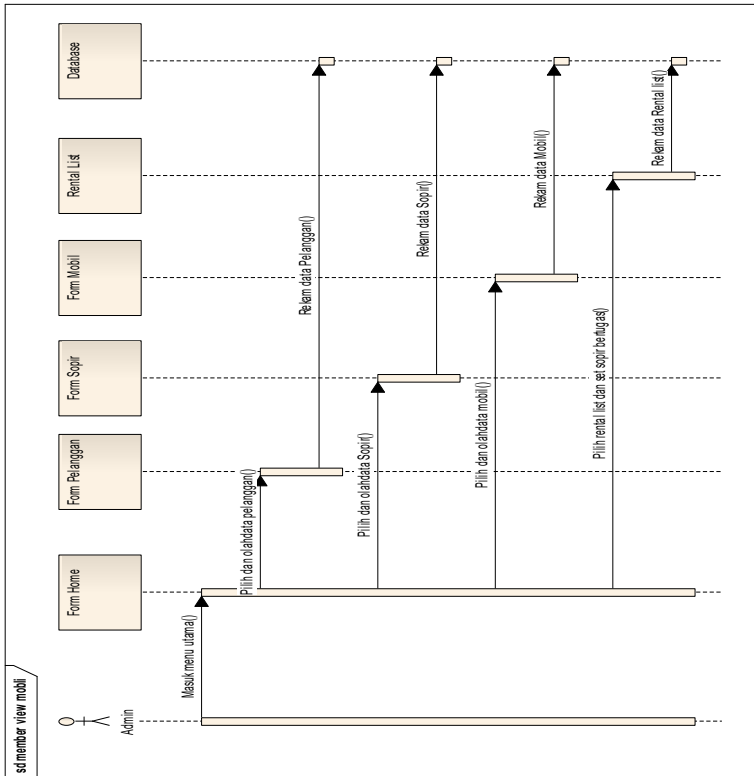


Gambar 4.6 Activity Diagram Rental Mobil User pelanggan

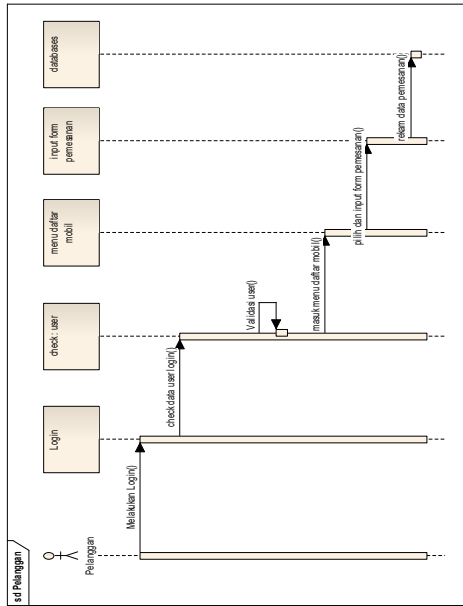


Gambar 4.7 Activity Diagram Rental Mobil User Sopir

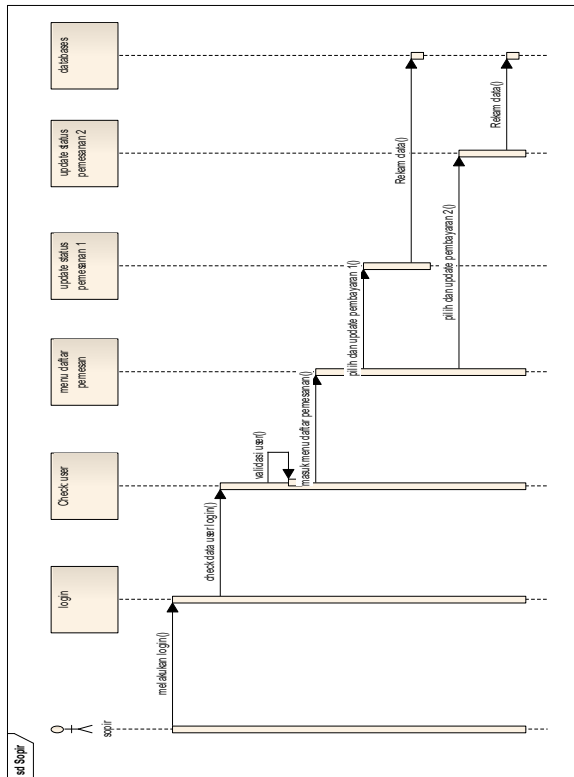
4.2.4 Sequence Diagram



Gambar 4.8 *Sequene Diagram Rental Mobil User Admin*



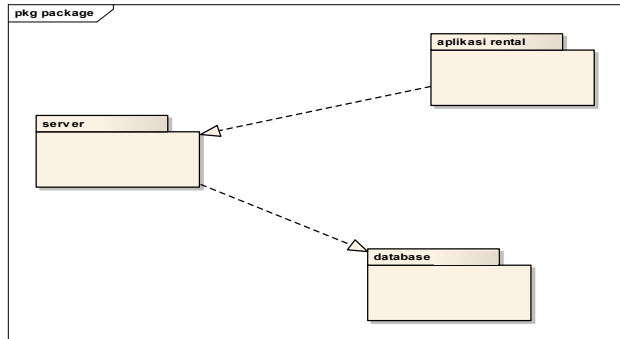
Gambar 4.9 *Sequence Diagram* Rental Mobil User Pelanggan



Gambar 4.10 *Sequenece Diagram* Rental Mobil User sopir

4.2.5 Package Diagram

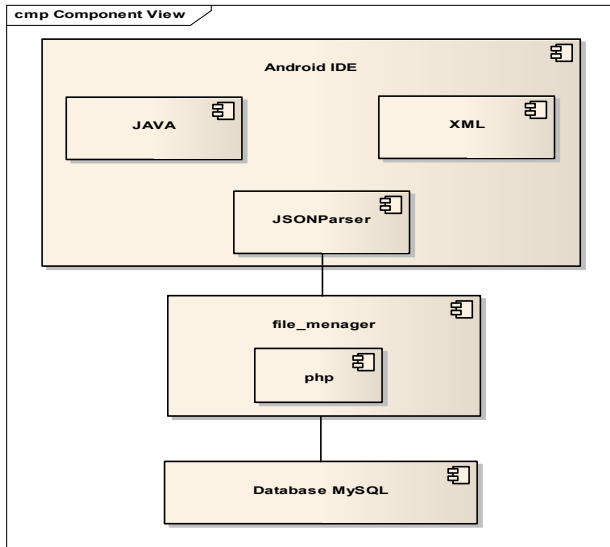
Di dalam diagram ini menjelaskan tentang elemen-elemen yang terkait dan berhubungan. Elemen-elemen tersebut berupa *database*, *server*, dan aplikasi penyewaan mobil.



Gambar 4.11. Package Diagram Rental mobil

4.2.6 Component Diagram

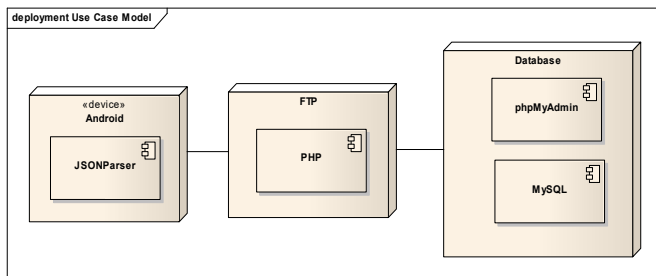
Component diagram digunakan untuk menggambarkan struktur dan hubungan antar komponen perangkat lunak, termasuk ketergantungan diantaranya, Komponen perangkat lunak berikan *source code* maupun *binary code* baik *library* maupun *executable*, dan bisa juga berupa interface yang ada didalam sistem aplikasi penyewaan mobil :



Gambar 4.12 Component Diagram

4.2.7 Deployment Diagram

Di dalam diagram ini menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Aplikasi ini dijalankan dengan perangkat *mobile* yang terhubung dengan *server* yang berfungsi menyimpan data aplikasi itu.



Gambar 4.13 Deployment Diagram

4.3 Aturan Penyewaan rental mobil

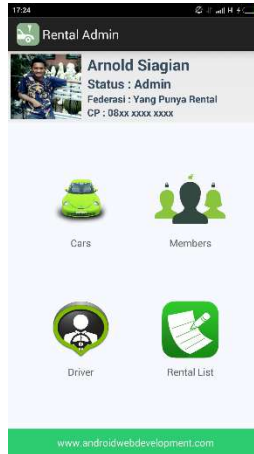
Dalam proses penyewaan terdapat aturan-aturan dalam melakukan penyewaan :

1. Pelanggan yang melakukan penyewaan mobil akan dihitung dalam 1 x 24 jam dalam 1 hari
2. Penyewaab mobil menggunakan jasa sopir dalam satu hari akan dikenakan biaya 2 x harga mobil yang disewa ditambah biaya supir 150 ribu dalam 1 hari diluar biaya bensin, tol ,dan biaya makan makan sopir minimal 14 ribu.
3. Pelanggan yang telat dalam pengembalian mobil akan dikenakan denda 10 persen perjam dari harga mobil.

4.4 Implementasi dan *Testing* Aplikasi

4.4.1 Implementasi Aplikasi

4.4.1.1 Implementasi Aplikasi Admin



Gambar 4.15 Menu Utama admin

Pada menu ini terdapat 4 pilihan menu yaitu menu :

1. *Cars*(Mobil) : Dalam menu ini terdapat daftar mobil dan pilihan untuk *edit* data, tsambah data, dan hapus data.
2. *Members* (Pelanggan): Dalam menu ini terdapat daftar Pelanggan dan pilihan untuk *edit* data, tambah data, dan hapus data.
3. *Driver* (Sopir): Dalam menu ini terdapat daftar Sopir dan pilihan untuk *edit* data, tambah data, dan hapus data.
4. *Rental List* : pada menu ini admin bisa melihat daftar pelanggan yang memesan mobil, dan admin bisa memilih supir yang akan bertugas.



Gambar 4.16 Daftar Mobil

Dalam menu ini terdapat daftar Pelanggan dan pilihan untuk *edit* data, tambah data, dan hapus data.



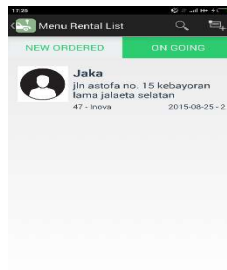
Gambar 4.17 Daftar pelanggan

Dalam menu ini terdapat daftar Pelanggan dan pilihan untuk *edit* data, tambah data, dan hapus data.



Gambar 4.18 Daftar sopir

Dalam menu ini terdapat daftar Sopir dan pilihan untuk *edit* data, tambah data, dan hapus data



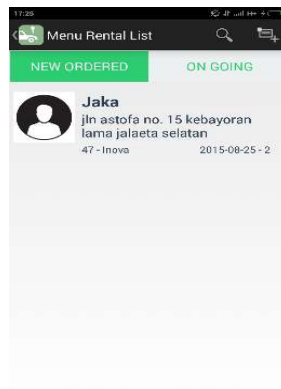
Gambar 4.19 Daftar penyewa

pada menu ini admin bisa melihat daftar pelanggan yang memesan mobil, dan admin bisa memilih supir yang akan bertugas.



Gambar 4.15 set sopir bertugas

Setelah admin melihat daftar penyewa mobil, admin memilih sopir yang akan bertugas baik untuk hanya sekedar mengantar mobil atau juga sebagai sopir pelanggan.



Gambar 4.20 Rental list

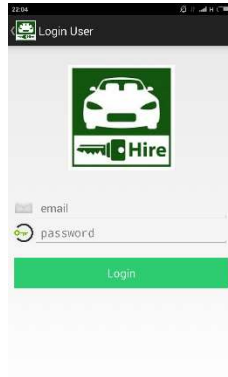
Setelah memilih supir yang bertugas maka status penyewaan menjadi *on going* yang artinya pemesanan dari pelanggan sudah di terima dan sopir yang ditugas akan bertemu dengan pelanggan, admin juga dapat mengetahui status pemesanan mobil dalam database apabila status pemesanan 1 maka status pemesan mobil belum diantar, apabila status pemesanan 2 maka mobil sudah diantar dan uang dp sudah di terima oleh supir, dan apabila status pemesanan 3 maka mobil sudah kembali dan sisa pembayaran sudah diterima oleh supir seperti dalam gambar.

dp_awal	sisa_bayar	grand_total	alamat_ambal	alamat_kembali	status	lama_sewa	pakai_sopir	denda
225000	225000	450000	kbn jeruk no. 5 jakarta barat		3	1	0	135000
175000	700000	875000	sevel tj duren		3	5	1	0
175000	1375000	1550000	cuu ou uf ig		2	2	1	0
200000	850000	1050000	jocucuu		3	3	0	0
175000	1375000	1550000	frgggr		2	2	1	0
225000	1725000	1950000	jin surgan no. 3 jakarta barat		2	2	1	0
200000	850000	1050000	kbn jeruk no. 6 jakarta barat		1	3	0	0
175000	1575000	1750000	Jakarta		3	5	0	0
175000	525000	700000	Jakarta		1	2	0	0

Gambar 4.21 Status Pemesanan

4.4.2 Implementasi aplikasi Pelanggan

4.4.2.1 Menu Pelanggan



Gambar 4.22 login Pelanggan

Sebelum pelanggan masuk ke halaman utama pelanggan, pelanggan harus melakukan login seperti memasukkan email dan password.



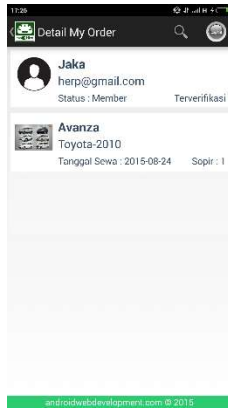
Gambar 4.23 daftar mobil

Pada menu ini pelanggan bisa melihat daftar mobil yang tersedia untuk disewa dan dimenu ini pelanggan juga bisa melihat *detail order*.



Gambar 4.24 form pemesanan

Setelah pelanggan memilih mobil yang akan disewa, setelah itu pelanggan akan mengisi form penyewaan, dalam form ini pelanggan akan menginput tanggal penyewaan, lama penyewaan, memilih memakai sopir atau tidak, dan memasukkan alamat ambil untuk alamat ambil yang dimaksud adalah alamat untuk bertemu atau menyerahkan mobil dan juga sebagai alamat pengembalian mobil.



Gambar 4.25 *Detail my order*

Pada menu detail order ini pelanggan akan melihat *detail order* atau pemesanan yang sudah dilakukan pelanggan

\

pada menu utama sopir ini, sopir akan dapat mengetahui daftar pemesanan mobil disini sopir akan mengantar mobil sesuai alamat yang tertera dan sopir juga bisa sebagai sopir apabila pelanggan ingin memakai sopir.



Gambar 4.28 proses pemesanan

Pada detail order ini terdapat detail pemesanan seperti jam bertemu, DP, sisa bayar, total bayar, pada tahap ini sopir akan menerima pembayaran DP dari pelanggan, setelah Sopir menerima uang DP dari pelanggan maka sopir mengklik *button* Proses yang artinya DP sudah diterima dan pelanggan menerima mobil.



Gambar 4.29 *close order*

Pada menu ini sopir akan melihat tanggal pengembalian mobil dan alamat pengembalian mobil, pada tahap pengembalian mobil ini sopir akan memilih detail order kembali dan akan memeriksa waktu pengembalian apakah tepat waktu atau tidak apabila telat maka langsung di denda 30% dari biaya sewa mobil 1 hari dan juga akan menerima sisa pembayaran penyewaan, setelah sopir menerima uang pembayaran maka sopir menklik *close order* dan transaksi akan disimpan ke database dan dengan mengklik *close order* maka transaksi akan selesai.

4.5 Analisis Proses Penyewaan baru

Berdasarkan proses penyewaan baru dengan aplikasi android maka di analisis kembali dengan PIECES yaitu sebagai berikut:

1. Kinerja (Performance)

Dengan adanya sebuah aplikasi untuk pelanggan, admin sangat membantu mempercepat segala proses manipulasi data dan pelanggan dapat langsung melihat dan tidak menunggu lama karna setiap proses penyewaan mobil bisa dilakukan *online* dimana saja

2. Informasi (Information)

Dengan adanya sebuah aplikasi android pihak rental mobil dapat memberikan informasi kepada pelanggan tentang ketersediaan mobil, pelanggan tidak harus datang ke rental mobil dan bisa langsung memesan mobil dengan aplikasi *mobile*.

3. Ekonomi (Economy)

Penggunaan kertas, tinta dan alat tulis lainnya bisa dikurangkan dengan adanya sebuah aplikasi android, karna sebagian aktivitas sudah dilakukan secara mobile seperti untuk memanipulasi data mobil, pelanggan, supir .

4. Kontrol (Control)

Segala informasi akan aman dan tidak sembarang orang dapat melihat karna mempunyai hak akses untuk masuk ke dalam sistem.

5. Efisiensi (Efficiency)

Dengan adanya aplikasi android pelanggan di CV Abu Sulaiman Rent car dapat dengan mudah untuk memesan mobil dan tentu lebih efisien dan sopir bisa melihat daftar penyewa mobil di aplikasi .

6. Pelayanan (Service)

Dengan adanya sebuah aplikasi penyewaan mobil yang *online* lewat android yang bisa diakses kapan saja meningkat kuliatas pelayanan penyewaan mobil terhadap pelanggan.

4.6 Pengujian (*Testing*)

Pada fase pengujian merupakan proses menjalankan program dengan tujuan mencari kesalahan (*error*) yang belum diketahui sebelumnya. Pengujian dikatakan "*Valid*" jika diperoleh kesesuaian antara kasus yang di ujikan dengan hasil yang diharapkan. Pengujian dilakukan dengan 3 tahap yaitu

1. *Unit Testing*

Unit testing yaitu mencoba alur yang spesifik pada struktur modul kontrol untuk memastikan kelengkapan secara penuh dan pendeteksian error.

Tabel 4.3 *Unit Testing* aplikasi pelanggan

No	Deskripsi Uji	Skenario	Data	Hasil	Status
1	<i>Login</i>	<i>Email/Password</i> benar	<i>Email</i> :test@gmail.com	<i>Login</i> diterima	OK
			Password : 123		
		<i>Email/Password</i> Salah	<i>Email</i> :test@gmail.com	<i>Login</i> ditolak	OK
			Password : 1234		
2	<i>Input form penyewaan</i>	<i>Input form penyewaan</i>	Tannggal sewa :	Data Tersimpan	OK
			Lama sewa:		
			Pakai supir :ya atau tidak		
			Alamat ambil		

Tabel 4.4 *Unit Testing* aplikasi Sopir

No	Deskripsi Uji	Skenario	Data	Hasil	Status
1	Login	ID/Password Benar	ID : 15	Login diterima	OK
			Password : 123		
		ID/Password Salah	ID : 15	Login ditolak	OK
			Password : 12		
2	Update status pemesanan / proses dan close order	Klik Proses atau close order	Nama : jaka	update Tersimpan	OK
			Alamat : jln assofa		
			Tanggal : 19-09-2015		
			Detail pembayaran		

Tabel 4.5 *Unit Testing* aplikasi admin

No	Deskripsi Uji	Skenario	Data	Hasil	Status
1	Tambah data sopir	Tambah data berhasil	Nama : Gunawan	Data disimpan	OK
			Email: gunawan@gmail.com		
			Alamat : kebon jeruk		OK
			Telp : 021 123435		
2		Tambah data mobil	Type : panthe	Data Tersimpan	OK
			Merk : isuzu		

	Tambah data mobil		Warna : merah		
			No Polisi : B 123 EFG		
			Harga sewa : 400000		
			Harga DP : 200000		
3	Tambah data Pelanggan		Nama : Solian	Data Tersimpan	ok
			Email: Solian@gmail.com		
			Alamat : kebon jeruk		
			Telp : 021 123435		
			Nama :Solian		
			Email: Solian@gmail.com		

2. Integration Testing

Integration Testing adalah tahapan proses pengujian aplikasi setelah fase unit testing dan sebelum *System Testing*

Tabel 4.6 *Integration* Aplikasi pelanggan

No	Skenario	Test Case	Hasil	Status
1	Berhasil <i>Login</i>	Pelanggan memasukkan Email dan password dan klik login	Masuk ke dalam Sistem Sebagai Pelanggan.	Berhasil
2	Berhasil <i>Input</i>	Pelanggan Mengklik mobil dan mengisi form	Data form Tersimpan	Berhasil

	form penyewaan			
--	-------------------	--	--	--

Tabel 4.7 *Intergration* Aplikasi Sopir

No	Skenario	Test Case	Hasil	Status
1	Berhasil <i>Login</i>	Sopir memasukkan id dan password dan klik login	Masuk ke dalam Sistem Sebagai Sopir.	Berhasil
2	Berhasil <i>Update</i> status pemesanan / proses dan close order	Sopir mengklik data pemesanan baru setelah itu klik proses saat pengantaran dan close order saat pengembalian mobil	Status pemesanan berhasil diubah	Berhasil

Tabel 4.8 *Intergration* Aplikasi Admin

No	Skenario	Test Case	Hasil	Status
----	----------	-----------	-------	--------




1	Berhasil Tambah data sopir	Admin menginput data sopir	Data sopir bertambah	Berhasil
2	Berhasil Tambah data pelangga n	Admin menginput data pelanggan	Data pelanggan bertambah	Berhasil
3	Berhasil Tambah data Sopir	Admin menginput data sopir	Data Sopir bertambah	Berhasil

3. *Acceptance Testing*

Acceptance Testing adalah pengujian formal dilakukan untuk menentukan apakah sistem sudah memenuhi kebutuhan fungsional sistem

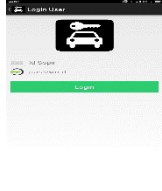


Tabel 4.9. *Acceptance Testing* aplikasi pelanggan

No	Deskripsi Uji	Skenario	Hasil yang diharapkan	Hasil	Hasil uji
----	---------------	----------	-----------------------	-------	-----------

1	<p>Login</p> 	<p>Mengisi id sopir dan password dengan benar</p>	<p>Sistem menerima login dan masuk ke halaman pelanggan</p>	<p>Berhasil Login</p>	<p>Valid</p>
2	<p>Input form penyewaan</p> 	<p>Input semua form tanggan sewa,lama sewa, pakai supir, alamat ambil</p>	<p>Sistem menerima form penyewaan dan disimpan</p>	<p>Sesuai harapan</p>	<p>Valid</p>
3	<p>Lihat detail order</p> 	<p>pelanggan mengklik detail order</p>	<p>Sistem menampilkan Detail order</p>	<p>Detail order berhasil di tampilkan</p>	<p>valid</p>




Tabel 4.10. Acceptance Testing aplikasi Sopir


No	Test case	Skenario	Hasil yang diharapkan	Hasil pengujian	Status
----	-----------	----------	-----------------------	-----------------	--------

1	<p>Login</p> 	mengisi Email dan password dengan benar	Sistem menerima login dan masuk ke halaman sopir	Berhasil Login	Valid
2	<p>Proses pemesanan</p> 	Sopir klik proses pemesanan	Sistem mengupdate status pemesanan	Sesuai harapan	valid
	<p>Close order</p> 	Sistem menampilkan identitas penyewa dan detail mobil	Sistem mengupdate status penyewaan	Close order berhasil di insert ke databases	Valid

Tabel 4.11 *Acceptance Testing* aplikasi Admin

No	Deskripsi Uji	Skenario	Hasil yang diharapkan	Hasil	Status
----	---------------	----------	-----------------------	-------	--------

1	<p>Tambah data pelanggan</p> 	Admin Input data pelanggan	Sistem Menyimpan data pelanggan ke database	Data pelanggan berhasil disimpan	Valid
2	<p>Tambah data mobil</p> 	Input data mobil	Sistem Menyimpan data mobil ke database	Data mobil berhasil disimpan	Valid
3	<p>Tambah data sopir</p> 	Input data mobil	Sistem Menyimpan data mobil ke database	Data Sopir berhasil disimpan	Valid

3.	<p>Set sopir bertugas</p> 	Admin memilih sopir yang bertugas	Sistem melakukan insert data	Sopir berhasil di tambah	valid

BAB V

KESIMPULAN DAN SARAN

KESIMPULAN

Aplikasi *mobile* Rental mobil ini membantu pelanggan di CV Abu sulaiman Rent car untuk melakukan penyewaan mobil dan mengetahui informasi tentang mobil dengan *online* tanpa harus datang ke rental mobil kembali. Dengan *mobile* aplikasi ini pengelola dapat memberi informasi kepada pelanggan dan pengelola dapat mengelola data mobil, pelanggan, supir, dengan mudah sehingga proses penyewaan mobil lebih cepat dan efisien.

SARAN

Perlunya pengembangan dari segi fitur yang dimiliki seperti penambahan fitur GPS dalam aplikasi *mobile* untuk mendukung penentuan posisi pelanggan saat meminjam mobil .

DAFTAR PUSTAKA

- Arifianto, Teguh. (2011). *Membuat Interface Aplikasi Android Lebih Keren dengan LWUIT*. Andi Publisher, Yogyakarta.
- A.S, Rossa dan Shalahuddin. (2013) *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*, Informatika Publisher, Bandung.
- Dharma Kasman, Akhmad. 2013. *Kolaborasi Dahsyat Android dengan PHP & MySQL*, Cetakan I. Lokomedia, Yogyakarta.
- Dhanta, Rizky. 2009. *Pengantar Ilmu Komputer*. Indah Publisher, Surabaya.
- Haviluddin. (2011). *Memahami Penggunaan UML*, Jurnal Informatika Mulawarman , Jakarta.
- Irawan, Budi . 2009 *jaringan Komputer*, Andi Publisher, Yogyakarta.
- Kustiyahningsih, Y & Annisa, Devie R. 2011. *Pemrograman Basis Data Berbasis Web Menggunakan PHP & MySQL* Graha Publisher, Yogyakarta.
- Meier, Reto., 2010. *Professional Android Application Development.*: Wiley Publishing.inc. Indianapolis.
- Prabawati,Arie 2010 *Tutorial 5 Hari Membangun GUI dengan Java Netbeans 6.5* , Andi Publisher. Yogyakarta.
- Pressman, R. S., 2010. *Software Engineering: A Practitioner's Approach*, Seventh Edition.:McGraww-Hill, pp. 120-125 In: New York.
- Purnama, Rangsang. 2010. *Pemrograman J2ME Tingkat Dasar*, PT. Gitamedi Publisher, Surabaya.

Safaat, Nazruddin. 2012 “*Pemograman Aplikasi MobileSmartphone dan Tablet PC*”, Informatika Publisher, Bandung.

Sutarbi Tata, 2012. *Konsep Dasar Sistem Informasi*, Andi Publisher, Yogyakarta.

Suwardi.s 2011 “*Pengaruh kualitas pelayanan terhadap kepuasan dan loyalitas pasien RSUD di Kabupate Boyolali Provinsi Jawa Tengah*”. Disertasi Universitas Tujuh Belas Agustus (Untag) Surabaya.

URL :

<http://www.kamusbesar.com>

<http://www.json.org>

<http://www.eclipse.org>

DAFTAR RIWAYAT HIDUP



Data Pribadi

Nama : Arnold Solian Siagian
Tempat, Tanggal Lahir : Tarutung, 29 Setember 1992
Alamat : Jl. Assofa Raya No.50 RT.007 RW.01 Kel.
Sukabumi Utara Kec. Kebon jeruk Jakarta Barat
11540.
Agama : Kristen Protestan
Status : Belum menikah
Tinggi / Berat Badan : 175 cm / 55 kg
E-mail : solian_arnold@yahoo.co.id

Riwayat Pendidikan

2011-2015 : Universitas Esa Unggul, Jakarta
2007-2010 : SMA Negeri 2, Tarutung
2004-2007 : SMP Negeri 4, Tarutung
1998-2004 : SD Lumban Baringin 173135 Sipoholon