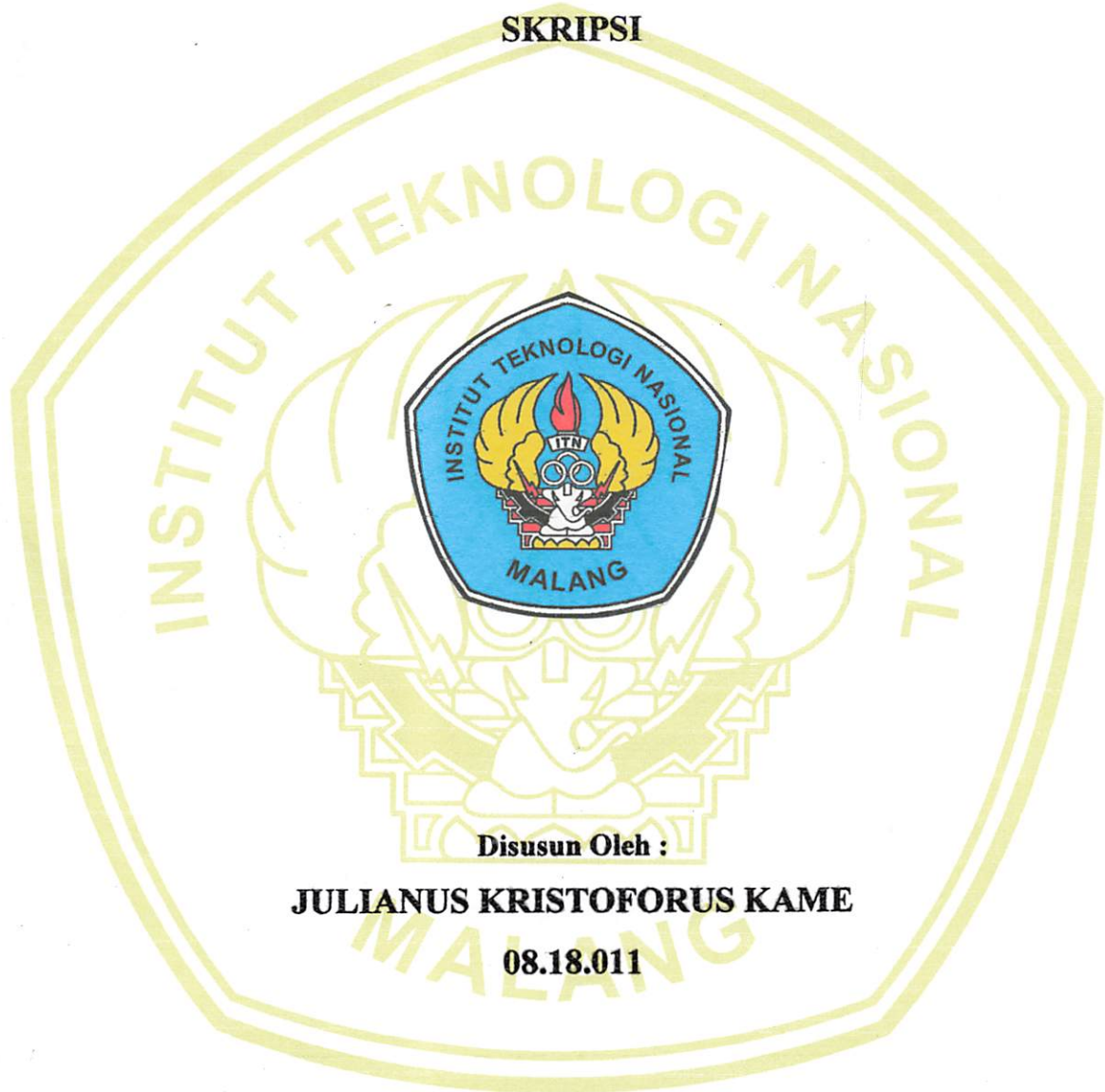


**IMPELEMENTASI APLIKASI ANDROID DALAM SISTEM INFORMASI
SENAYAN DI PERPUSTAKAAN ITN MALANG KAMPUS 2**

SKRIPSI



Disusun Oleh :

JULIANUS KRISTOFORUS KAME

08.18.011

**PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG**

2014

LEMBAR PERSETUJUAN
IMPLEMENTASI APLIKASI ANDROID DALAM SISTEM INORMASI
SENAYAN DI PERPUTAKAAN ITN MALANG KAMPUS 2

SKRIPSI

**Disusun dan diajukan untuk melengkapi dan memenuhi persyaratan guna
mencapai gelar Sarjana Teknik**

Disusun Oleh :

JULIANUS KRISTOFORUS KAME


08.18.011

Diperiksa dan disetujui,

Dosen Pembimbing 1

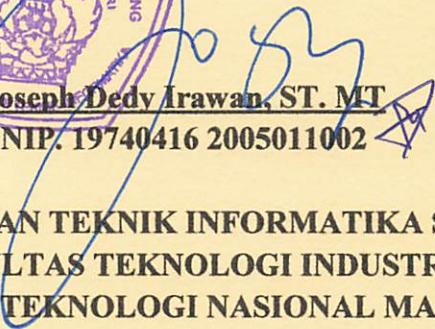
Dosen Pembimbing II


Josph Dedy Irawan, ST.MT
NIP.19740416 2005011002


Sonny Prasetyo, ST.MT
NIP.P. 1031000433

Mengetahui,
Ketua Program Studi Teknik Informatika S-1




Joseph Dedy Irawan, ST. MT
NIP. 19740416 2005011002

JURUSAN TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG

2014

IMPLEMENTASI APLIKASI ANDROID DALAM SISTEM INFORMASI SENAYAN DIPERPUSTAKAAN ITN MALANG KAMPUS 2

Julianus Kristoforus Kame (0818011)

Jurusan Teknik Informatika S-1
Fakultas Teknologi Industri
Institut Teknologi Nasional Malang
Jln.Raya Karanglo Km 2 Malang
Email: itokador@yahoo.co.id

Dosen Pembimbing : 1. Joseph Dedy Irawan, ST.MT
2. Sonny Prasetio, ST.MT

Abstrak

Perpustakaan kampus adalah tempat dimana mahasiswa mencari bahan dalam mencari ilmu, diantaranya buku materi, kamus dan lain-lain. Di sini mahasiswa harus datang ke perpustakaan untuk meminjam buku dengan syarat memiliki kartu perpustakaan namun hal ini memiliki kekurangan yaitu waktu yang terbatas dalam peminjaman buku dan waktu yang terbatas yang dimiliki perpustakaan. Berdasarkan permasalahan diatas penulis membuat aplikasi android dalam system informasi senayan di perpustakaan ITN Malang Kampus 2.

Dengan memanfaatkan teknologi informasi guna menyelesaikan persoalan diatas dalam bentuk rancang bangun sebuah Digital Library jurusan Teknik Informatika ITN Malang. Implementasi aplikasi android dalam system informasi senayan di perpustakaan ITN Malang kampus 2 ini diharapkan berguna bagi para mahasiswa dalam pencarian buku dan bahan refrensi.

Kata kunci : Aplikasi Perpustakaan, Android,



**PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL
MALANG**

PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan di bawah ini:

Nama : Julaianus Kristoforus Kame
Nim : 08.18.011
Program Studi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Menyatakan dengan sesungguhnya bahwa Skripsi saya yang berjudul:

**“IMPLEMENTASI APLIKASI ANDROID DALAM SISTEM INFORMASI
SENAYAN DI PERPUSTAKAAN ITN MALANGKAMPUS 2”**

Adalah hasil karya sendiri bukan hasil karya orang lain, kecuali kutipan yang telah saya sebutkan sumbernya

Malang, Mei 2014

Yang membuat pernyataan



Julianus Kristoforus Kame

KATA PENGANTAR

Dengan memanjatkan puji syukur kehadirat Tuhan Yang Maha Esa. Karena atas rahmat Nya yang telah dilimpahkan, sehingga kami dapat menyelesaikan Laporan Skripsi “Impelementasi Aplikasi Android Dalam Sistem Informasi Senayan Di Perpustakaan ITN Malang Kampus 2” ini dengan baik dan lancar.

Laporan “Impelementasi Aplikasi Android Dalam Sistem Informasi Senayan Di Perpustakaan ITN Malang Kampus 2” merupakan salah satu persyaratan akademik dalam menyelesaikan Program Strata 1 Jurusan Teknik Informatika, Institut Teknologi Nasional Malang.

Oleh karena itu pada kesempatan ini dengan segala kerendahan hati, perkenankanlah penyusun mengucapkan terima kasih kepada :

1. **Tuhan Yang Maha Esa**, yang selalu memberikan kesehatan bagi penyusun sehingga dapat melaksanakan Skripsi dengan baik.
2. **Kedua Orang Tua**, serta keluarga penyusun yang telah memberikan dorongan baik secara moril maupun materiel untuk menyelesaikan Skripsi ini.
3. **Bapak Ir. Soeparno Djiwo, MT.**, selaku Rektor Institut Teknologi Nasional Malang
4. **Bapak Joseph Dedy Irawan, ST.MT.**, Ketua Jurusan Teknik Informatika S-1 Institut Teknologi Nasional Malang.
5. **Bapak Joseph Dedy Irawan, ST.MT.**, selaku Dosen Pembimbing I.
6. **Bapak Sonny Prasetio, ST.MT.**, selaku Dosen Pembimbing II.
7. **Bapak Ali Mahmudi, BEng, Phd.**, selaku dosen penguji I.
8. **Ibu Karina Auliasari, ST, M.Eng.**, selaku dosen penguji II.
9. Serta semua pihak yang telah membantu secara langsung maupun tidak langsung sehingga penulis dapat menyelesaikan laporan ini.

Penulis menyadari Laporan ini masih belum sempurna, oleh karena itu kami mengharap kritik dan saran serta penilaian yang bersifat membangun dari semua pihak guna sempurnanya Laporan ini.

Akhir kata penyusun mohon maaf yang sebesar-besarnya bilamana dalam penyusunan Laporan ini terdapat kekurangan serta kesalahan. Semoga Laporan Skripsi “Impelementasi Aplikasi Android Dalam Sistem Informasi Senayan Di Perpustakaan ITN Malang Kampus 2” ini bermanfaat bagi kita semua.

Malang, Mei 2014

Penulis

DAFTAR ISI

COVER

LEMBAR PERSETUJUAN	i
ABSTRAK	ii
PERNYATAAN KEASLIAN SKRPSI	iii
KATA PENGANTAR.....	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR.....	ix
DAFTAR TABEL.....	xi

BAB I PENDAHULUAN

1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan	2
1.5 Metode Penelitian	3
1.6 Sistematika Penulisan	4

BAB II LANDASAN TEORI

2.1 Android	4
2.1.1 Teori Android	6
2.1.2 Versi Android	7
2.2 Arsitektur Android.....	9
2.3 Komponen Android	13
2.4 Eclipse Indigo	13
2.5 Android Virtual Device	14
2.6 Android Emulator	15
2.7 Java	15
2.7.1 Sejarah Java	15

2.7.2 Java Arsitektur	17
2.7.3 Fitur Java	19
2.8 Eclipse	24
2.9 DFD	25
2.10 Diagram Konteks	26
2.11 ERD.....	26
2.11.1 Simbol ERD	26
2.12 Flowchart	31
2.13 PHP	32
2.13.1 Sejarah PHP	32
2.13.2 Tentang PHP	33

BAB III ANALISA DAN PERANCANGAN SISTEM

3.1 Analisa	34
3.2 Rancangan Sistem	34
3.3 Rancangan Alur Program	35
3.4 Flowchart	35
3.4.1 Flowchart User	35
3.4.2 Flowchart Admin	37
3.4.3 Perancangan class Diagram	39
3.4.4 Perancangan Activity Diagram	40
3.4.5 Perancangan Squence	42
3.4.6 Struktur Data Base	43
3.4.7 Desain Struktur Navigasi	49
3.4.8 Perancangan Menu.....	49

BAB IV IMPLEMENTASI DAN PENELITIAN

4.1 Implementasi Sistem.....	51
4.1.1 Kebutuhan Hardware dan Software	51
4.2 Tampilan Aplikasi.....	52
4.2.1 Tampilan Halaman Beranda	52

4.2.2 Tampilan Halaman Form Petugas	52
4.2.3 Tampilan Halaman Form Anggota	53
4.2.4 Tampilan Menu Utama Petugas	53
4.2.5 Tampilan Halaman List Anggota	54
4.2.6 Tampilan Halaman Detail Anggota.....	54
4.2.7 Tampilan Halaman List Buku	55
4.2.8 Tampilan Halaman Hasil Pencarian Buku	55
4.2.9 Tampilan Halaman Semua List Peminjaman Buku.....	56
4.2.10 Tampilan Halaman Admin Untuk Menambah Buku.....	56
4.2.11 Tampilan Halaman Penambahan Buku.....	57
4.3 Pengujian.....	58
4.3.1 Pengujian Metode Black Box.....	58
4.3.2 Tabel Pengujian Pengguna	59
4.3.3 Tabel Pengujian Terhadap Versi Android	60
4.3.4 Tabel Pengujian Terhadap Resolusi Layar	60

BAB V PENUTUP

5.1 Kesimpulan	62
5.2 Saran	62

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

2.1	Arsitektur Android	9
2.5	Android Virtual Device.....	14
2.6	Android Emulator	15
2.7	Arsitektur J2EE	17
2.8	Arsitektur J2SE	18
2.9	Arsitektur J2ME	18
2.11	Data Flow Diagram.....	25
2.12	Simbol Konteks Diagram	26
2.13	Simbol Entity	27
2.14	Simbol Relationship	27
2.15	Relationship diantara 2 Entity	27
2.16	Simbol Atribut.....	28
2.17	Atribut dari 1 Entity	28
2.18	Relationship Derajat 1.....	28
2.19	Relationship derajat 2.....	29
2.20	Relationship Derajat 3.....	29
2.21	Rasio 1 to 1.....	30
2.22	Rasio 1 to M	30
2.23	Rasio M to 1	30
2.24	Rasio M to M.....	30
2.25	Simbol Flowchart	31
3.1	Rancangan Penelitian.....	34
3.2	Flowchart User.....	35
3.3	Flowchart Admin	37
3.4	Use Case Diagram di Server.....	38
3.5	Use Case Diagram Pada Client.....	38
3.6	Class Diagram.....	39
3.7	Activity Diagram	41
3.8	Diagram Squence.....	42

4.1 Menu Awal.....	52
4.2 Form Petugas.....	52
4.3 Form Anggota.....	53
4.4 Halaman Utama Petugas	53
4.5 List Anggota.....	54
4.6 Detail List anggota	54
4.7 Pencarian Buku	55
4.8 Halaman Hasil Pencarian Buku	55
4.9 Semua List Peminjaman Buku	56
4.10 List Buku Yang Akan Di Tambahkan Oleh Admin	56
4.11 List Penambahan Buku Oleh Admin	57

DAFTAR TABEL

3.1	Tabel_login	43
3.2	Tabel_Anggota	43
3.3	Tabel_Petugas	44
3.4	Tabel_menu Petugas	45
3.5	Tabel_Buku	45
3.6	Tabel_Server	46
3.7	Tabel_Peminjaman	46
3.8	Tabel_Daftar Peminjaman	47
3.9	Tabel_Tentang	47
3.10	Tabel_Petunjuk	47
3.11	Tabel_Logout	48
3.12	Tabel_Beranda	48
4.1	Tabel Pengujian Black Box	58
4.2	Tabel Pengujian Pengguna	59
4.3	Tabel Pengujian Terhadap Versi Android	60
4.4	Tabel Pengujian Terhadap Resolusi Layar	60

BAB I

PENDAHULUAN

.1 Latar Belakang

Di jaman yang serba modern sekarang ini banyak inovasi teknologi baru dan juga banyak terdapat banyak program penunjang yang bisa digunakan oleh berbagai kalangan masyarakat baik dari anak kecil maupun orang tua. Contoh nyatanya adalah pemanfaatan aplikasi android dan inovasi dari aplikasi android.

Android merupakan generasi smartphone dimana aplikasi ini merupakan aplikasi yang memiliki berbagai macam fitur pendukung yang memudahkan kita dalam pemanfaatan aplikasi ini dengan baik dan benar.

Tidak hanya menjadi sistem operasi smartphone, saat ini android juga menjadi pesaing utama dari Apple pada sistem operasi Tablet PC. Pesatnya pertumbuhan android karena Android itu sendiri adalah platform yang sangat lengkap baik itu sistem aplikasinya, aplikasi tool dan pengembangannya.

Android merupakan generasi baru dari platform mobile, platform yang memberikan pengembangan untuk melakukan pengembangan sesuai dengan yang diharapkan. Dengan pemilihan Android sebagai salah satu sistem informasi yang berbasis mobile dikarenakan perkembangan mobile phone di Indonesia berkembang dengan pesat.

Setiap pabrikan mobile phone menggunakan sistem operasi yang berbeda-beda, dan salah satunya adalah sistem operasi android. Dengan sistem operasi android yang bersifat open source maka pengguna sistem operasi android di Indonesia dapat memudahkan berinteraksi dengan mobile phone tersebut.

Android juga bisa digunakan di dalam pemanfaatan di dalam membuat aplikasi penataan buku dimana aplikasi ini juga bisa menampilkan tentang stok buku yang ada dalam perpustakaan. Aplikasi ini berfungsi sebagai alat pembantu petugas perpustakaan untuk mengecek sisa buku yang ada di dalam perpustakaan dan juga membantu petugas perpustakaan untuk melihat siapa yang meminjam buku dan judul buku yang di pinjam.

1.2 Rumusan Masalah

Berdasarkan dari uraian Rumusan Masalah, maka dapat di rumuskan masalah sebagai berikut :

1. Membuat aplikasi perpustakaan berbasis android.
2. Bagaimana cara membuat aplikasi android.
3. Bagaimana cara membuat aplikasi stok buku
4. Bagaimana membuat aplikasi yang bisa menghubungkan antara Android dengan database.

1.3 Batasan Masalah

Batasan masalah pembahasan yang diambil agar sesuai dengan tujuan dan tidak terjadi penyimpangan maksud dan tujuan utama, maka ditentukan ruang lingkup pembahasan sebagai berikut:

1. Pembuatan server sendiri dengan menggunakan XAMMP.
2. Menggunakan XAMMP sebagai database sever.
3. Mempersiapkan client.
4. Penanganan tampilan.
5. Melakukan paket konfigurasi.

1.4 Tujuan

Adapun yang menjadi tujuan penulisan dalam penyusunan Skripsi adalah sebagai berikut

1. Memudahkan pengguna dalam melakukan simpan pinjam buku dan pengecekan stock buku yang ada di dalam perpustakaan.
2. Merancang Aplikasi Implementasi Aplikasi android dalam sistem informasi senayan di perpustakaan ITN Malang kampus 2 yang berbasiskan *Android* yang berguna sebagai alat mempermudah dalam melakukan pengecekan buku oleh Petugas perpustakaan.
3. Merancang aplikasi ini untuk melakukan penyimpanan data di dalam database server. Di mana data base tersebut di buat menggunakan XAMMP.
4. Merancang aplikasi ini agar bisa diakses dengan menggunakan konfigurasi internet.

1.5 Metodologi Penelitian

Pada pembuatan Skripsi ini terdapat beberapa tahapan yang dilakukan yaitu tahap studi literatur, analisa kebutuhan Sistem dan eksperimen dan evaluasi.

A. Studi Literatur

Pengumpulan data dilakukan dengan cara mencari bahan referensi dari berbagai sumber sebagai landasan teori yang berhubungan dengan permasalahan yang dijadikan objek penelitian

B. Analisa Kebutuhan

Data dan informasi yang telah diperoleh akan dianalisa agar didapatkan suatu kerangka yang digunakan untuk acuan perancangan system pembuatan program.

C. Eksperimen dan evaluasi

Pada tahap ini, system yang telah selesai dibuat akan diuji coba, berdasarkan fungsional program dan akan dilakukan koreksi.

1.6 Sistematika Penulisan

Adapun sistematika penulisan skripsi ini terdiri atas :

BAB I : PENDAHULUAN

Pada bab pendahuluan berisi latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat penelitian, metode penelitian dan sistematika penulisan laporan skripsi.

BAB II : LANDASAN TEORI

Bab landaasan teori ini berisi tentang landasan teori mengenai permasalahan yang berhubungan dengan program yang akan dikerjakan.

BAB III : PERANCANGAN SISTEM

Membahas tentang anilisis dan perancangan system aplikasi.

BAB IV : IMPLEMENTASI DAN PENGUJIAN

Berisi tentang implementasi dan ujicoba dari aplikasi.

BAB V : KESIMPULAN DAN SARAN

Berisi tentang kesimpulan dan saran-saran yang digunakan untuk pengembangan program selanjutnya.

BAB II

LANDASAN TEORI

Pada bab ini akan di jelaskan mengenai teori – teori yang berhubungan dengan pembuatan aplikasi ini, sehingga dapat dijadikan sebagai landasan berpikir dan akan mempermudah dalam hal pembahasan hasil utama pada bab berikutnya. Adapun teori – teori tersebut mencakup pengertian tentang Android, Java, XML. ^[1]

2.1 Android

2.1.1 Teori tentang Android

Android adalah sistem operasi untuk telepon seluler yang berbasis linux. Android menyediakan *platform* yang bersifat *open source* bagi para pengembang untuk menciptakan sebuah aplikasi. Awalnya, Google In. mengakusisi Android Inc. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, yaitu konsorsium dari 34 perusahaan *hardware*, *software*, dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia. ^[1]

Telepon pertama yang memakai yang memakai sistem operasi Android adalah HTC Dream, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2009 diperkirakan di dunia ini paling sedikit terdapat 18 jenis telepon seluler yang menggunakan Android. ^[1]

2.1.2 Versi Android

2.1.2.1 Android Versi 1.1

Pada 9 Februari 2009, Google merilis Android versi 1.1. Android versi ini dilengkapi dengan pembaharuan: ^[1]

- i. Estetis pada aplikasi.
- ii. Jam alarm.
- iii. *Voice search* (pencarian suara).
- iv. Pengiriman pesan dengan Gmail.
- v. Pemberitahuan *email*.

2.1.2.2 Android Versi 1.5 (Cupcake)

Pada 30 April 2009, Google kembali merilis telepon seluler dengan menggunakan Android dan SDK (*Software Development Kit*) dengan versi 1.5 (Cupcake) dibangun di atas Linux Kernel 2.6.27. Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini adalah ^[1]:

- i. Kemampuan merekam dan menonton video dengan modus kamera.
- ii. Mengunggah *video* ke *youtube* dan gambar ke Picasa langsung dari telepon.
- iii. Dukungan *Bluetooth A2DP*.
- iv. Kemampuan terhubung secara otomatis ke *headset Bluetooth*, animasi layar.

2.1.2.3 Android Versi 1.6 (Donut)

Donut (versi 1.6) dibangun di atas Linux kernel 2.6.29 dirilis pada 15 September 2009 dengan beberapa pembaharuan yaitu: ^[1]

- i. Menampilkan proses pencarian yang lebih baik dibanding sebelumnya;
- ii. Penggunaan baterai indicator dan control *applet VPN*.
- iii. Galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus.
- iv. Kamera, *camcorder* dan galeri yang diintegrasikan.
- v. CDMA/EVDO, 802.1x, VPN, *gestures*, dan *text-to-speech engine*.
- vi. Kemampuan *dial contact*.
- vii. Teknologi *text to change speech*.
- viii. Pengaturan resolusi VWGA

2.1.2.4 Android Versi 2.0/2.1 (Éclair)

Pada tanggal 26 Oktober 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (Éclair) dibangun di atas Linux kernel 2.6.29, perubahan yang dilakukan adalah: ^[1]

- i. Peningkatan *googlemaps 3.1.2*.
- ii. Perubahan UI dengan browser baru dan dukungan HTML 5.
- iii. Daftar kontak yang baru.
- iv. Dukungan flash untuk kamera 3,2 MP.
- v. Digital zoom, dan Bluetooth 2.1.

2.1.2.5 Android Versi 2.2 Froyo (Frozen Yoghurt)

Android 2.2 atau yang biasa dikenal dengan *Froyo* dirilis pada tanggal 20 Mei 2010, *Froyo* dibangun di atas Linux kernel 2.6.32, *Froyo* memang merupakan versi terbaru dari sistem operasi Android yang telah dirilis oleh google untuk melengkapi versi terdahulu. Walaupun secara resmi telah dirilis google, namun tidak semua ponsel Android dapat menggunakan *Froyo*. Pengguna masih harus menunggu notifikasi resmi yang dikeluarkan masing-masing vendor ponsel. Berikut ini adalah peningkatan performa dari Android 2.2 *Froyo*: ^[1]

- i. Peningkatan performa meningkat hingga dua kali dari sistem sebelumnya (Éclair). Pengujian kinerja processor dalam mengolah multimedia, hingga kemampuan grafis untuk menangani konten 3D.
- ii. *Free memory* yang ada juga lebih besar dari sebelumnya. Jika biasanya pengguna hanya mendapatkan sekitar 100MB, kini dapat menggunakan sekitar 250MB dari total 512MB *memory* yang ada. Otomatis hal tersebut makin meningkatkan performa meski pengguna menjalankan beragam aplikasi sekaligus.
- iii. Perubahan lain dari HTC melalui sistem operasi. *Froyo Desire* adalah dapat meletakkan aplikasi di sd card berbeda dengan sistem operasi terdahulu yang hanya dapat meletakkan semua aplikasi pada memory utama. Dengan sistem operasi *Froyo*, pengguna dapat meletakkan seluruh file instalasi pada memory eksternal ^[1].

- iv. Merekam video dengan kualitas HD. Jika sebelumnya pengguna hanya dapat merekam gambar bergerak pada resolusi maksimal 800x280 pixel, kini dengan froyo, resolusi pengambilan video dapat ditingkatkan hingga 1280x720 pixel yang setara dengan kualitas *High Definition*.
- v. Setelah upgrade ke Froyo, pengguna kan menemukan *icon* baru pada deratan aplikasi yang ada yaitu Wi-fi Hotspot. Seperti namanya, aplikasi ini memungkinkan ponsel pengguna dijadikan access point.
- vi. Selain itu masih ada lagi aplikasi tambahan seperti Flashlight, App Sharing dan Navigation. Khusus untuk navigasi peta, hanya tersedia dalam versi beta dan belum dapat digunakan di beberapa lokasi.

2.1.2.6 Android Versi 2.3 (Gingerbread)

Pada tanggal 6 Desember 2010 Google merilis Android 2.3 dengan sebutan Gingerbread, dibangun di atas Linux Kernel 2.6.35 dengan beberapa pembaharuan sebagai berikut: ^[1]

- i. Perubahan UI.
- ii. Mendukung ukuran layar WXGA.
- iii. Mendukung native SIP VoIP.
- iv. Mendukung WebM/VPS playback vide., dan AAC audio encoding.
- v. Audio efek baru seperti reverb, equalization, headphone virtualization, dan bass boost.
- vi. Peningkatan grafis, audio dan input untuk pengembang game.
- vii. Mendukung Near Field Communication (NFC).
- viii. Peningkatan fungsi copy-paste.

Tidak semua perangkat dapat di upgrade ke versi 2.3 in. spesifikasi minimum agar dapat di upgrade ke versi Gingerbread adalah CPU 1 Ghz, Ram 512MB, diagonal layar minimal 3.5” ^[1]

2.1.2.7 Android versi 3.0/3.1 (Honeycomb)

Android Honeycomb dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. User interface pada Honeycomb juga berbeda karena sudah didesain untuk tablet. Honeycomb juga mendukung multi processor dan juga akselerasi perangkat keras (hardware) untuk grafis. Tablet pertama yang dibuat dengan menjalankan Honeycomb adalah Motorola Xoom^[1].

2.1.2.8 Android versi 4.0 ICS (Ice Cream Sandwich)

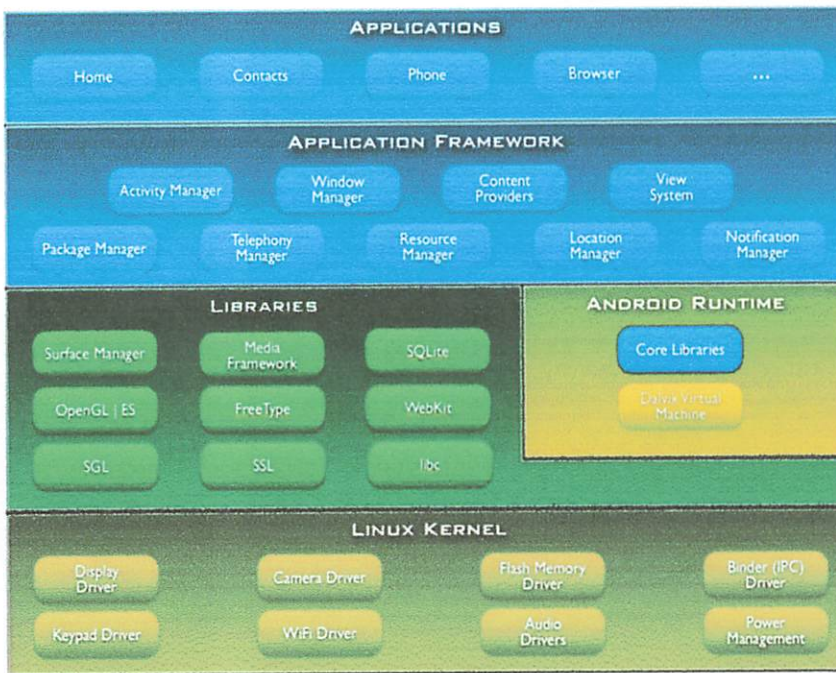
Diumumkan pada tanggal 10 Oktober 2011 membawa fitur Honeycomb untuk smartphone dan menambahkan fitur baru termasuk membuka kunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan control, terpadu kontak jaringan social, perangkat tambahan fotografi, mencari email secara offline, dan berbagi informasi dengan menggunakan NFC.^[1]

2.1.2.9 Android versi 4.1 (Jelly Bean)

Android Jelly Bean diluncurkan dengan membawa sejumlah keunggulan dan fitur baru. Penambahan baru diantaranya meningkatkan input keyboard, desain baru fitur pencarian, UI yang baru dan pencarian melalui *Voice Search* yang lebih cepat. Tak ketinggalan *Google Now* juga menjadi bagian yang diperbarui. *Google Now* memberikan informasi yang tepat pada waktu yang tepat pula. Salah satu kemampuannya adalah dapat mengetahui informasi cuaca, lalu-lintas, ataupun hasil pertandingan olahraga. OS Android Jelly Bean 4.1 muncul pertama kali dalam produk tablet Asus, yakni Google Nexus 7.^[1]

2.2 Arsitektur Android

Secara sederhana arsitektur android merupakan sebuah kernel linux dan sekumpulan pustaka (*library*) C / C++ dalam suatu framework yang menyediakan dan mengatur alur proses aplikasi. Dalam paket sistem operasi android terdiri dari beberapa unsure seperti tampak pada gambar 2.1. ^[2]



Gambar 2.1 Arsitektur Android^[2]

Dalam arsitektur android, yang seperti tampak pada gambar 2.1 terdapat 5 unsur antara lain :^[2]

a. Linux Kernel

Android dibangun di atas kernel Linux 2.6. Namun secara keseluruhan android bukanlah linux, karena dalam android tidak terdapat paket standar yang dimiliki oleh linux lainnya. Linux merupakan sistem operasi terbuka yang handal dalam manajemen memori dan proses. Oleh karenanya pada android hanya terdapat beberapa servis yang diperlukan seperti keamanan, manajemen memori, manajemen proses, jaringan dan driver. Kernel linux menyediakan driver layar, kamera, keypad, WiFi, Flash Memory, audio, dan IPC (Interprocess Communication) untuk mengatur aplikasi dan lubang keamanan.^[2]

b. Libraries

Android menggunakan beberapa paket pustaka yang terdapat pada C/C++ dengan standar Berkeley Software Distribution (BSD) hanya setengah dari yang aslinya untuk tertanam pada kernel Linux. Beberapa pustaka diantaranya:^[2]

- i. Surface Manager untuk mengatur hak akses layer dari berbagai aplikasi.
- ii. Graphic Library termasuk didalamnya SGL dan OpenGL, untuk tampilan 2D dan 3D.
- iii. SQLite untuk mengatur relasi database yang digunakan pada aplikasi.
- iv. SSL dan WebKit untuk browser dan keamanan internet.

Pustaka-pustaka tersebut bukanlah aplikasi yang berjalan sendiri, namun hanya dapat digunakan oleh program yang berada di level atasnya. Sejak versi Android 1.5, pengembang dapat membuat dan menggunakan pustaka sendiri menggunakan Native Development Toolkit (NDK).^[2]

c. Android Runtime

Pada android tertanam paket pustaka inti yang menyediakan sebagian besar fungsi android. Inilah yang membedakan Android dibandingkan dengan sistem operasi lain yang juga mengimplementasikan Linux. Android Runtime merupakan mesin virtual yang membuat aplikasi android menjadi lebih tangguh dengan paket pustaka yang telah ada. Dalam Android Runtime terdapat 2 bagian utama, diantaranya: ^[2]

- i. **Pustaka Inti**, android dikembangkan melalui bahasa pemrograman Java, tapi Android Runtime bukanlah mesin virtual Java. Pustaka inti android menyediakan hampir semua fungsi yang terdapat pada pustaka Java serta beberapa pustaka khusus android.
- ii. **Mesin Virtual Dalvik**, Dalvik merupakan sebuah mesin virtual yang dikembangkan oleh Dan Bornstein yang terinspirasi dari nama sebuah perkampungan yang berada di Iceland. Dalvik hanyalah interpreter mesin virtual yang mengeksekusi file dalam format Dalvik Executable (*.dex). Dengan format ini Dalvik akan mengoptimalkan efisiensi penyimpanan dan pengalamanan memori pada file yang dieksekusi. Dalvik berjalan di atas kernel Linux 2.6, dengan fungsi dasar seperti threading dan manajemen memori yang terbatas. ^[2]

d. Application Framework

Kerangka aplikasi menyediakan kelas-kelas yang dapat digunakan untuk mengembangkan aplikasi android. Selain itu, juga menyediakan abstraksi generik untuk mengakses perangkat, serta mengatur tampilan user interface dan sumber daya aplikasi. Bagian terpenting dalam kerangka aplikasi android adalah sebagai berikut : ^[2]

- i. **Activity Manager**, berfungsi untuk mengontrol siklus hidup aplikasi dan menjaga keadaan "Backstack" untuk navigasi penggunaan.
- ii. **Content Providers**, berfungsi untuk merangkum data yang memungkinkan digunakan oleh aplikasi lainnya, seperti daftar nama.
- iii. **Resource Manager**, untuk mengatur sumber daya yang ada dalam program. Serta menyediakan akses sumber daya diluar kode program, seperti karakter, grafik, dan file layout.
- iv. **Location Manager**, berfungsi untuk memberikan informasi detail mengenai lokasi perangkat android berada.
- v. **Notification Manager**, mencakup berbagai macam peringatan seperti, pesan masuk, janji, dan lain sebagainya yang akan ditampilkan pada status bar.

e. Application Layer

Puncak dari diagram arsitektur android adalah lapisan aplikasi dan widget. Lapisan aplikasi merupakan lapisan yang paling tampak pada pengguna ketika menjalankan program. Pengguna hanya akan melihat program ketika digunakan tanpa mengetahui proses yang terjadi dibalik lapisan aplikasi. Lapisan ini berjalan dalam Android runtime dengan menggunakan kelas dan service yang tersedia. Lapisan aplikasi android sangat berbeda dibandingkan dengan sistem operasi lainnya. Pada android semua aplikasi, baik aplikasi inti (native) maupun aplikasi pihak ketiga berjalan diatas lapisan aplikasi dengan menggunakan pustaka API (Application Programming Interface) yang sama. [2]

2.3 Komponen Android

Aplikasi Android ditulis dalam bahasa pemrograman Java. Java mengompilasi kode bersama dengan data resource dan file yang dibutuhkan oleh aplikasi dibundel ke dalam paket Android. Dan hasil dari arsip kompilasi kode-kode tersebut berekstensi file .apk. Ada 4 komponen utama yang terdapat pada aplikasi di Android, yaitu :^[2]

- a. Activity yaitu sebuah *user interface* atau UI yang disajikan untuk pengguna dapat melakukan interaksi.
- b. Service yaitu komponen yang tidak memiliki *user interface*. Akan tetapi berjalan pada background atau latar belakang.
- c. Broadcast Receiver merupakan komponen yang berfungsi menerima dan beraksi untuk menyampaikan pesan atau notifikasi.
- d. Content Provider merupakan komponen terakhir yang berfungsi sebagai sarana penyedia konten atau membuat kumpulan aplikasi data secara spesifik agar bisa digunakan oleh aplikasi lain.

2.4 Eclipse Indigo

Eclipse Indigo, adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (*platform-independent*). Berikut ini adalah sifat dari Eclipse:^[2]

- i. Multi-platform : Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
- ii. 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 lain sebagainya.
- iii. 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 selalu dilengkapi dengan JDK (Java Development Kit), plug-in yang membuat Eclipse kompatibel untuk mengembangkan program Java, dan PDE (Plug-in Development Environment) untuk mengembangkan plug-in baru. Eclipse beserta plug-innya diimplementasikan dalam bahasa pemrograman Java. Konsep Eclipse adalah IDE yang terbuka (open), mudah diperluas (extensible) untuk apa saja, dan tidak untuk sesuatu yang spesifik. Jadi, eclipse tidak saja untuk mengembangkan program Java, akan tetapi dapat digunakan untuk berbagai macam keperluan, cukup dengan menginstal plug-in yang dibutuhkan. JDK dibutuhkan karena Android adalah bahasa yang berbasis Java. Versi minimum yang dianjurkan adalah JDK 5 atau JDK 6. Selain JDK, untuk pemrograman Android diperlukan Android SDK Tools. Selain Android SDK Tools, diperlukan ADT (Android Development Tools) plugin. ADT plugin ini berfungsi agar Eclipse bisa menggunakan Android SDK. ^[2]

2.5 Android Virtual Device

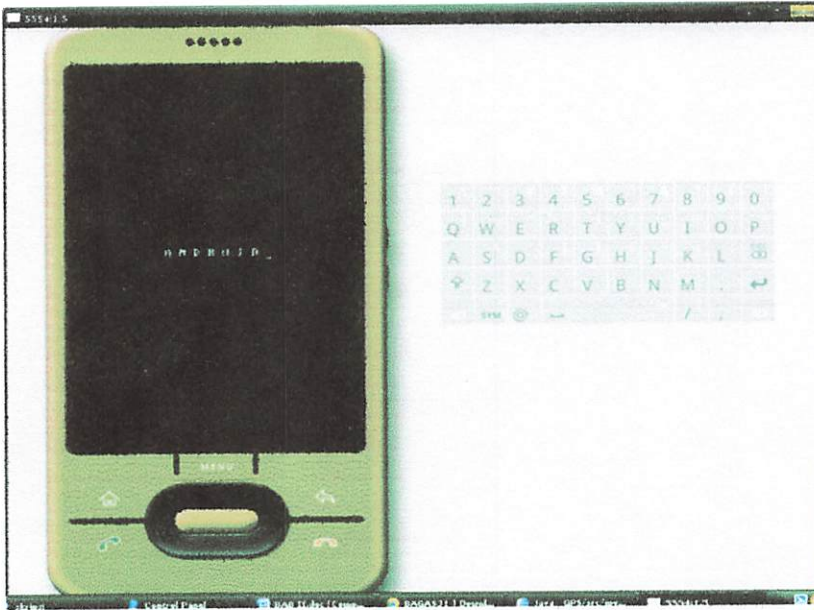
Android emulator pada eclipse sepenuhnya di kendalikan oleh AVD (Android Virtual Device) Manager. AVD manager memungkinkan seorang programmer untuk menjalankan program Android pada Eclipse yang mengemulasikan smartphone atau tablet Android, seperti tampak dalam gambar 2.2. Cara ini sangat membantu programmer karena programmer bisa menguji coba program android pada berbagai tipe perangkat Android tanpa harus menggunakan perangkat yang sesungguhnya. ^[2]



Gambar 2.5 Menampilkan Android Virtual Device Manager ^[2]

2.6 Android Emulator

Setelah menambahkan tipe perangkat android pada AVD Manager, maka kita bisa menjalankan program android kita pada Android Emulator, seperti pada gambar 2.3. ^[2]



Gambar 2.6 Menampilkan Android Emulator pada Eclipse. ^[2]

2.7 Java

2.7.1 Sejarah Java

Bahasa pemrograman Java terlahir dari The Green Project, yang berjalan selama 18 bulan, dari awal tahun 1991 hingga musim panas 1992. Proyek tersebut belum menggunakan versi yang dinamakan Oak. Proyek ini dimotori oleh Patrick Naughton, Mike Sheridan, James Gosling dan Bill Joy, beserta sembilan pemrogram lainnya dari Sun Microsystems. Salah satu hasil proyek ini adalah maskot *Duke* yang dibuat oleh Joe Palrang.

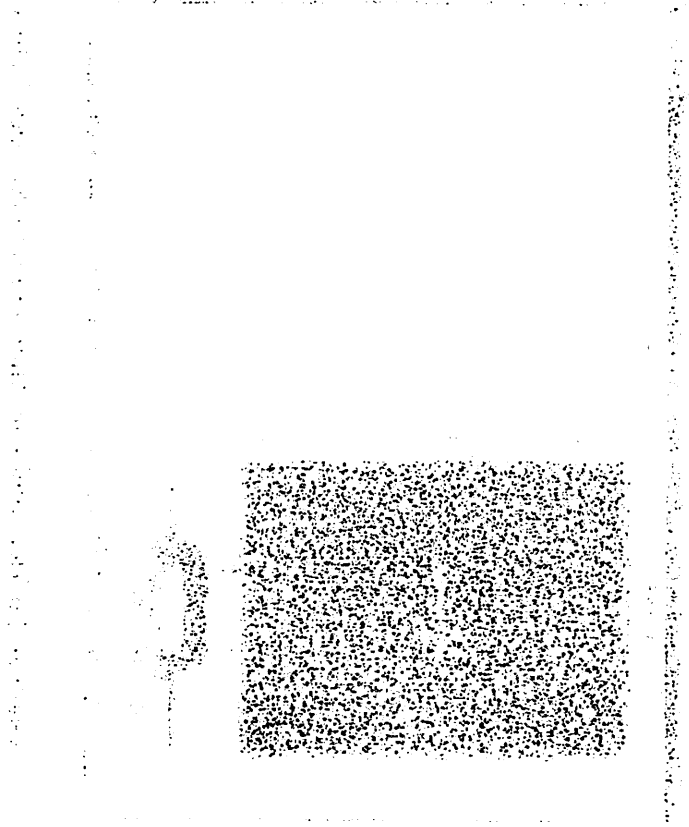
Pertemuan proyek berlangsung di sebuah gedung perkantoran *Sand Hill Road* di Menlo Park. Sekitar musim panas 1992 proyek ini ditutup dengan menghasilkan sebuah program *Java Oak* pertama, yang ditujukan sebagai pengendali sebuah peralatan

... ..
... ..
... ..

... ..
... ..
... ..

... ..

... ..



... ..
... ..

... ..

dengan teknologi layar sentuh (*touch screen*), seperti pada PDA sekarang ini. Teknologi baru ini dinamai "7" (*Star Seven*).

Setelah era *Star Seven* selesai, sebuah anak perusahaan Tv kabel tertarik ditambah beberapa orang dari proyek The Green Project. Mereka memusatkan kegiatannya pada sebuah ruangan kantor di 100 Hamilton Avenue, Palo Alto.^[7]

Perusahaan baru ini bertambah maju: jumlah karyawan meningkat dalam waktu singkat dari 13 menjadi 70 orang. Pada rentang waktu ini juga ditetapkan pemakaian *Internet* sebagai *medium* yang menjembatani kerja dan ide di antara mereka. Pada awal tahun 1990-an, *Internet* masih merupakan rintisan yang dipakainya dikalangan akademisi dan militer.

Mereka menjadikan perambah (*browser*) *Mosaic* sebagai landasan awal untuk membuat perambah Java pertama yang dinamai *Web Runner*, terinspirasi dari film 1980-an, *Blade Runner*. Pada perkembangan rilis pertama, *Web Runner* berganti nama menjadi *Hot Java*.

Pada sekitar bulan Maret 1995, untuk pertama kali kode sumber Java versi 1.0a2 dibuka. Kesuksesan mereka diikuti dengan pemberitaan pertama kali pada surat kabar *San Jose Mercury News* pada tanggal 23 Mei 1995. Sayangnya terjadi perpecahan di antara mereka suatu hari pada pukul 04.00 di sebuah ruangan hotel *Sheraton Palace*. Tiga dari pimpinan utama proyek, *Eric Schmidt* dan *George Paolini* dari *Sun Microsystems* bersama *Marc Andreessen*, membentuk *Netscape*.^[7]

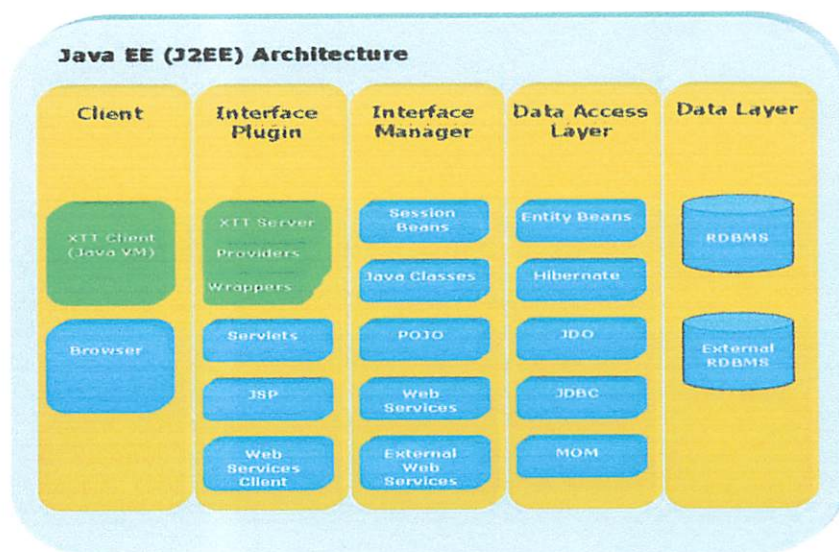
Nama *Oak*, diambil dari pohon oak yang tumbuh di depan jendela ruangan kerja "Bapak Java", *James Gosling*. Nama *Oak* ini tidak dipakai untuk versi release Java karena sebuah perangkat lunak lain sudah terdaftar dengan merek dagang tersebut, sehingga diambil nama penggantinya menjadi "Java". Nama ini diambil dari kopi murni yang digiling langsung dari biji (kopi tubruk) kesukaan *Gosling*. Konon kopi ini berasal dari Pulau Jawa. Jadi nama bahasa pemrograman Java tidak lain berasal dari kata Jawa (bahasa Inggris untuk Jawa adalah *Java*).^[7]

2.7.2 Java Arsitektur

Rancangan arsitektur processor Java berfungsi sebagai Java Virtual Machine (JVM). Hardware ini berupa processor Java yang akan mengeksekusi Java Instruction Set secara langsung tanpa perlu ada tahapan konversi ke native instruction set sebagaimana yang terjadi bila aplikasi Java tersebut dijalankan dengan general processor. Diharapkan waktu eksekusi Java menjadi lebih pendek. Rancangan mempertimbangkan karakteristik bahasa Java untuk memperoleh proses method calls, mempercepat context switching dan handling errors saat terjadi interupsi dan exception, dukungan kecepatan eksekusi Java Instruction Set yang beroperasi di stack, serta kemudahan dalam melakukan proses rekursi. Hardware diharapkan tidak kompleks sehingga bisa lebih murah dan konsumsi hardware stack khusus yang digunakan untuk subroutine return address, parameter passing dan tempat penyimpanan sementara semua variabel local.^[7]

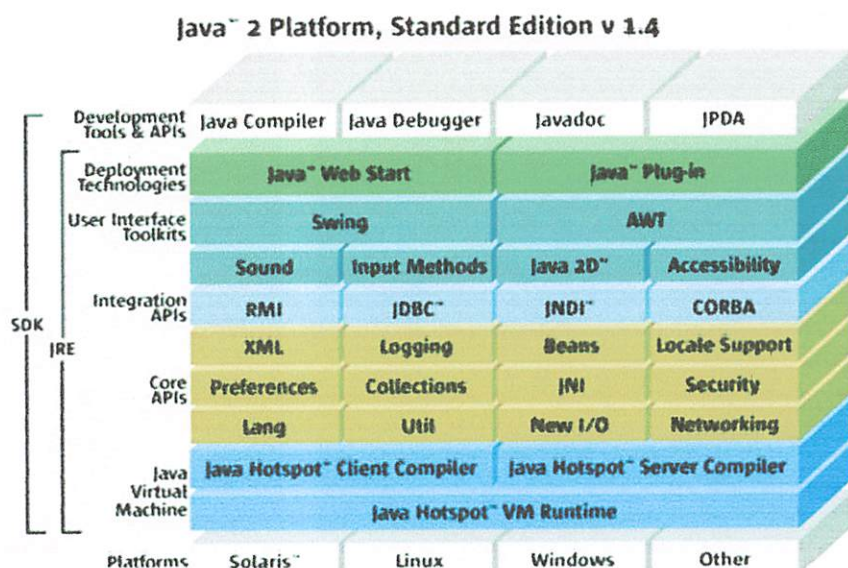
Sun membagi arsitektur Java menjadi tiga bagian, yaitu:^[7]

- a. Enterprise Java (J2EE) untuk aplikasi berbasis web, aplikasi sistem tersebar dengan beraneka ragam klien dengan kompleksitas yang tinggi. Dan juga merupakan superset dari Standar Java. J2EE memiliki arsitektur tersendiri, seperti dalam gambar 2.4.



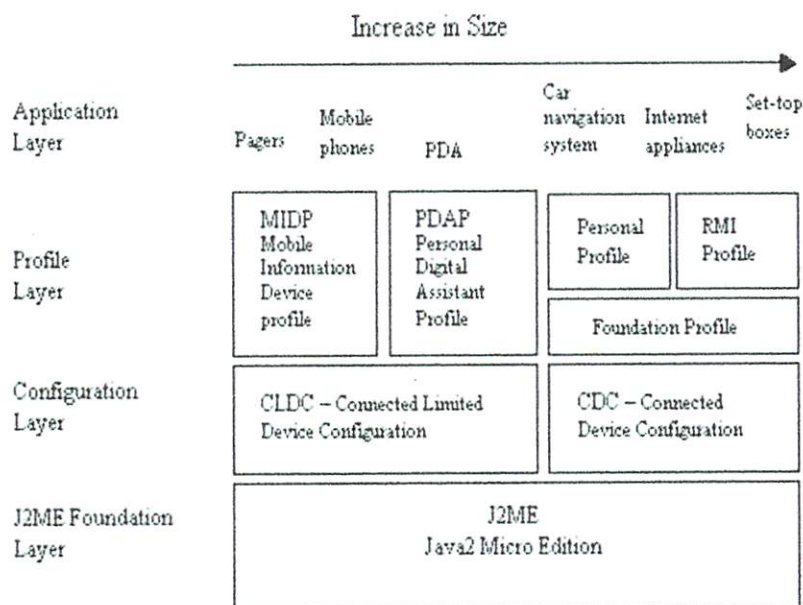
Gambar 2.7. Menampilkan Arsitektur J2EE^[7]

- b. Standar Java (J2SE), biasa dikenal sebagai bahasa Java dan J2SE ditujukan untuk aplikasi java berbasis desktop. Arsitektur J2SE seperti dalam gambar 2.5.



Gambar 2.8 Menampilkan Arsitektur J2SE [7]

- c. Micro Java (J2ME) merupakan subset dari J2SE dan salah satu aplikasinya yang banyak dipakai adalah untuk *wireless / mobile device*, seperti tampak dalam gambar 2.6.



Gambar 2.9 Menampilkan Arsitektur J2ME [7]

2.7.3 Fitur-fitur dalam Java

a. Sederhana

Banyak sistem pemrograman memiliki puluhan cara untuk mengerjakan satu hal. Jika suatu bahasa cukup terbuka sehingga orang dapat melihat lebih mendalam, kita bebas dan dapat melakukan apa saja dengan cara kita. Mungkin cara ini menawarkan kinerja yang sangat memuaskan bagi programmer yang ahli dan teliti., tetapi hanya ada sedikit cara untuk menyelesaikan suatu pekerjaan. Java member anda kemampuan untuk menuangkan ide anda dalam cara berorientasi objek yang mudah dan jelas., tanpa harus melihat proses bagian dalam sistem yang sering kali berbahaya. ^[7]

Java dirancang agar mudah dipelajari dan digunakan secara efektif. Java tidak menyediakan fitur-fitur rumit pemrograman level tinggi, serta banyak pekerjaan pemrograman yang mulanya harus dilakukan manual, sekarang dapat dikerjakan oleh Java secara otomatis. Java tidak mendukung fitur-fitur rumit berikut ini: ^[7]

- i. Explicit pointer manipulation
- ii. Implicit type casting
- iii. Structure atau Unions
- iv. Operator Overloading
- v. Templates
- vi. Header Files
- vii. Multiple Inheritance

b. Bahasa Berorientasi Objek (OOP)

Meskipun Java dipengaruhi bahasa-bahasa terdahulu, namun Bahasa Java bukan turunan langsung dari bahasa manapun, dan juga tidak dirancang untuk kompatibel secara kode sumber dengan bahasa-bahasa sebelumnya. Bahasa Java sepenuhnya baru. Kebijakan ini member kebebasan perancang Java. Kebebasan ini menghasilkan rancangan Bahasa Java yang bersih, berguna, serta sekaligus menerapkan pendekatan pragmatis terhadap objek^[7].

Object Oriented Programming (OOP) adalah cara ampuh dalam perkembangan dan pengorganisasian perangkat lunak. Pada OOP, program computer sebagai sekelompok objek yang berinteraksi. Deskripsi ringkas OOP adalah mengorganisasikan program sebagai kumpulan komponen yang disebut objek. Objek-objek ini ada secara independen, mempunyai aturan-aturan berkomunikasi dengan objek lain dan untuk memerintahkan objek lain untuk meminta informasi tertentu atau meminta objek lain mengerjakan sesuatu.^[7]

Kebanyakan sistem berorientasi objek lain memilih hirarki objek yang kaku dan susah diatur, atau memilih menggunakan model objek dinamik yang tidak memiliki kinerja yang tinggi dan kelengkapan. Java sekali lagi memiliki keseimbangan, menyediakan mekanisme pembagian class sederhana, dengan model antarmuka dinamik yang intuitif hanya jika diperlukan. Java tidak mengijinkan pewarisan jamak namun menyelesaikan kebutuhan pewarisan jamak dengan fasilitas yang lebih elegan.^[7]

c. Bahasa Statically Typed

Seluruh objek dalam program harus dideklarasikan lebih dahulu sebelum digunakan. Ini memungkinkan kompilator Java menentukan dan melaporkan terjadinya pertentangan (ketidakkompatibelan) tipe yang merupakan barikade awal untuk mencegah terjadinya kesalahan yang tidak perlu. Kelebihan dari fitur ini adalah kode program dapat dioptimasi untuk menghasilkan program berkinerja tinggi.^[7]

d. Bahasa Dikompilasi

Sebelum menjalankan program di bahasa Java, program dikompilasi menggunakan Java compiler. Kompilasi akan menghasilkan file bytecode yang serupa fungsinya dengan file kode mesin dan dapat dieksekusi disembarang Java Interpreter. Java Interpreter membaca file “bytecode” dan menerjemahkan perintah “bytecode” menjadi perintah-perintah bahasa mesin yang dapat dieksekusi.^[7]

e. Bahasa yang Aman

Salah satu prinsip kunci perancangan Java adalah keselamatan dan keamanan. Bahasa Java telah memperoleh aspek keamanan dari Java Applet. Cara kerja Applet sebagai berikut, saat web browser mengetahui adanya Java Applet di halaman web yang sedang diinterpretasi maka browser akan melakukan download applet dan keseluruhan halaman web. Applet itu kemudian dijalankan dikomputer dimana browser berada. Hal ini tentu dapat menjadi sangat berbahaya, banyak hal buruk yang dapat dilakukan applet terhadap komputer yang mengeksekusinya. Untuk itu Java telah member pengamanan terhadap applet dengan model pengamanan berlapis. Java menggunakan model pengaman tiga lapis (*three-layer security model*) untuk melindungi system dari untrusted Java code.^[7]

- i. *Bytecode* verifier membuka bytecode sebelum dijalankan dan menjamin bytecode memenuhi aturan-aturan dasar bahasa Java.
- ii. *Class Loader* menangani pemuatan kelas Java ke *runtime interpreter*.
- iii. Manajer keamanan menangani keamanan tingkat aplikasi dengan mengendalikan apakah program berhak mengakses sumber daya seperti sistem file, port, jaringan, proses eksternal dan sistem windowing.

Java juga menyediakan beragam teknik pengamanan, yaitu:

- i. Java tidak mengenal pointer. Peniadaan pointer merupakan langkah besar pengamanan. Bagi pemrogram handal, operasi pointer merupakan anugrah besar untuk optimasi program dan pembuatan program yang efisien serta mengagumkan. Namun juga dapat menjadi petaka yang besar, pointer merupakan sarana luar biasa untuk pengaksesan tak diotorisasi. Dengan peniadaan operasi pointer, Java dapat menjadi bahasa yang lebih aman.^[7]
- ii. Program Java dikompilasi menjadi serangkaian bytecode. Sebelum Java dijalankan, terdapat Java verifier yang memeriksa bytecode untuk menjamin ketiadaan kode yang mencurigakan.^[7]
- iii. Java mempunyai beberapa pengamanan terhadap applet. Untuk mencegah program bertindak mengganggu disk pemakai applet tidak diperbolehkan melakukan open, read, atau write terhadap file di sistem pemakai secara *random*. Karena Java applet dapat membuka jendela browser yang baru, maka jendela mempunyai Java logo dan teks indentifikasi terhadap jendela yang dibuka. Hal ini mencegah jendela pop-up menipu sebagai yang lain seperti untuk meminta nama pemakai dan password. Java telah menerapkan keamanan yang ketat namun fleksibel. Keamanan berdasar suatu file kebijakan sehingga dapat diatur dan dikendalikan untuk memperoleh kemampuan maksimal bagi program mandiri dan applet yang dipercaya (*trusted code*).^[7]

f. Bahasa Independen terhadap Platform

Platform independence adalah kemampuan program bekerja di sistem operasi atau sistem komputer berbeda. Bahasa Java merupakan bahasa yang secara sempurna tidak bergantung pada platform. Tipe variabel Java mempunyai ukuran yang sama di semua platform sehingga variabel bertipe integer (int, long) berukuran sama tidak peduli dimana program Java dikompilasi. Juga terhadap penggunaan Java applet di web adalah sama sekali tidak memerlukan perubahan sedikit pun terhadap file .class yang dihasilkan agar dapat dieksekusi di platform

manapun. Begitu telah tercipta file .class dengan menggunakan kompilator Java di platform manapun, maka file .class tersebut juga dapat dijalankan di platform manapun.^[7]

g. Bahasa Multithreading

Multithreading adalah untuk menyatakan program komputer melakukan lebih dari satu tugas di satu waktu yang sama.^[7]

h. Bahasa yang kuat

Program Java melakukan garbage collection yang berarti pemrogram tidak perlu menghapus sendiri objek-objek yang tidak digunakan lagi. Fasilitas ini mengurangi beban pengelolaan memori oleh pemrogram dan mengurangi atau mengeliminasi sumber kesalahan terbesar yang terdapat di bahasa yang memungkinkan kesalahan alokasi dinamis.^[7]

i. Bahasa yang mampu diperluas

Program Java mendukung native method, yaitu fungsi ditulis di bahasa lain, biasanya C/C++ dan Android. Dukungan native method memungkinkan pemrogram menulis fungsi yang dapat dieksekusi lebih cepat dibanding fungsi ekivalen di Java. Native method secara dinamis akan dihubungkan ke program Java, yaitu diasosiasikan dengan program. Bahasa mempunyai banyak fitur modern yang ampuh untuk pengelolaan sistem kompleks.^[7]

2.8 Eclipse

Eclipse awalnya dikembangkan oleh IBM untuk menggantikan perangkat lunak *IBM Visual Age for Java* 4.0. Produk ini diluncurkan oleh IBM pada tanggal 5 November 2001, yang menginvestasikan sebanyak US\$ 40 juta untuk pengembangannya. Semenjak itu konsorsium *Eclipse Foundation* mengambil alih untuk pengembangan *Eclipse* lebih lanjut dan pengaturan organisasinya. Sejak versi 3.0, *Eclipse* pada dasarnya merupakan sebuah kernel, yang mengangkat *plug-in*. Apa yang dapat digunakan di dalam *Eclipse* sebenarnya adalah fungsi dari *plug-in* yang sudah diinstal. Ini merupakan basis dari *Eclipse* yang dinamakan *Rich Client Platform (RCP)*. Berikut ini adalah komponen yang membentuk RCP: ^[7]

1. *Core platform*
2. *OSGi*
3. *SWT (Standard Widget Toolkit)*
4. *Jface*
5. *Eclipse Workbench*

Secara standar *Eclipse* selalu dilengkapi dengan *JDT (Java Development Tools)*, *plug-in* yang membuat *Eclipse* kompatibel untuk mengembangkan program *Java*, dan *PDE (Plug-in Development Environment)* untuk mengembangkan *plug-in* baru. *Eclipse* beserta *plug-in*-nya diimplementasikan dalam bahasa pemrograman *Java*.

Konsep *Eclipse* adalah IDE yang terbuka (open), mudah diperluas (extensible) untuk apa saja, dan tidak untuk sesuatu yang spesifik. Jadi, *Eclipse* tidak saja untuk mengembangkan program *Java*, akan tetapi dapat digunakan untuk berbagai macam keperluan, cukup dengan menginstal *plug-in* yang dibutuhkan. Apabila ingin mengembangkan program *C/C++* terdapat *plug-in CDT (C/C++ Development Tools)*. Selain itu, pengembangan secara visual bukan hal yang tidak mungkin oleh *Eclipse*, *plug-in UML2* tersedia untuk membuat *Diagram UML*. Dengan menggunakan *PDE* setiap orang bisa membuat *plug-in* sesuai dengan keinginannya. Salah satu situs yang menawarkan *plug-in* secara gratis seperti *Eclipse downloads by project*. Tujuan dari sistem ini adalah untuk menyediakan distribusi *Eclipse* dengan fitur-fitur dan versi yang terstandarisasi. ^[7]

2.9 DFD

Data Flow Diagram (DFD) adalah representasi grafik dari sebuah sistem. DFD menggambarkan komponen-komponen sebuah sistem, aliran-aliran data di mana komponen-komponen tersebut, dan asal, tujuan, dan penyimpanan dari data tersebut.

Kita dapat menggunakan DFD untuk dua hal utama, yaitu untuk membuat dokumentasi dari sistem informasi yang ada, atau untuk menyusun dokumentasi untuk sistem informasi yang baru. Empat simbol yang digunakan. ^[5]

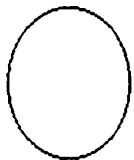
Notasi Yourdon /
DeMarco



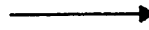
Notasi Gane &
Sarson



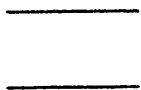
Simbol Entitas eksternal / Terminator menggambarkan asal atau tujuan data di luar sistem



Simbol lingkaran menggambarkan entitas atau proses dimana aliran data masuk ditransformasikan ke aliran data keluar.



Simbol aliran data menggambarkan aliran data


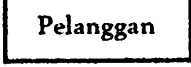
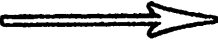
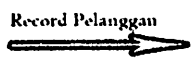




Simbol file menggambarkan tempat data di simpan

Gambar 2.11 Data Flow Diagram ^[5]

2.10 Diagram Konteks

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh input ke sistem atau output dari sistem. DFD akan memberi gambaran tentang keseluruhan sistem. Dalam diagram konteks hanya ada satu proses.^[5]

Simbol	Arti	Contoh
	Terminator/ Entitas	
	Aliran Data/ Data Flow	
	Proses	

Gambar 2.12 Simbol dalam Konteks Diagram^[5]

2.11 Entity Relationship Diagram (ERD)

Entity Relationship Diagram merupakan jaringan yang menggunakan susunan data yang disimpan dari system secara abstrak. Diagram Entity Relationship ini ditemukan oleh Chen tahun 1976. Tujuan dari Entity Relationship adalah untuk menunjukkan objek data dan relationship yang ada pada objek tersebut. Disamping itu Model ER ini merupakan salah satu alat untuk perancangan dalam basis data.^[5]

2.11.1 Komponen (Simbol) ERD

a. Entity

Adalah suatu objek yang dapat dibedakan atau dapat diidentifikasi secara unik dengan objek lainnya, dimana semua informasi yang berkaitan dengannya dikumpulkan. Kumpulan dari entity yang sejenis dinamakan Entity Set.^[5]

Simbol dari Entity :

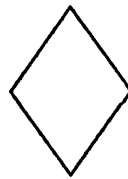


Gambar 2.13 Simbol Entity ^[5]

b. Relationship

Adalah hubungan yang terjadi antara satu entity dengan entity lainnya. Relationship tidak mempunyai keberadaan fisik atau konseptual kecuali yang sejenis dinamakan dengan Relationship Diagram. ^[5]

Simbol dari Relationship adalah :



Gambar 2.14 Simbol Relationship ^[5]

Contoh :



Gambar 2.15 Contoh Relationship diantara 2 entity ^[5]

Keterangan :

Memiliki adalah relationship set yang terbentuk antara entity Pegawai dengan entity Kendaraan.

c. Atribut

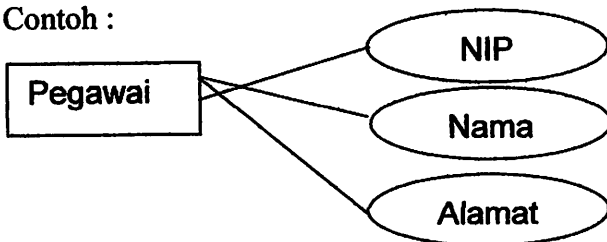
Adalah karakteristik dari entity atau relationship yang menyediakan penjelasan detail tentang entity atau relationship tersebut. ^[5]

Simbol dari Atribut adalah :



Gambar 2.16 Simbol Atribut ^[5]

Contoh :



Gambar 2.17 Contoh Atribut dari 1 Entity ^[5]

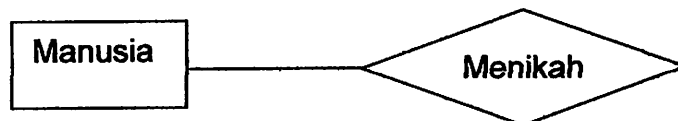
2.11.2 Derajat Relationship

ERD memiliki derajat relationship, antara lain : ^[5]

a. Unary (Derajat Satu)

Adalah satu buah relationship menghubungkan satu buah entity. ^[5]

Contoh :



Gambar 2.18 Sebuah Relationship Derajat 1 ^[5]

Keterangan :

Manusia menikah dengan manusia, relationship menikah hanya menghubungkan entity manusia.

b. Binary (Derajat Dua)

Adalah satu buah relationship yang menghubungkan dua buah entity. ^[5]

Contoh :



Gambar 2.19 Relationship Derajat 2 ^[5]

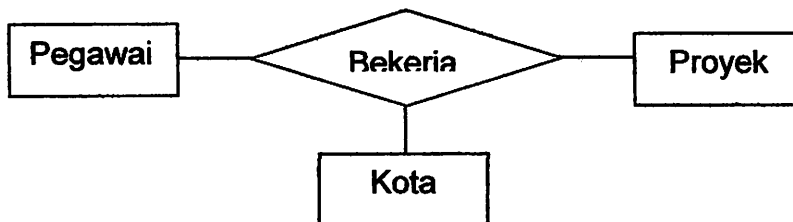
Keterangan :

Pegawai memiliki kendaraan, sebuah relationship memiliki menghubungkan entity Pegawai dan entity Kendaraan.

c. Ternary (Derajat Tiga)

Adalah satu buah relationship menghubungkan tiga buah entity. ^[5]

Contoh :



Gambar 2.20 Relationship Derajat 3 ^[5]

Keterangan :

Pegawai pada kota tertentu mempunyai suatu Proyek.

Entity Bekerja menghubungkan Entity Pegawai, Proyek dan Kota.

2.11.3 CARDINALITY RASIO

Yaitu menjelaskan batasan pada jumlah entity yang berhubungan melalui suatu relationship.

Jenis-jenis Cardinality Rasio : ^[5]

a. **One To One (1 : 1)**

Yaitu perbandingan antara entity pertama dengan entity kedua berbanding satu berbanding satu. ^[5]

Contoh :

Gambar 2.21 Rasio 1 to 1 ^[5]

b. One To Many (1 : M)

Yaitu perbandingan antara entity pertama dengan entity kedua berbanding satu berbanding banyak. ^[5]

Contoh :

Gambar 2.22 Rasio 1 to M ^[5]

c. Many To One (M : 1)

Yaitu perbandingan antara entity pertama dengan entity kedua berbanding banyak berbanding satu. ^[5]

Contoh :

Gambar 2.23 Rasio M to 1 ^[5]

d. Many To Many (M : M)


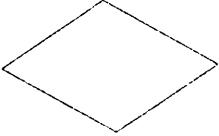




Yaitu perbandingan antara entity pertama dengan entity kedua berbanding banyak berbanding banyak. ^[5]

Contoh :

Gambar 2.24 Rasio M to M ^[5]

2.12 Flowchart

Flowchart merupakan sebuah diagram dengan simbol-simbol grafis yang menyatakan aliran algoritma atau proses yang menampilkan langkah-langkah yang disimbolkan dalam bentuk kotak, beserta urutannya dengan menghubungkan masing-masing langkah tersebut menggunakan tanda panah. Diagram ini bisa member solusi selangkah demi selangkah untuk penyelesaian masalah yang ada di dalam proses atau algoritma tersebut.

Gambar	Simbol untuk	Keterangan
	Proses / Langkah	Menyatakan kegiatan yang akan ditampilkan dalam diagram alir.
	Titik Keputusan	Proses / Langkah dimana perlu adanya keputusan atau adanya kondisi tertentu. Di titik ini selalu ada dua keluaran untuk melanjutkan aliran kondisi yang berbeda.
	Masukan / Keluaran Data	Digunakan untuk mewakili data masuk, atau data keluar.
	Terminasi	Menunjukkan awal atau akhir sebuah proses.
	Garis alir	Menunjukkan arah aliran proses atau algoritma.
	Kontrol / Inspeksi	Menunjukkan proses / langkah dimana ada inspeksi atau pengontrolan.

Gambar 2.25 Simbol-simbol Flowchart ^[5]

2.13 PHP

2.13.1 Sejarah PHP

Pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs Personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama *Form Interpreted* (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.^[4]

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perlisian kode sumber ini menjadi sumber terbuka (*open source*), maka banyak pemrogram yang tertarik untuk ikut mengembangkan PHP.^[4]

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini, *interpreter* PHP sudah diimplementasikan dalam program C. Dalam rilis ini, disertakan juga modul-modul ekstensi yang meningkatkan PHP/FI secara signifikan.^[4]

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang *intrepreter* PHP menjadi lebih bersih, lebih baik dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis *interpreter* baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang *PHP : Hypertext Preprocessing*. Pada pertengahan tahun 1999, Zend merilis *interpreter* PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.^[4]

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari *intrepreter* PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek (OOP) ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek.^[4]

2.13.2 Tentang PHP

PHP adalah singkatan dari *Hypertext Preprocessor* adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk memrogram situs web dinamis. PHP dapat digunakan untuk membangun sebuah CMS. Beberapa kelebihan PHP antara lain :^[4]

- a. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
- b. Web Server yang mendukung PHP dapat ditemukan dimana - mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi relative mudah.
- c. Dalam sisi pengembangan lebih mudah karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan.
- d. Dalam sisi pemahaman, PHP adalah bahasa *scripting* yang paling mudah karena memiliki referensi yang banyak.
- e. PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah - perintah system.

BAB III

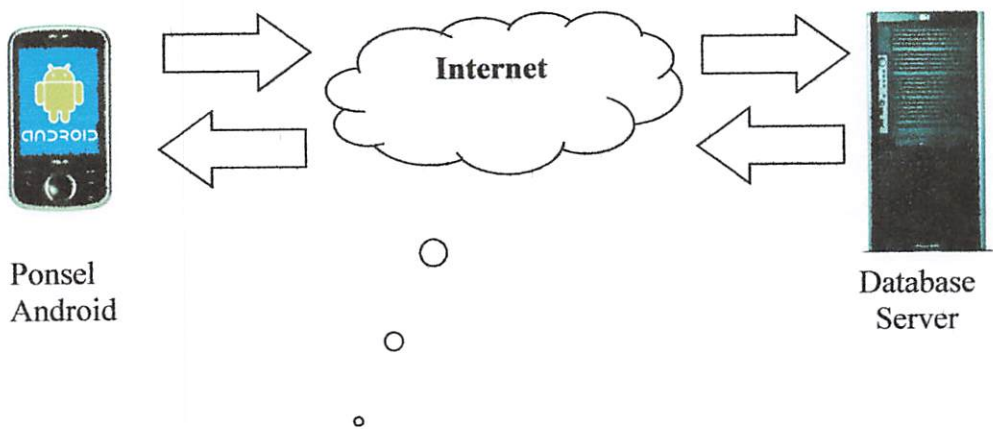
ANALISA DAN PERANCANGAN SISTEM

3.1 Analisa

Analisa kebutuhan sistem bertujuan untuk mengidentifikasi permasalahan - permasalahan yang ada pada sistem dimana aplikasi dibangun yang meliputi perangkat lunak (software), perangkat keras (hardware), dan pengguna. Analisa ini diperlukan diperlukan sebagai dasar bagi tahapan perancangan sistem dan untuk mengamati bagaimana sistem akan berjalan sesuai dengan persyaratan sistem yang mengikuti konsep dasar dari kebutuhan informasi di dalamnya. Analisa sistem dalam hal ini berorientasi objek dan akan membahas mengenai tahapan perangkat lunak dengan menentukan spesifikasi sistem atau *System Requirement Specification (SRS)* dan mengidentifikasi kelas - kelas serta hubungannya satu terhadap yang lain.

3.2 Rancangan Sistem

Aplikasi ini adalah aplikasi informasi lowongan kerja berbasis Android. Aplikasi ini akan otomatis terhubung ke internet dan terkoneksi dengan database untuk selalu mengupdate info lowongan terbaru yang akan dicari, rancangan sistem aplikasi Android ini seperti dalam gambar 3.2.1

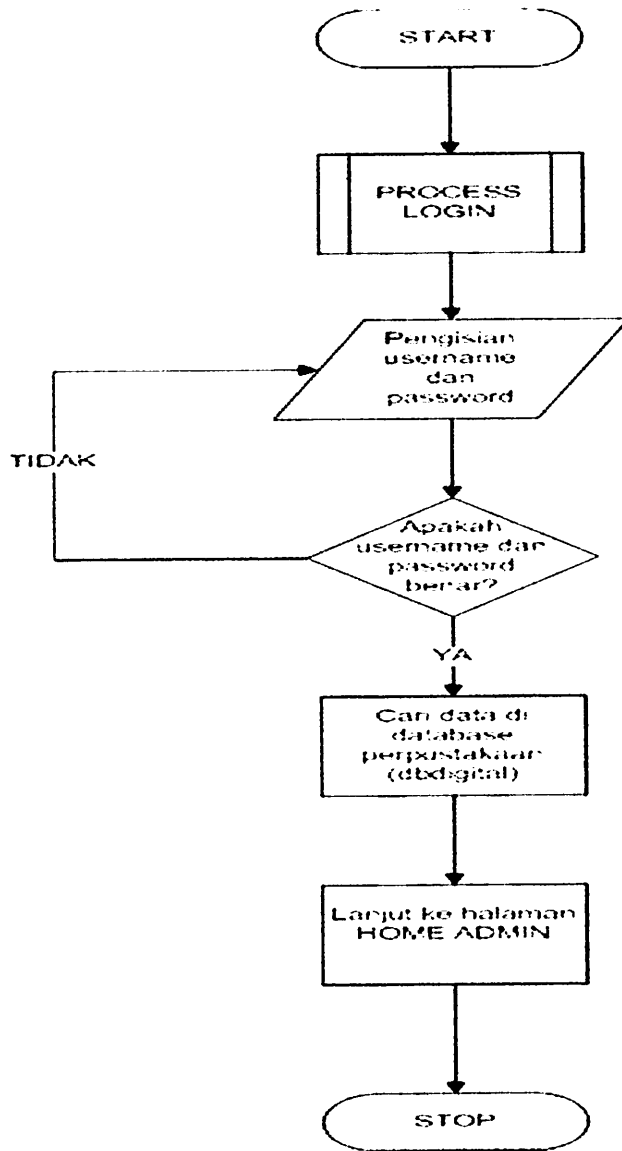


Gambar 3.1 Menampilkan Rancangan Penelitian

3.3 Rancangan alur program

Di dalam Diagram Alur Program menjelaskan setiap langkah alur data mulai dari user masuk ke dalam sistem (login), maka user tersebut harus memasukkan username dan password agar dapat mengakses database. Setelah itu melakukan login maka akan muncul tampilan kategori peminjaman buku. Setelah memilih kategori maka akan tampil list info buku dan judul buku yang akan di pinjam. Berikut adalah bentuk diagram alur .

3.4.1 Flowchart User



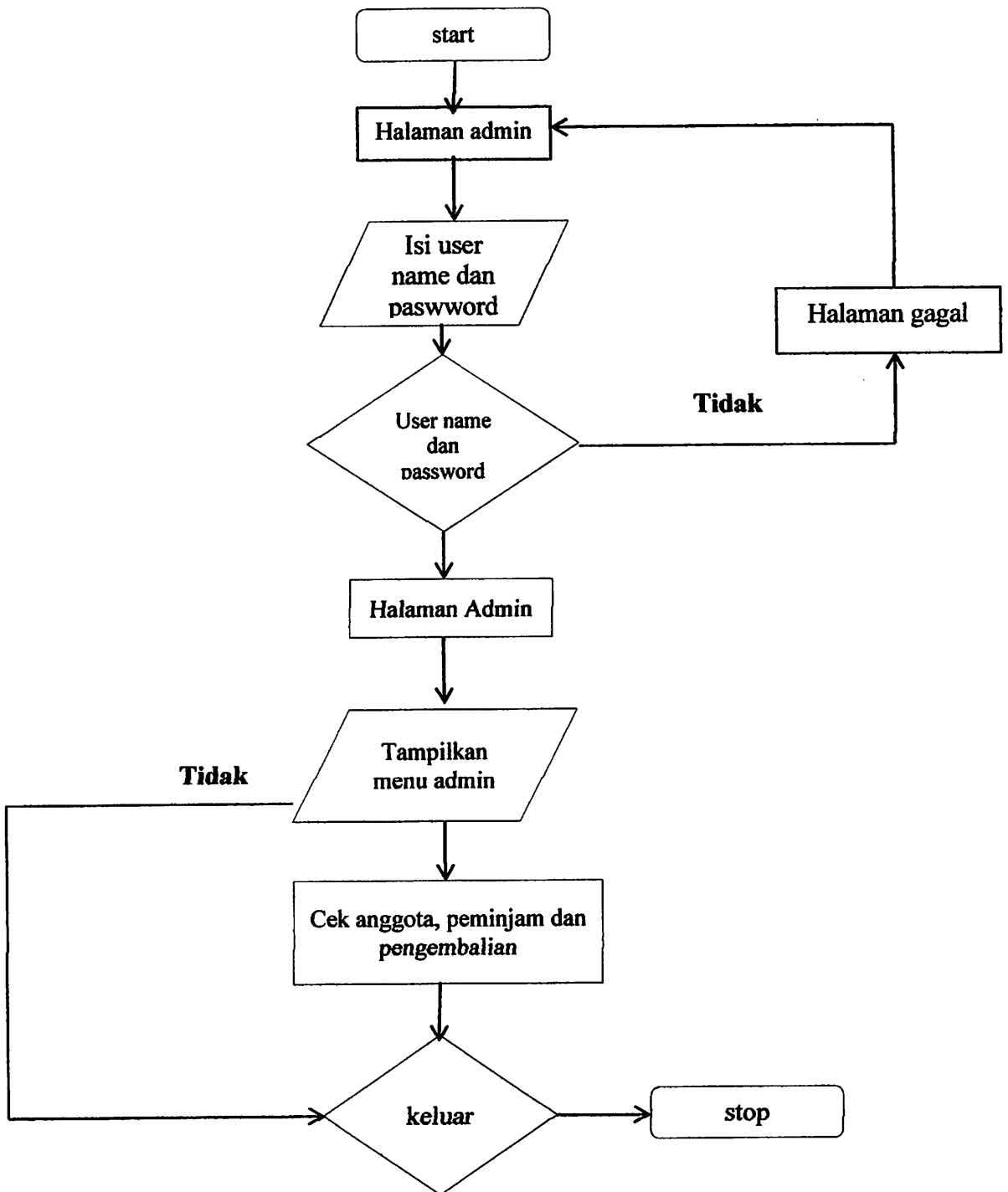
Gambar 3.2 Menampilkan Flowchart User

Keterangan dari gambar 3.2 diatas adalah:

Pada gambar 3.2 diatas di jelaskan bahwa user masuk kedalam sistem login. Dimana user memasukkan username dan password dan jika username dan password benar maka user langsung menuju database perpustakaan. Jika username dan password salah maka akan di arahkan menuju pengisian username dan password untuk mendaftar ulang.

Jika userrname dan password benar maka user akan menuju database perpustakaan. Dimana di dalam database perpustakaan terdapat informasi mengenai data buku yang di inginkan oleh user berupa data buku, nama buku, dan peminjam buku.

3.4.2 Flowchart Pada Admin



Gambar 3.3 Menampilkan Flowchart Admin

Penjelasan gambar 3.3

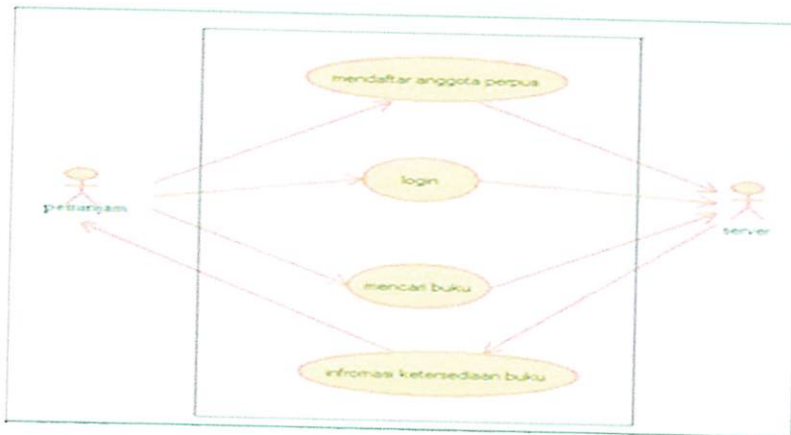
Pada halaman admin ini admin melakukan pengisian user name dan password jika user name dan password benar maka admin memasuki halaman admin jika admin salah atau belum mendaftar maka admin akan kembali kehalaman awal untuk melakukan pendaftaran.

Didalam menu admin terdapat menu admin berupa cek anggota perpustakaan, peminjaman buku dan pengembalian buku. Jika admin tidak ingin melakukan kegiatan maka admin bisa menuju menu keluar.

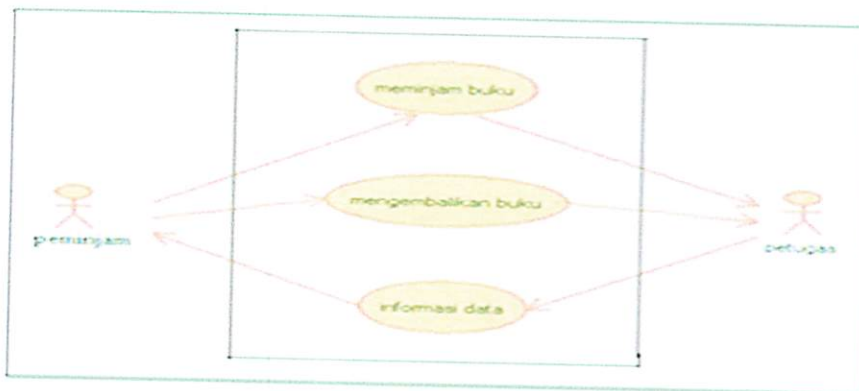
3.4.2 Perancangan Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang di harapkan dari sebuah system. Yang ditekankan adalah” Apa” yang di perbuat oleh system, dan bukan “ Bagaimana “.

Sebuah Use Case mempresentasikan sebuah interaksi antara user dengan system.



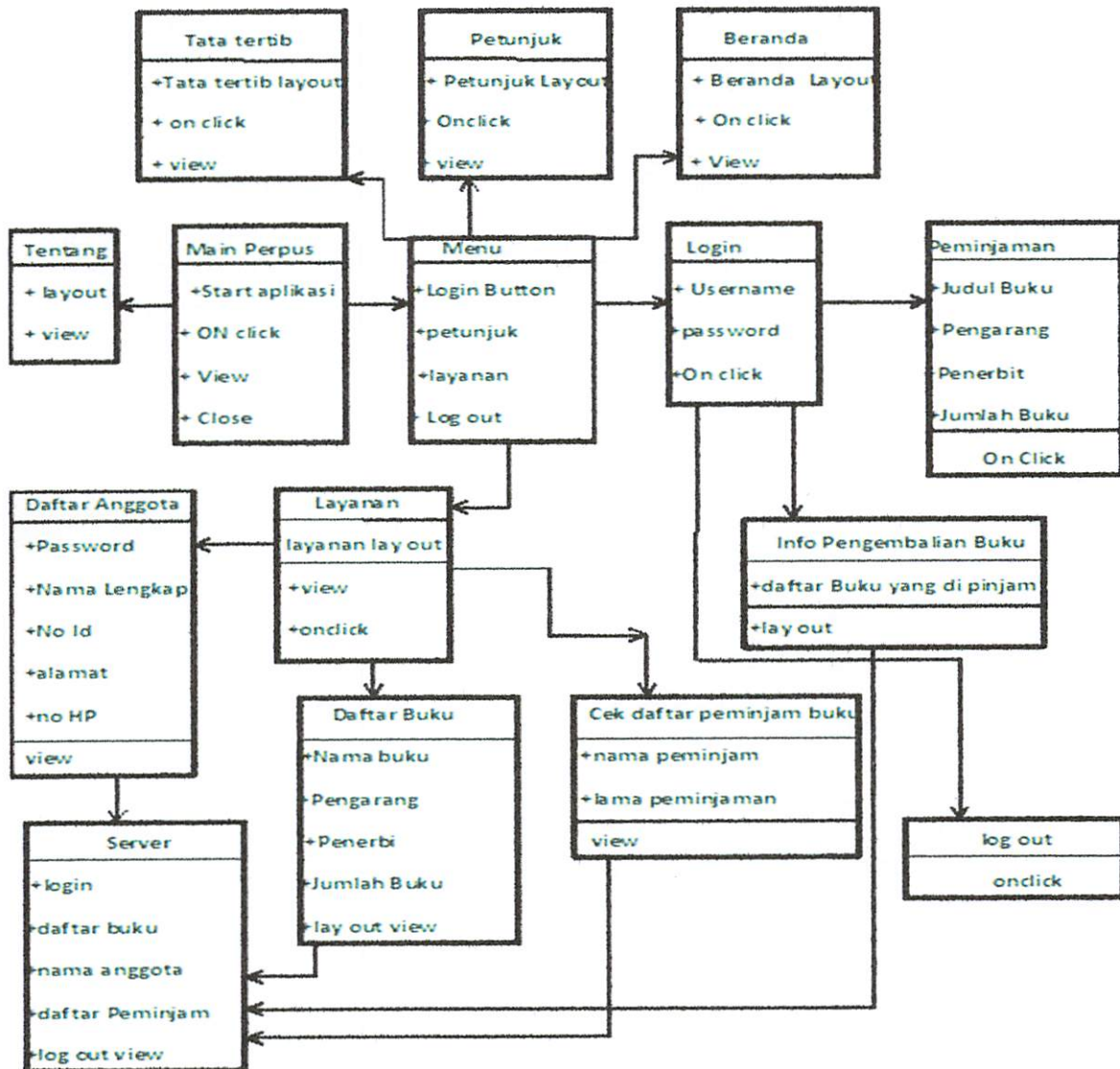
Gambar 3.4 Menampilkan Use Case Diagram di Server



Gambar 3.5 Menampilkan use Case Diagram Pada Client

3.4.3 Perancangan Class Diagram

Class adalah sebuah spesifikasi yang system jika di intansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut atau property) suatu system, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut. Dalam gambar 3.6 di tampilkan gambar class diagram



Gambar 3.6 Menampilkan Gambar Class Diagram

Diagrama Sistem Informasi (DSI) adalah suatu gambar yang menunjukkan struktur dan hubungan antara berbagai komponen sistem informasi.

Diagrama Sistem Informasi (DSI) dapat digunakan untuk menggambarkan struktur organisasi, proses bisnis, dan aliran informasi dalam suatu organisasi. Diagrama ini membantu dalam memahami bagaimana informasi mengalir dan diproses dalam organisasi, serta bagaimana sistem informasi mendukung operasi organisasi.

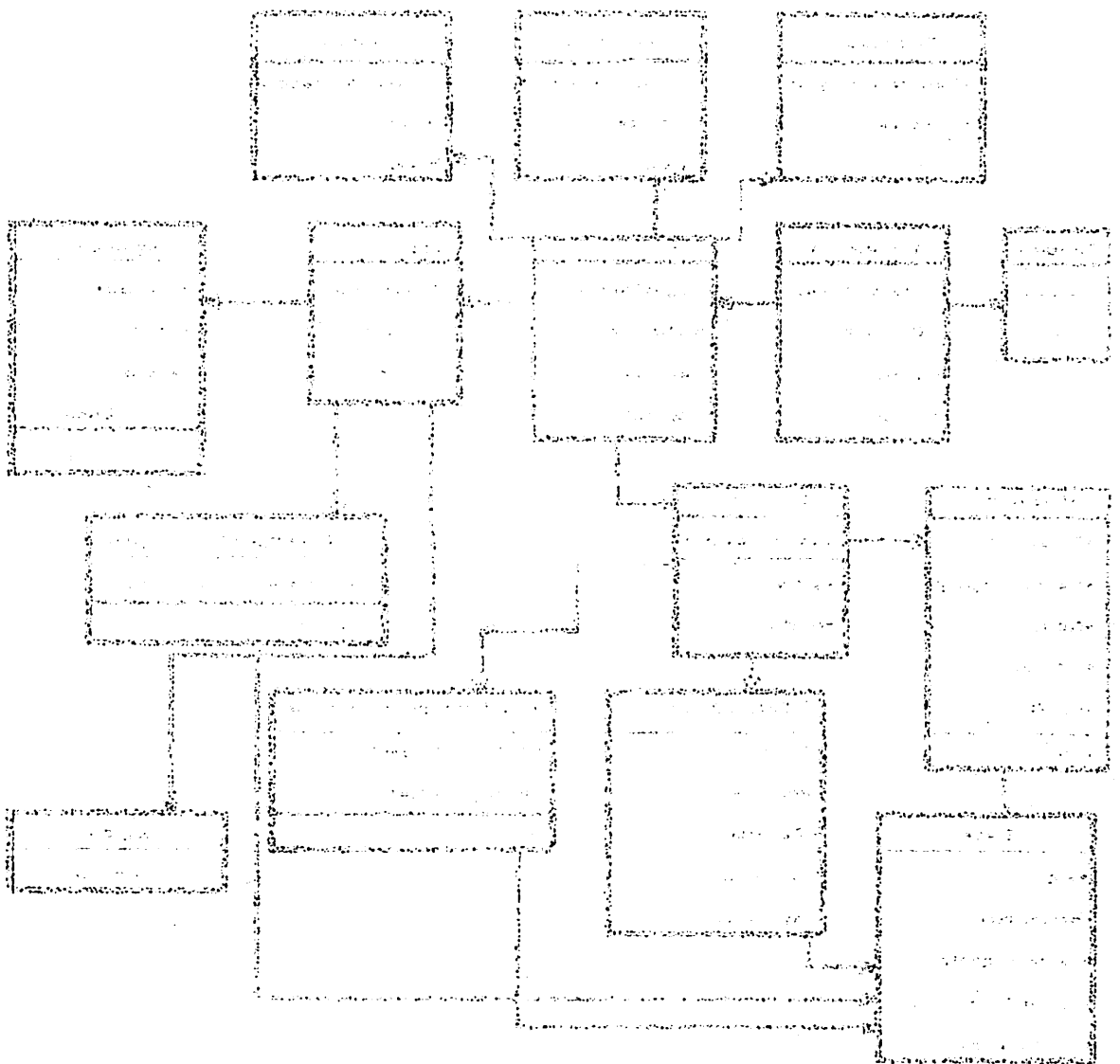


Diagrama Sistem Informasi (DSI) 2021

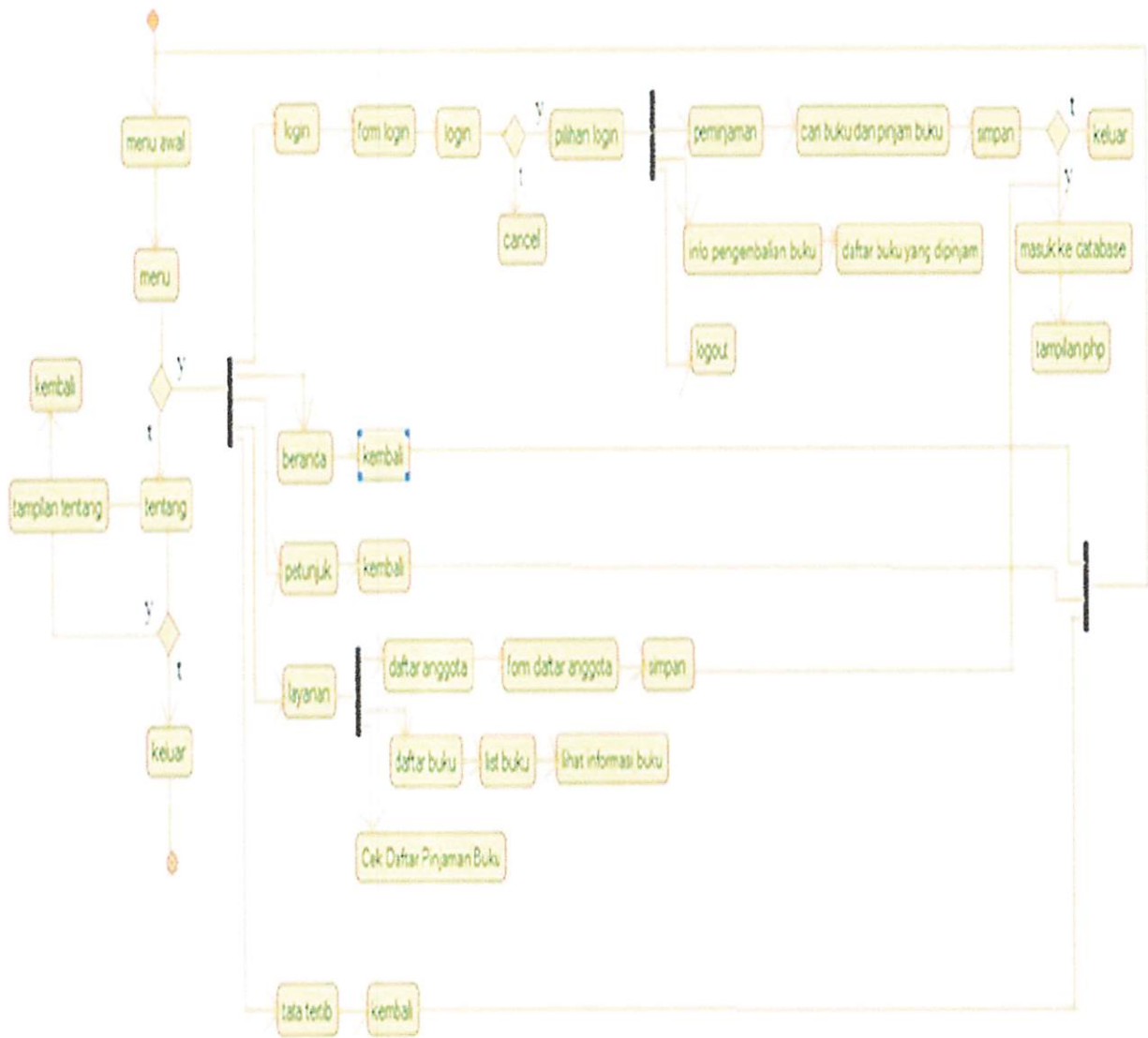
3.4.4 Perancangan Activity Diagram

Activity diagram memnggambarkan aliran awal berawal dari menu awal aplikasi kemudian menuju menu utama , disini terjadi pembagian informasi jika user ingin *melanjutkan maka user akan melakukan input data dan akan masuk langsung kedalam beranda aplikasi*. Di dalam beranda terdapat dua pilihan yaitu berupa petunjuk aplikasi dan juga layanan.

Jika user ingin login maka akan langsung muncul form login dan user bisa mengisikan data diri pada form login yang sudah disediakan, setelah melakukan login user akan di beri pilihan berupa menu peminjaman, info peminjaman, dan logut. Jika user memilih menu peminjaman buku maka user akan memasukkan data buku yang ingin di cari berupa nama buku, tahun buku dan penerbit setelah meminjam buku maka user bisa menyimpan data buku yang mau dipinjam dan dapat di cek di daftar buku yang dipinjam. Setelah kegiatan peminjaman buku maka user akan di kasih pilihan di dalam aplikasi berupa pilihan log out atau kembali ke beranda.

Di dalam diagram Activity ini admin bisa melihat data peminjam buku. Sebelum masuk menu layanan admin harus melakukan login agar bisa masuk kedalam menu utama, setelah masuk menu utama admin akan di kasih pilihan berupa daftar anggota, daftar buku yang di pinjam dan juga menu cek daftar peminjaman buku setelah amin selesai melakukan pengecekan maka admin bisa langsung menuju menu log out.

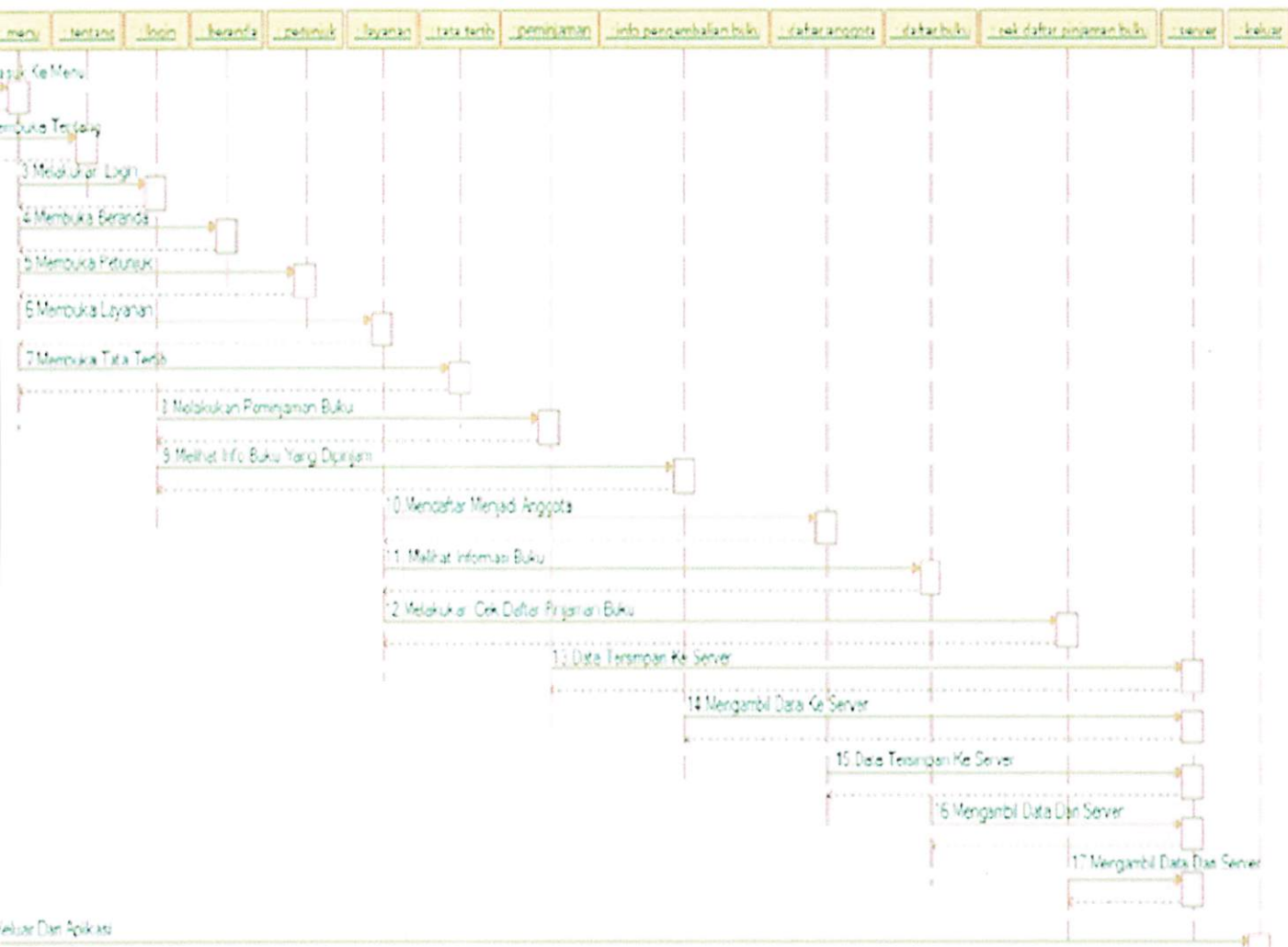
Gambar 3.7 menampilkan gambar actifity diagram



Gambar 3.7 Menampilkan Gambar Activity Diagram

3.4.5 Diagram Perancangan Sequence

Sequence diagram adalah diagram yang menggambarkan interaksi antara objek di dalam dan di sekitar system (termasuk pengguna, display, dan sebagainya) berupa message yang digambarkan terhadap waktu. Sequence diagram terdiri atas dimensi vertical (waktu) dan dimensi horizontal (objek-objek yang terkait). Dalam gambar 3.8 ditampilkan diagram perancangan sequence.



Gambar 3.8 Menampilkan Diagram Sequence

3.4.6 Struktur Data Base

Struktur data base menjelaskan tentang data-data yang terdapat pada system informasi ini yang di buat menjadi table-table database yang akan di proses ke dalam system.

1) Nama Tabel : Tb_login

Primary key: id_login

Fungsi : Tb_login digunakan oleh user dan admin untuk masuk kedalam system informasi perpustakaan ini. Pada level admin, admin dapat mengelola system informasi ini dimana admin dapat menambah, merubah dan menghapus data-data yang di dalam sitem informasi ini

Tabel 3.1 Menampilkan Tabel_login

no	Field	type	Size	Ket
1	Id_login	Int	11	Primary
2	User name	Varchar	10	-
3	NIM	Varchar	10	-
4	Password	Varchar	50	-
5	Level	Num	-	-

2) Nama Tabel : Tb_anggota

Primary_key: no_id

Fungsi : digunakan untuk menyimpan No_Id saat melakukan pendaftaran

Tabel 3.2 Menampilkan Tabel_Anggota

No	Field	Type	Size	Ket
1	Jenis_id	Varchar	10	Primary
2	No_id	Varchar	10	-
3	Nama	Varchar	50	-
4	User_name	Enum	50	-

5	Password	Varchar	50	-
6	Konfirmasi_password	Varchar	50	-
7	Alamat	Text	20	-
8	Kecamatan	Text	20	-
9	Kabupaten	Text	20	-

3) Nama Tabel: Tb_Petugas

Primary key : form_petugas

Fungsi: digunakan untuk menyimpan semua informasi tentang petugas

Tabel 3.3 Menampilkan Tabel_Petugas

no	Field	type	Size	Ket
1	Nama_lengkap	Varchar	50	Pimary
2	User_name	Varchar	50	Primary
3	Password	Varchar	50	Primary
4	Konfirmasi_passwird	Varchar	50	Primary
5	No_tlp	Varchar	20	-
6	Email	Text	20	-

4) **Nama Tabel: Tb_Menu Petugas**

Primary key: form_menu petugas

Fungsi: digunakan untuk informasi untuk mengetahui list anggota, buku dan list semua peminjaman.

Tabel 3.4 Menampilkan Tabel_Menu Petugas.

no	Field Field	type	Size	Ket
1	List_anggota	varchar	30	Primary
2	List_Peminjam	varchar	30	-
3	List_buku	varchar	30	-

5) **Nama Tabel: Tb_Buku**

Primary key: id_buku

Fungsi: digunakan untuk menyimpan data buku yang ada

Tabel 3.5 Menampilkan Tabel_Buku

no	Field Field	type	Size	Ket
1	Kode_buku	varchar	30	Primary
2	Nama_buku	varchar	30	Primary
3	Pengarang_buku	varchar	30	Forenkey
4	Penerbit	varchar	20	-
5	Jumlah_buku	varchar	30	-

6) Nama table: Tb_Server

Primary key: id_server

Fungsi: di gunakan untuk mengetahui data yang akan kita proses di server

Tabel 3.6 Menampilkan Tabel_server

no	Field Field	type	Size	Ket
1	Login	int	10	Primary
2	Daftar_buku	varchar	30	-
3	Daftar_anggota	varchar	30	-
4	Daftar_peminjam	varchar	30	-
5	Logout	text	-	-

7) Nama table: Tb_peminjaman

Primary key: id_peminjaman

Fungsi: digunakan untuk mengetahui tentang data peminjaman

Tabel 3.7 Menampilkan Tabel_Peminjaman

no	Field Field	type	Size	Ket
1	Kode_buku	Int	20	Primary
2	Nama_buku	Varchar	30	-
3	Pengarang	Varchar	30	-
4	Penerbit	Varchar	30	-
5	Jumlah_tersedia	Int	-	-

8) Nama table: Tb_daftar peminjaman**Primary key: id daftar peminjaman****Fungsi:** digunakan untuk mengetahui daftar peminjaman

Tabel 3.8 Menampilkan Tabel_Daftar Peminjaman

no	Field Field	type	Size	Ket
1	Check	Varchar	-	Primary

9) Nama table: Tb_Tentang**Primary key: id _ Tentang****Fungsi:** digunakan untuk memberi informasi tentang aplikasi

Tabel 3.9 Menampilkan Tabel_Tentang

no	Field Field	type	Size	Ket
1	Tentang_layout	Varchar	-	Primary

10) Nama table: Tb_Petunjuk**Primary key: id _ petunjuk****Fungsi:** digunakan untuk memberi informasi soal langkah langkah yang harus di lakukan oleh user.

Tabel 3.10 Menampilkan Tabel_Petunjuk

no	Field Field	type	Size	Ket
1	Petunjuk_layout	Varchar	-	Primary

11) Nama Table: Tb_logout**Primary key: id_logout****Fungsi:** untuk memberikan petunjuk keluar dari aplikasi

Tabel 3.11 Menampilkan Tabel_logout

no	Field Field	type	Size	Ket
1	On_click	Varchar	-	Primary

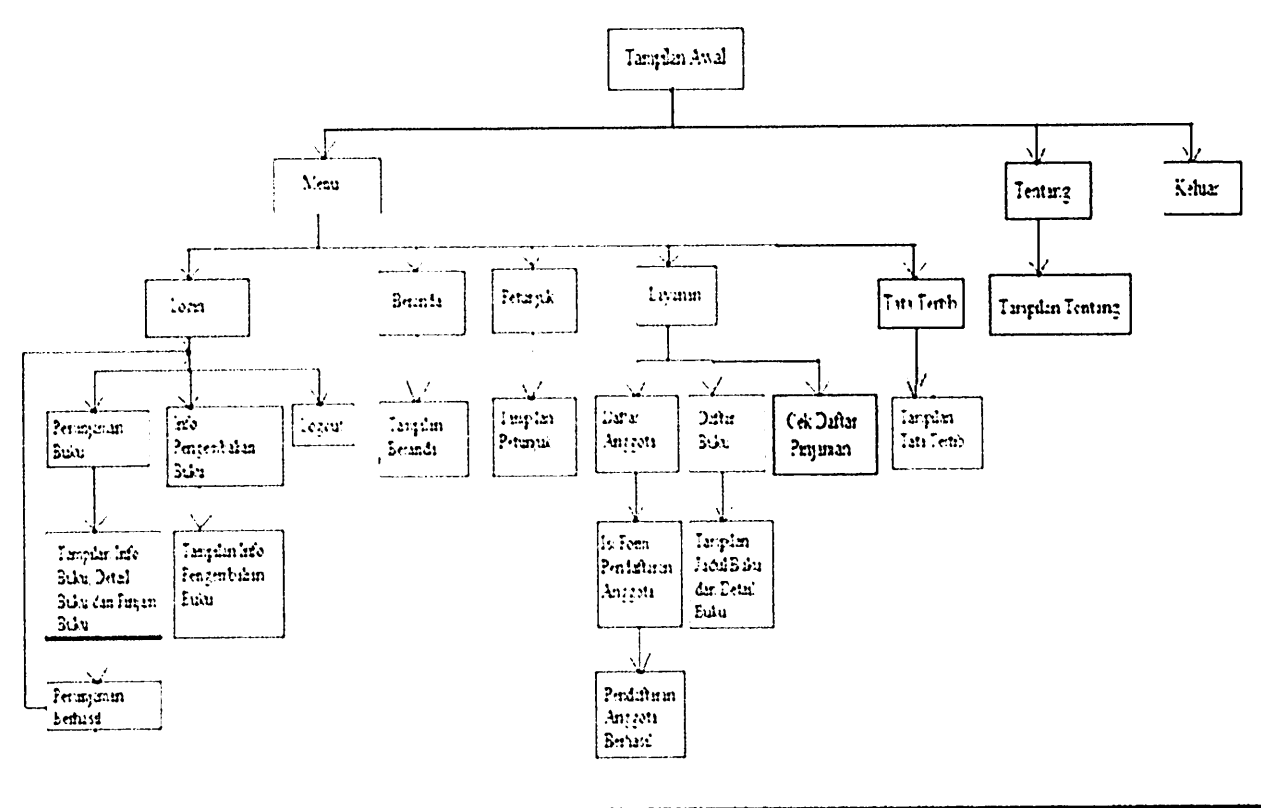
12) Nama Table: Tb_beranda**Primary key : id_beranda****Fungsi:** untuk memberi petunjuk user di beranda

Tabel 3.12 Menampilkan Tabel_Beranda

no	Field Field	type	Size	Ket
1	On_click	Varchar	-	Primary

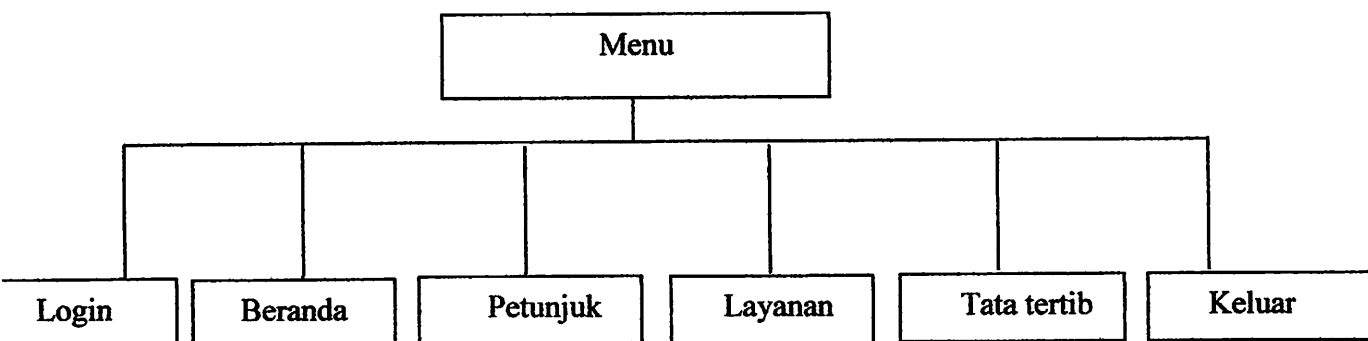
3.4.8 Desain Struktur Navigasi

Perancangan struktur navigasi berisikan menu dan submenu yang berfungsi untuk memudahkan user dalam menggunakan system. Menu-menu tersebut di bagi berdasarkan hak akses masing-masing user. Berikut ini adalah gambaran mengenai struktur navigasi.



Gambar 3.7 Menampilkan struktur Navigasi

3.4.9 Desain navigasi menu



Gambar 3.8 menampilkan struktur menu

Keterangan gambar 3.8:

a) Login

Halaman login merupakan halaman utama sebelum user masuk kedalam tampilan menu.

b) Beranda

Halaman beranda merupakan halaman utama sesudah user melakukan login pada awal tampilan.

c) Petunjuk

Merupakan halaman yang berisi informasi tentang langkah langkah yang di gunakan oleh user selama menggunakan aplikasi ini.

d) Layanan

Halaman ini memberikan informasi tentang layanan yang digunakan oleh user.

e) Tatatertib

Di dalam halaman ini memberikan informasi tentang tata tertib dari aplikasi ini.

f) Keluar

Menu keluar digunakan untuk keluar dari aplikasi.

BAB IV

IMPLEMENTASI DAN HASIL PENELITIAN

4.1 Implementasi

Penerapan aplikasi ini diharapkan dapat membantu mahasiswa dalam melakukan kegiatan pinjam meminjam buku baik secara langsung mengunjungi perpustakaan ataupun bisa melakukan peminjaman melalui online dengan menggunakan media internet yang sudah terhubung ke handphone android.

Aplikasi ini juga membantu pegawai dalam melakukan pengecekan jumlah stok buku yang ada di dalam perpustakaan.

4.1.1 Kebutuhan Software dan Hardware

Pada kebutuhan software dan hardware dibagi menjadi 2 bagian, kebutuhan software pada android mobile dan komputer.

a. Spesifikasi pada android mobile :

- i. Min. Android 1.5 (Cupcake).
- ii. Min. Processor 600 MHz.
- iii. Mendukung fitur GPS.

b. Spesifikasi pada komputer :

- i. Min. Processor (Pentium 4) 2.4 Ghz.
- ii. Windows Xp (min. memory 1 Gb).
- iii. Windows Vista atau Windows 7 (min. memory 2 Gb).
- iv. HDD min. 5 Gb.
- v. Xampp untuk windows.
- vi. Eclipse dan Android Plugin (min. memory 1 Gb).

4.2 Penjelasan Sistem

4.2.1 Halaman Beranda

Halaman beranda merupakan halaman utama pada aplikasi ini di mana user harus melakukan pendaftaran jika belum memiliki password.

Jika user sudah terdaftar maka user hanya memasukkan username dan password.

Disini terdapat 2 fasilitas yaitu daftar petugas dan anggota di mana daftar petugas di fungsikan sebagai petugas yang ingin mendaftar sebagai pengurus perpustakaan. Jika user ingin menjadi anggota maka user memilih button daftar anggota.



Gambar 4.1 Menampilkan menu awal aplikasi

4.2.2 Halaman form Petugas

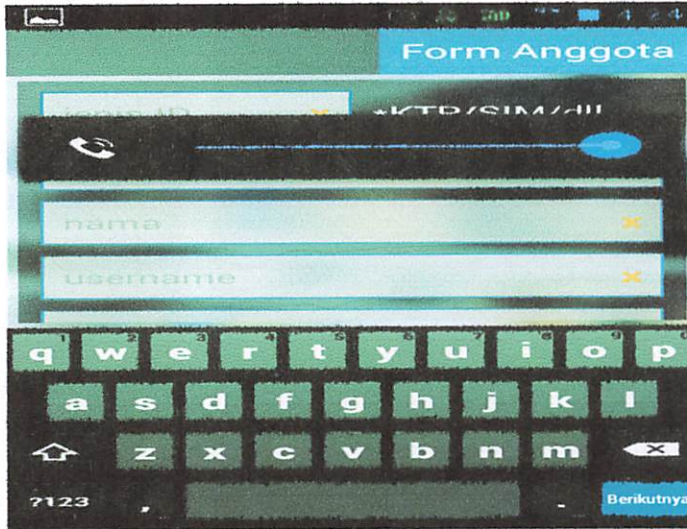
Halaman ini berisikan tentang informasi petugas dimana petugas melakukan registrasi awal atau mendaftar sebagai petugas.



Gambar 4.2 Menampilkan halaman form Petugas

4.2.3 Halaman form Anggota

Halaman ini berisikan form anggota di mana user harus mendaftar dan mengisi semua form yang telah di di menuform.



Gambar 4.3 Menampilkan Halaman form anggota

4.2.4 Menu Utama petugas

Halaman ini berisikan tentang menu utama petugas, setelah mendaftar sebagai petugas maka akan langsung muncul menu petugas.



Gambar 4.4 Menampilkan Halaman utama petugas

the following table shows the results of the analysis of variance for the data in Table 1.

TABLE 1. Analysis of Variance

Source	df	SS	MS	F	p
Between	1	10.5	10.5	1.2	.28
Within	18	157.5	8.75		
Total	19	168.0			

The analysis of variance shows that the difference between the two groups is not significant.

The results of the analysis of variance are given in Table 1.

The following table shows the results of the analysis of variance for the data in Table 2.

TABLE 2. Analysis of Variance

Source	df	SS	MS	F	p
Between	1	10.5	10.5	1.2	.28
Within	18	157.5	8.75		
Total	19	168.0			

The analysis of variance shows that the difference between the two groups is not significant.

4.2.5 Halaman list anggota

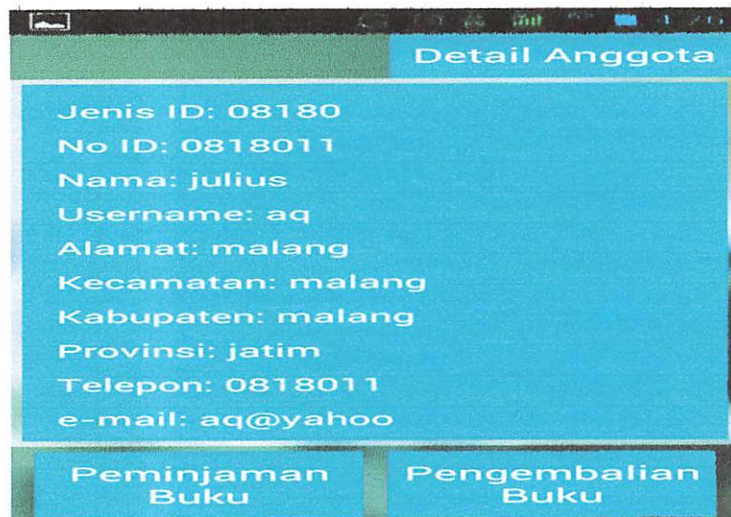
Halaman ini berisikan list dari semua anggota yang sudah mendaftar tampilan list dapat dilihat jika kita melakukan atau memasukkan keyword anggota.



Gambar 4.5 Menampilkan list anggota

4.2.6 Halaman detail list anggota

Halaman ini berisikan detail tentang anggota yang sudah mendaftar halaman ini memuat alamat anggota dan identitas lainnya



Gambar 4.6 Menampilkan Detail list anggota

... ..

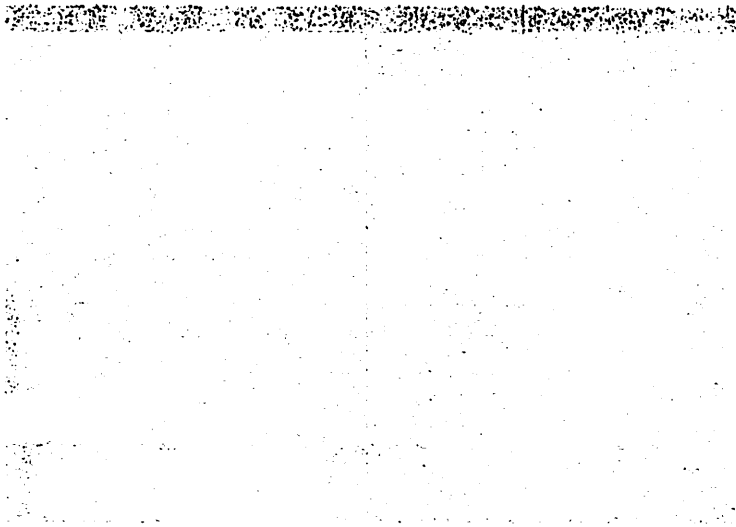
... ..



... ..

... ..

... ..



... ..

4.2.7. Halaman List Buku

Halaman ini menjelaskan tentang list buku yang akan di cari oleh user dengan cara memasukkan keyword buku dan kemudian menklik tombol cari maka nama buku yang di cari akan di tampilkan.

Gambar 4.7 Menampilkan Pencarian Buku

4.2.8 Halaman hasil pencarian Judul buku

Halaman ini menjelaskan setelah kita melakukan input nama buku yang di cari maka akan mendapatkan hasil berupa tanggal pengembalian, tanggal peminjaman dan juga berupa denda.

Gamabar 4.8 Menampilkan Halaman hasil pencarian buku

... ..
... ..
... ..

[REDACTED]

SECRET

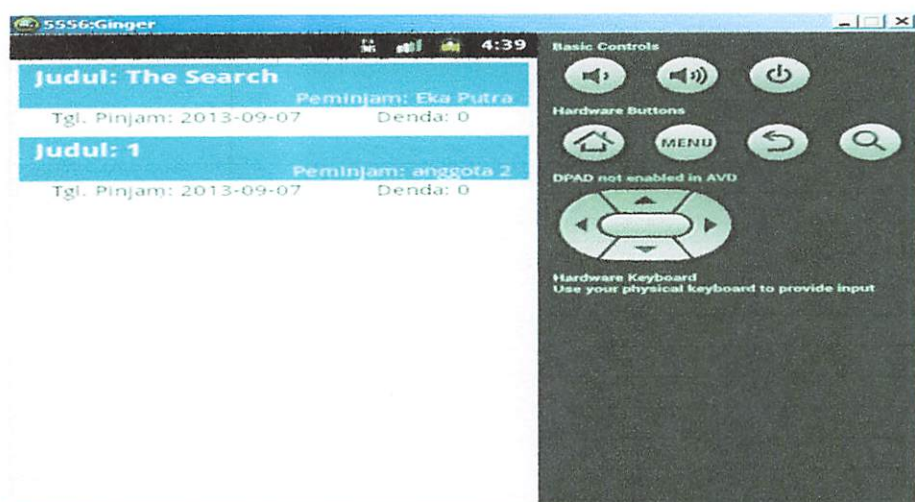
... ..
... ..
... ..

[REDACTED]

SECRET

4.2.9 Halaman Semua List Peminjaman Buku

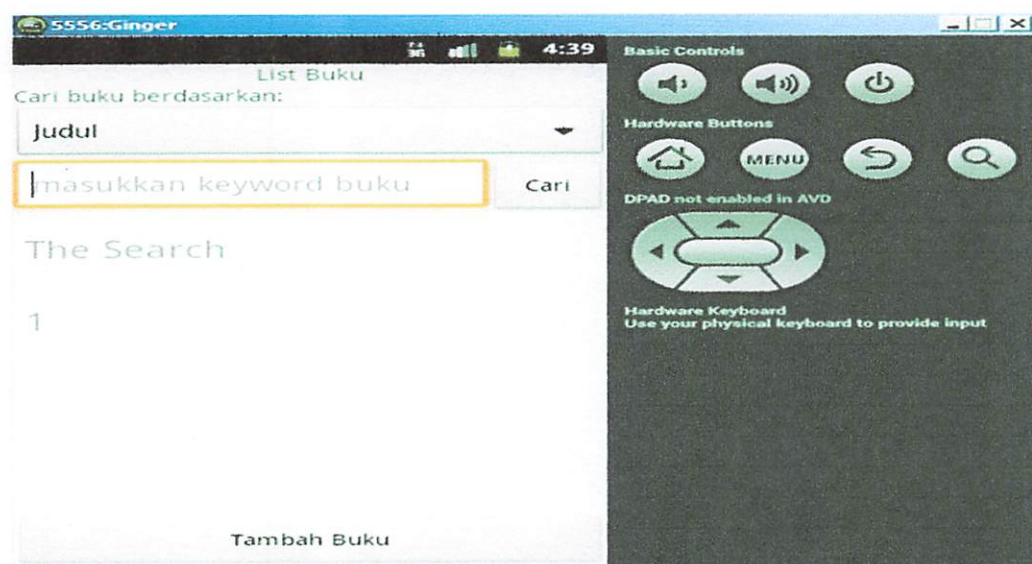
Halaman ini menampilkan semua list buku yang di pinjam oleh user didalam list ini terdapat tanggal peminjaman dan juga denda.



Gambar 4.9 Menampilkan Semua List Peminjaman Buku

4.2.10 Halaman Admin Untuk Menambah Buku

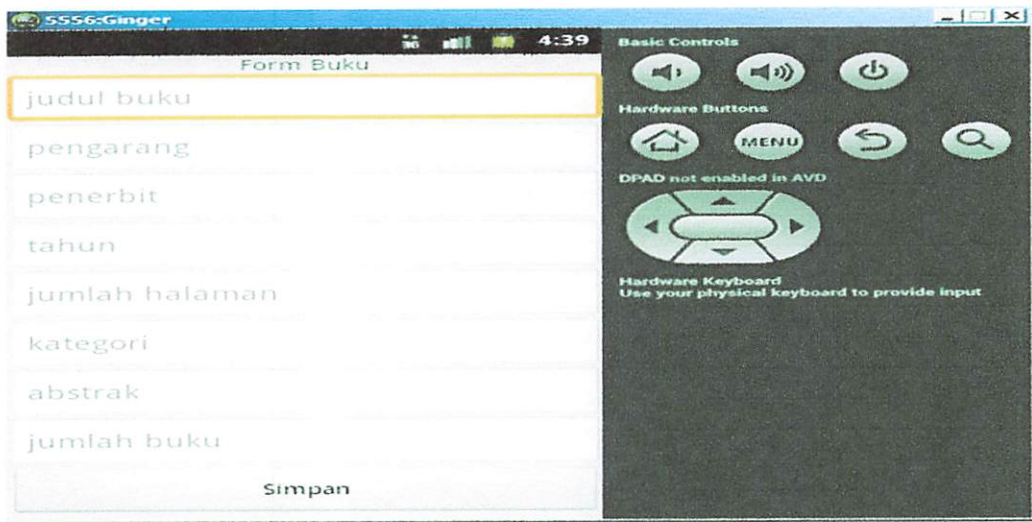
Halaman ini menampilkan list buku yang akan di tambahkan oleh admin.



Gambar 4.10 Menampilkan list buku yang akan di tambahkan oleh admin.

4.2.11 Halaman Penambahan Buku

Halaman ini berisikan list yang harus di isi oleh admin untuk melakukan tugas penambahan buku.

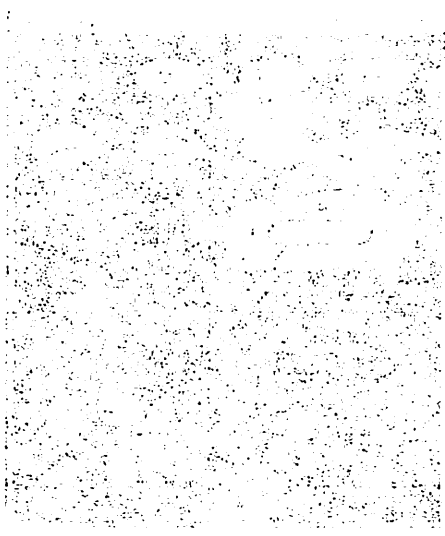


Gambar 4.11 menampilkan List penambahan Buku Oleh Admin.

1950-1951

Annual Report of the Board of Directors

1950-1951



Annual Report of the Board of Directors

4.3 Pengujian Aplikasi

4.3.1 Metode Black Box

Setelah system selesai di bangun, maka kita membuat langkah selanjutnya yaitu melakukan pengujian dan evaluasi terhadap system untuk mengetahui performa dari system.

Berikut ini adalah kasus yang menguji perangkat lunak yang menggunakan metode black box.

Tabel 4.1. Menampilkan Pengujian Black Box

No	Form Uji	scenario uji	Hasil pengujian
1	Login	Masukkan Data id Yang benar	Diterima
		Masukan data ID yang salah	Di tolak
2	Register	Mengisi form register dengan benar	Diterima
		Mengisi form register dengan salah	Di tolak
3	Pilih judul Buku	Masukkan judul buku Dengan benar	Diterima

4.3.2 Tabel Pengujian Pengguna

Tabel 4.2 Menampilkan Tabel Pengujian Pengguna

No	Parameter Materi	Hasil (%)
1	Materi informasi pada aplikasi	60 %
2	Tampilan di file hosting	80 %
3	Kelayakan sistem aplikasi	60 %
4	Menu pada aplikasi	60 %

Keterangan:

1. Materi informasi mendapat hasil 60% karena dari 5 responden yang mengisi kuisisioner 1 responden memberi 1 nilai cukup pada materi informasi pada aplikasi.
2. Tampilan di file hosting mendapat hasil 80% karena dari 5 responden yang mengisi kuisisioner memberi nilai baik.
3. Kelayakan sistem aplikasi mendapat hasil 60% karena dari 5 responden yang mengisi kuisisioner 1 responden memberi 1 nilai cukup pada kelayakan sistem aplikasi.
4. Menu aplikasi mendapat hasil 60% karena dari 5 responden yang mengisi kuisisioner 1 responden memberi 1 nilai cukup pada maenu aplikasi.

4.3.3 Tabel Pengujian Terhadap Versi *Android*

Tabel 4.3 Menampilkan Tabel Pengujian Terhadap Versi *Android*

No	Tipe Ponsel	Versi OS	Kualitas Tampilan	Akses Database
1	Samsung Galaxy Chat	4.1.2	Baik	100%
2	Samsung Galaxy W I8150	2.3.5	Baik	100%
3	Lenovo A390	4.0.4	Baik	100%
4	Samsung Galaxy SIII mini	4.1.2	Baik	100%
5	Samsung Galaxy Note II	4.1.1	Baik	100%

Pada tabel 4.3 Tingkat kualitas kinerja yang dihasilkan dari aplikasi ini pada setiap sistem operasi yang digunakan akan memiliki hasil yang berbeda, aplikasi ini dapat berjalan dengan baik pada *android* versi 4.1.2, dan kurang berjalan dengan baik pada *android* versi 2.3 karena tidak dapat membuka beberapa menu utama seperti, cari buku, menu registrasi pengaturan dan tentang.

4.3.4 Tabel Pengujian Terhadap Resolusi Layar (*landscape*)

Tabel 4.6 menampilkan Tabel Pengujian Terhadap Layar (*landscape*)

No	Tipe Ponsel	Resolusi Layar	Ukuran Layar (Inche)	Tampilan Menu
1	Samsung Galaxy Chat	240 x 320	3.0	Cukup Baik
2	Samsung Galaxy W I8150	480 x 800	3.7	Baik
3	Lenovo A390	480 x 800	4.0	Baik
4	Samsung Galaxy SIII mini	480 x 800	4.0	Baik
5	Samsung Galaxy Note II	720x 1280	5.5	Baik

Pada tabel 4.6 Tingkat kualitas tampilan yang dihasilkan dari aplikasi ini pada posisi layar *landscape* menghasilkan tampilan yang baik apabila menggunakan smartphone dengan minimum resolusi 480x800 dengan ukuran layar 4.0 inche. Sedangkan pada resolusi 240x320 dengan ukuran layar 3.0 inche ke bawah, terutama pada bagian menu aplikasi, sebagian tombolnya akan saling berhimpitan.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan permasalahan yang di bahas di dalam skripsi ini dapat diambil kesimpulan bahwa:

1. Materi informasi mendapat hasil 60% karena 1 kuisisioner memberi nilai C.
2. Tampilan di file hosting mendapat nilai 80% karena semua kuisisioner memberi nilai B.
3. Kelayakan sistem aplikasi mendapat hasil 60% karena 1 kuisisioner memberi nilai C.
4. Menu pada aplikasi 60% karena 1 kuisisioner memberi nilai C.
5. Pada tampilan login jika memasukkan data ID dengan salah maka hasil akan di tolak.
6. Untuk pengujian kualitas tampilan, akses *database* dan akses *maps* pada *versi android*, dapat berjalan 100% pada 5 perangkat android.
7. Pada pengujian kualitas aplikasi ini, terhadap resolusi layar(*landscape*) dapat berjalan 100% pada 5 perangkat android.
8. Aplikasi ini menampilkan data stock buku yang dapat dilihat pada handphone.
9. Aplikasi ini hanya bisa di jalankan di Android versi Jelly Bean.
10. Pembuatn aplikasi ini menggunakan bahasa pemrograman Java yang terdapat di dalam aplikasi Eclipse.

5.2 Saran

Adapun saran untuk pengembangan lebih lanjut dari system ini adalah:

1. Diperlukan jaringan internet untuk koneksi aplikasi ini karena aplikasi sangat berat.
2. Untuk pengembangan aplikasi selanjutnya perlu ditambahkan QR code yang berfungsi untuk mengscan barcode.
3. Penambahan *Share button* ke *social network* seperti *Facebook* dan *Twitter* untuk lebih di perluas lagi.
4. Untuk pengembangan aplikasi selanjutnya di harapkan tampilan menu di buat lebih menarik lagi.

DAFTAR PUSTAKA

- [1]. Meier,R. 2010.” *Profesional Android Application*” : wiley publishing,inc.
- [2]. Rogers, 2009.” *Android Application Development*”,. Reilly Media
- [3]. Michael Siregar,ST.,M.T,Ivan,2011 “membongkar Source Code berbagai Aplikasi Android Media:Yogyakarta.
- [4]. Sidik,Betha(2012).*PemrogramanPHP*. Bandung: Informatika Bandung.
- [5]. Safaat,Nazruddin (2012)*Pemrograman Aplikasi smartphone, Tablet PC*
- [6]. Sidik,Betha(2012).*PemrogramanHTML*.Bandung:Informatika
- [7]. R.H. Sianipar, ST. MT M.Eng PH.D (2013). *Teori dan Implementasi Java*.
- [8]. Akbarul Huda,S.Si, Arif. 2012. *24 jam pintar pemrograman*. Andi Publisher : Yogyakarta.
- [9].Komputer Wahana. 2013. *Membuat aplikasi Android untuk Tablet dan Handphone*. PT Elex Media Komputindo : Jakarta.
- [10]. (Pengertian javascript pop jquery dan html <http://www.it-suraji.tk/2012/06>)

Nama : Novia Chilmia

Pekerjaan : PNS

Berikan penilaian sesuai dengan grade yang sesuai pada kriteria penilaian dengan keterangan yang sesuai pada table di bawah ini, untuk penilaian " Implementasi Aplikasi Android Dalam Sistem Informasi Senayan Di Perpustakaan ITN Malang Kampus 2".

NO	PERTANYAAN	NILAI
1	Materi informasi pada aplikasi	C
2	Tampilan di file hosting	B
3	Kelayakan system aplikasi	B
4	Menu pada aplikasi	B

Ket:

SB = Sangat Baik

B = Baik

C = Cukup

K = Kurang

Tanda Tangan
User Pengguna

Nama : Erwin Syahputra

Pekerjaan : Pegawai Perpustakaan Kab. Manggarai NTT

Berikan penilaian sesuai dengan grade yang sesuai pada kriteria penilaian dengan keterangan yang sesuai pada table di bawah ini, untuk penilaian " Implementasi Aplikasi Android Dalam Sistem Informasi Senayan Di Perpustakaan ITN Malang Kampus 2".

NO	PERTANYAAN	NILAI
1	Materi informasi pada aplikasi	C
2	Tampilan di file hosting	B
3	Kelayakan system aplikasi	B
4	Menu pada aplikasi	B

Ket:

SB = Sangat Baik

B = Baik

C = Cukup

K = Kurang

Tanda Tangan
User Pengguna

Nama : Christian Rizky

Pekerjaan : Wiraswasta

Berikan penilaian sesuai dengan grade yang sesuai pada kriteria penilaian dengan keterangan yang sesuai pada table di bawah ini, untuk penilaian " Implementasi Aplikasi Android Dalam Sistem Informasi Senayan Di Perpustakaan ITN Malang Kampus 2".

NO	PERTANYAAN	NILAI
1	Materi informasi pada aplikasi	B
2	Tampilan di file hosting	C
3	Kelayakan system aplikasi	B
4	Menu pada aplikasi	B

Ket:

SB = Sangat Baik

B = Baik

C = Cukup

K = Kurang

Tanda Tangan
User Pengguna

Nama : Ardianto Pranata

Pekerjaan : PNS

Berikan penilaian sesuai dengan grade yang sesuai pada kriteria penilaian dengan keterangan yang sesuai pada table di bawah ini, untuk penilaian " Implementasi Aplikasi Android Dalam Sistem Informasi Senayan Di Perpustakaan ITN Malang Kampus 2".

NO	PERTANYAAN	NILAI
1	Materi informasi pada aplikasi	B
2	Tampilan di file hosting	B
3	Kelayakan system aplikasi	B
4	Menu pada aplikasi	B

Ket:

SB = Sangat Baik

B = Baik

C = Cukup

K = Kurang

**Tanda Tangan
User Pengguna**

Nama : Faizal Setiawan

Pekerjaan : Mahasiswa

Berikan penilaian sesuai dengan grade yang sesuai pada kriteria penilaian dengan keterangan yang sesuai pada table di bawah ini, untuk penilaian " Implementasi Aplikasi Android Dalam Sistem Informasi Senayan Di Perpustakaan ITN Malang Kampus 2".

NO	PERTANYAAN	NILAI
1	Materi informasi pada aplikasi	B
2	Tampilan di file hosting	B
3	Kelayakan system aplikasi	B
4	Menu pada aplikasi	C

Ket:

SB = Sangat Baik

B = Baik

C = Cukup

K = Kurang

Tanda Tangan
User Pengguna

LAMPIRAN

И А Я П Р М А И



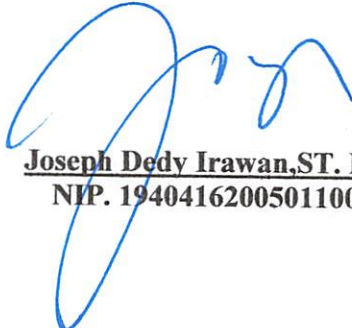
INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo Km. 2 Malang

FORMULIR BIMBINGAN SKRIPSI

Nama : Julianus Kristoforus Kame
NIM : 08.18.011
Masa Bimbingan : 21 Oktober 2013 – 19 Februari 2014
Judul Skripsi : **Implementasi Aplikasi Android Dalam Sistem Informasi Senayan Di Perpustakaan ITN Malang Kampus 2**

NO	TANGGAL	URAIAN	PARAF PEMBIMBING
1	11 Desember 2013	Konsultasi Bab I, II, III	
2	09 Januari 2014	Revisi bab II & III	
3	10 Januari 2014	Revisi Bab IV & V	
4	13 Januari 2013	Acc Makalah Seminar Hasil	
5	17 Februari 2014	Acc Kompre	
6	02 April 2014	Acc Laporan	

Malang, April 2014
Dosen Pembimbing I,



Joseph Dedy Irawan, ST. MT
NIP. 19404162005011002



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo Km. 2 Malang

FORMULIR BIMBINGAN SKRIPSI

Nama : Julianus Kristoforus Kame
NIM : 08.18.011
Masa Bimbingan : 09 Oktober 2013 – 20 Februari 2014
Judul Skripsi : **Implementasi Aplikasi Android Dalam Sistem Informasi Senayan Di Perpustakaan ITN Malang Kampus 2**

NO	TANGGAL	URAIAN	PARAF PEMBIMBING
1	04 Januari 2014	Revisi Bab I, II	
2	06 Januari 2014	Acc Bab I & III	
3	08 Januari 2014	Revisi Bab IV & V	
4	09 Januari 2014	Acc Bab IV & V	
5	11 Januari 2014	Revisi Makalah Seminar Hasil	
6	13 Januari 2014	Acc Makalah Seminar Hasil	
7	18 Februari 2014	Acc Kompre	
8	02 April 2014	Acc Laporan	

Malang, April 2014
Dosen Pembimbing II,

Sonny Prasetyo, ST.MT
NIP.P. 1031000433



FORMULIR PERBAIKAN SKRIPSI

Nama : Julianus Kristoforus Kame
NIM : 08.18.011
Masa Bimbingan : 09 Oktober 2013 – 20 Februari 2014
Judul Skripsi : **Implementasi Aplikasi Android Dalam Sistem Informasi Senayan Di Perpustakaan ITN Malang Kampus 2.**

TANGGAL	PENGUJI	URAIAN	PARAF
20 Februari 2014	I	1. Revisi Program Tambahkan Menu Tentang Kami	
	II	1. Perbaiki program nama buku harus lengkap 2. Referensi pustaka dari landasan teori 3. Perbaiki gambar harus dari smart phone 4. Perbaiki daftar pustaka minimal 10 sumber 5. Penjelasan dari flowchart	

Anggota Penguji

Penguji I

Ali Maharudi, BEng, PhD
NIP.P. 1031000429

Penguji II

Karina Auliasari, ST,M.Eng
NIP. P.1031000426

Mengetahui,

Dosen Pembimbing I

Joseph Dedi Irawan ST.MT
NIP.Y.19740416200511001

Dosen Pembimbing II

Sonny Prasetyo, ST.MT
NIP.P. 1031000433

STATE OF CALIFORNIA
COUNTY OF [illegible]

[illegible signature]

[illegible text]

STATE OF CALIFORNIA
COUNTY OF [illegible]

[illegible signature]

[illegible text]

STATE OF CALIFORNIA
COUNTY OF [illegible]

[illegible signature]

STATE OF CALIFORNIA
COUNTY OF [illegible]

[illegible signature]



[illegible text]

[illegible text block]

[illegible text]

[illegible signature]

[illegible text block]

[illegible text block]

[illegible text]

[illegible text]

[illegible text block]



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo Km. 2 Malang

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama : Julianus Kristoforus Kame
NIM : 08.18.011
Masa Bimbingan : 19 Oktober 2013 – 19 April 2014
Judul Skripsi : **Implementasi Aplikasi Android Dalam Sistem Informasi Senayan Di Perpustakaan ITN Malang Kampus 2**

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :

Hari : Kamis
Tanggal : 20 Februari 2014
Nilai : 78,42 (B+)

Panitia Ujian Skripsi
Ketua Majelis Penguji


Joseph Dedy Irawan, ST, MT
NIP. 19740416 2005011002

Anggota Penguji

Penguji I


Ali Mahmudi, BEng, PhD
NIP. P. 1031000429

Penguji II



Karina Auliasari, ST, M.Eng
NIP.P. 1031000426

LAMPIRAN

SCRIPT IMPLEMENTASI APLIKASI ANDROID DALAM SISTEM INFORMASI SENAYAN DI PERPUSTAKAAN ITN MALANG KAMPUS 2

Source Code Untuk Tampilan Mengakses Data Pertama Kali

```
package com.perpustakaanonline;

import java.io.BufferedReader;
import java.io.InputStream;
import java.io.InputStreamReader;

import org.apache.http.HttpResponse;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.impl.client.DefaultHttpClient;
import org.json.JSONArray;
import org.json.JSONObject;

import android.app.Activity;
import android.content.Intent;
import android.graphics.Typeface;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

public class Login extends Activity {
    private EditText txt_username, txt_password;
    private Button btn_login, btn_daftar_petugas,
    btn_daftar_anggota;
    private TextView lbl_user, lbl_lock;

    private int id_user, akses;
    private String hasil = "", username = "", password =
    "";
    private JSONObject json_obj;
    // private String url =
    "http://10.0.2.2/perpustakaan/login.php";
    private String url =
    "http://skripsiperpus.net76.net/perpustakaan/login.php";
    private Intent i = null;

    @Override
```

```

        Typeface font = Typeface.createFromAsset(
getAssets(), "fontawesome-webfont.ttf" );

        txt_username = (EditText)
findViewById(R.id.txt_username);
        txt_password = (EditText)
findViewById(R.id.txt_password);
        btn_login = (Button)
findViewById(R.id.btn_login);
        btn_daftar_petugas = (Button)
findViewById(R.id.btn_daftar_petugas);
        btn_daftar_anggota = (Button)
findViewById(R.id.btn_daftar_anggota);
        lbl_user = (TextView)
findViewById(R.id.lbl_user);
        lbl_lock = (TextView)
findViewById(R.id.lbl_lock);

        lbl_user.setTypeface(font);
        lbl_lock.setTypeface(font);

        btn_daftar_petugas.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                i = new Intent(Login.this,
DaftarPetugas.class);
                startActivity(i);
            }
        });

        btn_daftar_anggota.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                i = new Intent(Login.this,
DaftarAnggota.class);
                startActivity(i);
            }
        });

        btn_login.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                username =
txt_username.getText().toString();
                password =
txt_password.getText().toString();

                if(validasi(username, password)){
                    hasil = getRequest(url,
username, password);
                    try {

```

```

        proses_login();
    } catch (Exception e) {
        Toast.makeText(Login.this,
"Username atau Password Salah",
Toast.LENGTH_SHORT).show();
    }
}
});
}

private boolean validasi(String user, String pass){
if(!user.equals("") || !pass.equals("")){
    return true;
}
else{
    return false;
}
}

private String getRequest(String url, String user,
String pass){
    String bantu = null;
    HttpClient client = new DefaultHttpClient();
    HttpGet request = new
HttpGet(url+"?username="+user+"&password="+pass);
    try{
        HttpResponse response =
client.execute(request);
        bantu = tampilData(response);
    }
    catch(Exception e){
        Toast.makeText(this,"Koneksi Gagal:
"+e.getMessage(), Toast.LENGTH_SHORT).show();
    }

    return bantu;
}

private static String tampilData(HttpResponse
respon){
    String hasil = "";

    try{
        InputStream in =
respon.getEntity().getContent();
        BufferedReader read = new BufferedReader(new
InputStreamReader(in));

        hasil = read.readLine();

        in.close();
    }
}

```

```

        catch(Exception e){
            e.getMessage();
        }
        return hasil;
    }
    private void proses_login() throws Exception{
        json_obj = new JSONObject(hasil);
        JSONArray menuitemArray =
json_obj.getJSONArray("user");
        id_user =
menuitemArray.getJSONObject(0).getInt("id");
        akses =
menuitemArray.getJSONObject(0).getInt("akses");

        i = new Intent(Login.this, MenuUtama.class);
        i.putExtra("id_user", id_user);
        i.putExtra("akses", akses);

        startActivity(i);
        init();
    }

    private void init(){
        txt_username.setText(null);
        txt_password.setText(null);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        MenuInflater inflater = getMenuInflater();
        inflater.inflate(R.layout.optionmenu_login,
menu);

        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        switch (item.getItemId()) {
            case R.id.mn_tentang:
                i = new Intent(this, Tentang.class);
                startActivity(i);
                break;
            case R.id.mn_keluar:
                finish();
                break;
        }
        return true;
    }
}

```

SOURCE CODE DAFTAR ANGGOTA DAN PETUGAS

```
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class DaftarAnggota extends Activity implements
View.OnClickListener{
    private EditText txt_jns_id, txt_no_id, txt_nama,
txt_username, txt_password, txt_konf_password,
        txt_alamat, txt_kecamatan, txt_kabupaten,
txt_provinsi, txt_telepon, txt_email;
    private Button btn_daftar, btn_hapus_jns_id,
btn_hapus_no_id, btn_hapus_nama,
        btn_hapus_username, btn_hapus_password,
btn_hapus_konf_password, btn_hapus_alamat,
        btn_hapus_kecamatan, btn_hapus_kabupaten,
btn_hapus_provinsi, btn_hapus_telepon, btn_hapus_email;

    private String jns_id = "", no_id = "", nama = "",
username = "", password = "", konf_password = "",
        alamat = "", kecamatan = "", kabupaten =
"", provinsi = "", telepon = "", email = "";
    private String url =
"http://skripsiperpus.net76.net/perpustakaan/anggota.php?p
ilih=1";
// private String url =
"http://10.0.2.2/perpustakaan/anggota.php?pilih=1";
    private Intent i = null;

    @Override

    protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
        setContentView(R.layout.daftar_anggota);
    }
```



```
        txt_jns_id = (EditText)
findViewById(R.id.txt_jenis_id);
        txt_no_id = (EditText)
findViewById(R.id.txt_no_id);
        txt_nama = (EditText)
findViewById(R.id.txt_nama);
        txt_username = (EditText)
findViewById(R.id.txt_username);
        txt_password = (EditText)
findViewById(R.id.txt_password);
        txt_konf_password = (EditText)
findViewById(R.id.txt_konf_password);
        txt_alamat = (EditText)
findViewById(R.id.txt_alamat);
        txt_kecamatan = (EditText)
findViewById(R.id.txt_kecamatan);
        txt_kabupaten = (EditText)
findViewById(R.id.txt_kabupaten);
        txt_provinsi = (EditText)
findViewById(R.id.txt_provinsi);
        txt_telepon = (EditText)
findViewById(R.id.txt_telepon);
        txt_email = (EditText)
findViewById(R.id.txt_email);
        btn_daftar = (Button)
findViewById(R.id.btn_daftar);
        btn_hapus_jns_id = (Button)
findViewById(R.id.btn_hapus_jns_id);
        btn_hapus_no_id = (Button)
findViewById(R.id.btn_hapus_no_id);
        btn_hapus_nama = (Button)
findViewById(R.id.btn_hapus_nama);
        btn_hapus_username = (Button)
findViewById(R.id.btn_hapus_username);
        btn_hapus_password = (Button)
findViewById(R.id.btn_hapus_password);
        btn_hapus_konf_password = (Button)
findViewById(R.id.btn_hapus_konf_password);
        btn_hapus_alamat = (Button)
findViewById(R.id.btn_hapus_alamat);
        btn_hapus_kecamatan = (Button)
findViewById(R.id.btn_hapus_kecamatan);
        btn_hapus_kabupaten = (Button)
findViewById(R.id.btn_hapus_kabupaten);
        btn_hapus_provinsi = (Button)
findViewById(R.id.btn_hapus_provinsi);
```

DETAIL BUKU

```
package com.perpusatakaanonline;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.util.ArrayList;

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.HttpClient;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.message.BasicNameValuePair;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class DaftarBuku extends Activity implements
View.OnClickListener{
    private EditText txt_judul, txt_pengarang,
txt_penerbit,
    txt_tahun, txt_hal, txt_kategori, txt_abstrak,
txt_jumlah;
    private Button btn_simpan, btn_hapus_judul,
btn_hapus_pengarang, btn_hapus_penerbit,
    btn_hapus_tahun, btn_hapus_hal,
btn_hapus_kategori, btn_hapus_abstrak, btn_hapus_jumlah;

    private String judul, pengarang, penerbit, tahun,
hal, kategori, abstrak, jumlah;
    // private String url =
"http://10.0.2.2/perpustakaan/buku.php";
    private String url =
"http://skripsiperpus.net76.net/perpustakaan/buku.php";
    private Intent i = null;

    @Override

    protected void onCreate(Bundle savedInstanceState) {
```

LIST BUKU DAN NAMA BUKU

```
        super.onCreate(savedInstanceState);
        setContentView(R.layout.daftar_buku);

        txt_judul = (EditText)
findViewById(R.id.txt_judul);
        txt_pengarang = (EditText)
findViewById(R.id.txt_pengarang);
        txt_penerbit = (EditText)
findViewById(R.id.txt_penerbit);
        txt_tahun = (EditText)
findViewById(R.id.txt_tahun);
        txt_hal = (EditText)
findViewById(R.id.txt_hal);
        txt_kategori = (EditText)
findViewById(R.id.txt_kategori);
        txt_abstrak = (EditText)
findViewById(R.id.txt_abstrak);
        txt_jumlah = (EditText)
findViewById(R.id.txt_jumlah);
        btn_simpan = (Button)
findViewById(R.id.btn_simpan);
        btn_hapus_judul = (Button)
findViewById(R.id.btn_hapus_judul);
        btn_hapus_pengarang = (Button)
findViewById(R.id.btn_hapus_pengarang);
        btn_hapus_penerbit = (Button)
findViewById(R.id.btn_hapus_penerbit);
        btn_hapus_tahun = (Button)
findViewById(R.id.btn_hapus_tahun);
        btn_hapus_hal = (Button)
findViewById(R.id.btn_hapus_hal);
        btn_hapus_kategori = (Button)
findViewById(R.id.btn_hapus_kategori);
        btn_hapus_abstrak = (Button)
findViewById(R.id.btn_hapus_abstrak);
        btn_hapus_jumlah = (Button)
findViewById(R.id.btn_hapus_jumlah);

        btn_simpan.setOnClickListener(this);
        btn_hapus_judul.setOnClickListener(this);
        btn_hapus_pengarang.setOnClickListener(this);
        btn_hapus_penerbit.setOnClickListener(this);
        btn_hapus_tahun.setOnClickListener(this);
        btn_hapus_hal.setOnClickListener(this);
        btn_hapus_kategori.setOnClickListener(this);
        btn_hapus_abstrak.setOnClickListener(this);
        btn_hapus_jumlah.setOnClickListener(this);
    }
}
```

```

        if(judul.equals("")||pengarang.equals("")||penerbit.e
quals("")||kategori.equals("")||abstrak.equals("")){
            Toast.makeText(this, "Lengkapi semua data",
Toast.LENGTH_SHORT).show();
            return false;
        }
        else{
            return true;
        }
    }

    private void manage_buku(String judul, String
pengarang, String penerbit, String tahun,
        String hal, String kategori, String
abstrak, String jumlah) {
        HttpClient httpClient = new
DefaultHttpClient();
        HttpPost httpPost = new HttpPost(url);

        ArrayList<NameValuePair> param = new
ArrayList<NameValuePair>();
        param.add(new BasicNameValuePair("pilih",
"1"));
        param.add(new BasicNameValuePair("judul",
judul));
        param.add(new BasicNameValuePair("pengarang",
pengarang));
        param.add(new BasicNameValuePair("penerbit",
penerbit));
        param.add(new BasicNameValuePair("tahun",
tahun));
        param.add(new BasicNameValuePair("hal", hal));
        param.add(new BasicNameValuePair("kategori",
kategori));
        param.add(new BasicNameValuePair("abstrak",
abstrak));
        param.add(new BasicNameValuePair("jumlah",
jumlah));
        try {
            httpPost.setEntity(new
UrlEncodedFormEntity(param));
            HttpResponse httpResponse =
httpClient.execute(httpPost);
            HttpEntity httpEntity =
httpResponse.getEntity();
            InputStream in = httpEntity.getContent();
            BufferedReader read = new
BufferedReader(new InputStreamReader(in));

            String baris = read.readLine();

```

```

        Toast.makeText(this, "Data buku
berhasil dibuat", Toast.LENGTH_SHORT).show();

        try {
            i = new Intent(DaftarBuku.this,
ListBuku.class);
            startActivity(i);
            finish();
        } catch (Exception e) {
            Toast.makeText(this, "Error:
"+e.getMessage(), Toast.LENGTH_SHORT).show();
        }
        else {
            Toast.makeText(this, "Judul atau
Abstrak sudah digunakan", Toast.LENGTH_SHORT).show();
        }
        catch (ClientProtocolException e) {
            Toast.makeText(this, "Error:
"+e.getMessage(), Toast.LENGTH_SHORT).show();
        }
        catch (IOException e) {
            Toast.makeText(this, "Error:
"+e.getMessage(), Toast.LENGTH_SHORT).show();
        }
    }

    @Override
    public void onClick(View v) {
        switch(v.getId()){
            case R.id.btn_simpan:
                judul =
txt_judul.getText().toString();
                pengarang =
txt_pengarang.getText().toString();
                penerbit =
txt_penerbit.getText().toString();
                tahun =
txt_tahun.getText().toString();
                hal = txt_hal.getText().toString();
                kategori =
txt_kategori.getText().toString();
                abstrak =
txt_abstrak.getText().toString();
                jumlah =
txt_jumlah.getText().toString();
                if(validasi(judul, pengarang,
penerbit, kategori, abstrak)){
                    manage_buku(judul, pengarang,
penerbit, tahun, hal, kategori, abstrak, jumlah);
                }
                break;

```

```
case R.id.btn_hapus_judul:
    txt_judul.setText(null);
break;
case R.id.btn_hapus_pengarang:
    txt_pengarang.setText(null);
break;
case R.id.btn_hapus_penerbit:
    txt_penerbit.setText(null);
break;
case R.id.btn_hapus_tahun:
    txt_tahun.setText(null);
break;
case R.id.btn_hapus_hal:
    txt_kategori.setText(null);
break;
case R.id.btn_hapus_abstrak:
    txt_abstrak.setText(null);
break;
case R.id.btn_hapus_jumlah:
    txt_jumlah.setText(null);
break;
```

```
}
```

```
}
```

```
}
```

LIST PEMINJAMAN DAN PENGEMBALIAN BUKU

```
public class ListPeminjaman extends Activity {
    private ListView daftar;
    private ArrayList<dataBuku> list_buku;

    private String hasil = "";
    private JSONObject json_obj;
    // private String url =
    "http://10.0.2.2/perpustakaan/transaksi.php",
    //         url_update_pinjam =
    "http://10.0.2.2/perpustakaan/transaksi.php",
    //         url_update_buku =
    "http://10.0.2.2/perpustakaan/buku.php";
    private String url =
    "http://skripsiperpus.net76.net/perpustakaan/transaksi.php
    ",
        url_update_pinjam =
    "http://skripsiperpus.net76.net/perpustakaan/transaksi.php
    ",
        url_update_buku =
    "http://skripsiperpus.net76.net/perpustakaan/buku.php";
    private String[] bantu_judul = null,
    bantu_tgl_kembali = null, bantu_tgl_pinjam = null;
    private int[] bantu_denda = null, bantu_id_pinjam =
    null, bantu_id_buku = null;
    private int id_anggota, akses, bantu_jumlah;

    private ProgressDialog progressBar;
    private int progressBarStatus = 0;
    private Handler progressBarHandler = new Handler();
    private int persen_loading;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.list_peminjaman);

        persen_loading = 1;
        progressBar();
        baca_id_anggota();

        try {
            baca_buku();
        } catch (Exception e) {
            Toast.makeText(this, "Tidak ada buku",
            Toast.LENGTH_SHORT).show();
        }
        persen_loading = 10;
    }

    private void baca_id_anggota(){
```

```

        Bundle extras = getIntent().getExtras();

        if (extras != null) {
            id_anggota = extras.getInt("id_anggota");
            akses = extras.getInt("akses"); //0=petugas,
1=anggota
        }
    }

    private String getRequest(){
        String bantu = null;

        HttpClient client = new DefaultHttpClient();
        HttpGet request = new
HttpGet(url+"?pilih=3&id_anggota="+id_anggota);

        try{
            HttpResponse response =
client.execute(request);
            bantu = tampilData(response);
        }
        catch(Exception e){
            Toast.makeText(this, "Koneksi Gagal:
"+e.getMessage(), Toast.LENGTH_SHORT).show();
        }

        return bantu;
    }

    private static String tampilData(HttpResponse
respon){
        String hasil = "";

        try{
            InputStream in =
respon.getEntity().getContent();
            BufferedReader read = new BufferedReader(new
InputStreamReader(in));

            hasil = read.readLine();

            in.close();
        }
        catch(Exception e){
            e.getMessage();
        }

        return hasil;
    }

    private void baca_buku() throws Exception{
        list_buku = new ArrayList<dataBuku>();

        hasil = getRequest();
    }

```



```

        json_obj = new JSONObject(hasil);
        JSONArray menuitemArray =
json_obj.getJSONArray("pinjam");

        int jum_data = menuitemArray.length();
        bantu_jumlah = jum_data;

        bantu_id_pinjam = new int[jum_data];
        bantu_id_buku = new int[jum_data];
        bantu_judul = new String[jum_data];
        bantu_tgl_kembali = new String[jum_data];
        bantu_tgl_pinjam = new String[jum_data];
        bantu_denda = new int[jum_data];

        for(int i=0; i<jum_data; i++){
            bantu_id_pinjam[i] =
menuitemArray.getJSONObject(i).getInt("id");
            bantu_id_buku[i] =
menuitemArray.getJSONObject(i).getInt("id_buku");
            bantu_judul[i] =
menuitemArray.getJSONObject(i).getString("judul");
            bantu_tgl_kembali[i] =
menuitemArray.getJSONObject(i).getString("tgl_kembali");
            bantu_tgl_pinjam[i] =
menuitemArray.getJSONObject(i).getString("tgl_pinjam");
            bantu_denda[i] =
menuitemArray.getJSONObject(i).getInt("denda");

            list_buku.add(new
dataBuku(bantu_judul[i],bantu_tgl_kembali[i],bantu_tgl_pin
jam[i],bantu_denda[i]));

            daftar = (ListView)
findViewById(R.id.list_peminjaman_anggota);
            daftar.setAdapter(new BukuItemAdapter(this,
                android.R.layout.simple_list_item_1,
list_buku));
        }
        if(akses == 0){
            daftar.setOnItemClickListener(new
ListListener(bantu_id_pinjam, bantu_id_buku));
        }
    }

    private class dataBuku {
        private String judul, tgl_kembali, tgl_pinjam;
        private int denda;

        private dataBuku(String judul, String
tgl_kembali, String tgl_pinjam, int denda) {
            this.judul = judul;
            this.tgl_kembali = tgl_kembali;

```

```

        this.tgl_pinjam = tgl_pinjam;
        this.denda = denda;
    }
}

private class BukuItemAdapter extends
ArrayAdapter<dataBuku> {
    private ArrayList<dataBuku> buku;

    public BukuItemAdapter(Context context, int
textViewResourceId, ArrayList<dataBuku> buku) {
        super(context, textViewResourceId, buku);
        this.buku = buku;
    }

    @Override
    public View getView(int position, View
convertView, ViewGroup parent) {
        View v = convertView;
        if (v == null) {
            LayoutInflater vi =
(LayoutInflater) getSystemService(Context.LAYOUT_INFLATER_S
ERVICE);
            v =
vi.inflate(R.drawable.list_adapter_peminjaman_anggota,
null);
        }

        dataBuku list_job = buku.get(position);
        if (list_job != null) {
            TextView lbl_judul= (TextView)
v.findViewById(R.id.lbl_judul_buku);
            TextView lbl_tgl_kembali = (TextView)
v.findViewById(R.id.lbl_tgl_kembali);
            TextView lbl_tgl_pinjam = (TextView)
v.findViewById(R.id.lbl_tgl_pinjam);
            TextView lbl_denda = (TextView)
v.findViewById(R.id.lbl_denda);

            if (lbl_judul != null) {
                lbl_judul.setText("Judul:
"+list_job.judul);
            }
            if (lbl_tgl_kembali != null) {
                lbl_tgl_kembali.setText("Tgl.
Kembali: "+list_job.tgl_kembali);
            }
            if (lbl_tgl_pinjam != null) {
                lbl_tgl_pinjam.setText("Tgl.
Pinjam: "+list_job.tgl_pinjam);
            }
            if (lbl_denda != null) {
                lbl_denda.setText("Denda:
"+list_job.denda);
            }
        }
    }
}

```

MENU TENTANG

```
package com.perpusatakaanonline;

import android.app.Activity;
import android.os.Bundle;
import android.widget.TextView;

public class Tentang extends Activity {
    private TextView lbl_tentang;
    private StringBuilder str;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.tentang);

        lbl_tentang = (TextView)
findViewById(R.id.lbl_tentang);

        str = new StringBuilder();
        str.append("Aplikasi Perpustakaan Online");
        str.append("\n\nITN Malang");

        lbl_tentang.setText(str);
    }
}
```