



TUGAS AKHIR

**SISTEM INFORMASI RUMAH KOS DISEKITAR KAMPUS IAIN
BATUSANGKAR BERBASIS ANDROID**

*Diajukan Kepada Jurusan Manajemen Informatika D.III
Sebagai Syarat Mencapai Gelar Ahli Madya (A.Md)
Dalam Ilmu Manajemen Informatika*

YESI WIRNA ELIZA

NIM. 14 205 121

**JURUSAN MANAJEMEN INFORMATIKA
FAKULTAS EKONOMI DAN BISNIS ISLAM
INSTITUT AGAMA ISLAM NEGERI (IAIN)
BATUSANGKAR**

2018

SURAT PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini :

Nama : Yesi Wirna Eliza
NIM : 14 205 121
Tempat/Tanggal Lahir : Labuh Baru/06 Mei 1995
Fakultas : Ekonomi Dan Bisnis Islam
Jurusan : Manajemen Informatika

Menyatakan dengan sesungguhnya bahwa Tugas Akhir saya yang berjudul **“SISTEM INFORMASI RUMAH KOS DISEKITAR KAMPUS IAIN BATUSANGKAR BERBASIS ANDROID“** adalah benar karya saya sendiri bukan plagiat kecuali yang dicantumkan sumbernya.

Apabila di kemudian hari terbukti bahwa karya ilmiah ini plagiat, maka saya bersedia menerima sanksi sesuai dengan ketentuan perundang-undangan yang berlaku. Demikian pernyataan ini saya buat dengan sebenarnya untuk digunakan sebagaimana mestinya.

Batusangkar, 05 Februari 2018

Saya yang Menyatakan



Yesi Wirna Eliza
Nim.14 205 121

PERSETUJUAN PEMBIMBING

Pembimbing Penulis Tugas Akhir atas Nama : YESI WIRNA ELIZA,
Nim : 14 205 121 dengan judul, "SISTEM INFORMASI RUMAH KOS
DISEKITAR KAMPUS IAIN BATUSANGKAR BERBASIS ANDROID"
memandang bahwa Tugas Akhir yang bersangkutan telah memenuhi persyaratan
ilmiah dan dapat disetujui untuk dilanjutkan ke Sidang Munaqasyah.

Dengan persetujuan ini diberikan untuk dapat dipergunakan sebagaimana
mestinya.

Batusangkar, 05 Februari 2018

Ketua Jurusan
Manajemen Informatika,

Pembimbing,


Iswandi, M. Kom
Nip. 19700510 200312 1004





Iswandi, M. Kom
Nip. 19700510 200312 1004

Mengetahui,
Dekan Fakultas Ekonomi Dan Bisnis Islam
IAIN Batusangkar


Dr. Ulva Atsani, S.H., M.Hum
Nip. 19750303 199903 1 004

PENGESAHAN TIM PENGUJI

Tugas Akhir yang berjudul "SISTEM INFORMASI RUMAH KOS DISEKITAR KAMPUS IAIN BATUSANGKAR BERBASIS ANDROID" oleh YESI WIRNA ELIZA, NIM. 14 205 121, telah diajukan pada sidang munaqasyah Institut Agama Islam Negeri (IAIN) Batusangkar, Sabtu 17 Februari 2018 dan dinyatakan telah diterima sebagai salah satu syarat untuk mencapai gelar Ahli Madya Program Diploma III (D.III) Manajemen Informatika.

No.	Nama Penguji	Jabatan	Tanda Tangan	Tanggal
1.	Iswandi, M.Kom Nip. 19700510 200312 1 004	Ketua Sidang		20/2-18
2.	Fitra Kasma Putra, M.Kom Nip. 19850207 201503 1 004	Anggota		20 Februari 2018
3.	Lidya Rahmi, M.Pd.T Nip -	Anggota		20 Feb 2018

Mengetahui,
Dekan Fakultas Ekonomi dan Bisnis Islam
IAIN Batusangkar



Dr. Usk Atsani, S.H., M.Hum
NIP. 19750303 199903 1 004

HALAMAN PERSEMBAHAN

“Barang siapa keluar untuk mencari ilmu maka ia berada di jalan Allah.SWT”

(HR. Tirmidzi)

“Allah menganugerahkan Al hikmah (ilmu yang berguna) kepada siapa yang dikehendaki-Nya. Dan barangsiapa yang dianugerahi hikmah, ia benar-benar telah dianugerahi karunia yang banyak. Dan hanya orang-orang yang berakal lah yang dapat mengambil pelajaran (dari firman Allah).”

(Q.S. Al-Baqarah: 269)

Halaman persembahan ini mulai ditulis pada tanggal 20 Februari 2018 jam 22.00 WIB dimana Sidang Tugas Akhir saya pada Kamis 15 Februari 2018, saat itu saya dinyatakan tidak lulus sehingga membuat batin terguncang begitu dalam sampai membuat suasana histeris dikarenakan hati yang tidak terima bahwasanya saya dinyatakan tidak lulus dengan rata-rata nilai yang diberikan oleh penguji hanya dengan angka 5 dan cuma satu penguji yang memberi angka 6. Tentunya dengan nilai seperti itu rasanya tidak membayar jerih payah saya selama ini terlebih yang membuat air mata bercucuran yaitu dengan sikap penguji yang mengerjai saya dengan pergi begitu saja meninggalkan ruang sidang.

*Dan yang paling menjadi kesan terlucu bagi saya pribadi adalah suasana ketika saya harus berteriak-teriak dan menghukum diri sendiri karena keputusan sidang yang tidak pasti saya terima sampai-sampai kejadian dalam sidang itu menjadi trending topic digroup yang menyatakan **Telah Terjadi Drama FTV**. hahaha tapi sudahlah jika diceritakan terlalu panjang...yang jelas Alhamdulillah nilai yang saya terima cukup buat saya Bahagia...*

Rasa Syukur “Alhamdulillah” kepada Allah swt yang tiada taranya, begitu besar Nikmat, Rahmat dan Karunia yang telah Engkau berikan kepada ku. Aku percaya bahwasanya akan ada kebahagiaan, akan ada pelangi setelah hujan. Sebab Engkau tidak akan memberi cobaan melewati batas kemampuan umat-Mu. Dan Engkau tidak akan pernah meninggalkan umat-Mu. Sebab sebaik apapun rencana manusia lebih baiklah rencana mu ya Rob.

TERUNTUK UMAK AYAH TERCINTA

*Terimakasih banyak Umak Ayah untuk segala do'a yang tiada hentinya kau hembuskan disetiap sujudmu, terimakasih untuk segala perjuangan yang teramat besar yang kau perjuangkan untuk uni. Semoga Allah selalu limpahkan Rezeki, Kesehatan, dan Umur Panjang untuk Umak dan Ayah. Uni menyadari bahwa apa yang uni capai saat ini belum mampu mengantikan sejuta keringat ayah yang berjuang untuk keluarga terutama untuk uni. Maaf uni juga belum mampu memberi kebahagiaan untuk Umak yang slalu berusaha memenuhi apapun yang uni butuhkan, yang tak pernah henti berdo'a untuk uni. **Umak adalah Wanita***

Terhebat Sedunia Ayah adalah Lelaki Terbaik Sedunia. Untuk saat ini uni baru mampu memberi sedikit hadiah yaitu gelar A.Md yang uni capai walau tidak sesuai target namun uni berharap dapat menganti lelah Umak dan Ayah menjadi sebuah Senyuman yang nantinya mampu uni ganti dengan sebuah Kebahagiaan. Aamiin

TERUNTUK SAUDARAKU

*Terimakasih Uda Sisulung (**Andri Tanjung**) yang slalu menjadi pedoman kami adik-adikmu dibalik dirimu yang kasar, ganas, jutek, cuek namun kau adalah sosok uda yang perhatian dengan cara mu sendiri. Uda yang slalu mencemooh uni, sekarang uni sudah mendapatkan A.Md itu uda. Terimakasih kasih juga untuk uda (**Agus Wirman Afrigon**) uda yang masa bodo sama adik-adiknya namun sekali peduli bahagia tiada henti. Terimakasih untuk sibungsu (**Wira Safitri**) kalau dekat tak pernah akur jikalau jauh dia pun rindu hahaha dia adalah adik sekaligus teman terbaik yang perhatiannya cuman bisa kalau udah jauh. Makasih saudara-saudaraku tercinta yang slalu memberi uni semangat untuk mencapai semuanya.*

Uni saying kalian love you all...

*Selanjutnya terimakasih kakak ipar (**Eka**), dan ipar baru kak (**Sisi**), dan juga ni (**Wat**) yang slalu jadi pendengar setia uci, yang nggak pernah bosan ngasih saran dan arahan, makasih udah jadi mak kedua uci setelah mak kandung, uni udah kayak saudara sendiri, makasih uni dan akhirnya uci dapatkan yang sudah uci tangis-tangisi dirumah uni waktu itu. Thanks u so much....*

MANAJEMEN INFORMATIKA

*Terimakasih ici ucapkan kepada Bapak **Iswandi. M.Kom** selaku Ketua Jurusan Manajemen Informatika, sekaligus pembimbing tugas akhir ini, terimakasih pak atas semua dukungan, bimbingan dan arahan yang Bapak berikan. Dan terimakasih juga kepada semua dosen Manajemen Informatika IAIN Batusangkar yang telah memberikan banyak ilmu yang bermanfaat baik saat perkuliahan maupun diluar perkuliahan. Terima kasih kepada teman-teman semua Mahasiswa Manajemen Informatika khususnya angkatan 14 yang telah memberikan semangat dan saran yang bermanfaat, terimakasih untuk kebersamaan, canda tawa, dan motivasi selama ini. Terimakasih untuk para pejuang Maret terutama selama sidang munaqasyah. **Henny Salma** yang harus nangis karna tau mau tampil sidang di hari pertama yang hujan-hujan demi sidangnya esok, **Ega Fulvia** yang minta ditemani ke padang demi sehelai kertas, nonton dilan sampai-sampai pulang diterpa hujan, **Eka Putriyanti**, **Fani Ruktami**, **Fitri Zulkarnain**, **Martono** yang jadi partner kerja dalam acara siding munaqasyah penuh dengan kebersamaan, gila-gilaan. Terimakasih juga untuk geng abal-abal Alam Nofri, **Brians Pratama**, **Depi Supta**, **Amelia Gusti**, **Bismi Hayati**, **Aprila Fista Dwianti** yang slama ini slalu sama-sama akhirnya kita A.Md juga. Walau **Dian Saputra**, **Ananda Audensi Putra**, **Aditya Winata**, **Sukra Alhamda***

belum bisa A.Md bersama kita semua. Semoga cepat nyusul dan tetap semangat ya...

Kepada semua junior-junior Manajemen Informatika makasih banyak, kalian juga harus tetap semangat untuk mencapai A.Md kalian nantinya dan selamat berjuang. Dan Kepada semua yang telah berjasa kepada saya yang tidak dapat saya ucapkan satu persatu. Terimakasih Semuanya...

MY BEST FRIENDS FOREVER

*Untuk Sabahat yang sudah berasa keluarga, terimakasih untuk 3 tahun lebih bersamanya. Incim (**Dian Hurriyyah**) yang udah sama-sama dari SD sampai Kuliah hingga satu kos, terimakasih tetap slalu jadi teman, sahabat, keluarga buat mami cepat nyusul ya ncm... semoga cepat S.E nya. Ante (**Nuri Adiningsih**) makasih juga sudah jadi teman, sahabat, keluarga buat mami walau cuman jalan 3 tahun lebih tapi sudah berasa kenal lama, cepat dapat S.Pd nya juga nte. Tete (**Winda Saputri**) makasih teh udah jadi teman, sahabat, keluarga buat mami semoga nyusul dengan gelar S.Pd nya teh. Kalian semua luar biasa akan banyak ungkapan kata dan cerita jika mami utarakan semuanya disini, yang penting mami sayang kalian semua.*

*Makasih karna slalu memberi semangat, motivasi, dan slalu ada untuk mami Untuk Kakak (**Erdilla Puspita Sari**), Adik (**Emrina Rosyada**), Alink (**Delfi**), Amak (**Aprillia Dwiana Priska**), Ola (**Dola Olivia Veranti**) makasih juga udah slalu memberi semangat mami untuk mencapai ini semua. Semoga S.E cepat didapat ya but dan kembali lah kalian sekompak dulu.*

*Makasih buat Cixmey (**Meicy Sasmitha**), dedeq (**Nadya Yolanda Putri**) yang udah nemanin mami selama di padang sampe panas-panasan, hujan-hujan demi perjuangan mami untuk A.Md. oom (**Dafni Viktorio**), abang (**Dicky Vanesa**), Umak (**Meta Putri Amdani**) cepat selesaikan skripsinya.*

*Makasih juga buat kak (**Yogi Putra Anjaya**) yang slalu ngasih semangat kala adek lagi geluh, makasih udah mau bantu dan nemanin adek dari Pasaman ke Padang yang luntang lantung 3 hari makasih udah jadi kakak terbaik untuk adek. Makasih juga **bang yuam** yang slalu tiap saat ngasih semangat buat ici yang nggak pernah bosan denger ocehan, keluh kesan ici, yang slalu jadi pendengar setia setiap ici cerita, **bang yudha** juga yang udah mau bantu nemanin ici ke solok yang pulangnya hujan-hujan makasih ya bang. **Bang dodon** makasih juga jadi teman gila terkocak, adek malah dipanggil abang. Makasih juga (**Ridho Kurniawan**) slalu jadi adek akak yang setia dengerin akak, yang slalu ngasih semangat, makasih ido...*

*Dan terakhir untuk teristimewa buat kamu (**AB**) makasih untuk segalanya, yang tidak mampu aku ungkapkan semua yang jelas makasih slalu ada walau kadang menyakitkan, walau kadang mengesalkan, dan sangat menjengkelkan. Tapi dibalik itu semua aku mengerti apa maksudmu. Akhirnya pencapaian ku tercapai, tidak*

ada kata terbaik yang bisa ku utarakan yang bisa ku sampaikan tetaplh menjadi kamu untukku.

KOS ABU-ABU

*Makasih buat keluarga kos abu-abu untuk hari-harinya buat adek-adek dikos abu-abu cabe, siti, caca, ulan udah jadi adek geng ribut. Makasih udah slalu ngasih semangat buat mami. **Kuliahnya jangan main-main lagi udah semester tua juga, bentar lagi bakal berjuang juga untuk S.E nya. Kalian bakal merasakan apa yang mami rasakan tapi jangan pernah menyerah ya... tetap semangat oke !!!***

Tepat 21 Februari 2018 jam 22.00 berakhir pula kata persembahan yang saya tulis mohon maaf untuk nama yang tidak dapat saya sebutkan satu persatu yang jelas dihati dan pikiran saya masih teringat dan akan tersimpan semuanya.

KARNA KESEMPURNAAN HANYA MILIK ALLAH SWT



Yesi Wirna Eliza, A.Md

ABSTRAK

Judul Tugas Akhir : **SISTEM INFORMASI RUMAH KOS
DISEKITAR KAMPUS IAIN
BATUSANGKAR BERBASIS ANDROID**

Nama Mahasiswa : **Yesi Wirna Eliza**

Nomor Induk Mahasiswa : **14 205 121**

Jurusan : **Manajemen Informatika**

Dosen Pembimbing : **Iswandi, M.Kom**

Di Kabupaten Tanah Datar khususnya Batusangkar berdiri sebuah Institut Agama Islam Negeri Batusangkar. Semakin tingginya minat masyarakat untuk menempuh pendidikan di IAIN Batusangkar tentu semakin tinggi pula kebutuhan tempat tinggal sementara terlebihnya bagi para pendatang dari luar daerah. Dalam mencari tempat tinggal sementara tentunya cukup kesulitan bagi para pendatang luar daerah untuk mengetahui mengenai rumah kos karena minimnya informasi mengenai lokasi, ketersediaan kamar, fasilitas serta biaya sewa dan lain-lain. Dengan dirancangnya sistem informasi rumah kos ini, yang pertama dapat membantu pencari kos dalam menemukan rumah kos yang diinginkan, kedua untuk pemilik kos dapat mempromosikan rumah kos dan memudahkan dalam mengelola data mengenai rumah kos. Alat bantu perancangan sistem menggunakan UML (*Unified Modeling Language*), sedangkan perangkat lunak yang digunakan dalam sistem adalah menggunakan bahasa pemograman PHP dengan database MySql. Dengan memanfaatkan sistem komputerisasi diharapkan dapat membantu dan menggantikan cara yang kurang efektif dan efisien.

Kata Kunci : *Sistem Informasi, Pemograman PHP, Android, MySql dan UML*

KATA PENGANTAR



Puji syukur penulis haturkan atas kehadiran Allah SWT yang telah melimpahkan rahmat dan karunia-Nya sehingga pembuatan tugas akhir ini dapat terselesaikan. Salawat dan salam buat junjungan umat, yakni Nabi Muhammad SAW yang telah membawa manusia dari alam kebodohan ke alam yang penuh dengan ilmu pengetahuan seperti kita rasakan seperti sekarang ini.

Dalam penulisan tugas akhir ini penulis banyak mendapat bimbingan dan bantuan baik moril maupun materil dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis menyampaikan ucapan terima kasih kepada:

1. Bapak Dr. H. Kasmuri, MA selaku Rektor IAIN Batusangkar.
2. Bapak Dr. Ulya Atsani, S.H., M.Hum selaku Dekan Fakultas Ekonomi dan Bisnis Islam.
3. Bapak Iswandi, M.Kom selaku ketua Jurusan Manajemen Informatika IAIN Batusangkar, sekaligus sebagai pembimbing yang telah memberikan bimbingan dan arahan kepada penulis dalam pembuatan tugas akhir ini.
4. Ibunda tercinta Jusnimar dan Ayahanda tercinta Sawirman yang telah banyak memberikan dorongan, motivasi, semangat, bantuan moril maupun materil, nasehat serta doa yang tiada henti-hentinya.
5. Sanak Famili yang telah banyak memberikan semangat dan dorongan kepada penulis hingga penulis bisa menyelesaikan tugas akhir ini.
6. Teman-teman seperjuangan angkatan (2014) yang telah memberikan saran-saran yang bermanfaat dalam menyelesaikan tugas akhir ini.
7. Pihak-pihak yang tidak dapat penulis sebutkan satu persatu, yang telah memberikan motivasi dan semangat serta sumbangan pemikirannya kepada penulis sehingga selesainya tugas akhir ini.

Penulis sadar bahwasanya tugas akhir ini jauh dari kesempurnaan, oleh karena itu kritikan dan saran yang sifatnya membangun sangat penulis harapkan

untuk kesempurnaan tugas akhir ini. Penulis juga berharap semoga penulisan tugas akhir ini memberikan manfaat kepada kita semua. Amiin...

Akhirnya kepada Allah SWT jualah penulis bermohon dan bersujud semoga keikhlasan yang diberikan akan dibalas-Nya. *Amiin Ya Robbal'alamin.*

Batusangkar, 15 Februari 2018

Penulis

DAFTAR ISI

PERNYATAAN KEASLIAN	
PERSETUJUAN PEMBIMBING	
PENGESAHAN TIM PENGUJI	
ABSTRAK	
KATA PENGANTAR.....	i
DAFTAR ISI.....	iii
DAFTAR GAMBAR.....	vii
DAFTAR TABEL.....	x
BAB I PENDAHULUAN	1
A. Latar Belakang Masalah.....	1
B. Identifikasi Masalah	2
C. Batasan Masalah.....	2
D. Rumusan Masalah	2
E. Tujuan Penelitian.....	2
F. Kegunaan Penelitian	3
G. Metodologi Penelitian	3
H. Sistematika Penulisan.....	4
BAB II LADASAN TEORI	5
A. Konsep Dasar Sistem Informasi.....	5
1. Pengertian Sistem	5
2. Informasi.....	7
3. Sistem Informasi.....	9
B. Android.....	11
1. Sejarah Android.....	11
2. Android Platform Masa Depan.....	13
3. DVM (The Dalvik Virtual Machine).....	15
4. Android SDK (Software Development Kit)	16
5. ADT (Android Development Tools)	17
6. Arsitektur Android.....	18

7. Fundamental Aplikasi.....	21
8. Versi Android	22
C. Unified Modelling Language	25
1. Use Case Diagram	25
2. Sequence dan Collaboration Diagram	27
3. Activity Diagram	27
4. Class Diagram	28
D. Internet, Web dan Jaringan.....	29
1. Internet.....	29
2. Web (World Wide Web)	31
3. Jaringan.....	33
E. Perangkat Lunak Pembangun Sistem	34
1. Database	34
2. PHP	34
3. Bootstrap.....	37
4. JQuery	37
5. Adobe Dreamweaver CS5	39
6. MySQL.....	42
BAB III ANALISA DAN HASIL.....	45
A. Analisa Masalah	45
B. Analisa Aplikasi	45
C. Analisa Kebutuhan Aplikasi.....	46
1. Kebutuhan Perangkat Keras	47
2. Kebutuhan Perangkat Lunak	47
D. Perancangan Sistem.....	47
1. Actor	47
2. Use Case Diagram	48
3. Sequence Diagram.....	49
4. Activity Diagram	53
5. Collaboration Diagram	55
5. Class Diagram	57

7. Struktur Program	58
E. Desain Input.....	58
1. Tampilan Halaman Login Admin.....	59
2. Tampilan Halaman Akun	59
3. Tampilan Halaman Laporan Data Transaksi.....	59
4. Tampilan Halaman Laporan Data Pencairan.....	60
5. Tampilan Halaman Pendaftaran Pemilik Kos	60
6. Tampilan Halaman Login Pemilik Kos	61
7. Tampilan Halaman Add Rumah Kos	61
8. Tampilan Halaman Add Rekening Pemilik Kos	62
9. Tampilan Halaman Booking Kamar.....	62
10. Tampilan Halaman Login Pembayaran.....	63
11. Tampilan Halaman Pembayaran	63
12. Tampilan Halaman Login Cek Status	64
F. Desain Output	64
1. Tampilan Halaman Admin	64
2. Tampilan Halaman Pemilik Kos	65
3. Tampilan Halaman Pencari Kos.....	65
4. Tampilan Halaman List Kos Admin.....	65
5. Tampilan Halaman Rekening Admin.....	66
6. Tampilan Halaman Saldo Admin	66
7. Tampilan Laporan Data Pemilik Kos	66
8. Tampilan Laporan Data Kos	67
9. Tampilan Laporan Data Transaksi	67
10. Tampilan Laporan Pencairan	68
11. Tampilan Halaman Rekening Pemilik	69
12. Tampilan Halaman List Kos Pemilik.....	69
13. Tampilan Halaman Konfirmasi.....	70
14. Tampilan Halaman Saldo Pemilik.....	70
15. Tampilan Halaman List Kos Pencari	70
16. Tampilan Halaman Lihat Kamar	71

17. Tampilan Halaman Help	71
18. Tampilan Bukti Pemesanan.....	72
G. Desain File.....	72
1. Tabel Admin.....	72
2. Tabel Kos.....	73
3. Tabel Konfirmasi.....	74
4. Tabel Pemilik Kos	74
5. Tabel Pencari Kos.....	75
6. Tabel Rekening.....	75
7. Tabel Saldo.....	76
8. Tabel Transaksi	76
BAB IV PENUTUP	78
A. Kesimpulan.....	78
B. Saran	78

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar 2.1	Siklus Informasi	8
Gambar 2.2	Detail Anatomi Android.....	20
Gambar 2.3	Tampilan Halaman Welcome Dreamweaver CS5	40
Gambar 2.4	Tampilan Lembar Kerja Dreamweaver SC5	41
Gambar 2.5	Tampilan Aplication Bar	41
Gambar 2.6	Document Toolbar.....	41
Gambar 3.1	Use Case Diagram	48
Gambar 3.2	Sequence Diagram Login	49
Gambar 3.3	Sequence Diagram Input Admin	49
Gambar 3.4	Sequence Diagram Lihat Kos.....	50
Gambar 3.5	Sequence Diagram Lihat Rekening.....	50
Gambar 3.6	Sequence Diagram Lihat Saldo	50
Gambar 1.7	Sequence Diagram Laporan Data Pemilik Kos.....	51
Gambar 3.8	Sequence Diagram Laporan Data Rumah Kos.....	51
Gambar 3.9	Sequence Diagram Laporan Data Transaksi	51
Gambar 3.10	Sequence Diagram Laporan Data Pencairan	52
Gambar 3.11	Sequence Diagram Input Kos.....	52
Gambar 3.12	Sequence Diagram Input Rekening.....	52
Gambar 3.13	Sequence Diagram Input Booking	53
Gambar 2.14	Sequence Diagram Logout	53
Gambar 3.15	Activity Diagram Pencari Kos	53
Gambar 3.16	Activity Diagram Admin.....	54
Gambar 3.17	Activity Diagram Pemilik Kos	54
Gambar 3.18	Collaboration Diagram Admin.....	56
Gambar 3.19	Collaboration Diagram Laporan.....	56
Gambar 3.20	Collaboration Diagram Input Kos	57
Gambar 3.21	Class Diagram	57
Gambar 3.22	Struktur Diagram.....	58
Gambar 3.23	Desain Input Login Admin.....	59

Gambar 4.24 Desain Input Akun Admin	59
Gambar 5.25 Desain Input Laporan Data Transaksi.....	59
Gambar 3.26 Desain Input Laporan Data Pencairan.....	60
Gambar 3.27 Desain Input Pendaftaran Pemilik Kos	60
Gambar 3.28 Desain Input Login Pemilik Kos	61
Gambar 3.29 Desain Input Kos	61
Gambar 6.30 Desain Input Add Rekening Pemilik Kos	62
Gambar 3.31 Desain Input Booking Kamar.....	62
Gambar 3.32 Desain Input Login Pembayaran	63
Gambar 3.33 Desain Input Pembayaran.....	63
Gambar 3.34 Desain Input Cek Status	64
Gambar 3.35 Desain Output Halaman Admin	64
Gambar 3.36 Desain Output Halaman Pemilik Kos	65
Gambar 3.37 Desain Output Halaman Pencari Kos	65
Gambar 3.38 Desain Output List Kos Pada Admin	65
Gambar 3.39 Desain Output Daftar Rekening Pada Admin	66
Gambar 3.40 Desain Output Saldo Pada Admin.....	66
Gambar 3.41 Desain Output Laporan Data Pemilik Kos	66
Gambar 3.42 Desain Output Laporan Data Kos	67
Gambar 3.43 Desain Output Laporan Data Transaksi Perbulan Perpemilik Kos .	67
Gambar 3.44 Desain Output Laporan Data Transaksi Perbulan	67
Gambar 3.45 Desain Output Laporan Data Transaksi Pertahun Perpemilik Kos .	68
Gambar 3.46 Desain Output Laporan Data Transaksi Pertahun	68
Gambar 3.47 Desain Output Laporan Data Pencairan Perbulan.....	68
Gambar 3.48 Desain Output Laporan Data Pencairan Pertahun.....	69
Gambar 3.49 Desain Output Daftar Rekening Pada Pemilik.....	69
Gambar 3.50 Desain Output List Kos Pada Pemilik.....	69
Gambar 3.51 Desain Output Konfirmasi Pemesan	70
Gambar 3.52 Desain Output Saldo Pada Pemilik	70
Gambar 3.53 Desain Output List Kos Pada Pencari	70
Gambar 3.54 Desain Output Lihat Kamar	71

Gambar 3.55 Desain Output Panduan Penggunaan Aplikasi.....	71
Gambar 3.56 Desain Output Bukti Pemesanan.....	72

DAFTAR TABEL

Tabel 2.1 Simbol-Simbol Use Case Diagram	26
Tabel 2.2 Simbol-Simbol Sequence Diagram	27
Tabel 2.3 Simbol-Simbol Activity Diagram	28
Tabel 2.4 Simbol-Simbol Class Diagram.....	28
Tabel 3.2 Peran Actor	48
Tabel 3.3 Tabel Admin	72
Tabel 3.4 Tabel Kos	73
Tabel 3.5 Tabel Id Konfirmasi	74
Tabel 3.6 Tabel Pemilik Kos.....	74
Tabel 3.7 Tabel Pencari Kos	75
Tabel 3.8 Tabel Rekening	76
Tabel 3.9 Tabel Saldo	76
Tabel 3.10 Tabel Transaksi	77

BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Teknologi komunikasi saat ini mengalami perkembangan yang sangat pesat. Kehidupan manusia telah dibantu dengan adanya komputer, salah satu manfaat komputer dalam kehidupan manusia adalah untuk mengelola, mengakses, dan menyebarkan informasi.

Saat ini kebutuhan akan pendidikan yang layak telah mengalami peningkatan yang signifikan. Kabupaten Tanah Datar khususnya Batusangkar merupakan salah satu kota budaya di Sumatera Barat serta terdapat Institut Agama Islam Negeri Batusangkar. Banyaknya para pendatang dari berbagai daerah untuk menempuh pendidikan yang lebih tinggi di Batusangkar merupakan salah satu contoh bahwa masyarakat saat ini semakin sadar akan pentingnya pendidikan. Seiring dengan banyaknya kebutuhan tempat tinggal sementara atau rumah kos juga semakin meningkat.

Adapun kesulitan mencari tempat tinggal sementara atau rumah kos yaitu minimnya informasi mengenai lokasi, ketersediaan kamar, fasilitas, serta biaya sewa. Disamping itu, timbul suatu masalah yaitu bagaimana melakukan pemesanan kamar kos secara cepat dan mudah. Salah satu solusi untuk mengatasi permasalahan tersebut adalah dengan media komunikasi terkomputerisasi yang dapat memfasilitasi pencari kos untuk mencari informasi mengenai rumah kos secara cepat dan mudah.

Aplikasi ini dapat menjadi sarana penyedia informasi yang efektif bagi pencari kos maupun bagi pemilik kos untuk mempromosikan rumah kos yang akan disewakan dan mengelola segala hal yang berkaitan dengan rumah kos pada sebuah sistem informasi yang berbasis *android* secara cepat dan mudah. Maka dari itu, akan dibangun sebuah sistem informasi berbasis *android* untuk mengelola dan menyediakan informasi yang

berguna untuk membantu menyediakan berbagai macam informasi berkaitan dengan rumah kos serta pemesanan kamar kos secara *online*.

Berdasarkan uraian diatas penulis mencoba membahasnya dalam tugas akhir yang berjudul **“SISTEM INFORMASI RUMAH KOS DISEKITAR KAMPUS IAIN BATUSANGKAR BERBASIS ANDROID”**.

B. Identifikasi Masalah

Berdasarkan latar belakang masalah diatas, maka dapat diidentifikasi beberapa masalah sebagai berikut :

1. Kesulitan dalam mencari tempat tinggal sementara atau rumah kos seperti informasi mengenai lokasi, ketersediaan kamar, fasilitas, dan biaya sewa.
2. Bagaimana melakukan pemesanan kamar kos secara cepat dan mudah.

C. Batasan Masalah

Agar tidak terjadi pembahasan masalah yang menyimpang dari judul, maka penulis memberi batasan masalah sebagai berikut :

1. Aplikasi sistem informasi ini hanya digunakan oleh 3 user yaitu admin, pemilik kos, dan pencari kos.
2. Sistem informasi rumah kos ini hanya sekitaran Kampus IAIN Batusangkar.
3. Aplikasi berfungsi untuk mengelola data rumah kos dan melakukan pemesanan kos secara *online*.

D. Rumusan Masalah

Berdasarkan latar belakang masalah yang penulis jabarkan di atas maka dapat dirumuskan suatu permasalahan. Bagaimana merancang dan membuat sistem informasi rumah kos berbasis *android* ?

E. Tujuan Penelitian

Adapun tujuan yang ingin dicapai dari permasalahan ini adalah untuk :

1. Merancang sebuah sistem informasi yang menyediakan berbagai informasi bagi pencari kos sehingga dapat melakukan pemesanan dan pembayaran kos secara *online*.
2. Merancang sebuah sistem informasi bagi pemilik kos untuk mengelola data-data yang berkaitan dengan rumah kos.

F. Kegunaan Penelitian

Kegunaan penelitian ini adalah :

1. Sebagai sarana menerapkan dan pengembangan ilmu yang telah penulis dapatkan selama masa perkuliahan.
2. Sebagai tambahan referensi bagi pembaca yang akan melakukan penelitian lebih lanjut.
3. Dapat memberikan sumbangan pemikiran untuk masyarakat, sehingga dapat merasakan manfaat dari perkembangan teknologi informasi.
4. Sebagai salah satu syarat untuk memperoleh gelar Ahli Madya (Amd) Program Diploma III Manajemen Informatika pada Institut Agama Islam Negeri (IAIN) Batusangkar.

G. Metodologi Penelitian

Dalam penulisan tugas akhir ini, penulis menggunakan beberapa metode antara lain :

1. Penelitian Lapangan (*Field Research*)

Dalam penelitian ini penulis mendapat data langsung dari hasil peninjauan ke lapangan yaitu pada rumah-rumah kos yang berada di sekitar Kampus IAIN Batusangkar dan mengadakan wawancara pada bagian yang berhubungan langsung terhadap pembahasan yang penulis teliti.

2. Penelitian Perpustakaan (*Library Research*)

Penelitian ini dilakukan dengan cara mengumpulkan bahan-bahan dari buku-buku, karangan ilmiah, ataupun tulisan yang berhubungan dengan permasalahan yang dibahas dalam penelitian ini.

3. Penelitian Laboratorium (*Laboratory Research*)

Penulis melakukan pengolahan data yang telah dikumpulkan dengan menggunakan komputer sebagai alat bantu dalam pembuatan tugas akhir ini.

H. Sistematika Penulisan

Sistematika penulisan laporan tugas akhir ini dibagi dalam IV BAB yang disusun sebagai berikut :

1. BAB I PENDAHULUAN

Bab ini merupakan penguraian mengenai latar belakang masalah, identifikasi masalah, batasan dan rumusan masalah, tujuan dan kegunaan penelitian, metodologi penelitian, sistematika penulisan.

2. BAB II LANDASAN TEORI

Bab ini berisi teori yang diambil dari buku-buku panduan dan referensi lainnya.

3. BAB III ANALISA DAN PERANCANGAN

Bab ini membahas analisa sistem yang sedang berjalan dan rancangan sistem yang diusulkan.

4. BAB IV PENUTUP

Bab ini berisi kesimpulan yang didapat selama pembuatan laporan tugas akhir serta saran-saran yang akan menjadi masukan bagi perkembangan sistem selanjutnya.

BAB II

LANDASAN TEORI

A. Konsep Dasar Sistem Informasi

1. Pengertian Sistem

Menurut (Sutabri, 2012) sistem merupakan bentuk integrasi antara satu komponen dengan komponen lain karena sistem memiliki saran yang berbeda untuk setiap kasus yang terjadi yang ada di dalam sistem tersebut.

Menurut (M.Faisal, 2008) ada beberapa pendapat para ahli tentang pengertian sistem yaitu:

- a. L. Ackof, sistem adalah setiap kesatuan secara konseptual atau fisik yang terdiri dari bagian-bagian dalam keadaan saling tergantung satu sama lainnya.
- b. Ludwig Von Bertalanfy, sistem merupakan seperangkat unsur yang saling terikat dalam suatu relasi di antara unsur-unsur tersebut dalam lingkungan.
- c. Anatol Rapoport, sistem adalah suatu kumpulan kesatuan dan perangkat hubungan satu sama lain.

Dari beberapa pengertian sistem diatas dapat disimpulkan bahwa sistem merupakan suatu kesatuan utuh yang terdiri dari beberapa bagian yang saling berhubungan atau terikat antara yang satu dengan yang lainnya dan berinteraksi untuk mencapai suatu tujuan tertentu.

1) Karakteristik Sistem

Karakteristik sistem adalah sebagai berikut:

- a) Komponen-komponen sistem (*components*), suatu sistem terdiri dari beberapa komponen yang saling berinteraksi membentuk satu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Suatu sistem dapat mempunyai sistem yang lebih besar yang disebut dengan “supra sistem”.

- b) Batas sistem (*boundary*), ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem di pandang satu kesatuan yang tidak dapat dipisahkan.
- c) Lingkungan luar (*environment*), lingkungan luar dari suatu sistem adalah apapun di luar batas sistem yang mempengaruhi operasi sistem. Lingkungan sistem dapat bersifat menguntungkan dan juga merugikan sistem tersebut. Lingkungan luar sistem yang menguntungkan merupakan energi dari sistem, dengan demikian harus tetap dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus ditahan dan dikendalikan, jika tidak maka akan mengganggu kelangsungan hidup dari sistem.
- d) Penghubung (*interface*), merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya.
- e) Masukan sistem (*input*), adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*).
- f) Keluaran sistem (*output*), merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.
- g) Pengolahan sistem (*process*), adalah suatu pengolahan yang akan mengolah *input* menjadi *output*. Sistem produksi akan mengolah *input* bahan baku menjadi *output* berupa bahan jadi.
- h) Sasaran sistem (*objective*), merupakan tujuan atau sasaran yang ingin dicapai oleh sistem. Sistem akan dikatakan berhasil apabila mengenai sasaran atau tujuan.

2) Pengelompokan atau Klasifikasi Sistem

a) Sistem nyata dan sistem abstrak

Sistem nyata bisa juga disebut sistem fisik, yaitu sistem yang terlihat wujudnya dan nyata. Sedangkan sistem abstrak, yaitu sistem yang terdiri dari ide-ide dan pemikiran yang tidak terlihat wujudnya.

b) Sistem deterministik dan probabilistik

Sistem deterministik yaitu suatu sistem yang tingkah lakunya bisa diprediksi. Sedangkan probabilistik yaitu suatu sistem yang nantinya tidak bisa diprediksi karena hanya berupa kemungkinan.

2. Informasi

a. Pengertian Informasi

Menurut (Wahyono, 2004) pengertian informasi menurut para ahli sebagai berikut:

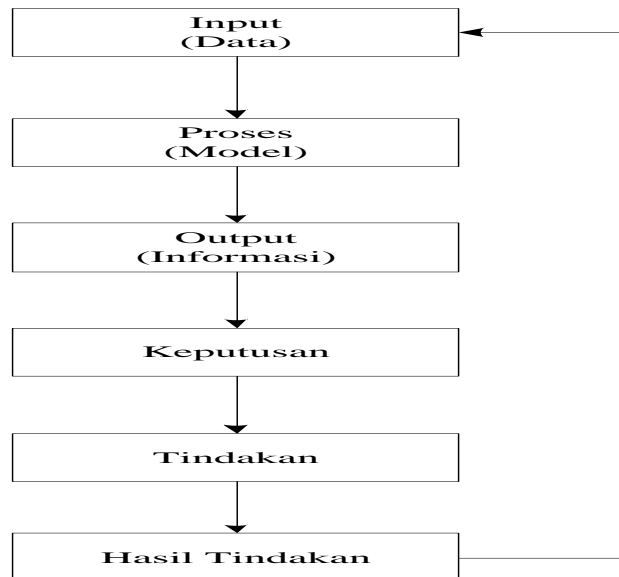
- 1) Menurut Gordon B. Davis, informasi adalah sebagian data yang telah diolah menjadi bentuk yang berguna bagi penerimanya dan nyata berupa nilai yang dapat dipahami di dalam keputusan sekarang maupun masa depan.
- 2) Barry E. Cushing, mengatakan informasi merupakan sesuatu yang menunjukkan hasil pengolahan data yang diorganisasi dan berguna kepada orang yang menerimanya.
- 3) Robert N. Anthony dan Jhon Dearden, menyebutkan informasi sebagai suatu kenyataan, data, item yang menambah pengetahuan bagi penggunanya.
- 4) Stephen A. Moscovice dan Mark G. Simkin, mengatakan informasi sebagai kenyataan atau bentuk-bentuk yang berguna yang dapat digunakan untuk pengambilan keputusan bisnis.

Dari keempat pengertian di atas dapat disimpulkan bahwa informasi merupakan hasil dari pengolahan data menjadi bentuk yang lebih berguna bagi yang menerimanya, menggambarkan suatu

kejadian-kejadian nyata dan dapat digunakan sebagai alat bantu untuk pengambilan suatu keputusan.

b. Siklus Informasi

Pengolahan data menjadi suatu informasi dapat digambarkan sebagai sebuah siklus yang berkesinambungan seperti berikut:



Gambar 2. 1 Siklus Informasi

Secara sederhana dapat dikatakan bahwa data diolah menjadi suatu informasi. Dan pada tahapan selanjutnya sebuah informasi akan menjadi data untuk terciptanya informasi yang lain.

c. Karakteristik Informasi

1) Benar atau salah

Karakteristik tersebut berhubungan dengan sesuatu yang realitas atau tidak dari sebuah informasi.

2) Baru

Sebuah informasi dapat berarti sekali bagi penerimanya.

3) Tambahan

Sebuah informasi dapat memperbaharui atau memberikan nilai tambahan pada informasi yang telah ada.

4) Korektif

Sebuah informasi dapat menjadi bahan koreksi bagi informasi sebelumnya, salah atau palsu.

5) Penegas

Informasi dapat mempertegas informasi yang telah ada, hal ini masih berguna karena dapat meningkatkan persepsi penerima atas kebenaran informasi tersebut.

3. Sistem Informasi

a. Pengertian Sistem Informasi

Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan. Sistem ini memanfaatkan perangkat keras dan perangkat lunak komputer, prosedur manual, model manajemen dan basis data (Sutabri, 2012).

Menurut James B Bower sistem informasi adalah suatu cara tertentu untuk menyediakan informasi yang dibutuhkan oleh organisasi untuk beroperasi dengan cara yang sukses dan untuk organisasi bisnis dengan cara yang menguntungkan.

Pada dasarnya sistem informasi merupakan suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi. Sistem informasi di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Wahyono, 2004).

Dari pendapat para ahli diatas dapat disimpulkan bahwa sistem informasi adalah cara-cara yang diorganisasi untuk

mengolah, memasukkan, menyimpan data dan cara yang diorganisasi untuk menyimpan, mengelola, mengendalikan dan melaporkan informasi sedemikian rupa sehingga sebuah organisasi dapat mencapai tujuan yang telah ditetapkan.

b. Komponen Sistem Informasi

- 1) Model merupakan komponen pembentuk kerangka dasar dari suatu informasi.
- 2) Basis data merupakan komponen pendukung informasi yang biasanya dikelola oleh program penyimpanan.
- 3) Input merupakan komponen input terdiri dari elemen pendukung berhubungan dengan pemasukan data sumber.
- 4) Keluaran merupakan komponen keluaran berupa produk yang siap disajikan dan biasanya sudah teruji serta diketahui betul oleh pimpinan terkait.
- 5) Teknologi merupakan bagian komponen informasi yang sangat mendukung kelancaran dan kecepatan penyajian informasi adalah komponen teknologi.
- 6) Kontrol merupakan komponen informasi sebagai pengendali sistem, melindungi sistem dari kejadian-kejadian yang merugikan sistem.

c. Perangkat Sistem Informasi

Sebuah sistem informasi yang lengkap memiliki kelengkapan sistem sebagai berikut:

1) Hardware

Bagian ini merupakan bagian perangkat keras sistem informasi, seperti komputer, printer, dan teknologi jaringan komputer.

2) Software

Bagian ini merupakan bagian perangkat lunak sistem informasi untuk memerintahkan komputer melaksanakan tugas yang harus dilakukannya.

3) Data

Merupakan komponen dasar dari informasi yang akan diproses lebih lanjut untuk menghasilkan informasi.

4) Prosedur

Merupakan bagian yang berisikan dokumentasi prosedur atau proses-proses yang terjadi dalam sistem.

5) Manusia

Merupakan bagian utama dalam suatu sistem informasi, yang terlihat dalam komponen manusia antara lain adalah *First Level Manager, Staff Specialist, Management*.

d. Pengelola Sistem Informasi

Pengelola sistem informasi terorganisasi dalam suatu struktur manajemen. Oleh karena itu bentuk atau jenis sistem informasi yang diperlukan sesuai dengan level manajemennya.

1) Manajemen level atas untuk perencanaan strategis, kebijakan dan pengambilan keputusan.

2) Manajemen level menengah untuk perencanaan taktis dan pengambilan keputusan.

3) Manajemen level bawah untuk perencanaan dan pengawasan operasi dan pengambilan keputusan.

4) Operator untuk pemrosesan transaksi dan merespon permintaan.

B. Android

1. Sejarah Android

Menurut mAster.com (2012:5) *Android* adalah sistem operasi untuk telepon seluler yang berbasis Linux. *Android* menyediakan *platform* terbuka bagi para pengembang buat menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam piranti bergerak.

Menurut Safaat (2014:1) *Android* adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi,

middleware, dan aplikasi. *Android* menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya, *Google Inc* membeli *Android Inc* yang merupakan pendatang baru yang membuat peranti lunak untuk ponsel atau *smartphone*. Kemudian untuk mengembangkan *android*, dibentuklah *Open Handset Alliance*. Konsorsium dari 34 perusahaan peranti keras, peranti lunak dan telekomunikasi termasuk *Google*, HTC, Intel, Motorola, Qualcomm, T-Mobile dan Nvidia.

Pada saat perilis perdana *Android*, 5 November 2007, *android* bersama *Open Handset Alliance* menyatakan mendukung pengembangan *open source* pada perangkat *Mobile*. Di lain pihak, *Google* merilis kode-kode *android* di bawah lisensi *Apache*, sebuah lisensi perangkat lunak dan *open platform* perangkat seluler.

Di dunia ini terdapat dua jenis distributor sistem operasi *android*. Pertama, yang mendapat dukungan penuh dari *Google* atau *Google Mail Services* (GMS). Kedua, yang benar-benar bebas distribusinya, tanpa dukungan langsung dari *Google* atau dikenal dengan *Open Handset Distribution* (OHD).

Sekitar September 2007 *Google* mengenalkan *Nexus One*, salah satu jenis *smartphone* yang menggunakan *android* sebagai sistem operasinya. Telepon seluler ini diproduksi oleh HTS Corporation dan tersedia di pasaran pada 5 Januari 2010. Pada 9 Desember 2008, diumumkan anggota baru yang bergabung dalam program kerja *android* ARM Holdings, Atheros Communications, diproduksi oleh Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp, dan Vodafone Group Plc. Seiring pembentukan *Open Handset Alliance*, OHA mengumumkan produk perdana mereka, *android* sebagai perangkat *mobile* yang merupakan modifikasi kernel Linux 2.6.

Pada masa sekarang kebanyakan vendor-vendor *smartphone* sudah memproduksi *smartphone* berbasis *android*, vendor-vendor yang dimaksud, yaitu HTC, Motorola, Samsung, LG, HKC, Huawei, Archos,

Webstation Camangi, Dell, Nexus, SciPhone, WayteQ, Sony Ericsson, Acer, Philips, T-Mobile, Nexian, IMO, Asus serta masih banyak lagi vendor *smartphone* di dunia yang memproduksi *android*. Hal ini karena *Android* adalah sistem operasi yang *open source* sehingga bebas didistribusikan dan dipakai oleh vendor manapun.

Tidak hanya menjadi sistem operasi *smartphone*, saat ini *android* menjadi pesaing utama dari *apple* pada sistem operasi Tablet PC. pesatnya pertumbuhan *android* yang menjadi faktor yang telah disebutkan di atas adalah karena *android* merupakan *platform* yang sangat lengkap baik itu sistem operasinya, aplikasi dan Tool pengembangannya, Market aplikasi *android* serta dukungan yang sangat tinggi dari komunitas *open source* di dunia, sehingga *android* terus berkembang pesat baik dari segi teknologi maupun segi jumlah *device* yang ada di dunia.

2. Android Platform Masa Depan

Android disebut sebagai “*platform mobile* pertama yang lengkap, terbuka dan bebas”.

- a. Lengkap (*Complete Platform*), para desainer dapat melakukan pendekatan yang komprehensif ketika mereka sedang mengembangkan *platform android*. *Android* merupakan sistem operasi yang aman dan banyak menyediakan *tools* dalam membangun *software* dan memungkinkan untuk peluang pengembangan aplikasi.
- b. Terbuka (*Open Source Platform*), *platform android* disediakan melalui lisensi *open source*. Pengembang dapat bebas untuk mengembangkan aplikasi. *Android* sendiri mengembangkan Linux Kernel 2.6.
- c. Free (*Free Platform*), *android* adalah *platform* atau aplikasi yang bebas untuk *develop*. Tidak ada lesensi atau biaya royalty untuk dikembangkan pada *platform android*, tidak ada diperlukan biaya keanggotaan, tidak diperlukan biaya pengujian, tidak ada kontrak

yang diperlukan. Aplikasi untuk *android* dapat didistribusikan dan diperdagangkan dalam bentuk apapun.

Android merupakan generasi baru *platform mobile*, *platform* yang memberikan pengembang untuk melakukan pengembangan sesuai dengan yang diharapkan. Sistem operasi yang mendasari *Android* dilisensikan di bawah GNU, General Public License Versi 2 (GPLv2) yang sering dikenal dengan istilah “copyleft” lisensi dimana setiap perbaikan pihak ketiga harus terus jatuh di bawah terms. *Android* didistribusikan di bawah lesensi *Apache Software* (ASL/Apache2) yang memungkinkan untuk distribusi kedua dan seterusnya. Komersialisasi pengembang (produsen *handset* khususnya) dapat memilih untuk meningkatkan *platform* tanpa harus memberikan perbaikan mereka ke masyarakat *open source*. Sebaliknya, pengembang dapat keuntungan dan perangkat tambahan seperti perbaikan dan mendistribusikan ulang pekerjaan mereka di bawah lesensi apapun yang mereka inginkan. Pengembang aplikasi *android* diperbolehkan untuk mendistribusikan aplikasi mereka di bawah skema lisensi apapun yang mereka inginkan.

Pengembang memiliki beberapa pilihan ketika membuat aplikasi yang berbasis *android*. Kebanyakan pengembang menggunakan *eclipse* yang tersedia secara bebas untuk merancang dan mengembangkan aplikasi *android*. *Eclipse* adalah IDE yang paling populer untuk pengembangan *android*, karena memiliki *android plug-in* yang tersedia untuk memfasilitasi pengembangan *Android*. Selain itu *eclipse* juga mendapat dukungan langsung dari *Google* untuk menjadi IDE pengembangan aplikasi *android*, ini terbukti dengan adanya penambahan *plugins* untuk *eclipse* dalam membuat *project android* dimana *source software* langsung dari situs resminya *Google*. Tetapi hal diatas tidak menutup kemungkinan untuk menggunakan IDE yang lain seperti *Netbeans* untuk melakukan pengembangan *android*.

Aplikasi *android* dapat dikembangkan pada sistem operasi berikut:

- 1) Windows XP, Vista/Seven
- 2) Mac OSX (Mac OS X 10.4.8)
- 3) Linux

3. DVM (*The Dalvik Virtual Machine*)

Salah satu elemen kunci dari *android* adalah *Dalvik Virtual Machine* (DVM). *android* berjalan di dalam *Dalvik Virtual Machine* (DVM) bukan di *Java Virtual Machine* (JVM), sebenarnya banyak persamaannya dengan *Java Virtual Machine* (JVM) seperti *Java Mobile Edition* (JME), tetapi *android* menggunakan *virtual machine* sendiri yang dikustomisasi dan dirancang untuk memastikan bahwa beberapa *feature-feature* berjalan lebih efisien pada perangkat *mobile*.

Dalvik Virtual Machine (DVM) adalah “*register bases*” sementara *Java Virtual Machine* (JVM) adalah “*stack based*”, DVM didesain dan ditulis oleh Dan Bornsten dan beberapa *engineer Google* lainnya. Jadi bisa dikatakan “*Dalvik Equals (Java)==False*”. *Dalvik Virtual Machine* menggunakan kernel Linux untuk menangani fungsionalitas tingkat rendah termasuk keamanan, *threading*, dan proses serta manajemen memori. Ini memungkinkan untuk menulis Aplikasi C/C+ sama halnya seperti OS Linux. Meskipun dalam kenyataannya kita harus banyak memahami arsitektur dan proses sistem dari kernel Linux yang digunakan dalam *android*.

Semua *hardware* yang berbasis *android* dijalankan dengan menggunakan *virtual machine* untuk eksekusi aplikasi, pengembang tidak perlu khawatir tentang implementasi perangkat keras tertentu. *Dalvik Virtual Machine* mengeksekusi *executable file*, sebuah format yang dioptimalkan untuk memastikan memori yang digunakan sangat kecil. *The executable file* diciptakan dengan mengubah bahasa *Java* dan dikompilasi menggunakan *tools* yang disediakan dalam SDK *android*.

4. Android SDK (*Software Development Kit*)

Android SDK adalah *tools API (Application Programming Interface)*, yang diperlukan untuk mulai mengembangkan aplikasi pada *platform android* menggunakan bahasa pemrograman *Java*. *Android* merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang direlease oleh *Google*. Saat ini disediakan *Android SDK (Software Development Kit)* sebagai alat bantu dan *API (Application Programming Interface)* untuk mulai mengembangkan aplikasi pada *platform android* menggunakan bahasa pemrograman *Java*. Sebagai *platform* aplikasi-netral, *android* memberi anda kesempatan aplikasi bawaan *handphone/smartphone*. Beberapa fitur-fitur *android* yang paling penting adalah:

- a. *Framework* aplikasi yang mendukung penggantian komponen dan *reusable*.
- b. Mesin *Virtual Dalvik* dioptimalkan untuk perangkat *mobile*.
- c. *Integrated browser* berdasarkan *engine open source Webkit*.
- d. Grafis yang dioptimalkan dan didukung oleh *libraries* grafis 2D, grafis 3D berdasarkan spesifikasi *opengl ES 1,0 (Optional Akselerasi Hardware)*.
- e. *SQLite* untuk penyimpanan data.
- f. *Media Support* yang mendukung audio, video, dan gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF) *GSM Telephony* (tergantung hardware).
- g. *Bluetooth*, *EDGE*, *3G*, dan *Wifi* (tergantung hardware).
- h. Kamera, *GPS*, Kompas, dan *Accelerometer* (tergantung hardware).
- i. Lingkungan *Development* yang lengkap dan kaya termasuk perangkat emulator, *tools* untuk *debugging*, profil dan kinerja memori, dan *plugin* untuk *IDE Eclipse*.

5. ADT (*Android Development Tools*)

Android Development Tools (ADT) adalah *plugin* yang didesain untuk IDE *Eclipse* yang memberikan kita kemudahan dalam mengembangkan aplikasi *android* dengan menggunakan *Eclipse* IDE. Dengan menggunakan *Android Development Tools* (ADT) *Eclipse* akan memudahkan kita dalam membuat aplikasi *project android*, membuat GUI aplikasi, dan menambahkan komponen-komponen yang lainnya, begitu juga kita dapat melakukan *running* aplikasi menggunakan *Android SDK* (*Software Development Kit*) melalui *Eclipse*, dengan *Android Development Tools* kita juga dapat melakukan pembuatan *package android* (.apk) yang digunakan untuk distribusi aplikasi *android* yang dirancang.

Mengembangkan aplikasi *android* dengan menggunakan *Android Development Tools* (ADT) di *Eclipse* sangat dianjurkan dan sangat mudah untuk memulai mengembangkan aplikasi *android*. Berikut adalah versi ADT (*Android Development Tools*) untuk *Eclipse* yang sudah dirilis:

- a. ADT 12.00 (July 2011)
- b. ADT 11.00 (June 2011)
- c. ADT 10.0.1 (March 2011)
- d. ADT 10.0.0 (February 2011)
- e. ADT 9.0.0 (January 2011)
- f. ADT 8.0.1 (December 2010)
- g. ADT 8.0.0 (December 2010)
- h. ADT 0.9.9 (September 2010)
- i. ADT 0.9.8 (September 2010)
- j. ADT 0.9.7 (May 2010)
- k. ADT 0.9.6 (March 2010)
- l. ADT 0.9.5 (December 2009)
- m. ADT 0.9.4 (October 2009)

Semakin tinggi *platform android* yang digunakan, dianjurkan menggunakan ADT (*Android Development Tools*) yang lebih terbaru, karena biasanya muncul *platform* baru diikuti oleh munculnya versi ADT yang terbaru. Untuk melakukan instalasi ADT (*Android Development Tools*) di *Eclipse* dapat dilakukan secara *on-line* maupun *off-line*.

6. Arsitektur Android

Secara garis besar Arsitektur *Android* dapat dijelaskan dan digambarkan sebagai berikut:

a. Application dan Widget

Application dan *Widget* adalah layer dimana kita berhubungan dengan aplikasi saja, dimana biasanya kita *download* aplikasi kemudian kita lakukan instalasi dan jalankan aplikasi tersebut. Di layer terdapat aplikasi inti termasuk klien *email*, program sms, kalender, peta, *browser*, kontak dan lain-lain. Semua aplikasi ditulis menggunakan bahasa pemrograman *Java*.

b. Application Frameworks

Android adalah “*Open Development Platform*” yaitu menawarkan kepada pengembang atau memberi kemampuan kepada pengembang untuk membangun aplikasi yang bagus dan inovatif. Pengembang bebas untuk mengakses perangkat keras, akses informasi *resources*, menjalankan *service background*, mengatur alarm, dan menambahkan status *notifications*, dan sebagainya. Pengembang memiliki akses penuh menuju *API Framework* seperti yang dilakukan oleh aplikasi yang kategori inti.

Arsitektur aplikasi dirancang supaya kita dengan mudah dapat menggunakan kembali komponen yang sudah digunakan (*reuse*). Sehingga bisa kita simpulkan *Application Frameworks* adalah layer dimana para pembuat aplikasi melakukan pengembangan atau pembuatan aplikasi yang akan dijalankan di sistem operasi *android*, karena pada *layer* inilah aplikasi dapat

dirancang dan dibuatm seperti *content-providers* yang berupa sms dan panggilan telepon. Komponen-komponen yang termasuk di dalam *Applications Frameworks* adalah sebagai berikut:

- 1) *Views*
- 2) *Content Provider*
- 3) *Resource Manager*
- 4) *Notification Manager*
- 5) *Activity Manager*

c. *Libraries*

Libraries adalah layer dimana fitur-fitur *android* berada, biasanya para pembuat aplikasi mengakses *libraries* untuk menjalankan aplikasinya. Berjalan dia atas kernel, layer ini meliputi berbagai library C/C++ inti seperti Libc dan SLL, serta:

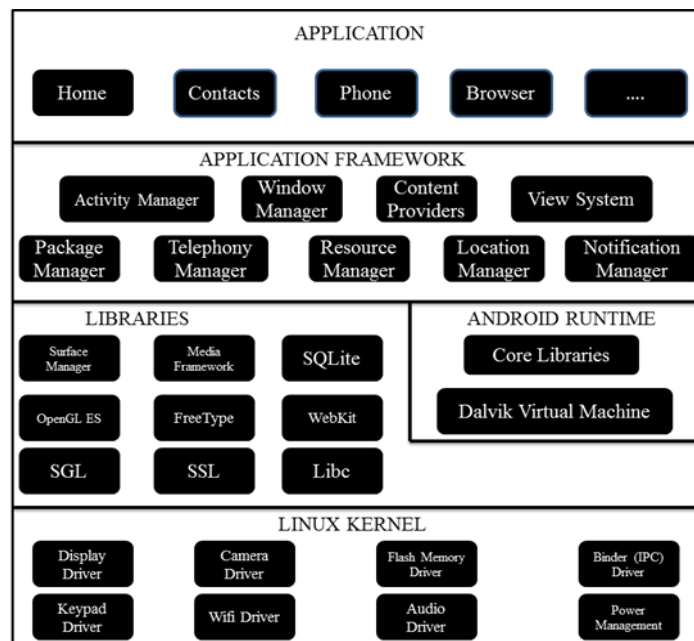
- 1) *Libraries* media untuk pemutaran media audio dan video
- 2) *Libraries* untuk manajemen tampilan
- 3) *Libraries Graphics* mencakup SGL dan OpenGL untuk grafis 2D dan 3D
- 4) *Libraries SQLite* untuk dukungan database
- 5) *Libraries SSL* dan Webkit terintegrasi dengan *web browser* dan *security*
- 6) *Libraries LiveWebcore* mencakup modern *web browser* dengan engine embedded *web view*
- 7) *Libraries 3D* yang mencakup implementasi OpenGL ES 1.0 API's

d. *Android Run Time*

Layer yang membuat aplikasi *android* dapat dijalankan dimana dalam prosesnya menggunakan Implementasi Linux. *Dalvik Virtual Machine* (DVM) merupakan mesin yang membentuk dasar kerangka aplikasi *android*. Di dalam *android run time* dibagi menjadi dua bagian yaitu :

- 1) *Core Libraries*, aplikasi *android* dibangun dalam bahasa *Java*, sementara *Dalvik* sebagai *virtual* mesinnya bukan *Virtual Machine Java*, sehingga diperlukan sebuah *libraries* yang berfungsi untuk menterjemahkan bahasa *Java/C* yang ditangani oleh *Core Libraries*.
 - 2) *Dalvik Virtual Machine*, virtual mesin berbasis register yang dioptimalkan untuk menjalankan fungsi-fungsi secara efisien, dimana merupakan pengembangan yang mampu membuat linux kernel untuk melakukan *threading* dan manajemen tingkat rendah.
- e. Linux Kernel

Linux kernel adalah dimana inti dari operating sistem dari *android* itu berada. Berisi file-file sistem yang mengatur sistem *processing*, *memory*, *resource*, *drivers*, dan sistem-sistem operasi *android* lainnya. *Linux Kernel* yang digunakan *android* adalah *linux kernel release 2.6*. Jika anda ingin mempelajari *linux kernel* ini, anda dapat mengunduh *linux kernel* secara gratis di situs www.kernel.org.



Gambar 2. 2 Detail Anatomi Android

7. Fundamental Aplikasi

Aplikasi *android* ditulis dalam bahasa pemrograman *Java*. Kode *Java* dikompilasi bersama dengan *file resource* yang dibutuhkan oleh aplikasi, dimana prosesnya dipackage oleh *tools* yang dinamakan “*apt tools*” ke dalam paket *android* sehingga menghasilkan file dengan ekstensi apk. File apk itulah yang kita sebut dengan aplikasi, dan nantinya dapat diinstall di perangkat seluler.

Ada enam jenis komponen pada aplikasi *android* yaitu:

a. Activities

Suatu *activity* akan menyajikan *user interface* (UI) kepada pengguna, sehingga pengguna dapat melakukan interaksi. Sebuah aplikasi *android* bisa jadi hanya memiliki satu *activity*, tetapi umumnya aplikasi memiliki banyak *activity* tergantung pada tujuan aplikasi dan desain dari aplikasi tersebut. Satu *activity* biasanya akan dipakai untuk menampilkan aplikasi atau yang bertindak sebagai *user interface* (UI) saat aplikasi diperlihatkan kepada *user*. Untuk pindah dari satu *activity* ke *activity* lain kita dapat melakukannya dengan satu *event*, misalnya *click* tombol, memilih opsi atau menggunakan *triggers* tertentu. Secara hirarki sebuah *windows activity* dinyatakan dengan *method Activity.setView()*. *Content View* adalah objek yang berada pada root hirarki.

b. Services

Service tidak memiliki *Graphic User Interface* (GUI), tetapi *service* berjalan secara *background*, sebagai contoh dalam memainkan musik, *service* mungkin memainkan musik atau mengambil data dari jaringan, tetapi setiap *service* harus berada dalam kelas induknya. Misalnya, media *player* sedang memutar lagu dari *list* yang ada, aplikasi ini akan memiliki dua atau *lebih activity* yang memungkinkan *user* untuk memilih lagu atau menulis sms sambil *player* sedang berjalan. Untuk menjaga musik tetap di

jalankan, *activity player player* dapat menjalankan *service*. *Service* dijalankan pada *thread* utama dari proses aplikasi.

c. Broadcast Receiver

Broadcast receiver berfungsi menerima dan bereaksi untuk menyampaikan notifikasi. Contoh *broadcast* seperti notifikasi zona waktu berubah, baterai *low*, gambar telah selesai diambil oleh kamera atau perubahan referensi bahasa yang digunakan. Aplikasi juga dapat menginisiasi *broadcast* misalnya memberikan informasi pada aplikasi lain bahwa ada data yang telah diubah ke perangkat dan siap untuk digunakan.

Broadcast receiver tidak memiliki *user interface* (UI), tetapi memiliki sebuah *activity* untuk merespon informasi yang mereka terima atau mungkin menggunakan *Notification Manager* untuk memberitahu kepada pengguna, seperti lampu latar atau *vibrating* (getaran) perangkat, dan lain sebagainya.

d. Content Provider

Content provider membuat kumpulan aplikasi data secara spesifik sehingga bisa digunakan oleh aplikasi lain. Data disimpan dalam file sistem seperti database SQLite. *Content provider* menyediakan cara untuk mengakses data yang dibutuhkan oleh suatu *activity*, misalnya ketika kita menggunakan aplikasi yang membutuhkan peta (*Map*) atau aplikasi yang membutuhkan untuk mengakses data kontak dan navigasi, maka disinilah fungsi *content provider*.

8. Versi Android

Telepon pertama yang memakai sistem operasi *android* adalah HTC Dream, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2010 diperkirakan hampir semua vendor seluler di dunia menggunakan *android* sebagai *operating system*. Adapun versi-versi *android* yang pernah dirilis adalah sebagai berikut:

a. Android versi 1.1

Pada 9 Maret 2009 *Google* merilis *Android* versi 1.1 yang dilengkapi dengan pembaruan estetis pada aplikasi, jam, alarm, *voice search* (pencarian suara), pengiriman pesan dengan *Gmail*, dan pemberitahuan *email*.

b. Android versi 1.5 (*Cupcake*)

Pada pertengahan Mei 2009 *Google* kembali merilis telepon seluler dengan menggunakan *Android* dan SDK (*Software Development Kit*) dengan versi 1.5 (*Cupcake*). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, mengupload video ke *Youtube* dan gambar ke *Picasa* langsung dari telepon, dukungan *bluetooth* A2DP, kemampuan terhubung secara otomatis ke *headset Bluetooth*.

c. Android versi 1.6 (*Donut*)

Pada September dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indicator dan control applet VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus, kamera, *camcorder* dan galeri yang diintegrasikan, CDMA/EVDO, 802.1x, VPN, *Gestures* dan *Text-to-speech engine*, kemampuan dial kontak, teknologi *text to change speech* (tidak tersedia pada semua ponsel, pengadaan resolusi VWGA).

d. Android versi 2.0/2.1 (*Éclair*)

Pada 3 Desember 2009 kembali diluncurkan ponsel *Android* dengan versi 2.0/2.1 (*Éclair*), perubahan yang dilakukan adalah pengoptimalan *hardware*, peningkatan *Google Maps* 3.1.2, perubahan UI dengan *browser* baru dan dukungan HTML5, daftar kontak baru, dukungan *flash* untuk kamera 3.2 MP, digital *zoom*, dan *Bluetooth* 2.1.

e. Android versi 2.2 (*Froyo*)

Pada bulan Mei 2010 *Android* versi 2.2 Rev 1 diluncurkan. *android* inilah yang sekarang sangat banyak beredar di pasaran, salah satunya adalah dipakai di Samsung FX tab yang sudah ada di pasaran. Fitur yang tersedia di *android* versi ini sudah kompleks diantaranya adalah:

- 1) Kerangka aplikasi memungkinkan penggunaan dan penghapusan komponen yang tersedia
- 2) *Dalvik Virtual Machine* dioptimalkan untuk perangkat media
- 3) Grafik 2D dan grafis 3D berdasarkan *libraries OpenGL*
- 4) SQLite untuk penyimpanan data
- 5) Mendukung media audio, video, dan berfungsi format gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
- 6) GSM, *Bluetooth*, EDGE, 3G, dan Wifi (*hardware independent*)
- 7) Kamera, *Global Positioning System* (GPS), kompas dan *accelerometer* (tergantung *hardware*)

f. Android versi 2.3 (*Gingerbread*)

Android versi 2.3 diluncurkan pada Desember 2010, hal-hal yang direvisi dari versi sebelumnya adalah kemampuan seperti berikut:

- 1) *SIP-based VoIP*
- 2) *Near Field Communications* (NFC)
- 3) *Gyroscope* dan *sensor*
- 4) *Multiple cameras support*
- 5) *Mixable audio effects*
- 6) *Download Manager*

g. Android versi 3.0 (*Honeycomb*)

Android ini dirilis pada Februari 2011 sebagai *Android* 3.0 revisi 1 serta *Android* versi 3.0 revisi 2 yang telah dirilis pada Juli 2011.

h. Android versi 3.1

Android versi 3.1 dirilis pada Mei 2011, sedangkan *Android* 3.1 revisi 2 juga dirilis bulan Mei 2011 serta revisi 3 dirilis pada Juli 2011

i. Android versi 3.2 yang dirilis pada Juli 2011

j. Android versi 4.0 dirilis pada November 2011

Android versi 3.0 ke atas adalah generasi *platform* yang digunakan untuk tablet pc. Sementara versi 4.0 sudah merupakan *platform* yang bisa dipakai di *smartphone* dan tablet pc (Kurniawan, 2016).




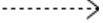

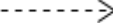


C. Unified Modelling Language (UML)

Menurut Nugroho:2010 UML (*Unified Modelling Language*) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Serta merupakan standar *modeling language* yang terdiri dari kumpulan-kumpulan diagram, dikembangkan untuk membantu para pengembang sistem dan *software* agar bisa menyelesaikan tugas-tugas seperti spesifikasi, visualisasi, desain arsitektur, konstruksi, simulasi, testing serta dokumentasi.

1. Use Case Diagram

Use Case pada dasarnya merupakan unit fungsionalitas koheren yang diekspresikan sebagai transaksi-transaksi yang terjadi antara *actor* dan *sistem*. Kegunaan *use case* sesungguhnya adalah untuk mendefinisikan suatu bagian perilaku sistem yang bersifat koheren tanpa perlu menyikapi struktur internal sistem atau perangkat lunak yang sedang dikembangkan, berikut adalah beberapa simbol di dalam *use case diagram*:

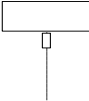


Tabel 2. 1 Simbol-Simbol Use Case Diagram

No	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.
2		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
3		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
4		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (independent).
5		<i>Generalization</i>	Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor).
6		<i>Include</i>	Menspesifikasikan bahwa use case sumber secara eksplisit.
7		<i>Extend</i>	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan.
8		<i>Assosiation</i>	Menghubungkan antara objek satu dengan objek lainnya.

2. Sequence dan Collaboration Diagram

Sequence diagram menggambarkan interaksi antara objek di dalam dan disekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu, menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah *event* untuk menghasilkan *output* tertentu. *Collaboration diagram* juga menggambarkan interaksi antara objek seperti *sequence diagram*, akan tetapi lebih menekankan pada masing masing objek dan bukan pada waktu penyampaian *message*. Setiap *message* memiliki *sequence number*, dimana *message* dari level tertinggi memiliki nomor 1 (Komala, 2015). Berikut adalah beberapa simbol didalam *sequence diagram*:

Tabel 2. 2 Simbol-Simbol Sequence Diagram






No.	Simbol	Nama	Keterangan
1		<i>Object</i> dan <i>lifeline</i>	Orang, tempat, benda, kejadian atau konsep yang ada dalam dunia nyata yang penting bagi suatu aplikasi yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi.
3		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.

3. Activity Diagram

Activity diagram merupakan bentuk khusus dari *state machine* yang bertujuan memodelkan komputasi-komputasi dan aliran-aliran kerja yang terjadi dalam sistem atau perangkat lunak yang sedang dikembangkan. *Activity diagram* menggambarkan berbagai alir aktifitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana

mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Berikut adalah beberapa simbol didalam *activity diagram* (Prawita, 2016):

Tabel 2. 3 Simbol-Simbol Activity Diagram

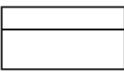


No	Simbol	Nama	Keterangan
1		Activity	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		Decision	Pilihan untuk pengambilan keputusan.
3		Initial Node	Titik awal
4		Activity Final Node	Titik akhir
5		Fork	Menunjukkan kegiatan yang dilakukan secara paralel.

4. Class Diagram

Diagram kelas atau *class diagram* menunjukkan interaksi antar kelas dalam sistem. Diagram kelas mengandung informasi dan tingkah laku segala sesuatu yang berkaitan dengan informasi tersebut. Adapun kegunaan dari *class diagram* adalah sebagai berikut :

- a. Mengelompokkan objek-objek menjadi kelas-kelas berarti mengapresiasi masalah yang sedang dihadapi.
- b. Definisi-definisi *common* (seperti nama kelas dan atribut) cukup disimpan sekali per instan kelas (objek).

Tabel 2. 4 Simbol-Simbol Class Diagram

No	Simbol	Nama	Keterangan
1		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
2		Nary Association	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
3		Asosiasi	Hubungan statis antar <i>class</i> yang menggambarkan class yang memiliki atribut berupa

			<i>class</i> lain atau <i>class</i> yang harus mengetahui eksistensi <i>class</i> lain
4	←	Generalization	Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor)
5	----->	Dependency	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (independent).

D. Internet, Web dan Jaringan

1. Internet

a. Pengertian Internet

Internet adalah jaringan komputer yang terhubung secara internasional dan tersebar di seluruh dunia. Jaringan ini meliputi jutaan pesawat komputer yang terhubung satu dengan yang lainnya dengan memanfaatkan jaringan telepon baik kabel maupun gelombang elektromagnetik (Rami, 2017).

b. Sejarah Internet

Internet dimulai ketika Departemen Pertahanan Amerika Serikat (*Department of Defense USA*) membangun sebuah jaringan komputer di tahun 1969, yang diberi nama ARPANET (*Advanced Research Project Agency NETWORK*) dengan tujuan untuk menghubungkan beberapa komputer yang berada di beberapa universitas melakukan riset militer terutama untuk membangun jaringan komunikasi komputer yang mampu bertahan terhadap serangan nuklir.

Jaringan ini berkembang terus semakin banyak komputer yang terlibat dan riset disisi pengembangan perangkat lunak juga berkembang. Pada bulan Mei tahun 1974, Vinton G.Cerf dari

Stanford University dan Robert E.Kahn dari Departemen Pertahanan USA, mempublikasi sebuah paper yang berjudul “*A Protocol for Packet Network Intercommunication*”, konsep ini kemudian populer sebagai protocol TCP/IP. Ketika ARPANET meng-adopsi protokol menjadi protokol standard untuk ARPANET pada tahun 1983. Pihak universitas terutama *University of California at Berkeley* kemudian membangun sistem operasi *Berkeley Software Distribution Unix*) atau BSD UNIX (dikenal dengan nama Free BSD Unix) dan pihak departemen pertahanan membiayai *Bolt Baranek dan Newman* (BBN) untuk mengimplementasi protokol TCP/IP pada BSD Unix untuk diterapkan pada ARPANET dengan demikian cikal-bakal internet terbentuk.

Walaupun pada tahun 1990 secara resmi ARPANET ditutup, namun jaringan internet yang telah terbentuk diteruskan oleh pihak universitas di Amerika dan memasukkan jaringan universitas di benua Amerika (Kanada dan Amerika Selatan) serta jaringan di Eropa menjadi bagian dari internet.

c. Manfaat Internet

Dengan banyaknya manfaat melalui internet berbagai sektor kehidupan manusia senantiasa memanfaatkan internet dengan cara dan kepentingan yang berbeda. Beberapa sektor yang saat ini banyak menggunakan internet, yaitu dunia Pendidikan, dunia usaha, pemasaran barang dan jasa, pemerintahan, dan masyarakat.

d. Dampak Negatif Internet

Ancaman virus, Carding atau pencurian nomor kartu kredit, Adanya aktivitas *cracking*, Pembajakan karya intelektual, Penyebaran situs-situs yang tidak sesuai dengan moral (pornografi), *Violence and Gore* (Kekejaman dan kesadisan), Penipuan, Perjudian, Kecanduan

Untuk menanggulangi dampak negatif penggunaan internet, maka perlu adanya pengawasan khusus dan aturan yang jelas tentang pemakaian internet terutama dikalangan pendidikan.

e. Layanan Internet

Berikut ini beberapa layanan yang umum digunakan oleh pengguna Internet:

1. WWW (*World Wide Web*)

Layanan WWW merupakan layanan internet yang paling dikenal. dengan layanan ini Anda bisa menikmati berbagai situs web di dunia. Layanan ini disediakan oleh protokol yang dinamakan HTTP (*HyperText Transfer Protocol*).

2. Email (Elektronik Mail)

Layanan untuk mengirimkan surat elektronik melalui internet.

3. IRC (Internet Relay Chat) / YM (Yahoo Messenger)

Berguna untuk melakukan chatting lewat internet.

4. FTP (File Transfer Protocol)

Melalui FTP, Kita bisa mengirimkan file-file yang Kita miliki ke komputer yang sangat jauh lewat Internet. Kita bisa juga mengambil file-file dari komputer yang jauh tersebut.

2. Web (World Wide Web)

a. Pengertian Web

World Wide Web (WWW) adalah nama yang diberikan untuk semua bagian internet yang dapat di akses dengan software web browser. WWW disingkat dengan web, yang terdiri dari jutaan situs web (website) dan setiap website terdiri dari banyak halaman web (*web-page*). Website dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi, gambar gerak, suara dan atau gabungan dari keduanya baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait yang masing-masing dihubungkan dengan link-link.

b. Sejarah Web

Tahun 1993, Tim Berners-Lee dan peneliti lain di European Particle Physics Lab (Conseil Européen pour la Recherche Nucléaire, atau CERN) di Geneva, Swis, mengembangkan suatu cara untuk men-share data antar koleganya menggunakan sesuatu yang disebut dengan hypertext. Pemakai di CERN dapat menampilkan dokumen pada layar komputer dengan menggunakan software browser baru. Kode-kode khusus disisipkan ke dalam dokumen elektronik ini, memungkinkan pemakai untuk meloncat dari satu dokumen ke dokumen lainnya pada layar dengan hanya memilih sebuah hyperlink.

c. Bagaimana WWW Bekerja

1. Informasi web disimpan dalam dokumen yang disebut dengan halaman-halaman web (web-pages).
2. Web page adalah file-file yang disimpan dalam komputer yang disebut dengan server-server web (web servers).
3. Komputer-komputer membaca web page disebut web client
4. Web client menampilkan page dengan menggunakan program yang disebut dengan browser web (web browser).
5. Browser web yang populer adalah internet explorer, Netscape Navigator dan Mozilla Firefox.

d. Server Web

Server Web adalah komputer yang digunakan untuk menyimpan dokumen-dokumen web, komputer ini akan melayani permintaan dokumen web dari kliennya. Browser web berkomunikasi melalui jaringan dengan server web menggunakan HTTP. Browser akan mengirimkan request kepada server untuk meminta dokumen tertentu yang disediakan oleh server

e. URL, Surfing dan Surfer

URL (*Universal Resource Locator*) adalah konsep nama file standard yang diperluas dengan jaringannya. Nama file ini tidak

hanya menunjukkan direktori dan nama filenya, tetapi juga nama mesinnya dalam jaringan.

Surfing adalah kegiatan menelusuri dan mencari informasi dalam internet melalui layanan informasi web. Surfer adalah Pemakai komputer yang melakukan surfing.

3. Jaringan

Jaringan internet pada saat ini merupakan jaringan yang sangat kompleks dan memiliki struktur yang dinamis (selalu berkembang). Internet tersusun atas banyak jaringan WAN dan LAN. Pemakai dapat berhubungan dengan jaringan internet melalui sebuah *ISP (Internet Service Provider)*. Sedangkan jaringan komputer dapat diartikan sebagai sebuah rangkaian (kumpulan) dua atau lebih komputer.

Tujuan utama dari sebuah jaringan komputer adalah sharing resource (baca: sumber daya), dimana sebuah komputer dapat memanfaatkan sumber daya yang dimiliki komputer lain yang berada dalam jaringan yang sama.

a. WAN (*World Area Network*)

WAN adalah jaringan komunikasi data yang luas jangkauannya global atau mendunia. Misal, jaringan komunikasi antara satu negara dengan negara lainnya.

b. LAN (*Local Area Network*)

LAN adalah jaringan komunikasi data yang luas jangkauannya meliputi suatu areal lokal tertentu. Misal, jaringan komunikasi data disuatu gedung.

c. Protokol (*Protocol*)

Protokol merupakan suatu himpunan aturan yang mengatur komunikasi data. Dimisalkan sebagai dua orang yang berasal dari bangsa yang berbeda akan berdialog dan berkomunikasi, kemudian keduanya hanya dapat mengerti dan berbicara dengan bahasa kebangsaannya masing-masing sehingga dapat dipastikan bahwa tujuan dialog dan komunikasi tersebut tidak akan tercapai. Oleh

karena itu agar dialog dan komunikasi dapat berjalan dengan lancar masing-masing orang tersebut harus berdialog dengan memakai jasa penterjemah atau protocol.

d. TCP (*Transmission Control Protokol*)

Transmission Control Protocol berfungsi untuk melakukan transmisi data per-segmen, artinya packet data dipecah dalam jumlah yang sesuai dengan besaran packet kemudian dikirim satu persatu hingga selesai.

e. IP (*Internet Protocol*)

Internet Protocol (IP) *address* atau alamat IP yang bahasa awamnya bisa disebut dengan kode pengenalan komputer pada jaringan.

f. ISP (*Internet Service Provider*)

ISP adalah penyedia layanan internet. Beberapa provider di Indonesia antara lain: Indosat,IM2, XL Corporate, StarOne, Flexi Home, Wifone, Fren, Kabelvision, Telkomnet Instan.

E. Perangkat Lunak Pembangun Sistem

1. Database

Database adalah sekumpulan data yang terdiri dari suatu atau lebih tabel yang saling berhubungan. User mempunyai wewenang untuk mengakses data tersebut, baik untuk menambah, mengubah atau menghapus data yang ada dalam table tersebut. Database digunakan untuk menampung beberapa tabel atau query yang dijadikan media untuk menyimpan data sebagai sumber pengolahan data (Rami, 2017).

2. PHP

a. Pengertian PHP

PHP (Perl Hypertext Preprocessor) adalah bahasa *server-side scripting* yang menyatu dengan *HTML* untuk membuat halaman web yang dinamis. Karena *PHP* merupakan *server-side scripting*

maka sintaks dan perintah-perintah *PHP* akan dieksekusi di *server* kemudian hasilnya dikirimkan ke browser dalam format *HTML*. Dengan demikian kode program yang ditulis dalam *PHP* tidak akan terlihat oleh user sehingga keamanan halaman *web* yang dinamis, yaitu halaman *web* yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke halaman *web* (Komala, 2015).

PHP dapat dibangun sebagai modul pada *web server Apache* dan sebagai *binary* yang dapat berjalan sebagai *CGI* (*Common Gateway Interface*). *PHP* termasuk dalam *Open Source Product*, sehingga source code *PHP* dapat diubah dan didistribusikan secara bebas. *PHP* juga mampu lintas *Platform*. Artinya *PHP* dapat berjalan di banyak sistem operasi yang beredar saat ini, diantaranya Sistem Operasi *Microsoft Windows* (semua versi), *Linux*, *Mac OS*, *Solaris*.

PHP diciptakan pertama kali oleh Rasmus Lerdorf pada tahun 1994. Awalnya, *PHP* digunakan untuk mencatat jumlah serta untuk mengetahui siapa saja pengunjung pada *homepage*-nya. Rasmus Lerdorf adalah salah satu pendukung *open source*.

b. Kelebihan-kelebihan *PHP*

PHP memiliki banyak kelebihan yang tidak dimiliki oleh bahasa script sejenis. Kelebihan – kelebihan diantaranya adalah :

- 1) *PHP* difokuskan pada pembuatan *script server-side*, yang bisa melakukan apa saja yang dapat dilakukan oleh *CGI*, seperti mengumpulkan data dari form, menghasilkan isi halaman *web* dinamis, dan kemampuan mengirim serta menerima cookies, bahkan lebih daripada kemampuan *CGI*.
- 2) *PHP* dapat digunakan pada semua sistem operasi antara lain *linux*, *Unix* (termasuk variannya *HP-UX*, *Solaris* dan *OpenBSD*), *microsoft windows*, *Mac OS X*, *RISC OS*.

- 3) *PHP* mendukung banyak *WEB Server* seperti *Apache*, *Microsoft Internet Information Server (MIIS)*, *Personal Web Server (PWS)*, dan masih banyak lagi lainnya, bahkan *PHP* dapat bekerja sebagai suatu *CGI processor*.
- 4) *PHP* tidak terbatas pada hasil keluaran *HTML (Hypertext Markup Languages)*. *PHP* juga memiliki kemampuan untuk mengolah keluaran gambar, *File PDF*, dan *movies Flash*. *PHP* juga dapat menghasilkan teks seperti *XHTML* dan *file XML* lainnya.

c. Sintak atau Script *PHP*

Script *PHP* termasuk dalam *HTML-embedded*, artinya kode *PHP* dapat disisipkan pada sebuah halaman *HTML*.

Ada empat macam pasangan tag *PHP* yang dapat digunakan untuk menandai blok *script PHP* (Komala, 2015)

- 1) `<?php...?>`
- 2) `<script language = "PHP"> ... </script>`
- 3) `<? ... ?>`
- 4) `<% .. %>`

Cara pertama merupakan format yang dianjurkan tetapi mungkin cara kedua akan sering digunakan karena lebih ringkas. Cara yang ketiga digunakan untuk mengantisipasi editor-editor yang tidak dapat menerima kedua cara diatas. Cara keempat juga dimungkinkan sebagai kemudahan bagi anda yang sudah terbiasa dengan *ASP (Active Server Pages)*. Namun, bila itu tidak dikenal, maka harus dilakukan pengaktifan pada file konfigurasi *php* ini.

d. Konsep Kerja *PHP*

Model kerja *HTML*, diawali dengan permintaan suatu halaman *web* oleh *browser*. Berdasarkan *URL* atau dikenal dengan alamat internet, *browser* mendapatkan alamat dari *web server*, mengidentifikasi halaman yang dikehendaki, dan menyampaikan segala informasi yang dibutuhkan oleh *web server*.

Selanjutnya, *web server* akan mencari berkas yang diminta dan memberikan isinya ke *browser*. *Browser* yang mendapatkan isinya segera melakukan penerjemahan kode *HTML* dan menampilkan isinya ke layar pemakai.

3. Bootstrap

Bootstrap adalah sebuah framework yang dapat menyelesaikan permasalahan dalam mendesain web. Slogan dari framework ini adalah “*Sleek, intuitive, and powerful front-end framework for faster and easier web development*”, yang berarti kita dapat mendesain sebuah website dengan lebih rapi, cepat dan mudah. Selain itu Bootstrap juga responsive terhadap banyak platform, artinya tampilan halaman website yang menggunakan Bootstrap ini akan tampak tetap rapi, baik versi mobile maupun desktop.

Saat ini penggunaan Bootstrap sudah meluas di kalangan disainer front-end web, perkembangannya pun masih terus berlangsung hingga sekarang. Penggunaannya pun tidak begitu rumit, mudah, karna kita tinggal memanggil CS5 dan JS yang tersedia lalu menuliskan class-class nya di kodingan. Bootstrap memiliki 12-column responsive grid, macam-macam components, JavaScript plugins, typography, form controls, dan juga sebuah webbased Customizer untuk membuat Bootstrap.

4. JQuery

Jquery adalah java script library, JQuery mempunyai semboyan “*write less, do more*”. JQuery dirancang untuk memperingkas kode-kode javascript, menangani event, membuat animasi dan interaksi ajax. JQuery dirancang untuk mengubah cara menulis javascript.

Library jQuery mempunyai kemampuan :

1. Kemudahan mengakses elemen-elemen HTML
2. Memanipulasi elemen HTML
3. Memanipulasi CSS
4. Penanganan event HTML

5. Efek-efek javascript dan animasi
6. Modifikasi HTML DOM
7. Menyederhanakan kode javascript lainnya

Untuk memulai jQuery, download jquery.js dari situs <http://www.jquery.com>. Setiap menulis kode javascript dengan menggunakan jquery, jangan lupa untuk memasukan file jquery.js kedalam kode javascript

```
<script type="text/javascript" src="jquery.js"></script>
```

contoh sederhana pemograman dengan menggunakan jquery

```
<html>
```

```
<head>
```

```
<script type="text/javascript" src="jquery.js"></script>
```

```
<script type="text/javascript">
```

```
$(document).ready(function(){
```

```
$(".tombol1").click(function(){
```

```
$("#p").hide(1000);
```

```
});
```

```
$(".tombol2").click(function(){
```

```
$("#p").show(1000);
```

```
});
```

```
});</script>
```

```
</head>
```

```
<body>
```

```
<p>Hello World!</p>
```

```
<button class="tombol1">Sembunyikan</button>
```

```
<button class="tombol2">Tampilkan</button>
```

```
</body>
```

```
</html>
```

Sintaks JQuery

Sintaks jquery biasanya dibuat untuk memilih elemen-elemen HTML dan melakukan aksi terhadap elemen yang dipilih.

Sintaks :

`$(selector).action()`

- 1) Tanda dollar, untuk mendefinisikan jQuery
- 2) (selector), untuk menunjukkan elemen yang dipilih atau dituju
- 3) action(), adalah jQuery action yang akan dilakukan terhadap elemen yang dipilih.

Contoh :

- 1) `$(this).hide()` – menyembunyikan elemen saat ini
- 2) `$("p").hide()` – menyembunyikan semua paragraf atau konten dari tag `<p>`
- 3) `$(".test").hide()` – menyembunyikan elemen yang mempunyai `class="test"`
- 4) `$("#test").show()` – menampilkan elemen yang mempunyai `id="test"`

Karena hampir segala sesuatu yang kita lakukan bila menggunakan jQuery membaca atau memanipulasi document object model (DOM), kita perlu memastikan bahwa kita mulaimenambahkan event segera setelah DOM siap.

Untuk melakukan hal ini, kita menambahkan kode ready event untuk dokumen.

```
$(document).ready(function(){
//kode anda di sini
});
```

Kode di atas berarti kita ingin kode dijalankan apabila halaman HTML telah di-load semuanya. Atau dengan kode javascript biasanya seperti ini :

```
window.onload = function(){ //kode anda di sini
}
```

5. Adobe Dreamweaver CS5

Dalam Buku Madcoms (2012) *Dreamweaver* adalah sebuah *HTML* editor profesional untuk mendesain *web* secara visual dan

mengelola situs atau halaman web. *Dreamweaver* merupakan software utama yang digunakan oleh *web* desainer maupun *web* programmer dalam mengembangkan suatu situs *web*, *Dreamweaver* mempunyai ruang kerja, fasilitas dan kemampuan yang mampu meningkatkan produktivitas dan efektivitas dalam desain maupun membangun suatu situs web. Saat ini terdapat *software* dari kelompok *adobe* yang belakangan banyak digunakan untuk mendesain suatu situs *web*. Versi terbaru dari *Dreamweaver* saat ini adalah *Dreamweaver CS5*.

Dreamweaver merupakan *software* utama yang digunakan oleh *web* Desainer maupun *web* Programmer dalam mengembangkan suatu situs *web*. Hal ini disebabkan ruang kerja, fasilitas dan kemampuan *dreamweaver* yang mampu meningkatkan produktivitas dan efektivitas dalam desain maupun membangun suatu situs *web*.

Langkah menjalankan *Dreamweaver CS5* adalah pilih *start* → *All programs* → *Adobe Master Collection CS5* → *Adobe Dreamweaver CS5*



Gambar 2. 3 Tampilan Halaman Welcome Dreamweaver CS5

Dalam tampilan awal *Dreamweaver* terdapat pilihan *open a Recent Item* (*File* yang pernah terbuka), *create New* (membuat *file* baru), *Top Features* (fitur-fitur baru), Dan *Getting started* (Tuntunan Penggunaan *Dreamweaver*). Halaman *welcome screen* akan selalu ditampilkan saat anda menjalankan program *Dreamweaver*, jika anda

tidak menginginkan halaman tersebut tampil maka beri tanda centang pada pilihan *Dont show* again.

Selanjutnya Gambar berikut merupakan gambaran *layout* kerja *Dreamweaver CS5*.



Gambar 2. 4 Tampilan Lembar Kerja Dreamweaver

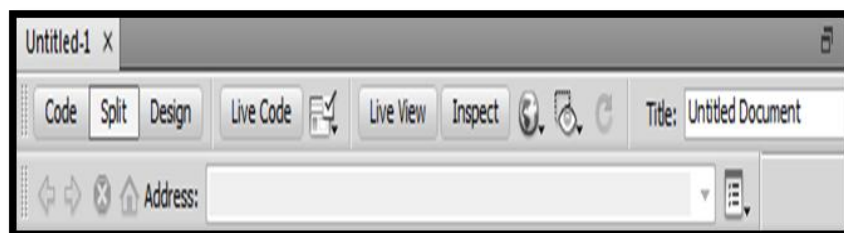
Application Bar, berada di bagian paling atas jendela aplikasi *dreamweaver CS5*. Baris ini berisi tombol *workspace* (*workspace switcher*), menu dan aplikasi lainnya.



Gambar 2. 5 Tampilan Application Bar

1) *Toolbar Document*

Berisi tombol-tombol yang digunakan untuk menampilkan jendela dokumen, seperti kita bisa menampilkan code saja, desain saja atau kedua-duanya.



Gambar 2. 6 Document Toolbar

2) *Panel Group*

Kumpulan panel yang saling berkaitan, panel-panel ini dikelompokkan pada judul-judul tertentu berdasarkan fungsinya. panel ini digunakan untuk memonitor dan memodifikasi pekerjaan. Panel group ini berisi panel insert, *CSS, Styles, Asset, AP Elemen* dan *Files*.

3) *Panel Propertis*

Menampilkan dan mengubah berbagai properti yang dipunyai elemen tertentu. Kita bisa langsung mengubah properti dari elemen tersebut dengan tool ini, misalnya merubah warna text, memberikan *background* pada elemen tabel, menggabungkan kolom, dan lain-lain.

4) *Panel Insert*

Digunakan untuk menyisipkan berbagai jenis objek, seperti *image*, tabel, atau objek media kedalam jendela dokumen.

5) *Panel File*

Digunakan untuk mengatur *file-file* dan folder-folder yang membentuk situs web.

6. MySQL

MySQL dikembangkan oleh sebuah perusahaan Swedia bernama *MYSQL AB* yang pada saat itu bernama *TcX DataKonsult AB* sekitar tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak 1979. Awalnya *TcX* membuat *MySQL* dengan tujuan mengembangkan aplikasi *web* untuk klien. Kepopuleran *MySQL* antara lain karena *MySQL* menggunakan *SQL* sebagai bahasa dasar untuk mengakses databasenya sehingga mudah untuk digunakan, kinerja *query* cepat, dan mencukupi untuk kebutuhan *database* perusahaan-perusahaan skala menengah kecil.

Keandalan suatu *system database (DBMS)* dapat diketahui dari cara kerja optimizer-nya dalam melakukan proses perintah-perintah *SQL*, yang dibuat oleh user maupun program-program

aplikasinya. Sebagai *databaseserver*, *MySQL* dapat dikatakan lebih unggul dibandingkan dengan *database server* yang lainnya dalam *query* data.

Adapun beberapa bentuk keistewaan dari *MySQL* adalah sebagai berikut :

1) *Portability*

MySQL dapat berjalan stabil pada berbagai sitem operasi di antaranya adalah seperti *Windows*, *Linux*, *FreeBSD*, *Mac OS X server*, *Solaris*, *Amiga*, *HP-UX* dan masih banyak lagi.

2) *Open Source*

MySQL didistribusikan secara *open source* (gratis), di bawah lisensi *GPL*.

3) *Multiuser*

MySQL dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami masalah atau konflik. Hal ini memungkinkan sebuah database server *MySQL* dapat diakses client secara bersamaan.

4) *Performance Tuning*

MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak *SQL* per satuan waktu.

5) *Column Types*

MySQL memiliki tipe kolom yang sangat kompleks, seperti *signed/unsigned integer*, *float*, *double*, *char*, *varchar*, *text*, *blob*, *date*, *time*, *datetime*, *year*, *set* serta *enum*.

6) *Command dan Function*

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah *SELECT* dan *WHERE* dalam *query*.

7) *Security*

MySQL memiliki beberapa lapisan sekuritas seperti level *subnetmask*, *nama host*, dan *user* dengan system perizinan yang mendetail serta *password terencripsi*.

8) *Stability dan Limits*

MySQL mampu menangani database dalam skala besar, dengan jumlah records lebih dari 50 juta dan 60 ribu table serta 5 miliar baris. Selain itu, batas indeks yang dapat di tampung mencapai 32 indeks pada tiap tabelnya.

9) *Connectivity*

MySQL dapat melakukan koneksi dengan client menggunakan *protocol TCP/IP*, *Unix socket (Unix)*, atau *Named Pipes (NT)*.

10) *Localisation*

MySQL dapat mendeteksi pesan kesalahan (*error code*) pada client dengan menggunakan lebih dari dua puluh bahasa. Meski demikian, bahasa Indonesia belum termasuk di dalamnya.

11) *Interface*

MySQL memiliki interface (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi *API (Application Programming Interface)*.

12) *Client dan Tools*

MySQL dilengkapi dengan berbagai *tool* yang dapat digunakan untuk administrasi *database*, dan pada setiap *tool* yang ada disertai petunjuk *online*.

13) *Struktur Tabel*

MySQL memiliki struktur table yang lebih fleksibel dalam menangani *ALTER TABLE*, dibandingkan database lainnya semacam *PostgreSQL* ataupun *Oracle*.

BAB III

ANALISA DAN HASIL

A. Analisa Masalah

Analisa masalah berguna untuk mengetahui berbagai masalah yang ada sehingga dengan adanya Sistem Informasi Rumah Kos Berbasis Android ini diharapkan dapat menyelesaikan permasalahan yang ada. Adapun beberapa masalah tersebut adalah sebagai berikut:

1. Belum tersediannya sistem informasi rumah kos yang berbasis *android*.
2. Belum tersedianya sistem informasi bagi pemilik kos untuk mengelola data-data yang berkaitan dengan rumah kos.
3. Belum tersedianya sistem informasi bagi pencari kos mengenai rumah kos seperti lokasi kos, ketersediaan kamar, fasilitas, biaya sewa, serta pembayaran dan pemesanan kamar secara online.

B. Analisa Aplikasi

Dalam pembuatan aplikasi perangkat lunak dibutuhkan adanya suatu analisa dan perancangan dari sistem yang akan dibuat. Ini bertujuan agar aplikasi yang akan dibuat dapat selesai tepat waktu dan sesuai dengan tujuan awal. Kesalahan dalam analisa akan berimbas pada tahapan proyek selanjutnya dan mengakibatkan aplikasi tidak maksimal atau tidak sesuai dengan tujuan pembuatan aplikasi.

Tahapan analisa berisi identifikasi dan evaluasi permasalahan yang ada serta kebutuhan apa saja yang dibutuhkan dalam pembuatan aplikasi perangkat lunak. Dengan adanya analisa dan perancangan maka akan didapatkan suatu gambaran mengenai kebutuhan apa saja yang dibutuhkan dalam program aplikasi ini. Yang menjadi *user* adalah admin, pemilik kos, dan pencari kos. Langkah-langkah dalam penggunaan aplikasi dari awal hingga akhir adalah sebagai berikut:

1. Di halaman utama untuk pencari kos terdapat sebuah pilihan menu yaitu lihat dashboard, kos, help, pembayaran, dan cek status.

2. Jika pencari kos ingin mengetahui daftar kos yang ada maka pencari dapat mengetahui dari menu kos.
3. Setelah pencari kos melihat kos yang ada, maka pencari dapat melakukan pembookingan kamar apabila telah sesuai dengan keinginan.
4. Selanjutnya pencari kos menginputkan data diri sehingga akan tampil bukti pemesanan.
5. Apabila pencari kos sudah dapat bukti pemesanan maka pencari dapat melakukan input pembayaran, jika pencari kos sudah melakukan pembayaran maka pencari kos dapat melihat status pada menu cek status.
6. Bagi pemilik kos yang sudah mendaftar untuk mengelola data, pemilik cukup melakukan login dengan memasukan username dan password.
7. Untuk pemilik kos yang baru dapat melakukan pemasangan iklan dengan mengisi biodata diri dan selanjutnya mengisi data-data mengenai kos.
8. Apabila ada pencari kos yang sudah melakukan pesan kos maka pemilik akan mengkonfirmasi pemesanan.
9. Selanjutnya pemilik dapat melihat saldo, maksud saldo disini ada total uang yang sudah ditransfer oleh pemesan kos yang sudah dikonfirmasi tersebut.
10. Namun saldo dapat dicairkan apabila pemilik kos sudah meminta pencair kepada admin. Yang mana adminlah yang akan memproses transaksi yang terjadi.
11. Bagian admin hanya mengontrol transaksi yang terjadi dan melihat laporan yang ada.

C. Analisa Kebutuhan Aplikasi

Dalam perancangan Sistem Informasi Rumah Kos Disekitar Kampus IAIN Batusangkar Berbasis Android terdapat beberapa kebutuhan perangkat keras (*Hardware*) dan kebutuhan perangkat lunak (*Software*).

1. Kebutuhan Perangkat Keras

Kebutuhan perangkat keras dalam perancangan Sistem Informasi Rumah Kos Disekitar Kampus IAIN Batusangkar Berbasis Android dapat dibagi menjadi 2 yaitu:

a. Personal Komputer (PC)

Adapun personal komputer (PC) yang penulis gunakan adalah laptop ASUS X455L dengan spesifikasi sebagai berikut:

- 1) Processor Intel® Inside CORE i3-4005U
- 2) Hardisk Drive 500GB
- 3) Random Access Memory (RAM) 2GB

b. Smartphone

Adapun smartphone yang penulis gunakan adalah SAMSUNG Tab 3 dengan spesifikasi sebagai berikut:

- 1) Chipset Intel SM-T211
- 2) RAM 1GB

2. Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak dalam perancangan Sistem Informasi Rumah Kos Disekitar Kampus IAIN Batusangkar Berbasis Android juga terbagi menjadi 2 yaitu:

a. Personal Komputer (PC)

Adapun perangkat lunak yang penulis gunakan pada laptop ASUS X455L adalah sebagai berikut:

- 1) Sistem Operasi Microsoft Windows 8 Pro 32 bit
- 2) Adobe Dreamweaver CS5 Extended
- 3) Microsoft Office Visio 2007

b. Smartphone

D. Perancangan Sistem

1. Actor

Actor yang berperan dalam Sistem Informasi Rumah Kos Disekitar Kampus IAIN Batusangkar Berbasis Android adalah 3 *user* yaitu

admin, pemilik kos, dan pencari kos. Peran *user* dapat dilihat pada tabel dibawah ini:

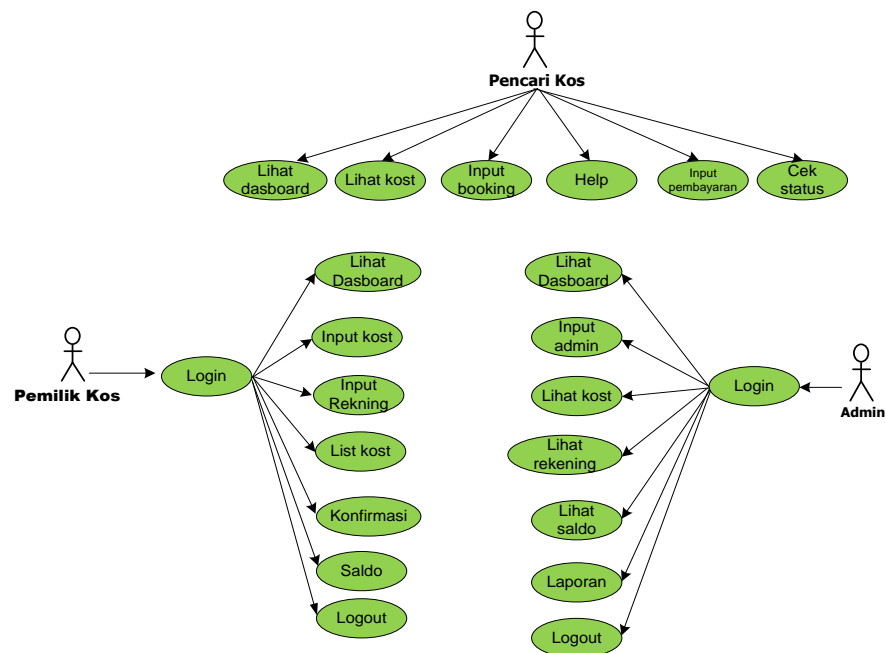
Tabel 3. 1 Peran Actor

Actor	Peran
Admin	Actor yang memonitoring semua system
Pencari kos	Actor yang melakukan pencari kos atau pembookingan kamar kos
Pemilik kos	Actor yang melakukan penginputan data-data kos

2. Use Case Diagram

a. Use Case Diagram Sistem Informasi Rumah Kos

Diagram ini digunakan untuk menggambarkan *user* aplikasi dan perilaku *user* terhadap aplikasi. *User* yang diwakili oleh actor sedangkan perilakunya diwakili oleh *use case*. Untuk lebih jelasnya dapat dilihat pada gambar berikut:



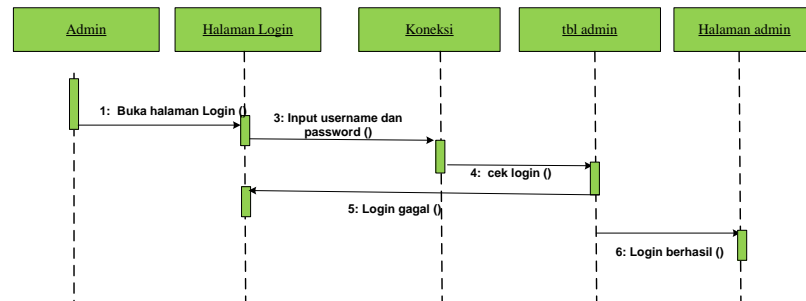
Gambar 3. 1 Use Case Diagram

3. Sequence Diagram

Sequence diagram menjelaskan interaksi antar objek di dalam dan di sekitar sistem berupa pesan (*message*) yang disusun dalam suatu urutan waktu yaitu urutan kejadian yang dilakukan oleh seorang aktor dalam menjalankan sistem. *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai response dari sebuah kegiatan untuk menghasilkan *output* tertentu. Adapun *sequence diagram* dapat dilihat dari gambar-gambar berikut :

a. Sequence Diagram Login

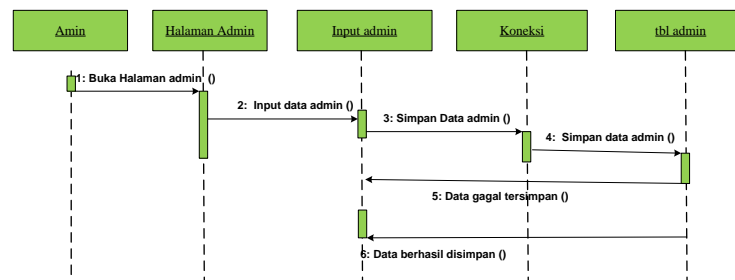
Sequence diagram login menggambarkan urutan event dan waktu saat admin atau kepala bagian melakukan login ke sistem, *sequence diagram* login dapat dilihat pada gambar 3.2.



Gambar 3. 2 Sequence Diagram Login

b. Sequence Diagram Input Admin

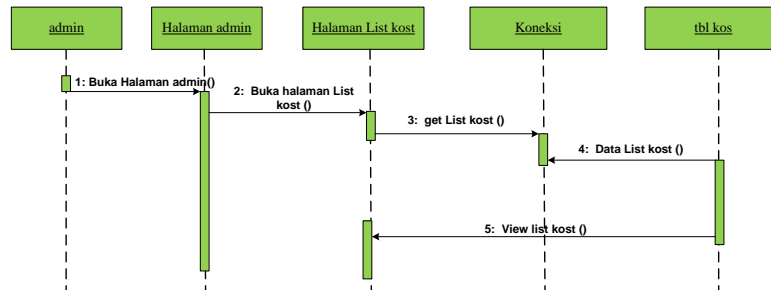
Sequence diagram kategori pada halaman admin dengan urutan event dan waktu admin entri data. *Sequence diagram* input admin dapat dilihat pada gambar 3.3.



Gambar 3. 3 Sequence Diagram Input Admin

c. Sequence Diagram Lihat Kos

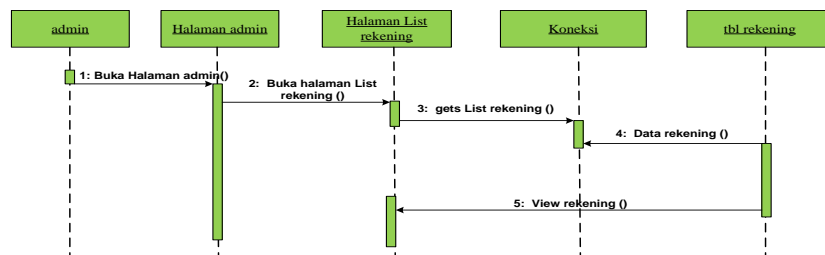
Diagram ini menjelaskan langkah-langkah yang dilakukan oleh admin untuk melihat halaman list kos atau daftar rumah kos, *sequence diagram* lihat kos dapat dilihat pada gambar 3.4.



Gambar 3. 4 Sequence Diagram Lihat Kos

d. Sequence Diagram Lihat Rekening

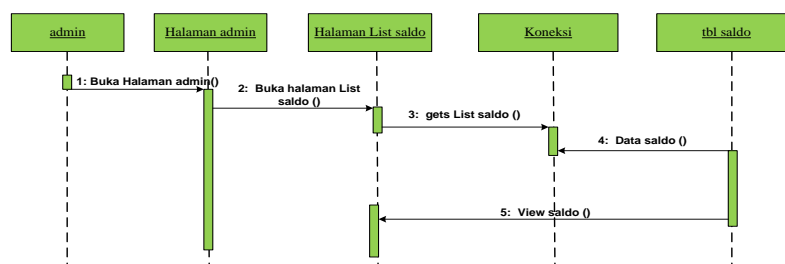
Diagram ini menjelaskan langkah-langkah yang dilakukan oleh admin untuk melihat halaman daftar rekening, *sequence diagram* lihat rekening dapat dilihat pada gambar 3.5.



Gambar 3. 5 Sequence Diagram Lihat Rekening

e. Sequence Diagram Lihat Saldo

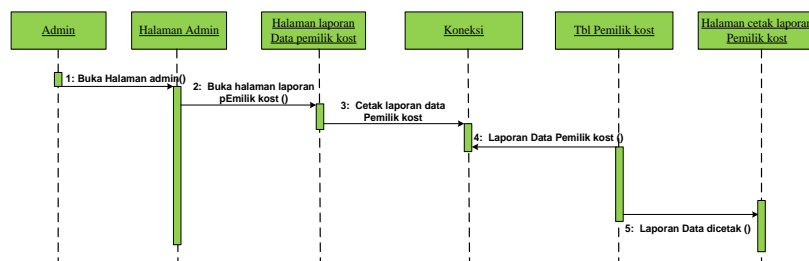
Diagram ini menjelaskan langkah-langkah yang dilakukan oleh admin untuk melihat halaman daftar saldo, *sequence diagram* lihat saldo dapat dilihat pada gambar 3.6.



Gambar 3. 6 Sequence Diagram Lihat Saldo

f. Sequence Diagram Laporan Data Pemilik Kos

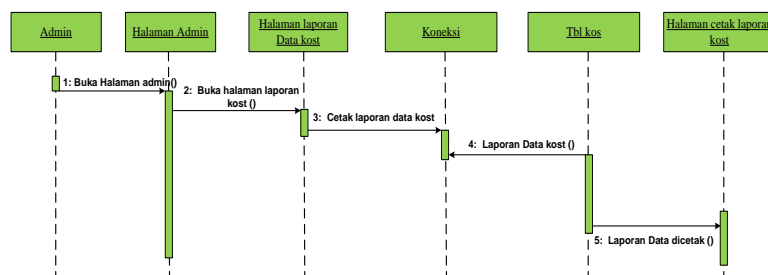
Diagram ini menjelaskan langkah-langkah yang dilakukan admin untuk mencetak laporan data pemilik kos, *sequence diagram* laporan data pemilik kos dapat dilihat pada gambar 3.7.



Gambar 3. 7 Sequence Diagram Laporan Data Pemilik Kos

g. Sequence Diagram Laporan Data Rumah Kos

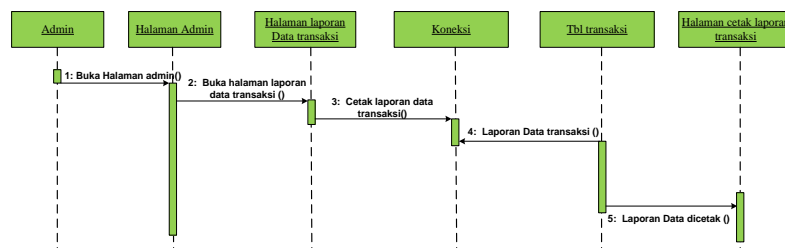
Diagram ini menjelaskan langkah-langkah yang dilakukan admin untuk mencetak laporan data rumah kos, *sequence diagram* laporan data rumah kos dapat dilihat pada gambar 3.8.



Gambar 3. 8 Sequence Diagram Laporan Data Rumah Kos

h. Sequence Diagram Laporan Data Transaksi

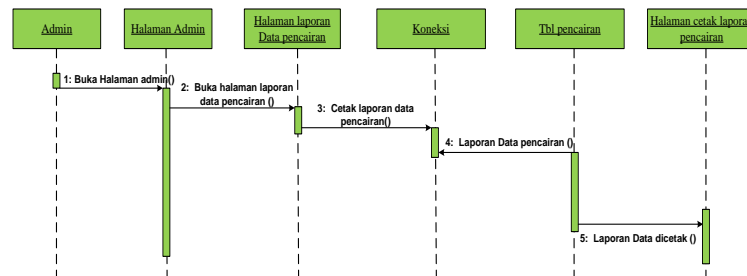
Diagram ini menjelaskan langkah-langkah yang dilakukan admin untuk mencetak laporan data transaksi yang sudah terjadi, *sequence diagram* laporan data transaksi dapat dilihat pada gambar 3.9.



Gambar 3. 9 Sequence Diagram Laporan Data Transaksi

i. Sequence Diagram Laporan Data Pencairan

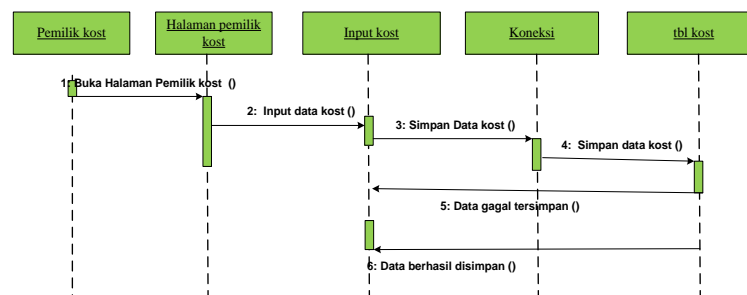
Diagram ini menjelaskan langkah-langkah yang dilakukan admin untuk mencetak laporan data pencairan, *sequence diagram* laporan data pencairan dapat dilihat pada gambar 3.10.



Gambar 3. 10 Sequence Diagram Laporan Data Pencairan

j. Sequence Diagram Input Kos

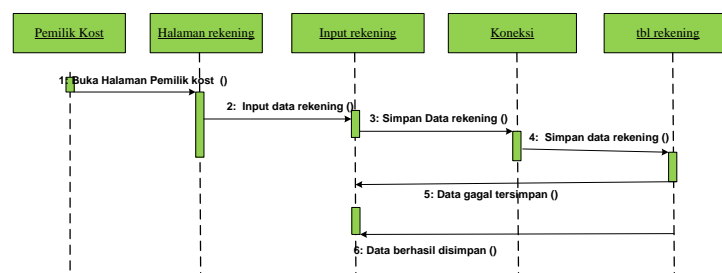
Diagram ini menjelaskan langkah-langkah yang dilakukan oleh pemilik kos untuk menginputkan kos, *sequence diagram input kos* dapat dilihat pada gambar 3.11.



Gambar 3. 11 Sequence Diagram Input Kos

k. Sequence Diagram Input Rekening

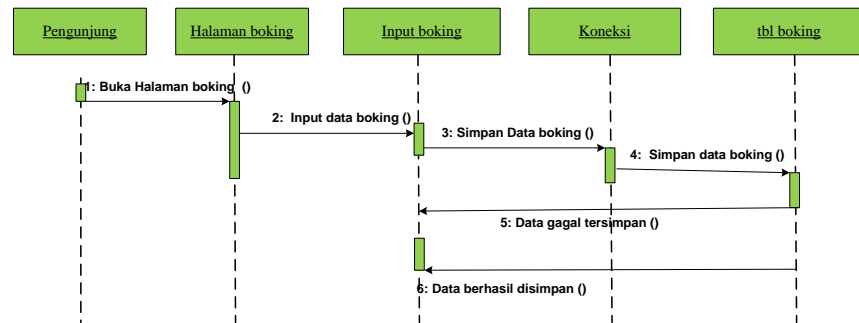
Diagram ini menjelaskan langkah-langkah yang dilakukan pemilik kos untuk menginputkan data rekening, *sequence diagram input rekening* dapat dilihat pada gambar 3.12.



Gambar 3. 12 Sequence Diagram Input Rekening

1. Sequence Diagram Input Booking

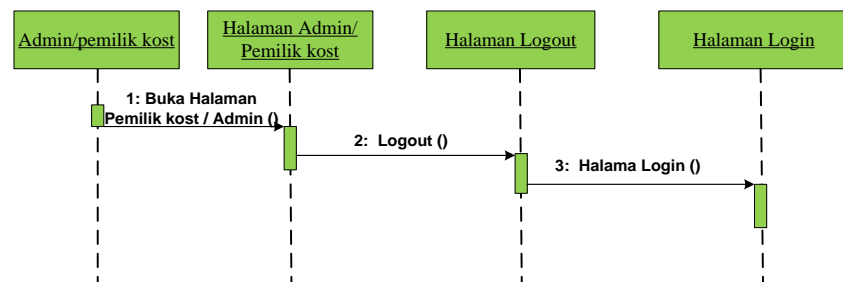
Diagram ini menjelaskan langkah-langkah yang dilakukan pencari kos untuk melihat daftar rumah kos dan melakukan pemesanan, *sequence diagram input booking* dapat dilihat pada gambar 3.13.



Gambar 3. 13 Sequence Diagram Input Booking

m. Sequence Diagram Logout

Diagram ini menjelaskan langkah-langkah yang dilakukan admin dan pemilik kos untuk logout atau keluar halaman, *sequence diagram logout* dapat dilihat pada gambar 3.14.



Gambar 3. 14 Sequence Diagram Logout

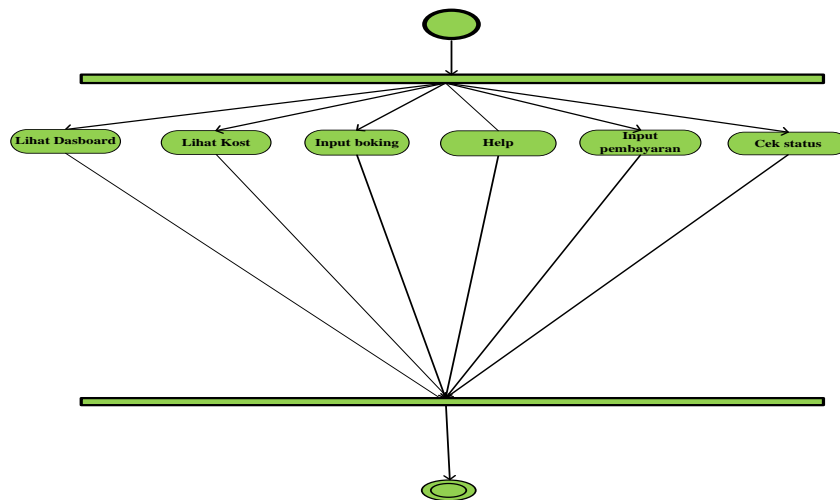
4. Activity Diagram

Merupakan diagram yang menggambarkan berbagai aliran aktifitas dalam sistem yang sedang dirancang bagaimana masing-masing aliran berawal, *decision* yang mungkin terjadi dan bagaimana mereka berakhir.

a. Activity Diagram Pencari Kos

Activity diagram pencari kos menggambarkan aktivitas-aktivitas yang dapat dilakukan oleh pencari kos di dalam sistem

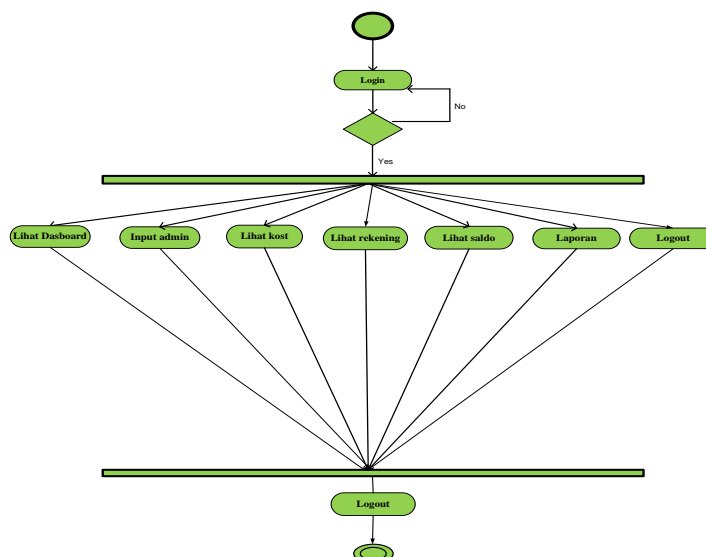
informasi pemesanan kos. *Activity diagram* pencari kos dapat dilihat pada gambar 3.15.



Gambar 3. 15 Activity Diagram Pencari Kos

b. Activity Diagram Admin

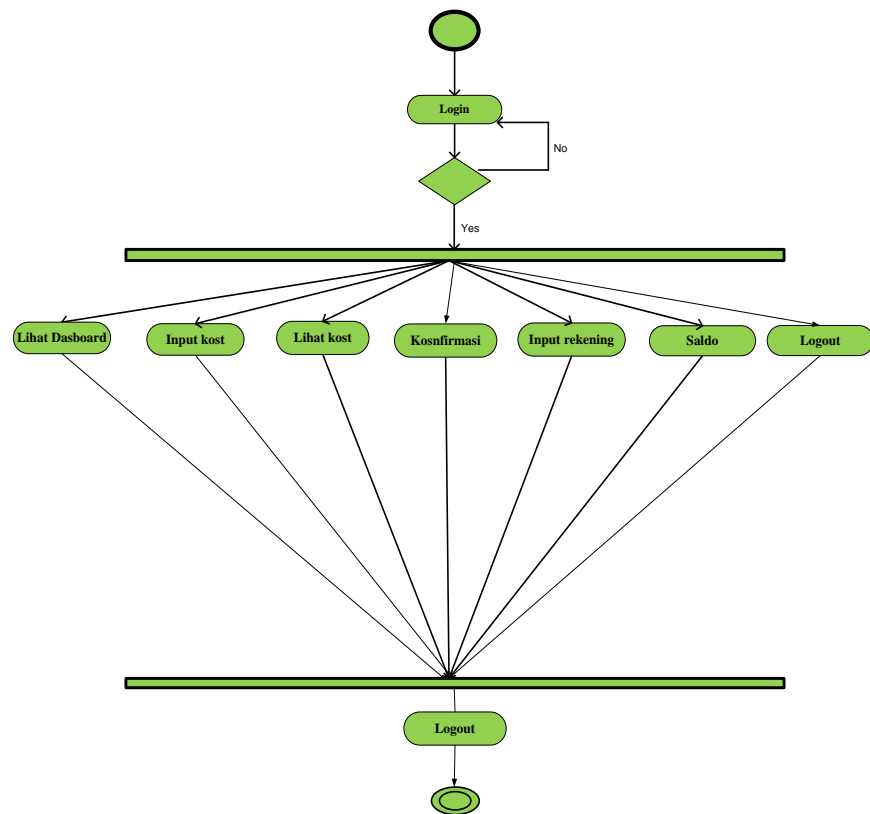
Activity diagram admin menggambarkan proses admin untuk melakukan aktifitas dan menu yang digunakan pada sistem. *Activity diagram* admin yang pertama admin harus melakukan *login* terlebih dahulu. *Activity diagram* admin dapat dilihat pada gambar 3.16.



Gambar 3. 16 Activity Diagram Admin

c. Activity Diagram Pemilik Kos

Activity diagram pemilik kos menggambarkan proses pemilik kos untuk melakukan aktifitas dan menu yang digunakan pada sistem. *Activity diagram* pemilik kos yang pertama harus melakukan *login* terlebih dahulu. *Activity diagram* pemilik kos dapat dilihat pada gambar 3.17.



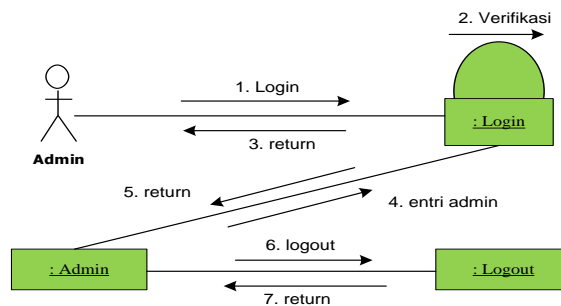
Gambar 3. 17 Activity Diagram Pemilik Kos

5. Collaboration Diagram

Collaboration diagram juga menggambarkan interaksi antara objek seperti *sequence diagram*, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian *message*, setiap *message* memiliki *sequence number*, dimana *message* dari level tertinggi memiliki nomor 1. *Message* dari level yang sama memiliki 55 refix yang sama. Adapun *collaboration diagram* dari analisa dan penerapan aplikasi ini dapat digambarkan sebagai berikut :

a. Collaboration Diagram Admin

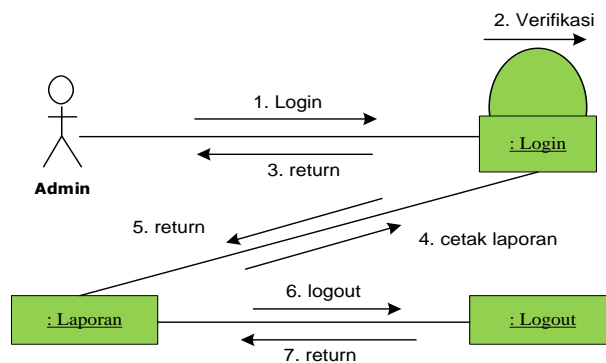
Collaboration diagram admin menggambarkan proses admin dalam melakukan input data admin yang digambarkan pada gambar 3.18.



Gambar 3. 18 Collaboration Diagram Admin

b. Collaboration Diagram Laporan

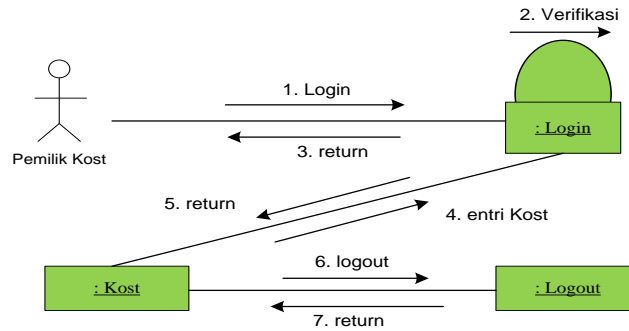
Collaboration diagram laporan menggambarkan proses admin dalam melihat laporan yang digambarkan pada gambar 3.19.



Gambar 3. 19 Collaboration Diagram Laporan

c. Collaboration Diagram Input Kos

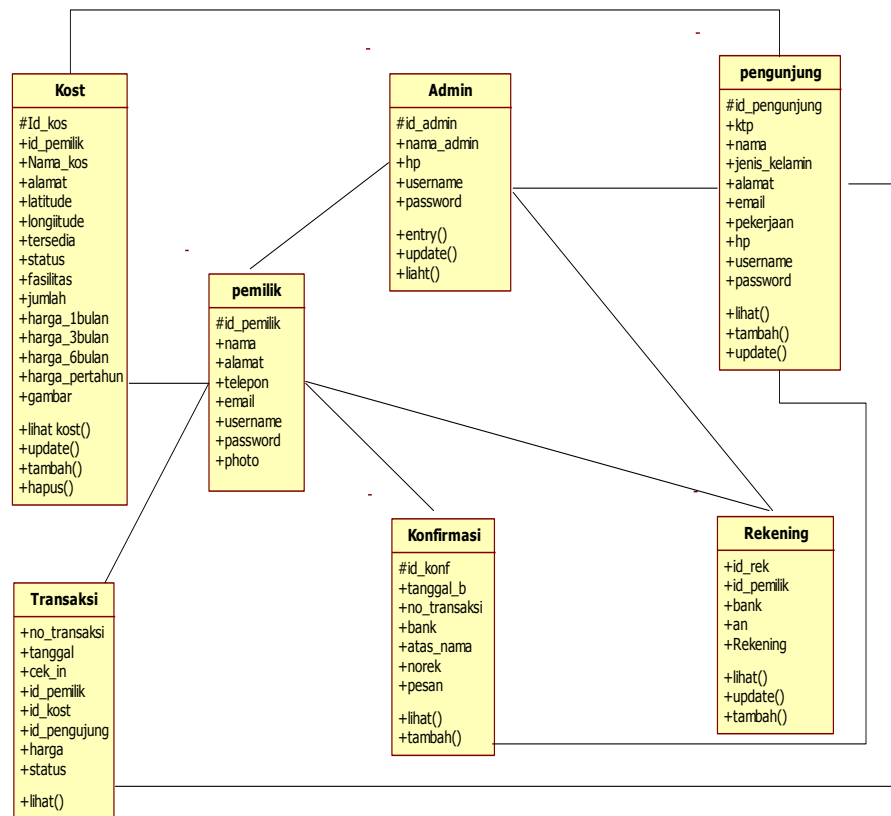
Collaboration diagram input kos menggambarkan proses pemilik kos dalam melakukan input kos yang digambarkan pada gambar 3.20.



Gambar 3. 20 Collaboration Diagram Input Kos

6. Class Diagram

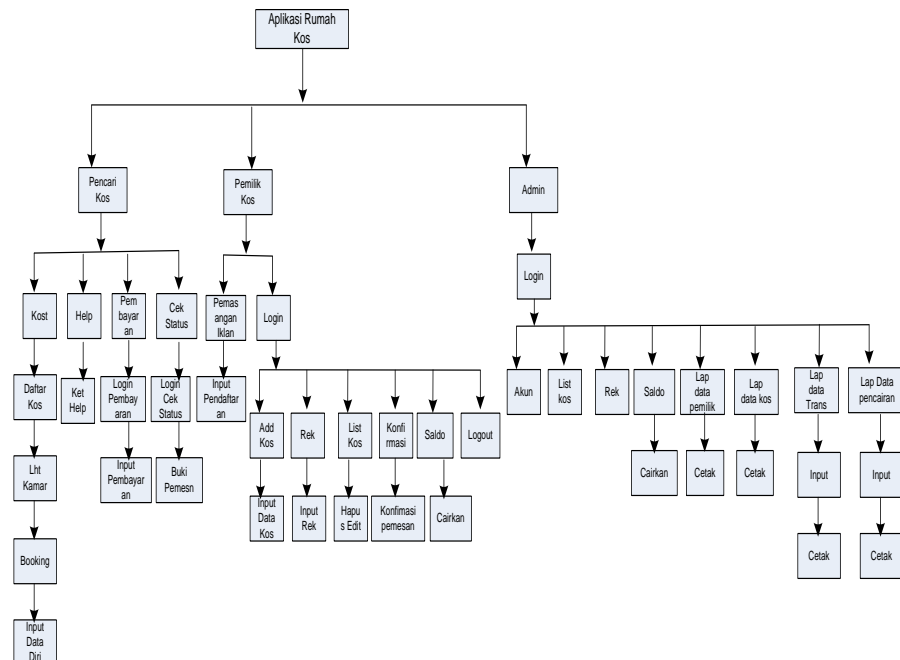
Class diagram adalah sebuah spesifikasi yang jika diinstansi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class diagram* dari Sistem informasi Rumah Kos Disekitar Kampus IAIN Batusangkar Berbasis Android dapat dilihat pada gambar 3.21.



Gambar 3. 21 Class Diagram

7. Struktur Program

Desain struktur program merupakan suatu desain yang menggambarkan suatu hubungan modul program dengan modul program yang lainnya. Desain struktur program dari yang diusulkan oleh penulis dapat dilihat pada gambar sebagai berikut:



Gambar 3. 22 Struktur Program

E. Desain Input

Dalam setiap pemrosesan perlu ada data masukan, dimana data yang akan diproses harus dimasukkan terlebih dahulu, tentunya melalui *interface* (perangkat penghubung) antara pengguna dengan hardware dan software. Untuk itu agar memudahkan dan tidak terjadinya kesalahan pemasukan data, maka dirancang bentuk menu tampilan yang mudah digunakan untuk memasukkan data tersebut.

Input atau masukan merupakan awal dimulainya proses informasi. Data input merupakan salah satu bahan untuk beberapa keluaran yang lebih dahulu melalui proses komputer. Adapun bentuk desain input yang dirancang adalah sebagai berikut :

1. Tampilan Halaman Login Admin

Forms Login Admin	
Form Login Admin	
User Name	<input type="text" value="Varchar(20)"/>
Password	<input type="text" value="Varchar(60)"/>
<input type="button" value="Login"/>	<input type="button" value="Reset"/>

Gambar 3. 23 Desain Input Login Admin

2. Tampilan Halaman Akun Admin

Header	
Logo	Akun
	Akun
<input type="button" value="Dashboard"/> <input type="button" value="Akun"/> <input type="button" value="List Kos"/> <input type="button" value="Rekening"/> <input type="button" value="Saldo"/> <input type="button" value="Laporan Data Pemilik"/> <input type="button" value="Laporan Data Kos"/> <input type="button" value="Laporan Data Transaksi"/> <input type="button" value="Laporan Data Pencairan"/> <input type="button" value="Logout"/>	Nama Admin <input type="text" value="Varchar (25)"/> Telpon <input type="text" value="Int (12)"/> UserName <input type="text" value="Varchar(25)"/> Password <input type="text" value="Varchar (20)"/> <input type="button" value="Sumbit"/> <input type="button" value="Reset"/>

Gambar 3. 24 Desain Input Akun Admin

3. Tampilan Halaman Laporan Data Transaksi

Header	
Logo	Laporan Data Booking
	Laporan Data Booking Perbulan Perpemilik Kos
<input type="button" value="Dashboard"/> <input type="button" value="Akun"/> <input type="button" value="List Kos"/> <input type="button" value="Rekening"/> <input type="button" value="Saldo"/> <input type="button" value="Laporan Data Pemilik"/> <input type="button" value="Laporan Data Kos"/> <input type="button" value="Laporan Data Transaksi"/> <input type="button" value="Laporan Data Pencairan"/> <input type="button" value="Logout"/>	Nama pemilik Kost <input type="text" value="-Pilih Pemilik- ▼"/> Bulan <input type="text" value="Bulan ▼"/> <input type="text" value="Tahun ▼"/> <input type="button" value="Sumbit"/> <input type="button" value="Reset"/>
	Laporan Data Booking Perbulan
	Bulan <input type="text" value="Bulan ▼"/> <input type="text" value="Tahun ▼"/> <input type="button" value="Sumbit"/> <input type="button" value="Reset"/>
	Laporan Data Booking Pertahun Perpemilik Kos
	Nama pemilik Kost <input type="text" value="-Pilih Pemilik- ▼"/> Tahun <input type="text" value="Tahun ▼"/> <input type="button" value="Sumbit"/> <input type="button" value="Reset"/>
	Laporan Data Booking Pertahun
	Tahun <input type="text" value="Tahun ▼"/> <input type="button" value="Sumbit"/> <input type="button" value="Reset"/>

Gambar 3. 25 Desain Input Laporan Data Transaksi

4. Tampilan Halaman Laporan Data Pencairan

Header	
Logo	Laporan Data Pencairan
	Laporan Data Pencairan Perbulan
Dashboard Akun List Kos Rekening Saldo Laporan Data Pemilik Laporan Data Kos Laporan Data Transaksi Laporan Data Pencairan Logout	Bulan Bulan ▼ Tahun ▼ Submit Reset
	Laporan Data Pencairan Pertahun
	Tahun Tahun ▼ Submit Reset

Gambar 3. 26 Desain Input Laporan Data Pencairan

5. Tampilan Halaman Pendaftaran Pemilik Kos

Header	
Logo	Forms Pendaftaran
	Form Pendaftaran Pemilik Kos
Dashboard Pasang Iklan Kost Help Pembayaran Cek Status Login	Nama Varchar(20)
	Alamat Varchar(30)
	Nomor Telepon Varchar(12)
	Email Varchar(30)
	User Name Varchar(20)
	Password Varchar(60)
	Photo Browser
	Submit Reset

Gambar 3. 27 Desain Input Pendaftaran Pemilik Kos

6. Tampilan Halaman Login Pemilik Kos

Header	
Logo	Froms Login
	Form Login Pemilik Kos
<p>Dashboard</p> <p>Pasang Iklan</p> <p>Kost</p> <p>Help</p> <p>Pembayaran</p> <p>Cek Status</p> <p>Login</p>	<p>User Name</p> <p><input type="text" value="Varchar(20)"/></p> <p>Password</p> <p><input type="text" value="Varchar(60)"/></p> <p><input type="button" value="Login"/> <input type="button" value="Reset"/></p>

Gambar 3. 28 Desain Input Login Pemilik Kos

7. Tampilan Halaman Add Rumah Kos

Form Add Kos	
Logo	Form Add Kos
	<p>Nama Kos</p> <p><input type="text" value="Varchar(25)"/></p> <p>Status Kos</p> <p><input type="text" value="Varchar(20) ▼"/></p> <p>Alamat Kos</p> <p><input type="text" value="Text"/></p> <p>Fasilitas</p> <p><input type="text" value="Text"/></p> <p>Jumlah Kamar</p> <p><input type="text" value="Int(3)"/></p> <p>Kamar yang tersedia</p> <p><input type="text" value="Int(3)"/></p> <p>Harga 1 bulan</p> <p><input type="text" value="Int(11)"/></p> <p>Harga 3 bulan</p> <p><input type="text" value="Int(9)"/></p> <p>Harga 6 bulan</p> <p><input type="text" value="Int(9)"/></p> <p>Harga 1 tahun</p> <p><input type="text" value="Int(9)"/></p> <p>Photo Rumah Kos</p> <p><input type="button" value="Browser"/></p> <p>Photo Kamar</p> <p><input type="button" value="Browser"/></p> <p><input type="button" value="Sumbit"/> <input type="button" value="Reset"/></p>
<p>Dashboard</p> <p>Add Kost</p> <p>Rekening</p> <p>List Kos</p> <p>Konfirmasi</p> <p>Saldo</p> <p>Logout</p>	

Gambar 3. 29 Desain Input Kos

8. Tampilan Halaman Add Rekening Pemilik Kos

Header	
Picture	Forms Add Rekening
	Form Add Rekening
<p>Dashboard</p> <p>Add Kost</p> <p>Rekening</p> <p>List Kos</p> <p>Konfirmasi</p> <p>Saldo</p> <p>Logout</p>	Bank <input type="text" value="Varchar(10)"/>
	Atas nama <input type="text" value="Varchar(25)"/>
	No Rek <input type="text" value="Int(30)"/>
	<input type="button" value="Submit"/> <input type="button" value="Reset"/>

Gambar 3. 30 Desain Input Add Rekening Pemilik Kos

9. Tampilan Halaman Booking Kamar

Header	
Logo	Forms Booking
	Form Booking
<p>Dashboard</p> <p>Pasang Iklan</p> <p>Kost</p> <p>Help</p> <p>Pembayaran</p> <p>Cek Status</p> <p>Login</p>	Nama kost <input type="text" value="Varchar (25)"/>
	Pemilik <input type="text" value="Varchar (20)"/>
	Cek in <input type="text" value="Tanggal"/>
	<input type="text" value="Bulan"/>
	<input type="text" value="Tahun"/>
	Lama <input type="radio"/> 1 Bulan : Rp.999.999.999 <input type="radio"/> 3 Bulan : Rp.999.999.999 <input type="radio"/> 6 Bulan : Rp.999.999.999 <input type="radio"/> 1 Tahun : Rp.999.999.999
	Nomor KTP <input type="text" value="Int(3)"/>
	Nama <input type="text" value="Int(11)"/>
	Jenis Kelamin <input type="radio"/> Laki - Laki <input type="radio"/> Perempuan
	Alamat <input type="text" value="Int(9)"/>
	Email <input type="text" value="Int(9)"/>
	Pekerjaan <input type="text" value="Int(9)"/>
	Telpon <input type="text"/>
	Username <input type="text"/>
	Password <input type="text"/>
	<input type="button" value="Submit"/> <input type="button" value="Reset"/>

Gambar 3. 31 Desain Input Booking Kamar

10. Tampilan Halaman Login Pembayaran

Header	
Logo	Froms Login
	Form Login Pembayaran
<p>Dashboard</p> <p>Pasang Iklan</p> <p>Kost</p> <p>Help</p> <p>Pembayaran</p> <p>Cek Status</p> <p>Login</p>	<p>User Name</p> <p><input type="text" value="Varchar(20)"/></p> <p>Password</p> <p><input type="text" value="Varchar(60)"/></p> <p><input type="button" value="Login"/> <input type="button" value="Reset"/></p>

Gambar 3. 32 Desain Input Login Pembayaran

11. Tampilan Halaman Pembayaran

Header	
Logo	Forms Pembayaran
	Input Pembayaran
<p>Dashboard</p> <p>Pasang Iklan</p> <p>Kost</p> <p>Help</p> <p>Pembayaran</p> <p>Cek Status</p> <p>Login</p>	<p>Tanggal Transaksi : 99-99-9999</p> <p>No. Transaksi : (varchar20)</p> <p>Nomor KTP : (varchar20)</p> <p>Nama : (varchar40)</p> <p>Jenis Kelamin : (varchar30)</p> <p>No Telepon : (varchar13)</p> <p>Pekerjaan : (varchar50)</p> <p>Chek-in : 99-99-9999</p> <p>Harga Kost : (int11)</p> <p>Status Transaksi : (varchar35)</p> <p>Total Bayar : (int20)</p> <p>Bank</p> <p><input type="text" value="Varchar(10)"/> ▼</p> <p>Atas nama</p> <p><input type="text" value="Varchar(25)"/></p> <p>No. Rekening</p> <p><input type="text" value="Int(30)"/></p> <p>Pesan</p> <p><input type="text" value="Text"/></p> <p><input type="button" value="Submit"/> <input type="button" value="Reset"/></p>

Gambar 3. 33 Desain Input Pembayaran

12. Tampilan Halaman Login Cek Status

Header	
Logo	Froms Login
	Form Login Cek Status
<p>Dashboard</p> <p>Pasang Iklan</p> <p>Kost</p> <p>Help</p> <p>Pembayaran</p> <p>Cek Status</p> <p>Login</p>	<p>User Name</p> <p><input type="text" value="Varchar(20)"/></p> <p>Password</p> <p><input type="text" value="Varchar(60)"/></p> <p><input type="button" value="Cek Status"/> <input type="button" value="Reset"/></p>

Gambar 3. 34 Desain Input Login Cek Status

F. Desain Output

Desain Output merupakan bentuk laporan yang dihasilkan sistem yang dirancang sedemikian rupa sehingga dapat digunakan untuk kemajuan suatu usaha dan dapat dipakai sebagai bahan perbandingan oleh pimpinan dalam mengambil keputusan.

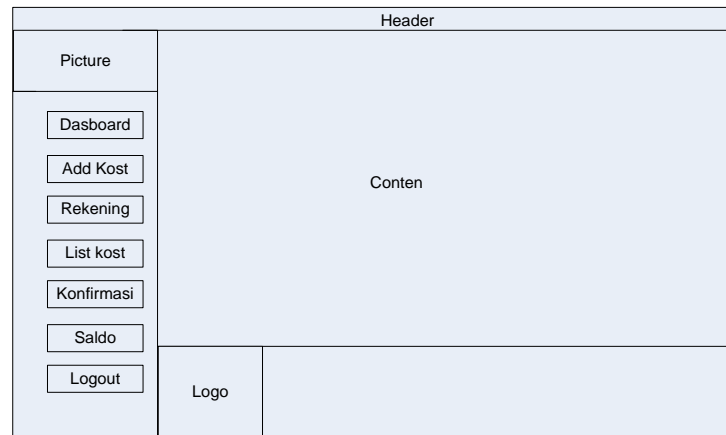
Adapun disain output yang telah penulis rancang adalah sebagai berikut :

1. Tampilan Halaman Admin

Header	
Logo	Conten
Dashboard	
Akun	
List Kos	
Rekening	
Saldo	
Laporan Data Pemilik	
Laporan Data Kos	
Laporan Data Transaksi	
Laporan Data Pencairan	
Logout	
Logo	

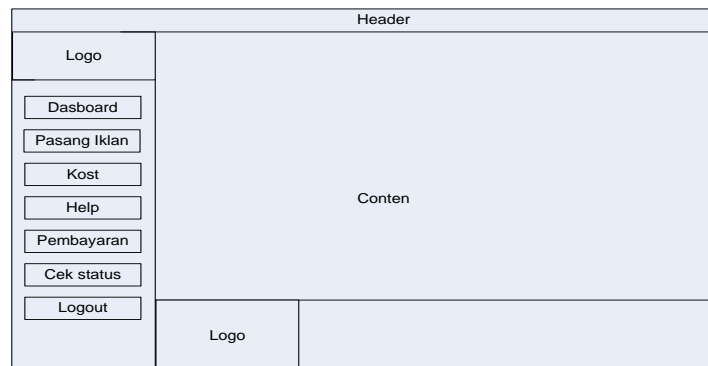
Gambar 3. 35 Desain Output Halaman Admin

2. Tampilan Halaman Pemilik Kos



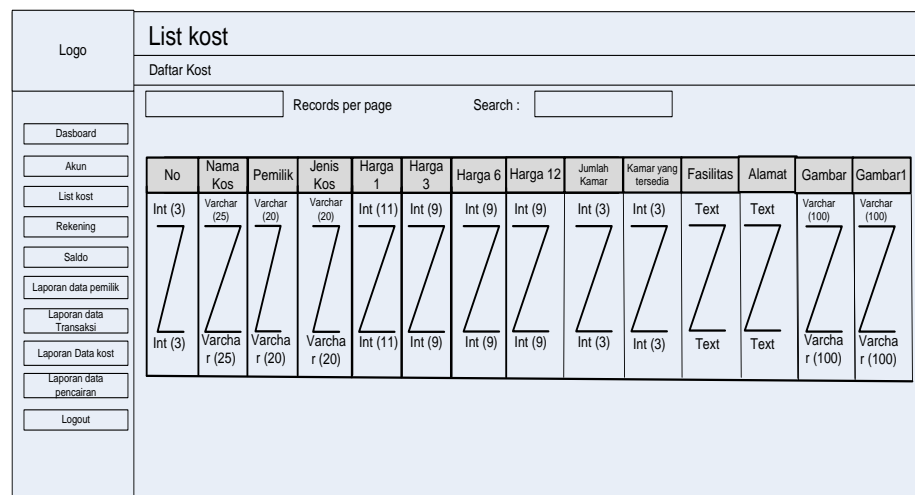
Gambar 3. 36 Desain Output Halaman Pemilik Kos

3. Tampilan Halaman Pencari Kos



Gambar 3. 37 Desain Output Halaman Pencari Kos

4. Tampilan Halaman List Kos Admin



Gambar 3. 38 Desain Output List Kos Pada Admin

5. Tampilan Halaman Rekening Admin

Header																				
Logo	Forms Data Rekening																			
	Daftar Rekening																			
<ul style="list-style-type: none"> Dashboard Akun List Kos Rekening Saldo Laporan data pemilik Laporan data Transaksi Laporan Data kost Laporan data pencairan Logout 	<table border="1"> <thead> <tr> <th>Pemilik Kos</th> <th>No Rekening</th> <th>Nama Bank</th> <th>Atas Nama</th> </tr> </thead> <tbody> <tr> <td>Varchar (20)</td> <td>Varchar (20)</td> <td>Varchar (35)</td> <td>Varchar (20)</td> </tr> <tr> <td>Z</td> <td>Z</td> <td>Z</td> <td>Z</td> </tr> <tr> <td>Varchar (20)</td> <td>Varchar (20)</td> <td>Varchar (35)</td> <td>Varchar (20)</td> </tr> </tbody> </table>	Pemilik Kos	No Rekening	Nama Bank	Atas Nama	Varchar (20)	Varchar (20)	Varchar (35)	Varchar (20)	Z	Z	Z	Z	Varchar (20)	Varchar (20)	Varchar (35)	Varchar (20)			
Pemilik Kos	No Rekening	Nama Bank	Atas Nama																	
Varchar (20)	Varchar (20)	Varchar (35)	Varchar (20)																	
Z	Z	Z	Z																	
Varchar (20)	Varchar (20)	Varchar (35)	Varchar (20)																	


Gambar 3. 39 Desain Output Daftar Rekening Pada Admin

6. Tampilan Halaman Saldo Admin

Header																													
Logo	Saldo																												
	Saldo																												
<ul style="list-style-type: none"> Dashboard Akun List Kos Rekening Saldo Laporan data pemilik Laporan data Transaksi Laporan Data kost Laporan data pencairan Logout 	<input type="text"/>	Records per page	Search : <input type="text"/>																										
	<table border="1"> <thead> <tr> <th>No</th> <th>Pemilik Kos</th> <th>Tanggal</th> <th>Saldo</th> <th>Status</th> <th>Aksi</th> </tr> </thead> <tbody> <tr> <td>Int (3)</td> <td>Varchar (20)</td> <td>Date</td> <td>Int (11)</td> <td>Enum</td> <td>Varchar (5)</td> </tr> <tr> <td>Z</td> <td>Z</td> <td>Z</td> <td>Z</td> <td>Z</td> <td>Z</td> </tr> <tr> <td>Int(3)</td> <td>Varchar (20)</td> <td>Date</td> <td>Int (11)</td> <td>Enum</td> <td>Varchar (5)</td> </tr> </tbody> </table>	No	Pemilik Kos	Tanggal	Saldo	Status	Aksi	Int (3)	Varchar (20)	Date	Int (11)	Enum	Varchar (5)	Z	Z	Z	Z	Z	Z	Int(3)	Varchar (20)	Date	Int (11)	Enum	Varchar (5)				
No	Pemilik Kos	Tanggal	Saldo	Status	Aksi																								
Int (3)	Varchar (20)	Date	Int (11)	Enum	Varchar (5)																								
Z	Z	Z	Z	Z	Z																								
Int(3)	Varchar (20)	Date	Int (11)	Enum	Varchar (5)																								
	Total Saldo = Rp.																												

Gambar 3. 40 Desain Output Saldo Pada Admin


7. Tampilan Laporan Data Pemilik Kos

 PAPI KOST LAPORAN DATA PEMILIK KOST				
No.	Nama Pemilik Kos	Alamat	Telpon	Email
Int (3)	Varchar (20)	Varchar (30)	Varchar (12)	Varchar (30)
Z	Z	Z	Z	Z
Int (3)	Varchar (20)	Varchar (30)	Varchar (12)	Varchar (30)

Tanah Datar,99-99-9999
Pemimpin
(.....)

Gambar 3. 41 Desain Output Laporan Data Pemilik Kos

8. Tampilan Laporan Data Kos


 PAPI KOST LAPORAN DATA KOS												
No.	Id Kos	Nama Kos	Pemilik	Jenis Kos	Harga1	Harga3	Harga6	Harga12	Jumlah Kamar	Kamar yang tersedia	Fasilitas	Alamat
Int (3)	Int (11)	Int (11)	Varchar (40)	Varchar (20)	Int (11)	Int (9)	Int (9)	Int (9)	Int (3)	Int (3)	Text	Varchar (50)
Int (3)	Int (11)	Int (11)	Varchar (40)	Varchar (20)	Int (11)	Int (9)	Int (9)	Int (9)	Int (3)	Int (3)	Text	Varchar (50)

Tanah Datar,99-99-9999
Pimpinan
(.....)

Gambar 3. 42 Desain Output Laporan Data Kos

9. Tampilan Laporan Data Transaksi


a. Laporan Data Transaksi Perbulan Perpemilik Kos

 PAPI KOST LAPORAN DATA BOOKING BULAN : TAHUN : PEMILIK KOST :												
No	No. Trans	Tanggal	Chekin	id kos	Nama Kos	Pemilik	Jenis Kos	No.KTP	Pencari Kos	Harga	Bayar	
Int (3)	Varchar (20)	Date	Date	Int (5)	Varchar (25)	Varchar (20)	Varchar (20)	Varchar (20)	Varchar (20)	Int (11)	Int (11)	
Int (3)	Varchar (20)	Date	Date	Int(5)	Varchar (25)	Varchar (20)	Varchar (20)	Varchar (20)	Varchar (20)	Int (11)	Int (11)	

Tanah Datar,99-99-9999
Pimpinan
(.....)

**Gambar 3. 43 Desain Output Laporan Data Transaksi
Perbulan Perpemilik Kos**


b. Laporan Data Transaksi Perbulan

 PAPI KOST LAPORAN DATA BOOKING BULAN : TAHUN :												
No	No. Trans	Tanggal	Chekin	id kos	Nama Kos	Pemilik	Jenis Kos	No.KTP	Pencari Kos	Harga	Bayar	
Int (3)	Varchar (20)	Date	Date	Int (5)	Varchar (25)	Varchar (20)	Varchar (20)	Varchar (20)	Varchar (20)	Int (11)	Int (11)	
Int (3)	Varchar (20)	Date	Date	Int(5)	Varchar (25)	Varchar (20)	Varchar (20)	Varchar (20)	Varchar (20)	Int (11)	Int (11)	

Tanah Datar,99-99-9999
Pimpinan
(.....)

Gambar 3. 44 Desain Output Laporan Data Transaksi Perbulan


c. Laporan Data Transaksi Tahunan Perpemilik Kos

 PAPI KOST LAPORAN DATA BOOKING TAHUN : PEMILIK KOST :							
No	No.Trans	Tanggal	Chekin	id kos	No.KTP	Harga	Bayar
Int (3)	Varchar (20)	Date	Date	Int (5)	Varchar (20)	Int (11)	Int (11)
Z	Z	Z	Z	Z	Z	Z	Z
Int (3)	Varchar (20)	Date	Date	Int(5)	Varchar (20)	Int (11)	Int (11)

Tanah Datar.99-99-9999
Pimpinan
(.....)

Gambar 3. 45 Desain Output Laporan Data Transaksi Tahunan Perpemilik Kos

d. Laporan Data Transaksi Tahunan


 PAPI KOST LAPORAN DATA BOOKING TAHUN :							
No	No.Trans	Tanggal	Chekin	id kos	No.KTP	Harga	Bayar
Int (3)	Varchar (20)	Date	Date	Int (5)	Varchar (20)	Int (11)	Int (11)
Z	Z	Z	Z	Z	Z	Z	Z
Int (3)	Varchar (20)	Date	Date	Int(5)	Varchar (20)	Int (11)	Int (11)

Tanah Datar.99-99-9999
Pimpinan
(.....)

Gambar 3. 46 Desain Output Laporan Data Transaksi Tahunan

10. Tampilan Laporan Data Pencairan


a. Laporan Data Pencairan Perbulan

 PAPI KOST LAPORAN DATA PENCAIRAN BULAN : TAHUN :								
No.	Id Saldo	Tanggal	id Pemilik	Pemilik	Bank	Rekening	Atas Nama	Saldo
Int (3)	Int (11)	Date	Int (11)	Varchar (20)	Varchar (35)	Varchar (20)	Varchar (20)	Int (11)
Z	Z	Z	Z	Z	Z	Z	Z	Z
Int (3)	Int (11)	Date	Int (11)	Varchar (20)	Varchar (35)	Varchar (20)	Varchar (20)	Int (11)

Tanah Datar.99-99-9999
Pimpinan
(.....)

Gambar 3. 47 Desain Output Laporan Data Pencairan Perbulan

b. Laporan Data Pencairan Pertahun

 PAPI KOST LAPORAN DATA PENCAIRAN TAHUN :								
No.	Id Saldo	Tanggal	id Pemilik	Pemilik	Bank	Rekening	Atas Nama	Saldo
Int (3)	Int (11)	Date	Int (11)	Varchar (20)	Varchar (35)	Varchar (20)	Varchar (20)	Int (11)
Int (3)	Int (11)	Date	Int (11)	Varchar (20)	Varchar (35)	Varchar (20)	Varchar (200)	Int (11)

Tanah Datar,99-99-9999
Pimpinan
(.....)

**Gambar 3. 48 Desain Output Laporan Data Pencairan
Pertahun**

11. Tampilan Halaman Rekening Pemilik

Header													
Picture <input type="button" value="Dashboard"/> <input type="button" value="Add Kost"/> <input type="button" value="Rekening"/> <input type="button" value="List Kos"/> <input type="button" value="Konfirmasi"/> <input type="button" value="Saldo"/> <input type="button" value="Logout"/>	Forms Add Rekening Form Add Rekening Bank <input type="text" value="Varchar(10)"/>												
	Atas nama <input type="text" value="Varchar(25)"/> No.Rek <input type="text" value="Int(30)"/> <input type="button" value="Submit"/> <input type="button" value="Reset"/>												
	Daftar Rekening <table border="1"> <thead> <tr> <th>No Rekening</th> <th>Nama Bank</th> <th>Atas Nama</th> <th>Aksi</th> </tr> </thead> <tbody> <tr> <td>Int (30)</td> <td>Varchar (10)</td> <td>Varchar (25)</td> <td>Varchar (5)</td> </tr> <tr> <td>Int (30)</td> <td>Varchar (10)</td> <td>Varchar (25)</td> <td>Varchar (5)</td> </tr> </tbody> </table>	No Rekening	Nama Bank	Atas Nama	Aksi	Int (30)	Varchar (10)	Varchar (25)	Varchar (5)	Int (30)	Varchar (10)	Varchar (25)	Varchar (5)
	No Rekening	Nama Bank	Atas Nama	Aksi									
Int (30)	Varchar (10)	Varchar (25)	Varchar (5)										
Int (30)	Varchar (10)	Varchar (25)	Varchar (5)										

Gambar 3. 49 Desain Output Daftar Rekening Pada Pemilik

12. Tampilan Halaman List Kos Pemilik

Logo	List kost												
Dashboard <input type="button" value="Add Kost"/> <input type="button" value="Rekening"/> <input type="button" value="List Kos"/> <input type="button" value="Konfirmasi"/> <input type="button" value="Saldo"/> <input type="button" value="Logout"/>	Daftar Kost												
	<input type="text"/> Records per page Search : <input type="text"/>												
	Nama Kos	Jenis Kos	Harga 1	Harga 3	Harga 6	Harga 12	Jumlah Kamar	Kamar yang tersedia	Fasilitas	Alamat	Gambar	Gambar1	Aksi
	Varchar(20)	Varchar(20)	Int (11)	Int (9)	Int (9)	Int (9)	Int (3)	Int (3)	Text	Text	Varchar (100)	Varchar (100)	Varchar (5)
	Varchar(20)	Varchar(20)	Int (11)	Int (9)	Int (9)	Int (9)	Int (3)	Int (3)	Text	Text	Varchar (100)	Varchar (100)	Varchar (5)

Gambar 3. 50 Desain Output List Kos Pada Pemilik

13. Tampilan Halaman Konfirmasi

Header										
Picture	KONfirmasi									
	KONfirmasi Pemesan									
Dashboard Add Kost Rekening List Kos Konfirmasi Saldo Logout	▼ Records per page		Search : <input type="text"/>							
	No	No. Tran	Nama Kos	Tanggal	Chek-in	KTP	Nama	Telepon	Harga	Aksi
	Int (3)	Varchar (20)	Varchar (25)	Date	Date	Varchar (20)	Varchar (40)	Varchar (13)	Int (11)	Varchar (5)
	/	/	/	/	/	/	/	/	/	/
	Int(3)	Varchar (20)	Varchar (25)	Date	Date	Varchar (20)	Varchar (40)	Varchar (13)	Int (11)	Varchar (5)
	Total Saldo = Rp. Total Saldo Telah Dicarikan = Rp.									

Gambar 3. 51 Desain Output Konfirmasi Pemesanan

14. Tampilan Halaman Saldo Pemilik

Header					
Picture	Saldo				
	Saldo				
Dashboard Add Kost Rekening List Kos Konfirmasi Saldo Logout	▼ Records per page		Search : <input type="text"/>		
	No	Tanggal	Saldo	Status	Aksi
	Int (3)	Date	Int (11)	Enum	Varchar (5)
	/	/	/	/	/
	Int(3)	Date	Int (11)	Enum	Varchar (5)
	Total Saldo = Rp. Total Saldo Telah Dicarikan = Rp.				

Gambar 3. 52 Desain Output Saldo Pada Pemilik

15. Tampilan Halaman List Kos Pencari

Header	
Logo	List kost
Dashboard Pasang Iklan Kost Help Pembayaran Cek status Logout	▼ Records per page Search : <input type="text"/>
	Nama kost (varchar25) Jenis Kos : (varchar 20)
	gambar
	Fasilitas (text)
	Jumlah Kamar : (int3) Kamar yang tersedia : (int3)
	1 Bulan :(int11) 6 Bulan :(int9) Pertahun :(int9) 3 Bulan :(int9)
	Lihat Kamar Booking

Gambar 3. 53 Desain Output List Kos Pada Pencari

16. Tampilan Halaman Lihat Kamar

Header		
Logo	List kost	
	Nama kost (varchar25) Pemilik : (varchar20) Jenis Kos : (varchar20)	
Dashboard	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px; text-align: center;">gambar</div> Fasilitas (text) Jumlah Kamar : (int3) Kamar yang tersedia : (int3) 1 Bulan :(int11) 6 Bulan :(int9) Pertahun :(int9) 3 Bulan :(int9) <div style="text-align: right; margin-right: 50px;">Booking</div>	
Pasang Iklan		
Kost		
Help		
Pembayaran		
Cek status		
Logout		
		<input type="text" value="icon"/> (varchar12) <input type="text" value="icon"/> Alamat : (varchar30) @Email : (varchar30)

Gambar 3. 54 Desain Output Lihat Kamar

17. Tampilan Halaman Help

Header		
Logo	Help	
	Panduan Booking Kos	
Dashboard	<div style="text-align: center; padding: 20px;">Conten</div>	
Pasang Iklan		
Kost		
Help		
Pembayaran		
Cek Status		
Login		
		Panduan Pemilik Kos
		Conten
		Warning
	Conten	

Gambar 3. 55 Desain Output Panduan Penggunaan Aplikasi

18. Tampilan Bukti Pemesanan

Header	
Logo	Data Booking
	Tanggal Transaksi : 99-99-9999 No. Transaksi : (varchar20) Nomor KTP : (varchar20) Nama : (varchar40) Jenis Kelamin : (varchar30) No Telepon : (varchar13) Pekerjaan : (varchar50) Chek-in : 99-99-9999 Harga Kost : (int11) Status Transaksi : (varchar35) Total Bayar : (int20)
Dashboard	Conten Nama kost (varchar25) Pemilik : (varchar20) Jenis Kos : (varchar20) <div style="border: 1px solid black; width: 100px; height: 30px; margin: 5px auto; text-align: center;">gambar</div> Fasilitas (text) Jumlah Kamar : (int3) Kamar yang tersedia : (int3) 1 Bulan :(int11) 6 Bulan :(int9) Pertahun :(int9) 3 Bulan :(int9)
Pasang Iklan	
Kost	
Help	
Pembayaran	
Cek status	
Logout	<input type="text" value="icn"/> (varchar12) <input type="text" value="icn"/> Alamat : (varchar50)

Gambar 3. 56 Desain Output Bukti Pemesanan

G. Desain File

1. Tabel Admin

Tabel ini digunakan untuk menyimpan data admin yang telah di inputkan dapat dilihat pada table 3.2.

Database : ekost.sql

Tabel : admin

Field Key : id_admin

Tabel 3. 2 Tabel Admin

No.	Field Name	Type	Width	Description
1.	id_admin	Int	5	Id admin
2.	nama_admin	Varchar	40	Nama admin
3	hp	Varchar	12	Nomor telpon

4	username	Varchar	20	Username
5	password	Varchar	60	Password

2. Tabel Kos

Tabel ini digunakan untuk menyimpan data kos yang diinputkan oleh pemilik kos dan dapat dilihat pada tabel 3.3.

Database : ekost.sql

Tabel : Kost

Field Key : id_kost

Tabel 3. 3 Tabel Kos

No	Field Name	Type	Width	Description
1.	id_kost	Int	5	Id kos
2.	id_pemilik	Int	5	Id pemilik kos
3.	nama_kos	Varchar	40	Nama kos
4.	alamat	Varchar	50	Alamat kos
5.	latitude	Float		Latitude
6.	longitude	Float		Longitude
7.	tersedia	Int	2	Tersedia
8.	status	Enum		Status kos
9.	fasilitas	Text		Fasilitas kos
10.	jumlah	Int	3	Jumlah kamar
11.	jumlah_tersedia	Int	3	Kamar yang tersedia
12.	harga_1bulan	Int	11	Harga per 1 bulan
13.	harga_3bulan	Int	9	Harga per 3 bulan
14.	harga_6bulan	Int	9	Harga per 6 bulan
15.	harga_pertahun	Int	9	Harga per tahun

16.	gambar	Varchar	100	Gambar rumah kos
17.	gambar1	Varchar	100	Gambar kamar kos

3. Tabel Konfirmasi

Tabel ini digunakan untuk mengkonfirmasi data pemesanan kos dapat dilihat pada tabel 3.4.

Database : ekost.sql

Tabel : konfirmasi

Field Key : Id_konf

Tabel 3. 4 Tabel Id Konfirmasi

No	Field Name	Type	Width	Description
1.	id_konf	Integer	5	Id konfirmasi
2.	tanggal_b	Date		Tanggal bayar
3.	no_transaksi	Varchar	20	No transaksi
4.	bank	Varchar	35	Bank
5	atas_nama	Varchar	40	Atas nama
6	norek	Varchar	20	Norek
7	pesan	Text		Pesan

4. Tabel Pemilik Kos

Tabel ini digunakan untuk menyimpan data pemilik kos yang dapat dilihat oleh admin dan dapat dilihat pada tabel 3.5.

Database : ekost.sql

Tabel : Pemilik

Field Key : id_Pemilik

Tabel 3. 5 Tabel Pemilik Kos

No	Field Name	Type	Width	Description
1.	id_pemilik	Int	11	Id pemilik
2.	Nama	Varchar	20	Nama
3.	Alamat	Varchar	30	alamat
4.	Telepon	Varchar	12	telepon

5	Email	Varchar	30	email
6	Username	Varchar	20	Username
7	Password	Varchar	60	Password
8	Photo	Varchar	100	Photo

5. Tabel Pencari Kos

Tabel ini digunakan untuk menyimpan data pemesanan kost bagi pengunjung yang dapat dilihat pada tabel 3.6.

Database : ekost.sql

Tabel : Pengunjung

Field Key : id_pengunjung

Tabel 3. 6 Tabel Pencari Kos

No	Field Name	Type	Width	Description
1.	id_pengunjung	Int	5	Kode kategori
2.	ktp	Vachar	20	Ktp
3	nama	Vachar	40	Nama
4	jenis_kelamin	Varchar	30	Jenis kelamin
5	alamat	Text		Alamat
6	email	Varchar	20	Email
7	pekerjaan	Varchar	50	Pekerjaan
8	hp	Varchar	13	Hp
9	username	Varchar	12	Username
10	password	Varchar	12	Password

6. Tabel Rekening

Tabel ini digunakan untuk menyimpan data rekening pemilik kos yang dapat dilihat pada tabel 3.7.

Database : ekost.sql

Tabel : rekening

Field Key : id_rek

Tabel 3. 7 Tabel Rekening

No	Field Name	Type	Width	Description
1.	id_rek	Int	11	Id rekening
2.	id_pemilik	Int	11	Id pemilik
3.	bank	Varchar	35	Bank
4.	an	Varchar	50	Atas nama
5	rekening	Varchar	30	Rekening

7. Tabel Saldo

Tabel ini digunakan untuk menyimpan data saldo yg dimiliki pemilik kos dan dapat dilihat pada tabel 3.8.

Database : ekost.sql

Tabel : Saldo

Field Key : id_saldo

Tabel 3. 8 Tabel Saldo

No	Field Name	Type	Width	Description
1.	id_saldo	Int	11	Id saldo
2.	id_pemilik	Int	11	Id pemilik
3	tanggal	Date		Tanggal
4	saldo	Int	11	Saldo
5	status_sal	Varchar	35	Status saldo

8. Tabel Transaksi

Tabel ini digunakan untuk menyimpan data transaksi yang dilakukan oleh pencari kos dan dapat dilihat pada tabel 3.9.

Database : ekost.sql

Tabel : Transaksi

Field Key : no_transaksi

Tabel 3. 9 Tabel Transaksi

No	Field Name	Type	Width	Description
1.	no_transaksi	Integer	11	No transaksi
2.	tanggal	Date		Tanggal
3.	cek_in	Date		Cek in
4.	id_pemilik	Int	11	Id pemilik
5	id_kost	Int	11	Id kost
6	id_pengunjung	Varchar	20	Id pengunjung
7	harga	Int	11	Harga
8	status	Varchar	35	Status

BAB IV

PENUTUP

A. Kesimpulan

Setelah melakukan pengujian sistem yang telah dirancang, dan beberapa analisa dari sistem tersebut, penulis menarik kesimpulan sebagai berikut:

1. Aplikasi sistem informasi rumah kos dirancang dan dibuat untuk menyediakan berbagai informasi bagi pencari kos sehingga dapat melakukan pemesanan dan pembayaran kos secara online.
2. Aplikasi sistem informasi rumah kos dirancang berguna bagi pemilik kos untuk mengelola data-data yang berkaitan dengan rumah kos.

B. Saran

Untuk tercapainya efektifitas dan efisiensi kerja dengan sistem informasi rumah kos ini, maka dalam meningkatkannya berikut penulis menganjurkan beberapa saran-saran antara lain :

1. Cara pemesanan kos yang biasanya dilakukan selama ini yang kurang efektif mulai beralih pada sistem informasi rumah kos yang sudah online yang bisa digunakan diwebsite dan android agar masalah yang dihadapi dalam kesulitan pemesanan kamar selama ini dapat segera teratasi.
2. Disarankan juga pada pemilik kos terlebih dahulu dilakukan penyesuaian dan melakukan training pada pemilik kos yang menjadi inti data yang akan digunakan dalam aplikasi ini.

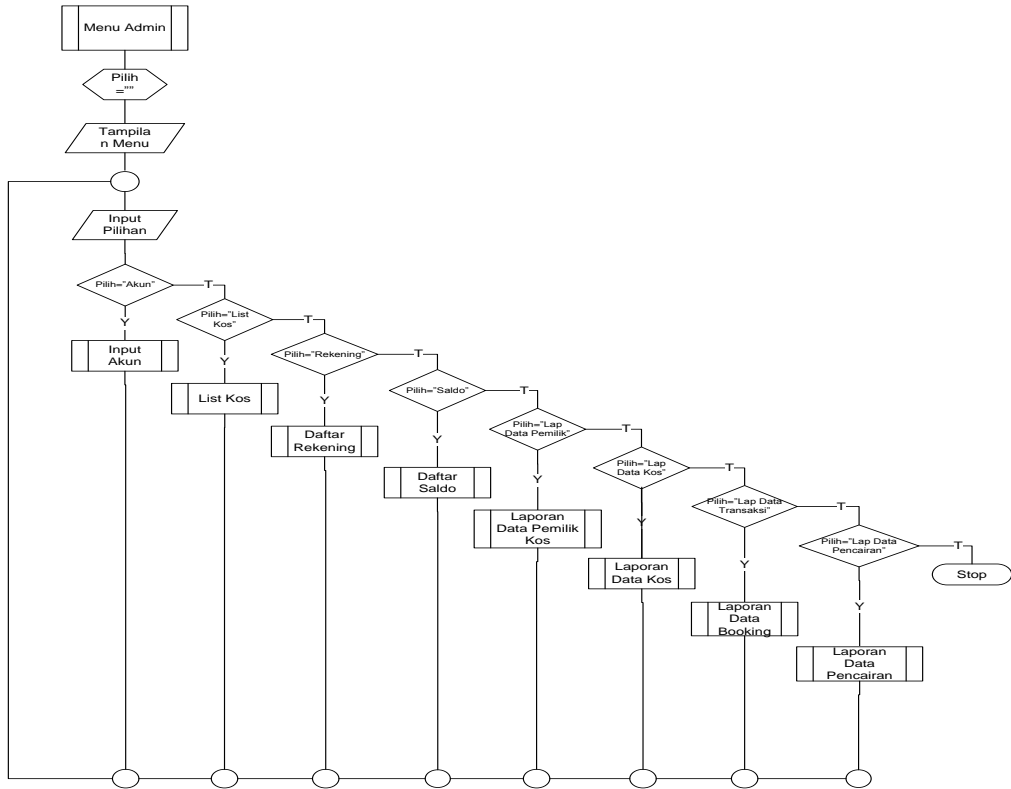
DAFTAR PUSTAKA

- Faisal, M. 2008. *Sistem Informasi Manajemen: Jaringan*. Sukses Offset. Yogyakarta.
- Wahyono, T. 2004. *Sistem Informasi: Konsep Dasar Analisis dan Desain dan Implementasi*. Graha Ilmu. Yogyakarta.
- Nugroho, A. 2010. *Analisis Dan Perancangan Sistem Informasi Dengan Metodologi Berorientasi Objek*. Informatika. Bandung.
- Safaat, Nazruddin. 2014. *Android: Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Informatika. Bandung.
- Komala, Reno. 2015. *Sistem Pakar Menentukan Jenis Rumah Berdasarkan Ciri-Ciri Perancangan Rumah Pada Perumahan Cv. Taruko Usaha Jaya Berbasis Web Dengan Metode Forward Chaining* (Tugas Akhir). Prodi Manajemen Informatika, IAIN Batusangkar.
- Kurniawan, Zaini. 2016. *Sistem Informasi Geografis Pencarian Lokasi Penginapan Dikota Bukittinggi* (Tugas Akhir). Jurusan Manajemen Informaika, IAIN Batusangkar.
- Prawita, Riska. 2016. *Aplikasi Sistem Pakar Pengukuran Self Esteem Dan Locus Of Control Untuk Siswa Pada SMK N 1 Batusangkar Berbasis Web* (Tugas Akhir). Jurusan Manajemen Informaika, IAIN Batusangkar.
- Rami, Amelia. 2017. *Perancangan Sistem Informasi Laboratorium Unit Transfusi Darah (UTD) Pada RSUD Prof.Dr.M.A Hanafiah Kota Batusangkar Berbasis Web* (Tugas Akhir). Jurusan Manajemen Informaika, IAIN Batusangkar.

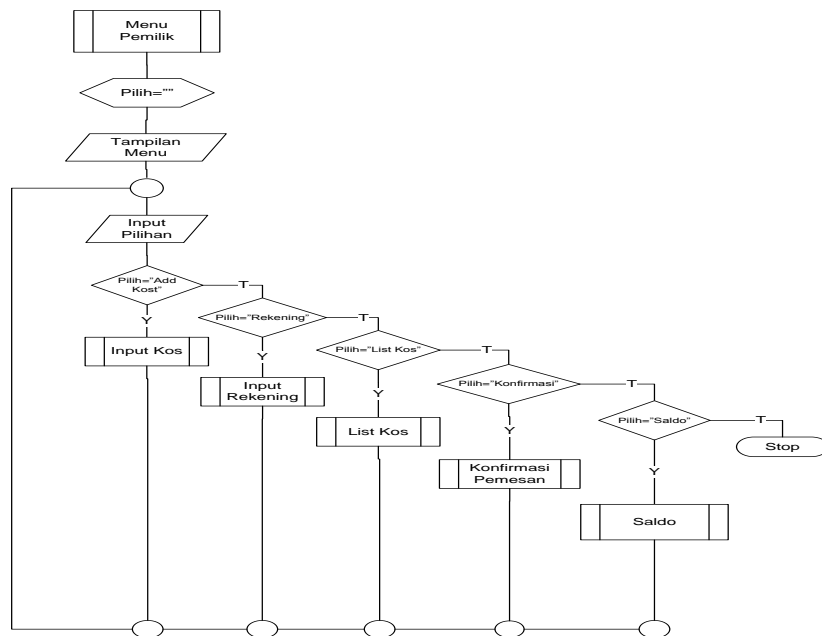
FLOWCHART

FLOWCHART

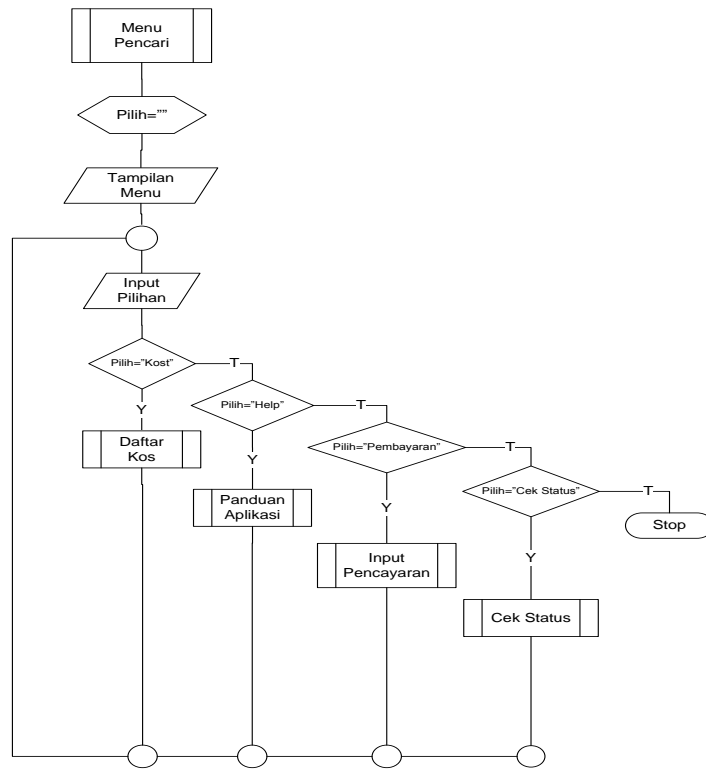
1. Flow Chart Menu Admin



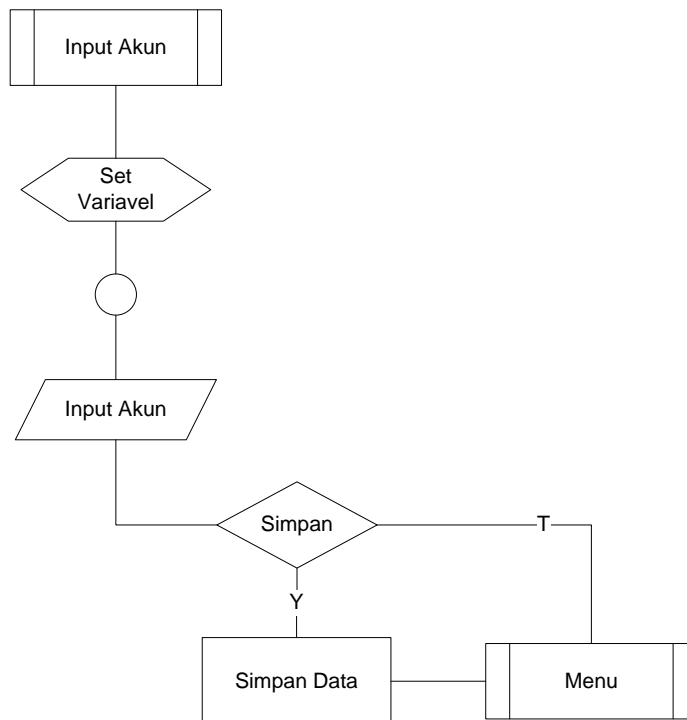
2. Flow Chart Menu Pemilik



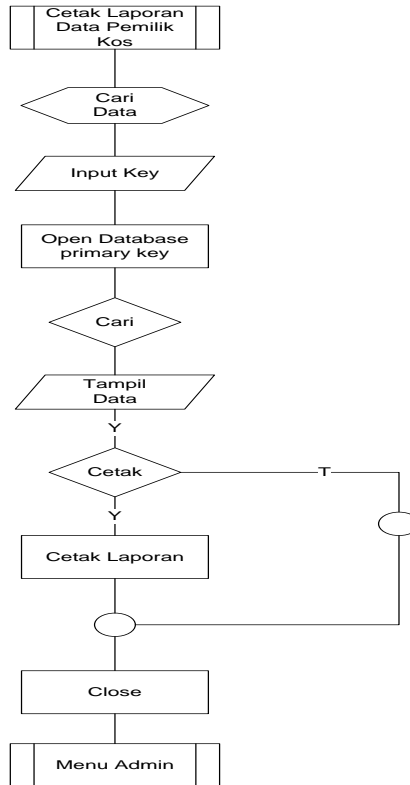
3. Flow Chart Menu Pencari



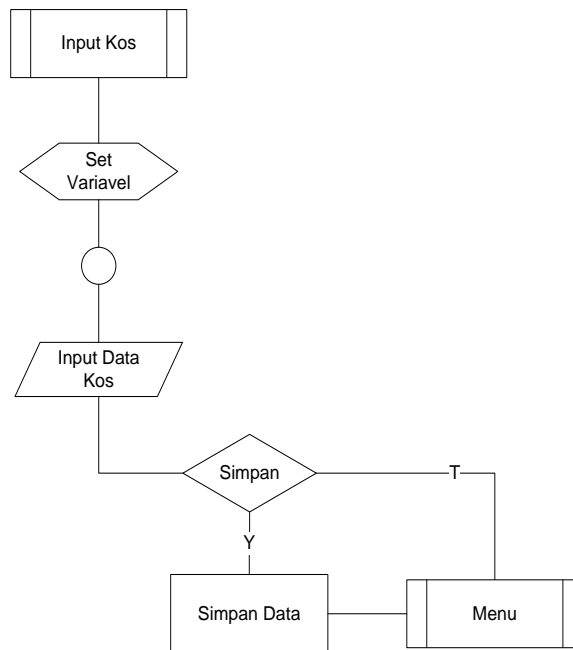
4. Flow Chart Input Akun



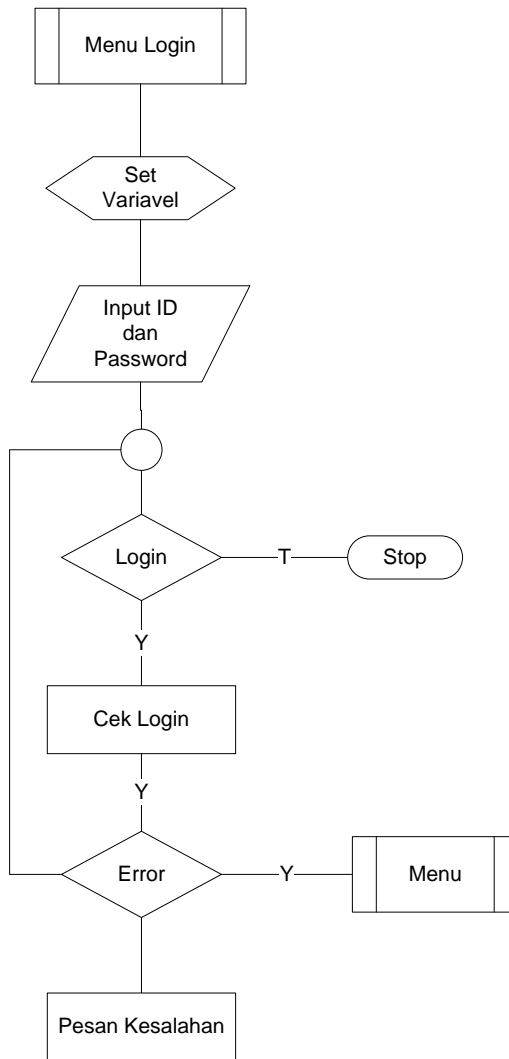
5. Flow Chart Cetak Laporan Data Pemilik Kos



6. Flow Chart Add Kos



7. Flow Chart Login



LISTING PROGRAM

LISTING PROGRAM

Script login

```
<?php
if($_POST['simpan']=="simpan"){
error_reporting(E_ALL ^ (E_NOTICE | E_WARNING));
include "koneksi.php";
$login=mysql_query("SELECT * FROM pemilik WHERE
username='".$_POST['user']."' AND password='".$_POST['pass']."'");
$ketemu=mysql_num_rows($login);
$r=mysql_fetch_array($login);
// Apabila username dan password ditemukan
if ($ketemu > 0){
    session_start();
    $_SESSION['id_pemilik'] = $r['id_pemilik'];
    $_SESSION['username'] = $r['username'];
    $_SESSION['nama'] = $r['nama'];
    $_SESSION['password'] = $r['password'];
    $_SESSION['telepon'] = $r['telepon'];
    $_SESSION['email'] = $r['email'];
    $_SESSION['alamat'] = $r['alamat'];
    $_SESSION['photo'] = $r['photo'];
    echo "<meta http-equiv='refresh' content='0; pemilik.php'>";
}
else{
    echo "<script type='text/javascript'>
        alert('Password dan User Name salah...!!!');
        </script>";
    echo "<meta http-equiv='refresh' content='0; login.php'>";
}
}
?>
```

Script daftar

```
<?php
if($_POST['simpan']=="simpan"){
include "koneksi.php";
$nama_file = $_FILES['file_gambar']['name'];
$lokasi_file = $_FILES['file_gambar']['tmp_name'];
if(! empty($lokasi_file)){
move_uploaded_file($lokasi_file, "pemilik/$nama_file");
$sql = "insert into pemilik values(NULL,
    '".$_POST['nama']."',
    '".$_POST['alamat']."',
    '".$_POST['tlp']."',
    '".$_POST['email']."',
    '".$_POST['user']."',
```

```

    "".$_POST['pass'].",
    "".$nama_file."");
$result = mysql_query($sql);
if ($result){
echo "<script type='text/javascript'>
alert('Pendaftaran Berhasil...!!!');
</script>";
echo "<meta http-equiv='refresh' content='0; index.php'>";
}else{
echo "<script type='text/javascript'>
alert('Data Ada yang kurang...!!!');
</script>";
echo "<meta http-equiv='refresh' content='0; daftar.php'>";
}
}
else
{
$sql = "insert into pemilik values(NULL,
"".$_POST['nama'].",
"".$_POST['alamat'].",
"".$_POST['tlp'].",
"".$_POST['email'].",
"".$_POST['user'].",
"".$_POST['pass'].",'-)";
$result = mysql_query($sql);
if ($result){
echo "<script type='text/javascript'>
alert('Pendaftaran Berhasil...!!!');
</script>";
echo "<meta http-equiv='refresh' content='0; index.php'>";
}else{
echo "<script type='text/javascript'>
alert('Data Ada yang kurang...!!!');
</script>";
echo "<meta http-equiv='refresh' content='0; daftar.php'>";
}
}
?>
Script add kos
<?php
error_reporting(E_ALL ^ (E_NOTICE | E_WARNING));
session_start();
if($_POST['simpan']=="simpan"){
include "koneksi.php";
$nama_file = $_FILES['file_gambar']['name'];
$lokasi_file = $_FILES['file_gambar']['tmp_name'];

```

```

if(! empty($lokasi_file)){
move_uploaded_file($lokasi_file, "kos/$nama_file");
$sql = "insert into kost values(NULL,
".$_SESSION['id_pemilik'].",
".$_POST['nama'].",
".$_POST['alamat'].",
0,
0,
'1',
".$_POST['jns'].",
".$_POST['fas'].",
".$_POST['jml'].",
".$_POST['hrg1'].",
".$_POST['hrg2'].",
".$_POST['hrg3'].",
".$_POST['hrg4'].",
".$nama_file.");";
$result = mysql_query($sql);
if ($result){
echo "<script type='text/javascript'>
alert(' Berhasil...!!!');
</script>";
echo "<meta http-equiv='refresh' content='0; add_kos.php'>";
}else{
echo "<script type='text/javascript'>
alert('Data Ada yang kurang...!!!');
</script>";
echo "<meta http-equiv='refresh' content='0; add_kos.php'>";
}
}
else
{
$sql = "insert into kost values(NULL, ".$_SESSION['id_pemilik'].",
".$_POST['nama'].",
".$_POST['alamat'].",
0,
0,
'1',
".$_POST['jns'].",
".$_POST['fas'].",
".$_POST['jml'].",
".$_POST['hrg1'].",
".$_POST['hrg2'].",
".$_POST['hrg3'].",
".$_POST['hrg4'].",
'-)";

```

```

$result = mysql_query($sql);
if ($result){
echo "<script type='text/javascript'>
alert('Berhasil...!!!');
</script>";
echo "<meta http-equiv='refresh' content='0; add_kos.php'>";
}else{
echo "<script type='text/javascript'>
alert('Data Ada yang kurang...!!!');
</script>";
echo "<meta http-equiv='refresh' content='0; add_kos.php'>";
}
}
?>

```

Script kos

```

<?php
error_reporting(E_ALL ^ (E_NOTICE | E_WARNING));
session_start();
include "koneksi.php";
?>
<?php include "header.php";?>
<?php include "menu.php";?>
<div id="page-wrapper" >
<div id="page-inner">
<div class="row">
<div class="col-md-12">
<h2>List Kost</h2>
<h5>&nbsp;</h5>
</div>
</div>
<!-- /. ROW -->
<hr />
<div class="row">
<div class="col-md-12">
<!-- Advanced Tables -->
<div class="panel-body">
<div class="table-responsive">
<table class="table table-striped table-bordered table-hover" id="dataTables-
example">
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>

```



```

mysql_query ("insert into pengunjung values('".$_POST['id']."',
".$_POST['ktp']."',
".$_POST['nama']."',
".$_POST['j_kel']."',
".$_POST['alamat']."',
".$_POST['email']."',
".$_POST['kerja']."',
".$_POST['telp']."',
".$_POST['user']."',
".$_POST['pass']."')");
$result = mysql_query($sql);
if ($result){
echo "<meta http-equiv='refresh' content='0;
view_transaksi.php?id=".$_POST[idt]>";
}else{
echo "<script type='text/javascript'>
alert('Data Ada yang kurang...!!!');
</script>";
echo "<meta http-equiv='refresh' content='0; kos.php'>";
}
}
?>
<?php include "header.php";?>
<?php include "menu.php";?>
<div id="page-wrapper" >
<div id="page-inner">
<div class="row">
<div class="col-md-12">
<h2>Forms Booking</h2>
</div>
</div>
<!-- /. ROW -->
<hr />
<div class="row">
<div class="col-md-12">
<!-- Form Elements -->
<div class="panel panel-default">
<div class="panel-heading">
Form Booking</div>
<div class="panel-body">
<div class="row">
<div class="col-md-6">
<?php // membuat query max untuk kode barang
include "koneksi.php";
$carikode = mysql_query("select max(id_pengunjung) from pengunjung") or die
(mysql_error());

```

```

// menjadikannya array
$datakode = mysql_fetch_array($carikode);
// jika $datakode
if ($datakode) {
// membuat variabel baru untuk mengambil kode barang mulai dari 1
$nilaikode = substr($datakode[0], 1);
// menjadikan $nilaikode ( int )
$kode = (int) $nilaikode;
// setiap $kode di tambah 1
$kode = $kode + 1;
// hasil untuk menambahkan kode
// angka 3 untuk menambahkan tiga angka setelah B dan angka 0 angka yang
berada di tengah
// atau angka sebelum $kode
$hasilkode = "P".str_pad($kode, 6, "0", STR_PAD_LEFT);
} else {
$hasilkode = "P0001";
}
?>
<?php // membuat query max untuk kode barang
$carikodetamu = mysql_query("select max(no_transaksi) from transaksi") or die
(mysql_error());
// menjadikannya array
$datakodetamu = mysql_fetch_array($carikodetamu);
// jika $datakode
if ($datakodetamu) {
// membuat variabel baru untuk mengambil kode barang mulai dari 1
$nilaikodetamu = substr($datakodetamu[0], 1);
// menjadikan $nilaikode ( int )
$kodetamu = (int) $nilaikodetamu;
// setiap $kode di tambah 1
$kodetamu = $kodetamu + 1;
// hasil untuk menambahkan kode
// angka 3 untuk menambahkan tiga angka setelah B dan angka 0 angka yang
berada di tengah
// atau angka sebelum $kode
$hasilkodetamu = "0".str_pad($kodetamu, 4, "0", STR_PAD_LEFT);
} else {
$hasilkodetamu = "0001";
}
?>
<?php
$wis = mysql_query("SELECT * FROM `kost`, pemilik
WHERE kost.`id_pemilik`=pemilik.`id_pemilik`
and kost.tersedia='1' and kost.id_kost='".$_$_GET['id'].'"");
$r mysql_fetch_array($wis);

```

```

$harga1 = number_format($r['harga_1bulan'],0,"",".");
$harga2 = number_format($r['harga_6bulan'],0,"",".");
$harga3 = number_format($r['harga_pertahun'],0,"",".");
$harga4 = number_format($r['harga_3bulan'],0,"",".");
?>
<form action="booking.php" method="post" enctype="multipart/form-data"
role="form" onSubmit="return validasi(this)">
<div class="form-group">
<label>Nama Kost </label><br>
<?php echo $r['nama_kos']; ?>
</div>
<div class="form-group">
<label>Pemilik</label><br>
<?php echo $r['nama']; ?>
</div>
<div class="form-group">
<label>Cek In</label><br>
<?php
echo "<select name='tanggal' class='form-control'>";
echo "<option value=0 selected>Tanggal</option>";
for($tgl=01;$tgl<=31;$tgl++)
{
echo "<option value=$tgl>$tgl</option>";
}
echo "<select>";
$nm_bulan=array(1 =>"Januari", "Februari", "Maret", "April", "Mei", "Juni",
"Juli", "Agustus", "September", "Oktober", "November", "Desember");
echo "<select name='bulan' class='form-control'>";
echo "<option value=0 selected>Bulan</option>";
for($bln=1;$bln<=12;$bln++)
{
echo "<option value=$bln>$nm_bulan[$bln]</option>";
}
echo "</select>";
$thn_skrng = date('Y');
echo "<select name='tahun' class='form-control'>";
echo "<option value=0 selected>Tahun</option>";
for($thn=1990;$thn<=$thn_skrng; $thn++)
{
echo "<option value=$thn>$thn</option>";
}
echo "</select>";
?>

```




INSTITUT AGAMA ISLAM NEGERI
BATUSANGKAR

PROGRAM DIII MANAJEMEN INFORMATIKA

Jl. Sudirman No. 137 Kubu Raja Lima Kaum Batusangkar 27213 Telp. (0752) 71150, 574221, Fax. (0752) 71879
<http://www.stainbatusangkar.ac.id> e-mail: mi@stainbatusangkar.ac.id

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

KARTU BIMBINGAN PENULISAN TUGAS AKHIR

Nim : 14 205 121
Nama : YESI WIRNA ELIZA
Jurusan : D.III MANAJEMEN INFORMATIKA
Dosen Pembimbing : Iswandi, M.Kom
Judul Tugas Akhir : Sistem Informasi Rumah Kos Disekitar Kampus IAIN Batusangkar
Berbasis Android

NO	Hari/Tanggal	Materi Bimbingan	Paraf
1.	31-5-2017	Konsultasi BAB I dan II Lanjutkan BAB III	[Signature]
2.	6-7-2017	Konsultasi BAB I, II, III Lanjutkan Program	[Signature]
3.	15-7-2018	Konsultasi Rancangan Program	[Signature]
4.	16-1-2018	Konsultasi Perbaikan rancangan aplikasi	[Signature]
5.	24-1-2018	Perbaikan Rancangan O/R/A	[Signature]
6.	29-1-2018	Konsultasi perbaikan program Aplikasi	[Signature]
7.	2-2-2018	Lengkapi dgn Lampiran	[Signature]
8.	5-2-2018	Acc Agenda	[Signature]

Catatan : Setiap konsultasi dengan dosen pembimbing kartu ini harap dibawa, diisi, dan diparaf oleh dosen pembimbing

Batusangkar, 05 FEBRUARI 2018
Mahasiswa

Yesi Wirna Eliza
Nim. 14 205 121

Tanda Tangan,
Dosen Pembimbing Tugas Akhir

Mengetahui,
Dosen Penasehat Akademik

Iswandi, M.Kom

Yusrizal Etendi, S.A.G., M.